

AVGRADERT  
Dato: 10.01.2019  
Sign: Bjørn Berg (digitalt)

**BEGRENSET**  
iht sikkerhetsloven § 11 og § 12  
jf offentleglova § 13

KUNDEKONFIDENSIELL

# Rapport

## Arkitektur- og kildekodegjennomgang EVA Admin

### Valgdirektoratet

Versjon 1.0

13.6.2018

#### 1 Sammendrag

mnemonic har gjennomført en kodegjennomgang av de ulike delene av EVA applikasjonen. Denne rapporten beskriver funn og forbedringsforslag for EVA Admin. Denne rapporten er hovedsakelig basert på funn fra analyseverktøyet MicroFocus Fortify SCA.

#### 2 Om deloppdraget

For denne komponenten fikk vi i oppstartsmøtet følgende ønske om spissing av oppdraget:

BEGRENSET

- Sikring gjennom infrastruktur
- Applikasjonssikkerhet
- Kodekvalitet

Analyse fra kildekodeverktøy har tatt mesteparten av tiden for dette oppdraget. Vi har i tillegg sett på kodestruktur, delvis da dette ble en utfordring da vi gikk gjennom funn fra kildekodeverktøy.

## 3 Generelt om applikasjonen

Dette kapittelet inneholder observasjoner knyttet til kvalitet og arkitektur. Dette er basert både på vår jobbing med koden og få den til å kompilere og bygge, samt dypere analyser av strukturen og potensielle problemer vi ser med tanke på vedlikehold, videreutvikling og rene sikkerhetsaspekter.

### 3.1 Kodekvalitet og arkitektur

Vi fikk noen problemer da vi skulle bygge koden med Maven. For å kompilere koden var vi nødt til å ekskludere all testing av kode, både enhetstester og integrasjonstester. Dette kan være et tegn på for sterk kobling mellom koden i seg selv og testrammeverket. Generelt kan sies at enhetstester bør lages slik at de ulike kodeenhetene testes hver for seg uavhengig av andre enheter.

*Anbefaling 1 Se over eventuelle avhengigheter mellom applikasjons- og testkode*

Deler av koden er gjenbruk fra det gamle «evote» systemet, og det er fortsatt mange deler som ikke er refaktorisert til ny struktur (no.valg). Dette gir et forvirrende inntrykk og gjør det vanskeligere å lese koden. Videre er oppdelingen av del-prosjekter ikke helt konsekvent i forhold til «package»-strukturen i koden. Som eksempel på dette kan nevnes:

- **admin-counting** inneholder to ulike «package»-strukturer, en på norsk og en på engelsk:
  - no.valg.eva.admin.counting...
  - no.valg.eva.admin.opptelling...
- **admin-backend, admin-backend-common, admin-common og adminconfiguration** inneholder alle:
  - no.valg.eva.admin.configuration...
- **admin-common** inneholder:
  - no.valg.eva.backend

*Anbefaling 2 Alle del-prosjekter (repositories) bør være avgrenset til logiske blokker som gir mening og en konsekvent oppdeling av koden*

*Anbefaling 3 Java “package”-strukturen bør gjenspeile strukturen i koden (repositories)*

Vi ble bedt om å se på admin-anonymisering. Denne koden ser dog ikke ut til å være ryddet i på en stund; I no.evote.Tools gjøres det en throw av com.scytl.jbasis.crypto.CryptoException, men den er ikke er i bruk i no.evote.export.GenerateTenMillion, som er den eneste klassen som kalles i dag. Det er ingen scytl-kode igjen i kodebasen og vi antar at denne ikke er ryddet vekk ennå fordi det ikke har vært prioritert ennå å se på denne koden.

no.evote.export.GenerateTenMillion inkluderer også en en hjelpetekst som referer til en opsjon som ikke ser ut til å finnes (i dag):

```
157: System.out
158:     .println("\t-e NNNNNN \texport numbers from the election event with
the specified ID");
```

Vi antar at dette referer til en tidligere funksjonalitet, eller tidligere planlagt funksjonalitet. *Anbefaling*

#### *4 Fullfør rydding av evote-spesifikk funksjonalitet*

## 4 Kryptorelaterte funn

Dette kapittelet inneholder kryptofunn som er spesifikke for EVA Skanning

### 4.1 no.evote.service.CryptoServiceBean

Dette er gammel kryptografisk kode basert på BouncyCastle. Vi ser flere ting ved denne koden som virker *snodig* eller utgjør en potensiell risiko, sett med modern øyne.

```
71: private static final String AES_CBC_PKCS5PADDING = "AES/CBC/PKCS5PADDING";
72: private static final String PBEWITHMD5AND256BITAES_CBC_OPENSSL =
"PBEWITHMD5AND256BITAES-CBC-OPENSSL";
```

Bruk av AES i cipher block chaining (CBC) modus med PKCS #5 padding vil i mange tilfeller kunne åpne for såkalte «[Padding Oracle](#)» angrep – avhengig av hvorvidt og hvordan integritetssjekking er kombinert med krypteringen. I situasjoner hvor en angriper kan sende inn mange krypterte meldinger (f.eks. en tokenvalideringstjeneste) vil dette åpne for muligheten for å kryptere eller dekryptere trafikk.

Videre benytter man en algoritme for passordbasert kryptering basert på MD5 for passord-hashing. Dette er lite robust med tanke på forsøk på passordknekking. Fordelen med dette valget ser primært ut til å være kompatibilitet med OpenSSL sine krypteringsfunksjoner.

Gjeldende anbefaling er å bruke en krypteringsmodus som støtter såkalt AEAD (Authenticated Encryption with Associated Data), eksempel på dette er AES\_GCM, og å bruke en passordhashalgoritme som er mer resistent mot knekking.

*Anbefaling 5 Se på muligheten for å benytte mer moderne krypto-primitiver.*

```
110: SecureRandom random = SecureRandom.getInstance("SHA1PRNG");
```

Vi anbefaler vanligvis å initialisere SecureRandom-klassen med NativePRNG dersom man er basert på Unix eller Linux; denne benytter underliggende operativsystemmekanismer direkte, og vil normalt ha bedre sikkerhetsmekanismer enn SHA1PRNG som er en software-basert generator. Dersom koden skal kjøre på Windows er bildet noe mer nyansert, men i den grad det er mulig bør en plattform-«native» mekanisme som regel benyttes..

*Anbefaling 6 Benytt operativsystemets mekanismer for sikker slumptallgenerering*

```
114: PBEKeySpec keySpec = new PBEKeySpec(password.toCharArray(), salt, 1, 256);
```

I dette tilfellet lager koden en nøkkel basert på et kjent passord. Den 3. parameteren er i dette tilfellet en «iteration count», som sier noe om hvor mange ganger passordet skal hashes. Ved å øke antall iterasjoner, gjør man det vanskeligere å knekke passordet. Her er imidlertid antall iterasjoner 1, noe som gjør det lettere å knekke (særlig i kombinasjon med bruken av MD5 nevnt over).

Signaturvalideringskoden er i hovedsak basert på CmsDecoder-klassen (og Cryptographic Message Syntax), men vi merker oss at valideringslogikken i verifyScanningCountSignature og verifyCrlmetodene også er vanskelig å følge.

*Anbefaling 7 Sett antall iterasjoner for passordbasert kryptering til en høyere verdi*

## 4.2 no.valg.eva.admin.crypto.CmsDecoder

Dette er en nyere klasse som skal verifisere kryptografiske signaturer basert på CMS (PKCS #7 / RFC 5652 / S-MIME). Det er uklart ut fra koden om denne klassen i tilstrekkelig grad verifiserer hvilke signatur-parametere som er satt i SignerInfo – med andre ord om det er mulig å sende inn en PKCS #7 signatur med en SignatureAlgorithmIdentifiser eller andre parametere som er valgt til å være usikre, og at dette likevel vil valideres OK. Dette ser ut som noe som burde testes nærmere.

Bruken av PKCS #7 er ellers generelt sett grei – men man kan (som i flere av punktene over) vurdere å se etter mer lettvektsalternativer som dekker behovet for kryptering og signatur, men som egner seg for mindre kompleks og mer vedlikeholdbar/testbar implementasjon.

*Anbefaling 8 Verifikasjon av kryptografiske signaturer må ikke tillate usikre signeringsmekanismer (f.eks. ved å validere parametere satt i CMS SignerInfo)*

## 5 Verktøybaserte funn: komponenter med sårbarheter

Dette kapittelet inneholder funn fra OWASP Dependency Check-verktøyet. Se vedlagte rårapporter for å få mer informasjon om hvilke versjoner som ikke er sårbare, hvilke CVE (Common Vulnerability Enumeration)-sårbarheter det dreier seg om, m.m.

Følgende Java-bibliotek er i bruk i koden og har kjente sårbarheter:

- commons-httpclient:commons-httpclient:3.0 ○ **Dette bibliotek er end-of-life og bør erstattes med f.eks. Apache HttpComponents!**
- ca.juliusdavies:not-yet-commons-ssl:0.3.9
- xerces:xercesImpl:2.11.0
- com.fasterxml.jackson.core:jackson-databind:2.7.4
- commons-httpclient:commons-httpclient:3.1 ○ **Dette bibliotek er end-of-life og bør erstattes med f.eks. Apache HttpComponents!**
- org.owasp.esapi:esapi:2.0.1
- org.xerial:sqlite-jdbc:3.7.15-M1
- org.primefaces:primefaces:5.3

## 6 Verktøybaserte funn: kildekodefunn

Dette kapittelet inneholder detaljert informasjon om hvert enkelt funn, slik Fortify SCA rapporterer det.

Vi har prøvd å kvalitetssikre funnene ved å fjerne de som er falske positive. Det er dog ikke alltid lett å komme utenfra å forstå koden og det vil nok være funn som vi har tatt med, men som utviklerne vil kategorisere som falske positive. De færreste funnene er direkte utnyttbare, men vi mener at ved å følge anbefalingene vil det være mindre fare for sikkerhetsrelaterte hendelser, og vi bruker her «sikkerhet i dybden» som prinsipp. Funnene vi anser som reelle har vi kategorisert iht Fortifys 4 innebygde analysekategorier:

- Reliability Issue – Dette er svakheter som kan gjøre at applikasjonen ikke oppfører seg som den skal. Det er ofte relatert til fysiske begrensninger og gjelder gjerne filaksess, nettverkskommunikasjon, minneproblemer, o.l.
- Bad Practice – Denne kategorien inneholder både sikkerhetsrelaterte sårbarheter og kvalitetsrelaterte.
- Suspicious – Her har vi funn som vi enten er usikre på om er utnyttbare, eller som vil kunne være én av flere faktorer som skal til for at en sårbarhet oppstår, mtp sikkerhet i dybden.
- Exploitable – Dette er funn som er direkte utnyttbare. Det kan være manglende validering og outputkoding (Injection-/Cross-Site Scripting-angrep), manglende transportsikring (klartekstkommunikasjon i nettverk), eller andre angrep som vil kunne skje uten at andre

Arkitektur- og kildekodegjennomgang EVA Admin – KUNDEKONFIDENSIELL mnemonic as sikkerhetsmekanismer brytes. Det vil dog være ulikt hvor alvorlige disse er, f.eks. om det krever aksess til lokalt nettverk eller om det kan gjøres direkte fra Internett.

Et funn kan høre hjemme i flere kategorier, og det er særlig «Suspicious» hvor vi er usikre på om sårbarheten er reell eller ikke.

Funnene er sortert etter alvorlighetsgrad slik at de mest kritiske funnene kommer først. Vi har her ikke gjort noen endring av kritikalitet.

## [High] Password Management / Hardcoded Password

*Security Features*

**Analysis:** Suspicious

### Abstract

Hardcoded passwords may compromise system security in a way that cannot be easily remedied.

### Explanation

It is never a good idea to hardcode a password. Not only does hardcoding a password allow all of the project's developers to view the password, it also makes fixing the problem extremely difficult. Once the code is in production, the password cannot be changed without patching the software. If the account protected by the password is compromised, the owners of the system will be forced to choose between security and availability.

**Example 1:** The following code uses a hardcoded password to connect to a database:

```
...  
DriverManager.getConnection(url, "scott", "tiger");  
...
```

This code will run successfully, but anyone who has access to it will have access to the password. Once the program has shipped, there is likely no way to change the database user "scott" with a password of "tiger" unless the program is patched. An employee with access to this information could use it to break into the system. Even worse, if attackers have access to the bytecode for the application they can use the `javap -c` command to access the disassembled code, which will contain the values of the passwords used. The result of this operation might look something like the following for the example above:

```
javap -c ConnMngr.class  
22: ldc #36; //String jdbc:mysql://ixne.com/rxsql  
24: ldc #38; //String scott  
26: ldc #17; //String tiger
```

In the mobile world, password management is even trickier, considering a much higher chance of device loss. **Example 2:** The code below uses hardcoded username and password to setup authentication for viewing protected pages with Android's WebView.

```
... webView.setWebViewClient(new WebViewClient() {  
public void onReceivedHttpAuthRequest(WebView view,  
HttpAuthHandler handler, String host, String realm) {  
handler.proceed("guest", "allow");  
}  
});  
...
```

Similar to Example 1, this code will run successfully, but anyone who has access to it will have access to the password.

### Recommendation

Passwords should never be hardcoded and should generally be obfuscated and managed in an external source. Storing passwords in plaintext anywhere on the system allows anyone with sufficient

Arkitektur- og kildekodegjennomgang EVA Admin – KUNDEKONFIDENSIELL mnemonic as permissions to read and potentially misuse the password. At the very least, passwords should be hashed before being stored.

Some third-party products claim the ability to manage passwords in a more secure way. For example, WebSphere Application Server 4.x uses a simple XOR encryption algorithm for obfuscating values, but be skeptical about such facilities. WebSphere and other application servers offer outdated and relatively weak encryption mechanisms that are insufficient for security-sensitive environments. For a secure generic solution, the best option today appears to be a proprietary mechanism that you create.

For Android, as well as any other platform that uses SQLite database, a good option is SQLCipher -- an extension to SQLite database that provides transparent 256-bit AES encryption of database files. Thus, credentials can be stored in an encrypted database.

**Example 3:** The code below demonstrates how to integrate SQLCipher into an Android application after downloading the necessary binaries, and store credentials into the database file.

```
import net.sqlcipher.database.SQLiteDatabase;
...
    SQLiteDatabase.loadLibs(this);
    File dbFile =
getDatabasePath("credentials.db");
dbFile.mkdirs(); dbFile.delete();
    SQLiteDatabase db = SQLiteDatabase.openOrCreateDatabase(dbFile, "credentials", null);
db.execSQL("create table credentials(u, p)");
    db.execSQL("insert into credentials(u, p) values(?, ?)", new Object[]{username, password});
...

```

Note that references to `android.database.sqlite.SQLiteDatabase` are substituted with those of `net.sqlcipher.database.SQLiteDatabase`.

To enable encryption on the WebView store, WebKit has to be re-compiled with the `sqlcipher.so` library.

## 6.1 JasperserverRestClientProducer.java

*Merk:* Vi vet at EVA Admins' infrastruktur er ansett som sikker, men vi sliter når vi ser passord = brukernavn. Det anser vi som like svakt som å ikke bruke password, og vi vil gjerne at det er et bevisst valg å ha et slikt svakt passord. Særlig da det virker enkelt å sette et sterkt passord, et som forhåpentlig tilfredsstillende Valgdirektoratets passordpolicy.

**Issue** 41B093EC69ED39588F8FD8EC0636D630:

**Source :**

**File** admin-backend/src/main/java/no/valg/eva/admin/backend/reporting/jasperserver/api/JasperserverRestClientProducer.java 42

**Fact** Location: this.password

```
39|private static final Logger LOGGER = Logger.getLogger(JasperserverRestClientProducer.class); 40|
41|private String username = "jasperadmin";
>42|private String password = "jasperadmin";
43|private String baseUrl = "http://localhost:8081/";
44|private String context = "jasperserver";
45|private long timeout;
```

**Sink :**

**File** admin-backend/src/main/java/no/valg/eva/admin/backend/reporting/jasperserver/api/JasperserverRestClientProducer.java 42

**Fact** Location: this.password

```
39|private static final Logger LOGGER = Logger.getLogger(JasperserverRestClientProducer.class); 40|
41|private String username = "jasperadmin";
>42|private String password = "jasperadmin";
43|private String baseUrl = "http://localhost:8081/";
44|private String context = "jasperserver";
45|private long timeout;
```

## [High] XML Entity Expansion Injection

*Input Validation and Representation*

## Abstract

Using XML parsers configured to not prevent nor limit Document Type Definition (DTD) entity resolution can expose the parser to an XML Entity Expansion injection

## Explanation

XML Entity Expansion injection also known as XML Bombs are Denial Of Service (DoS) attacks that benefit from valid and well-formed XML blocks that expand exponentially until they exhaust the server allocated resources. XML allows to define custom entities which act as string substitution macros. By nesting recurrent entity resolutions, an attacker may easily crash the server resources.

The following XML document shows an example of an XML Bomb.

```
<?xml version="1.0"?>
<!DOCTYPE lolz [
  <!ENTITY lol "lol">
  <!ENTITY lol2 "&lol;&lol;&lol;&lol;&lol;&lol;&lol;&lol;&lol;">
  <!ENTITY lol3 "&lol2;&lol2;&lol2;&lol2;&lol2;&lol2;&lol2;&lol2;&lol2;">
  <!ENTITY lol4 "&lol3;&lol3;&lol3;&lol3;&lol3;&lol3;&lol3;&lol3;&lol3;">
  <!ENTITY lol5 "&lol4;&lol4;&lol4;&lol4;&lol4;&lol4;&lol4;&lol4;&lol4;">
  <!ENTITY lol6 "&lol5;&lol5;&lol5;&lol5;&lol5;&lol5;&lol5;&lol5;&lol5;">
  <!ENTITY lol7 "&lol6;&lol6;&lol6;&lol6;&lol6;&lol6;&lol6;&lol6;&lol6;">
  <!ENTITY lol8 "&lol7;&lol7;&lol7;&lol7;&lol7;&lol7;&lol7;&lol7;&lol7;">
  <!ENTITY lol9 "&lol8;&lol8;&lol8;&lol8;&lol8;&lol8;&lol8;&lol8;&lol8;">
]
</lolz>&lol9;</lolz>
```

This test could crash the server by expanding the small XML document into more than 3GB in memory.

## Recommendation

An XML parser should be configured securely so that it does not allow document type definition (DTD) custom entities as part of an incoming XML document.

To avoid XML Entity Expansion injection the "secure-processing" property should be set for an XML factory, parser or reader:

```
factory.setFeature("http://javax.xml.XMLConstants/feature/secure-processing", true);
```

In JAXP 1.3 and earlier versions, when the secure processing feature is on, default limitations are set for DOM and SAX parsers. These limits are:

```
entityExpansionLimit = 64,000
elementAttributeLimit = 10,000
```

Since JAXP 1.4, the secure processing feature is turned on by default. In addition to the above limits, a new maxOccur limit is added to the validating parser. The limit is:

```
maxOccur = 5,000
```

If inline DOCTYPE declaration is not needed, it can be completely disabled with the following property:

```
factory.setFeature("http://apache.org/xml/features/disallow-doctype-decl", true);
```

To protect a TransformerFactory, the following feature should be set:

```
TransformerFactory transFact = TransformerFactory.newInstance();
transFact.setFeature(XMLConstants.FEATURE_SECURE_PROCESSING, true);
Transformer trans = transFact.newTransformer(xsltSource);
trans.transform(xmlSource, result);
```

Alternatively, it is possible to use a securely configured XMLReader to set the transformation sources:

```
XMLReader reader = XMLReaderFactory.createXMLReader();
reader.setFeature("http://apache.org/xml/features/disallow-doctype-decl", true);
Source xmlSource = new SAXSource(reader, new InputSource(new FileInputStream(xmlFile)));
Source xsltSource = new SAXSource(reader, new InputSource(new FileInputStream(xsltFile)));
Result result = new StreamResult(System.out);
TransformerFactory transFact = TransformerFactory.newInstance();
Transformer trans = transFact.newTransformer(xsltSource);
trans.transform(xmlSource, result);
```

## 6.2 ExportServiceBean.java

### Issue 2E6BC0BE3AE585C4CDBAA7A1F461FA22:

**Source :**
**File** admin-backend/src/main/java/no/evote/service/ExportServiceBean.java 513

**Action** BranchNotTaken Branch not taken: (is != null)

```
510|InputStream is;
511|is = this.getClass().getClassLoader().getResourceAsStream("xsl" + File.separator + xslFilename);
512|
>513|if (is == null) {
514|    throw new EvoteException("Unable to find file: " + xslFilename);
```

**Sink :**
**File** admin-backend/src/main/java/no/evote/service/ExportServiceBean.java 521

**Action** InCall transFact.newTemplates(...): XML document parsed allowing external entity resolution

**Rule id** EF757EC9-C8ED-4931-A7D9-11A42E2D83E4

```
518|TransformerFactory transFact = TransformerFactory.newInstance();
519|Transformer transformer;
520|try {
>521|    Templates templates = transFact.newTemplates(xslSource);
522|    transformer = templates.newTransformer();
523|} catch (TransformerConfigurationException e) {
524|    throw new EvoteException("Error creating xslTransformer for stylesheet: " + xslFilename, e);
```

### Issue EBBF8E361B8E8B8BE6C3AA1304AA4475:

**Source :**
**File** admin-backend/src/main/java/no/evote/service/ExportServiceBean.java 544 **Action** Assign tf = newInstance()

**Rule id** EF757EC9-C8ED-4931-A7D9-11A42E2D83E4

```
541|DOMSource domSource = new DOMSource(doc); 542|
543|StreamResult result = new StreamResult(baos);
>544|TransformerFactory tf = TransformerFactory.newInstance();
545|Transformer transformer;
546|
547|transformer = tf.newTransformer();
```

**Sink :**
**File** admin-backend/src/main/java/no/evote/service/ExportServiceBean.java 547

**Action** InCall tf.newTransformer(): XML document parsed allowing external entity resolution

**Rule id** EF757EC9-C8ED-4931-A7D9-11A42E2D83E4

```
544|TransformerFactory tf = TransformerFactory.newInstance();
545|Transformer transformer;
546|
>547|transformer = tf.newTransformer();
548|
549|transformer.transform(domSource, result);
550|bytes = baos.toByteArray();
```

## [High] XML External Entity Injection

*Input Validation and Representation*

**Analysis:** Suspicious

### Abstract

Using XML parsers configured to not prevent nor limit external entities resolution can expose the parser to an XML External Entities attack

### Explanation

XML External Entities attacks benefit from an XML feature to build documents dynamically at the time of processing. An XML entity allows inclusion of data dynamically from a given resource. External entities allow an XML document to include data from an external URI. Unless configured to do otherwise, external entities force the XML parser to access the resource specified by the URI, e.g., a file on the local machine or on a remote system. This behavior exposes the application to XML External Entity (XXE) attacks, which can be used to perform denial of service of the local system, gain unauthorized access to files on the local machine, scan remote machines, and perform denial of service of remote systems.

The following XML document shows an example of an XXE attack.



```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE foo [
<!ELEMENT foo ANY >
<!ENTITY xxe SYSTEM "file:///dev/random" >]><foo>&xxe;</foo>
```

This example could crash the server (on a UNIX system), if the XML parser attempts to substitute the entity with the contents of the /dev/random file.

## Recommendation

An XML parser should be configured securely so that it does not allow external entities as part of an incoming XML document.

To avoid XXE injections the following properties should be set for an XML factory, parser or reader:

```
factory.setFeature("http://xml.org/sax/features/external-general-entities", false);
factory.setFeature("http://xml.org/sax/features/external-parameter-entities", false);
```

If inline DOCTYPE declaration is not needed, it can be completely disabled with the following property:

```
factory.setFeature("http://apache.org/xml/features/disallow-doctype-decl", true);
```

To protect a TransformerFactory, the following attribute should be set:

```
TransformerFactory transFact = TransformerFactory.newInstance();
transFact.setAttribute(XMLConstants.ACCESS_EXTERNAL_DTD, "");
Transformer trans = transFact.newTransformer(xsltSource);
trans.transform(xmlSource, result);
```

Alternatively, it is possible to use a securely configured XMLReader to set the transformation sources:

```
XMLReader reader = XMLReaderFactory.createXMLReader();
reader.setFeature("http://xml.org/sax/features/external-general-entities", false);
reader.setFeature("http://xml.org/sax/features/external-parameter-entities", false);
Source xmlSource = new SAXSource(reader, new InputSource(new FileInputStream(xmlFile)));
Source xsltSource = new SAXSource(reader, new InputSource(new FileInputStream(xsltFile)));
Result result = new StreamResult(System.out);
TransformerFactory transFact = TransformerFactory.newInstance();
Transformer trans = transFact.newTransformer(xsltSource);
trans.transform(xmlSource, result);
```

## 6.3 ExportServiceBean.java

### Issue 3D9716D47693DF5627957E53B22FoA65:

#### Source :

File admin-backend/src/main/java/no/evote/service/ExportServiceBean.java 513

Action BranchNotTaken Branch not taken: (is != null)

```
510|InputStream is;
511|is = this.getClass().getClassLoader().getResourceAsStream("xsl" + File.separator + xslFilename);
512|
>513|if (is == null) {
514|    throw new EvoteException("Unable to find file: " + xslFilename);
```

#### Sink :

File admin-backend/src/main/java/no/evote/service/ExportServiceBean.java 521

Action InCall transFact.newTemplates(...): XML document parsed allowing external entity resolution

Rule id oDE6DCC8-25D6-4224-AE1B-B7F8Co24AB95

```
518|TransformerFactory transFact = TransformerFactory.newInstance();
519|Transformer transformer;
520|try {
>521|    Templates templates = transFact.newTemplates(xslSource);
522|    transformer = templates.newTransformer();
523|} catch (TransformerConfigurationException e) {
524|    throw new EvoteException("Error creating xslTransformer for stylesheet: " + xslFilename, e);
```

### Issue B8DC69250CEAB75160A33E7B8CA8ADD9:

#### Source :

File admin-backend/src/main/java/no/evote/service/ExportServiceBean.java 544 Action Assign tf = newInstance()

Rule id oDE6DCC8-25D6-4224-AE1B-B7F8Co24AB95

```
541|DOMSource domSource = new DOMSource(doc); 542|
543|StreamResult result = new StreamResult(baos);
>544|TransformerFactory tf = TransformerFactory.newInstance();
545|Transformer transformer;
546|
547|transformer = tf.newTransformer();
```

#### Sink :

File admin-backend/src/main/java/no/evote/service/ExportServiceBean.java 547

Action InCall tf.newTransformer(): XML document parsed allowing external entity resolution

```

544|TransformerFactory tf = TransformerFactory.newInstance();
545|Transformer transformer;
546|
>547|transformer = tf.newTransformer();
548|
549|transformer.transform(domSource, result);
550|bytes = baos.toByteArray();

```

## [Medium] Code Correctness / Constructor Invokes Overridable Function

Code Quality

**Analysis:** Bad Practice

### Abstract

A constructor of the class calls a function that can be overridden.

### Explanation

When a constructor calls an overridable function, it may allow an attacker to access the `this` reference prior to the object being fully initialized, which can in turn lead to a vulnerability.

**Example 1:** The following calls a method that can be overridden.

```

... class User {
private String username;
private boolean valid;
public User(String username, String password){ this.username
= username;
this.valid = validateUser(username, password);
}
public boolean validateUser(String username, String
password){ //validate user is real and can authenticate ... }
public final boolean isValid(){ return
valid;
}
}

```

Since the function `validateUser` and the class are not `final`, it means that they can be overridden, and then initializing a variable to the subclass that overrides this function would allow bypassing of the `validateUser` functionality. For example:

```

... class Attacker extends User{ public
Attacker(String username, String password){
super(username, password);
}
public boolean validateUser(String username, String password){
return true;
}
} ... class MainClass{ public static
void main(String[] args){
User hacker = new Attacker("Evil", "Hacker"); if
(hacker.isValid()){
System.out.println("Attack successful!");
}else{
System.out.println("Attack failed"); }
}
}

```

The above code prints "Attack successful!", since the `Attacker` class overrides the `validateUser()` function that is called from the constructor of the superclass `User`, and Java will first look in the subclass for functions called from the constructor.

### Recommendation

Constructors should not call functions that can be overridden, either by specifying them as `final`, or specifying the class as `final`. Alternatively if this code is only ever needed in the constructor, the

private access specifier can be used, or the logic could be placed directly into the constructor of the superclass.

**Example 2:** The following makes the class `final` to prevent the function from being overridden elsewhere.

```
... final class User {
private String username;
private boolean valid;
    public User(String username, String password){
this.username = username;
    this.valid = validateUser(username, password);
    }
    private boolean validateUser(String username, String password){
//validate user is real and can
authenticate ... }
    public final boolean isValid(){
return valid;
    }
}
```

This example specifies the class as `final`, so that it cannot be subclassed, and changes the `validateUser()` function to `private`, since it is not needed elsewhere in this application. This is programming defensively, since at a later date it may be decided that the `User` class needs to be subclassed, which would result in this vulnerability reappearing if the `validateUser()` function was not set to `private`.

## 6.4 Access.java

**Issue 2B0854F2C63026766A1449B2A463A138:**

**Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/common/rbac/Access.java 27

**Fact Name:** no.valg.eva.admin.common.rbac.Access.getPath

```
24|     } else {
25|         int index = path.lastIndexOf('.');
26|         this.parent = new Access(path.substring(0, index));
>27|         this.path = this.parent.getPath() + "." + path.substring(index + 1);
28|     }
29| } else {
30|     this.parent = parent;
```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/common/rbac/Access.java 27

**Fact Name:** no.valg.eva.admin.common.rbac.Access.getPath

```
24|     } else {
25|         int index = path.lastIndexOf('.');
26|         this.parent = new Access(path.substring(0, index));
>27|         this.path = this.parent.getPath() + "." + path.substring(index + 1);
28|     }
29| } else {
30|     this.parent = parent;
```

**Issue 8BB11A61B6B013BC1DF17ECE5A6E9AFC:**

**Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/common/rbac/Access.java 31

**Fact Name:** no.valg.eva.admin.common.rbac.Access.getPath

```
29|     } else {
30|         this.parent = parent;
>31|         this.path = parent.getPath() + "." + path;
32|     }
33|     this.name = "@access." + path;
```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/common/rbac/Access.java 31

**Fact Name:** no.valg.eva.admin.common.rbac.Access.getPath

```
29|     } else {
30|         this.parent = parent;
>31|         this.path = parent.getPath() + "." + path; 32|     }
33|     this.name = "@access." + path;
```

## 6.5 Affiliation.java

**Issue 0B3AD8943BDE578452C47BB47A8C10E0:**

**Source :****File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Affiliation.java 77**Fact Name:** no.valg.eva.admin.configuration.domain.model.Affiliation.getBallot

```

74|public Affiliation(final Affiliation affiliation) {
75|    super();
76|    party = affiliation.getParty();
>77|    ballot = affiliation.getBallot();
78|    displayOrder = affiliation.getDisplayOrder();
79|    showCandidateResidence = affiliation.isShowCandidateResidence();
80|    showCandidateProfession = affiliation.isShowCandidateProfession();

```

**Sink :****File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Affiliation.java 77**Fact Name:** no.valg.eva.admin.configuration.domain.model.Affiliation.getBallot

```

74|public Affiliation(final Affiliation affiliation) {
75|    super();
76|    party = affiliation.getParty();
>77|    ballot = affiliation.getBallot();
78|    displayOrder = affiliation.getDisplayOrder();
79|    showCandidateResidence = affiliation.isShowCandidateResidence();
80|    showCandidateProfession = affiliation.isShowCandidateProfession();

```

**Issue 22138BEEC283719F6D7AD58E0400874A:****Source :****File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Affiliation.java 79**Fact Name:** no.valg.eva.admin.configuration.domain.model.Affiliation.isShowCandidateResidence

```

76|    party = affiliation.getParty();
77|    ballot = affiliation.getBallot();
78|    displayOrder = affiliation.getDisplayOrder();
>79|    showCandidateResidence = affiliation.isShowCandidateResidence();
80|    showCandidateProfession = affiliation.isShowCandidateProfession();
81|    approved = affiliation.isApproved();

```

**Sink :****File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Affiliation.java 79**Fact Name:** no.valg.eva.admin.configuration.domain.model.Affiliation.isShowCandidateResidence

```

76|    party = affiliation.getParty();
77|    ballot = affiliation.getBallot();
78|    displayOrder = affiliation.getDisplayOrder();
>79|    showCandidateResidence = affiliation.isShowCandidateResidence();
80|    showCandidateProfession = affiliation.isShowCandidateProfession();
81|    approved = affiliation.isApproved();

```

**Issue 3417FBA7FFF05E57D743386BC205F100:****Source :****File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Affiliation.java 80**Fact Name:** no.valg.eva.admin.configuration.domain.model.Affiliation.isShowCandidateProfession

```

77|    ballot = affiliation.getBallot();
78|    displayOrder = affiliation.getDisplayOrder();
79|    showCandidateResidence = affiliation.isShowCandidateResidence();
>80|    showCandidateProfession = affiliation.isShowCandidateProfession();
81|    approved = affiliation.isApproved();

```

**Sink :****File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Affiliation.java 80**Fact Name:** no.valg.eva.admin.configuration.domain.model.Affiliation.isShowCandidateProfession

```

77|    ballot = affiliation.getBallot();
78|    displayOrder = affiliation.getDisplayOrder();
79|    showCandidateResidence = affiliation.isShowCandidateResidence();
>80|    showCandidateProfession = affiliation.isShowCandidateProfession();
81|    approved = affiliation.isApproved();

```

**Issue 48FA6CF10FFB6FF0EE5173932481643D:****Source :****File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Affiliation.java 78**Fact Name:** no.valg.eva.admin.configuration.domain.model.Affiliation.getDisplayOrder

```

75|super();
76|party = affiliation.getParty();
77|ballot = affiliation.getBallot();
>78|displayOrder = affiliation.getDisplayOrder();
79|showCandidateResidence = affiliation.isShowCandidateResidence(); 80|showCandidateProfession =
affiliation.isShowCandidateProfession();
81|approved = affiliation.isApproved();

```

**Sink :****File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Affiliation.java 78**Fact Name:** no.valg.eva.admin.configuration.domain.model.Affiliation.getDisplayOrder

```

75|super();

```

```

76| party = affiliation.getParty();
77| ballot = affiliation.getBallot();
>78| displayOrder = affiliation.getDisplayOrder();
79| showCandidateResidence = affiliation.isShowCandidateResidence();
80| showCandidateProfession = affiliation.isShowCandidateProfession();
81| approved = affiliation.isApproved();

```

**Issue 755C55B757CD93D7808184D08E643E19:****Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Affiliation.java 81

**Fact Name:** no.valg.eva.admin.configuration.domain.model.Affiliation.isApproved

```

78| displayOrder = affiliation.getDisplayOrder();
79| showCandidateResidence = affiliation.isShowCandidateResidence();
80| showCandidateProfession = affiliation.isShowCandidateProfession();
>81| approved = affiliation.isApproved();
82|}
83|
84|@ManyToOne(fetch = EAGER)

```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Affiliation.java 81

**Fact Name:** no.valg.eva.admin.configuration.domain.model.Affiliation.isApproved

```

78| displayOrder = affiliation.getDisplayOrder();
79| showCandidateResidence = affiliation.isShowCandidateResidence();
80| showCandidateProfession = affiliation.isShowCandidateProfession();
>81| approved = affiliation.isApproved();
82|}
83|
84|@ManyToOne(fetch = EAGER)

```

**Issue D07057932541B324E73CCD34EAE1137F:****Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Affiliation.java 76

**Fact Name:** no.valg.eva.admin.configuration.domain.model.Affiliation.getParty

```

74|public Affiliation(final Affiliation affiliation) {
75|    super();
>76|    party = affiliation.getParty();
77|    ballot = affiliation.getBallot();
78|    displayOrder = affiliation.getDisplayOrder();
79|    showCandidateResidence = affiliation.isShowCandidateResidence();

```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Affiliation.java 76

**Fact Name:** no.valg.eva.admin.configuration.domain.model.Affiliation.getParty

```

74|public Affiliation(final Affiliation affiliation) {
75|    super();
>76|    party = affiliation.getParty();
77|    ballot = affiliation.getBallot();
78|    displayOrder = affiliation.getDisplayOrder();
79|    showCandidateResidence = affiliation.isShowCandidateResidence();

```

## 6.6 BinaryData.java

**Issue 120E4CA663EC05DAD8634205708E87F1:****Source :**

**File** admin-common/src/main/java/no/evote/model/BinaryData.java 45

**Fact Name:** no.evote.model.BinaryData.getMimeType

```

42| this.tableName = binaryData.getTableName();
43| this.columnName = binaryData.getColumnName();
44| this.fileName = binaryData.getFileName(); >45| this.mimeType = binaryData.getMimeType();
46|}
47|
48|@ManyToOne(fetch = FetchType.LAZY)

```

**Sink :**

**File** admin-common/src/main/java/no/evote/model/BinaryData.java 45

**Fact Name:** no.evote.model.BinaryData.getMimeType

```

42| this.tableName = binaryData.getTableName();
43| this.columnName = binaryData.getColumnName();
44| this.fileName = binaryData.getFileName();
>45| this.mimeType = binaryData.getMimeType();
46|}
47|
48|@ManyToOne(fetch = FetchType.LAZY)

```

**Issue 6B2B982055298DC8E2C70D99797EDD22:****Source :****File** admin-common/src/main/java/no/evote/model/BinaryData.java 43**Fact Name:** no.evote.model.BinaryData.getColumnName

```

40| this.electionEvent = binaryData.getElectionEvent();
41| this.binaryData = binaryData.getBinaryData();
42| this.tableName = binaryData.getTableName();
>43| this.columnName = binaryData.getColumnName();
44| this.fileName = binaryData.getFileName();
45| this.mimeType = binaryData.getMimeType();

```

**Sink :****File** admin-common/src/main/java/no/evote/model/BinaryData.java 43**Fact Name:** no.evote.model.BinaryData.getColumnName

```

40| this.electionEvent = binaryData.getElectionEvent();
41| this.binaryData = binaryData.getBinaryData();
42| this.tableName = binaryData.getTableName();
>43| this.columnName = binaryData.getColumnName();
44| this.fileName = binaryData.getFileName();
45| this.mimeType = binaryData.getMimeType();

```

**Issue 9F43260C8891FF4FoE2C317D009383D9:****Source :****File** admin-common/src/main/java/no/evote/model/BinaryData.java 41**Fact Name:** no.evote.model.BinaryData.getBinaryData

```

38| super();
39| this.number = binaryData.getNumber();
40| this.electionEvent = binaryData.getElectionEvent();
>41| this.binaryData = binaryData.getBinaryData(); 42| this.tableName = binaryData.getTableName();
43| this.columnName = binaryData.getColumnName();
44| this.fileName = binaryData.getFileName();

```

**Sink :****File** admin-common/src/main/java/no/evote/model/BinaryData.java 41**Fact Name:** no.evote.model.BinaryData.getBinaryData

```

38| super();
39| this.number = binaryData.getNumber();
40| this.electionEvent = binaryData.getElectionEvent();
>41| this.binaryData = binaryData.getBinaryData();
42| this.tableName = binaryData.getTableName();
43| this.columnName = binaryData.getColumnName();
44| this.fileName = binaryData.getFileName();

```

**Issue A04E22E0A70521B52964863C6D8D3A45:****Source :****File** admin-common/src/main/java/no/evote/model/BinaryData.java 42**Fact Name:** no.evote.model.BinaryData.getTableName

```

39| this.number = binaryData.getNumber();
40| this.electionEvent = binaryData.getElectionEvent();
41| this.binaryData = binaryData.getBinaryData(); >42| this.tableName = binaryData.getTableName();
43| this.columnName = binaryData.getColumnName();
44| this.fileName = binaryData.getFileName();
45| this.mimeType = binaryData.getMimeType();

```

**Sink :****File** admin-common/src/main/java/no/evote/model/BinaryData.java 42**Fact Name:** no.evote.model.BinaryData.getTableName

```

39| this.number = binaryData.getNumber();
40| this.electionEvent = binaryData.getElectionEvent();
41| this.binaryData = binaryData.getBinaryData();
>42| this.tableName = binaryData.getTableName();
43| this.columnName = binaryData.getColumnName();
44| this.fileName = binaryData.getFileName();
45| this.mimeType = binaryData.getMimeType();

```

**Issue CC2E72B05486D3796C34C929FA72959A:****Source :****File** admin-common/src/main/java/no/evote/model/BinaryData.java 40**Fact Name:** no.evote.model.BinaryData.getElectionEvent

```

37| public BinaryData(final BinaryData binaryData) {
38|     super();
39|     this.number = binaryData.getNumber();
>40|     this.electionEvent = binaryData.getElectionEvent();
41|     this.binaryData = binaryData.getBinaryData();
42|     this.tableName = binaryData.getTableName();
43|     this.columnName = binaryData.getColumnName();

```

**Sink :****File** admin-common/src/main/java/no/evote/model/BinaryData.java 40**Fact Name:** no.evote.model.BinaryData.getElectionEvent

```

37| public BinaryData(final BinaryData binaryData) {
38|     super();
39|     this.number = binaryData.getNumber();
>40|     this.electionEvent = binaryData.getElectionEvent();
41|     this.binaryData = binaryData.getBinaryData();
42|     this.tableName = binaryData.getTableName();
43|     this.columnName = binaryData.getColumnName();

```

**Issue D9D565E8B91D575C91ED1916F8576240:****Source :**

File admin-common/src/main/java/no/evote/model/BinaryData.java 44

Fact Name: no.evote.model.BinaryData.getFileName

```

41|     this.binaryData = binaryData.getBinaryData(); 42|     this.tableName = binaryData.getTableName();
43|     this.columnName = binaryData.getColumnName();
>44|     this.fileName = binaryData.getFileName();
45|     this.mimeType = binaryData.getMimeType();

```

**Sink :**

File admin-common/src/main/java/no/evote/model/BinaryData.java 44

Fact Name: no.evote.model.BinaryData.getFileName

```

41|     this.binaryData = binaryData.getBinaryData(); 42|     this.tableName = binaryData.getTableName();
43|     this.columnName = binaryData.getColumnName();
>44|     this.fileName = binaryData.getFileName();
45|     this.mimeType = binaryData.getMimeType();

```

**Issue E5ACCCEB5CC748E1A1350E0E3DFC84:****Source :**

File admin-common/src/main/java/no/evote/model/BinaryData.java 39

Fact Name: no.evote.model.BinaryData.getNumber

```

37| public BinaryData(final BinaryData binaryData) {
38|     super();
>39|     this.number = binaryData.getNumber();
40|     this.electionEvent = binaryData.getElectionEvent();
41|     this.binaryData = binaryData.getBinaryData();
42|     this.tableName = binaryData.getTableName();

```

**Sink :**

File admin-common/src/main/java/no/evote/model/BinaryData.java 39

Fact Name: no.evote.model.BinaryData.getNumber

```

37| public BinaryData(final BinaryData binaryData) {
38|     super();
>39|     this.number = binaryData.getNumber();
40|     this.electionEvent = binaryData.getElectionEvent();
41|     this.binaryData = binaryData.getBinaryData();
42|     this.tableName = binaryData.getTableName();

```

## 6.7 Candidate.java

**Issue 14C72D80CF23139A1BD309468206C6B1:****Source :**

File admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Candidate.java 139

Fact Name: no.valg.eva.admin.configuration.domain.model.Candidate.getNameLine

```

136| id = candidate.getId();
137| dateOfBirth = candidate.getDateOfBirth();
138| baselineVotes = candidate.isBaselineVotes();
>139| nameLine = candidate.getNameLine();
140| firstName = candidate.getFirstName();
141| middleName = candidate.getMiddleName();
142| lastName = candidate.getLastName();

```

**Sink :**

File admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Candidate.java 139

Fact Name: no.valg.eva.admin.configuration.domain.model.Candidate.getNameLine

```

136| id = candidate.getId();
137| dateOfBirth = candidate.getDateOfBirth();
138| baselineVotes = candidate.isBaselineVotes();
>139| nameLine = candidate.getNameLine();
140| firstName = candidate.getFirstName();
141| middleName = candidate.getMiddleName();
142| lastName = candidate.getLastName();

```

**Issue 200770ECB5D1C27673BD65F1BB65F10F:****Source :**

File admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Candidate.java 134

Fact Name: no.valg.eva.admin.configuration.domain.model.Candidate.getMaritalStatus

```

131| super();
132| affiliation = candidate.getAffiliation();
133| ballot = candidate.getBallot();

```

```

>134|maritalStatus = candidate.getMaritalStatus();
135|displayOrder = candidate.getDisplayOrder();
136|id = candidate.getId();
137|dateOfBirth = candidate.getDateOfBirth();

```

**Sink :****File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Candidate.java 134**Fact Name:** no.valg.eva.admin.configuration.domain.model.Candidate.getMaritalStatus

```

131|super();
132|affiliation = candidate.getAffiliation();
133|ballot = candidate.getBallot();
>134|maritalStatus = candidate.getMaritalStatus();
135|displayOrder = candidate.getDisplayOrder();
136|id = candidate.getId();
137|dateOfBirth = candidate.getDateOfBirth();

```

**Issue 28F3ECDD15596E3BF9CDF3E18FD8AAE9:****Source :****File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Candidate.java 153**Fact Name:** no.valg.eva.admin.configuration.domain.model.Candidate.isApproved

```

150| residence = candidate.getResidence();
151| profession = candidate.getProfession();
152| infoText = candidate.getInfoText();
>153| approved = candidate.isApproved();
154|}
155|
156|@ManyToOne(fetch = FetchType.LAZY)

```

**Sink :****File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Candidate.java 153**Fact Name:** no.valg.eva.admin.configuration.domain.model.Candidate.isApproved

```

150| residence = candidate.getResidence();
151| profession = candidate.getProfession();
152| infoText = candidate.getInfoText();
>153| approved = candidate.isApproved();
154|}
155|
156|@ManyToOne(fetch = FetchType.LAZY)

```

**Issue 3BE3515DF60B27731C16E3F48BED77FB:****Source :****File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Candidate.java 141**Fact Name:** no.valg.eva.admin.configuration.domain.model.Candidate.getMiddleName

```

138|baselineVotes = candidate.isBaselineVotes();
139|nameLine = candidate.getNameLine();
140|firstName = candidate.getFirstName();
>141|middleName = candidate.getMiddleName();
142|lastName = candidate.getLastName();
143|addressLine1 = candidate.getAddressLine1();
144|addressLine2 = candidate.getAddressLine2();

```

**Sink :****File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Candidate.java 141**Fact Name:** no.valg.eva.admin.configuration.domain.model.Candidate.getMiddleName

```

138|baselineVotes = candidate.isBaselineVotes();
139|nameLine = candidate.getNameLine();
140|firstName = candidate.getFirstName();
>141|middleName = candidate.getMiddleName();
142|lastName = candidate.getLastName();
143|addressLine1 = candidate.getAddressLine1();
144|addressLine2 = candidate.getAddressLine2();

```

**Issue 45459BF639551CE4B7AB1871098ABDD5:****Source :****File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Candidate.java 133**Fact Name:** no.valg.eva.admin.configuration.domain.model.Candidate.getBallot

```

130|public Candidate(Candidate candidate) {
131| super();
132| affiliation = candidate.getAffiliation();
>133| ballot = candidate.getBallot();
134| maritalStatus = candidate.getMaritalStatus();
135| displayOrder = candidate.getDisplayOrder();
136| id = candidate.getId();

```

**Sink :****File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Candidate.java 133**Fact Name:** no.valg.eva.admin.configuration.domain.model.Candidate.getBallot

```

130|public Candidate(Candidate candidate) {
131| super();

```



```

132|   affiliation = candidate.getAffiliation();
>133|   ballot = candidate.getBallot();
134|   maritalStatus = candidate.getMaritalStatus();
135|   displayOrder = candidate.getDisplayOrder();
136|   id = candidate.getId();

```

**Issue 552DFBF6EFCCFC3211ED262863146953:****Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Candidate.java 137 **Fact**  
**Name:** no.valg.eva.admin.configuration.domain.model.Candidate.getDateOfBirth

```

134|maritalStatus = candidate.getMaritalStatus();
135|displayOrder = candidate.getDisplayOrder();
136|id = candidate.getId();
>137|dateOfBirth = candidate.getDateOfBirth();
138|baselineVotes = candidate.isBaselineVotes();
139|nameLine = candidate.getNameLine();
140|firstName = candidate.getFirstName();

```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Candidate.java 137 **Fact**  
**Name:** no.valg.eva.admin.configuration.domain.model.Candidate.getDateOfBirth

```

134|maritalStatus = candidate.getMaritalStatus();
135|displayOrder = candidate.getDisplayOrder();
136|id = candidate.getId();
>137|dateOfBirth = candidate.getDateOfBirth();
138|baselineVotes = candidate.isBaselineVotes();
139|nameLine = candidate.getNameLine();
140|firstName = candidate.getFirstName();

```

**Issue 66B4C12E4B8CDA6B55FAD82287219BB5:****Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Candidate.java 140  
**Fact** **Name:** no.valg.eva.admin.configuration.domain.model.Candidate.getFirstName

```

137|dateOfBirth = candidate.getDateOfBirth();
138|baselineVotes = candidate.isBaselineVotes();
139|nameLine = candidate.getNameLine();
>140|firstName = candidate.getFirstName();
141|middleName = candidate.getMiddleName();
142|lastName = candidate.getLastName();
143|addressLine1 = candidate.getAddressLine1();

```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Candidate.java 140  
**Fact** **Name:** no.valg.eva.admin.configuration.domain.model.Candidate.getFirstName

```

137|dateOfBirth = candidate.getDateOfBirth();
138|baselineVotes = candidate.isBaselineVotes();
139|nameLine = candidate.getNameLine();
>140|firstName = candidate.getFirstName();
141|middleName = candidate.getMiddleName();
142|lastName = candidate.getLastName();
143|addressLine1 = candidate.getAddressLine1();

```

**Issue 6C2939EEAC2A798DB7EAE249C8171231:****Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Candidate.java 146  
**Fact** **Name:** no.valg.eva.admin.configuration.domain.model.Candidate.getPostalCode

```

143|addressLine1 = candidate.getAddressLine1();
144|addressLine2 = candidate.getAddressLine2();
145|addressLine3 = candidate.getAddressLine3();
>146|postalCode = candidate.getPostalCode();
147|postTown = candidate.getPostTown();
148|email = candidate.getEmail();
149|telephoneNumber = candidate.getTelephoneNumber();

```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Candidate.java 146  
**Fact** **Name:** no.valg.eva.admin.configuration.domain.model.Candidate.getPostalCode

```

143|addressLine1 = candidate.getAddressLine1();
144|addressLine2 = candidate.getAddressLine2();
145|addressLine3 = candidate.getAddressLine3();
>146|postalCode = candidate.getPostalCode();
147|postTown = candidate.getPostTown();
148|email = candidate.getEmail();
149|telephoneNumber = candidate.getTelephoneNumber();

```

**Issue 7A45E2E30AA280A57D0410C0B5319B6F:****Source :**

File admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Candidate.java 142

Fact Name: no.valg.eva.admin.configuration.domain.model.Candidate.getLastName

```
139|nameLine = candidate.getNameLine();
140|firstName = candidate.getFirstName();
141|middleName = candidate.getMiddleName();
>142|lastName = candidate.getLastName();
143|addressLine1 = candidate.getAddressLine1();
144|addressLine2 = candidate.getAddressLine2();
145|addressLine3 = candidate.getAddressLine3();
```

**Sink :**

File admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Candidate.java 142

Fact Name: no.valg.eva.admin.configuration.domain.model.Candidate.getLastName

```
139|nameLine = candidate.getNameLine();
140|firstName = candidate.getFirstName();
141|middleName = candidate.getMiddleName();
>142|lastName = candidate.getLastName();
143|addressLine1 = candidate.getAddressLine1();
144|addressLine2 = candidate.getAddressLine2();
145|addressLine3 = candidate.getAddressLine3();
```

### Issue 8463B03C14BE83A4DA215435ACC6328C:

**Source :**

File admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Candidate.java 150

Fact Name: no.valg.eva.admin.configuration.domain.model.Candidate.getResidence

```
147|postTown = candidate.getPostTown();
148|email = candidate.getEmail();
149|telephoneNumber = candidate.getTelephoneNumber();
>150|residence = candidate.getResidence();
151|profession = candidate.getProfession();
152|infoText = candidate.getInfoText();
153|approved = candidate.isApproved();
```

**Sink :**

File admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Candidate.java 150

Fact Name: no.valg.eva.admin.configuration.domain.model.Candidate.getResidence

```
147|postTown = candidate.getPostTown();
148|email = candidate.getEmail();
149|telephoneNumber = candidate.getTelephoneNumber();
>150|residence = candidate.getResidence();
151|profession = candidate.getProfession();
152|infoText = candidate.getInfoText();
153|approved = candidate.isApproved();
```

### Issue 8C02354BF0CoF57451D29F07AEF624AB:

**Source :**

File admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Candidate.java 145

Fact Name: no.valg.eva.admin.configuration.domain.model.Candidate.getAddressLine3

```
142|lastName = candidate.getLastName();
143|addressLine1 = candidate.getAddressLine1();
144|addressLine2 = candidate.getAddressLine2();
>145|addressLine3 = candidate.getAddressLine3();
146|postalCode = candidate.getPostalCode();
147|postTown = candidate.getPostTown();
148|email = candidate.getEmail();
```

**Sink :**

File admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Candidate.java 145

Fact Name: no.valg.eva.admin.configuration.domain.model.Candidate.getAddressLine3

```
142|lastName = candidate.getLastName();
143|addressLine1 = candidate.getAddressLine1();
144|addressLine2 = candidate.getAddressLine2();
>145|addressLine3 = candidate.getAddressLine3();
146|postalCode = candidate.getPostalCode();
147|postTown = candidate.getPostTown();
148|email = candidate.getEmail();
```

### Issue 933A65508973817D639852C8363BD688:

**Source :**

File admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Candidate.java 143

Fact Name: no.valg.eva.admin.configuration.domain.model.Candidate.getAddressLine1

```
140|firstName = candidate.getFirstName();
141|middleName = candidate.getMiddleName();
142|lastName = candidate.getLastName();
>143|addressLine1 = candidate.getAddressLine1();
144|addressLine2 = candidate.getAddressLine2();
145|addressLine3 = candidate.getAddressLine3();
146|postalCode = candidate.getPostalCode();
```

**Sink :****File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Candidate.java 143**Fact Name:** no.valg.eva.admin.configuration.domain.model.Candidate.getAddressLine1

```

140|firstName = candidate.getFirstName();
141|middleName = candidate.getMiddleName();
142|lastName = candidate.getLastName();
>143|addressLine1 = candidate.getAddressLine1();
144|addressLine2 = candidate.getAddressLine2();
145|addressLine3 = candidate.getAddressLine3();
146|postalCode = candidate.getPostalCode();

```

**Issue A4591B60B17C41D6FoA9FBFo360FB67F:****Source :****File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Candidate.java 132**Fact Name:** no.valg.eva.admin.configuration.domain.model.Candidate.getAffiliation

```

130|public Candidate(Candidate candidate) {
131|    super();
>132|    affiliation = candidate.getAffiliation();
133|    ballot = candidate.getBallot();
134|    maritalStatus = candidate.getMaritalStatus();
135|    displayOrder = candidate.getDisplayOrder();

```

**Sink :****File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Candidate.java 132**Fact Name:** no.valg.eva.admin.configuration.domain.model.Candidate.getAffiliation

```

130|public Candidate(Candidate candidate) {
131|    super();
>132|    affiliation = candidate.getAffiliation();
133|    ballot = candidate.getBallot();
134|    maritalStatus = candidate.getMaritalStatus();
135|    displayOrder = candidate.getDisplayOrder();

```

**Issue B93BCACD25C97E21B9ED80076DD2E2E5:****Source :****File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Candidate.java 152 **Fact****Name:** no.valg.eva.admin.configuration.domain.model.Candidate.getInfoText

```

149|    telephoneNumber = candidate.getTelephoneNumber();
150|    residence = candidate.getResidence();
151|    profession = candidate.getProfession();
>152|    infoText = candidate.getInfoText();
153|    approved = candidate.isApproved();

```

**Sink :****File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Candidate.java 152 **Fact****Name:** no.valg.eva.admin.configuration.domain.model.Candidate.getInfoText

```

149|    telephoneNumber = candidate.getTelephoneNumber();
150|    residence = candidate.getResidence();
151|    profession = candidate.getProfession();
>152|    infoText = candidate.getInfoText();
153|    approved = candidate.isApproved();

```

**Issue BA93AA539E9C7811CB5C7EEEB0854203:****Source :****File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Candidate.java 136**Fact Name:** no.valg.eva.admin.configuration.domain.model.Candidate.getId

```

133|ballot = candidate.getBallot();
134|maritalStatus = candidate.getMaritalStatus();
135|displayOrder = candidate.getDisplayOrder();
>136|id = candidate.getId();
137|dateOfBirth = candidate.getDateOfBirth();
138|baselineVotes = candidate.isBaselineVotes();
139|nameLine = candidate.getNameLine();

```

**Sink :****File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Candidate.java 136**Fact Name:** no.valg.eva.admin.configuration.domain.model.Candidate.getId

```

133|ballot = candidate.getBallot();
134|maritalStatus = candidate.getMaritalStatus();
135|displayOrder = candidate.getDisplayOrder();
>136|id = candidate.getId();
137|dateOfBirth = candidate.getDateOfBirth();
138|baselineVotes = candidate.isBaselineVotes();
139|nameLine = candidate.getNameLine();

```

**Issue C637FoFE224B661ECF790ED1A9491E31:****Source :****File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Candidate.java 144

**Fact** Name: no.valg.eva.admin.configuration.domain.model.Candidate.getAddressLine2

```
141|middleName = candidate.getMiddleName();
142|lastName = candidate.getLastName();
143|addressLine1 = candidate.getAddressLine1();
>144|addressLine2 = candidate.getAddressLine2();
145|addressLine3 = candidate.getAddressLine3();
146|postalCode = candidate.getPostalCode();
147|postTown = candidate.getPostTown();
```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Candidate.java 144

**Fact** Name: no.valg.eva.admin.configuration.domain.model.Candidate.getAddressLine2

```
141|middleName = candidate.getMiddleName();
142|lastName = candidate.getLastName();
143|addressLine1 = candidate.getAddressLine1();
>144|addressLine2 = candidate.getAddressLine2();
145|addressLine3 = candidate.getAddressLine3();
146|postalCode = candidate.getPostalCode();
147|postTown = candidate.getPostTown();
```

### Issue C6693E878BAA438C7BFEDD884C7F77F7:

**Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Candidate.java 147 **Fact**

Name: no.valg.eva.admin.configuration.domain.model.Candidate.getPostTown

```
144|addressLine2 = candidate.getAddressLine2();
145|addressLine3 = candidate.getAddressLine3();
146|postalCode = candidate.getPostalCode();
>147|postTown = candidate.getPostTown();
148|email = candidate.getEmail();
149|telephoneNumber = candidate.getTelephoneNumber();
150|residence = candidate.getResidence();
```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Candidate.java 147 **Fact**

Name: no.valg.eva.admin.configuration.domain.model.Candidate.getPostTown

```
144|addressLine2 = candidate.getAddressLine2();
145|addressLine3 = candidate.getAddressLine3();
146|postalCode = candidate.getPostalCode();
>147|postTown = candidate.getPostTown();
148|email = candidate.getEmail();
149|telephoneNumber = candidate.getTelephoneNumber();
150|residence = candidate.getResidence();
```

### Issue C7D1275F25E40E8499EDEC40C707ACF:

**Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Candidate.java 135

**Fact** Name: no.valg.eva.admin.configuration.domain.model.Candidate.getDisplayOrder

```
132|affiliation = candidate.getAffiliation();
133|ballot = candidate.getBallot();
134|maritalStatus = candidate.getMaritalStatus();
>135|displayOrder = candidate.getDisplayOrder();
136|id = candidate.getId();
137|dateOfBirth = candidate.getDateOfBirth();
138|baselineVotes = candidate.isBaselineVotes();
```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Candidate.java 135

**Fact** Name: no.valg.eva.admin.configuration.domain.model.Candidate.getDisplayOrder

```
132|affiliation = candidate.getAffiliation();
133|ballot = candidate.getBallot();
134|maritalStatus = candidate.getMaritalStatus();
>135|displayOrder = candidate.getDisplayOrder();
136|id = candidate.getId();
137|dateOfBirth = candidate.getDateOfBirth();
138|baselineVotes = candidate.isBaselineVotes();
```

### Issue CE7119259BA0C9AD14FFA91946C20111:

**Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Candidate.java 149

**Fact** Name: no.valg.eva.admin.configuration.domain.model.Candidate.getTelephoneNumber

```
146|postalCode = candidate.getPostalCode();
147|postTown = candidate.getPostTown();
148|email = candidate.getEmail();
>149|telephoneNumber = candidate.getTelephoneNumber();
150|residence = candidate.getResidence();
151|profession = candidate.getProfession();
152|infoText = candidate.getInfoText();
```

**Sink :**

File admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Candidate.java 149

Fact Name: no.valg.eva.admin.configuration.domain.model.Candidate.getTelephoneNumber

```
146|postalCode = candidate.getPostalCode();
147|postTown = candidate.getPostTown();
148|email = candidate.getEmail();
>149|telephoneNumber = candidate.getTelephoneNumber();
150|residence = candidate.getResidence();
151|profession = candidate.getProfession();
152|infoText = candidate.getInfoText();
```

### Issue D471F98E2CAB97695BE052BFAE4EEAF2:

Source :

File admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Candidate.java 151

Fact Name: no.valg.eva.admin.configuration.domain.model.Candidate.getProfession

```
148| email = candidate.getEmail();
149| telephoneNumber = candidate.getTelephoneNumber();
150| residence = candidate.getResidence();
>151| profession = candidate.getProfession();
152| infoText = candidate.getInfoText();
153| approved = candidate.isApproved();
```

Sink :

File admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Candidate.java 151

Fact Name: no.valg.eva.admin.configuration.domain.model.Candidate.getProfession

```
148| email = candidate.getEmail();
149| telephoneNumber = candidate.getTelephoneNumber();
150| residence = candidate.getResidence();
>151| profession = candidate.getProfession();
152| infoText = candidate.getInfoText();
153| approved = candidate.isApproved();
```

### Issue D8E6D4EB73DAFB950EE18B32A98338B4:

Source :

File admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Candidate.java 138

Fact Name: no.valg.eva.admin.configuration.domain.model.Candidate.isBaselineVotes

```
135|displayOrder = candidate.getDisplayOrder();
136|id = candidate.getId();
137|dateOfBirth = candidate.getDateOfBirth();
>138|baselineVotes = candidate.isBaselineVotes();
139|nameLine = candidate.getNameLine();
140|firstName = candidate.getFirstName();
141|middleName = candidate.getMiddleName();
```

Sink :

File admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Candidate.java 138

Fact Name: no.valg.eva.admin.configuration.domain.model.Candidate.isBaselineVotes

```
135|displayOrder = candidate.getDisplayOrder();
136|id = candidate.getId();
137|dateOfBirth = candidate.getDateOfBirth();
>138|baselineVotes = candidate.isBaselineVotes();
139|nameLine = candidate.getNameLine();
140|firstName = candidate.getFirstName();
141|middleName = candidate.getMiddleName();
```

### Issue DF7E10FC826FFD4DF74461484FAC53A3:

Source :

File admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Candidate.java 148

Fact Name: no.valg.eva.admin.configuration.domain.model.Candidate.getEmail

```
145|addressLine3 = candidate.getAddressLine3();
146|postalCode = candidate.getPostalCode();
147|postTown = candidate.getPostTown();
>148|email = candidate.getEmail();
149|telephoneNumber = candidate.getTelephoneNumber();
150|residence = candidate.getResidence();
151|profession = candidate.getProfession();
```

Sink :

File admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Candidate.java 148

Fact Name: no.valg.eva.admin.configuration.domain.model.Candidate.getEmail

```
145|addressLine3 = candidate.getAddressLine3();
146|postalCode = candidate.getPostalCode();
147|postTown = candidate.getPostTown();
>148|email = candidate.getEmail();
149|telephoneNumber = candidate.getTelephoneNumber();
150|residence = candidate.getResidence();
151|profession = candidate.getProfession();
```

## 6.8 Contest.java

### Issue 00ABF7DA1B7A0027E7454E6ABDC607F6:

**Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Contest.java 121 **Fact**  
**Name:** no.valg.eva.admin.configuration.domain.model.Contest.getMaxRenumber

```
118|this.minCandidates = contest.getMinCandidates();
119|this.minProposersOldParty = contest.getMinProposersOldParty();
120|this.minProposersNewParty = contest.getMinProposersNewParty();
>121|this.maxRenumber = contest.getMaxRenumber();
122|this.minVotes = contest.getMinVotes();
123|this.numberOfPositions = contest.getNumberOfPositions();
124|this.penultimateRecount = contest.getPenultimateRecount();
```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Contest.java 121 **Fact**  
**Name:** no.valg.eva.admin.configuration.domain.model.Contest.getMaxRenumber

```
118|this.minCandidates = contest.getMinCandidates();
119|this.minProposersOldParty = contest.getMinProposersOldParty();
120|this.minProposersNewParty = contest.getMinProposersNewParty();
>121|this.maxRenumber = contest.getMaxRenumber();
122|this.minVotes = contest.getMinVotes();
123|this.numberOfPositions = contest.getNumberOfPositions();
124|this.penultimateRecount = contest.getPenultimateRecount();
```

### Issue 0C41CC1315F9A97043E2B80E2738E1A0:

**Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Contest.java 113  
**Fact Name:** no.valg.eva.admin.configuration.domain.model.Contest.getEndDateOfBirth

```
110|super();
111|this.contestStatus = contest.getContestStatus();
112|this.election = contest.getElection();
>113|this.endDateOfBirth = contest.getEndDateOfBirth();
114|this.id = contest.getId();
115|this.maxCandidates = contest.getMaxCandidates();
116|this.maxVotes = contest.getMaxVotes();
```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Contest.java 113  
**Fact Name:** no.valg.eva.admin.configuration.domain.model.Contest.getEndDateOfBirth

```
110|super();
111|this.contestStatus = contest.getContestStatus();
112|this.election = contest.getElection();
>113|this.endDateOfBirth = contest.getEndDateOfBirth();
114|this.id = contest.getId();
115|this.maxCandidates = contest.getMaxCandidates();
116|this.maxVotes = contest.getMaxVotes();
```

### Issue 1658DDF30C25254211ABDE9F5424EAEF:

**Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Contest.java 115  
**Fact Name:** no.valg.eva.admin.configuration.domain.model.Contest.getMaxCandidates

```
112|this.election = contest.getElection();
113|this.endDateOfBirth = contest.getEndDateOfBirth();
114|this.id = contest.getId();
>115|this.maxCandidates = contest.getMaxCandidates();
116|this.maxVotes = contest.getMaxVotes();
117|this.maxWriteIn = contest.getMaxWriteIn();
118|this.minCandidates = contest.getMinCandidates();
```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Contest.java 115  
**Fact Name:** no.valg.eva.admin.configuration.domain.model.Contest.getMaxCandidates

```
112|this.election = contest.getElection();
113|this.endDateOfBirth = contest.getEndDateOfBirth();
114|this.id = contest.getId();
>115|this.maxCandidates = contest.getMaxCandidates();
116|this.maxVotes = contest.getMaxVotes();
117|this.maxWriteIn = contest.getMaxWriteIn();
118|this.minCandidates = contest.getMinCandidates();
```

### Issue 1BE1C5A961F4F6B42159138926FCFC7F:

**Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Contest.java 120  
**Fact Name:** no.valg.eva.admin.configuration.domain.model.Contest.getMinProposersNewParty

```
117|this.maxWriteIn = contest.getMaxWriteIn();
118|this.minCandidates = contest.getMinCandidates();
119|this.minProposersOldParty = contest.getMinProposersOldParty();
```

```

>120|this.minProposersNewParty = contest.getMinProposersNewParty();
121|this.maxRenumber = contest.getMaxRenumber();
122|this.minVotes = contest.getMinVotes();
123|this.numberofPositions = contest.getNumberofPositions();

```

**Sink :****File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Contest.java 120**Fact Name:** no.valg.eva.admin.configuration.domain.model.Contest.getMinProposersNewParty

```

117|this.maxWriteIn = contest.getMaxWriteIn();
118|this.minCandidates = contest.getMinCandidates();
119|this.minProposersOldParty = contest.getMinProposersOldParty();
>120|this.minProposersNewParty = contest.getMinProposersNewParty();
121|this.maxRenumber = contest.getMaxRenumber();
122|this.minVotes = contest.getMinVotes();
123|this.numberofPositions = contest.getNumberofPositions();

```

**Issue 20926723388292D65958AF8C5F6C7564:****Source :****File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Contest.java 118**Fact Name:** no.valg.eva.admin.configuration.domain.model.Contest.getMinCandidates

```

115|this.maxCandidates = contest.getMaxCandidates();
116|this.maxVotes = contest.getMaxVotes();
117|this.maxWriteIn = contest.getMaxWriteIn();
>118|this.minCandidates = contest.getMinCandidates();
119|this.minProposersOldParty = contest.getMinProposersOldParty();
120|this.minProposersNewParty = contest.getMinProposersNewParty();
121|this.maxRenumber = contest.getMaxRenumber();

```

**Sink :****File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Contest.java 118**Fact Name:** no.valg.eva.admin.configuration.domain.model.Contest.getMinCandidates

```

115|this.maxCandidates = contest.getMaxCandidates();
116|this.maxVotes = contest.getMaxVotes();
117|this.maxWriteIn = contest.getMaxWriteIn();
>118|this.minCandidates = contest.getMinCandidates();
119|this.minProposersOldParty = contest.getMinProposersOldParty();
120|this.minProposersNewParty = contest.getMinProposersNewParty();
121|this.maxRenumber = contest.getMaxRenumber();

```

**Issue 3785DB30471D9904EB4E29FAC666EB67:****Source :****File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Contest.java 125 **Fact****Name:** no.valg.eva.admin.configuration.domain.model.Contest.getName

```

122|    this.minVotes = contest.getMinVotes();
123|    this.numberofPositions = contest.getNumberofPositions();
124|    this.penultimateRecount = contest.getPenultimateRecount();
>125|    this.name = contest.getName();
126|}
127|
128|@ManyToOne(fetch = EAGER)

```

**Sink :****File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Contest.java 125 **Fact****Name:** no.valg.eva.admin.configuration.domain.model.Contest.getName

```

122|    this.minVotes = contest.getMinVotes();
123|    this.numberofPositions = contest.getNumberofPositions();
124|    this.penultimateRecount = contest.getPenultimateRecount();
>125|    this.name = contest.getName();
126|}
127|
128|@ManyToOne(fetch = EAGER)

```

**Issue 4E70A5B2E51477B9E5389602829CE7Do:****Source :****File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Contest.java 123**Fact Name:** no.valg.eva.admin.configuration.domain.model.Contest.getNumberofPositions

```

120|    this.minProposersNewParty = contest.getMinProposersNewParty();
121|    this.maxRenumber = contest.getMaxRenumber();
122|    this.minVotes = contest.getMinVotes();
>123|    this.numberofPositions = contest.getNumberofPositions();
124|    this.penultimateRecount = contest.getPenultimateRecount();
125|    this.name = contest.getName();

```

**Sink :****File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Contest.java 123**Fact Name:** no.valg.eva.admin.configuration.domain.model.Contest.getNumberofPositions

```

120|    this.minProposersNewParty = contest.getMinProposersNewParty();
121|    this.maxRenumber = contest.getMaxRenumber();

```

```

122|   this.minVotes = contest.getMinVotes();
>123|   this.numberOfPositions = contest.getNumberOfPositions();
124|   this.penultimateRecount = contest.getPenultimateRecount();
125|   this.name = contest.getName();

```

**Issue 4F70FoAC6F74505757A722C4C168587A:****Source :****File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Contest.java 114**Fact Name:** no.valg.eva.admin.configuration.domain.model.Contest.getId

```

111|this.contestStatus = contest.getContestStatus();
112|this.election = contest.getElection();
113|this.endDateOfBirth = contest.getEndDateOfBirth();
>114|this.id = contest.getId();
115|this.maxCandidates = contest.getMaxCandidates();
116|this.maxVotes = contest.getMaxVotes();
117|this.maxWriteIn = contest.getMaxWriteIn();

```

**Sink :****File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Contest.java 114**Fact Name:** no.valg.eva.admin.configuration.domain.model.Contest.getId

```

111|this.contestStatus = contest.getContestStatus();
112|this.election = contest.getElection();
113|this.endDateOfBirth = contest.getEndDateOfBirth();
>114|this.id = contest.getId();
115|this.maxCandidates = contest.getMaxCandidates();
116|this.maxVotes = contest.getMaxVotes();
117|this.maxWriteIn = contest.getMaxWriteIn();

```

**Issue 5AF64A9249E895C18C69BE59B96BAAEC:****Source :****File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Contest.java 124 **Fact****Name:** no.valg.eva.admin.configuration.domain.model.Contest.getPenultimateRecount

```

121|   this.maxRenummer = contest.getMaxRenummer();
122|   this.minVotes = contest.getMinVotes();
123|   this.numberOfPositions = contest.getNumberOfPositions();
>124|   this.penultimateRecount = contest.getPenultimateRecount();
125|   this.name = contest.getName();

```

**Sink :****File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Contest.java 124 **Fact****Name:** no.valg.eva.admin.configuration.domain.model.Contest.getPenultimateRecount

```

121|   this.maxRenummer = contest.getMaxRenummer();
122|   this.minVotes = contest.getMinVotes();
123|   this.numberOfPositions = contest.getNumberOfPositions();
>124|   this.penultimateRecount = contest.getPenultimateRecount();
125|   this.name = contest.getName();

```

**Issue 5EF2F8A1C2E14FBE95DF08AD0FE962Co:****Source :****File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Contest.java 111**Fact Name:** no.valg.eva.admin.configuration.domain.model.Contest.getContestStatus

```

109|public Contest(Contest contest) {
110|    super();
>111|    this.contestStatus = contest.getContestStatus();
112|    this.election = contest.getElection();
113|    this.endDateOfBirth = contest.getEndDateOfBirth();
114|    this.id = contest.getId();

```

**Sink :****File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Contest.java 111**Fact Name:** no.valg.eva.admin.configuration.domain.model.Contest.getContestStatus

```

109|public Contest(Contest contest) {
110|    super();
>111|    this.contestStatus = contest.getContestStatus();
112|    this.election = contest.getElection();
113|    this.endDateOfBirth = contest.getEndDateOfBirth();
114|    this.id = contest.getId();

```

**Issue 65D7A976F84E6DB819ABB53509EBCE31:****Source :****File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Contest.java 122**Fact Name:** no.valg.eva.admin.configuration.domain.model.Contest.getMinVotes

```

119|this.minProposersOldParty = contest.getMinProposersOldParty();
120|this.minProposersNewParty = contest.getMinProposersNewParty();
121|this.maxRenummer = contest.getMaxRenummer();
>122|this.minVotes = contest.getMinVotes();
123|this.numberOfPositions = contest.getNumberOfPositions();

```



```
124|this.penultimateRecount = contest.getPenultimateRecount();
125|this.name = contest.getName();
```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Contest.java 122

**Fact Name:** no.valg.eva.admin.configuration.domain.model.Contest.getMinVotes

```
119|this.minProposersOldParty = contest.getMinProposersOldParty();
120|this.minProposersNewParty = contest.getMinProposersNewParty();
121|this.maxRenumber = contest.getMaxRenumber();
>122|this.minVotes = contest.getMinVotes();
123|this.numberOfPositions = contest.getNumberOfPositions();
124|this.penultimateRecount = contest.getPenultimateRecount();
125|this.name = contest.getName();
```

**Issue 6D85AA5C22AC2B5C7E3FEA7799028FFB:****Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Contest.java 112

**Fact Name:** no.valg.eva.admin.configuration.domain.model.Contest.getElection

```
109|public Contest(Contest contest) {
110|    super();
111|    this.contestStatus = contest.getContestStatus();
>112|    this.election = contest.getElection();
113|    this.endDateOfBirth = contest.getEndDateOfBirth();
114|    this.id = contest.getId();
115|    this.maxCandidates = contest.getMaxCandidates();
```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Contest.java 112

**Fact Name:** no.valg.eva.admin.configuration.domain.model.Contest.getElection

```
109|public Contest(Contest contest) {
110|    super();
111|    this.contestStatus = contest.getContestStatus();
>112|    this.election = contest.getElection();
113|    this.endDateOfBirth = contest.getEndDateOfBirth();
114|    this.id = contest.getId();
115|    this.maxCandidates = contest.getMaxCandidates();
```

**Issue A52F7633DBC3DF31C21D721CB74AEF35:****Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Contest.java 119

**Fact Name:** no.valg.eva.admin.configuration.domain.model.Contest.getMinProposersOldParty

```
116|this.maxVotes = contest.getMaxVotes();
117|this.maxWriteIn = contest.getMaxWriteIn();
118|this.minCandidates = contest.getMinCandidates(); >119|this.minProposersOldParty =
contest.getMinProposersOldParty();
120|this.minProposersNewParty = contest.getMinProposersNewParty();
121|this.maxRenumber = contest.getMaxRenumber();
122|this.minVotes = contest.getMinVotes();
```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Contest.java 119

**Fact Name:** no.valg.eva.admin.configuration.domain.model.Contest.getMinProposersOldParty

```
116|this.maxVotes = contest.getMaxVotes();
117|this.maxWriteIn = contest.getMaxWriteIn();
118|this.minCandidates = contest.getMinCandidates();
>119|this.minProposersOldParty = contest.getMinProposersOldParty();
120|this.minProposersNewParty = contest.getMinProposersNewParty();
121|this.maxRenumber = contest.getMaxRenumber();
122|this.minVotes = contest.getMinVotes();
```

**Issue BDE7EBD609643FD7D5A2B5624E364F59:****Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Contest.java 116

**Fact Name:** no.valg.eva.admin.configuration.domain.model.Contest.getMaxVotes

```
113|this.endDateOfBirth = contest.getEndDateOfBirth();
114|this.id = contest.getId();
115|this.maxCandidates = contest.getMaxCandidates();
>116|this.maxVotes = contest.getMaxVotes();
117|this.maxWriteIn = contest.getMaxWriteIn();
118|this.minCandidates = contest.getMinCandidates();
119|this.minProposersOldParty = contest.getMinProposersOldParty();
```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Contest.java 116

**Fact Name:** no.valg.eva.admin.configuration.domain.model.Contest.getMaxVotes

```
113|this.endDateOfBirth = contest.getEndDateOfBirth();
114|this.id = contest.getId();
115|this.maxCandidates = contest.getMaxCandidates();
```

```

>116|this.maxVotes = contest.getMaxVotes();
117|this.maxWriteIn = contest.getMaxWriteIn();
118|this.minCandidates = contest.getMinCandidates();
119|this.minProposersOldParty = contest.getMinProposersOldParty();

```

**Issue D604E3663DEE894E26525993D8C55CD6:****Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Contest.java 117

**Fact Name:** no.valg.eva.admin.configuration.domain.model.Contest.getMaxWriteIn

```

114|this.id = contest.getId();
115|this.maxCandidates = contest.getMaxCandidates();
116|this.maxVotes = contest.getMaxVotes();
>117|this.maxWriteIn = contest.getMaxWriteIn();
118|this.minCandidates = contest.getMinCandidates();
119|this.minProposersOldParty = contest.getMinProposersOldParty();
120|this.minProposersNewParty = contest.getMinProposersNewParty();

```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Contest.java 117

**Fact Name:** no.valg.eva.admin.configuration.domain.model.Contest.getMaxWriteIn

```

114|this.id = contest.getId();
115|this.maxCandidates = contest.getMaxCandidates();
116|this.maxVotes = contest.getMaxVotes();
>117|this.maxWriteIn = contest.getMaxWriteIn();
118|this.minCandidates = contest.getMinCandidates();
119|this.minProposersOldParty = contest.getMinProposersOldParty();
120|this.minProposersNewParty = contest.getMinProposersNewParty();

```

## 6.9 ElectionCardModel.java

**Issue 620D73D32DE0CEA999B917A93C1E73CF:****Source :**

**File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/configuration/models/ElectionCardModel.java 26

**Fact Name:** no.valg.eva.admin.frontend.configuration.models.ElectionCardModel.getElectionCardInfoText

```

23|public ElectionCardModel(ElectionCardConfig electionCard, ElectionDayPollingPlace pollingPlace) { 24|
this.electionCard = electionCard;
25|   this.pollingPlace = pollingPlace;
>26|   this.customInfoText = !isEmpty(getElectionCardInfoText()) && !StringUtils.equals(electionCard.getInfoText(),
getElectionCardInfoText());
27|}
28|
29|public ElectionCardConfig getElectionCard() {

```

**Sink :**

**File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/configuration/models/ElectionCardModel.java 26

**Fact Name:** no.valg.eva.admin.frontend.configuration.models.ElectionCardModel.getElectionCardInfoText

```

23|public ElectionCardModel(ElectionCardConfig electionCard, ElectionDayPollingPlace pollingPlace) {
24|   this.electionCard = electionCard;
25|   this.pollingPlace = pollingPlace;
>26|   this.customInfoText = !isEmpty(getElectionCardInfoText()) && !StringUtils.equals(electionCard.getInfoText(),
getElectionCardInfoText());
27|}
28|
29|public ElectionCardConfig getElectionCard() {

```

**Issue 620D73D32DE0CEA999B917A93C1E73D0:****Source :**

**File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/configuration/models/ElectionCardModel.java 26

**Fact Name:** no.valg.eva.admin.frontend.configuration.models.ElectionCardModel.getElectionCardInfoText

```

23|public ElectionCardModel(ElectionCardConfig electionCard, ElectionDayPollingPlace pollingPlace) {
24|   this.electionCard = electionCard;
25|   this.pollingPlace = pollingPlace;
>26|   this.customInfoText = !isEmpty(getElectionCardInfoText()) && !StringUtils.equals(electionCard.getInfoText(),
getElectionCardInfoText());
27|}
28|
29|public ElectionCardConfig getElectionCard() {

```

**Sink :**

**File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/configuration/models/ElectionCardModel.java 26

**Fact Name:** no.valg.eva.admin.frontend.configuration.models.ElectionCardModel.getElectionCardInfoText

```

23|public ElectionCardModel(ElectionCardConfig electionCard, ElectionDayPollingPlace pollingPlace) {
24|   this.electionCard = electionCard;
25|   this.pollingPlace = pollingPlace;
>26|   this.customInfoText = !isEmpty(getElectionCardInfoText()) && !StringUtils.equals(electionCard.getInfoText(),
getElectionCardInfoText());

```

```
27|}
28|
29|public ElectionCardConfig getElectionCard() {
```

## 6.10 FinalCountBuilder.java

### Issue CF070DDB4EECC71B7A3FB71105325F53:

#### Source :

File admin-counting/src/main/java/no/valg/eva/admin/counting/builder/FinalCountBuilder.java 36

Fact Name: no.valg.eva.admin.counting.builder.FinalCountBuilder.initCount

```
33|         boolean manualCount,
34|         ReportingUnit reportingUnit) {
35|     this.reportingUnit = reportingUnit;
>36|     count = initCount(category, areaPath, areaName, manualCount);
37|} 38|
39|protected FinalCount initCount(CountCategory category, AreaPath areaPath, String areaName, boolean manualCount) {
```

#### Sink :

File admin-counting/src/main/java/no/valg/eva/admin/counting/builder/FinalCountBuilder.java 36

Fact Name: no.valg.eva.admin.counting.builder.FinalCountBuilder.initCount

```
33|         boolean manualCount,
34|         ReportingUnit reportingUnit) {
35|     this.reportingUnit = reportingUnit;
>36|     count = initCount(category, areaPath, areaName, manualCount);
37|} 38|
39|protected FinalCount initCount(CountCategory category, AreaPath areaPath, String areaName, boolean manualCount) {
```

## 6.11 IEModeFilter.java

### Issue CAC8BB9E35C76DoDCF970F28A1952CCA:

#### Source :

File admin-frontend/src/main/java/no/evote/presentation/util/filters/IEModeFilter.java 27

Fact Name: no.evote.presentation.util.filters.IEModeFilter.modeFromEvoteProperties

```
24|private final String mode;
25|
26|public IEModeFilter() {
>27|     String modeFromEvoteProperties = modeFromEvoteProperties();
28|     if (modeFromEvoteProperties == null) {
29|         this.mode = DEFAULT_MODE;
30|     } else {
```

#### Sink :

File admin-frontend/src/main/java/no/evote/presentation/util/filters/IEModeFilter.java 27

Fact Name: no.evote.presentation.util.filters.IEModeFilter.modeFromEvoteProperties

```
24|private final String mode;
25|
26|public IEModeFilter() {
>27|     String modeFromEvoteProperties = modeFromEvoteProperties();
28|     if (modeFromEvoteProperties == null) {
29|         this.mode = DEFAULT_MODE;
30|     } else {
```

## 6.12 MeldingerWidget.java

### Issue 8C5E119A006F4BE52C2610CEE6E87A45:

#### Source :

File admin-frontend/src/main/java/no/valg/eva/admin/frontend/common/MeldingerWidget.java 38 Fact

Name: no.valg.eva.admin.frontend.common.MeldingerWidget.addAll

```
37|public MeldingerWidget(List<Melding> meldinger) {
>38|     addAll(meldinger);
39|}
40|
41|@Override
```

#### Sink :

File admin-frontend/src/main/java/no/valg/eva/admin/frontend/common/MeldingerWidget.java 38 Fact

Name: no.valg.eva.admin.frontend.common.MeldingerWidget.addAll

```
37|public MeldingerWidget(List<Melding> meldinger) {
>38|     addAll(meldinger);
39|}
```

```
40|
41|@Override
```

## 6.13 Operator.java

### Issue 450CC589709125D4486055FC84F28F25:

**Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/common/rbac/Operator.java 45

**Fact Name:** no.valg.eva.admin.common.rbac.Operator.getKeySerialNumber

```
42|    super(person);
43|    if (person instanceof Operator) {
44|        active = ((Operator) person).isActive();
>45|        keySerialNumber = ((Operator) person).getKeySerialNumber();
```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/common/rbac/Operator.java 45

**Fact Name:** no.valg.eva.admin.common.rbac.Operator.getKeySerialNumber

```
42|    super(person);
43|    if (person instanceof Operator) {
44|        active = ((Operator) person).isActive();
>45|        keySerialNumber = ((Operator) person).getKeySerialNumber();
```

### Issue B69A4FEC9E6B9DCEFOA853C43BB6EEA2:

**Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/common/rbac/Operator.java 44 **Fact**

**Name:** no.valg.eva.admin.common.rbac.Operator.isActive

```
41|public Operator(Person person) {
42|    super(person);
43|    if (person instanceof Operator) {
>44|        active = ((Operator) person).isActive();
45|        keySerialNumber = ((Operator) person).getKeySerialNumber();
```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/common/rbac/Operator.java 44 **Fact**

**Name:** no.valg.eva.admin.common.rbac.Operator.isActive

```
41|public Operator(Person person) {
42|    super(person);
43|    if (person instanceof Operator) {
>44|        active = ((Operator) person).isActive();
45|        keySerialNumber = ((Operator) person).getKeySerialNumber();
```

## 6.14 ParametersDialogBean.java

### Issue 471DB08FBE673A6C471C11349B2BF8E9:

**Source :**

**File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/reports/ctrls/ParametersDialogBean.java 47

**Fact Name:** no.valg.eva.admin.frontend.reports.ctrls.ParametersDialogBean.getParameters

```
44|canonicalReportParameterParentIdMap = reportLinksController.getCanonicalReportParameterParentIdMap();
45|selectedReport = jasperReportService.getReportTemplate(userData, valghendelsesRapport); 46|
>47|getParameters().stream().filter(parameter -> parameter.isInferred() || parameter.isFixed())
48|    .forEach(parameter -> arguments.put(parameter.getId(), parameter.getDefaultValue().toString()));
49|
50|if (isAvkryssningsmanntall()) {
```

**Sink :**

**File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/reports/ctrls/ParametersDialogBean.java 47

**Fact Name:** no.valg.eva.admin.frontend.reports.ctrls.ParametersDialogBean.getParameters

```
44|canonicalReportParameterParentIdMap = reportLinksController.getCanonicalReportParameterParentIdMap();
45|selectedReport = jasperReportService.getReportTemplate(userData, valghendelsesRapport); 46|
>47|getParameters().stream().filter(parameter -> parameter.isInferred() || parameter.isFixed())
48|    .forEach(parameter -> arguments.put(parameter.getId(), parameter.getDefaultValue().toString()));
49|
50|if (isAvkryssningsmanntall()) {
```

### Issue 73EF7D936073C4BE543BA61C15303AC9:

**Source :**

**File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/reports/ctrls/ParametersDialogBean.java 50 **Fact**

**Name:** no.valg.eva.admin.frontend.reports.ctrls.ParametersDialogBean.isAvkryssningsmanntall

```
47|    getParameters().stream().filter(parameter -> parameter.isInferred() || parameter.isFixed())
48|        .forEach(parameter -> arguments.put(parameter.getId(), parameter.getDefaultValue().toString()));
49|
>50|    if (isAvkryssningsmanntall()) {
```

```
51|         countUnapprovedAdvanceVotings =
reportLinksController.getVotingService().countUnapprovedAdvanceVotings(userData, userData.getOperatorAreaPath());
```

**Sink :**

**File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/reports/ctrls/ParametersDialogBean.java 50 **Fact**  
**Name:** no.valg.eva.admin.frontend.reports.ctrls.ParametersDialogBean.isAvkryssningsmanntall

```
47|     getParameters().stream().filter(parameter -> parameter.isInferred() || parameter.isFixed())
48|         .forEach(parameter -> arguments.put(parameter.getId(), parameter.getDefaultValue().toString()));
49|
>50|     if (isAvkryssningsmanntall()) {
51|         countUnapprovedAdvanceVotings =
reportLinksController.getVotingService().countUnapprovedAdvanceVotings(userData, userData.getOperatorAreaPath());
```

## 6.15 Party.java

### Issue 14FCBoAFB129549935979FA907F957E2:

**Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Party.java 81  
**Fact Name:** no.valg.eva.admin.configuration.domain.model.Party.setId

```
78|public Party(final String id, final Integer shortCode, final PartyCategory partyCategory,
79|            final ElectionEvent electionEvent) {
80|    this.electionEvent = electionEvent;
>81|    setId(id);
82|    this.shortCode = shortCode;
83|    this.partyCategory = partyCategory;
```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Party.java 81  
**Fact Name:** no.valg.eva.admin.configuration.domain.model.Party.setId

```
78|public Party(final String id, final Integer shortCode, final PartyCategory partyCategory,
79|            final ElectionEvent electionEvent) {
80|    this.electionEvent = electionEvent;
>81|    setId(id);
82|    this.shortCode = shortCode;
83|    this.partyCategory = partyCategory;
```

## 6.16 Person.java

### Issue 70953BEF55A1BAB6E63D51B83A05A75E:

**Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/common/Person.java 33  
**Fact Name:** no.valg.eva.admin.common.Person.getPersonId

```
32|protected Person(Person person) {
>33|    this.personId = person.getPersonId();
34|    this.dateOfBirth = person.getDateOfBirth();
35|    this.firstName = requireNonNull(person.getFirstName());
36|    this.middleName = person.getMiddleName();
```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/common/Person.java 33  
**Fact Name:** no.valg.eva.admin.common.Person.getPersonId

```
32|protected Person(Person person) {
>33|    this.personId = person.getPersonId();
34|    this.dateOfBirth = person.getDateOfBirth();
35|    this.firstName = requireNonNull(person.getFirstName());
36|    this.middleName = person.getMiddleName();
```

### Issue 8936DCF928DBE141D6946E88BA667D37:

**Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/common/Person.java 37 **Fact**  
**Name:** no.valg.eva.admin.common.Person.getLastName

```
34|this.dateOfBirth = person.getDateOfBirth();
35|this.firstName = requireNonNull(person.getFirstName());
36|this.middleName = person.getMiddleName();
>37|this.lastName = requireNonNull(person.getLastName());
38|this.address = person.getAddress();
39|this.email = person.getEmail();
40|this.telephoneNumber = person.getTelephoneNumber();
```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/common/Person.java 37 **Fact**  
**Name:** no.valg.eva.admin.common.Person.getLastName

```
34|this.dateOfBirth = person.getDateOfBirth();
35|this.firstName = requireNonNull(person.getFirstName());
36|this.middleName = person.getMiddleName();
```

```

>37|this.lastName = requireNonNull(person.getLastName());
38|this.address = person.getAddress();
39|this.email = person.getEmail();
40|this.telephoneNumber = person.getTelephoneNumber();

```

**Issue B18EF9B7CAAFD289D67506AA01FBD454:****Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/common/Person.java 34 **Fact**  
Name: no.valg.eva.admin.common.Person.getDateOfBirth

```

32|protected Person(Person person) {
33|   this.personId = person.getPersonId();
>34|   this.dateOfBirth = person.getDateOfBirth();
35|   this.firstName = requireNonNull(person.getFirstName());
36|   this.middleName = person.getMiddleName();
37|   this.lastName = requireNonNull(person.getLastName());

```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/common/Person.java 34 **Fact**  
Name: no.valg.eva.admin.common.Person.getDateOfBirth

```

32|protected Person(Person person) {
33|   this.personId = person.getPersonId();
>34|   this.dateOfBirth = person.getDateOfBirth();
35|   this.firstName = requireNonNull(person.getFirstName());
36|   this.middleName = person.getMiddleName();
37|   this.lastName = requireNonNull(person.getLastName());

```

**Issue BDo6B598A87937E9BCoEE66F8FBD76D7:****Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/common/Person.java 38  
**Fact** Name: no.valg.eva.admin.common.Person.getAddress

```

35|   this.firstName = requireNonNull(person.getFirstName());
36|   this.middleName = person.getMiddleName();
37|   this.lastName = requireNonNull(person.getLastName());
>38|   this.address = person.getAddress();
39|   this.email = person.getEmail();
40|   this.telephoneNumber = person.getTelephoneNumber();

```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/common/Person.java 38  
**Fact** Name: no.valg.eva.admin.common.Person.getAddress

```

35|   this.firstName = requireNonNull(person.getFirstName());
36|   this.middleName = person.getMiddleName();
37|   this.lastName = requireNonNull(person.getLastName());
>38|   this.address = person.getAddress();
39|   this.email = person.getEmail();
40|   this.telephoneNumber = person.getTelephoneNumber();

```

**Issue CA683C6D6767B388CC4005294EE15919:****Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/common/Person.java 36  
**Fact** Name: no.valg.eva.admin.common.Person.getMiddleName

```

33|this.personId = person.getPersonId();
34|this.dateOfBirth = person.getDateOfBirth();
35|this.firstName = requireNonNull(person.getFirstName());
>36|this.middleName = person.getMiddleName();
37|this.lastName = requireNonNull(person.getLastName());
38|this.address = person.getAddress();
39|this.email = person.getEmail();

```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/common/Person.java 36  
**Fact** Name: no.valg.eva.admin.common.Person.getMiddleName

```

33|this.personId = person.getPersonId();
34|this.dateOfBirth = person.getDateOfBirth();
35|this.firstName = requireNonNull(person.getFirstName());
>36|this.middleName = person.getMiddleName();
37|this.lastName = requireNonNull(person.getLastName());
38|this.address = person.getAddress();
39|this.email = person.getEmail();

```

**Issue D30D38BB696B06F49A2480F4B3489D1E:****Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/common/Person.java 35  
**Fact** Name: no.valg.eva.admin.common.Person.getFirstName

```

32|protected Person(Person person) {
33|   this.personId = person.getPersonId();
34|   this.dateOfBirth = person.getDateOfBirth();

```

```

>35|   this.firstName = requireNonNull(person.getFirstName());
36|   this.middleName = person.getMiddleName();
37|   this.lastName = requireNonNull(person.getLastName());
38|   this.address = person.getAddress();

```

**Sink :****File** admin-common/src/main/java/no/valg/eva/admin/common/Person.java 35**Fact Name:** no.valg.eva.admin.common.Person.getFirstName

```

32|protected Person(Person person) {
33|   this.personId = person.getPersonId();
34|   this.dateOfBirth = person.getDateOfBirth();
>35|   this.firstName = requireNonNull(person.getFirstName());
36|   this.middleName = person.getMiddleName();
37|   this.lastName = requireNonNull(person.getLastName());
38|   this.address = person.getAddress();

```

**Issue DB35B28C2CB8815AC897B3FEBF810361:****Source :****File** admin-common/src/main/java/no/valg/eva/admin/common/Person.java 39**Fact Name:** no.valg.eva.admin.common.Person.getEmail

```

36|   this.middleName = person.getMiddleName();
37|   this.lastName = requireNonNull(person.getLastName());
38|   this.address = person.getAddress();
>39|   this.email = person.getEmail();
40|   this.telephoneNumber = person.getTelephoneNumber();

```

**Sink :****File** admin-common/src/main/java/no/valg/eva/admin/common/Person.java 39**Fact Name:** no.valg.eva.admin.common.Person.getEmail

```

36|   this.middleName = person.getMiddleName();
37|   this.lastName = requireNonNull(person.getLastName());
38|   this.address = person.getAddress();
>39|   this.email = person.getEmail();
40|   this.telephoneNumber = person.getTelephoneNumber();

```

**Issue E569FDC178AE4E25D3A1BF2F04981F13:****Source :****File** admin-common/src/main/java/no/valg/eva/admin/common/Person.java 40**Fact Name:** no.valg.eva.admin.common.Person.getTelephoneNumber

```

37|   this.lastName = requireNonNull(person.getLastName());
38|   this.address = person.getAddress();
39|   this.email = person.getEmail();
>40|   this.telephoneNumber = person.getTelephoneNumber();
41|}
42|
43|public String nameLine() {

```

**Sink :****File** admin-common/src/main/java/no/valg/eva/admin/common/Person.java 40**Fact Name:** no.valg.eva.admin.common.Person.getTelephoneNumber

```

37|   this.lastName = requireNonNull(person.getLastName());
38|   this.address = person.getAddress();
39|   this.email = person.getEmail();
>40|   this.telephoneNumber = person.getTelephoneNumber();
41|}
42|
43|public String nameLine() {

```

## 6.17 Proposer.java

**Issue 2CB283A979518794865A15610DE966F3:****Source :****File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Proposer.java 107**Fact Name:** no.valg.eva.admin.configuration.domain.model.Proposer.getTelephoneNumber

```

104|   postalCode = proposer.getPostalCode();
105|   postTown = proposer.getPostTown();
106|   email = proposer.getEmail();
>107|   telephoneNumber = proposer.getTelephoneNumber();
108|   approved = proposer.isApproved();

```

**Sink :****File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Proposer.java 107**Fact Name:** no.valg.eva.admin.configuration.domain.model.Proposer.getTelephoneNumber

```

104|   postalCode = proposer.getPostalCode();
105|   postTown = proposer.getPostTown();
106|   email = proposer.getEmail();

```

```

107|     telephoneNumber = proposer.getTelephoneNumber();
108|     approved = proposer.isApproved();

```

**Issue 334A7F990CE48FD719CA52B4715D13E7:****Source :****File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Proposer.java 108**Fact Name:** no.valg.eva.admin.configuration.domain.model.Proposer.isApproved

```

105|     postTown = proposer.getPostTown();
106|     email = proposer.getEmail();
107|     telephoneNumber = proposer.getTelephoneNumber();
>108|     approved = proposer.isApproved();
109| }
110|
111| @ManyToMany(fetch = FetchType.EAGER)

```

**Sink :****File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Proposer.java 108**Fact Name:** no.valg.eva.admin.configuration.domain.model.Proposer.isApproved

```

105|     postTown = proposer.getPostTown();
106|     email = proposer.getEmail();
107|     telephoneNumber = proposer.getTelephoneNumber();
>108|     approved = proposer.isApproved();
109| }
110|
111| @ManyToMany(fetch = FetchType.EAGER)

```

**Issue 3D794E1D5967F9BBCCD4745D8C5C6EB8:****Source :****File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Proposer.java 92 **Fact Name:**

no.valg.eva.admin.configuration.domain.model.Proposer.getBallot

```

90| public Proposer(Proposer proposer) {
91|     super();
>92|     ballot = proposer.getBallot();
93|     displayOrder = proposer.getDisplayOrder(); 94|     id = proposer.getId();
95|     proposerRole = proposer.getProposerRole();

```

**Sink :****File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Proposer.java 92**Fact Name:** no.valg.eva.admin.configuration.domain.model.Proposer.getBallot

```

90| public Proposer(Proposer proposer) {
91|     super();
>92|     ballot = proposer.getBallot();
93|     displayOrder = proposer.getDisplayOrder(); 94|     id = proposer.getId();
95|     proposerRole = proposer.getProposerRole();

```

**Issue 4398FBC6C5301C0B12F7ED69CCA1DA64:****Source :****File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Proposer.java 106**Fact Name:** no.valg.eva.admin.configuration.domain.model.Proposer.getEmail

```

103|     addressLine3 = proposer.getAddressLine3();
104|     postalCode = proposer.getPostalCode();
105|     postTown = proposer.getPostTown();
>106|     email = proposer.getEmail();
107|     telephoneNumber = proposer.getTelephoneNumber();
108|     approved = proposer.isApproved();

```

**Sink :****File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Proposer.java 106**Fact Name:** no.valg.eva.admin.configuration.domain.model.Proposer.getEmail

```

103|     addressLine3 = proposer.getAddressLine3();
104|     postalCode = proposer.getPostalCode();
105|     postTown = proposer.getPostTown();
>106|     email = proposer.getEmail();
107|     telephoneNumber = proposer.getTelephoneNumber();
108|     approved = proposer.isApproved();

```

**Issue 5B499E1F51CA932C9A8D1D1FB1063FEF:****Source :****File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Proposer.java 97**Fact Name:** no.valg.eva.admin.configuration.domain.model.Proposer.getNameLine

```

94| id = proposer.getId();
95| proposerRole = proposer.getProposerRole();
96| dateOfBirth = proposer.getDateOfBirth();
> 97| nameLine = proposer.getNameLine();
98| firstName = proposer.getFirstName();
99| middleName = proposer.getMiddleName();
100| lastName = proposer.getLastName();

```

**Sink :****File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Proposer.java 97



**Fact Name:** no.valg.eva.admin.configuration.domain.model.Proposer.getNameLine

```

94|id = proposer.getId();
95|proposerRole = proposer.getProposerRole();
96|dateOfBirth = proposer.getDateOfBirth();
> 97|nameLine = proposer.getNameLine();
98|firstName = proposer.getFirstName();
99|middleName = proposer.getMiddleName();
100|lastName = proposer.getLastName();

```

### Issue 5E13051CoE868002DFo48E321A788D15:

**Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Proposer.java 94

**Fact Name:** no.valg.eva.admin.configuration.domain.model.Proposer.getId

```

91|super();
92|ballot = proposer.getBallot();
93|displayOrder = proposer.getDisplayOrder();
>94|id = proposer.getId();
95|proposerRole = proposer.getProposerRole();
96|dateOfBirth = proposer.getDateOfBirth();
97|nameLine = proposer.getNameLine();

```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Proposer.java 94

**Fact Name:** no.valg.eva.admin.configuration.domain.model.Proposer.getId

```

91|super();
92|ballot = proposer.getBallot();
93|displayOrder = proposer.getDisplayOrder();
>94|id = proposer.getId();
95|proposerRole = proposer.getProposerRole();
96|dateOfBirth = proposer.getDateOfBirth();
97|nameLine = proposer.getNameLine();

```

### Issue 6B8FB75B983541F11DDBAA2FC69FBFB1:

**Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Proposer.java 95

**Fact Name:** no.valg.eva.admin.configuration.domain.model.Proposer.getProposerRole

```

92|ballot = proposer.getBallot();
93|displayOrder = proposer.getDisplayOrder(); 94|id = proposer.getId();
>95|proposerRole = proposer.getProposerRole();
96|dateOfBirth = proposer.getDateOfBirth();
97|nameLine = proposer.getNameLine();
98|firstName = proposer.getFirstName();

```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Proposer.java 95

**Fact Name:** no.valg.eva.admin.configuration.domain.model.Proposer.getProposerRole

```

92|ballot = proposer.getBallot();
93|displayOrder = proposer.getDisplayOrder();
94|id = proposer.getId();
>95|proposerRole = proposer.getProposerRole();
96|dateOfBirth = proposer.getDateOfBirth();
97|nameLine = proposer.getNameLine();
98|firstName = proposer.getFirstName();

```

### Issue 74A6501DD290544BE8E12B752CCB7E61:

**Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Proposer.java 93

**Fact Name:** no.valg.eva.admin.configuration.domain.model.Proposer.getDisplayOrder

```

90|public Proposer(Proposer proposer) {
91|    super();
92|    ballot = proposer.getBallot();
>93|    displayOrder = proposer.getDisplayOrder(); 94|    id = proposer.getId();
95|    proposerRole = proposer.getProposerRole();
96|    dateOfBirth = proposer.getDateOfBirth();

```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Proposer.java 93

**Fact Name:** no.valg.eva.admin.configuration.domain.model.Proposer.getDisplayOrder

```

90|public Proposer(Proposer proposer) {
91|    super();
92|    ballot = proposer.getBallot();
>93|    displayOrder = proposer.getDisplayOrder(); 94|    id = proposer.getId();
95|    proposerRole = proposer.getProposerRole(); 96|    dateOfBirth = proposer.getDateOfBirth();

```

### Issue 88FB5DB12571C00AF8EC4F513C89C599:

**Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Proposer.java 98

**Fact Name:** no.valg.eva.admin.configuration.domain.model.Proposer.getFirstName

```

95|proposerRole = proposer.getProposerRole();
96|dateOfBirth = proposer.getDateOfBirth();
97|nameLine = proposer.getNameLine();
> 98|firstName = proposer.getFirstName();
99|middleName = proposer.getMiddleName();
100|lastName = proposer.getLastName();
101|addressLine1 = proposer.getAddressLine1();

```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Proposer.java 98

**Fact Name:** no.valg.eva.admin.configuration.domain.model.Proposer.getFirstName

```

95|proposerRole = proposer.getProposerRole();
96|dateOfBirth = proposer.getDateOfBirth();
97|nameLine = proposer.getNameLine();
> 98|firstName = proposer.getFirstName();
99|middleName = proposer.getMiddleName();
100|lastName = proposer.getLastName();
101|addressLine1 = proposer.getAddressLine1();

```

**Issue 8AA4D8ADF46BBFCE78AAEA75D7FB1C81:****Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Proposer.java 104 **Fact**

**Name:** no.valg.eva.admin.configuration.domain.model.Proposer.getPostalCode

```

101|addressLine1 = proposer.getAddressLine1();
102|addressLine2 = proposer.getAddressLine2();
103|addressLine3 = proposer.getAddressLine3();
>104|postalCode = proposer.getPostalCode();
105|postTown = proposer.getPostTown();
106|email = proposer.getEmail();
107|telephoneNumber = proposer.getTelephoneNumber();

```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Proposer.java 104 **Fact**

**Name:** no.valg.eva.admin.configuration.domain.model.Proposer.getPostalCode

```

101|addressLine1 = proposer.getAddressLine1();
102|addressLine2 = proposer.getAddressLine2();
103|addressLine3 = proposer.getAddressLine3();
>104|postalCode = proposer.getPostalCode();
105|postTown = proposer.getPostTown();
106|email = proposer.getEmail();
107|telephoneNumber = proposer.getTelephoneNumber();

```

**Issue 9241915B07A78EF4AA7B21562C5E3607:****Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Proposer.java 102

**Fact Name:** no.valg.eva.admin.configuration.domain.model.Proposer.getAddressLine2

```

99|middleName = proposer.getMiddleName();
100|lastName = proposer.getLastName();
101|addressLine1 = proposer.getAddressLine1();
>102|addressLine2 = proposer.getAddressLine2();
103|addressLine3 = proposer.getAddressLine3();
104|postalCode = proposer.getPostalCode();
105|postTown = proposer.getPostTown();

```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Proposer.java 102

**Fact Name:** no.valg.eva.admin.configuration.domain.model.Proposer.getAddressLine2

```

99|middleName = proposer.getMiddleName();
100|lastName = proposer.getLastName();
101|addressLine1 = proposer.getAddressLine1();
>102|addressLine2 = proposer.getAddressLine2();
103|addressLine3 = proposer.getAddressLine3();
104|postalCode = proposer.getPostalCode();
105|postTown = proposer.getPostTown();

```

**Issue A5026A792F24092EC15E0AA6429C36C2:****Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Proposer.java 103

**Fact Name:** no.valg.eva.admin.configuration.domain.model.Proposer.getAddressLine3

```

100|lastName = proposer.getLastName();
101|addressLine1 = proposer.getAddressLine1();
102|addressLine2 = proposer.getAddressLine2();
>103|addressLine3 = proposer.getAddressLine3();
104|postalCode = proposer.getPostalCode();
105|postTown = proposer.getPostTown();
106|email = proposer.getEmail();

```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Proposer.java 103

**Fact Name:** no.valg.eva.admin.configuration.domain.model.Proposer.getAddressLine3

```

100|lastName = proposer.getLastName();
101|addressLine1 = proposer.getAddressLine1();
102|addressLine2 = proposer.getAddressLine2();
>103|addressLine3 = proposer.getAddressLine3();
104|postalCode = proposer.getPostalCode();
105|postTown = proposer.getPostTown();
106|email = proposer.getEmail();

```

**Issue AB0C91B4B2197A26C0F398A38B1BFF71:****Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Proposer.java 105 **Fact**  
**Name:** no.valg.eva.admin.configuration.domain.model.Proposer.getPostTown

```

102|addressLine2 = proposer.getAddressLine2();
103|addressLine3 = proposer.getAddressLine3();
104|postalCode = proposer.getPostalCode();
>105|postTown = proposer.getPostTown();
106|email = proposer.getEmail();
107|telephoneNumber = proposer.getTelephoneNumber();
108|approved = proposer.isApproved();

```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Proposer.java 105 **Fact**  
**Name:** no.valg.eva.admin.configuration.domain.model.Proposer.getPostTown

```

102|addressLine2 = proposer.getAddressLine2();
103|addressLine3 = proposer.getAddressLine3();
104|postalCode = proposer.getPostalCode();
>105|postTown = proposer.getPostTown();
106|email = proposer.getEmail();
107|telephoneNumber = proposer.getTelephoneNumber();
108|approved = proposer.isApproved();

```

**Issue C63C2308E3812AC87CE1E660432EE5F5:****Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Proposer.java 100 **Fact**  
**Name:** no.valg.eva.admin.configuration.domain.model.Proposer.getLastName

```

97|nameLine = proposer.getNameLine();
98|firstName = proposer.getFirstName();
99|middleName = proposer.getMiddleName();
>100|lastName = proposer.getLastName();
101|addressLine1 = proposer.getAddressLine1();
102|addressLine2 = proposer.getAddressLine2();
103|addressLine3 = proposer.getAddressLine3();

```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Proposer.java 100 **Fact**  
**Name:** no.valg.eva.admin.configuration.domain.model.Proposer.getLastName

```

97|nameLine = proposer.getNameLine();
98|firstName = proposer.getFirstName();
99|middleName = proposer.getMiddleName();
>100|lastName = proposer.getLastName();
101|addressLine1 = proposer.getAddressLine1();
102|addressLine2 = proposer.getAddressLine2();
103|addressLine3 = proposer.getAddressLine3();

```

**Issue CD675D8354853002F069CA37C6A57DF9:****Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Proposer.java 99 **Fact**  
**Name:** no.valg.eva.admin.configuration.domain.model.Proposer.getMiddleName

```

96|dateOfBirth = proposer.getDateOfBirth();
97|nameLine = proposer.getNameLine();
98|firstName = proposer.getFirstName();
> 99|middleName = proposer.getMiddleName();
100|lastName = proposer.getLastName();
101|addressLine1 = proposer.getAddressLine1();
102|addressLine2 = proposer.getAddressLine2();

```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Proposer.java 99 **Fact**  
**Name:** no.valg.eva.admin.configuration.domain.model.Proposer.getMiddleName

```

96|dateOfBirth = proposer.getDateOfBirth();
97|nameLine = proposer.getNameLine();
98|firstName = proposer.getFirstName();
> 99|middleName = proposer.getMiddleName();
100|lastName = proposer.getLastName();
101|addressLine1 = proposer.getAddressLine1();
102|addressLine2 = proposer.getAddressLine2();

```

**Issue DD56ACB4D6F641A27C990BFCBACoAA2A:**

**Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Proposer.java 101  
**Fact Name:** no.valg.eva.admin.configuration.domain.model.Proposer.getAddressLine1

```

98|firstName = proposer.getFirstName();
99|middleName = proposer.getMiddleName();
100|lastName = proposer.getLastName();
>101|addressLine1 = proposer.getAddressLine1();
102|addressLine2 = proposer.getAddressLine2();
103|addressLine3 = proposer.getAddressLine3();
104|postalCode = proposer.getPostalCode();

```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Proposer.java 101  
**Fact Name:** no.valg.eva.admin.configuration.domain.model.Proposer.getAddressLine1

```

98|firstName = proposer.getFirstName();
99|middleName = proposer.getMiddleName();
100|lastName = proposer.getLastName();
>101|addressLine1 = proposer.getAddressLine1();
102|addressLine2 = proposer.getAddressLine2();
103|addressLine3 = proposer.getAddressLine3();
104|postalCode = proposer.getPostalCode();

```

**Issue EEE74CFC27E329FC43FF870A1DC5DF69:****Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Proposer.java 96  
**Fact Name:** no.valg.eva.admin.configuration.domain.model.Proposer.getDateOfBirth

```

93|displayOrder = proposer.getDisplayOrder(); 94|id = proposer.getId();
95|proposerRole = proposer.getProposerRole();
>96|dateOfBirth = proposer.getDateOfBirth();
97|nameLine = proposer.getNameLine();
98|firstName = proposer.getFirstName();
99|middleName = proposer.getMiddleName();

```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Proposer.java 96  
**Fact Name:** no.valg.eva.admin.configuration.domain.model.Proposer.getDateOfBirth

```

93|displayOrder = proposer.getDisplayOrder(); 94|id = proposer.getId();
95|proposerRole = proposer.getProposerRole();
>96|dateOfBirth = proposer.getDateOfBirth();
97|nameLine = proposer.getNameLine();
98|firstName = proposer.getFirstName();
99|middleName = proposer.getMiddleName();

```

## 6.18 RejectedBallot.java

**Issue 3113626C658B1D970786E7560F19BD1B:****Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/common/counting/model/RejectedBallot.java 21 **Fact**  
**Name:** no.valg.eva.admin.common.counting.model.RejectedBallot.getBallotRejectionId

```

19|public RejectedBallot(RejectedBallot rejectedBallot) {
20|    this.id = rejectedBallot.getId();
>21|    this.ballotRejectionId = rejectedBallot.getBallotRejectionId();
22|    this.state = rejectedBallot.getState();
23|    this.selectedBallotRejectionId = rejectedBallot.getSelectedBallotRejectionId(); 24|    this.selectedBallotId
= rejectedBallot.getSelectedBallotId();

```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/common/counting/model/RejectedBallot.java 21 **Fact**  
**Name:** no.valg.eva.admin.common.counting.model.RejectedBallot.getBallotRejectionId

```

19|public RejectedBallot(RejectedBallot rejectedBallot) {
20|    this.id = rejectedBallot.getId();
>21|    this.ballotRejectionId = rejectedBallot.getBallotRejectionId();
22|    this.state = rejectedBallot.getState();
23|    this.selectedBallotRejectionId = rejectedBallot.getSelectedBallotRejectionId();
24|    this.selectedBallotId = rejectedBallot.getSelectedBallotId();

```

**Issue 345CC2F1DCDC5AF715BD124727437D37:****Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/common/counting/model/RejectedBallot.java 22 **Fact**  
**Name:** no.valg.eva.admin.common.counting.model.RejectedBallot.getState

```

19|public RejectedBallot(RejectedBallot rejectedBallot) {
20|    this.id = rejectedBallot.getId();
21|    this.ballotRejectionId = rejectedBallot.getBallotRejectionId();
>22|    this.state = rejectedBallot.getState();
23|    this.selectedBallotRejectionId = rejectedBallot.getSelectedBallotRejectionId(); 24|    this.selectedBallotId
= rejectedBallot.getSelectedBallotId();

```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/common/counting/model/RejectedBallot.java 22 **Fact**  
**Name:** no.valg.eva.admin.common.counting.model.RejectedBallot.getState

```
19|public RejectedBallot(RejectedBallot rejectedBallot) {
20|    this.id = rejectedBallot.getId();
21|    this.ballotRejectionId = rejectedBallot.getBallotRejectionId();
>22|    this.state = rejectedBallot.getState();
23|    this.selectedBallotRejectionId = rejectedBallot.getSelectedBallotRejectionId();
24|    this.selectedBallotId = rejectedBallot.getSelectedBallotId();
```

**Issue AFDB9159B8197432EDFC0D9A0A321A75:****Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/common/counting/model/RejectedBallot.java 23  
**Fact Name:** no.valg.eva.admin.common.counting.model.RejectedBallot.getSelectedBallotRejectionId

```
20|    this.id = rejectedBallot.getId();
21|    this.ballotRejectionId = rejectedBallot.getBallotRejectionId();
22|    this.state = rejectedBallot.getState();
>23|    this.selectedBallotRejectionId = rejectedBallot.getSelectedBallotRejectionId(); 24|    this.selectedBallotId
= rejectedBallot.getSelectedBallotId();
```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/common/counting/model/RejectedBallot.java 23  
**Fact Name:** no.valg.eva.admin.common.counting.model.RejectedBallot.getSelectedBallotRejectionId

```
20|    this.id = rejectedBallot.getId();
21|    this.ballotRejectionId = rejectedBallot.getBallotRejectionId();
22|    this.state = rejectedBallot.getState();
>23|    this.selectedBallotRejectionId = rejectedBallot.getSelectedBallotRejectionId();
24|    this.selectedBallotId = rejectedBallot.getSelectedBallotId();
```

**Issue B6BFEDFB4891C14CD899291BB642AD95:****Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/common/counting/model/RejectedBallot.java 24  
**Fact Name:** no.valg.eva.admin.common.counting.model.RejectedBallot.getSelectedBallotId

```
21|    this.ballotRejectionId = rejectedBallot.getBallotRejectionId();
22|    this.state = rejectedBallot.getState();
23|    this.selectedBallotRejectionId = rejectedBallot.getSelectedBallotRejectionId();
>24|    this.selectedBallotId = rejectedBallot.getSelectedBallotId();
25|}
26|
27|public RejectedBallot(String id, String ballotRejectionId) {
```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/common/counting/model/RejectedBallot.java 24  
**Fact Name:** no.valg.eva.admin.common.counting.model.RejectedBallot.getSelectedBallotId

```
21|    this.ballotRejectionId = rejectedBallot.getBallotRejectionId();
22|    this.state = rejectedBallot.getState();
23|    this.selectedBallotRejectionId = rejectedBallot.getSelectedBallotRejectionId();
>24|    this.selectedBallotId = rejectedBallot.getSelectedBallotId();
25|}
26|
27|public RejectedBallot(String id, String ballotRejectionId) {
```

**Issue F9208BD5DECD90FB54588FA43042C0C5:****Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/common/counting/model/RejectedBallot.java 20  
**Fact Name:** no.valg.eva.admin.common.counting.model.RejectedBallot.getId

```
17|private String selectedBallotId;
18|
19|public RejectedBallot(RejectedBallot rejectedBallot) {
>20|    this.id = rejectedBallot.getId();
21|    this.ballotRejectionId = rejectedBallot.getBallotRejectionId();
22|    this.state = rejectedBallot.getState();
23|    this.selectedBallotRejectionId = rejectedBallot.getSelectedBallotRejectionId();
```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/common/counting/model/RejectedBallot.java 20  
**Fact Name:** no.valg.eva.admin.common.counting.model.RejectedBallot.getId

```
17|private String selectedBallotId;
18|
19|public RejectedBallot(RejectedBallot rejectedBallot) {
>20|    this.id = rejectedBallot.getId();
21|    this.ballotRejectionId = rejectedBallot.getBallotRejectionId();
22|    this.state = rejectedBallot.getState();
23|    this.selectedBallotRejectionId = rejectedBallot.getSelectedBallotRejectionId();
```

## 6.19 ReportingUnit.java

**Issue 1E867DoE70F8CF78B03BADD44AAAEBoB:****Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/ReportingUnit.java 113  
**Fact Name:** no.valg.eva.admin.configuration.domain.model.ReportingUnit.getAddressLine2

```

111|nameLine = reportingUnit.getNameLine();
112|addressLine1 = reportingUnit.getAddressLine1();
>113|addressLine2 = reportingUnit.getAddressLine2();
114|addressLine3 = reportingUnit.getAddressLine3();
115|postalCode = reportingUnit.getPostalCode();
116|postTown = reportingUnit.getPostTown();

```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/ReportingUnit.java 113

**Fact Name:** no.valg.eva.admin.configuration.domain.model.ReportingUnit.getAddressLine2

```

111|nameLine = reportingUnit.getNameLine();
112|addressLine1 = reportingUnit.getAddressLine1();
>113|addressLine2 = reportingUnit.getAddressLine2(); 114|addressLine3 = reportingUnit.getAddressLine3();
115|postalCode = reportingUnit.getPostalCode();
116|postTown = reportingUnit.getPostTown();

```

**Issue 5B6DDA76148617AAE4855A4C8E1A92BB:****Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/ReportingUnit.java 118

**Fact Name:** no.valg.eva.admin.configuration.domain.model.ReportingUnit.getTelephoneNumber

```

115| postalCode = reportingUnit.getPostalCode();
116| postTown = reportingUnit.getPostTown();
117| email = reportingUnit.getEmail();
>118| telephoneNumber = reportingUnit.getTelephoneNumber();
119| reportingUnitType = reportingUnit.getReportingUnitType();

```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/ReportingUnit.java 118

**Fact Name:** no.valg.eva.admin.configuration.domain.model.ReportingUnit.getTelephoneNumber

```

115| postalCode = reportingUnit.getPostalCode();
116| postTown = reportingUnit.getPostTown();
117| email = reportingUnit.getEmail();
>118| telephoneNumber = reportingUnit.getTelephoneNumber();
119| reportingUnitType = reportingUnit.getReportingUnitType();

```

**Issue 6C9E0B1A0ACDB373641BBAF5E4C1753B:****Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/ReportingUnit.java 112

**Fact Name:** no.valg.eva.admin.configuration.domain.model.ReportingUnit.getAddressLine1

```

109|this.mvElection = mvElection;
110|
111|nameLine = reportingUnit.getNameLine();
>112|addressLine1 = reportingUnit.getAddressLine1();
113|addressLine2 = reportingUnit.getAddressLine2();
114|addressLine3 = reportingUnit.getAddressLine3();
115|postalCode = reportingUnit.getPostalCode();

```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/ReportingUnit.java 112

**Fact Name:** no.valg.eva.admin.configuration.domain.model.ReportingUnit.getAddressLine1

```

109|this.mvElection = mvElection;
110|
111|nameLine = reportingUnit.getNameLine();
>112|addressLine1 = reportingUnit.getAddressLine1();
113|addressLine2 = reportingUnit.getAddressLine2();
114|addressLine3 = reportingUnit.getAddressLine3();
115|postalCode = reportingUnit.getPostalCode();

```

**Issue 72BCA3E74C8E716F602FEA9C4B88E3AE:****Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/ReportingUnit.java 116

**Fact Name:** no.valg.eva.admin.configuration.domain.model.ReportingUnit.getPostTown

```

113|addressLine2 = reportingUnit.getAddressLine2();
114|addressLine3 = reportingUnit.getAddressLine3();
115|postalCode = reportingUnit.getPostalCode();
>116|postTown = reportingUnit.getPostTown();
117|email = reportingUnit.getEmail();
118|telephoneNumber = reportingUnit.getTelephoneNumber();
119|reportingUnitType = reportingUnit.getReportingUnitType();

```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/ReportingUnit.java 116

**Fact Name:** no.valg.eva.admin.configuration.domain.model.ReportingUnit.getPostTown

```

113|addressLine2 = reportingUnit.getAddressLine2();
114|addressLine3 = reportingUnit.getAddressLine3();
115|postalCode = reportingUnit.getPostalCode();
>116|postTown = reportingUnit.getPostTown();
117|email = reportingUnit.getEmail();

```

```
118|telephoneNumber = reportingUnit.getTelephoneNumber();
119|reportingUnitType = reportingUnit.getReportingUnitType();
```

**Issue 7436FA2C6DBA3390C0974A0F2D31A09B:****Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/ReportingUnit.java 117

**Fact Name:** no.valg.eva.admin.configuration.domain.model.ReportingUnit.getEmail

```
114|    addressLine3 = reportingUnit.getAddressLine3();
115|    postalCode = reportingUnit.getPostalCode();
116|    postTown = reportingUnit.getPostTown();
>117|    email = reportingUnit.getEmail();
118|    telephoneNumber = reportingUnit.getTelephoneNumber();
119|    reportingUnitType = reportingUnit.getReportingUnitType();
```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/ReportingUnit.java 117

**Fact Name:** no.valg.eva.admin.configuration.domain.model.ReportingUnit.getEmail

```
114|    addressLine3 = reportingUnit.getAddressLine3();
115|    postalCode = reportingUnit.getPostalCode();
116|    postTown = reportingUnit.getPostTown();
>117|    email = reportingUnit.getEmail();
118|    telephoneNumber = reportingUnit.getTelephoneNumber();
119|    reportingUnitType = reportingUnit.getReportingUnitType();
```

**Issue 8A3D74FBBF255FB076274A169113C681:****Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/ReportingUnit.java 111

**Fact Name:** no.valg.eva.admin.configuration.domain.model.ReportingUnit.getNameLine

```
108|this.mvArea = mvArea;
109|this.mvElection = mvElection;
110|
>111|nameLine = reportingUnit.getNameLine();
112|addressLine1 = reportingUnit.getAddressLine1();
113|addressLine2 = reportingUnit.getAddressLine2();
114|addressLine3 = reportingUnit.getAddressLine3();
```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/ReportingUnit.java 111

**Fact Name:** no.valg.eva.admin.configuration.domain.model.ReportingUnit.getNameLine

```
108|this.mvArea = mvArea;
109|this.mvElection = mvElection;
110|
>111|nameLine = reportingUnit.getNameLine();
112|addressLine1 = reportingUnit.getAddressLine1();
113|addressLine2 = reportingUnit.getAddressLine2();
114|addressLine3 = reportingUnit.getAddressLine3();
```

**Issue 9ABF980D55870E40F718980DDC56667C:****Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/ReportingUnit.java 114

**Fact Name:** no.valg.eva.admin.configuration.domain.model.ReportingUnit.getAddressLine3

```
111|nameLine = reportingUnit.getNameLine();
112|addressLine1 = reportingUnit.getAddressLine1();
113|addressLine2 = reportingUnit.getAddressLine2();
>114|addressLine3 = reportingUnit.getAddressLine3();
115|postalCode = reportingUnit.getPostalCode();
116|postTown = reportingUnit.getPostTown();
117|email = reportingUnit.getEmail();
```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/ReportingUnit.java 114

**Fact Name:** no.valg.eva.admin.configuration.domain.model.ReportingUnit.getAddressLine3

```
111|nameLine = reportingUnit.getNameLine();
112|addressLine1 = reportingUnit.getAddressLine1();
113|addressLine2 = reportingUnit.getAddressLine2();
>114|addressLine3 = reportingUnit.getAddressLine3();
115|postalCode = reportingUnit.getPostalCode();
116|postTown = reportingUnit.getPostTown();
117|email = reportingUnit.getEmail();
```

**Issue A9B835394E207E79FC74C3626151DA4A:****Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/ReportingUnit.java 115

**Fact Name:** no.valg.eva.admin.configuration.domain.model.ReportingUnit.getPostalCode

```
112|addressLine1 = reportingUnit.getAddressLine1();
113|addressLine2 = reportingUnit.getAddressLine2();
114|addressLine3 = reportingUnit.getAddressLine3();
```

```

>115|postalCode = reportingUnit.getPostalCode();
116|postTown = reportingUnit.getPostTown();
117|email = reportingUnit.getEmail();
118|telephoneNumber = reportingUnit.getTelephoneNumber();

```

**Sink :****File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/ReportingUnit.java 115**Fact Name:** no.valg.eva.admin.configuration.domain.model.ReportingUnit.getPostalCode

```

112|addressLine1 = reportingUnit.getAddressLine1();
113|addressLine2 = reportingUnit.getAddressLine2();
114|addressLine3 = reportingUnit.getAddressLine3();
>115|postalCode = reportingUnit.getPostalCode();
116|postTown = reportingUnit.getPostTown();
117|email = reportingUnit.getEmail();
118|telephoneNumber = reportingUnit.getTelephoneNumber();

```

**Issue E97E93B7060CCE515026CE7E077CAF9C:****Source :****File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/ReportingUnit.java 119**Fact Name:** no.valg.eva.admin.configuration.domain.model.ReportingUnit.getReportingUnitType

```

116| postTown = reportingUnit.getPostTown();
117| email = reportingUnit.getEmail();
118| telephoneNumber = reportingUnit.getTelephoneNumber();
>119| reportingUnitType = reportingUnit.getReportingUnitType();
120|}
121|
122|@ManyToOne(fetch = FetchType.EAGER)

```

**Sink :****File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/ReportingUnit.java 119**Fact Name:** no.valg.eva.admin.configuration.domain.model.ReportingUnit.getReportingUnitType

```

116| postTown = reportingUnit.getPostTown();
117| email = reportingUnit.getEmail();
118| telephoneNumber = reportingUnit.getTelephoneNumber();
>119| reportingUnitType = reportingUnit.getReportingUnitType();
120|}
121|
122|@ManyToOne(fetch = FetchType.EAGER)

```

## 6.20 ResponsibleOfficer.java

**Issue 01C030B63F3158015A42158C2E8EFDEA:****Source :****File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/ResponsibleOfficer.java 95**Fact Name:** no.valg.eva.admin.configuration.domain.model.ResponsibleOfficer.getPostTown

```

92|this.addressLine2 = respOfficer.getAddressLine2();
93|this.addressLine3 = respOfficer.getAddressLine3();
94|this.postalCode = respOfficer.getPostalCode();
>95|this.postTown = respOfficer.getPostTown();
96|this.email = respOfficer.getEmail();
97|this.telephoneNumber = respOfficer.getTelephoneNumber();

```

**Sink :****File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/ResponsibleOfficer.java 95**Fact Name:** no.valg.eva.admin.configuration.domain.model.ResponsibleOfficer.getPostTown

```

92|this.addressLine2 = respOfficer.getAddressLine2();
93|this.addressLine3 = respOfficer.getAddressLine3();
94|this.postalCode = respOfficer.getPostalCode();
>95|this.postTown = respOfficer.getPostTown();
96|this.email = respOfficer.getEmail();
97|this.telephoneNumber = respOfficer.getTelephoneNumber();

```

**Issue 42B0419497C069CD6C04DD9F2ECCEC1E:****Source :****File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/ResponsibleOfficer.java 87**Fact Name:** no.valg.eva.admin.configuration.domain.model.ResponsibleOfficer.getNameLine

```

84|this.reportingUnit = respOfficer.getReportingUnit();
85|this.responsibility = respOfficer.getResponsibility();
86|this.displayOrder = respOfficer.getDisplayOrder();
>87|this.nameLine = respOfficer.getNameLine();
88|this.firstName = respOfficer.getFirstName();
89|this.middleName = respOfficer.getMiddleName();
90|this.lastName = respOfficer.getLastName();

```

**Sink :****File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/ResponsibleOfficer.java 87**Fact Name:** no.valg.eva.admin.configuration.domain.model.ResponsibleOfficer.getNameLine



```

84|this.reportingUnit = respOfficer.getReportingUnit();
85|this.responsibility = respOfficer.getResponsibility();
86|this.displayOrder = respOfficer.getDisplayOrder();
>87|this.nameLine = respOfficer.getNameLine();
88|this.firstName = respOfficer.getFirstName();
89|this.middleName = respOfficer.getMiddleName();
90|this.lastName = respOfficer.getLastName();

```

**Issue 4E51EB4DF15242D8794B8D1605639DB1:****Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/ResponsibleOfficer.java 91

**Fact Name:** no.valg.eva.admin.configuration.domain.model.ResponsibleOfficer.getAddressLine1

```

88|this.firstName = respOfficer.getFirstName();
89|this.middleName = respOfficer.getMiddleName();
90|this.lastName = respOfficer.getLastName();
>91|this.addressLine1 = respOfficer.getAddressLine1();
92|this.addressLine2 = respOfficer.getAddressLine2();
93|this.addressLine3 = respOfficer.getAddressLine3();
94|this.postalCode = respOfficer.getPostalCode();

```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/ResponsibleOfficer.java 91

**Fact Name:** no.valg.eva.admin.configuration.domain.model.ResponsibleOfficer.getAddressLine1

```

88|this.firstName = respOfficer.getFirstName();
89|this.middleName = respOfficer.getMiddleName();
90|this.lastName = respOfficer.getLastName();
>91|this.addressLine1 = respOfficer.getAddressLine1();
92|this.addressLine2 = respOfficer.getAddressLine2();
93|this.addressLine3 = respOfficer.getAddressLine3();
94|this.postalCode = respOfficer.getPostalCode();

```

**Issue 5959C5747010CA2954C90E88309DoB27:****Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/ResponsibleOfficer.java 97

**Fact Name:** no.valg.eva.admin.configuration.domain.model.ResponsibleOfficer.getTelephoneNumber

```

94| this.postalCode = respOfficer.getPostalCode();
95| this.postTown = respOfficer.getPostTown();
96| this.email = respOfficer.getEmail();
>97| this.telephoneNumber = respOfficer.getTelephoneNumber();

```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/ResponsibleOfficer.java 97

**Fact Name:** no.valg.eva.admin.configuration.domain.model.ResponsibleOfficer.getTelephoneNumber

```

94| this.postalCode = respOfficer.getPostalCode();
95| this.postTown = respOfficer.getPostTown();
96| this.email = respOfficer.getEmail();
>97| this.telephoneNumber = respOfficer.getTelephoneNumber();

```

**Issue 728CACA0FB821131489293E628BB61BA:****Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/ResponsibleOfficer.java 90

**Fact Name:** no.valg.eva.admin.configuration.domain.model.ResponsibleOfficer.getLastName

```

87|this.nameLine = respOfficer.getNameLine();
88|this.firstName = respOfficer.getFirstName();
89|this.middleName = respOfficer.getMiddleName();
>90|this.lastName = respOfficer.getLastName();
91|this.addressLine1 = respOfficer.getAddressLine1();
92|this.addressLine2 = respOfficer.getAddressLine2();
93|this.addressLine3 = respOfficer.getAddressLine3();

```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/ResponsibleOfficer.java 90

**Fact Name:** no.valg.eva.admin.configuration.domain.model.ResponsibleOfficer.getLastName

```

87|this.nameLine = respOfficer.getNameLine();
88|this.firstName = respOfficer.getFirstName();
89|this.middleName = respOfficer.getMiddleName();
>90|this.lastName = respOfficer.getLastName();
91|this.addressLine1 = respOfficer.getAddressLine1();
92|this.addressLine2 = respOfficer.getAddressLine2();
93|this.addressLine3 = respOfficer.getAddressLine3();

```

**Issue 8199E0BE434DB0D5C449045D8722A98B:****Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/ResponsibleOfficer.java 89

**Fact Name:** no.valg.eva.admin.configuration.domain.model.ResponsibleOfficer.getMiddleName

```

86|this.displayOrder = respOfficer.getDisplayOrder();

```

```

87|this.nameLine = respOfficer.getNameLine();
88|this.firstName = respOfficer.getFirstName();
>89|this.middleName = respOfficer.getMiddleName();
90|this.lastName = respOfficer.getLastName();
91|this.addressLine1 = respOfficer.getAddressLine1();
92|this.addressLine2 = respOfficer.getAddressLine2();

```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/ResponsibleOfficer.java 89

**Fact Name:** no.valg.eva.admin.configuration.domain.model.ResponsibleOfficer.getMiddleName

```

86|this.displayOrder = respOfficer.getDisplayOrder();
87|this.nameLine = respOfficer.getNameLine();
88|this.firstName = respOfficer.getFirstName();
>89|this.middleName = respOfficer.getMiddleName();
90|this.lastName = respOfficer.getLastName();
91|this.addressLine1 = respOfficer.getAddressLine1();
92|this.addressLine2 = respOfficer.getAddressLine2();

```

**Issue 863CECCoB46FC1F8C50EF029426A9E1A:****Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/ResponsibleOfficer.java 92

**Fact Name:** no.valg.eva.admin.configuration.domain.model.ResponsibleOfficer.getAddressLine2

```

89|this.middleName = respOfficer.getMiddleName();
90|this.lastName = respOfficer.getLastName();
91|this.addressLine1 = respOfficer.getAddressLine1();
>92|this.addressLine2 = respOfficer.getAddressLine2();
93|this.addressLine3 = respOfficer.getAddressLine3();
94|this.postalCode = respOfficer.getPostalCode();
95|this.postTown = respOfficer.getPostTown();

```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/ResponsibleOfficer.java 92

**Fact Name:** no.valg.eva.admin.configuration.domain.model.ResponsibleOfficer.getAddressLine2

```

89|this.middleName = respOfficer.getMiddleName();
90|this.lastName = respOfficer.getLastName();
91|this.addressLine1 = respOfficer.getAddressLine1();
>92|this.addressLine2 = respOfficer.getAddressLine2();
93|this.addressLine3 = respOfficer.getAddressLine3();
94|this.postalCode = respOfficer.getPostalCode();
95|this.postTown = respOfficer.getPostTown();

```

**Issue 945F936873C4E4661BD7A73CC0868CAD:****Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/ResponsibleOfficer.java 88

**Fact Name:** no.valg.eva.admin.configuration.domain.model.ResponsibleOfficer.getFirstName

```

85|this.responsibility = respOfficer.getResponsibility();
86|this.displayOrder = respOfficer.getDisplayOrder();
87|this.nameLine = respOfficer.getNameLine();
>88|this.firstName = respOfficer.getFirstName();
89|this.middleName = respOfficer.getMiddleName();
90|this.lastName = respOfficer.getLastName();
91|this.addressLine1 = respOfficer.getAddressLine1();

```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/ResponsibleOfficer.java 88

**Fact Name:** no.valg.eva.admin.configuration.domain.model.ResponsibleOfficer.getFirstName

```

85|this.responsibility = respOfficer.getResponsibility();
86|this.displayOrder = respOfficer.getDisplayOrder();
87|this.nameLine = respOfficer.getNameLine();
>88|this.firstName = respOfficer.getFirstName();
89|this.middleName = respOfficer.getMiddleName();
90|this.lastName = respOfficer.getLastName();
91|this.addressLine1 = respOfficer.getAddressLine1();

```

**Issue A35FoDCFDBDDCF626266F4E74Do4EB8o:****Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/ResponsibleOfficer.java 93

**Fact Name:** no.valg.eva.admin.configuration.domain.model.ResponsibleOfficer.getAddressLine3

```

90|this.lastName = respOfficer.getLastName();
91|this.addressLine1 = respOfficer.getAddressLine1();
92|this.addressLine2 = respOfficer.getAddressLine2();
>93|this.addressLine3 = respOfficer.getAddressLine3();
94|this.postalCode = respOfficer.getPostalCode();
95|this.postTown = respOfficer.getPostTown();
96|this.email = respOfficer.getEmail();

```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/ResponsibleOfficer.java 93

**Fact Name:** no.valg.eva.admin.configuration.domain.model.ResponsibleOfficer.getAddressLine3

```

90|this.lastName = respOfficer.getLastName();
91|this.addressLine1 = respOfficer.getAddressLine1();
92|this.addressLine2 = respOfficer.getAddressLine2();
>93|this.addressLine3 = respOfficer.getAddressLine3();
94|this.postalCode = respOfficer.getPostalCode();
95|this.postTown = respOfficer.getPostTown();
96|this.email = respOfficer.getEmail();

```

**Issue C3EAFC27FCB3763D44B4DoDE106237C:****Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/ResponsibleOfficer.java 86

**Fact Name:** no.valg.eva.admin.configuration.domain.model.ResponsibleOfficer.getDisplayOrder

```

83|super();
84|this.reportingUnit = respOfficer.getReportingUnit();
85|this.responsibility = respOfficer.getResponsibility();
>86|this.displayOrder = respOfficer.getDisplayOrder();
87|this.nameLine = respOfficer.getNameLine();
88|this.firstName = respOfficer.getFirstName();
89|this.middleName = respOfficer.getMiddleName();

```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/ResponsibleOfficer.java 86

**Fact Name:** no.valg.eva.admin.configuration.domain.model.ResponsibleOfficer.getDisplayOrder

```

83|super();
84|this.reportingUnit = respOfficer.getReportingUnit();
85|this.responsibility = respOfficer.getResponsibility();
>86|this.displayOrder = respOfficer.getDisplayOrder();
87|this.nameLine = respOfficer.getNameLine();
88|this.firstName = respOfficer.getFirstName();
89|this.middleName = respOfficer.getMiddleName();

```

**Issue C53C5522A66A0D4B5F44FD8EF3A1831C:****Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/ResponsibleOfficer.java 85

**Fact Name:** no.valg.eva.admin.configuration.domain.model.ResponsibleOfficer.getResponsibility

```

83|super();
84|this.reportingUnit = respOfficer.getReportingUnit();
>85|this.responsibility = respOfficer.getResponsibility();
86|this.displayOrder = respOfficer.getDisplayOrder();
87|this.nameLine = respOfficer.getNameLine();
88|this.firstName = respOfficer.getFirstName();

```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/ResponsibleOfficer.java 85

**Fact Name:** no.valg.eva.admin.configuration.domain.model.ResponsibleOfficer.getResponsibility

```

83|super();
84|this.reportingUnit = respOfficer.getReportingUnit();
>85|this.responsibility = respOfficer.getResponsibility();
86|this.displayOrder = respOfficer.getDisplayOrder();
87|this.nameLine = respOfficer.getNameLine();
88|this.firstName = respOfficer.getFirstName();

```

**Issue D7D14B026379A61B8FCB2F741A01F5A9:****Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/ResponsibleOfficer.java 94

**Fact Name:** no.valg.eva.admin.configuration.domain.model.ResponsibleOfficer.getPostalCode

```

91|this.addressLine1 = respOfficer.getAddressLine1();
92|this.addressLine2 = respOfficer.getAddressLine2();
93|this.addressLine3 = respOfficer.getAddressLine3();
>94|this.postalCode = respOfficer.getPostalCode();
95|this.postTown = respOfficer.getPostTown();
96|this.email = respOfficer.getEmail();
97|this.telephoneNumber = respOfficer.getTelephoneNumber();

```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/ResponsibleOfficer.java 94

**Fact Name:** no.valg.eva.admin.configuration.domain.model.ResponsibleOfficer.getPostalCode

```

91|this.addressLine1 = respOfficer.getAddressLine1();
92|this.addressLine2 = respOfficer.getAddressLine2();
93|this.addressLine3 = respOfficer.getAddressLine3();
>94|this.postalCode = respOfficer.getPostalCode();
95|this.postTown = respOfficer.getPostTown();
96|this.email = respOfficer.getEmail();
97|this.telephoneNumber = respOfficer.getTelephoneNumber();

```

**Issue DAF64E88023F341EED42A9FD32C1008B:****Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/ResponsibleOfficer.java 96

**Fact Name:** no.valg.eva.admin.configuration.domain.model.ResponsibleOfficer.getEmail

```
93| this.addressLine3 = respOfficer.getAddressLine3();
94| this.postalCode = respOfficer.getPostalCode();
95| this.postTown = respOfficer.getPostTown();
>96| this.email = respOfficer.getEmail();
97| this.telephoneNumber = respOfficer.getTelephoneNumber();
```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/ResponsibleOfficer.java 96

**Fact Name:** no.valg.eva.admin.configuration.domain.model.ResponsibleOfficer.getEmail

```
93| this.addressLine3 = respOfficer.getAddressLine3();
94| this.postalCode = respOfficer.getPostalCode();
95| this.postTown = respOfficer.getPostTown();
>96| this.email = respOfficer.getEmail();
97| this.telephoneNumber = respOfficer.getTelephoneNumber();
```

### Issue ED48C19006DECoBBD2DFB05097025960:

**Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/ResponsibleOfficer.java 84

**Fact Name:** no.valg.eva.admin.configuration.domain.model.ResponsibleOfficer.getReportingUnit

```
81| public ResponsibleOfficer(ResponsibleOfficer respOfficer) {
82|
83|     super();
>84|     this.reportingUnit = respOfficer.getReportingUnit();
85|     this.responsibility = respOfficer.getResponsibility();
86|     this.displayOrder = respOfficer.getDisplayOrder();
87|     this.nameLine = respOfficer.getNameLine();
```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/ResponsibleOfficer.java 84

**Fact Name:** no.valg.eva.admin.configuration.domain.model.ResponsibleOfficer.getReportingUnit

```
81| public ResponsibleOfficer(ResponsibleOfficer respOfficer) {
82|
83|     super();
>84|     this.reportingUnit = respOfficer.getReportingUnit();
85|     this.responsibility = respOfficer.getResponsibility();
86|     this.displayOrder = respOfficer.getDisplayOrder(); 87|     this.nameLine = respOfficer.getNameLine();
```

## 6.21 SkdVoterFileParser.java

### Issue 48267336DAB3DD8D53F709BE4416B4C6:

**Source :**

**File** admin-voting/src/main/java/no/valg/eva/admin/voting/domain/electoralroll/SkdVoterFileParser.java 29

**Fact Name:** no.valg.eva.admin.voting.domain.electoralroll.SkdVoterFileParser.header

```
26|     reader =
ImportElectoralRollUtil.getBufferedReaderForFile(filename); 27|     header
= reader.readLine(); 28|
>29|     if (header == null || header.length() != RECORD_SIZE || !header().equals("HD")) {
30|         throw new IOException("Invalid or non-existent header in import file for electoral roll");
```

**Sink :**

**File** admin-voting/src/main/java/no/valg/eva/admin/voting/domain/electoralroll/SkdVoterFileParser.java 29

**Fact Name:** no.valg.eva.admin.voting.domain.electoralroll.SkdVoterFileParser.header

```
26|     reader = ImportElectoralRollUtil.getBufferedReaderForFile(filename);
27|     header = reader.readLine();
28|
>29|     if (header == null || header.length() != RECORD_SIZE || !header().equals("HD")) {
30|         throw new IOException("Invalid or non-existent header in import file for electoral roll");
```

## [Medium] J2EE Misconfiguration / Excessive Session Timeout

*Environment*

**Analysis:** Suspicious

### Abstract

An overly long session timeout gives attackers more time to potentially compromise user accounts.

### Explanation

The longer a session stays open, the larger the window of opportunity an attacker has to compromise user accounts. While a session remains active, an attacker may be able to brute-force a user's password,

Arkitektur- og kildekodegjennomgang EVA Admin – KUNDEKONFIDENSIELL mnemonic as crack a user's wireless encryption key, or commandeer a session from an open browser. Longer session timeouts can also prevent memory from being released and eventually result in a denial of service if a sufficiently large number of sessions are created.

**Example 1:** If the session timeout is zero or less than zero, the session never expires. The following example shows a session timeout set to -1, which will cause the session to remain active indefinitely.

```
<session-config>
  <session-timeout>-1</session-timeout> </session-
config>
```

The `<session-timeout>` tag defines the default session timeout interval for all sessions in the web application. If the `<session-timeout>` tag is missing, it is left to the container to set the default timeout. This category was derived from the Cigital Java Rulepack. <http://www.cigital.com/>

## Recommendation

Set a session timeout that is 30 minutes or less, which both allows users to interact with the application over a period of time and provides a reasonable bound for the window of attack.

**Example 2:** The following example sets the session timeout to 20 minutes.

```
<session-config>
  <session-timeout>20</session-timeout> </session-
config>
```

## 6.22 web.xml

**Issue** C2F39B963AB1AFACD8D57325EEAF7480:

**Sink :**

**File** admin-frontend/src/main/webapp/WEB-INF/web.xml 188

```
185|</servlet-mapping>
186|
187|<!-- Misc. configuration -->
>188|<session-config>
189|   <session-timeout>120</session-timeout>
190|   <cookie-config>
191|     <path>/</path>
```

## [Medium] J2EE Misconfiguration / Incomplete Error Handling

*Environment*

**Analysis:** Bad Practice

### Abstract

A web application must define a default error page for 500 errors to prevent attackers from mining information from the application container's built-in error response.

### Explanation

When an attacker explores a web site looking for vulnerabilities, the amount of information that the site provides is crucial to the eventual success or failure of any attempted attacks. If the application shows the attacker a stack trace, it relinquishes information that makes the attacker's job significantly easier. For example, a stack trace might show the attacker a malformed SQL query string, the type of database being used, and the version of the application container. This information enables the attacker to target known vulnerabilities in these components.

The application configuration should specify a default error page in order to guarantee that the application will never leak error messages to an attacker. Handling standard HTTP error codes is useful and user-friendly in addition to being a good security practice, and a good configuration will also

Arkitektur- og kildekodegjennomgang EVA Admin – KUNDEKONFIDENSIELL mnemonic as define a last-chance error handler that catches any exception that could possibly be thrown by the application.

## Recommendation

A web application must be configured with a default error page. Your `web.xml` should include at least the following entries:

```
<error-page>
  <exception-type>java.lang.Throwable</exception-type>
</error-page>
<error-page>
  <error-code>404</error-code> <location>/error.jsp</location>
</error-page>
<error-page>
  <error-code>500</error-code> <location>/error.jsp</location>
</error-page>
```

## 6.23 web.xml

Issue F3AD381592A7A025EB4A252010C526F6:

Sink :

File admin-frontend/src/main/webapp/WEB-INF/web.xml 211

Action /web-app/error-page/error-code

```
208|</listener>
209|
210|<error-page>
>211|   <error-code>404</error-code>
212|   <location>/404</location>
213|</error-page>
```

## [Medium] J2EE Misconfiguration / Incomplete Error Handling

*Environment*

**Analysis:** Bad Practice

### Abstract

A web application must define a default error page for `java.lang.Throwable` to handle any uncaught exceptions in order to prevent attackers from mining information from the application container's built-in error response.

### Explanation

When an attacker explores a web site looking for vulnerabilities, the amount of information that the site provides is crucial to the eventual success or failure of any attempted attacks. If the application shows the attacker a stack trace, it relinquishes information that makes the attacker's job significantly easier. For example, a stack trace might show the attacker a malformed SQL query string, the type of database being used, and the version of the application container. This information enables the attacker to target known vulnerabilities in these components.

The application configuration should specify a default error page in order to guarantee that the application will never leak error messages to an attacker. Handling standard HTTP error codes is useful and user-friendly in addition to being a good security practice, and a good configuration will also define a last-chance error handler that catches any exception that could possibly be thrown by the application.

### Recommendation

A web application must be configured with a default error page. Your `web.xml` should include at least the following entries:

```

<error-page>
  <exception-type>java.lang.Throwable</exception-type>
</error-page>
<error-page>
  <error-code>404</error-code> <location>/error.jsp</location>
</error-page>
<error-page>
  <error-code>500</error-code> <location>/error.jsp</location>
</error-page>

```

## 6.24 web.xml

### Issue 20A5DD06F2BD65FoCD5FBBF38CE4BE7C:

**Sink :**

**File** admin-frontend/src/main/webapp/WEB-INF/web.xml 210

**Action** /web-app/error-page/exception-type

```

207|   <listener-class>no.evote.lifecycle.LifecycleListener</listener-class> 208|</listener>
209|
>210|<error-page>
211|   <error-code>404</error-code>
212|   <location>/404</location>
213|</error-page>

```

## [Medium] Missing Check against Null

*API Abuse*

**Analysis:** Reliability Issue

### Abstract

The program can dereference a null pointer because it does not check the return value of a function that might return null.

### Explanation

Just about every serious attack on a software system begins with the violation of a programmer's assumptions. After the attack, the programmer's assumptions seem flimsy and poorly founded, but before an attack many programmers would defend their assumptions well past the end of their lunch break.

Two dubious assumptions that are easy to spot in code are "this function call can never fail" and "it doesn't matter if this function call fails". When a programmer ignores the return value from a function, they implicitly state that they are operating under one of these assumptions.

**Example 1:** The following code does not check to see if the string returned by `getParameter()` is `null` before calling the member function `compareTo()`, potentially causing a null dereference.

```

String itemName =
request.getParameter(ITEM_NAME); if
(itemName.compareTo(IMPORTANT_ITEM)) { ... }
...

```

**Example 2:** The following code shows a system property that is set to `null` and later dereferenced by a programmer who mistakenly assumes it will always be defined.

```

System.clearProperty("os.name");
...
String os = System.getProperty("os.name");
if (os.equalsIgnoreCase("Windows 95") )
System.out.println("Not supported");

```

The traditional defense of this coding error is:

Arkitektur- og kildekodegjennomgang EVA Admin – KUNDEKONFIDENSIELL mnemonic as  
"I know the requested value will always exist because.... If it does not exist, the program cannot perform the desired behavior so it doesn't matter whether I handle the error or simply allow the program to die dereferencing a null value."

But attackers are skilled at finding unexpected paths through programs, particularly when exceptions are involved.

## Recommendation

If a function can return an error code or any other evidence of its success or failure, always check for the error condition, even if there is no obvious way for it to occur. In addition to preventing security errors, many initially mysterious bugs have eventually led back to a failed method call with an unchecked return value.

Create an easy to use and standard way for dealing with failure in your application. If error handling is straightforward, programmers will be less inclined to omit it. One approach to standardized error handling is to write wrappers around commonly-used functions that check and handle error conditions without additional programmer intervention. When wrappers are implemented and adopted, the use of non-wrapped equivalents can be prohibited and enforced by using custom rules.

**Example 3:** The following code implements a wrapper around `getParameter()` that checks the return value of `getParameter()` against `null` and uses a default value if the requested parameter is not defined.

```
String safeGetParameter (HttpRequest request, String name)
{
    String value = request.getParameter(name);
    if (value == null) {
        return getDefaultVAlue(name)
    }
    return
value;
}
```

## 6.25 AccessServiceBean.java

**Issue AB1c7B4E8A7FD8BCA8E0E71358CE2E1E:**

**Source :**

**File** admin-rbac/src/main/java/no/valg/eva/admin/rbac/service/AccessServiceBean.java 40

**Action** Assign electionEvent = findByPk(...): ElectionEventRepository.findByPk may return NULL

**Trace id** 3

```
37|String asString = StringUtils.join(accessPaths.toArray(new String[accessPaths.size()]), ",");
38|
39|// Don't try to sign data for the admin event, since it doesn't have any key.
>40|ElectionEvent electionEvent = electionEventRepository.findByPk(userData.getElectionEventPk());
41|byte[] signature = null;
42|if (!electionEvent.getId().equals(ROOT_ELECTION_EVENT_ID)) {
43|    try {
```

**Sink :**

**File** admin-rbac/src/main/java/no/valg/eva/admin/rbac/service/AccessServiceBean.java 40

**Action** Assign electionEvent = findByPk(...): ElectionEventRepository.findByPk may return NULL **Trace id** 3

```
37|String asString = StringUtils.join(accessPaths.toArray(new String[accessPaths.size()]), ",");
38|
39|// Don't try to sign data for the admin event, since it doesn't have any key.
>40|ElectionEvent electionEvent = electionEventRepository.findByPk(userData.getElectionEventPk());
41|byte[] signature = null;
42|if (!electionEvent.getId().equals(ROOT_ELECTION_EVENT_ID)) {
43|    try {
```

## 6.26 AdminOperatorApplicationService.java

**Issue B12E23F3C1D1201B55ACD6429726D130:**

**Source :**

**File** admin-rbac/src/main/java/no/valg/eva/admin/rbac/application/AdminOperatorApplicationService.java 328 **Action** Assign operator = findByPk(...): OperatorRepository.findByPk may return NULL

**Trace id** 51

```
325|@Override
326|@SecurityNone
327|public ContactInfo contactInfoForOperator(UserData userData) {
>328|    no.valg.eva.admin.rbac.domain.model.Operator operator =
operatorRepository.findByPk(userData.getOperator().getPk());
329|    return new ContactInfo(operator.getTelephoneNumber(), operator.getEmail());
```

**Sink :**

**File** admin-rbac/src/main/java/no/valg/eva/admin/rbac/application/AdminOperatorApplicationService.java

328 **Action** Assign operator = findByPk(...): OperatorRepository.findByPk may return NULL **Trace id** 51



```

325|@Override
326|@SecurityNone
327|public ContactInfo contactInfoForOperator(UserData userData) {
>328|    no.valg.eva.admin.rbac.domain.model.Operator operator =
operatorRepository.findByPk(userData.getOperator().getPk());
329|    return new ContactInfo(operator.getTelephoneNumber(), operator.getEmail());

```

**Issue B6A93DD46E64A64CF279AFC6C6CA8F71:****Source :**

**File** admin-rbac/src/main/java/no/valg/eva/admin/rbac/application/AdminOperatorApplicationService.java 336 **Action** Assign findByPk(...): OperatorRepository.findByPk may return NULL

**Trace id** 51

```

333|@SecurityNone
334|@AuditLog(eventClass = ContactInfoChangedAuditEvent.class, eventType = AuditEventTypes.ContactInfoChanged)
335|public void updateContactInfoForOperator(UserData userData, ContactInfo contactInfo) {
>336|    operatorRepository.findByPk(userData.getOperator().getPk()).updateContactInfo(contactInfo);
337|}
338|
339|@Override

```

**Sink :**

**File** admin-rbac/src/main/java/no/valg/eva/admin/rbac/application/AdminOperatorApplicationService.java 336 **Action** Assign findByPk(...): OperatorRepository.findByPk may return NULL

**Trace id** 51

```

333|@SecurityNone
334|@AuditLog(eventClass = ContactInfoChangedAuditEvent.class, eventType = AuditEventTypes.ContactInfoChanged)
335|public void updateContactInfoForOperator(UserData userData, ContactInfo contactInfo) {
>336|    operatorRepository.findByPk(userData.getOperator().getPk()).updateContactInfo(contactInfo);
337|}
338|
339|@Override

```

## 6.27 AffiliationServiceBean.java

**Issue 3D8577B6942CE0D580779B2AE85F6FC1:****Source :**

**File** admin-backend/src/main/java/no/evote/service/configuration/AffiliationServiceBean.java 61

**Action** Assign dbAffiliation = findAffiliationByPk(...): AffiliationRepository.findAffiliationByPk may return NULL **Trace id** 21

```

58| * Saves columns(residence/profession) shown on ballot.
59| */
60|public Affiliation saveColumns(UserData userData, Affiliation affiliation) {
>61|    Affiliation dbAffiliation = affiliationRepository.findAffiliationByPk(affiliation.getPk());
62|    dbAffiliation.setShowCandidateProfession(affiliation.isShowCandidateProfession());
63|    dbAffiliation.setShowCandidateResidence(affiliation.isShowCandidateResidence()); 64|    return
affiliationRepository.updateAffiliation(userData, dbAffiliation);

```

**Sink :**

**File** admin-backend/src/main/java/no/evote/service/configuration/AffiliationServiceBean.java 61

**Action** Assign dbAffiliation = findAffiliationByPk(...): AffiliationRepository.findAffiliationByPk may return NULL **Trace id** 21

```

58| * Saves columns(residence/profession) shown on ballot.
59| */
60|public Affiliation saveColumns(UserData userData, Affiliation affiliation) {
>61|    Affiliation dbAffiliation = affiliationRepository.findAffiliationByPk(affiliation.getPk());
62|    dbAffiliation.setShowCandidateProfession(affiliation.isShowCandidateProfession());
63|    dbAffiliation.setShowCandidateResidence(affiliation.isShowCandidateResidence()); 64|    return
affiliationRepository.updateAffiliation(userData, dbAffiliation);

```

## 6.28 AffiliationServiceEjb.java

**Issue C75F6CF5F531352077A92F5015859F19:****Source :**

**File** admin-backend/src/main/java/no/evote/service/configuration/AffiliationServiceEjb.java 68

**Action** Assign currentContest = findByPk(...): ContestRepository.findByPk may return NULL

**Trace id** 0

```

65|@Security(accesses = Listeforslag_Opprett_Nytt_Part_i, type = WRITE)
66|@AuditLog(eventClass = AffiliationAuditEvent.class, eventType = AuditEventTypes.Create, objectSource =
AuditedObjectSource.ReturnValue)
67|public Affiliation createNewPartyAndAffiliation(UserData userData, Contest detachedContest, Parti newParty,
Locale locale) {
>68|    Contest currentContest = contestRepository.findByPk(detachedContest.getPk());
69|    newParty.leggTilOmraadeHvisLokalT(currentContest.getFirstContestArea().getAreaPath());
70|    return affiliationService.createNewPartyAndAffiliation(userData, currentContest,
partyMapper.toParty(newParty, userData.electionEvent(), locale);

```

**Sink :**

**File** admin-backend/src/main/java/no/evote/service/configuration/AffiliationServiceEjb.java 68

**Action** Assign currentContest = findByPk(...): ContestRepository.findByPk may return NULL

**Trace id** 0

```

65|@Security(accesses = Listeforslag_Opprett_Nytt_Partii, type = WRITE)
66|@AuditLog(eventClass = AffiliationAuditEvent.class, eventType = AuditEventTypes.Create, objectSource =
AuditedObjectSource.ReturnValue)
67|public Affiliation createNewPartyAndAffiliation(UserData userData, Contest detachedContest, Parti newParty,
Locale locale) {
>68|    Contest currentContest = contestRepository.findByPk(detachedContest.getPk());
69|    newParty.leggTilOmraadeHvisLokalt(currentContest.getFirstContestArea().getAreaPath());
70|    return affiliationService.createNewPartyAndAffiliation(userData, currentContest,
partyMapper.toParty(newParty, userData.electionEvent(), locale);

```

## 6.29 AntiSamyUtil.java

### Issue 96B86734BE08D59B2A8D5F014779A2BE:

#### Source :

File admin-common/src/main/java/no/evote/util/AntiSamyUtil.java 28

Action BranchTaken Branch taken: (policyFile == null)

```

25|private AntiSamyUtil() {
26|    try {
27|        String policyFile =
EvoteProperties.getProperty(EvoteProperties.NO_EVOTE_UTIL_ANTI_SAMY_FILTER_POLICY_FILE, true);
>28|        if (policyFile == null) {
29|            policy =
Policy.getInstance(this.getClass().getClassLoader().getResourceAsStream(POLICY_FILE_LOCATION));
30|            LOG.info("Using built-in policy file");
31|        } else {

```

#### Sink :

File admin-common/src/main/java/no/evote/util/AntiSamyUtil.java 29

Action Assign getClassLoader() : Class.getClassLoader may return NULL

Rule id 4280F38B-9FDB-454E-B495-89CF45CD51B7

```

26|try {
27|    String policyFile = EvoteProperties.getProperty(EvoteProperties.NO_EVOTE_UTIL_ANTI_SAMY_FILTER_POLICY_FILE,
true);
28|    if (policyFile == null) {
>29|        policy = Policy.getInstance(this.getClass().getClassLoader().getResourceAsStream(POLICY_FILE_LOCATION));
30|        LOG.info("Using built-in policy file");
31|    } else {
32|        policy = Policy.getInstance(new FileInputStream(policyFile));

```

## 6.30 AreaImportChangesApplicationService.java

### Issue 62A2EEE27367E4EBDFA6FE0B0187E937:

#### Source :

File admin-configuration/src/main/java/no/valg/eva/admin/configuration/application/AreaImportChangesApplicationService.java 80

Action Assign electionEvent = findByPk(...) : ElectionEventRepository.findByPk may return NULL Trace id 3

```

77|StringBuilder logFileDelete = new StringBuilder();
78|StringBuilder logFileDeleteFailed = new StringBuilder();
79|
>80|ElectionEvent electionEvent = electionEventRepository.findByPk(userData.getElectionEventPk());
81|String countryAreaPath = retrieveAreaPath(electionEvent, null);
82|
83|Collection<MvArea> allExistingPollingDistrictMvAreas = mvAreaRepository.findByPathAndLevel(countryAreaPath,
AreaLevelEnum.POLLING_DISTRICT.getLevel());

```

#### Sink :

File admin-configuration/src/main/java/no/valg/eva/admin/configuration/application/AreaImportChangesApplicationService.java

80 Action Assign electionEvent = findByPk(...) : ElectionEventRepository.findByPk may return NULL Trace id 3

```

77|StringBuilder logFileDelete = new StringBuilder();
78|StringBuilder logFileDeleteFailed = new StringBuilder();
79|
>80|ElectionEvent electionEvent = electionEventRepository.findByPk(userData.getElectionEventPk());
81|String countryAreaPath = retrieveAreaPath(electionEvent, null);
82|
83|Collection<MvArea> allExistingPollingDistrictMvAreas = mvAreaRepository.findByPathAndLevel(countryAreaPath,
AreaLevelEnum.POLLING_DISTRICT.getLevel());

```

## 6.31 AreaImportServiceBean.java

### Issue B4E741F4B3235044A7F6B4617C62E59B:

#### Source :

File admin-backend/src/main/java/no/evote/service/configuration/AreaImportServiceBean.java 70

Action Assign electionEvent = findByPk(...) : ElectionEventRepository.findByPk may return NULL

Trace id 3

```

67|private PollingPlaceRepository pollingPlaceRepository;
68|
69|public void importAreaHierarchy(final UserData userData, final byte[] data) throws IOException {
>70|     ElectionEvent electionEvent = electionEventRepository.findByPk(userData.getElectionEventPk()); 71|
72|     // Check if there actually is data to import
73|     if (data == null || data.length == 0) {

```

**Sink :****File** admin-backend/src/main/java/no/evote/service/configuration/AreaImportServiceBean.java 70**Action** Assign electionEvent = findByPk(...): ElectionEventRepository.findByPk may return NULL**Trace id** 3

```

67|private PollingPlaceRepository pollingPlaceRepository;
68|
69|public void importAreaHierarchy(final UserData userData, final byte[] data) throws IOException {
>70|     ElectionEvent electionEvent = electionEventRepository.findByPk(userData.getElectionEventPk()); 71|
72|     // Check if there actually is data to import
73|     if (data == null || data.length == 0) {

```

## 6.32 BoroughServiceBean.java

**Issue** 5964AB321522D04B5768941280444E5B:**Source :****File** admin-backend/src/main/java/no/evote/service/configuration/BoroughServiceBean.java 36**Action** BranchTaken Branch taken: borough.isMunicipality1()

```

34|public Borough update(UserData userData, Borough borough) {
35|     // The municipality can only have one borough for the whole municipality
>36|     if (borough.isMunicipality1() &&
isBoroughForWholeMunicipalityAlreadyExisting(borough.getMunicipality().getPk())
37|         && !boroughRepository.findBoroughByPk(borough.getPk()).isMunicipality1()) {
38|         throw new
EvoteException("@common.message.evote_application_exception.DUPLICATE_BOROUGH_WHOLE_MUNICIPALITY");

```

**Sink :****File** admin-backend/src/main/java/no/evote/service/configuration/BoroughServiceBean.java 37**Action** Assign findBoroughByPk(...): BoroughRepository.findBoroughByPk may return NULL **Trace id** 31

```

34|public Borough update(UserData userData, Borough borough) {
35|     // The municipality can only have one borough for the whole municipality
36|     if (borough.isMunicipality1() &&
isBoroughForWholeMunicipalityAlreadyExisting(borough.getMunicipality().getPk())
>37|         && !boroughRepository.findBoroughByPk(borough.getPk()).isMunicipality1()) {
38|         throw new
EvoteException("@common.message.evote_application_exception.DUPLICATE_BOROUGH_WHOLE_MUNICIPALITY");

```

## 6.33 CentralConfigurationApplicationService.java

**Issue** A50563A507CC221A11582E771144BFAE:**Source :****File** admin-backend/src/main/java/no/valg/eva/admin/configuration/application/CentralConfigurationApplicationService.java 77 **Action** Assign electionEvent = findByPk(...): ElectionEventDomainService.findByPk may return NULL**Trace id** 45

```

74|@Override
75|@Security(accesses = Konfigurasjon_Oversikt, type = READ)
76|public CentralConfigurationSummary getCentralConfigurationSummary(UserData userData) {
>77|     ElectionEvent electionEvent = electionEventService.findByPk(userData.getElectionEventPk());
78|
79|     CentralConfigurationSummary summary = new CentralConfigurationSummary();
80|
summary.setHasAnyMunicipalitiesWithNonElectronicMarkOffs(municipalityService.getHasAnyMunicipalitiesWithNonElectronicMarkOffs(userData));

```

**Sink :****File** admin-backend/src/main/java/no/valg/eva/admin/configuration/application/CentralConfigurationApplicationService.java**Action** Assign electionEvent = findByPk(...): ElectionEventDomainService.findByPk may return NULL **Trace id** 45

```

74|@Override
75|@Security(accesses = Konfigurasjon_Oversikt, type = READ)
76|public CentralConfigurationSummary getCentralConfigurationSummary(UserData userData) {
>77|     ElectionEvent electionEvent = electionEventService.findByPk(userData.getElectionEventPk());
78|
79|     CentralConfigurationSummary summary = new CentralConfigurationSummary();
80|
summary.setHasAnyMunicipalitiesWithNonElectronicMarkOffs(municipalityService.getHasAnyMunicipalitiesWithNonElectronicMarkOffs(userData));

```

## 6.34 ContestAreaServiceEjb.java

**Issue** 9B0DBF56053DF6987FFCF7473D13B174:**Source :****File** admin-backend/src/main/java/no/evote/service/configuration/ContestAreaServiceEjb.java 78**Action** Assign area = findByPk(...): ContestAreaRepository.findByPk may return NULL **Trace****id** 44

```

75|@Override
76|@Security(accesses = Konfigurasjon_Valg_Valgdistrikt, type = WRITE)

```

```

77|public void delete(UserData userData, @SecureEntity(electionLevel = ElectionLevelEnum.CONTEST, entity =
ContestArea.class) Long pk) {
>78|    ContestArea area = contestAreaRepository.findByPk(pk);
79|    // DEV-NOTE: This is a data integrity check that when a parent is removed, no child can be present.
80|    // This should be placed in domain model.
81|    if (area.isParentArea()) {

```

**Sink :****File** admin-backend/src/main/java/no/evote/service/configuration/ContestAreaServiceEjb.java**78 Action** Assign area = findByPk(...): ContestAreaRepository.findByPk may return NULL **Trace****id** 44

```

75|@Override
76|@Security(accesses = Konfigurasjon_Valg_Valgdistrikt, type = WRITE)
77|public void delete(UserData userData, @SecureEntity(electionLevel = ElectionLevelEnum.CONTEST, entity =
ContestArea.class) Long pk) {
>78|    ContestArea area = contestAreaRepository.findByPk(pk);
79|    // DEV-NOTE: This is a data integrity check that when a parent is removed, no child can be present.
80|    // This should be placed in domain model.
81|    if (area.isParentArea()) {

```

## 6.35 ContestServiceBean.java

**Issue** 00435E6D3EFFF7C238B8A54BA442F416:**Source :****File** admin-backend/src/main/java/no/evote/service/configuration/ContestServiceBean.java 154**Action** Assign contest = findByPk(...): ContestRepository.findByPk may return NULL **Trace****id** 0

```

152|public void delete(UserData userData, Long pk, TransactionSynchronizationRegistry registry) {
153|    List<ContestArea> contestAreas = checkAccessToContestAreas(userData, contestRepository.findByPk(pk),
registry);
>154|    Contest contest = contestRepository.findByPk(pk);
155|    if (!isStatusCentralConfig(contest)) {
156|        throw new EvoteSecurityException("Cannot delete contest on electionEvent with other status than Central
config: " + contest.getPk());

```

**Sink :****File** admin-backend/src/main/java/no/evote/service/configuration/ContestServiceBean.java 154 **Action** Assign contest = findByPk(...): ContestRepository.findByPk may return NULL**Trace id** 0

```

152|public void delete(UserData userData, Long pk, TransactionSynchronizationRegistry registry) {
153|    List<ContestArea> contestAreas = checkAccessToContestAreas(userData, contestRepository.findByPk(pk),
registry);
>154|    Contest contest = contestRepository.findByPk(pk);
155|    if (!isStatusCentralConfig(contest)) {
156|        throw new EvoteSecurityException("Cannot delete contest on electionEvent with other status than Central
config: " + contest.getPk());

```

**Issue** 950A63F4987057A2F82374C2340B89EF:**Source :****File** admin-backend/src/main/java/no/evote/service/configuration/ContestServiceBean.java 153 **Action** Assign findByPk(...): ContestRepository.findByPk may return NULL**Trace id** 0

```

152|public void delete(UserData userData, Long pk, TransactionSynchronizationRegistry registry) {
>153|    List<ContestArea> contestAreas = checkAccessToContestAreas(userData, contestRepository.findByPk(pk),
registry);
154|    Contest contest = contestRepository.findByPk(pk);
155|    if (!isStatusCentralConfig(contest)) {
156|        throw new EvoteSecurityException("Cannot delete contest on electionEvent with other status than Central
config: " + contest.getPk());

```

**Sink :****File** admin-backend/src/main/java/no/evote/service/configuration/ContestServiceBean.java 153 **Action** Assign findByPk(...): ContestRepository.findByPk may return NULL**Trace id** 0

```

152|public void delete(UserData userData, Long pk, TransactionSynchronizationRegistry registry) {
>153|    List<ContestArea> contestAreas = checkAccessToContestAreas(userData, contestRepository.findByPk(pk),
registry);
154|    Contest contest = contestRepository.findByPk(pk);
155|    if (!isStatusCentralConfig(contest)) {
156|        throw new EvoteSecurityException("Cannot delete contest on electionEvent with other status than Central
config: " + contest.getPk());

```

## 6.36 CountingImportServiceBean.java

**Issue** 55AFF61A681998033B96E7D2E4A32A17:**Source :**

**File** admin-backend/src/main/java/no/evote/service/counting/CountingImportServiceBean.java  
**522 Action** Assign findByPk(...): ElectionEventRepository.findByPk may return NULL **Trace id** 3

```
519| */
520|private ElectionEvent verifyAndGetElectionEvent(UserData userData, String electionEventId) {
521|     ElectionEvent electionEvent = electionEventRepository.findById(electionEventId);
>522|     String userElectionEventId = electionEventRepository.findByPk(userData.getElectionEventPk()).getId(); 523|
if (!userElectionEventId.equals(electionEventId)) {
524|         throw new EvoteException(ErrorCode.ERROR_CODE_0306_WRONG_ELECTION_EVENT, null, electionEventId,
userElectionEventId);
```

**Sink :**

**File** admin-backend/src/main/java/no/evote/service/counting/CountingImportServiceBean.java  
**522 Action** Assign findByPk(...): ElectionEventRepository.findByPk may return NULL **Trace id** 3

```
519| */
520|private ElectionEvent verifyAndGetElectionEvent(UserData userData, String electionEventId) {
521|     ElectionEvent electionEvent = electionEventRepository.findById(electionEventId);
>522|     String userElectionEventId = electionEventRepository.findByPk(userData.getElectionEventPk()).getId(); 523|
if (!userElectionEventId.equals(electionEventId)) {
524|         throw new EvoteException(ErrorCode.ERROR_CODE_0306_WRONG_ELECTION_EVENT, null, electionEventId,
userElectionEventId);
```

**Issue ED3DF845997830CCF53DB66B81D9EBA2:****Source :**

**File** admin-backend/src/main/java/no/evote/service/counting/CountingImportServiceBean.java 961  
**Action** Assign electionEvent = findByPk(...): ElectionEventRepository.findByPk may return NULL  
**Trace id** 3

```
959|Element votesE1 = root.getChild("Votes", ns);
960|
>961|ElectionEvent electionEvent = electionEventRepository.findByPk(userData.getElectionEventPk());
962|Contest contest = voteCount.getContestReport().getContest();
963|List<Element> castVotes = votesE1.getChildren("CastVote", ns);
```

**Sink :**

**File** admin-backend/src/main/java/no/evote/service/counting/CountingImportServiceBean.java 961  
**Action** Assign electionEvent = findByPk(...): ElectionEventRepository.findByPk may return NULL **Trace id** 3

```
959|Element votesE1 = root.getChild("Votes", ns);
960|
>961|ElectionEvent electionEvent = electionEventRepository.findByPk(userData.getElectionEventPk());
962|Contest contest = voteCount.getContestReport().getContest();
963|List<Element> castVotes = votesE1.getChildren("CastVote", ns);
```

**6.37 CountingOverviewApplicationService.java****Issue EB46D2C9844A31B88070AC0605EF0BE1:****Source :**

**File** admin-counting/src/main/java/no/valg/eva/admin/counting/application/CountingOverviewApplicationService.java  
**77 Action** Assign mvElection = findByPk(...): MvElectionRepository.findByPk may return NULL **Trace id** 58

```
76|private List<ContestInfo> electionsForOpptellingsvalgstyret(UserData userData) {
>77|     MvElection mvElection = mvElectionRepository.findByPk(userData.getOperatorMvElection().getPk());
78|     return singletonList(new ContestInfo(mvElection.getElectionPath(), mvElection.getElectionName(),
mvElection.getContestName(), null));
```

**Sink :**

**File** admin-counting/src/main/java/no/valg/eva/admin/counting/application/CountingOverviewApplicationService.java  
**77 Action** Assign mvElection = findByPk(...): MvElectionRepository.findByPk may return NULL **Trace id** 58

```
76|private List<ContestInfo> electionsForOpptellingsvalgstyret(UserData userData) {
>77|     MvElection mvElection = mvElectionRepository.findByPk(userData.getOperatorMvElection().getPk());
78|     return singletonList(new ContestInfo(mvElection.getElectionPath(), mvElection.getElectionName(),
mvElection.getContestName(), null));
```

**6.38 CountyServiceBean.java****Issue 53B1B8457C31D1A588229EC9173491C8:****Source :**

**File** admin-backend/src/main/java/no/evote/service/configuration/CountyServiceBean.java  
**30 Action** Assign country = findByPk(...): CountryRepository.findByPk may return NULL  
**Trace id** 25

```
27|public County create(UserData userData, County county) {
28|     // Set correct countyStatus on new county. Either central or local, based on electionEvent.
29|     // DEV-NOTE Only county status is updated. Municipality is handled by database triggers. Should be moved
here.
>30|     Country country = countryRepository.findByPk(county.getCountry().getPk());
31|     CountyStatusEnum status = CountyStatusEnum.CENTRAL_CONFIGURATION;
```

```
32|    if (country.getElectionEvent().isLocalConfiguration()) { 33|        status =
CountyStatusEnum.LOCAL_CONFIGURATION;
```

**Sink :****File** admin-backend/src/main/java/no/evote/service/configuration/CountyServiceBean.java**30 Action** Assign country = findByPk(...): CountyRepository.findByPk may return NULL**Trace id** 25

```
27|public County create(UserData userData, County county) {
28|    // Set correct countyStatus on new county. Either central or local, based on electionEvent.
29|    // DEV-NOTE Only county status is updated. Municipality is handled by database triggers. Should be moved
here.
>30|    County county = countyRepository.findByPk(county.getCountry().getPk());
31|    CountyStatusEnum status = CountyStatusEnum.CENTRAL_CONFIGURATION;
32|    if (country.getElectionEvent().isLocalConfiguration()) { 33|        status =
CountyStatusEnum.LOCAL_CONFIGURATION;
```

## 6.39 CountyServiceEjb.java

### Issue 2485B0359EC171B0AECCA492DE11BCA7:

**Source :****File** admin-backend/src/main/java/no/evote/service/configuration/CountyServiceEjb.java 72**Action** Assign dbCounty = findByPk(...): CountyRepository.findByPk may return NULL**Trace id** 18

```
69|@Security(accesses = { Konfigurasjon_Grunnlagsdata_Redigere }, type = WRITE)
70|@AuditLog(eventClass = CountyAuditEventForLocalConfiguration.class, eventType = Update, objectSource =
ReturnValue)
71|public County updateScanningConfiguration(final UserData userData, @SecureEntity(areaLevel =
AreaLevelEnum.COUNTY) final County county) {
>72|    County dbCounty = countyRepository.findByPk(county.getPk()); // Dette hentes ut for å unngå at denne metoden
skal brukes for å endre County-objektet
73|    dbCounty.setScanningConfig(county.getScanningConfig()); 74|    return countyRepository.update(userData,
dbCounty);
```

**Sink :****File** admin-backend/src/main/java/no/evote/service/configuration/CountyServiceEjb.java 72**Action** Assign dbCounty = findByPk(...): CountyRepository.findByPk may return NULL**Trace id** 18

```
69|@Security(accesses = { Konfigurasjon_Grunnlagsdata_Redigere }, type = WRITE)
70|@AuditLog(eventClass = CountyAuditEventForLocalConfiguration.class, eventType = Update, objectSource =
ReturnValue)
71|public County updateScanningConfiguration(final UserData userData, @SecureEntity(areaLevel =
AreaLevelEnum.COUNTY) final County county) {
>72|    County dbCounty = countyRepository.findByPk(county.getPk()); // Dette hentes ut for å unngå at denne metoden
skal brukes for å endre County-objektet
73|    dbCounty.setScanningConfig(county.getScanningConfig());
74|    return countyRepository.update(userData, dbCounty);
```

### Issue 4BD42F42DFA4ECD78B7C285F9A528D3B:

**Source :****File** admin-backend/src/main/java/no/evote/service/configuration/CountyServiceEjb.java 139**Action** Assign county = findByPk(...): CountyRepository.findByPk may return NULL**Trace id** 18

```
136|@Security(accesses = Konfigurasjon_Grunnlagsdata_Godkjenne, type = WRITE)
137|@AuditLog(eventClass = CountyAuditEventForLocalConfiguration.class, eventType = StatusChanged, objectSource =
ReturnValue)
138|public County approve(final UserData userData, @SecureEntity(areaLevel = COUNTY) final Long countyPk) { >139|
County county = countyRepository.findByPk(countyPk);
140|    return countyRepository.update(userData, setCountyStatus(county,
MunicipalityStatusEnum.APPROVED_CONFIGURATION.id()));
```

**Sink :****File** admin-backend/src/main/java/no/evote/service/configuration/CountyServiceEjb.java**139 Action** Assign county = findByPk(...): CountyRepository.findByPk may return NULL**Trace id** 18

```
136|@Security(accesses = Konfigurasjon_Grunnlagsdata_Godkjenne, type = WRITE)
137|@AuditLog(eventClass = CountyAuditEventForLocalConfiguration.class, eventType = StatusChanged, objectSource =
ReturnValue)
138|public County approve(final UserData userData, @SecureEntity(areaLevel = COUNTY) final Long countyPk) {
>139|    County county = countyRepository.findByPk(countyPk);
140|    return countyRepository.update(userData, setCountyStatus(county,
MunicipalityStatusEnum.APPROVED_CONFIGURATION.id()));
```

### Issue EBAD465EEEEB09D617AA7DB40CAC5CFC:

**Source :****File** admin-backend/src/main/java/no/evote/service/configuration/CountyServiceEjb.java 131**Action** Assign county = findByPk(...): CountyRepository.findByPk may return NULL**Trace id** 18

```
128|@Security(accesses = Konfigurasjon_Grunnlagsdata_Oppheve, type = WRITE)
```

```

129|@AuditLog(eventClass = CountyAuditEventForLocalConfiguration.class, eventType = StatusChanged, objectSource =
ReturnVaue)
130|public County reject(final UserData userData, @SecureEntity(areaLevel = COUNTY) final Long countyPk) {
>131|     County county = countyRepository.findByPk(countyPk);
132|     return countyRepository.update(userData, setCountyStatus(county,
MunicipalityStatusEnum.LOCAL_CONFIGURATION.id()));

```

**Sink :****File** admin-backend/src/main/java/no/evote/service/configuration/CountyServiceEjb.java**131 Action** Assign county = findByPk(...): CountyRepository.findByPk may return NULL**Trace id** 18

```

128|@Security(accesses = Konfigurasjon_Grunnlagsdata_Oppheve, type = WRITE)
129|@AuditLog(eventClass = CountyAuditEventForLocalConfiguration.class, eventType = StatusChanged, objectSource =
ReturnVaue)
130|public County reject(final UserData userData, @SecureEntity(areaLevel = COUNTY) final Long countyPk) {
>131|     County county = countyRepository.findByPk(countyPk);
132|     return countyRepository.update(userData, setCountyStatus(county,
MunicipalityStatusEnum.LOCAL_CONFIGURATION.id()));

```

## 6.40 ElectionCardConfigApplicationService.java

### Issue 5FB0AA4A56134A5B509FA07F7119D8E7:

**Source :****File** admin-backend/src/main/java/no/valg/eva/admin/configuration/application/ElectionCardConfigApplicationService.java 96**Action** BranchTaken Branch taken: path.isMunicipalityLevel()

```

93|reportingUnit.setPostalCode(electionCard.getReportingUnit().getPostalCode());
94|reportingUnit.setPostTown(electionCard.getReportingUnit().getPostTown());
95|
>96|if (path.isMunicipalityLevel()) {
97|    Municipality municipality =
municipalityRepository.findByPk(reportingUnit.getMvArea().getMunicipality().getPk());
98|    municipality.checkVersion(electionCard);
99|    municipality.setElectionCardText(electionCard.getInfoText());

```

**Sink :****File** admin-backend/src/main/java/no/valg/eva/admin/configuration/application/ElectionCardConfigApplicationService.java**110 Action** Assign dbPlace = findByPk(...): PollingPlaceRepository.findByPk may return NULL **Trace id** 10

```

107|    dbPlace.setInfoText(place.getInfoText());
108|    pollingPlaceApplicationService.create(userData, dbPlace);
109|} else {
>110|    PollingPlace dbPlace = pollingPlaceRepository.findByPk(place.getPk());
111|    dbPlace.checkVersion(place);
112|    dbPlace.setInfoText(place.getInfoText());
113|    pollingPlaceRepository.update(userData, dbPlace);

```

### Issue 76A0F98C30EB12E21E8AA628E0DACE65:

**Source :****File** admin-backend/src/main/java/no/valg/eva/admin/configuration/application/ElectionCardConfigApplicationService.java 96**Action** BranchTaken Branch taken: path.isMunicipalityLevel()

```

93|reportingUnit.setPostalCode(electionCard.getReportingUnit().getPostalCode());
94|reportingUnit.setPostTown(electionCard.getReportingUnit().getPostTown());
95|
>96|if (path.isMunicipalityLevel()) {
97|    Municipality municipality =
municipalityRepository.findByPk(reportingUnit.getMvArea().getMunicipality().getPk());
98|    municipality.checkVersion(electionCard);
99|    municipality.setElectionCardText(electionCard.getInfoText());

```

**Sink :****File** admin-backend/src/main/java/no/valg/eva/admin/configuration/application/ElectionCardConfigApplicationService.java**105 Action** Assign dbDistrict = findByPk(...): PollingDistrictRepository.findByPk may return NULL **Trace id** 7

```

102|for (ElectionDayPollingPlace place : electionCard.getPlaces()) {
103|    if (place.getPk() == null) {
104|        // Create polling place with default values if it does not exist.
>105|        PollingDistrict dbDistrict = pollingDistrictRepository.findByPk(place.getParentPk());
106|        PollingPlace dbPlace = dbDistrict.createElectionDayPollingPlace();
107|        dbPlace.setInfoText(place.getInfoText());

```

```
108| pollingPlaceApplicationService.create(userData, dbPlace);
```

### Issue AA608A041937DC47CB4614655CoB6CB3:

**Source :**

**File** admin-backend/src/main/java/no/valg/eva/admin/configuration/application/ElectionCardConfigApplicationService.java 96

**Action** BranchTaken Branch taken: path.isMunicipalityLevel()

```
93| reportingUnit.setPostalCode(electionCard.getReportingUnit().getPostalCode());
94| reportingUnit.setPostTown(electionCard.getReportingUnit().getPostTown());
95|
>96| if (path.isMunicipalityLevel()) {
97|     Municipality municipality =
municipalityRepository.findByPk(reportingUnit.getMvArea().getMunicipality().getPk());
98|     municipality.checkVersion(electionCard);
99|     municipality.setElectionCardText(electionCard.getInfoText());
```

**Sink :**

**File** admin-backend/src/main/java/no/valg/eva/admin/configuration/application/ElectionCardConfigApplicationService.java 97

**Action** Assign municipality = findByPk(...): MunicipalityRepository.findByPk may return NULL **Trace id** 6

```
94| reportingUnit.setPostTown(electionCard.getReportingUnit().getPostTown());
95|
96| if (path.isMunicipalityLevel()) {
>97|     Municipality municipality =
municipalityRepository.findByPk(reportingUnit.getMvArea().getMunicipality().getPk());
98|     municipality.checkVersion(electionCard);
99|     municipality.setElectionCardText(electionCard.getInfoText());
```

## 6.41 ElectionEventApplicationService.java

### Issue 16AF582965EB388DB4EEA17C25F2D39D:

**Source :**

**File** admin-backend/src/main/java/no/valg/eva/admin/configuration/application/ElectionEventApplicationService.java 149

**Action** Assign findElectionDayByPk(...): ElectionEventRepository.findElectionDayByPk may return NULL **Trace id** 8

```
146| @Override
147| @Security(accesses = Konfigurasjon_Valghendelse_Rediger, type = READ)
148| public ElectionDay findElectionDayByPk(final UserData userData, final Long electionDayPk) { >149|     return
toDto(electionEventRepository.findElectionDayByPk(electionDayPk));
150| }
151|
152| @Override
```

**Sink :**

**File** admin-backend/src/main/java/no/valg/eva/admin/configuration/application/ElectionEventApplicationService.java 149

**Action** Assign findElectionDayByPk(...): ElectionEventRepository.findElectionDayByPk may return NULL **Trace id** 8

```
146| @Override
147| @Security(accesses = Konfigurasjon_Valghendelse_Rediger, type = READ)
148| public ElectionDay findElectionDayByPk(final UserData userData, final Long electionDayPk) { >149|     return
toDto(electionEventRepository.findElectionDayByPk(electionDayPk));
150| }
151|
152| @Override
```

## 6.42 ElectionEventDomainService.java

### Issue 03B8BD69A46ADC1CB6A1808A657F507E:

**Source :**

**File** admin-backend/src/main/java/no/valg/eva/admin/configuration/domain/service/ElectionEventDomainService.java 501 **Action** Assign findByPk(...): ElectionEventRepository.findByPk may return NULL

**Trace id** 3

```
500| public ElectionEvent update(final UserData userData, final ElectionEvent electionEvent, final Set<Locale>
localeSet) {
>501|     ElectionEventStatusEnum dbStatus =
electionEventRepository.findByPk(electionEvent.getPk()).getElectionEventStatusEnum();
502|     ElectionEvent updatedElectionEvent = electionEventRepository.update(userData, electionEvent);
503|     if (localeSet != null && !localeSet.isEmpty()) {
504|         updateLocalesForElectionEvent(userData, localeSet, updatedElectionEvent);
```

**Sink :**

**File** admin-backend/src/main/java/no/valg/eva/admin/configuration/domain/service/ElectionEventDomainService.java 501

**Action** Assign findByPk(...): ElectionEventRepository.findByPk may return NULL

**Trace id** 3



```

500|public ElectionEvent update(final UserData userData, final ElectionEvent electionEvent, final Set<Locale>
localeSet) {
>501|     ElectionEventStatusEnum dbStatus =
electionEventRepository.findByPk(electionEvent.getPk()).getElectionEventStatusEnum();
502|     ElectionEvent updatedElectionEvent = electionEventRepository.update(userData, electionEvent);
503|     if (localeSet != null && !localeSet.isEmpty()) {
504|         updateLocalesForElectionEvent(userData, localeSet, updatedElectionEvent);

```

### Issue 1EC244D7AFC4B5BBD6BC2F623AA49724:

#### Source :

File admin-backend/src/main/java/no/valg/eva/admin/configuration/domain/service/ElectionEventDomainService.java 550

Action Assign electionEvent = findByPk(...): ElectionEventRepository.findByPk may return NULL Trace id 3

```

549|public void approveConfiguration(UserData userData, Long pk) {
>550|     ElectionEvent electionEvent = electionEventRepository.findByPk(pk);
551|
electionEvent.setElectionEventStatus(electionEventRepository.findElectionEventStatusById(ElectionEventStatusEnum.APPRO
VED_CONFIGURATION.id()));
552|     electionEventRepository.update(userData, electionEvent);

```

#### Sink :

File admin-backend/src/main/java/no/valg/eva/admin/configuration/domain/service/ElectionEventDomainService.java 550

Action Assign electionEvent = findByPk(...): ElectionEventRepository.findByPk may return NULL Trace id 3

```

549|public void approveConfiguration(UserData userData, Long pk) {
>550|     ElectionEvent electionEvent = electionEventRepository.findByPk(pk);
551|
electionEvent.setElectionEventStatus(electionEventRepository.findElectionEventStatusById(ElectionEventStatusEnum.APPRO
VED_CONFIGURATION.id()));
552|     electionEventRepository.update(userData, electionEvent);

```

## 6.43 ElectionEventRepository.java

### Issue DBC1C465FF35F881239D60284E4E4F2F:

#### Source :

File admin-backend-common/src/main/java/no/valg/eva/admin/backend/configuration/repository/ElectionEventRepository.java 130 Action Assign electionEvent = findByPk(...): ElectionEventRepository.findByPk may return NULL

Trace id 3

```

129|public void deleteElectionEvent(final UserData userData, final Long electionEventPk) {
>130|     ElectionEvent electionEvent = findByPk(electionEventPk);
131|     if (electionEvent.getId().equals(ROOT_ELECTION_EVENT_ID)) {
132|         throw new EvoteException("Cannot delete admin election event");

```

#### Sink :

File admin-backend-common/src/main/java/no/valg/eva/admin/backend/configuration/repository/ElectionEventRepository.java 130 Action Assign electionEvent = findByPk(...): ElectionEventRepository.findByPk may return NULL

Trace id 3

```

129|public void deleteElectionEvent(final UserData userData, final Long electionEventPk) {
>130|     ElectionEvent electionEvent = findByPk(electionEventPk);
131|     if (electionEvent.getId().equals(ROOT_ELECTION_EVENT_ID)) { 132|         throw new EvoteException("Cannot
delete admin election event");

```

## 6.44 ElectionEventResourceBundleProvider.java

### Issue 1E845BFD298C4F64E1851A9C31DE8AFA:

#### Source :

File admin-backend/src/main/java/no/valg/eva/admin/backend/reporting/jasperserver/ElectionEventResourceBundleProvider.java 71

Action Assign getClassLoader(): Class.getClassLoader may return NULL

Rule id 4280F38B-9FDB-454E-B495-89CF45CD51B7

```

68|String bundleName = toBundleName(REPORT_BUNDLE_NAME, locale.toJavaLocale());
69|String resourceName = toResourceName(bundleName, "properties");
70|Properties properties = new Properties();
>71|try (InputStreamReader reader = new
InputStreamReader(getClass().getClassLoader().getResourceAsStream(resourceName), StandardCharsets.ISO_8859_1)) {
72|     properties.load(reader);
73|} catch (IOException e) {
74|     LOG.error("IOException ved lasting av " + resourceName, e);

```

#### Sink :

**File** admin-backend/src/main/java/no/valg/eva/admin/backend/reporting/jasperserver/ElectionEventResourceBundleProvider.java 71  
**Action** Assign getClassLoader(): Class.getClassLoader may return NULL

**Rule id** 4280F38B-9FDB-454E-B495-89CF45CD51B7

```
68|String bundleName = toBundleName(REPORT_BUNDLE_NAME, locale.toJavaLocale());
69|String resourceName = toResourceName(bundleName, "properties");
70|Properties properties = new Properties();
>71|try (InputStreamReader reader = new
InputStreamReader(getClass().getClassLoader().getResourceAsStream(resourceName), StandardCharsets.ISO_8859_1)) {
72|    properties.load(reader);
73|} catch (IOException e) {
74|    LOG.error("IOException ved lasting av " + resourceName, e);
```

## 6.45 ElectionGroupApplicationService.java

**Issue** 8BB879F1FE93C65DA8D15CFEC57FDF68:

**Source :**

**File** admin-backend/src/main/java/no/valg/eva/admin/configuration/application/ElectionGroupApplicationService.java 48  
**Action** Assign electionEvent = findByPk(...): ElectionEventRepository.findByPk may return NULL **Trace id** 3

```
45|@Override
46|@Security(accesses = { Konfigurasjon_Grunnlagsdata_Redigere, Konfigurasjon_Opptellingsvalgstyrer }, type = READ)
47|public List<ElectionGroup> getElectionGroups(UserData userData) {
>48|    ElectionEvent electionEvent = electionEventRepository.findByPk(userData.getElectionEventPk());
49|    return mvElectionRepository.findByPathAndLevel(electionEvent.electionPath(), ElectionLevelEnum.ELECTION_GROUP)
50|        .stream()
51|        .map(ElectionGroupMapper::toElectionGroup)
```

**Sink :**

**File** admin-backend/src/main/java/no/valg/eva/admin/configuration/application/ElectionGroupApplicationService.java 48  
**Action** Assign electionEvent = findByPk(...): ElectionEventRepository.findByPk may return NULL **Trace id** 3

```
45|@Override
46|@Security(accesses = { Konfigurasjon_Grunnlagsdata_Redigere, Konfigurasjon_Opptellingsvalgstyrer }, type = READ)
47|public List<ElectionGroup> getElectionGroups(UserData userData) {
>48|    ElectionEvent electionEvent = electionEventRepository.findByPk(userData.getElectionEventPk());
49|    return mvElectionRepository.findByPathAndLevel(electionEvent.electionPath(), ElectionLevelEnum.ELECTION_GROUP)
50|        .stream()
51|        .map(ElectionGroupMapper::toElectionGroup)
```

## 6.46 ElectionGroupServiceBean.java

**Issue** 6C502B9C4E2D449C5B7F780F4F2A3A20:

**Source :**

**File** admin-backend/src/main/java/no/evote/service/configuration/ElectionGroupServiceBean.java 45 **Action** Assign existing = findByPk(...): ElectionGroupRepository.findByPk may return NULL

**Trace id** 23

```
44|public SaveElectionResponse update(UserData userData, ElectionGroup electionGroup) {
>45|    no.valg.eva.admin.configuration.domain.model.ElectionGroup existing =
electionGroupRepository.findByPk(electionGroup.getElectionGroupRef().getPk());
46|    existing.checkVersion(electionGroup);
47|    if (hasIdChanged(electionGroup, existing)) {
48|        if (existing.getElectionEvent().hasElectionGroupsById(electionGroup.getId())) {
```

**Sink :**

**File** admin-backend/src/main/java/no/evote/service/configuration/ElectionGroupServiceBean.java 45  
**Action** Assign existing = findByPk(...): ElectionGroupRepository.findByPk may return NULL **Trace id** 23

```
44|public SaveElectionResponse update(UserData userData, ElectionGroup electionGroup) {
>45|    no.valg.eva.admin.configuration.domain.model.ElectionGroup existing =
electionGroupRepository.findByPk(electionGroup.getElectionGroupRef().getPk());
46|    existing.checkVersion(electionGroup);
47|    if (hasIdChanged(electionGroup, existing)) {
48|        if (existing.getElectionEvent().hasElectionGroupsById(electionGroup.getId())) {
```

## 6.47 ElectionServiceBean.java

**Issue** 0F52B3546C239F09D2E59F57A2788DD9:

**Source :**

**File** admin-backend/src/main/java/no/evote/service/configuration/ElectionServiceBean.java 52 **Action**  
Assign existingElection = findByPk(...) : ElectionRepository.findByPk may return NULL  
**Trace id** 4

```

49| * @param newElection election to update
50| */
51|public SaveElectionResponse update(UserData userData,
no.valg.eva.admin.common.configuration.model.election.Election newElection) {
>52|     Election existingElection = electionRepository.findByPk(newElection.getElectionRef().getPk());
53|     existingElection.checkVersion(newElection);
54|     if (hasIdChanged(newElection, existingElection)) {
55|         if (existingElection.getElectionGroup().hasElectionWithId(newElection.getId())) {

```

**Sink :****File** admin-backend/src/main/java/no/evote/service/configuration/ElectionServiceBean.java 52**Action** Assign existingElection = findByPk(...): ElectionRepository.findByPk may return NULL **Trace id** 4

```

49| * @param newElection election to update
50| */
51|public SaveElectionResponse update(UserData userData,
no.valg.eva.admin.common.configuration.model.election.Election newElection) {
>52|     Election existingElection = electionRepository.findByPk(newElection.getElectionRef().getPk());
53|     existingElection.checkVersion(newElection);
54|     if (hasIdChanged(newElection, existingElection)) {
55|         if (existingElection.getElectionGroup().hasElectionWithId(newElection.getId())) {

```

## 6.48 EntityManagerFactoryUtil.java

**Issue** 9FCF75A49ABC8EA80CDB143507F126EE:**Source :****File** admin-backend-testtools/src/main/java/no/valg/eva/admin/test/EntityManagerFactoryUtil.java 67**Action** Assign getClassLoader(): Class.getClassLoader may return NULL**Rule id** 4280F38B-9FDB-454E-B495-89CF45CD51B7

```

65|private URL persistenceXmlUrl() {
66|     try {
>67|         Enumeration<URL> resources = getClass().getClassLoader().getResources(PERSISTENCE_XML);
68|         if (resources.hasMoreElements()) {
69|             return resources.nextElement();
70|         } else {

```

**Sink :****File** admin-backend-testtools/src/main/java/no/valg/eva/admin/test/EntityManagerFactoryUtil.java 67**Action** Assign getClassLoader(): Class.getClassLoader may return NULL**Rule id** 4280F38B-9FDB-454E-B495-89CF45CD51B7

```

65|private URL persistenceXmlUrl() {
66|     try {
>67|         Enumeration<URL> resources = getClass().getClassLoader().getResources(PERSISTENCE_XML);
68|         if (resources.hasMoreElements()) {
69|             return resources.nextElement();
70|         } else {

```

## 6.49 ErrorPageRenderer.java

**Issue** EoCCAeF75746F26E3D3206FC76E42AB3:**Source :****File** admin-frontend/src/main/java/no/evote/presentation/exceptions/ErrorPageRenderer.java 122**Action** Assign getClassLoader(): Class.getClassLoader may return NULL**Rule id** 4280F38B-9FDB-454E-B495-89CF45CD51B7

```

120|private static String getPageTemplate(Error error, String incidentCode) throws IOException { 121|
>122|     InputStream is = ErrorPageRenderer.class.getClassLoader().getResourceAsStream("errorpage.html");
123|     String title = error.getTitle();
124|     String message = error.getMessage();
125|     int statusCode = error.getStatusCode();

```

**Sink :****File** admin-frontend/src/main/java/no/evote/presentation/exceptions/ErrorPageRenderer.java 122**Action** Assign getClassLoader(): Class.getClassLoader may return NULL**Rule id** 4280F38B-9FDB-454E-B495-89CF45CD51B7

```

120|private static String getPageTemplate(Error error, String incidentCode) throws IOException { 121|
>122|     InputStream is = ErrorPageRenderer.class.getClassLoader().getResourceAsStream("errorpage.html");
123|     String title = error.getTitle();
124|     String message = error.getMessage();
125|     int statusCode = error.getStatusCode();

```

## 6.50 ExportImportOperatorsServiceBean.java

**Issue** 11C9801394683A46B278F8DFEB988767:**Source :**

**File** admin-backend/src/main/java/no/evote/service/util/ExportImportOperatorsServiceBean.java 254 **Action** Assign electionEvent = findByPk(...): ElectionEventRepository.findByPk may return NULL

**Trace id** 3

```

252|public List<ImportOperatorMessage> importOperatorRoles(final UserData userData, final Long electionEventPk, final
byte[] data) {
253|    final ImportMessages messages = new ImportMessages();
>254|    final ElectionEvent electionEvent = electionEventRepository.findByPk(electionEventPk);
255|    try {
256|        List<String[]> rows = Lists.transform(ExcelUtil.getRowDataFromExcelFile(new
ByteArrayInputStream(data)).getRows(), VALUES_FROM_PAIRS_F::apply);

```

**Sink :**

**File** admin-backend/src/main/java/no/evote/service/util/ExportImportOperatorsServiceBean.java 254  
**Action** Assign electionEvent = findByPk(...): ElectionEventRepository.findByPk may return NULL **Trace**

**id** 3

```

252|public List<ImportOperatorMessage> importOperatorRoles(final UserData userData, final Long electionEventPk, final
byte[] data) {
253|    final ImportMessages messages = new ImportMessages();
>254|    final ElectionEvent electionEvent = electionEventRepository.findByPk(electionEventPk);
255|    try {
256|        List<String[]> rows = Lists.transform(ExcelUtil.getRowDataFromExcelFile(new
ByteArrayInputStream(data)).getRows(), VALUES_FROM_PAIRS_F::apply);

```

### Issue B742FB71406DBDB7018DB56D96DABD38:

**Source :**

**File** admin-backend/src/main/java/no/evote/service/util/ExportImportOperatorsServiceBean.java 254 **Action** Assign electionEvent = findByPk(...): ElectionEventRepository.findByPk may return NULL

**Trace id** 3

```

252|public List<ImportOperatorMessage> importOperatorRoles(final UserData userData, final Long electionEventPk, final
byte[] data) {
253|    final ImportMessages messages = new ImportMessages();
>254|    final ElectionEvent electionEvent = electionEventRepository.findByPk(electionEventPk);
255|    try {
256|        List<String[]> rows = Lists.transform(ExcelUtil.getRowDataFromExcelFile(new
ByteArrayInputStream(data)).getRows(), VALUES_FROM_PAIRS_F::apply);

```

**Sink :**

**File** admin-backend/src/main/java/no/evote/service/util/ExportImportOperatorsServiceBean.java 266

**Action** InOutCall ?.hasCorrectElectionEventId(electionEvent, ?): ExportImportOperatorsServiceBean.hasCorrectElectionEventId may return NULL

**Trace id** 53

```

263|for (String[] row : rows) {
264|    Logger.getLogger(this.getClass()).info(StringUtils.join(row, ','));
265|
>266|    if (hasCorrectElectionEventId(electionEvent, row)) {
267|
268|        String operatorId = row[0];
269|        Operator operator = operatorCache.get(operatorId);

```

## 6.51 ExportServiceBean.java

### Issue 49354C7D061D78917E3DF02A6AB18E47:

**Source :**

**File** admin-backend/src/main/java/no/evote/service/ExportServiceBean.java 885

**Action** Assign contest = findByPk(...): ContestRepository.findByPk may return NULL

**Trace id** 0

```

882|private void insertReportingUnits(
883|    Document doc, String localeId, Map<String, HashMap<String, HashMap<String, String>>> text, Map<String,
Object> contestRow, Element contestElement) {
884|    Integer contestPk = (Integer) contestRow.get(CONTEST_PK);
>885|    Contest contest = contestRepository.findByPk(contestPk.longValue());
886|    if (contest.isOnBoroughLevel()) {
887|        insertReportingUnitsForBoroughContest(doc, contestElement, contest, text, localeId);

```

**Sink :**

**File** admin-backend/src/main/java/no/evote/service/ExportServiceBean.java 885

**Action** Assign contest = findByPk(...): ContestRepository.findByPk may return NULL

**Trace id** 0

```

882|private void insertReportingUnits(
883|    Document doc, String localeId, Map<String, HashMap<String, HashMap<String, String>>> text, Map<String,
Object> contestRow, Element contestElement) {
884|    Integer contestPk = (Integer) contestRow.get(CONTEST_PK);
>885|    Contest contest = contestRepository.findByPk(contestPk.longValue());
886|    if (contest.isOnBoroughLevel()) {
887|        insertReportingUnitsForBoroughContest(doc, contestElement, contest, text, localeId);

```

### Issue 8AF6A454B88DC047E086E7939CF040A0:

**Source :**

**File** admin-backend/src/main/java/no/evote/service/ExportServiceBean.java 511

**Action** Assign getClassLoader(): Class.getClassLoader may return NULL

**Rule id** 4280F38B-9FDB-454E-B495-89CF45CD51B7

```
509|private byte[] xslTransform(final String xslFilename, final Document doc) {
510|    InputStream is;
>511|    is = this.getClass().getClassLoader().getResourceAsStream("xsl" + File.separator + xslFilename);
512|
513|    if (is == null) {
514|        throw new EvoteException("Unable to find file: " + xslFilename);
```

**Sink :**

**File** admin-backend/src/main/java/no/evote/service/ExportServiceBean.java 511

**Action** Assign getClassLoader(): Class.getClassLoader may return NULL

**Rule id** 4280F38B-9FDB-454E-B495-89CF45CD51B7

```
509|private byte[] xslTransform(final String xslFilename, final Document doc) {
510|    InputStream is;
>511|    is = this.getClass().getClassLoader().getResourceAsStream("xsl" + File.separator + xslFilename);
512|
513|    if (is == null) {
514|        throw new EvoteException("Unable to find file: " + xslFilename);
```

## Issue BA7ABEB5C9FBB598B44B3B9D9DC26820:

**Source :**

**File** admin-backend/src/main/java/no/evote/service/ExportServiceBean.java 341

**Action** Assign localBatch = findByPk(...): BatchRepository.findByPk may return NULL

**Trace id** 57

```
338|final no.valg.eva.admin.configuration.domain.model.Locale locale = electionEvent.getLocale();
339|String pattern = "YYYY-MM-dd'T'HH:mm:ssZ";
340|DateTimeFormatter dateTimeFormatter =
DateTimeFormat.forPattern(pattern).withLocale(locale.toJavaLocale()); >341|Batch localBatch =
batchRepository.findByPk(batch.getPk()); 342|
343|DocumentBuilderFactory factory = DocumentBuilderFactory.newInstance();
344|DocumentBuilder builder;
```

**Sink :**

**File** admin-backend/src/main/java/no/evote/service/ExportServiceBean.java 341

**Action** Assign localBatch = findByPk(...): BatchRepository.findByPk may return NULL **Trace id** 57

```
338|final no.valg.eva.admin.configuration.domain.model.Locale locale = electionEvent.getLocale();
339|String pattern = "YYYY-MM-dd'T'HH:mm:ssZ";
340|DateTimeFormatter dateTimeFormatter =
DateTimeFormat.forPattern(pattern).withLocale(locale.toJavaLocale()); >341|Batch localBatch =
batchRepository.findByPk(batch.getPk()); 342|
343|DocumentBuilderFactory factory = DocumentBuilderFactory.newInstance();
344|DocumentBuilder builder;
```

## Issue D17B47CF49FE1533E9858281C5B5DC59:

**Source :**

**File** admin-backend/src/main/java/no/evote/service/ExportServiceBean.java 314

**Action** Assign electionEvent = findByPk(...): ElectionEventRepository.findByPk may return NULL

**Trace id** 3

```
311|private ReportingUnitRepository reportingUnitRepository;
312|
313|public void generateEML(UserData userData, Long electionEventPk) {
>314|    ElectionEvent electionEvent = electionEventRepository.findByPk(electionEventPk);
315|    Batch batch = startBatch(userData, CONFIGURATION_DOWNLOAD, null);
316|    generateBatch(userData, electionEvent, batch, new BatchContent[]{
317|        (userData1, electionEventId, batch1, zout) -> {
```

**Sink :**

**File** admin-backend/src/main/java/no/evote/service/ExportServiceBean.java 314

**Action** Assign electionEvent = findByPk(...): ElectionEventRepository.findByPk may return NULL **Trace id** 3

```
311|private ReportingUnitRepository reportingUnitRepository;
312|
313|public void generateEML(UserData userData, Long electionEventPk) {
>314|    ElectionEvent electionEvent = electionEventRepository.findByPk(electionEventPk);
315|    Batch batch = startBatch(userData, CONFIGURATION_DOWNLOAD, null);
316|    generateBatch(userData, electionEvent, batch, new BatchContent[]{
317|        (userData1, electionEventId, batch1, zout) -> {
```

## 6.52 GenericCacheManager.java

### Issue D38A26844E64B7517FD7F8C986CC22EB:

**Source :**

**File** admin-frontend/src/main/java/no/evote/presentation/cache/GenericCacheManager.java 30

**Action** Assign getClassLoader(): Class.getClassLoader may return NULL **Rule id** 4280F38B-9FDB-454E-B495-89CF45CD51B7

```
27|    LOG.debug("init()");
28|
```

```

29| LOG.debug("Bruker lokal cache (eneste typen støttet p.t.");
>30| CacheManager.create(getClass().getClassLoader().getResourceAsStream("ehcache-local.xml")); 31| manager =
CacheManager.getInstance();

```

**Sink :**

**File** admin-frontend/src/main/java/no/evote/presentation/cache/GenericCacheManager.java 30

**Action** Assign getClassLoader() : Class.getClassLoader may return NULL **Rule id** 4280F38B-9FDB-454E-B495-89CF45CD51B7

```

27| LOG.debug("init()");
28|
29| LOG.debug("Bruker lokal cache (eneste typen støttet p.t.");
>30| CacheManager.create(getClass().getClassLoader().getResourceAsStream("ehcache-local.xml")); 31| manager =
CacheManager.getInstance();

```

## 6.53 ImportOperatorServiceEjb.java

### Issue 48C4C9488CDEA86F8C2660C8C7C735B7:

**Source :**

**File** admin-rbac/src/main/java/no/valg/eva/admin/rbac/service/ImportOperatorServiceEjb.java 81 **Action**

BranchNotTaken Branch not taken: (operatorInfoList != null)

```

80|private void importOperators(UserData userData, List<? extends ImportOperatorRoleInfo> operatorInfoList, String
roleId) {
>81|   if (operatorInfoList == null) {
82|       return;

```

**Sink :**

**File** admin-rbac/src/main/java/no/valg/eva/admin/rbac/service/ImportOperatorServiceEjb.java 85

**Action** Assign ee = findByPk(...): ElectionEventRepository.findByPk may return NULL **Trace id** 3

```

82|   return;
83|}
84|
>85|ElectionEvent ee = electionEventRepository.findByPk(userData.getElectionEventPk());
86|MvElection mvElection = userData.getOperatorRole().getMvElection();
87|Role role = findImportRole(ee, roleId);
88|for (ImportOperatorRoleInfo importOperator : operatorInfoList) {

```

## 6.54 LegacyPollingDistrictServiceEjb.java

### Issue 14393745BA23D7298738B3CEE6979169:

**Source :**

**File** admin-backend/src/main/java/no/evote/service/configuration/LegacyPollingDistrictServiceEjb.java 88 **Action** Assign pd = findByPk(...): PollingDistrictRepository.findByPk may return NULL

**Trace id** 7

```

85|@AuditLog(eventClass = PollingDistrictAuditEvent.class, eventType = Delete)
86|public void delete(final UserData userData, PollingDistrict pollingDistrict) {
87|   // Check first that the polling district has no parents
>88|   PollingDistrict pd = pollingDistrictRepository.findByPk(pollingDistrict.getPk());
89|   if (pd.getPollingDistrict() != null) {
90|       throw new EvoteException("Not possible to delete polling districts that have parents.");

```

**Sink :**

**File** admin-backend/src/main/java/no/evote/service/configuration/LegacyPollingDistrictServiceEjb.java 88

**Action** Assign pd = findByPk(...): PollingDistrictRepository.findByPk may return NULL **Trace id** 7

```

85|@AuditLog(eventClass = PollingDistrictAuditEvent.class, eventType = Delete)
86|public void delete(final UserData userData, PollingDistrict pollingDistrict) {
87|   // Check first that the polling district has no parents
>88|   PollingDistrict pd = pollingDistrictRepository.findByPk(pollingDistrict.getPk());
89|   if (pd.getPollingDistrict() != null) {
90|       throw new EvoteException("Not possible to delete polling districts that have parents.");

```

## 6.55 LegacyUserDataServiceBean.java

### Issue 7BAAE8F1FB96BoE7EFD42BC7CE45BEEC:

**Source :**

**File** admin-backend/src/main/java/no/evote/service/security/LegacyUserDataServiceBean.java 198

**Action** BranchNotTaken Branch not taken: exportableOperators.contains(operator)

```

195|public Document exportAccessToken(final UserData userData, final Operator operator) {
196|    List<Operator> exportableOperators = operatorRepository.findOperatorsById(userData.getId());
197|
>198|    if (!exportableOperators.contains(operator)) {
199|        return null;

```

**Sink :****File** admin-backend/src/main/java/no/evote/service/security/LegacyUserDataServiceBean.java 202**Action** Assign electionEvent = findByPk(...): ElectionEventRepository.findByPk may return NULL **Trace id** 3

```

199|    return null;
200|}
201|
>202|ElectionEvent electionEvent = electionEventRepository.findByPk(operator.getElectionEvent().getPk()); 203|
204|Element root = new Element(LegacyUserDataService.ROOT_TAG);

```

## 6.56 LifecycleListener.java

**Issue** C68CE157BA96F1774FAB5CCD172A70BD:**Source :****File** admin-frontend/src/main/java/no/evote/lifecycle/LifecycleListener.java 28**Action** Assign getClassLoader(): Class.getClassLoader may return NULL **Rule id** 4280F38B-9FDB-454E-B495-89CF45CD51B7

```

27|private void initializeLog4j() {
>28|    Log4jUtil.configure(this.getClass().getClassLoader());
29|    log = Logger.getLogger(LifecycleListener.class);

```

**Sink :****File** admin-frontend/src/main/java/no/evote/lifecycle/LifecycleListener.java 28**Action** Assign getClassLoader(): Class.getClassLoader may return NULL **Rule id** 4280F38B-9FDB-454E-B495-89CF45CD51B7

```

27|private void initializeLog4j() {
>28|    Log4jUtil.configure(this.getClass().getClassLoader());
29|    log = Logger.getLogger(LifecycleListener.class);

```

## 6.57 ListProposalApplicationService.java

**Issue** D99DE28142FC0C9B27BE60B045F4E4CA:**Source :****File** admin-backend/src/main/java/no/valg/eva/admin/configuration/application/ListProposalApplicationService.java 98**Action** Assign dbContest = findByPk(...): ContestRepository.findByPk may return NULL **Trace id** 0

```

97|private void doSave(UserData userData, ListProposalConfig listProposal) { >
98|    Contest dbContest =
contestRepository.findByPk(listProposal.getContestPk()); 99|
dbContest.checkVersion(listProposal);
100|    contestMapper.updateEntityFromListProposalData(dbContest, listProposal.getContestListProposalData());
101|    contestRepository.update(userData, dbContest);

```

**Sink :****File** admin-backend/src/main/java/no/valg/eva/admin/configuration/application/ListProposalApplicationService.java 98**Action** Assign dbContest = findByPk(...): ContestRepository.findByPk may return NULL**Trace id** 0

```

97|private void doSave(UserData userData, ListProposalConfig listProposal) { >
98|    Contest dbContest =
contestRepository.findByPk(listProposal.getContestPk());
99|    dbContest.checkVersion(listProposal);
100|    contestMapper.updateEntityFromListProposalData(dbContest, listProposal.getContestListProposalData());
101|    contestRepository.update(userData, dbContest);

```

## 6.58 Log4jConfigurator.java

**Issue** 18B27B05B99D22E5B32A57338DA930FB:**Source :****File** admin-backend/src/main/java/no/evote/util/Log4jConfigurator.java 13**Action** Assign getClassLoader(): Class.getClassLoader may return NULL**Rule id** 4280F38B-9FDB-454E-B495-89CF45CD51B7

```

11|    @PostConstruct
12|    public void init() {
>13|        Log4jUtil.configure(this.getClass().getClassLoader());

```

**Sink :****File** admin-backend/src/main/java/no/evote/util/Log4jConfigurator.java 13**Action** Assign getClassLoader(): Class.getClassLoader may return NULL



**Rule id** 4280F38B-9FDB-454E-B495-89CF45CD51B7

```
11|     @PostConstruct
12|     public void init() {
>13|         Log4jUtil.configure(this.getClass().getClassLoader());
```

## 6.59 ManntallsimportMappingRepository.java

**Issue** 0C290DBBE03782D4296B531AD4BD2604:

**Source :**

**File** admin-configuration/src/main/java/no/valg/eva/admin/configuration/repository/ManntallsimportMappingRepository.java 67

**Action** Assign getClassLoader(): Class.getClassLoader may return NULL

**Rule id** 4280F38B-9FDB-454E-B495-89CF45CD51B7

```
66|private ManntallsimportMapping lesMappingFraClasspath() {
>67|    InputStreamReader inputStreamReader = new
InputStreamReader(this.getClass().getClassLoader().getResourceAsStream(filnavn));
68|    return new Gson().fromJson(inputStreamReader, ManntallsimportMapping.class);
```

**Sink :**

**File** admin-configuration/src/main/java/no/valg/eva/admin/configuration/repository/ManntallsimportMappingRepository.java 67

**Action** Assign getClassLoader(): Class.getClassLoader may return NULL

**Rule id** 4280F38B-9FDB-454E-B495-89CF45CD51B7

```
66|private ManntallsimportMapping lesMappingFraClasspath() {
>67|    InputStreamReader inputStreamReader = new
InputStreamReader(this.getClass().getClassLoader().getResourceAsStream(filnavn));
68|    return new Gson().fromJson(inputStreamReader, ManntallsimportMapping.class);
```

**Issue** EA97012A511B5BE9ABD6B39C637D1FD5:

**Source :**

**File** admin-configuration/src/main/java/no/valg/eva/admin/configuration/repository/ManntallsimportMappingRepository.java 49

**Action** BranchNotTaken Branch not taken: (!mappingfil.exists())

```
47|private ManntallsimportMapping getManntallsimportMapping() {
48|    File mappingfil = new File(filnavn);
>49|    if (mappingfil.exists()) {
50|        return lesMappingFraFil(mappingfil);
51|    } else if (this.getClass().getClassLoader().getResource(filnavn) != null) { 52|        return
lesMappingFraClasspath();
```

**Sink :**

**File** admin-configuration/src/main/java/no/valg/eva/admin/configuration/repository/ManntallsimportMappingRepository.java 51

**Action** Assign getClassLoader(): Class.getClassLoader may return NULL **Rule id** 4280F38B-9FDB-454E-B495-89CF45CD51B7

```
48|File mappingfil = new File(filnavn);
49|if (mappingfil.exists()) {
50|    return lesMappingFraFil(mappingfil);
>51|} else if (this.getClass().getClassLoader().getResource(filnavn) != null) {
52|    return lesMappingFraClasspath();
53|} else {
54|    return ingenMapping();
```

## 6.60 MigrationFileUtils.java

**Issue** 5122A59DD62D0463A7795CD1CD95BBAE:

**Source :**

**File** admin-database/src/main/java/no/valg/eva/admin/database/tools/MigrationFileUtils.java 50

**Action** Assign getClassLoader(): Class.getClassLoader may return NULL

**Rule id** 4280F38B-9FDB-454E-B495-89CF45CD51B7

```
49|public static void copyFileFromClasspathToFilesystem(String onClasspath, String onFileSystemRelativeToBaseDir,
File baseDir) {
>50|    try (InputStream is = MigrationFileUtils.class.getClassLoader().getResourceAsStream(onClasspath)) {
51|        if (is == null) {
52|            throw new RuntimeException("Could not load '" + onClasspath + "' from class path.");
```

**Sink :**

**File** admin-database/src/main/java/no/valg/eva/admin/database/tools/MigrationFileUtils.java 50

**Action** Assign getClassLoader(): Class.getClassLoader may return NULL

**Rule id** 4280F38B-9FDB-454E-B495-89CF45CD51B7

```

49|public static void copyFileFromClasspathToFileSystem(String onClasspath, String onFileSystemRelativeToBaseDir,
File baseDir) {
>50|   try (InputStream is = MigrationFileUtils.class.getClassLoader().getResourceAsStream(onClasspath)) {
51|       if (is == null) {
52|           throw new RuntimeException("Could not load '" + onClasspath + "' from class path.");

```

**Issue 779BE2DDE7198EB7A3D692DD39821B1E:****Source :****File** admin-database/src/main/java/no/valg/eva/admin/database/tools/MigrationFileUtils.java 29**Action** Assign getClassLoader() : Class.getClassLoader may return NULL**Rule id** 4280F38B-9FDB-454E-B495-89CF45CD51B7

```

28|public static File createTempFileFromClasspath(String filename, File tempDir) {
>29|   try (InputStream is = MigrationFileUtils.class.getClassLoader().getResourceAsStream(filename)) { 30|
if (is == null) {
31|       throw new RuntimeException("Could not load '" + filename + "' from class path.");

```

**Sink :****File** admin-database/src/main/java/no/valg/eva/admin/database/tools/MigrationFileUtils.java 29**Action** Assign getClassLoader() : Class.getClassLoader may return NULL**Rule id** 4280F38B-9FDB-454E-B495-89CF45CD51B7

```

28|public static File createTempFileFromClasspath(String filename, File tempDir) {
>29|   try (InputStream is = MigrationFileUtils.class.getClassLoader().getResourceAsStream(filename)) { 30|
if (is == null) {
31|       throw new RuntimeException("Could not load '" + filename + "' from class path.");

```

## 6.61 ModifiedBallotDomainService.java

**Issue 876B5CF2EEE934075EE81282BDFC3A6E:****Source :****File** admin-counting/src/main/java/no/valg/eva/admin/counting/domain/modifiedballots/ModifiedBallotDomainService.java 69**Action** BranchTaken Branch taken: ballotCountForView~iterator.hasNext()

```

66|   finalCount finalCount, Map<String, BallotCount> idToBallotCountEntityMap,
67|   Map<String, String> idToActiveBatchIdMap, ModifiedBallotBatchProcess process) {
68|List<ModifiedBallotsStatus> modifiedBallotCounts = new ArrayList<>();
>69|for (no.valg.eva.admin.common.counting.model.BallotCount ballotCountForView : finalCount.getBallotCounts()) {
70|   String ballotCountMapId =
ballotCountId(ballotCountRepository.findByIdPk(ballotCountForView.getBallotCountRef().getPk()));
71|   BallotCount ballotCountEntity = idToBallotCountEntityMap.get(ballotCountMapId); 72|   ModifiedBallotsStatus
modifiedBallotsStatus = buildModifiedBallotsStatus(

```

**Sink :****File** admin-counting/src/main/java/no/valg/eva/admin/counting/domain/modifiedballots/ModifiedBallotDomainService.java 70**Action** Assign findByIdPk(...): BallotCountRepository.findByIdPk may return NULL **Trace****id** 36

```

67|   Map<String, String> idToActiveBatchIdMap, ModifiedBallotBatchProcess process) {
68|List<ModifiedBallotsStatus> modifiedBallotCounts = new ArrayList<>();
69|for (no.valg.eva.admin.common.counting.model.BallotCount ballotCountForView : finalCount.getBallotCounts()) {
>70|   String ballotCountMapId =
ballotCountId(ballotCountRepository.findByIdPk(ballotCountForView.getBallotCountRef().getPk()));
71|   BallotCount ballotCountEntity = idToBallotCountEntityMap.get(ballotCountMapId);
72|   ModifiedBallotsStatus modifiedBallotsStatus = buildModifiedBallotsStatus(
73|       ballotCountEntity, ballotCountForView, idToActiveBatchIdMap.get(ballotCountMapId), process);

```

## 6.62 MunicipalityApplicationService.java

**Issue 240123E95AF847FB621947987424E14A:****Source :****File** admin-configuration/src/main/java/no/valg/eva/admin/configuration/application/MunicipalityApplicationService.java 212**Action** Assign municipality = findByIdPk(...): MunicipalityRepository.findByIdPk may return NULL **Trace id** 6

```

209|@Security(accesses = Konfigurasjon_Grunnlagsdata_Godkjenne, type = WRITE)
210|@AuditLog(eventClass = MunicipalityAuditEventForLocalConfiguration.class, eventType = StatusChanged, objectSource
= ReturnValue)
211|public Municipality approve(UserData userData, @SecureEntity(areaLevel = MUNICIPALITY) Long municipalityPk) {
>212|   Municipality municipality = municipalityRepository.findByIdPk(municipalityPk);
213|   return municipalityRepository.update(userData,
214|       setMunicipalityStatus(municipality, MunicipalityStatusEnum.APPROVED_CONFIGURATION.id()));

```

**Sink :****File** admin-configuration/src/main/java/no/valg/eva/admin/configuration/application/MunicipalityApplicationService.java 212**Action** Assign municipality = findByIdPk(...): MunicipalityRepository.findByIdPk may return NULL **Trace id** 6

```

209|@Security(accesses = Konfigurasjon_Grunnlagsdata_Godkjenne, type = WRITE)
210|@AuditLog(eventClass = MunicipalityAuditEventForLocalConfiguration.class, eventType = StatusChanged, objectSource
= ReturnValue)
211|public Municipality approve(UserData userData, @SecureEntity(areaLevel = MUNICIPALITY) Long municipalityPk) {
>212|     Municipality municipality = municipalityRepository.findByPk(municipalityPk);
213|     return municipalityRepository.update(userData,
214|         setMunicipalityStatus(municipality, MunicipalityStatusEnum.APPROVED_CONFIGURATION.id()));

```

**Issue 49A7D1EC543DA62DE4F306241CB7448A:****Source :****File** admin-configuration/src/main/java/no/valg/eva/admin/configuration/application/MunicipalityApplicationService.java 203**Action** Assign municipality = findByPk(...): MunicipalityRepository.findByPk may return NULL **Trace id** 6

```

200|@Security(accesses = Konfigurasjon_Grunnlagsdata_Oppheve, type = WRITE)
201|@AuditLog(eventClass = MunicipalityAuditEventForLocalConfiguration.class, eventType = StatusChanged, objectSource
= ReturnValue)
202|public Municipality reject(UserData userData, @SecureEntity(areaLevel = MUNICIPALITY) Long municipalityPk) {
>203|     Municipality municipality = municipalityRepository.findByPk(municipalityPk);
204|     return municipalityRepository.update(userData,
205|         setMunicipalityStatus(municipality, MunicipalityStatusEnum.LOCAL_CONFIGURATION.id()));

```

**Sink :****File** admin-configuration/src/main/java/no/valg/eva/admin/configuration/application/MunicipalityApplicationService.java 203**Action** Assign municipality = findByPk(...): MunicipalityRepository.findByPk may return NULL **Trace id** 6

```

200|@Security(accesses = Konfigurasjon_Grunnlagsdata_Oppheve, type = WRITE)
201|@AuditLog(eventClass = MunicipalityAuditEventForLocalConfiguration.class, eventType = StatusChanged, objectSource
= ReturnValue)
202|public Municipality reject(UserData userData, @SecureEntity(areaLevel = MUNICIPALITY) Long municipalityPk) {
>203|     Municipality municipality = municipalityRepository.findByPk(municipalityPk);
204|     return municipalityRepository.update(userData,
205|         setMunicipalityStatus(municipality, MunicipalityStatusEnum.LOCAL_CONFIGURATION.id()));

```

**Issue C83272359B33021FB92A63BF012C28C6:****Source :****File** admin-configuration/src/main/java/no/valg/eva/admin/configuration/application/MunicipalityApplicationService.java 157**Action** Assign getStatus(...): MunicipalityRepository.getStatus may return NULL **Trace id** 60

```

154|@Override
155|@Security(accesses = Konfigurasjon_Grunnlagsdata_Oversikt_Manntallsavvik, type = READ)
156|public MunicipalityStatusEnum getStatus(UserData userData, KommuneSti kommuneSti) {
>157|     return municipalityRepository.getStatus(getMunicipality(userData,
kommuneSti.areaPath()).getPk()).toEnumValue();
158|}
159|
160|@Override

```

**Sink :****File** admin-configuration/src/main/java/no/valg/eva/admin/configuration/application/MunicipalityApplicationService.java 157**Action** Assign getStatus(...): MunicipalityRepository.getStatus may return NULL **Trace id** 60

```

154|@Override
155|@Security(accesses = Konfigurasjon_Grunnlagsdata_Oversikt_Manntallsavvik, type = READ)
156|public MunicipalityStatusEnum getStatus(UserData userData, KommuneSti kommuneSti) {
>157|     return municipalityRepository.getStatus(getMunicipality(userData,
kommuneSti.areaPath()).getPk()).toEnumValue();
158|}
159|
160|@Override

```

**Issue E46049225E6E2B06A4D92A900861D5C4:****Source :****File** admin-configuration/src/main/java/no/valg/eva/admin/configuration/application/MunicipalityApplicationService.java 127**Action** Assign dbMunicipality = findByPk(...): MunicipalityRepository.findByPk may return NULL **Trace id** 6

```

124|@AuditLog(eventClass = MunicipalityAuditEventForLocalConfiguration.class, eventType = Update, objectSource =
ReturnValue)
125|public Municipality updateScanningConfiguration(UserData userData, @SecureEntity(areaLevel = MUNICIPALITY) final
Municipality municipality) {
126|     // Dette hentes ut for å unngå at denne metoden skal brukes for å endre County-objektet
>127|     Municipality dbMunicipality = municipalityRepository.findByPk(municipality.getPk());
128|     dbMunicipality.setScanningConfig(municipality.getScanningConfig()); 129|     return
municipalityRepository.update(userData, municipality);

```

**Sink :****File** admin-configuration/src/main/java/no/valg/eva/admin/configuration/application/MunicipalityApplicationService.java 127**Action** Assign dbMunicipality = findByPk(...): MunicipalityRepository.findByPk may return NULL**Trace id** 6

```

124|@AuditLog(eventClass = MunicipalityAuditEventForLocalConfiguration.class, eventType = Update, objectSource =

```

```

ReturnValue)
125|public Municipality updateScanningConfiguration(UserData userData, @SecureEntity(areaLevel = MUNICIPALITY) final
Municipality municipality) {
126|    // Dette hentes ut for å unngå at denne metoden skal brukes for å endre County-objektet
>127|    Municipality dbMunicipality = municipalityRepository.findByPk(municipality.getPk());
128|    dbMunicipality.setScanningConfig(municipality.getScanningConfig()); 129|    return
municipalityRepository.update(userData, municipality);

```

## 6.63 MunicipalityDomainService.java

### Issue 13D9A2A0777FB20D43901A099C193B7E:

#### Source :

File admin-configuration/src/main/java/no/valg/eva/admin/configuration/domain/service/MunicipalityDomainService.java 65

Action Assign dbMunicipality = findByPk(...): MunicipalityRepository.findByPk may return NULL

Trace id 6

```

64|public List<MunicipalityOpeningHour> getOpeningHours(Municipality municipality) {
>65|    Municipality dbMunicipality = municipalityRepository.findByPk(municipality.getPk()); 66|
67|    return toSortedMunicipalityOpeningHourList(dbMunicipality.getOpeningHours());

```

#### Sink :

File admin-configuration/src/main/java/no/valg/eva/admin/configuration/domain/service/MunicipalityDomainService.java 65 Action Assign dbMunicipality = findByPk(...): MunicipalityRepository.findByPk may return NULL

Trace id 6

```

64|public List<MunicipalityOpeningHour> getOpeningHours(Municipality municipality) {
>65|    Municipality dbMunicipality = municipalityRepository.findByPk(municipality.getPk()); 66|
67|    return toSortedMunicipalityOpeningHourList(dbMunicipality.getOpeningHours());

```

## 6.64 MunicipalityRepository.java

### Issue B19935600B7E7F946412090D53D1569E:

#### Source :

File admin-configuration/src/main/java/no/valg/eva/admin/configuration/repository/MunicipalityRepository.java 204

Action Assign findEntityByPk(...): BaseRepository.findEntityByPk may return NULL Trace id 1

```

203|public MunicipalityStatus getStatus(Long municipalityPk) {
>204|    return findEntityByPk(MunicipalityStatus.class, findEntityByPk(Municipality.class,
municipalityPk).getMunicipalityStatus().getPk()); 205|}
206|
207|public boolean hasMunicipalities(County county) {

```

#### Sink :

File admin-configuration/src/main/java/no/valg/eva/admin/configuration/repository/MunicipalityRepository.java 204

Action Assign findEntityByPk(...): BaseRepository.findEntityByPk may return NULL Trace id 1

```

203|public MunicipalityStatus getStatus(Long municipalityPk) {
>204|    return findEntityByPk(MunicipalityStatus.class, findEntityByPk(Municipality.class,
municipalityPk).getMunicipalityStatus().getPk());
205|}
206|
207|public boolean hasMunicipalities(County county) {

```

## 6.65 MvAreaServiceBean.java

### Issue 5F0FB3169B028BF7B7D37CF3982AEBA1:

#### Source :

File admin-configuration/src/main/java/no/evote/service/configuration/MvAreaServiceBean.java 40

Action Assign municipality = findByPk(...): MunicipalityRepository.findByPk may return NULL Trace

id 6

```

39|public MvArea findByMunicipality(Long municipalityPk) {
>40|    Municipality municipality = municipalityRepository.findByPk(municipalityPk);
41|    County county = municipality.getCounty();
42|    Country country = county.getCountry();
43|    ElectionEvent electionEvent = country.getElectionEvent();

```

#### Sink :

File admin-configuration/src/main/java/no/evote/service/configuration/MvAreaServiceBean.java 40

Action Assign municipality = findByPk(...): MunicipalityRepository.findByPk may return NULL Trace

id 6

```

39|public MvArea findByMunicipality(Long municipalityPk) {
>40|    Municipality municipality = municipalityRepository.findByPk(municipalityPk);
41|    County county = municipality.getCounty();

```

```
42| Country country = county.getCountry();
43| ElectionEvent electionEvent = country.getElectionEvent();
```

**Issue C435624A8FE66E534085993FE1132D06:****Source :**

**File** admin-configuration/src/main/java/no/evote/service/configuration/MvAreaServiceBean.java 28 **Action** Assign pollingDistrict = findByPk(...):  
PollingDistrictRepository.findByPk may return NULL

**Trace id** 7

```
25|private MunicipalityRepository municipalityRepository;
26|
27|public MvArea findByPollingDistrict(Long pollingDistrictPk) {
>28|   PollingDistrict pollingDistrict = pollingDistrictRepository.findByPk(pollingDistrictPk);
29|   Borough borough = pollingDistrict.getBorough();
30|   Municipality municipality = borough.getMunicipality();
31|   County county = municipality.getCounty();
```

**Sink :**

**File** admin-configuration/src/main/java/no/evote/service/configuration/MvAreaServiceBean.java 28

**Action** Assign pollingDistrict = findByPk(...): PollingDistrictRepository.findByPk may return NULL **Trace id** 7

```
25|private MunicipalityRepository municipalityRepository;
26|
27|public MvArea findByPollingDistrict(Long pollingDistrictPk) {
>28|   PollingDistrict pollingDistrict = pollingDistrictRepository.findByPk(pollingDistrictPk);
29|   Borough borough = pollingDistrict.getBorough();
30|   Municipality municipality = borough.getMunicipality();
31|   County county = municipality.getCounty();
```

**6.66 MvElectionServiceBean.java****Issue C55A08414B1AABC574E83A9E4F4B7355:****Source :**

**File** admin-configuration/src/main/java/no/evote/service/configuration/MvElectionServiceBean.java 27

**Action** Assign mvElection = findByPk(...): MvElectionRepository.findByPk may return NULL **Trace id** 58

```
24|private MvElectionRepository mvElectionRepository;
25|
26|public boolean hasElectionsWithElectionTypeMinimal(MvElectionMinimal mvElectionMinimal, ElectionType electionType)
{
>27|   MvElection mvElection = mvElectionRepository.findByPk(mvElectionMinimal.getPk());
28|   return mvElectionRepository.hasElectionsWithElectionType(mvElection, electionType);
```

**Sink :**

**File** admin-configuration/src/main/java/no/evote/service/configuration/MvElectionServiceBean.java 27

**Action** Assign mvElection = findByPk(...): MvElectionRepository.findByPk may return NULL **Trace id** 58

```
24|private MvElectionRepository mvElectionRepository;
25|
26|public boolean hasElectionsWithElectionTypeMinimal(MvElectionMinimal mvElectionMinimal, ElectionType electionType)
{
>27|   MvElection mvElection = mvElectionRepository.findByPk(mvElectionMinimal.getPk());
28|   return mvElectionRepository.hasElectionsWithElectionType(mvElection, electionType);
```

**6.67 PapirmanntallDomainService.java****Issue BE26D6BD1FDB7A6E34B299553E339209:****Source :**

**File** admin-configuration/src/main/java/no/valg/eva/admin/configuration/domain/service/PapirmanntallDomainService.java 97 **Action** Assign findByPk(...):  
ElectionEventRepository.findByPk may return NULL

**Trace id** 3

```
96|private void oppdaterSideOgLinje(PollingDistrict stemmekrets, Long electionEventPk, List<Voter> velgereIKrets) {
> 97|   int maxAntallVelgerePerSide =
electionEventRepository.findByPk(electionEventPk).getElectoralRollLinesPerPage();
98|   SideOgLinjefordelerForRoder assigner = new SideOgLinjefordelerForRoder(velgereIKrets,
maxAntallVelgerePerSide);
99|   assigner.tilordneSideOgLinjeTilVelgere();
100|   voterRepository.updateLastLineLastPageNumber(stemmeKrets, assigner.getNesteLinje(), assigner.getNesteSide());
```

**Sink :**

**File** admin-configuration/src/main/java/no/valg/eva/admin/configuration/domain/service/PapirmanntallDomainService.java 97 **Action** Assign findByPk(...):  
ElectionEventRepository.findByPk may return NULL

**Trace id** 3

```
96|private void oppdaterSideOgLinje(PollingDistrict stemmekrets, Long electionEventPk, List<Voter> velgereIKrets) {
> 97|   int maxAntallVelgerePerSide =
electionEventRepository.findByPk(electionEventPk).getElectoralRollLinesPerPage();
```

```

98| SideOgLinjefordelerForRoder assigner = new SideOgLinjefordelerForRoder(velgereIKrets,
maxAntallVelgerePerSide);
99| assigner.tilordneSideOgLinjeTilVelgere();
100| voterRepository.updateLastLineLastPageNumber(stemmeKrets, assigner.getNesteLinje(), assigner.getNesteSide());

```

## 6.68 PartyRepository.java

### Issue 081D29594F82218044A310C3CB844369:

#### Source :

File admin-configuration/src/main/java/no/valg/eva/admin/configuration/repository/party/PartyRepository.java 36

Action Assign party = findEntityByPk(...) : BaseRepository.findEntityByPk may return NULL Trace id 1

```

35|public Party findDetachedByPk(Long pk) {
>36| Party party = findEntityByPk(Party.class, pk);
37| // trigg lasting før detach
38| party.getPartyContestAreas().size();
39| detach(Collections.singletonList(party));

```

#### Sink :

File admin-configuration/src/main/java/no/valg/eva/admin/configuration/repository/party/PartyRepository.java 36

Action Assign party = findEntityByPk(...) : BaseRepository.findEntityByPk may return NULL Trace id 1

```

35|public Party findDetachedByPk(Long pk) {
>36| Party party = findEntityByPk(Party.class, pk);
37| // trigg lasting før detach
38| party.getPartyContestAreas().size();
39| detach(Collections.singletonList(party));

```

## 6.69 PartyServiceBean.java

### Issue 2350058196DAC7F07AB9B103D85978D8:

#### Source :

File admin-backend/src/main/java/no/evote/service/configuration/PartyServiceBean.java 110

Action BranchTaken Branch taken: (parti.getPartyPk() != null)

```

108|private Party asParty(Parti parti, ElectionEvent electionEvent, boolean detach) {
109| Party party;
>110| if (parti.getPartyPk() != null) {
111| party = findParty(parti, detach);
112| partyMapper.updateParty(party, parti);
113| } else {

```

#### Sink :

File admin-backend/src/main/java/no/evote/service/configuration/PartyServiceBean.java 111

Action Assign party = findParty(...) : PartyServiceBean.findParty may return NULL

Trace id 13

```

108|private Party asParty(Parti parti, ElectionEvent electionEvent, boolean detach) {
109| Party party;
110| if (parti.getPartyPk() != null) {
>111| party = findParty(parti, detach);
112| partyMapper.updateParty(party, parti);
113| } else {
114| party = partyMapper.toParty(parti, electionEvent);

```

### Issue 324A0F60708F1A93EF6B1349E64FC9BC:

#### Source :

File admin-backend/src/main/java/no/evote/service/configuration/PartyServiceBean.java 63

Action BranchTaken Branch taken: (party.getPk() != null)

```

60|public Party update(UserData userData, Parti parti) {
61| Party party = asParty(parti, userData.electionEvent(), false);
62| party.setId(party.getId().toUpperCase());
>63| if (party.getPk() != null) {
64| Party dbParty = partyRepository.findByPk(party.getPk());
65| if (dbParty.getId().equals(party.getId())) {
66| String textIdName = party.getName();

```

#### Sink :

File admin-backend/src/main/java/no/evote/service/configuration/PartyServiceBean.java 64

Action Assign dbParty = findByPk(...) : PartyRepository.findByPk may return NULL Trace id

14

```

61|Party party = asParty(parti, userData.electionEvent(), false);
62|party.setId(party.getId().toUpperCase());

```

```

63|if (party.getPk() != null) {
>64|    Party dbParty = partyRepository.findByPk(party.getPk());
65|    if (dbParty.getId().equals(party.getId())) {
66|        String textIdName = party.getName();
67|        localeTextService.updateLocaleTextToAllLocale(userData, textIdName, party.getTranslatedPartyName(),
dbParty.getElectionEvent());

```

**Issue 97C67BBD22336B0802332F74495C23A9:****Source :****File** admin-backend/src/main/java/no/evote/service/configuration/PartyServiceBean.java 61**Action** Assign party = asParty(...): PartyServiceBean.asParty may return NULL **Trace****id** 43

```

60|public Party update(UserData userData, Parti parti) {
>61|    Party party = asParty(parti, userData.electionEvent(), false);
62|    party.setId(party.getId().toUpperCase());
63|    if (party.getPk() != null) {
64|        Party dbParty = partyRepository.findByPk(party.getPk());

```

**Sink :****File** admin-backend/src/main/java/no/evote/service/configuration/PartyServiceBean.java 61**Action** Assign party = asParty(...): PartyServiceBean.asParty may return NULL **Trace****id** 43

```

60|public Party update(UserData userData, Parti parti) {
>61|    Party party = asParty(parti, userData.electionEvent(), false);
62|    party.setId(party.getId().toUpperCase());
63|    if (party.getPk() != null) {
64|        Party dbParty = partyRepository.findByPk(party.getPk());

```

**6.70 PollingDistrictApplicationService.java****Issue A602A1CD01627E6ACB3A6746251A24DE:****Source :****File** admin-backend/src/main/java/no/valg/eva/admin/configuration/application/PollingDistrictApplicationService.java 173**Action** Assign dbDistrict = findByPk(...): PollingDistrictRepository.findByPk may return NULL **Trace id** 7

```

170|@Security(accesses = Konfigurasjon_Grunnlagsdata_Redigere, type = WRITE)
171|@AuditLog(eventClass = ParentPollingDistrictAuditEvent.class, eventType = Delete)
172|public void deleteParentPollingDistrict(UserData userData, ParentPollingDistrict district) {
>173|    PollingDistrict dbDistrict = pollingDistrictRepository.findByPk(district.getPk()); 174|
dbDistrict.checkVersion(district);
175|    pollingDistrictRepository.deleteParentPollingDistrict(userData, dbDistrict);

```

**Sink :****File** admin-backend/src/main/java/no/valg/eva/admin/configuration/application/PollingDistrictApplicationService.java 173**Action** Assign dbDistrict = findByPk(...): PollingDistrictRepository.findByPk may return NULL **Trace id** 7

```

170|@Security(accesses = Konfigurasjon_Grunnlagsdata_Redigere, type = WRITE)
171|@AuditLog(eventClass = ParentPollingDistrictAuditEvent.class, eventType = Delete)
172|public void deleteParentPollingDistrict(UserData userData, ParentPollingDistrict district) {
>173|    PollingDistrict dbDistrict = pollingDistrictRepository.findByPk(district.getPk());
174|    dbDistrict.checkVersion(district);
175|    pollingDistrictRepository.deleteParentPollingDistrict(userData, dbDistrict);

```

**Issue C2E9825206A4C5F307CA35C37C7F26D8:****Source :****File** admin-backend/src/main/java/no/valg/eva/admin/configuration/application/PollingDistrictApplicationService.java 121**Action** BranchNotTaken Branch not taken: (district.getPk() != null)

```

118|@AuditLog(eventClass = TechnicalPollingDistrictAuditEvent.class, eventType = Save)
119|public TechnicalPollingDistrict saveTechnicalPollingDistrict(UserData userData, TechnicalPollingDistrict
district) {
120|    PollingDistrict pollingDistrict;
>121|    if (district.getPk() == null) {
122|        district.getPath().assertLevel(AreaLevelEnum.MUNICIPALITY);
123|        Municipality municipality = getMunicipality(userData, district.getPath());
124|        pollingDistrict = PollingDistrictMapper.toPollingDistrict(new PollingDistrict(), district);

```

**Sink :****File** admin-backend/src/main/java/no/valg/eva/admin/configuration/application/PollingDistrictApplicationService.java 129 **Action** Assign pollingDistrict = findByPk(...): PollingDistrictRepository.findByPk may return NULL**Trace id** 7

```

126|    pollingDistrict = pollingDistrictRepository.create(userData, pollingDistrict); 127|} else {
128|    district.getPath().assertLevel(AreaLevelEnum.POLLING_DISTRICT);
>129|    pollingDistrict = pollingDistrictRepository.findByPk(district.getPk());
130|    pollingDistrict.checkVersion(district);

```

```
131|    pollingDistrict = PollingDistrictMapper.toPollingDistrict(pollingDistrict, district);
132|    pollingDistrict = pollingDistrictRepository.update(userData, pollingDistrict);
```

**Issue E16C51D98F3B87D154C70EE081001B7B:****Source :**

**File** admin-backend/src/main/java/no/valg/eva/admin/configuration/application/PollingDistrictApplicationService.java 164

**Action** Assign dbDistrict = findByPk(...): PollingDistrictRepository.findByPk may return NULL **Trace id** 7

```
161|@Security(accesses = Konfigurasjon_Grunnlagsdata_Redigere, type = WRITE)
162|@AuditLog(eventClass = TechnicalPollingDistrictAuditEvent.class, eventType = Delete)
163|public void deleteTechnicalPollingDistrict(UserData userData, TechnicalPollingDistrict district) {
>164|    PollingDistrict dbDistrict = pollingDistrictRepository.findByPk(district.getPk());
165|    dbDistrict.checkVersion(district);
166|    pollingDistrictRepository.delete(userData, dbDistrict.getPk());
```

**Sink :**

**File** admin-backend/src/main/java/no/valg/eva/admin/configuration/application/PollingDistrictApplicationService.java 164

**Action** Assign dbDistrict = findByPk(...): PollingDistrictRepository.findByPk may return NULL **Trace id** 7

```
161|@Security(accesses = Konfigurasjon_Grunnlagsdata_Redigere, type = WRITE)
162|@AuditLog(eventClass = TechnicalPollingDistrictAuditEvent.class, eventType = Delete)
163|public void deleteTechnicalPollingDistrict(UserData userData, TechnicalPollingDistrict district) {
>164|    PollingDistrict dbDistrict = pollingDistrictRepository.findByPk(district.getPk());
165|    dbDistrict.checkVersion(district);
166|    pollingDistrictRepository.delete(userData, dbDistrict.getPk());
```

**Issue EEFD677BCE12BB90A63BF2028977C5BA:****Source :**

**File** admin-backend/src/main/java/no/valg/eva/admin/configuration/application/PollingDistrictApplicationService.java 142

**Action** BranchTaken Branch taken: (district.getPk() == null)

```
139|@AuditLog(eventClass = ParentPollingDistrictAuditEvent.class, eventType = Save)
140|public ParentPollingDistrict saveParentPollingDistrict(UserData userData, ParentPollingDistrict district) {
141|    PollingDistrict pollingDistrict;
>142|    if (district.getPk() == null) {
143|        district.getPath().assertLevel(AreaLevelEnum.MUNICIPALITY);
144|        List<PollingDistrict> children = new ArrayList<>();
145|        for (RegularPollingDistrict child : district.getChildren()) {
```

**Sink :**

**File** admin-backend/src/main/java/no/valg/eva/admin/configuration/application/PollingDistrictApplicationService.java 146

**Action** Assign dbChild = findByPk(...): PollingDistrictRepository.findByPk may return NULL **Trace id** 7

```
143|district.getPath().assertLevel(AreaLevelEnum.MUNICIPALITY);
144|List<PollingDistrict> children = new ArrayList<>();
145|for (RegularPollingDistrict child : district.getChildren()) {
>146|    PollingDistrict dbChild = pollingDistrictRepository.findByPk(child.getPk()); 147|
dbChild.checkVersion(child);
148|    children.add(dbChild);
```

**6.71 PollingPlaceApplicationService.java****Issue C920BA2DB602476D224FA9D93B60C3FD:****Source :**

**File** admin-configuration/src/main/java/no/valg/eva/admin/configuration/application/PollingPlaceApplicationService.java 89

**Action** Assign pollingPlace = findByPk(...): PollingPlaceRepository.findByPk may return NULL **Trace id** 10

```
86|@Security(accesses = Konfigurasjon_Grunnlagsdata_Redigere, type = WRITE)
87|@AuditLog(eventClass = AdvancePollingPlaceAuditEvent.class, eventType = Delete)
88|public void deleteAdvancePollingPlace(UserData userData, AdvancePollingPlace advancePollingPlace) {
>89|    PollingPlace pollingPlace = pollingPlaceRepository.findByPk(advancePollingPlace.getPk());
90|    pollingPlace.checkVersion(advancePollingPlace);
91|    pollingPlaceRepository.delete(userData, pollingPlace.getPk());
```

**Sink :**

**File** admin-configuration/src/main/java/no/valg/eva/admin/configuration/application/PollingPlaceApplicationService.java 89

**Action** Assign pollingPlace = findByPk(...): PollingPlaceRepository.findByPk may return NULL **Trace id** 10

```
86|@Security(accesses = Konfigurasjon_Grunnlagsdata_Redigere, type = WRITE)
87|@AuditLog(eventClass = AdvancePollingPlaceAuditEvent.class, eventType = Delete)
88|public void deleteAdvancePollingPlace(UserData userData, AdvancePollingPlace advancePollingPlace) {
>89|    PollingPlace pollingPlace = pollingPlaceRepository.findByPk(advancePollingPlace.getPk());
90|    pollingPlace.checkVersion(advancePollingPlace);
91|    pollingPlaceRepository.delete(userData, pollingPlace.getPk());
```

**Issue CA9F3EB18F9196C5886D21EEEDD60A3E:****Source :**

**File** admin-configuration/src/main/java/no/valg/eva/admin/configuration/application/PollingPlaceApplicationService.java 61

**Action** BranchNotTaken Branch not taken: (advancePollingPlace.getPk() != null)



```

58|public AdvancePollingPlace saveAdvancePollingPlace(UserData userData, ElectionPath electionGroupPath,
AdvancePollingPlace advancePollingPlace) {
59|    PollingPlace pollingPlace;
60|
>61|    if (advancePollingPlace.getPk() == null) {
62|        advancePollingPlace.getPath().assertLevel(AreaLevelEnum.MUNICIPALITY);
63|        MvElection mvElection = mvElectionRepository.finnEnkeltMedSti(electionGroupPath.tilValghierarkiSti());
64|        pollingPlace = PollingPlaceMapper.toPollingPlace(new PollingPlace(), advancePollingPlace);

```

**Sink :****File** admin-configuration/src/main/java/no/valg/eva/admin/configuration/application/PollingPlaceApplicationService.java 70**Action** Assign pollingPlace = findByPk(...): PollingPlaceRepository.findByPk may return NULL **Trace id** 10

```

67|    pollingPlace = pollingPlaceRepository.create(userData, pollingPlace);
68|} else {
69|    advancePollingPlace.getPath().assertLevel(AreaLevelEnum.POLLING_PLACE);
>70|    pollingPlace = pollingPlaceRepository.findByPk(advancePollingPlace.getPk());
71|    pollingPlace.checkVersion(advancePollingPlace); 72|    pollingPlace =
PollingPlaceMapper.toPollingPlace(pollingPlace, advancePollingPlace);
73|    pollingPlace = pollingPlaceRepository.update(userData, pollingPlace);

```

## 6.72 PollingPlaceDomainService.java

### Issue 17A7CA8D499481F2A71D146FFF1B1D40:

**Source :****File** admin-configuration/src/main/java/no/valg/eva/admin/configuration/domain/service/PollingPlaceDomainService.java 98 **Action**

BranchNotTaken Branch not taken: (electionDayPollingPlace.getPk() != null)

```

96|PollingDistrict pollingDistrict;
97|PollingPlace pollingPlace;
> 98|if (electionDayPollingPlace.getPk() == null) {
99|    electionDayPollingPlace.getPath().assertLevel(POLLING_DISTRICT);
100|    pollingDistrict = pollingDistrictRepository.findByPk(electionDayPollingPlace.getParentPk()); 101|
pollingPlace = toPollingPlace(new PollingPlace(), electionDayPollingPlace);

```

**Sink :****File** admin-configuration/src/main/java/no/valg/eva/admin/configuration/domain/service/PollingPlaceDomainService.java 106**Action** Assign pollingPlace = findByPk(...): PollingPlaceRepository.findByPk may return NULL **Trace id** 10

```

103|    pollingPlace = create(userData, pollingPlace);
104|} else {
105|    electionDayPollingPlace.getPath().assertLevel(POLLING_PLACE);
>106|    pollingPlace = pollingPlaceRepository.findByPk(electionDayPollingPlace.getPk());
107|    pollingPlace.checkVersion(electionDayPollingPlace);
108|    pollingDistrict = pollingPlace.getPollingDistrict();
109|    pollingPlace = toPollingPlace(pollingPlace, electionDayPollingPlace);

```

### Issue 2BAA4A7B156EAF28500A0E3D5B198F3C:

**Source :****File** admin-configuration/src/main/java/no/valg/eva/admin/configuration/domain/service/PollingPlaceDomainService.java 98 **Action**

BranchTaken Branch taken: (electionDayPollingPlace.getPk() == null)

```

96|PollingDistrict pollingDistrict;
97|PollingPlace pollingPlace;
> 98|if (electionDayPollingPlace.getPk() == null) {
99|    electionDayPollingPlace.getPath().assertLevel(POLLING_DISTRICT);
100|    pollingDistrict = pollingDistrictRepository.findByPk(electionDayPollingPlace.getParentPk()); 101|
pollingPlace = toPollingPlace(new PollingPlace(), electionDayPollingPlace);

```

**Sink :****File** admin-configuration/src/main/java/no/valg/eva/admin/configuration/domain/service/PollingPlaceDomainService.java 100**Action** Assign pollingDistrict = findByPk(...): PollingDistrictRepository.findByPk may return NULL **Trace id** 7

```

97|PollingPlace pollingPlace;
98|if (electionDayPollingPlace.getPk() == null) {
99|    electionDayPollingPlace.getPath().assertLevel(POLLING_DISTRICT);
>100|    pollingDistrict = pollingDistrictRepository.findByPk(electionDayPollingPlace.getParentPk());
101|    pollingPlace = toPollingPlace(new PollingPlace(), electionDayPollingPlace);
102|    pollingPlace.setPollingDistrict(pollingDistrict);
103|    pollingPlace = create(userData, pollingPlace);

```

### Issue 6EoCDB6965B5CBB277AC1E83171EE7Fo:

**Source :****File** admin-configuration/src/main/java/no/valg/eva/admin/configuration/domain/service/PollingPlaceDomainService.java 202 **Action** Assign dbMunicipality = findByPk(...): MunicipalityRepository.findByPk may return NULL**Trace id** 6

```

201|private List<MunicipalityOpeningHour> municipalityOpeningHours(Municipality municipality) {
>202|    Municipality dbMunicipality = municipalityRepository.findByPk(municipality.getPk());
203|    return OpeningHoursSorter.toSortedMunicipalityOpeningHourList(dbMunicipality.getOpeningHours());

```

**Sink :**

**File** admin-configuration/src/main/java/no/valg/eva/admin/configuration/domain/service/PollingPlaceDomainService.java 202 **Action** Assign dbMunicipality = findByPk(...): MunicipalityRepository.findByPk may return NULL

**Trace id** 6

```
201|private List<MunicipalityOpeningHour> municipalityOpeningHours(Municipality municipality) {
>202|    Municipality dbMunicipality = municipalityRepository.findByPk(municipality.getPk());
203|    return OpeningHoursSorter.toSortedMunicipalityOpeningHourList(dbMunicipality.getOpeningHours());
```

## 6.73 PollingStationServiceBean.java

**Issue** 472DDE0B791B52320824E4C2D73EEFD5:**Source :**

**File** admin-backend/src/main/java/no/evote/service/configuration/PollingStationServiceBean.java 95

**Action** BranchNotTaken Branch not taken: validator.isValid(divisionList)

```
92|PollingStationsDivisionValidator validator = new PollingStationsDivisionValidator(); 93|
94|Rodefordeler.sorterRoder(divisionList);
>95|if (!validator.isValid(divisionList)) {
96|    throw new IllegalArgumentException("The polling station division is not valid");
```

**Sink :**

**File** admin-backend/src/main/java/no/evote/service/configuration/PollingStationServiceBean.java 99

**Action** Assign pollingPlace = findByPk(...): PollingPlaceRepository.findByPk may return NULL

**Trace id** 10

```
96|    throw new IllegalArgumentException("The polling station division is not valid");
97|}
98|
> 99|PollingPlace pollingPlace = pollingPlaceRepository.findByPk(detachedPollingPlace.getPk()); 100|
101|// Remove the old polling stations
102|pollingStationRepository.delete(userData, pollingStationRepository.findByPollingPlace(pollingPlace.getPk()));
```

## 6.74 RapportDomainService.java

**Issue** 21D7525760DEA9CA08E330EE8C338063:**Source :**

**File** admin-rapport/src/main/java/no/valg/eva/admin/rapport/domain/service/RapportDomainService.java 123

**Action** BranchNotTaken Branch not taken: (municipality != null)

```
121|boolean isAvkryssingsMantallRapportDisabled(UserData userData) {
122|    Municipality municipality = userData.getOperatorMvArea().getMunicipality();
>123|    if (municipality == null || !municipalityRepository.findByPk(municipality.getPk()).isElectronicMarkoffs()) {
124|        // User needs to be on at least municipality level and XiM
125|        return false;
```

**Sink :**

**File** admin-rapport/src/main/java/no/valg/eva/admin/rapport/domain/service/RapportDomainService.java 123

**Action** Assign findByPk(...): MunicipalityRepository.findByPk may return NULL **Trace id** 6

```
121|boolean isAvkryssingsMantallRapportDisabled(UserData userData) {
122|    Municipality municipality = userData.getOperatorMvArea().getMunicipality();
>123|    if (municipality == null || !municipalityRepository.findByPk(municipality.getPk()).isElectronicMarkoffs()) {
124|        // User needs to be on at least municipality level and XiM
125|        return false;
```

**Issue** 8EECFD27C69780658D04C7D96A81D36F:**Source :**

**File** admin-rapport/src/main/java/no/valg/eva/admin/rapport/domain/service/RapportDomainService.java 123

**Action** BranchNotTaken Branch not taken: (municipality != null)

```
121|boolean isAvkryssingsMantallRapportDisabled(UserData userData) {
122|    Municipality municipality = userData.getOperatorMvArea().getMunicipality();
>123|    if (municipality == null || !municipalityRepository.findByPk(municipality.getPk()).isElectronicMarkoffs()) {
124|        // User needs to be on at least municipality level and XiM
125|        return false;
```

**Sink :**

**File** admin-rapport/src/main/java/no/valg/eva/admin/rapport/domain/service/RapportDomainService.java 131

**Action** Assign electionEvent = findByPk(...): ElectionEventRepository.findByPk may return NULL

**Trace id** 3

```
128|for (int i = COUNTY.getLevel(); i <= userData.getOperatorAreaLevel().getLevel(); i++) {
129|    areaPaths.add(userData.getOperatorAreaPath().toAreaLevelPath(AreaLevelEnum.getLevel(i))); 130|}
>131|ElectionEvent electionEvent = electionEventRepository.findByPk(userData.getElectionEventPk());
132|List<MvElection> contests = mvElectionRepository.findContestsByElectionEventAndAreas(electionEvent, areaPaths);
133|// All contests must be settled
```

```
134|for (MvElection contest : contests) {
```

## 6.75 ReduserGeografiServlet.java

### Issue 9A1B5DEE6A9E49EA9B411B91B0EFDA51:

**Source :**

**File** admin-backend/src/main/java/no/valg/eva/admin/backend/web/ReduserGeografiServlet.java 113

**Action** Assign getParameter(...) : ServletRequest.getParameter may return NULL

**Rule id** 4280F38B-9FDB-454E-B495-89CF45CD51B7

```
112|private GeografiReduksjonsmodus finnReduksjonsmodus(HttpServletRequest req) { >113|
switch (req.getParameter(PARAMETER_REDUKSJONSMODUS)) {
114|     case VALUE_SLETT:
115|         return GeografiReduksjonsmodus.SLETT; 116|     case VALUE_FLYTT:
```

**Sink :**

**File** admin-backend/src/main/java/no/valg/eva/admin/backend/web/ReduserGeografiServlet.java 113

**Action** Assign getParameter(...) : ServletRequest.getParameter may return NULL

**Rule id** 4280F38B-9FDB-454E-B495-89CF45CD51B7

```
112|private GeografiReduksjonsmodus finnReduksjonsmodus(HttpServletRequest req) { >113|
switch (req.getParameter(PARAMETER_REDUKSJONSMODUS)) {
114|     case VALUE_SLETT:
115|         return GeografiReduksjonsmodus.SLETT; 116|     case VALUE_FLYTT:
```

## 6.76 ReferenceDataMigration.java

### Issue 85638AECA14F214C9ABAF4C69500C159:

**Source :**

**File** admin-database/src/main/java/no/valg/eva/admin/database/migrations/ReferenceDataMigration.java 68

**Action** Assign classLoader = getClassLoader() : Class.getClassLoader may return NULL

**Rule id** 4280F38B-9FDB-454E-B495-89CF45CD51B7

```
67|private List<URL> mapMigrationElementsToUrls(List<MigrationElement> migrationElements) {
>68|     ClassLoader classLoader = getClass().getClassLoader();
69|     List<URL> classpathUrls = new ArrayList<>(migrationElements.size());
70|
71|     for (MigrationElement migrationElement : migrationElements) {
```

**Sink :**

**File** admin-database/src/main/java/no/valg/eva/admin/database/migrations/ReferenceDataMigration.java 68

**Action** Assign classLoader = getClassLoader() : Class.getClassLoader may return NULL

**Rule id** 4280F38B-9FDB-454E-B495-89CF45CD51B7

```
67|private List<URL> mapMigrationElementsToUrls(List<MigrationElement> migrationElements) {
>68|     ClassLoader classLoader = getClass().getClassLoader();
69|     List<URL> classpathUrls = new ArrayList<>(migrationElements.size());
70|
71|     for (MigrationElement migrationElement : migrationElements) {
```

## 6.77 ReportCountCategoryServiceBean.java

### Issue 8330400BB9D50DD20FF35612CAB4389E:

**Source :**

**File** admin-configuration/src/main/java/no/evote/service/configuration/ReportCountCategoryServiceBean.java 43 **Action** Assign domainMunicipality = findByPk(...) : MunicipalityRepository.findByPk may return NULL

**Trace id** 6

```
41|public List<ReportCountCategory> findReportCountCategoryElementByArea(Municipality municipality, ElectionGroup
electionGroup) { 42|
>43|     Municipality domainMunicipality = municipalityRepository.findByPk(municipality.getPk());
44|     List<ReportCountCategory> reportCountCategoryListFromDb =
reportCountCategoryRepository.findReportCountCategories(domainMunicipality, electionGroup);
45|     List<ElectionVoteCountCategory> centralCats =
electionVoteCountCategoryRepository.findElectionVoteCountCategories(electionGroup, BF);
```

**Sink :**

**File** admin-configuration/src/main/java/no/evote/service/configuration/ReportCountCategoryServiceBean.java 43 **Action** Assign domainMunicipality = findByPk(...) : MunicipalityRepository.findByPk may return NULL

**Trace id** 6

```
41|public List<ReportCountCategory> findReportCountCategoryElementByArea(Municipality municipality, ElectionGroup
electionGroup) { 42|
>43|     Municipality domainMunicipality = municipalityRepository.findByPk(municipality.getPk());
44|     List<ReportCountCategory> reportCountCategoryListFromDb =
```

```
reportCountCategoryRepository.findReportCountCategories(domainMunicipality, electionGroup);
45|     List<ElectionVoteCountCategory> centralCats =
electionVoteCountCategoryRepository.findElectionVoteCountCategories(electionGroup, BF);
```

## 6.78 ReportingUnitApplicationService.java

### Issue 23DA50159C152B522F54D1BDEE63304B:

**Source :**

**File** admin-backend/src/main/java/no/valg/eva/admin/configuration/application/ReportingUnitApplicationService.java 124 **Action** Assign dbOfficer = findByPk(...): ResponsibleOfficerRepository.findByPk may return NULL

**Trace id** 16

```
121|@Security(accesses = {Konfigurasjon_Grunnlagsdata_Redigere, Konfigurasjon_Opptellingsvalgstyrer}, type = WRITE)
122|@AuditLog(eventClass = ResponsibleOfficerAuditEvent.class, eventType = AuditEventTypes.Delete)
123|public void delete(UserData userData, ResponsibleOfficer responsibleOfficer) {
>124|     no.valg.eva.admin.configuration.domain.model.ResponsibleOfficer dbOfficer =
responsibleOfficerRepository.findByPk(responsibleOfficer.getPk());
125|     dbOfficer.checkVersion(responsibleOfficer);
126|     responsibleOfficerService.delete(userData, dbOfficer);
```

**Sink :**

**File** admin-backend/src/main/java/no/valg/eva/admin/configuration/application/ReportingUnitApplicationService.java 124

**Action** Assign dbOfficer = findByPk(...): ResponsibleOfficerRepository.findByPk may return NULL **Trace id** 16

```
121|@Security(accesses = {Konfigurasjon_Grunnlagsdata_Redigere, Konfigurasjon_Opptellingsvalgstyrer}, type = WRITE)
122|@AuditLog(eventClass = ResponsibleOfficerAuditEvent.class, eventType = AuditEventTypes.Delete)
123|public void delete(UserData userData, ResponsibleOfficer responsibleOfficer) {
>124|     no.valg.eva.admin.configuration.domain.model.ResponsibleOfficer dbOfficer =
responsibleOfficerRepository.findByPk(responsibleOfficer.getPk());
125|     dbOfficer.checkVersion(responsibleOfficer);
126|     responsibleOfficerService.delete(userData, dbOfficer);
```

### Issue 2A774A217775BC7C7365401C867C8266:

**Source :**

**File** admin-backend/src/main/java/no/valg/eva/admin/configuration/application/ReportingUnitApplicationService.java 158

**Action** BranchNotTaken Branch not taken: (responsibleOfficer.getPk() != null)

```
156|private no.valg.eva.admin.configuration.domain.model.ResponsibleOfficer getDBResponsibleOfficer(UserData
userData, ResponsibleOfficer responsibleOfficer) {
157|     no.valg.eva.admin.configuration.domain.model.ResponsibleOfficer dbOfficer;
>158|     if (responsibleOfficer.getPk() == null) {
159|         dbOfficer = new no.valg.eva.admin.configuration.domain.model.ResponsibleOfficer();
160|         dbOfficer.setReportingUnit(reportingUnitDomainService.getReportingUnit(userData,
responsibleOfficer.getAreaPath()));
161|     } else {
```

**Sink :**

**File** admin-backend/src/main/java/no/valg/eva/admin/configuration/application/ReportingUnitApplicationService.java 162 **Action** Assign dbOfficer = findByPk(...): ResponsibleOfficerRepository.findByPk may return NULL

**Trace id** 16

```
159|     dbOfficer = new no.valg.eva.admin.configuration.domain.model.ResponsibleOfficer(); 160|
dbOfficer.setReportingUnit(reportingUnitDomainService.getReportingUnit(userData, responsibleOfficer.getAreaPath()));
161|} else {
>162|     dbOfficer = responsibleOfficerRepository.findByPk(responsibleOfficer.getPk()); 163|
164|     //Detaching to make sure we are not updating the object
165|     responsibleOfficerRepository.detach(dbOfficer);
```

### Issue 63B8090E6E7B25522288C880E5829CC8:

**Source :**

**File** admin-backend/src/main/java/no/valg/eva/admin/configuration/application/ReportingUnitApplicationService.java 94 **Action**

BranchTaken Branch taken: displayOrder~iterator.hasNext()

```
91|@Override
92|@Security(accesses = {Konfigurasjon_Grunnlagsdata_Redigere, Konfigurasjon_Opptellingsvalgstyrer}, type = WRITE)
93|public List<ResponsibleOfficer> saveResponsibleOfficerDisplayOrder(UserData userData, AreaPath areaPath,
List<DisplayOrder> displayOrders) {
>94|     for (DisplayOrder displayOrder : displayOrders) {
95|         no.valg.eva.admin.configuration.domain.model.ResponsibleOfficer officer =
responsibleOfficerRepository.findByPk(displayOrder.getPk());
96|         officer.checkVersion(displayOrder);
97|         officer.setDisplayOrder(displayOrder.getDisplayOrder());
```

**Sink :**

**File** admin-backend/src/main/java/no/valg/eva/admin/configuration/application/ReportingUnitApplicationService.java 95 **Action** Assign officer = findByPk(...): ResponsibleOfficerRepository.findByPk may return NULL

**Trace id** 16

```
92|@Security(accesses = {Konfigurasjon_Grunnlagsdata_Redigere, Konfigurasjon_Opptellingsvalgstyrer}, type = WRITE)
```

```

93|public List<ResponsibleOfficer> saveResponsibleOfficerDisplayOrder(UserData userData, AreaPath areaPath,
List<DisplayOrder> displayOrders) {
94|    for (DisplayOrder displayOrder : displayOrders) {
>95|        no.valg.eva.admin.configuration.domain.model.ResponsibleOfficer officer =
responsibleOfficerRepository.findByPk(displayOrder.getPk());
96|        officer.checkVersion(displayOrder);
97|        officer.setDisplayOrder(displayOrder.getDisplayOrder()); 98|
responsibleOfficerService.save(userData, officer);

```

## 6.79 ReportingUnitTypeServiceEjb.java

### Issue 8CAF7D877A6381157704C37F3A279FoD:

#### Source :

File admin-backend/src/main/java/no/evote/service/configuration/ReportingUnitTypeServiceEjb.java 70

Action Assign reportingUnitType = findReportingUnitTypeByPk(...): ReportingUnitRepository.findReportingUnitTypeByPk may return

NULL

Trace id 38

```

67|@AuditLog(eventClass = CompositeAuditEvent.class, eventType = AuditEventTypes.Update, objectSource =
AuditedObjectSource.Collecte
68|public void updateMvElectionReportingUnits(
69|    UserData userData, @SecureEntity(electionLevelDynamic = true) List<MvElection> mvElectionList, long
reportingUnitTypePk) {
>70|    ReportingUnitType reportingUnitType = reportingUnitRepository.findReportingUnitTypeByPk(reportingUnitTypePk);
71|    updateMvElectionReportingUnits(userData, mvElectionList, reportingUnitType);

```

#### Sink :

File admin-backend/src/main/java/no/evote/service/configuration/ReportingUnitTypeServiceEjb.java 70

Action Assign reportingUnitType = findReportingUnitTypeByPk(...): ReportingUnitRepository.findReportingUnitTypeByPk may return

NULL

Trace id 38

```

67|@AuditLog(eventClass = CompositeAuditEvent.class, eventType = AuditEventTypes.Update, objectSource =
AuditedObjectSource.Collecte
68|public void updateMvElectionReportingUnits(
69|    UserData userData, @SecureEntity(electionLevelDynamic = true) List<MvElection> mvElectionList, long
reportingUnitTypePk) {
>70|    ReportingUnitType reportingUnitType = reportingUnitRepository.findReportingUnitTypeByPk(reportingUnitTypePk);
71|    updateMvElectionReportingUnits(userData, mvElectionList, reportingUnitType);

```

## 6.80 ResourceNotFoundServlet.java

### Issue oBDC8AC834BE2CB661AB3BD6D54467A7:

#### Source :

File admin-frontend/src/main/java/no/evote/presentation/exceptions/ResourceNotFoundServlet.java 14

Action Assign getAttribute(...): ServletRequest.getAttribute may return NULL

Rule id 4280F38B-9FDB-454E-B495-89CF45CD51B7

```

12|    @Override
13|    public void doGet(final HttpServletRequest req, final HttpServletResponse res) {
>14|        String message = req.getAttribute("javax.servlet.error.message").toString();
15|        ErrorPageRenderer.renderError(req, res, ErrorPageRenderer.Error.NOT_FOUND, message, message);

```

#### Sink :

File admin-frontend/src/main/java/no/evote/presentation/exceptions/ResourceNotFoundServlet.java 14

Action Assign getAttribute(...): ServletRequest.getAttribute may return NULL

Rule id 4280F38B-9FDB-454E-B495-89CF45CD51B7

```

12|    @Override
13|    public void doGet(final HttpServletRequest req, final HttpServletResponse res) {
>14|        String message = req.getAttribute("javax.servlet.error.message").toString();
15|        ErrorPageRenderer.renderError(req, res, ErrorPageRenderer.Error.NOT_FOUND, message, message);

```

## 6.81 ResponsibilityValidationApplicationService.java

### Issue 43C3247F1BCD50E32FC16A16FD4FC8B8:

#### Source :

File admin-configuration/src/main/java/no/valg/eva/admin/configuration/application/ResponsibilityValidationApplicationService.java 63

Action Assign findElectionGroup(...): ResponsibilityValidationApplicationService.findElectionGroup may return NULL Trace id 22

```

60|@Security(accesses = Konfigurasjon_Grunnlagsdata_Redigere, type = READ)

```

```

61|public List<ResponsibilityConflict> checkIfBoardMemberHasCandidateConflict(UserData userData, String firstName,
String middleName, String lastName,
62|                                     AreaPath areaPath) {
>63|   if (findElectionGroup(userData).isValidPollingPlaceElectoralBoardAndListProposal()) {
64|       return responsibilityValidationDomainService.checkIfBoardMemberHasCandidateConflict(firstName, middleName,
lastName, areaPath);
65|   }
66|   return emptyList();

```

**Sink :****File** admin-configuration/src/main/java/no/valg/eva/admin/configuration/application/ResponsibilityValidationApplicationService.java 63**Action** Assign findElectionGroup(...): ResponsibilityValidationApplicationService.findElectionGroup may return NULL **Trace id** 22

```

60|@Security(accesses = Konfigurasjon_Grunnlagsdata_Redigere, type = READ)
61|public List<ResponsibilityConflict> checkIfBoardMemberHasCandidateConflict(UserData userData, String firstName,
String middleName, String lastName,
62|                                     AreaPath areaPath) {
>63|   if (findElectionGroup(userData).isValidPollingPlaceElectoralBoardAndListProposal()) {
64|       return responsibilityValidationDomainService.checkIfBoardMemberHasCandidateConflict(firstName, middleName,
lastName, areaPath);
65|   }
66|   return emptyList();

```

**Issue** FB64BFE68FC339716DA899BBB998AAFE:**Source :****File** admin-configuration/src/main/java/no/valg/eva/admin/configuration/application/ResponsibilityValidationApplicationService.java 38**Action** Assign findElectionGroup(...): ResponsibilityValidationApplicationService.findElectionGroup may return NULL **Trace id** 22

```

35|@Override
36|@Security(accesses = Tilgang_Brukere_Administrere, type = READ)
37|public List<ResponsibilityConflict> checkIfRoleHasCandidateConflict(UserData userData, PersonId personId, AreaPath
areaPath, String roleId) {
>38|   if (findElectionGroup(userData).isValidRoleAndListProposal()) {
39|       return responsibilityValidationDomainService.checkIfPersonHasCandidateConflict(personId, areaPath, roleId,
userData.electionEvent());
40|   }
41|   return emptyList();

```

**Sink :****File** admin-configuration/src/main/java/no/valg/eva/admin/configuration/application/ResponsibilityValidationApplicationService.java 38**Action** Assign findElectionGroup(...): ResponsibilityValidationApplicationService.findElectionGroup may return NULL **Trace id** 22

```

35|@Override
36|@Security(accesses = Tilgang_Brukere_Administrere, type = READ)
37|public List<ResponsibilityConflict> checkIfRoleHasCandidateConflict(UserData userData, PersonId personId, AreaPath
areaPath, String roleId) {
>38|   if (findElectionGroup(userData).isValidRoleAndListProposal()) {
39|       return responsibilityValidationDomainService.checkIfPersonHasCandidateConflict(personId, areaPath, roleId,
userData.electionEvent());
40|   }
41|   return emptyList();

```

## 6.82 RoleServiceBean.java

**Issue** 55C242112EBC822CA8157AEF4AC6EE8D:**Source :****File** admin-backend/src/main/java/no/evote/service/rbac/RoleServiceBean.java 173**Action** BranchTaken Branch taken: (rvo.getPk() != null)

```

171|public Role toRoleEntity(UserData userData, no.valg.eva.admin.common.rbac.Role rvo, Map<String, Access>
allAccesses,
172|   final Map<AreaLevel, AreaLevel> allAreaLevels) {
>173|   Role role = rvo.getPk() != null ? roleRepository.findByPk(rvo.getPk()) : new Role();
174|
175|   if (rvo.containsAccessesToMap()) {
176|       role.setAccesses(toAccessEntities(rvo.getAccesses(), allAccesses));

```

**Sink :****File** admin-backend/src/main/java/no/evote/service/rbac/RoleServiceBean.java 173**Action** Assign role = findByPk(...): RoleRepository.findByPk may return NULL **Trace id** 26

```

171|public Role toRoleEntity(UserData userData, no.valg.eva.admin.common.rbac.Role rvo, Map<String, Access>
allAccesses,
172|    final Map<AreaLevel, AreaLevel> allAreaLevels) {
>173|    Role role = rvo.getPk() != null ? roleRepository.findByPk(rvo.getPk()) : new Role();
174|
175|    if (rvo.containsAccessesToMap()) {
176|        role.setAccesses(toAccessEntities(rvo.getAccesses(), allAccesses));

```

**Issue 65C1D60C7EAF34B3D3C912558A2CE820:****Source :**

**File** admin-backend/src/main/java/no/evote/service/rbac/RoleServiceBean.java 218 **Action** Assign r = findByPk(...): RoleRepository.findByPk may return NULL  
**Trace id** 26

```

215| * This metod deletes a role, and removes the role entity from the objects that depend on it.
216| */
217|public void delete(UserData userData, Long pk) {
>218|    Role r = roleRepository.findByPk(pk);
219|    TextId textId =
translationService.findTextIdByElectionEvent(electionEventRepository.findByPk(r.getElectionEvent().getPk()),
r.getName());
220|    translationService.deleteTextId(userData, textId); 221|    roleRepository.delete(userData, r);

```

**Sink :**

**File** admin-backend/src/main/java/no/evote/service/rbac/RoleServiceBean.java 218 **Action** Assign r = findByPk(...): RoleRepository.findByPk may return NULL  
**Trace id** 26

```

215| * This metod deletes a role, and removes the role entity from the objects that depend on it.
216| */
217|public void delete(UserData userData, Long pk) {
>218|    Role r = roleRepository.findByPk(pk);
219|    TextId textId =
translationService.findTextIdByElectionEvent(electionEventRepository.findByPk(r.getElectionEvent().getPk()),
r.getName());
220|    translationService.deleteTextId(userData, textId);
221|    roleRepository.delete(userData, r);

```

**Issue 951D76C8A465A373FEFA14420D81AE79:****Source :**

**File** admin-backend/src/main/java/no/evote/service/rbac/RoleServiceBean.java 295  
**Action** Assign parentRole = findByPk(...): RoleRepository.findByPk may return NULL  
**Trace id** 26

```

294|public boolean isCircularReference(UserData userData, CircularReferenceCheckRequest
circularReferenceCheckRequest) {
>295|    Role parentRole = roleRepository.findByPk(circularReferenceCheckRequest.getRole().getPk());
296|    Role includedRole = roleRepository.findByElectionEventAndId(userData.electionEvent(),
circularReferenceCheckRequest.getNewIncludedRoleId());
297|    for (Role r : roleRepository.getRoleWithIncludedRoles(includedRole)) {
298|        if (r.equals(parentRole)) {

```

**Sink :**

**File** admin-backend/src/main/java/no/evote/service/rbac/RoleServiceBean.java 295  
**Action** Assign parentRole = findByPk(...): RoleRepository.findByPk may return NULL **Trace id** 26

```

294|public boolean isCircularReference(UserData userData, CircularReferenceCheckRequest
circularReferenceCheckRequest) {
>295|    Role parentRole = roleRepository.findByPk(circularReferenceCheckRequest.getRole().getPk());
296|    Role includedRole = roleRepository.findByElectionEventAndId(userData.electionEvent(),
circularReferenceCheckRequest.getNewIncludedRoleId());
297|    for (Role r : roleRepository.getRoleWithIncludedRoles(includedRole)) {
298|        if (r.equals(parentRole)) {

```

## 6.83 SecurityInterceptor.java

**Issue A60CDAFEEE98A264D9014C34AB835482:****Source :**

**File** admin-backend/src/main/java/no/evote/service/security/SecurityInterceptor.java 178  
**Action** BranchNotTaken Branch not taken: (hasBeenVerifiedWithinTransaction == null)

```

176|private void verifyAccessCacheFromUserData(final UserData userData) {
177|    Boolean hasBeenVerifiedWithinTransaction = (Boolean) registry.getResource(ACCESS_CACHE_VERIFIED);
>178|    if (hasBeenVerifiedWithinTransaction != null && hasBeenVerifiedWithinTransaction) {
179|        return;

```

**Sink :**

**File** admin-backend/src/main/java/no/evote/service/security/SecurityInterceptor.java 184  
**Action** Assign findByPk(...): ElectionEventRepository.findByPk may return NULL **Trace id** 3

```

182|AccessCache accessCache = userData.getAccessCache();
183|if (accessCache.getSignature() == null) {
>184|    if (!electionEventRepository.findByPk(userData.getElectionEventPk()).getId().equals(ROOT_ELECTION_EVENT_ID))
{

```

```

185|         throw new EvoteSecurityException("Unable to find access cache signature.");
186|     } else {
187|         return;

```

## 6.84 SigningKeyRepository.java

### Issue E46331DC7C8BF2AA213BoBBFF2B61E07:

#### Source :

**File** admin-backend-common/src/main/java/no/valg/eva/admin/backend/common/repository/SigningKeyRepository.java 42 **Action** BranchTaken Branch taken: array~iterator.hasNext()

```

39|List<SigningKeyData> rows = new ArrayList<>();
40|
41|
>42|for (Object[] array : result) {
43|    SigningKeyData row = new SigningKeyData((String) array[1], (String) array[0], (String) array[3], (String)
array[2]);
44|    if (array[4] != null) {
45|        SigningKey signingKey = findByPk(Long.valueOf((Integer) array[4]));

```

#### Sink :

**File** admin-backend-common/src/main/java/no/valg/eva/admin/backend/common/repository/SigningKeyRepository.java 45

**Action** Assign signingKey = findByPk(...): SigningKeyRepository.findByPk may return NULL **Trace id** 62

```

42|for (Object[] array : result) {
43|    SigningKeyData row = new SigningKeyData((String) array[1], (String) array[0], (String) array[3], (String)
array[2]);
44|    if (array[4] != null) {
>45|        SigningKey signingKey = findByPk(Long.valueOf((Integer) array[4]));
46|        row.setSigningKeyPk(signingKey.getPk());
47|        if (signingKey.getBinaryData() != null) {
48|            row.setFileName(signingKey.getBinaryData().getFileName());

```

## 6.85 StartCountingController.java

### Issue 7F3665AA17F215AF391E4F740013161E:

#### Source :

**File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/counting/ctrls/StartCountingController.java 139

**Action** Assign getRequestServletPath(): ExternalContext.getRequestServletPath may return NULL

**Rule id** 4280F38B-9FDB-454E-B495-89CF45CD51B7

```

138|private boolean isStartCountingXhtmlPath() {
>139|    return getFacesContext().getExternalContext().getRequestServletPath().contains("startCounting.xhtml");
140|}
141|
142|public ElectionPath getContestPath() {

```

#### Sink :

**File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/counting/ctrls/StartCountingController.java 139

**Action** Assign getRequestServletPath(): ExternalContext.getRequestServletPath may return NULL

**Rule id** 4280F38B-9FDB-454E-B495-89CF45CD51B7

```

138|private boolean isStartCountingXhtmlPath() {
>139|    return getFacesContext().getExternalContext().getRequestServletPath().contains("startCounting.xhtml");
140|}
141|
142|public ElectionPath getContestPath() {

```

## 6.86 UserDataServiceEjb.java

### Issue 226E65883DB502A6E398A48CD8ECE9C1:

#### Source :

**File** admin-rbac/src/main/java/no/valg/eva/admin/rbac/service/UserDataServiceEjb.java 234

**Action** BranchTaken Branch taken: (municipality != null)

```

232|private boolean isElectronicMarkoffsConfigured(UserData userData) {
233|    Municipality municipality = userData.getOperatorRole().getMvArea().getMunicipality();
>234|    if (municipality != null) {
235|        municipality = municipalityRepository.findByPk(municipality.getPk()); // Lookup again to find latest
status for isElectronicMarkoffs
236|        return municipality.isElectronicMarkoffs();

```

#### Sink :

**File** admin-rbac/src/main/java/no/valg/eva/admin/rbac/service/UserDataServiceEjb.java 235

**Action** Assign municipality = findByPk(...): MunicipalityRepository.findByPk may return NULL **Trace id** 6



```

232|private boolean isElectronicMarkoffsConfigured(UserData userData) {
233|   Municipality municipality = userData.getOperatorRole().getMvArea().getMunicipality();
234|   if (municipality != null) {
>235|       municipality = municipalityRepository.findByPk(municipality.getPk()); // Lookup again to find latest
status for isElectronicMarkoffs
236|       return municipality.isElectronicMarkoffs();
237|   }
238|   return true;

```

## 6.87 V3\_\_Create\_Reference\_Data\_Baseline.java

**Issue** 45773847ED4D48D1A4AAF4BCoB67430F:

**Source :**

**File** admin-database/src/main/java/no/valg/eva/admin/database/migrations/V3\_\_Create\_Reference\_Data\_Baseline.java 132 **Action**

Assign classLoader = getClassLoader() : Class.getClassLoader may return NULL

**Rule id** 4280F38B-9FDB-454E-B495-89CF45CD51B7

```

129|private List<URL> urlsToBaselineDatafiles() {
130|   List<String> filenameBases = getFilenameBases();
131|   List<URL> classpathUrls = new ArrayList<>(filenameBases.size()); >132|   ClassLoader classLoader =
getClass().getClassLoader(); 133|
134|   for (String filenameBase : filenameBases) {
135|       String path = filePathOnClasspath(filenameBase);

```

**Sink :**

**File** admin-database/src/main/java/no/valg/eva/admin/database/migrations/V3\_\_Create\_Reference\_Data\_Baseline.java 132 **Action**

Assign classLoader = getClassLoader() : Class.getClassLoader may return NULL

**Rule id** 4280F38B-9FDB-454E-B495-89CF45CD51B7

```

129|private List<URL> urlsToBaselineDatafiles() {
130|   List<String> filenameBases = getFilenameBases();
131|   List<URL> classpathUrls = new ArrayList<>(filenameBases.size()); >132|   ClassLoader classLoader =
getClass().getClassLoader();
133|
134|   for (String filenameBase : filenameBases) {
135|       String path = filePathOnClasspath(filenameBase);

```

## 6.88 ValgnattElectoralRollDomainService.java

**Issue** A091DABAAA811C898562F120E6F47B3C:

**Source :**

**File** admin-valgnatt/src/main/java/no/valg/eva/admin/valgnatt/domain/service/grunnlagsdata/ValgnattElectoralRollDomainService.java 89

**Action** Assign municipalityPollingDistrict = findByPk(...): PollingDistrictRepository.findByPk may return NULL **Trace id** 7

```

87|private void leggTilSentraltSamlede(ElectoralRollCountReport electoralRollCountReport, Map<Long,
ElectoralRollCount> electoralRollForPollingDistrictMap,
88|   ReportConfiguration reportConfiguration) {
>89|   PollingDistrict municipalityPollingDistrict =
pollingDistrictRepository.findByPk(reportConfiguration.getPollingDistrictPk());
90|   Municipality municipality = municipalityPollingDistrict.getBorough().getMunicipality();
91|   electoralRollCountReport.add(createElectoralRollForPollingDistricts(
92|       electoralRollForPollingDistrictMap,

```

**Sink :**

**File** admin-valgnatt/src/main/java/no/valg/eva/admin/valgnatt/domain/service/grunnlagsdata/ValgnattElectoralRollDomainService.java 89

**Action** Assign municipalityPollingDistrict = findByPk(...): PollingDistrictRepository.findByPk may return NULL **Trace id** 7

```

87|private void leggTilSentraltSamlede(ElectoralRollCountReport electoralRollCountReport, Map<Long,
ElectoralRollCount> electoralRollForPollingDistrictMap,
88|   ReportConfiguration reportConfiguration) {
>89|   PollingDistrict municipalityPollingDistrict =
pollingDistrictRepository.findByPk(reportConfiguration.getPollingDistrictPk());
90|   Municipality municipality = municipalityPollingDistrict.getBorough().getMunicipality();
91|   electoralRollCountReport.add(createElectoralRollForPollingDistricts(
92|       electoralRollForPollingDistrictMap,

```

**Issue** D4C32C39CA2AB57A2FoBC41984490943:

**Source :**

**File** admin-valgnatt/src/main/java/no/valg/eva/admin/valgnatt/domain/service/grunnlagsdata/ValgnattElectoralRollDomainService.java 80

**Action** Assign parentPollingDistrict = findByPk(...): PollingDistrictRepository.findByPk may return NULL

**Trace id** 7

```

78|private void leggTilTellekretser(ElectoralRollCountReport electoralRollCountReport, Map<Long, ElectoralRollCount>
electoralRollForPollingDistrictMap,
79|   ReportConfiguration reportConfiguration) {
>80|   PollingDistrict parentPollingDistrict =
pollingDistrictRepository.findByPk(reportConfiguration.getPollingDistrictPk());
81|   electoralRollCountReport.add(createElectoralRollForPollingDistricts(

```

```
82|      electoralRollForPollingDistrictMap,
83|      reportConfiguration,
```

**Sink :**

**File** admin-valgnatt/src/main/java/no/valg/eva/admin/valgnatt/domain/service/grunnlagsdata/ValgnattElectoralRollDomainService.java 80 **Action** Assign parentPollingDistrict = findByPk(...): PollingDistrictRepository.findByPk may return NULL

**Trace id** 7

```
78|private void leggTilTellekretser(ElectoralRollCountReport electoralRollCountReport, Map<Long, ElectoralRollCount>
electoralRollForPollingDistrictMap,
79|      ReportConfiguration reportConfiguration) {
>80|      PollingDistrict parentPollingDistrict =
pollingDistrictRepository.findByPk(reportConfiguration.getPollingDistrictPk());
81|      electoralRollCountReport.add(createElectoralRollForPollingDistricts(
82|      electoralRollForPollingDistrictMap,
83|      reportConfiguration,
```

## 6.89 ValgnattReportApplicationService.java

### Issue ED7BoDoE143F912B7BC405EEDA14553F:

**Source :**

**File** admin-valgnatt/src/main/java/no/valg/eva/admin/valgnatt/application/service/ValgnattReportApplicationService.java 211

**Action** Assign valgnattrapport = byPk(...): ValgnattrapportRepository.byPk may return NULL **Trace id** 65

```
208|      contestPath.assertContestLevel();
209|      MvElection mvElectionContest = mvElectionRepository.finnEnkeltMedSti(contestPath.tilValghierarkiSti());
210|      MvArea reportForArea = mvAreaRepository.findSingleByPath(areaPath);
>211|      Valgnattrapport valgnattrapport =
valgnattrapportRepository.byPk(valgnattrapportering.getValgnattrapportPk());
212|      Resultatskjema stemmeskjema = stemmeskjemaDomainService.fraMøtebok(mvElectionContest, reportForArea); 213|
uploadAndLog(userData, stemmeskjema, valgnattrapport);
```

**Sink :**

**File** admin-valgnatt/src/main/java/no/valg/eva/admin/valgnatt/application/service/ValgnattReportApplicationService.java 211

**Action** Assign valgnattrapport = byPk(...): ValgnattrapportRepository.byPk may return NULL **Trace id** 65

```
208|      contestPath.assertContestLevel();
209|      MvElection mvElectionContest = mvElectionRepository.finnEnkeltMedSti(contestPath.tilValghierarkiSti());
210|      MvArea reportForArea = mvAreaRepository.findSingleByPath(areaPath);
>211|      Valgnattrapport valgnattrapport =
valgnattrapportRepository.byPk(valgnattrapportering.getValgnattrapportPk());
212|      Resultatskjema stemmeskjema = stemmeskjemaDomainService.fraMøtebok(mvElectionContest, reportForArea); 213|
uploadAndLog(userData, stemmeskjema, valgnattrapport);
```

## 6.90 VoteCountCategoryRepository.java

### Issue 771990A9E17AB25D359422E415FA7926:

**Source :**

**File** admin-configuration/src/main/java/no/valg/eva/admin/configuration/repository/VoteCountCategoryRepository.java 76

**Action** BranchNotTaken Branch not taken: includeBoroughCategory

```
75|private boolean includeBoroughCategory(Long municipalityPk, boolean includeBoroughCategory) {
>76|      if (!includeBoroughCategory) {
77|          return false;
78|      }
79|      Municipality municipality = findEntityByPk(Municipality.class, municipalityPk);
```

**Sink :**

**File** admin-configuration/src/main/java/no/valg/eva/admin/configuration/repository/VoteCountCategoryRepository.java 79

**Action** Assign municipality = findEntityByPk(...): BaseRepository.findEntityByPk may return NULL **Trace id** 1

```
76|      if (!includeBoroughCategory) {
77|          return false;
78|      }
>79|      Municipality municipality = findEntityByPk(Municipality.class, municipalityPk); 80|      return
municipality.getId().equals(AreaPath.OSLO_MUNICIPALITY_ID);
```

## 6.91 VoterServiceEjb.java

### Issue 8A850AA60B14C7ACCF8B5B4F16151771:

**Source :**

**File** admin-backend/src/main/java/no/evote/service/configuration/VoterServiceEjb.java 72

**Action** Assign electionEvent = findByPk(...): ElectionEventRepository.findByPk may return NULL **Trace id** 3

```
69|@Security(accesses = Manntall_Rediger_Person, type = WRITE)
70|@AuditLog(eventClass = VoterAuditEvent.class, eventType = AuditEventTypes.Create)
71|public Voter create(UserData userData, Voter voter) {
>72|     ElectionEvent electionEvent = electionEventRepository.findByPk(userData.getElectionEventPk());
73|     if (!findByElectionEventAndId(userData, voter.getId(), electionEvent.getPk()).isEmpty()) { 74|         throw
new EvoteException("@electoralRoll.ssnAlreadyInElectoralRoll");
```

**Sink :**

**File** admin-backend/src/main/java/no/evote/service/configuration/VoterServiceEjb.java 72

**Action** Assign electionEvent = findByPk(...): ElectionEventRepository.findByPk may return NULL **Trace id** 3

```
69|@Security(accesses = Manntall_Rediger_Person, type = WRITE)
70|@AuditLog(eventClass = VoterAuditEvent.class, eventType = AuditEventTypes.Create)
71|public Voter create(UserData userData, Voter voter) {
>72|     ElectionEvent electionEvent = electionEventRepository.findByPk(userData.getElectionEventPk());
73|     if (!findByElectionEventAndId(userData, voter.getId(), electionEvent.getPk()).isEmpty()) { 74|         throw
new EvoteException("@electoralRoll.ssnAlreadyInElectoralRoll");
```

## 6.92 VotingServiceBean.java

**Issue** 574A971B9D5564286DFF699FF9EC0340:

**Source :**

**File** admin-voting/src/main/java/no/valg/eva/admin/voting/service/VotingServiceBean.java 305

**Action** Assign velger = findByPk(...): VoterRepository.findByPk may return NULL **Trace id** 27

```
303|valggruppeSti.assertElectionGroupLevel();
304|
>305|velger = voterRepository.findByPk(velger.getPk());
306|MvArea stemmested = mvAreaService.findSingleByPath(stemmestedSti);
307|VelgerSomSkalStemme resultat = initVelgerMeldinger(stemmegivningsType, stemmested, velger);
```

**Sink :**

**File** admin-voting/src/main/java/no/valg/eva/admin/voting/service/VotingServiceBean.java 307

**Action** InOutCall ?.initVelgerMeldinger(?, ?, velger): VotingServiceBean.initVelgerMeldinger may return NULL **Trace id** 28

```
305|velger = voterRepository.findByPk(velger.getPk());
306|MvArea stemmested = mvAreaService.findSingleByPath(stemmestedSti);
>307|VelgerSomSkalStemme resultat = initVelgerMeldinger(stemmegivningsType, stemmested, velger);
308|
309|if (sjekkForhåndsstemmerStengtPgaAvkryssingsmanntallKjort(stemmegivningsType, stemmested.getMunicipality(),
velger, resultat)) {
310|     return resultat;
```

**Issue** 784869C3046CB05D34A936FFA4CBE869:

**Source :**

**File** admin-voting/src/main/java/no/valg/eva/admin/voting/service/VotingServiceBean.java 305

**Action** Assign velger = findByPk(...): VoterRepository.findByPk may return NULL **Trace id** 27

```
303|valggruppeSti.assertElectionGroupLevel();
304|
>305|velger = voterRepository.findByPk(velger.getPk());
306|MvArea stemmested = mvAreaService.findSingleByPath(stemmestedSti);
307|VelgerSomSkalStemme resultat = initVelgerMeldinger(stemmegivningsType, stemmested, velger);
```

**Sink :**

**File** admin-voting/src/main/java/no/valg/eva/admin/voting/service/VotingServiceBean.java 309 **Action**

InOutCall ?.sjekkForhåndsstemmerStengtPgaAvkryssingsmanntallKjort(?, ?, velger, ?):

VotingServiceBean.sjekkForhåndsstemmerStengtPgaAvkryssingsmanntallKjort may return NULL **Trace id** 33

```
306|MvArea stemmested = mvAreaService.findSingleByPath(stemmestedSti);
307|VelgerSomSkalStemme resultat = initVelgerMeldinger(stemmegivningsType, stemmested, velger);
308|
>309|if (sjekkForhåndsstemmerStengtPgaAvkryssingsmanntallKjort(stemmegivningsType, stemmested.getMunicipality(),
velger, resultat)) {
310|     return resultat;
```

**Issue** 80DA821AAD0FB891B64DD7C15725B45F:

**Source :**

**File** admin-voting/src/main/java/no/valg/eva/admin/voting/service/VotingServiceBean.java 305

**Action** Assign velger = findByPk(...): VoterRepository.findByPk may return NULL **Trace id** 27

```

303|valggruppeSti.assertElectionGroupLevel();
304|
>305|velger = voterRepository.findByPk(velger.getPk());
306|MvArea stemmested = mvAreaService.findSingleByPath(stemmestedSti);
307|VelgerSomSkalStemme resultat = initVelgerMeldinger(stemmegivningsType, stemmested, velger);

```

**Sink :****File** admin-voting/src/main/java/no/valg/eva/admin/voting/service/VotingServiceBean.java 305**Action** Assign velger = findByPk(...): VoterRepository.findByPk may return NULL **Trace id** 27

```

303|valggruppeSti.assertElectionGroupLevel();
304|
>305|velger = voterRepository.findByPk(velger.getPk());
306|MvArea stemmested = mvAreaService.findSingleByPath(stemmestedSti);
307|VelgerSomSkalStemme resultat = initVelgerMeldinger(stemmegivningsType, stemmested, velger);

```

## 6.93 XMLUtil.java

### Issue B262353B30B797919239C4A5583E2313:

**Source :****File** admin-common/src/main/java/no/evote/util/XMLUtil.java 44**Action** Assign classLoader = getClassLoader(): Class.getClassLoader may return NULL**Rule id** 4280F38B-9FDB-454E-B495-89CF45CD51B7

```

>44|ClassLoader classLoader = XMLUtil.class.getClassLoader();
45|Source[] sources = new Source[xsdFileNames.length];
46|int index = 0;
47|for (String xsdFilename : xsdFileNames) {

```

**Sink :****File** admin-common/src/main/java/no/evote/util/XMLUtil.java 44**Action** Assign classLoader = getClassLoader(): Class.getClassLoader may return NULL**Rule id** 4280F38B-9FDB-454E-B495-89CF45CD51B7

```

>44|ClassLoader classLoader = XMLUtil.class.getClassLoader();
45|Source[] sources = new Source[xsdFileNames.length];
46|int index = 0;
47|for (String xsdFilename : xsdFileNames) {

```

## [Medium] Missing Check for Null Parameter

*API Abuse*

### Analysis: Bad Practice

### Abstract

This function violates the contract that it must compare its parameter with null.

### Explanation

The Java standard requires that implementations of `Object.equals()`, `Comparable.compareTo()`, and `Comparator.compare()` must return a specified value if their parameters are null. Failing to follow this contract may result in unexpected behavior.

**Example 1:** The following implementation of the `equals()` method does not compare its parameter with null.

```

public boolean equals(Object object)
{
    return (toString().equals(object.toString())); }

```

### Recommendation

Always follow the contract that `Object.equals()` must return `null` if it receives a null parameter.

**Example 2:** The previous example is rewritten to explicitly check for a null argument and return false if one is found.

```

public boolean equals(Object object)
{
    if (object ==
null)
        return
false;
    return (toString().equals(object.toString()));
}

```

## 6.94 BatchInfoDto.java

**Issue** 2D57546D20AA6677521FC161C2236770:

**Source :**

**File** admin-common/src/main/java/no/evote/dto/BatchInfoDto.java 55

**Action** o = #param(o)

**Rule id** 1834C857-2848-41F2-98F7-D5C5F157E729

```

54|@Override
>55|public boolean equals(final Object o) {
56|    if (this.getClass() != o.getClass()) {
57|        return false;

```

**Sink :**

**File** admin-common/src/main/java/no/evote/dto/BatchInfoDto.java 55 **Action** o = #param(o)

**Rule id** 1834C857-2848-41F2-98F7-D5C5F157E729

```

54|@Override
>55|public boolean equals(final Object o) {
56|    if (this.getClass() != o.getClass()) {
57|        return false;

```

## 6.95 OperatorRole.java

**Issue** E97F19BA8F8B2476AE66772B8767A7F5:

**Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/rbac/domain/model/OperatorRole.java 122 **Action** obj = #param(o)

**Rule id** 1834C857-2848-41F2-98F7-D5C5F157E729

```

121|@Override
>122|public boolean equals(final Object obj) {
123|    if (this.getClass() == obj.getClass()) {
124|        return ((OperatorRole) obj).getRole().getPk().equals(getRole().getPk()) && ((OperatorRole)
obj).getOperator().equals(getOperator())
125|        && ((OperatorRole) obj).getMvArea().getPk().equals(getMvArea().getPk())

```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/rbac/domain/model/OperatorRole.java 122 **Action** obj = #param(o)

**Rule id** 1834C857-2848-41F2-98F7-D5C5F157E729

```

121|@Override
>122|public boolean equals(final Object obj) {
123|    if (this.getClass() == obj.getClass()) {
124|        return ((OperatorRole) obj).getRole().getPk().equals(getRole().getPk()) && ((OperatorRole)
obj).getOperator().equals(getOperator())
125|        && ((OperatorRole) obj).getMvArea().getPk().equals(getMvArea().getPk())

```

## 6.96 Role.java

**Issue** 25C9DFD015CoB73D3530E871757752D4:

**Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/rbac/domain/model/Role.java 238 **Action** obj = #param(o)

**Rule id** 1834C857-2848-41F2-98F7-D5C5F157E729

```

237|@Override
>238|public boolean equals(final Object obj) {
239|    if (this.getClass() == obj.getClass()) {
240|        return getElectionEvent().getPk().equals(((Role) obj).getElectionEvent().getPk()) &&
getId().equals(((Role) obj).getId());

```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/rbac/domain/model/Role.java 238 **Action** obj = #param(o)

**Rule id** 1834C857-2848-41F2-98F7-D5C5F157E729

```

237|@Override
>238|public boolean equals(final Object obj) {
239|    if (this.getClass() == obj.getClass()) {

```

```
240|         return getElectionEvent().getPk().equals(((Role) obj).getElectionEvent().getPk()) &&
|        getId().equals(((Role) obj).getId());
```

## [Medium] Poor Error Handling / Throw Inside Finally

*Errors*

**Analysis:** Bad Practice

### Abstract

Using a `throw` statement inside a `finally` block breaks the logical progression through the `trycatch-finally`.

### Explanation

In Java, `finally` blocks are always executed after their corresponding `try-catch` blocks and are often used to free allocated resources, such as file handles or database cursors. Throwing an exception in a `finally` block can bypass critical cleanup code since normal program execution will be disrupted.

**Example 1:** In the following code, the call to `stmt.close()` is bypassed when the `FileNotFoundException` is thrown.

```
public void processTransaction(Connection conn) throws FileNotFoundException
{
    FileInputStream fis =
    null;    Statement stmt =
    null;    try    {
        stmt = conn.createStatement();
        fis = new
    FileInputStream("badFile.txt");        ...
    }
    catch (FileNotFoundException fe)
    {
        log("File not found.");
    }
    catch (SQLException se)
    {
        //handle error
    }
    finally
    {
        if (fis == null)
        {
            throw new FileNotFoundException();
        }
        if (stmt != null)
        {
            try
            {
                stmt.close();
            }
            catch (SQLException e)
            {
                log(e);
            }
        }
    }
}
```

This category is from the Cigital Java Rulepack. <http://www.cigital.com/>

### Recommendation

Never throw exceptions from within `finally` blocks. If you must re-throw an exception, do it inside a `catch` block so as not to interrupt the normal execution of the `finally` block. **Example 2:** The following code re-throws the `FileNotFoundException` in the `catch` block.

```

public void processTransaction(Connection conn) throws FileNotFoundException
{
    FileInputStream fis =
null;    Statement stmt =
null;    try    {
        stmt = conn.createStatement();
        fis = new
FileInputStream("badFile.txt");    ...
    }
    catch (FileNotFoundException fe)
    {
        log("File not
found.");
        throw fe;
    }
    catch
(SQLException se)
    {
        //handle error
    }
finally
{
    if (fis != null)
    {
        fis.close();
    }
    catch (IOException ie)
    {
        log(ie);
    }
    if (stmt != null)
    {
        stmt.close();
    }
    catch (SQLException e)
    {
        log(e);
    }
}
}
}

```

## 6.97 CSVUtil.java

**Issue** 00E096704467C4B0900685E088F9AC54:

**Sink :**

**File** admin-common/src/main/java/no/evote/util/CSVUtil.java 76-78

```

74|    } catch (IOException ioe) {
75|        throw new EvoteException("Error reading from csv", ioe);
>76|    } finally {
>77|        stream.close();
>78|    }
79|
80|    return rows;

```

## 6.98 ExportServiceBean.java

**Issue** FD9421F8B10F4924217E4D73EC41D3A4:

**Sink :**

**File** admin-backend/src/main/java/no/evote/service/ExportServiceBean.java 1086-1094

```

1083|        rowData.put(resMeta.getColumnName(columnIndex), res.getObject(columnIndex));
1084|    }
1085|    }
>1086|    } finally {
>1087|        if (res != null) {
>1088|            res.close();
>1089|        }
>1090|    }

```

```

>1091|         if (stmt != null) {
>1092|             stmt.close();
>1093|         }
>1094|     }
1095| }
1096|});

```

## 6.99 RoleAccessMigration.java

**Issue** 2F3C420097CFF864F6E1A320494058B5:

**Sink :**

**File** admin-database/src/main/java/no/valg/eva/admin/database/migrations/RoleAccessMigration.java 130-134

```

127|         while (rs.next()) {
128|             roleMap.put(rs.getString(2), rs.getLong(1));
129|         }
>130|     } finally {
>131|         if (rs != null) {
>132|             rs.close();
>133|         }
>134|     }
135|     election.setRoleMap(roleMap);

```

### [Medium] Privacy Violation / Heap Inspection

*Security Features*

**Analysis:** Suspicious

#### Abstract

Storing sensitive data in a `String` object makes it impossible to reliably purge the data from memory.

#### Explanation

Sensitive data (such as passwords, social security numbers, credit card numbers etc) stored in memory can be leaked if memory is not cleared after use. Often, `Strings` are used store sensitive data, however, since `String` objects are immutable, removing the value of a `String` from memory can only be done by the JVM garbage collector. The garbage collector is not required to run unless the JVM is low on memory, so there is no guarantee as to when garbage collection will take place. In the event of an application crash, a memory dump of the application might reveal sensitive data.

**Example 1:** The following code converts a password from a character array to a `String`.

```

private JPasswordField pf;
...
final char[] password = pf.getPassword();
...
String passwordAsString = new String(password);

```

This category was derived from the Cigital Java Rulepack. <http://www.cigital.com/>

#### Recommendation

Always be sure to clear sensitive data when it is no longer needed. Instead of storing sensitive data in immutable objects like `Strings`, use byte arrays or character arrays that can be programmatically cleared.

**Example 2:** The following code clears memory after a password is used.

```

private JPasswordField pf; ... final char[] password = pf.getPassword(); // use the password ...
// erase when finished
Arrays.fill(password, ' ');

```



## 6.100 CryptoServiceBean.java

**Issue** 303337088935B61BF82BF235E7D80850:

**Source :**

**File** admin-backend-common/src/main/java/no/evote/service/CryptoServiceBean.java 334 **Action** Read Read p12Password

**Rule id** FE990778-39D5-4C60-AFDB-F7ED34EE2AC10

```
332|try (InputStream fis = new ByteArrayInputStream(electionEventSigningKey.getBinaryData().getBinaryData())) {
333|    Pkcs12Decoder pkcs12Decoder = new Pkcs12Decoder();
>334|    SertifikatOgNøkkel sertifikatOgNøkkel = pkcs12Decoder.lesPkcs12(fis, new String(p12Password,
CHARACTER_SET)); 335|    electionEventPKCS12Cache.put(new Element(cacheKey, sertifikatOgNøkkel)); 336|
337|    return sertifikatOgNøkkel;
```

**Sink :**

**File** admin-backend-common/src/main/java/no/evote/service/CryptoServiceBean.java 334 **Action** InCall String(o)

**Rule id** 410586B4-5269-4727-8289-1BAF5FF376A3

**Fact** Call Direct : java.lang.String.String

**Fact** TaintFlags PRIVATE

```
332|try (InputStream fis = new ByteArrayInputStream(electionEventSigningKey.getBinaryData().getBinaryData())) {
333|    Pkcs12Decoder pkcs12Decoder = new Pkcs12Decoder();
>334|    SertifikatOgNøkkel sertifikatOgNøkkel = pkcs12Decoder.lesPkcs12(fis, new String(p12Password, CHARACTER_SET));
335|    electionEventPKCS12Cache.put(new Element(cacheKey, sertifikatOgNøkkel)); 336|
337|    return sertifikatOgNøkkel;
```

## 6.101 SystemPasswordApplicationService.java

**Issue** AB3BD019940D75583798746A5E3EB58C:

**Source :**

**File** admin-backend/src/main/java/no/valg/eva/admin/backend/application/service/SystemPasswordApplicationService.java 80

**Action** Read Read p12Password

**Rule id** FE990778-39D5-4C60-AFDB-F7ED34EE2AC10

```
78|    try (InputStream fis = new ByteArrayInputStream(adminSigningkeys.get(0).getBinaryData().getBinaryData())) {
79|        Pkcs12Decoder pkcs12Decoder = new Pkcs12Decoder();
>80|        return pkcs12Decoder.validerPassord(fis, new String(p12Password, CHARACTER_SET));
81|    }
82|} catch (IOException | CryptoException e) {
83|    throw new EvoteException(e.getMessage(), e);
```

**Sink :**

**File** admin-backend/src/main/java/no/valg/eva/admin/backend/application/service/SystemPasswordApplicationService.java 80

**Action** InCall String(o)

**Rule id** 410586B4-5269-4727-8289-1BAF5FF376A3

**Fact** Call Direct : java.lang.String.String

**Fact** TaintFlags PRIVATE

```
78|    try (InputStream fis = new ByteArrayInputStream(adminSigningkeys.get(0).getBinaryData().getBinaryData())) {
79|        Pkcs12Decoder pkcs12Decoder = new Pkcs12Decoder();
>80|        return pkcs12Decoder.validerPassord(fis, new String(p12Password, CHARACTER_SET));
81|    }
82|} catch (IOException | CryptoException e) {
83|    throw new EvoteException(e.getMessage(), e);
```

**[Medium] Redundant Null Check**

*Code Quality*

**Analysis:** Bad Practice

### Abstract

The program can potentially dereference a null pointer, thereby causing a null pointer exception.

### Explanation

Null pointer exceptions usually occur when one or more of the programmer's assumptions is violated. A check-after-dereference error occurs when a program dereferences an object that can be `null` before checking if the object is `null`.

Most null pointer issues result in general software reliability problems, but if attackers can intentionally trigger a null pointer dereference, they can use the resulting exception to bypass security logic or to

cause the application to reveal debugging information that will be valuable in planning subsequent attacks.

**Example:** In the following code, the programmer assumes that the variable `foo` is not `null` and confirms this assumption by dereferencing the object. However, the programmer later contradicts the assumption by checking `foo` against `null`. If `foo` can be `null` when it is checked in the `if` statement then it can also be `null` when it is dereferenced and might cause a null pointer exception. Either the dereference is unsafe or the subsequent check is unnecessary.

```
foo.setBar(val); ...
if (foo != null) { ...
}
```

## Recommendation

Implement careful checks before dereferencing objects that might be `null`. When possible, abstract null checks into wrappers around code that manipulates resources to ensure that they are applied in all cases and to minimize the places where mistakes can occur.

## 6.102 SpecialPurposeReportsServiceEjb.java

**Issue** 6E42ED3517726DA9F34B4F8402AB2007:

**Source :**

**File** admin-backend/src/main/java/no/evote/service/SpecialPurposeReportsServiceEjb.java 309

**Action** Dereferenced : mvArea

```
306|if (isSamiElection) {
307|    // Decide if municipality of voter is normal municipality or child area in sami election
308|    List<ContestArea> contestAreas =
contestAreaRepository.findContestAreaChildForElectionGroupAndMunicipality(electionGroup.getPk(), >309|
mvArea.getPk());
310|    if (contestAreas.size() == 1) {
311|        // Municipality is a child Area. Assume this is Sami election. Find the parent counting electoral board
(opptellings valgstyre)
312|        ReportingUnit countElectoralBoardReportingUnit =
reportingUnitRepository.findCountElectoralBoardByContest(contestAreas.get(0).getContest());
```

**Sink :**

**File** admin-backend/src/main/java/no/evote/service/SpecialPurposeReportsServiceEjb.java 309

**Action** Dereferenced : mvArea

```
306|if (isSamiElection) {
307|    // Decide if municipality of voter is normal municipality or child area in sami election
308|    List<ContestArea> contestAreas =
contestAreaRepository.findContestAreaChildForElectionGroupAndMunicipality(electionGroup.getPk(), >309|
mvArea.getPk());
310|    if (contestAreas.size() == 1) {
311|        // Municipality is a child Area. Assume this is Sami election. Find the parent counting electoral board
(opptellings valgstyre)
312|        ReportingUnit countElectoralBoardReportingUnit =
reportingUnitRepository.findCountElectoralBoardByContest(contestAreas.get(0).getContest());
```

## [Medium] System Information Leak / HTML Comment in JSP

*Encapsulation*

**Analysis:** Bad Practice

### Abstract

Any information revealed in an HTML comment might help an adversary learn about the system and form a plan of attack.

## Explanation

HTML comments provide an attacker with an easy source of information about a dynamically generated web page.

### Example 1:

```
<!-- TBD: this needs a security audit -->
<form method="POST" action="recalcOrbit"> ...
```

Even comments that seem innocuous may be useful to someone trying to understand the way the system is built.

## Recommendation

Replace HTML comments with JSP comments (which will not be transmitted to the user).

**Example 2:** The previous example is rewritten to use JSP comments, which will not be displayed to the user.

```
<%-- TBD: this needs a security audit --%>
<form method="POST" action="recalcOrbit"> ...
```

## 6.103 accessOverviewOperators.xhtml

**Issue** 0FCF22EBB40A3169B6908DBF207E859C:

**Sink :**

**File** admin-frontend/src/main/webapp/secure/rbac/accessOverviewOperators.xhtml 28

```
25| <div class="row">
26|   <div class="col-md-12">
27|     <h4><h:outputText value="#{msgs['@rbac.inspect.operators.title']}" /> - <h:outputText style="font-size:
1rem" value="#{accessOverviewOperatorsController.accessPathName}" /></h4>
>28|     <!--@elvariable id="operator" type="no.valg.eva.admin.rbac.domain.model.Operator"-->
29|     <p:dataTable id="operatorTable"
30|                 var="operator"
31|                 pt:data-af= "operatorTable"
```

**Issue** 3943D336993650D28A482F8ADA9F5155:

**Sink :**

**File** admin-frontend/src/main/webapp/secure/rbac/accessOverviewOperators.xhtml 77

```
74| <div class="row">
75|   <div class="col-md-12">
76|     <h4><h:outputText value="#{msgs['@rbac.inspect.roles.title']}" /></h4>
>77|     <!--@elvariable id="operatorRole" type="no.valg.eva.admin.rbac.domain.model.OperatorRole"--> 78|
<p:dataTable id="rolesTable"
79|                 var="operatorRole"
80|                 value="#{accessOverviewOperatorsController.operatorRoles}"
```

## 6.104 advancePollingPlace.xhtml

**Issue** 216E4E86DA922D89E22A0768E091284B:

**Sink :**

**File** admin-frontend/src/main/webapp/resources/config/advancePollingPlace.xhtml 33

```
31| <div class="row">
32|   <div class="col-xs-12">
>33|     <!--@elvariable id="pd" type="no.valg.eva.admin.common.configuration.model.local.AdvancePollingPlace"-->
34|     <!--@elvariable id="pollingDistrict"
type="no.valg.eva.admin.common.configuration.model.local.AdvancePollingPlace"-->
35|     <p:selectOneListBox
36|         id="list-box"
```

**Issue** DFB1868C3BFCD943C94C5F3125AED723:

**Sink :**

**File** admin-frontend/src/main/webapp/resources/config/advancePollingPlace.xhtml 34

```

31|<div class="row">
32|   <div class="col-xs-12">
33|     <!--@elvariable id="pd" type="no.valg.eva.admin.common.configuration.model.local.AdvancePollingPlace"-->
>34|     <!--@elvariable id="pollingDistrict"
type="no.valg.eva.admin.common.configuration.model.local.AdvancePollingPlace"-->
35|     <p:selectOneListBox
36|       id="list-box"
37|       pt:data-aft="selectPlace"

```

## 6.105 antallStemmesedlerLagtTilSide.xhtml

**Issue** 9915941DB7A8CA43CF7A6B5EA93D8BB5:

**Sink :**

**File** admin-frontend/src/main/webapp/secure/counting/antallStemmesedlerLagtTilSide.xhtml 29

```

27|<p:remoteCommand name="refreshFooter" update="antallStemmesedlerModel:antallStemmesedlerLagtTilSideTotal"/>
28|
>29|<!--@elvariable id="row" type="no.valg.eva.admin.frontend.counting.view.ballotcount.BallotCountRow"-->
30|<p:dataTable value="#{antallStemmesedlerLagtTilSideController.antallStemmesedlerModel}" var="row"
id="antallStemmesedlerModel"
31|   tableStyleClass="table table-striped #{antallStemmesedlerLagtTilSideController.headerFooterStyle}"
style="width: 50%;"
32|   rowStyleClass="#{row.rowStyleClass}">

```

## 6.106 areaSelectPanel.xhtml

**Issue** EB900DB371A9EFDEAFBF7EA5B598CBF9:

**Sink :**

**File** admin-frontend/src/main/webapp/resources/widget/areaSelectPanel.xhtml 63

```

60|</h:panelGroup>
61|
62|<!-- Municipality -->
>63|<!-- NOTE EVA-811: It must be possible to select municipality also on county level -->
64|<h:panelGroup id="municipality" layout="block"
65|   styleClass="#{mvAreaController.selectedAreaLevel == 2 ? '' : 'required'}form-group col-md-3
#{mvAreaController.selectedAreaLevel==3 ?
66|   'updateNameId' : ''}"

```

## 6.107 avansertTab.xhtml

**Issue** 7A4D5B4AD0073F32E9946272963736D2:

**Sink :**

**File** admin-frontend/src/main/webapp/resources/manntall/partials/manntallsSok/avansertTab.xhtml 8

```

5|   xmlns:f="http://java.sun.com/jsf/core"
6|   contentType="text/html">
7|
> 8|<!--@elvariable id="kommuneListe" type="java.util.List"-->
9|<!--@elvariable id="visIMantallKolonne" type="java.lang.Boolean"--> 10|
11|<div class="row">

```

**Issue** DB5CF5F1B13131544A7B0CEAA00A448F:

**Sink :**

**File** admin-frontend/src/main/webapp/resources/manntall/partials/manntallsSok/avansertTab.xhtml 9

```

6|   contentType="text/html">
7|
8|<!--@elvariable id="kommuneListe" type="java.util.List"-->
> 9|<!--@elvariable id="visIMantallKolonne" type="java.lang.Boolean"--> 10|
11|<div class="row">
12|   <div class="col-md-12 search-container">

```

**Issue** FDD3B58C187D08297C57569CB59E9506:

**Sink :**

**File** admin-frontend/src/main/webapp/resources/manntall/partials/manntallsSok/avansertTab.xhtml 53

```

50|</div>
51|
52|<div class="form-group">
>53|   <!--@elvariable id="kommune" type="no.valg.eva.admin.felles.valggeografi.model.Kommune"-->
54|   <h:outputLabel for="municipality" value="#{msgs['@person.municipality']}"/>
55|   <h:inputText
56|       id="municipalityStatic"

```

## 6.108 ballotCounts.xhtml

### Issue 4520CF84DCB594EC00BE182418CCBEF4:

Sink :

File admin-frontend/src/main/webapp/resources/counting/ballotCounts.xhtml 26

```

24|<h4>#{msgs[cc.attrs.model.getTabTitle()]}</h4> 25|
>26|<!--@elvariable id="row" type="no.valg.eva.admin.frontend.counting.view.ballotcount.BallotCountsModelRow"-->
27|<p:dataTable value="#{cc.attrs.model}" var="row" id="ballotCountsTable" tableStyleClass="table table-striped"
rowStyleClass="#{row.rowStyleClass}">
28|
29|   <p:column headerText="#{msgs['@common.party']}">

```

### Issue C128DF73DA4D424C2D7236C9E4D7FBE5:

Sink :

File admin-frontend/src/main/webapp/resources/widget/ballotCounts.xhtml 8

```

5|   xmlns:h="http://java.sun.com/jsf/html"
6|   xmlns:p="http://primefaces.org/ui">
7|
> 8|<!-- NOTE: ballotCountsList must contain objects of type
no.valg.eva.admin.frontend.counting.view.CompareBallotCountView -->
9|<composite:interface>
10|   <composite:attribute name="ballotCountList"
type="no.valg.eva.admin.frontend.counting.view.CompareBallotCounts" required="true" /> 11|</composite:interface>

```

## 6.109 ballotCountsWithSplit.xhtml

### Issue 4520CF84DCB594EC00BE182418CCBEF5:

Sink :

File admin-frontend/src/main/webapp/resources/counting/ballotCountsWithSplit.xhtml 26

```

24|<h4>#{msgs[cc.attrs.model.getTabTitle()]}</h4> 25|
>26|<!--@elvariable id="row" type="no.valg.eva.admin.frontend.counting.view.ballotcount.BallotCountsModelRow"-->
27|<p:dataTable value="#{cc.attrs.model}" var="row" id="ballotCountsTable" tableStyleClass="table table-striped"
rowStyleClass="#{row.rowStyleClass}">
28|
29|   <p:column headerText="#{msgs['@common.party']}">

```

## 6.110 behandleManueltForkastede.xhtml

### Issue 5D77B1B3A38807BDC0DEF6D1607C019:

Sink :

File admin-frontend/src/main/webapp/secure/opptelling/behandleManueltForkastede.xhtml 18

```

15|       <h:outputText value="#{msgs[behandleManueltForkastedeController.sideTittle]}/>
16|       </h1>
17|
>18|       <!-- DEV-NOTE! Placeholder xhtml. Bruker redirectes til startCount.xhtml etter valg i kontekstvelger. -->
19|
20|   </ui:define>
21|</ui:composition>

```

## 6.111 behandleSkannetForkastede.xhtml

### Issue 5D77B1B3A38807BDC0DEF6D1607C01A:

Sink :

File admin-frontend/src/main/webapp/secure/opptelling/behandleSkannetForkastede.xhtml 18

```

15|         <h:outputText value="#{msgs[behandleSkannetForkastedeController.sideTittle]}"/>
16|     </h1>
17|
>18|     <!-- DEV-NOTE! Placeholder xhtml. Bruker redirectes til startCount.xhtml etter valg i kontekstvelger. -->
19|
20|     </ui:define>
21| </ui:composition>

```

## 6.112 boardMemberAddFromElectoralRoll.xhtml

Issue 88F57AFDA48E90A26BCF0E3A17232E41:

Sink :

File admin-frontend/src/main/webapp/resources/config/boardMemberAddFromElectoralRoll.xhtml 72

```

69| <section class="panel-body">
70|     <div class="form-group required"> 71|
<h:outputLabel for="responsibility"
value="#{msgs['@reporting_unit.responsible_officer.responsibility.name']}"/>
>72|     <!--@elvariable id="responsibility" type="no.valg.eva.admin.common.counting.constants.ResponsibilityId"-->
73|     <p:selectOneMenu
74|         id="responsibility"
75|         pt:data-aft="responsibility"

```

## 6.113 boardMemberForm.xhtml

Issue 88F57AFDA48E90A26BCF0E3A17232E42:

Sink :

File admin-frontend/src/main/webapp/resources/config/boardMemberForm.xhtml 19

```

16| <div class="row">
17|     <div class="col-md-12 form-group required">
18|         <h:outputLabel for="responsibilityId"
value="#{msgs['@reporting_unit.responsible_officer.responsibility.name']}"/>
>19|         <!--@elvariable id="responsibility" type="no.valg.eva.admin.common.counting.constants.ResponsibilityId"-->
20|         <p:selectOneMenu
21|             id="responsibilityId"
22|             pt:data-aft="responsibilityId"

```

## 6.114 boardMemberSearchResult.xhtml

Issue FC3D3FB62D882EEA5ACF43C24BB1F353:

Sink :

File admin-frontend/src/main/webapp/resources/config/boardMemberSearchResult.xhtml 17

```

15| <div class="row searchResult">
16|     <div class="col-xs-12">
>17|         <!--@elvariable id="responsibleOfficer"
type="no.valg.eva.admin.common.configuration.model.local.ResponsibleOfficer"-->
18|         <p:dataTable
19|             id="responsibleOfficer"
20|             var="responsibleOfficer"

```

## 6.115 chooseEditListProposal.xhtml

Issue F4FDfE36996F5C79C194BFCB98D1DD19:

Sink :

File admin-frontend/src/main/webapp/secure/listProposal/chooseEditListProposal.xhtml 76

```

74| </div>
75|
>76| <!--@elvariable id="affiliation" type="no.valg.eva.admin.configuration.domain.model.Affiliation"-->
77| <p:dataTable
78|     id="affiliationTable"
79|     update="form"

```

## 6.116 configureCentralCountCategories.xhtml

Issue 79249AE2E2DAE4ECC49D3F3F707659BC:

Sink :

File admin-frontend/src/main/webapp/secure/categories/configureCentralCountCategories.xhtml 53

```

51| <div class="row">
52|     <div class="col-md-12 form-group">
>53|         <!--@elvariable id="cat" type="no.evote.presentation.config.counting.ElectionVoteCountCategoryElement"-->

```

```

54|         <h3>#{msgs['@election_vote_count_category.list.panelheader.list_election_vote_count_categories']}</h3>
55|         <p:dataTable id="CountCategory" var="cat"
56|             value="#{electionVoteCountCategoryController.elements}"

```

## 6.117 contextEditor.xhtml

### Issue oBE2FEC30E49FB09151845583F43316C:

Sink :

File admin-frontend/src/main/webapp/resources/widget/contextEditor.xhtml 158

```

155|         action="#{mvAreaPickerController.editMvArea(areaItem)}"
156|         update=":level#{areaType.level}Form"
157|         rendered="#{mvAreaPickerController.getIsEditable(areaType.level, areaItem)}">
>158|         <!--oncomplete="PF('editAreaLevel#{areaType.level}Widget').show()";--> 159|</p:commandLink>
160|
161|</p:commandLink

```

### Issue oBE2FEC30E49FB09151845583F43316D:

Sink :

File admin-frontend/src/main/webapp/resources/widget/contextEditor.xhtml 168

```

165|         action="#{mvAreaPickerController.readMvArea(areaItem)}"
166|         update=":level#{areaType.level}Form"
167|         rendered="#{mvAreaPickerController.getIsEditable(areaType.level, areaItem)}">
>168|         <!--oncomplete="PF('editAreaLevel#{areaType.level}Widget').show()";-->
169|         </p:commandLink>
170|</h:panelGroup>

```

### Issue 3E4914760A3E27AE5372CA6AE34CC638:

Sink :

File admin-frontend/src/main/webapp/resources/widget/contextEditor.xhtml 182

```

179|</h:panelGroup>
180|
181|<c:if test="#{mvAreaPickerController.isPollingDistrictLevel(areaType.level)}">
>182|     <!-- Special handling for exposing polling_district_type for functional tests -->
183|     <input type="hidden" class="context-polling-district-type"
184|         value="#{pollingDistrictController.getPollingDistrictTypeText(areaItem.pollingDistrict)}"/>
185|</c:if>

```

### Issue 87DF0738873C67B2CA6DEFE8FA055350:

Sink :

File admin-frontend/src/main/webapp/resources/widget/contextEditor.xhtml 200

```

197|         action="#{mvAreaPickerController.injectParentMvArea(areaType.level)}"
198|         update=":level#{areaType.level}createForm"
199|         rendered="#{mvAreaPickerController.getIsCreatable(areaType)}"/>
>200|         <!--oncomplete="PF('createAreaLevel#{areaType.level}Widget').show()";-->
201|         </h:panelGroup>
202|     </h:panelGroup>
203|</c:forEach>

```

## 6.118 createOrEditRoleDialog.xhtml

### Issue EA5483D1C343B5E697E9544FF0D54B66:

Sink :

File admin-frontend/src/main/webapp/secure/rbac/dialogs/createOrEditRoleDialog.xhtml 204

```

201|             styleClass="btn btn-xs btn-danger"/>
202|</div>
203|<h:panelGroup id="includedRolesTableContainer" styleClass="col-md-12" layout="block">
>204|     <!--@elvariable id="role" type="no.valg.eva.admin.common.rbac.Role"-->
205|     <p:dataTable id="includedRolesTable"
206|         pt:data-aft="inkluderteRollerTabell"
207|         rendered="#{not empty roleController.includedRolesList}"

```

## 6.119 dailyMarkOffCounts.xhtml

### Issue CCCB3068F604D216D37FC0F28FD02901:

Sink :

File admin-frontend/src/main/webapp/resources/counting/dailyMarkOffCounts.xhtml 27

```

24| <div class="row" data-counting-widget="dailyMarkOffCounts">
25|   <div class="col-md-12">
26|     <h4>#{msgs[cc.attrs.model.getPreviousTabTitle()]}</h4>
>27|   <!--@elvariable id="row" type="no.valg.eva.admin.frontend.counting.view.ModelRow"-->
28|   <p:dataTable value="#{cc.attrs.model}"
29|               var="row"
30|               id="dailyMarkOffCounts"

```

## 6.120 dailyMarkOffs.xhtml

### Issue B68B372575A402850AF7841FB41DEAD7:

Sink :

File admin-frontend/src/main/webapp/resources/counting/dailyMarkOffs.xhtml 25

```

22| </composite:interface>
23|
24| <composite:implementation>
>25|   <!-- DEV-NOTE: These needs to be placed inside this composite because of weird Ajax update behaviour -->
26|   <div data-counting-widget="dailyMarkOffs">
27|     <counting:dailyMarkOffsTable id="dailyMarkOffsTable"
28|                               caption="#{(cc.attrs.protocolCount.includeBallotCountFromOtherContests ?
msgs['@area_level[4].name'].concat(' ').concat(cc.attrs.counts.contestName).concat(' - ') :
'').concat(msgs['@count.votes.ordinaryCroostedInElectoralRo11'])}"

```

### Issue D4D3B168DD4E8A6440E8EE59489138BD:

Sink :

File admin-frontend/src/main/webapp/resources/counting/dailyMarkOffs.xhtml 34

```

31|           disabled="#{cc.attrs.disabled}"/>
32|
33| <!-- Added for markoffs for other contests -->
>34| <!-- Note: The label in the include below is hardcoded to assume "Bydelsvalg" when ballotCountForOtherContests is
set. Needs to be changed if this functionality is generalized -->
35| <counting:dailyMarkOffsTable id="dailyMarkOffsTableForOtherContests" 36|
rendered="#{cc.attrs.protocolCount.includeMarkOffsFromOtherContests}" 37|
caption="#{msgs['@config.boroughs.other']}"

```

### Issue F7EDA4CD7C02C7C11F6E0BBADE8DA62:

Sink :

File admin-frontend/src/main/webapp/resources/counting/dailyMarkOffs.xhtml 6-9

```

3|   xmlns:composite="http://java.sun.com/jsf/composite"
4|   xmlns:counting="http://java.sun.com/jsf/composite/counting">
5|
> 6| <!--
> 7| Component intended as a reusable view of daily markoffs used in a protocol count context
> 8| Prerequisites: The component must be inside a form to benefit from the form elements included. See
protocolCount.xhtml for example usage.
> 9| -->
10|
11| <composite:interface>
12|   <composite:attribute name="protocolCount"

```

## 6.121 dailyMarkOffsTable.xhtml

### Issue 7E9FEC05173A86B0DB8A414F5073BAD5:

Sink :

File admin-frontend/src/main/webapp/resources/counting/dailyMarkOffsTable.xhtml 33

```

30| </composite:interface>
31|
32| <composite:implementation>
>33|   <!--@elvariable id="dailyMarkOffCount" type="no.valg.eva.admin.common.counting.model.DailyMarkOffCount"-->
34|   <p:dataTable value="#{cc.attrs.dailyMarkOffCounts}"
35|               var="dailyMarkOffCount" id="dailyMarkOffCounts"
36|               tableStyleClass="table table-striped"

```

### Issue B71FoC86F785AEF666163585377AF8B1:

Sink :

File admin-frontend/src/main/webapp/resources/counting/dailyMarkOffsTable.xhtml 9-12



```

6|     xmlns:p="http://primefaces.org/ui"
7|     xmlns:pt="http://xmlns.jcp.org/jsf/passthrough">
8|
> 9|<!--
>10| Component intended as a reusable view of a single daily markoffs table used in a protocol count context
>11| Prerequisites: The component must be inside a form to benefit from the form elements included. See
protocolCount.xhtml for example usage.
>12| -->
13|
14|<composite:interface>
15|     <composite:attribute name="dailyMarkOffCounts"

```

## 6.122 electionCard.xhtml

**Issue F136EFCDB34DDB468AB27419558C6E50:**

**Sink :**

**File** admin-frontend/src/main/webapp/resources/config/electionCard.xhtml 28

```

25|<h:panelGroup id="listGroup" layout="block" styleClass="col-xs-3">
26|     <h:panelGroup layout="block" rendered="{not
localConfigurationController.electionCardCtrl.opptellingsvalgstyret}"> 27|
<h5><h:outputText value="{msgs['@voting.pollingPlaces']}" /></h5>
>28|     <!--@elvariable id="node" type="no.valg.eva.admin.frontend.configuration.models.ElectionCardModel"-->
29|         <p:tree
30|             id="municipalityTree"
31|             pt:data-aft="municipalityTree"

```

## 6.123 electionDayPollingPlaceView.xhtml

**Issue 5B192737DB8E12CFC57041592F22F4F1:**

**Sink :**

**File** admin-frontend/src/main/webapp/resources/config/electionDayPollingPlaceView.xhtml 28

```

25|<h:panelGroup id="listGroup" layout="block" styleClass="col-xs-3">
26|     <h5><h:outputText value="{msgs['@count.overview.pollingdistricts']}"
({electionDayPollingPlacesConfigurationController.pollingPlaceViewModelsSize})"
27|         styleClass="bold" /></h5>
>28|     <!--@elvariable id="node"
type="no.valg.eva.admin.frontend.configuration.models.ElectionDayPollingPlaceViewModel"-->
29|         <p:tree
30|             id="tree" 31|             pt:data-aft="tree"

```

## 6.124 footer.xhtml

**Issue B873EC4BEAA023B21392F30F67FD931B:**

**Sink :**

**File** admin-frontend/src/main/webapp/templates/footer.xhtml 12

```

10|<div class="page-footer">
11|
>12|     <!--<div class="row">-->
13|
14|     <div class="col-xs-8 left">
15|         <h:outputText value="{msgs['@footer.text']}" escape="false" />

```

**Issue F7A808D6F65EFCFB2A1AE700B6A51594:**

**Sink :**

**File** admin-frontend/src/main/webapp/templates/footer.xhtml 22

```

19|         EVA - v#{versionController.version} - #{versionController.hostId} 20|         </div>
21|
>22|     <!--</div>-->
23|
24| </div>
25|</div>

```

## 6.125 indexMenuItem.xhtml

**Issue 780A6021DB8BF374C0BD21D270AEA46D:**

**Sink :**

**File** admin-frontend/src/main/webapp/secure/includes/indexMenuItem.xhtml 9

```

6|   xmlns:ui="http://java.sun.com/jsf/facelets"
7|   contentType="text/html">
8|
> 9|<!--@elvariable id="menuItem" type="no.valg.eva.admin.frontend.common.menu.Menu"-->
10|
11|<ui:fragment rendered="#{menuItem.centralConfirmDialog}">
12|   <p:commandLink

```

## 6.126 lateValidationCovers.xhtml

**Issue** 4520CF84DCB594EC00BE182418CCBEF6:

**Sink :**

**File** admin-frontend/src/main/webapp/resources/counting/lateValidationCovers.xhtml 27

```

24|<div class="col-md-12">
25|   <h4>#{msgs['@count.lateValidation']}</h4>
26|   <span class="countCategory" style="display:none;"><h:outputText
value="#{cc.attrs.model.getCountCategory()}"></span>
>27|   <!--@elvariable id="row" type="no.valg.eva.admin.frontend.counting.view.ballotcount.BallotCountsModelRow"-->
28|   <p:dataTable value="#{cc.attrs.model}" var="row" id="lateValidationCoversModel"
29|               tableStyleClass="table table-striped" style="width: 50%; rowStyleClass="#{row.rowStyleClass}">
30|       <p:column headerText="">

```

## 6.127 listProposalTree.xhtml

**Issue** 2C2B116DCB2093DE5661F426D9A0B2DD:

**Sink :**

**File** admin-frontend/src/main/webapp/resources/config/listProposalTree.xhtml 25

```

22|<div class="col-xs-12">
23|   <div class="row">
24|     <h:panelGroup id="listGroup" layout="block" styleClass="col-xs-3">
>25|       <!--@elvariable id="node"
type="no.valg.eva.admin.common.configuration.model.local.ListProposalConfig"-->
26|       <p:tree
27|           id="tree"
28|           pt:data-aft="tree"

```

## 6.128 manntallsInfoVelgerLese.xhtml

**Issue** FDF8126CF0942CDB5D63D47284320FF8:

**Sink :**

**File** admin-frontend/src/main/webapp/resources/manntall/partials/manntallsInfoVelgerLese.xhtml 7

```

4|   xmlns:ui="http://java.sun.com/jsf/facelets"
5|   contentType="text/html">
6|
>7|<!--@elvariable id="velger" type="no.valg.eva.admin.configuration.domain.model.Voter"-->
8|
9|<div class="col-md-6">

```

## 6.129 manntallsInfoVelgerRedigere.xhtml

**Issue** D7AD03D34284AC54F09FB51C849FECEE:

**Sink :**

**File** admin-frontend/src/main/webapp/resources/manntall/partials/manntallsInfoVelgerRedigere.xhtml 119

```

117|<div class="form-group row required">
118|   <h:outputLabel styleClass="col-md-4" for="wPollingDistrictId"
value="#{msgs['@electoralRoll.pollingDistrict']}">
>119|   <!--@elvariable id="stemmekrets" type="no.valg.eva.admin.felles.valggeografi.model.StemmeKrets"-->
120|   <h:selectOneMenu
121|       id="wPollingDistrictId"
122|       pt:data-aft="pollingDistrictId"

```

**Issue** FDD3B58C187D08297C57569CB59E9505:

**Sink :**

File admin-frontend/src/main/webapp/resources/manntall/partials/manntallsInfoVelgerRedigere.xhtml 97

```

94|<ui:fragment rendered="#{!endreVelgerManntallController.velger.approved}">
95|  <div class="form-group row required">
96|    <h:outputLabel styleClass="col-md-4" value="#{msgs['@person.municipality']}" for="municipalityDropdown"/>
> 97|    <!--@elvariable id="kommune" type="no.valg.eva.admin.felles.valggeografi.model.Kommune"-->
98|    <h:selectOneMenu
99|      id="municipalityDropdown"
100|      pt:data-aft="municipalityId"

```

## 6.130 manntallshistorikk.xhtml

Issue C824A410FD9CF3FA6B8FBBE9F0439BFF:

Sink :

File admin-frontend/src/main/webapp/resources/manntall/dialogs/manntallshistorikk.xhtml 7

```

4|  xmlns:p="http://primefaces.org/ui"
5|  contentType="text/html">
6|
> 7|<!--@elvariable id="valist" type="no.evote.model.views.VoterAudit"-->
8|
9|<p:dialog
10|  id="manntallshistorikkDialog"

```

## 6.131 meldinger.xhtml

Issue A4C91769E5C4BBCDA8DBD8E60DoEDB53:

Sink :

File admin-frontend/src/main/webapp/resources/widget/meldinger.xhtml 13

```

11|<composite:implementation>
12|
>13|  <!--@elvariable id="melding" type="no.valg.eva.admin.felles.melding.Melding"-->
14|
15|  <div class="ui-messages ui-widget">
16|    <ui:fragment rendered="#{cc.attrs.meldingerWidget.getErrors().size() gt 0}">

```

## 6.132 openingHoursRead.xhtml

Issue 8E01FE347C302B3A8A376AF1ABEACAF1:

Sink :

File admin-frontend/src/main/webapp/resources/config/openingHoursRead.xhtml 15

```

13|<composite:implementation>
14|
>15|  <!--@elvariable id="electionDayViewModel"
type="no.valg.eva.admin.frontend.configuration.models.ElectionDayViewModel"-->
16|  <!--@elvariable id="openingHours" type="no.valg.eva.admin.common.configuration.model.OpeningHours"-->
17|  <ui:repeat id="list" var="electionDayViewModel" value="#{cc.attrs.electionDayViewModels}">
18|    <h:panelGroup styleClass="row" layout="block" rendered="#{electionDayViewModel.activated}">

```

Issue C890FoA563DE953FD3B2F34337C35C13:

Sink :

File admin-frontend/src/main/webapp/resources/config/openingHoursRead.xhtml 16

```

13|<composite:implementation>
14|
15|  <!--@elvariable id="electionDayViewModel"
type="no.valg.eva.admin.frontend.configuration.models.ElectionDayViewModel"-->
>16|  <!--@elvariable id="openingHours" type="no.valg.eva.admin.common.configuration.model.OpeningHours"-->
17|  <ui:repeat id="list" var="electionDayViewModel" value="#{cc.attrs.electionDayViewModels}">
18|    <h:panelGroup styleClass="row" layout="block" rendered="#{electionDayViewModel.activated}">
19|      <div class="date-text col-xs-3">

```

## 6.133 openingHoursWrite.xhtml

Issue 8E01FE347C302B3A8A376AF1ABEACAF2:

Sink :

File admin-frontend/src/main/webapp/resources/config/openingHoursWrite.xhtml 16

```

14|<composite:implementation>

```

```

15|
>16| <!--@elvariable id="electionDayViewModel"
type="no.valg.eva.admin.frontend.configuration.models.ElectionDayViewModel"-->
17| <!--@elvariable id="openingHours" type="no.valg.eva.admin.common.configuration.model.OpeningHours"--> 18|
<h:panelGroup id="opening-hours-write-container" styleClass="opening-hours opening-hours-write col-xs-12"
layout="block"
19| rendered="#{electionDayPollingPlacesConfigurationController.writeMode}">

```

**Issue C890FoA563DE953FD3B2F34337C35C14:****Sink :****File** admin-frontend/src/main/webapp/resources/config/openingHoursWrite.xhtml 17

```

14| <composite:implementation>
15|
16| <!--@elvariable id="electionDayViewModel"
type="no.valg.eva.admin.frontend.configuration.models.ElectionDayViewModel"-->
>17| <!--@elvariable id="openingHours" type="no.valg.eva.admin.common.configuration.model.OpeningHours"-->
18| <h:panelGroup id="opening-hours-write-container" styleClass="opening-hours opening-hours-write col-xs-12"
layout="block"
19| rendered="#{electionDayPollingPlacesConfigurationController.writeMode}">
20| <ui:repeat id="list"

```

**6.134 parentPollingDistrict.xhtml****Issue 6E69C545EF4A1A33D70B368CE55D6225:****Sink :****File** admin-frontend/src/main/webapp/resources/config/parentPollingDistrict.xhtml 55

```

53| <p>#{msgs['@config.local.parent_polling_district.childAddHelp']}</p>
54|
>55| <!--@elvariable id="pd" type="no.valg.eva.admin.common.configuration.model.local.RegularPollingDistrict"-->
56| <p:selectManyMenu
57| id="childListBox"
58| pt:data-aft="childListBox"

```

**6.135 pollingStation.xhtml****Issue 46BF01D2B6E790E5E026B19CECE9357A:****Sink :****File** admin-frontend/src/main/webapp/resources/config/pollingStation.xhtml 27

```

24| <h:panelGroup id="listGroup" layout="block" styleClass="col-xs-4">
25| <h5><h:outputText value="#{msgs['@config.local.polling_station.pollingPlaces']}"
({pollingStationConfigurationController.places.size()}" styleClass="bold"/></h5>
26| <!--@elvariable id="pd" type="no.valg.eva.admin.common.configuration.model.local.ElectionDayPollingPlace"-->
>27| <!--@elvariable id="pollingDistrict"
type="no.valg.eva.admin.common.configuration.model.local.ElectionDayPollingPlace"-->
28| <p:selectOneListbox
29| id="list-box"
30| pt:data-aft="selectPlace"

```

**Issue AE86CE87F66845C1238180290CBDFB06:****Sink :****File** admin-frontend/src/main/webapp/resources/config/pollingStation.xhtml 26

```

23| <div class="row">
24| <h:panelGroup id="listGroup" layout="block" styleClass="col-xs-4">
25| <h5><h:outputText value="#{msgs['@config.local.polling_station.pollingPlaces']}"
({pollingStationConfigurationController.places.size()}" styleClass="bold"/></h5>
>26| <!--@elvariable id="pd" type="no.valg.eva.admin.common.configuration.model.local.ElectionDayPollingPlace"-->
->
27| <!--@elvariable id="pollingDistrict"
type="no.valg.eva.admin.common.configuration.model.local.ElectionDayPollingPlace"--> 28| <p:selectOneListbox
29| id="list-box"

```

**6.136 protocolBallotCounts.xhtml****Issue E32A4233811390515DDDB4D18070BB30:****Sink :****File** admin-frontend/src/main/webapp/resources/counting/protocolBallotCounts.xhtml 18

```

15| </composite:interface>
16|
17| <composite:implementation>

```

```

18| <!-- DEV-NOTE: This needs to be placed inside this composite because of weird Ajax update behaviour -->
19| <div data-counting-widget="protocolBallotCounts">
20|   <p:dataTable value="#{protocolCountController.protocolBallotCounts}"
21|     var="count"

```

**Issue F7EDA4CD7C02C7C111F6E0BBADE8DA63:****Sink :****File** admin-frontend/src/main/webapp/resources/counting/protocolBallotCounts.xhtml 9-12

```

6|   xmlns:p="http://primefaces.org/ui"
7|   xmlns:pt="http://xmlns.jcp.org/jsf/passthrough">
8|
> 9|<!--
>10|Component intended as a reusable view of daily markoffs used in a protocol count context
>11|Prerequisites: The component must be inside a form to benefit from the form elements included. See
protocolCount.xhtml for example usage.
>12|-->
13|
14|<composite:interface>
15|</composite:interface>

```

**6.137 provingSamletResultat.xhtml****Issue 104C8A81915B14FE0CB5C27599DCD7A1:****Sink :****File** admin-frontend/src/main/webapp/resources/widget/provingSamletResultat.xhtml 174

```

171|   #{msgs['@voting.approveVoting.notHandled']} (#{cc.attrs.ctrl.pickListItems.size()})
#{msgs['@voting.approveVoting.numbers']})
172|</h3>
173|
>174|<!--@elvariable id="pickListItem" type="no.evote.dto.PickListItem"-->
175|<p:dataTable
176|   id="notApprovedVotingsDataTable"
177|   tableStyleClass="table table-striped table-condensed"

```

**6.138 provingSamletResultatTitle.xhtml****Issue F2AA051D8E476024E042AF7B8D44ED42:****Sink :****File** admin-frontend/src/main/webapp/resources/widget/partials/provingSamletResultatTitle.xhtml 7

```

4|   xmlns:ui="http://java.sun.com/jsf/facelets"
5|   contentType="text/html">
6|
> 7|<!--@elvariable id="provingSamletController"
type="no.valg.eva.admin.frontend.stemmegivning.ctrls.proving.ProvingSamletController"-->
8|
9|<h4>
10|   <ui:param name="hasVotingNumber" value="#{provingSamletController.form.startVotingNumber and
provingSamletController.form.endVotingNumber}"/>

```

**6.139 provingSamletSok.xhtml****Issue 3AC1E2C343F8C0213EA95DA226E54CC6:****Sink :****File** admin-frontend/src/main/webapp/resources/widget/provingSamletSok.xhtml 13

```

10|   <composite:attribute name="ctrl"
type="no.valg.eva.admin.frontend.stemmegivning.ctrls.proving.ProvingSamletController" required="true"/>
11|</composite:interface>
12|
>13|<!--@elvariable id="pollingPlaceVoting" type="no.evote.model.views.PollingPlaceVoting"-->
14|
15|<composite:implementation>

```

**6.140 registerModifiedBallotTemplate.xhtml****Issue 8FDE5C4D57DF3ABDC2441BE8624EE763:****Sink :**

File admin-frontend/src/main/webapp/secure/counting/templates/registerModifiedBallotTemplate.xhtml 14

```

11|<ui:define name="content">
12|
13|
>14| <!--@elvariable id="controller"
type="no.valg.eva.admin.frontend.counting.ctrls.RegisterModifiedBallotController"-->
15|
16| <f:event type="preRenderView" listener="#{controller.onPageUpdate}"/>

```

## 6.141 registrerOpptellinger.xhtml

Issue 5D77B1B3A38807BDCoDEFE6D1607C01B:

Sink :

File admin-frontend/src/main/webapp/secure/opptelling/registerOpptellinger.xhtml 18

```

15| <h:outputText value="#{msgs[registrerOpptellingerController.sideTittel]}"/>
16| </h1>
17|
>18| <!-- DEV-NOTE! Placeholder.xhtml. Bruker redirectes til startCount.xhtml etter valg i kontekstvelger. -->
19|
20| </ui:define>
21|</ui:composition>

```

## 6.142 reportMenuItem.xhtml

Issue 780A6021DB8BF374CoBD21D270AEA46C:

Sink :

File admin-frontend/src/main/webapp/secure/election/includes/reportMenuItem.xhtml 7

```

4| xmlns:pt="http://xmlns.jcp.org/jsf/passthrough"
5| contentType="text/html">
6|
>7|<!--@elvariable id="menuItem" type="no.valg.eva.admin.frontend.common.menu.Menu"-->
8|
9|<span style="white-space: nowrap;">
10| <h:selectBooleanCheckbox

```

Issue 780A6021DB8BF374CoBD21D270AEA46E:

Sink :

File admin-frontend/src/main/webapp/secure/reporting/includes/reportMenuItem.xhtml 8

```

5| xmlns:pt="http://xmlns.jcp.org/jsf/passthrough"
6| contentType="text/html">
7|
>8|<!--@elvariable id="menuItem" type="no.valg.eva.admin.frontend.common.menu.Menu"--> 9|
10|<p:commandLink id="report"
11| pt:data-aft="report_#{menuItem.source.rapportId}"

```

## 6.143 scanningOperatorList.xhtml

Issue 0FCF22EBB40A3169B6908DBF207E859B:

Sink :

File admin-frontend/src/main/webapp/resources/config/scanningOperatorList.xhtml 17

```

14|<composite:implementation>
15|
16| <h4>#{msgs['@config.local.scanning.search.operator.results']}</h4>
>17| <!--@elvariable id="operator" type="no.valg.eva.admin.rbac.domain.model.Operator"-->
18| <p:dataTable id="operatorSearchResultTable"
19| pt:data-aft="operatorSearchResultTable"
20| var="operator"

```

## 6.144 settlementSummary.xhtml

Issue C8EA1999C2643FEA4313E32D2D2EA59E:

Sink :

File admin-frontend/src/main/webapp/resources/widget/settlementSummary.xhtml 41

```

39|<div class="row row-table">
40| <div class="col-md-12">
>41| <!--@elvariable id="summaryRow" type="no.valg.eva.admin.frontend.settlement.SummaryRow"-->

```

```

42|         <p:dataTable id="settlementSummaryTable"
43|             var="ballotCountSummary"
44|             tableStyleClass="table"

```

## 6.145 stemmegivningForhand.xhtml

### Issue 1BDC2BC18AB611DF6441F28F525E944D:

Sink :

File admin-frontend/src/main/webapp/resources/widget/partials/stemmegivningForhand.xhtml 10

```

7|<!--@elvariable id="ctrl"
type="no.valg.eva.admin.frontend.stemmegivning.ctrls.registrering.ForhandRegistreringController"-->
8|<!--@elvariable id="sentInnkometDisabled" type="java.lang.Boolean"-->
9|<!--@elvariable id="stemmetypeDisabled" type="java.lang.Boolean"--
> 10|<!--@elvariable id="stemmetypeListe" type="java.util.List"-->
11|
12|<section class="panel-body">
13|     <div class="col-md-6">

```

### Issue 73B8DAC162E9D377F0B10A0D44E5FDE2:

Sink :

File admin-frontend/src/main/webapp/resources/widget/partials/stemmegivningForhand.xhtml 8

```

5|     contentType="text/html">
6|
7|<!--@elvariable id="ctrl"
type="no.valg.eva.admin.frontend.stemmegivning.ctrls.registrering.ForhandRegistreringController"-->
> 8|<!--@elvariable id="sentInnkometDisabled" type="java.lang.Boolean"-->
9|<!--@elvariable id="stemmetypeDisabled" type="java.lang.Boolean"-->
10|<!--@elvariable id="stemmetypeListe" type="java.util.List"-->

```

### Issue 89092F38973F21FC19950328FE5B33C3:

Sink :

File admin-frontend/src/main/webapp/resources/widget/partials/stemmegivningForhand.xhtml 7

```

4|     xmlns:pt="http://xmlns.jcp.org/jsf/passthrough"
5|     contentType="text/html">
6|
> 7|<!--@elvariable id="ctrl"
type="no.valg.eva.admin.frontend.stemmegivning.ctrls.registrering.ForhandRegistreringController"-->
8|<!--@elvariable id="sentInnkometDisabled" type="java.lang.Boolean"-->
9|<!--@elvariable id="stemmetypeDisabled" type="java.lang.Boolean"-->
10|<!--@elvariable id="stemmetypeListe" type="java.util.List"-->

```

### Issue A117CoA2C80BFAA7C36FFE11FC4E67C5:

Sink :

File admin-frontend/src/main/webapp/resources/widget/partials/stemmegivningForhand.xhtml 29

```

27|<h:panelGroup styleClass="col-md-6" layout="block" rendered="{ctrl.velgerEgenKommune}">
28|     <div class="form-group required">
>29|         <!--@elvariable id="stemmetype" type="no.valg.eva.admin.configuration.domain.model.VotingCategory"-->
30|         <h:outputLabel value="{msgs['@voting.votingCategory']}" for="selectedVotingCategoryId"
styleClass="labelfloat"/>
31|         <h:selectOneMenu
32|             id="selectedVotingCategoryId"

```

### Issue F87AD211B3A5472195117ED2948D82C3:

Sink :

File admin-frontend/src/main/webapp/resources/widget/partials/stemmegivningForhand.xhtml 9

```

7|<!--@elvariable id="ctrl"
type="no.valg.eva.admin.frontend.stemmegivning.ctrls.registrering.ForhandRegistreringController"-->
8|<!--@elvariable id="sentInnkometDisabled" type="java.lang.Boolean"-->
> 9|<!--@elvariable id="stemmetypeDisabled" type="java.lang.Boolean"--
> 10|<!--@elvariable id="stemmetypeListe" type="java.util.List"-->
11|
12|<section class="panel-body">

```

## 6.146 stemmegivningValgting.xhtml

### Issue 1BDC2BC18AB611DF6441F28F525E944E:

Sink :

File admin-frontend/src/main/webapp/resources/widget/partials/stemmegivningValgting.xhtml 10

```

8|<!--@elvariable id="ctrl"
type="no.valg.eva.admin.frontend.stemmegivning.ctrls.registrering.ValgtingRegistreringController"-->
9|<!--@elvariable id="stemmetypeDisabled" type="java.lang.Boolean"--
> >10|<!--@elvariable id="stemmetypeListe" type="java.util.List"-->
11|
12|<ui:fragment rendered="#{ctrl.stemmegivningsType.valgtingSentralt}">
13|   <section class="panel-body">

```

**Issue 617EC9E10E3F01EDB624A22E9810944A:****Sink :****File** admin-frontend/src/main/webapp/resources/widget/partials/stemmegivningValgting.xhtml 8

```

5|   xmlns:pt="http://xmlns.jcp.org/jsf/passthrough"
6|   contentType="text/html">
7|
> 8|<!--@elvariable id="ctrl"
type="no.valg.eva.admin.frontend.stemmegivning.ctrls.registrering.ValgtingRegistreringController"-->
9|<!--@elvariable id="stemmetypeDisabled" type="java.lang.Boolean"-->
10|<!--@elvariable id="stemmetypeListe" type="java.util.List"-->

```

**Issue A117CoA2C80BFAA7C36FFE11FC4E67C6:****Sink :****File** admin-frontend/src/main/webapp/resources/widget/partials/stemmegivningValgting.xhtml 16

```

13|<section class="panel-body">
14|   <div class="col-md-6">
15|     <div class="form-group required">
>16|       <!--@elvariable id="stemmetype" type="no.valg.eva.admin.configuration.domain.model.VotingCategory"-->
17|       <h:outputLabel value="#{msgs['@voting.votingCategory']}"
for="selectedVotingCategoryId" styleClass="label-float"/> 18|       <h:selectOneMenu
19|         id="selectedVotingCategoryId"

```

**Issue F87AD211B3A5472195117ED2948D82C4:****Sink :****File** admin-frontend/src/main/webapp/resources/widget/partials/stemmegivningValgting.xhtml 9

```

6|   contentType="text/html">
7|
8|<!--@elvariable id="ctrl"
type="no.valg.eva.admin.frontend.stemmegivning.ctrls.registrering.ValgtingRegistreringController"-->
> 9|<!--@elvariable id="stemmetypeDisabled" type="java.lang.Boolean"--
> 10|<!--@elvariable id="stemmetypeListe" type="java.util.List"-->
11|
12|<ui:fragment rendered="#{ctrl.stemmegivningsType.valgtingSentralt}">

```

**6.147 stemmestyre.xhtml****Issue 41AF72D5B7C61919AAF6E2965F2472BF:****Sink :****File** admin-frontend/src/main/webapp/resources/config/stemmestyre.xhtml 24

```

21|<h5>
22|   <h:outputText value="#{msgs['@count.overview.pollingdistricts']}"
({#stemmestyreConfigurationController.places.size()})" styleClass="bold"/> 23|</h5>
>24|<!--@elvariable id="pd" type="no.valg.eva.admin.common.configuration.model.local.PollingDistrict"--> 25|<!--
@elvariable id="pollingDistrict" type="no.valg.eva.admin.common.configuration.model.local.PollingDistrict"-->
26|<p>selectOneListbox
27|   id="list-box"

```

**Issue DE8E3505598DA771CEE1570950B684FD:****Sink :****File** admin-frontend/src/main/webapp/resources/config/stemmestyre.xhtml 25

```

22|   <h:outputText value="#{msgs['@count.overview.pollingdistricts']}"
({#stemmestyreConfigurationController.places.size()})" styleClass="bold"/> 23|</h5>
24|<!--@elvariable id="pd" type="no.valg.eva.admin.common.configuration.model.local.PollingDistrict"-->
>25|<!--@elvariable id="pollingDistrict" type="no.valg.eva.admin.common.configuration.model.local.PollingDistrict"-->
26|<p>selectOneListbox
27|   id="list-box"
28|   pt:data-aft="selectPollingDistrict"

```

**6.148 stemmestyreBorough.xhtml****Issue ADA5A374A750F6715D3D4C7BB7ACD82F:**



**Sink :**

File admin-frontend/src/main/webapp/resources/config/stemmestyreBorough.xhtml 24

```

21| <h:outputText value="{msgs['@count.overview.pollingdistricts']}"
(#{boroughsStemmestyreConfigurationController.places.size()}"
22|         styleClass="bold"/>
23|</h5>
>24|<!--@elvariable id="node" type="no.valg.eva.admin.frontend.configuration.models.PlaceTreeData"-->
25|<p:tree
26|     id="tree"
27|     pt:data-aft="tree"

```

**6.149 technicalPollingDistrict.xhtml****Issue 9E7967CFE798EB2B97E731D668A91D84:****Sink :**

File admin-frontend/src/main/webapp/resources/config/technicalPollingDistrict.xhtml 31

```

28|<div class="row">
29| <div class="col-xs-12">
30| <!--@elvariable id="pd"
type="no.valg.eva.admin.common.configuration.model.local.TechnicalPollingDistrict"-->
>31| <!--@elvariable id="pollingDistrict"
type="no.valg.eva.admin.common.configuration.model.local.TechnicalPollingDistrict"-->
32| <p:selectOneListBox
33|     id="list-box"
34|     pt:data-aft="selectPlace"

```

**Issue DDE4396EF78A683A2EB3E3D9A8B315F4:****Sink :**

File admin-frontend/src/main/webapp/resources/config/technicalPollingDistrict.xhtml 30

```

28|<div class="row">
29| <div class="col-xs-12">
>30| <!--@elvariable id="pd"
type="no.valg.eva.admin.common.configuration.model.local.TechnicalPollingDistrict"-->
31| <!--@elvariable id="pollingDistrict"
type="no.valg.eva.admin.common.configuration.model.local.TechnicalPollingDistrict"-->
32| <p:selectOneListBox
33|     id="list-box"

```

**6.150 updateApproveNegativeVoting.xhtml****Issue 1417744A6E4C2B89B4B11FEA0B65FDD4:****Sink :**

File admin-frontend/src/main/webapp/resources/widget/updateApproveNegativeVoting.xhtml 221

```

218|     value="Godkjenn stemmegivninger"
219|     icon="eva-icon-checkmark"
220|     styleClass="btn btn-sm btn-primary"/>
>221|<!--#{msgs['@voting.approveVoting.buttonUpdateApproveNegative']}<!-->
222|<!--rendered=#{ and cc.attrs.provider.votingStatistics.size() gt 0}"--> 223|
224|<p:commandButton

```

**Issue 1C33253E3E70DE27A369D1DDD98FAC53:****Sink :**

File admin-frontend/src/main/webapp/resources/widget/updateApproveNegativeVoting.xhtml 222

```

219|     icon="eva-icon-checkmark"
220|     styleClass="btn btn-sm btn-primary"/>
221|<!--#{msgs['@voting.approveVoting.buttonUpdateApproveNegative']}<!-->
>222|<!--rendered=#{ and cc.attrs.provider.votingStatistics.size() gt 0}"--> 223|
224|<p:commandButton
225|     id="back"

```

**6.151 voteToOtherMunicipalityConfirmDialog.xhtml****Issue 239B07BAF59C710C73DD14E0EE2A5E02:****Sink :**

File admin-frontend/src/main/webapp/secure/stemmegivning/includes/voteToOtherMunicipalityConfirmDialog.xhtml 8

```

5| xmlns:h="http://xmlns.jcp.org/jsf/html"

```

```

6|     contentType="text/html">
7|
> 8|<!--@elvariable id="ctrl"
type="no.valg.eva.admin.frontend.stemmegivning.ctrls.registrering.RegistreringController"-->
9|<h:form id="voteToOtherMunicipalityConfirmForm">
10|     <p:confirmDialog
11|         header="#{msgs['@voting.markOff.advanceInBallotBox.voteToOtherMunicipality.warningDialog.header']}"

```

## [Medium] System Information Leak / Incomplete Servlet Error Handling

*Encapsulation*

**Analysis:** Suspicious

### Abstract

If a Servlet fails to catch all exceptions, it might reveal debugging information that will help an adversary form a plan of attack.

### Explanation

When a Servlet throws an exception, the default error response the Servlet container sends back to the user typically includes debugging information. This information is of great value to an attacker. For example, a stack trace might show the attacker a malformed SQL query string, the type of database being used, and the version of the application container. This information enables the attacker to target known vulnerabilities in these components.

**Example 1:** In the following method a DNS lookup failure will cause the Servlet to throw an exception.

```

protected void doPost (HttpServletRequest req,
    HttpServletResponse res)
    throws IOException {
    String ip = req.getRemoteAddr();
    InetAddress addr = InetAddress.getByName(ip);
    ...
    out.println("hello " + addr.getHostName());
}

```

**Example 2:** The following method will throw a `NullPointerException` if the parameter "name" is not part of the request.

```

protected void doPost (HttpServletRequest req,
    HttpServletResponse res)
    throws IOException {
    String name =
    getParameter("name");
    out.println("hello " + name.trim());
}

```

### Recommendation

All top-level Servlet methods should catch `Throwable`, thereby minimizing the chance that the Servlet's error response mechanism is invoked.

**Example 3:** The method from Example 1 should be rewritten as follows:

```

protected void doPost (HttpServletRequest req,
    HttpServletResponse res) {
    try {
        String ip = req.getRemoteAddr();
        InetAddress addr =
        InetAddress.getByName(ip);
        ...
        out.println("hello " + addr.getHostName());
    } catch (Throwable t) {
        logger.error("caught throwable at top level", t);
    }
}

```

## 6.152 BackendStatusServlet.java

**Issue 20FA4D29CEF002A214FFADEE50C01650:**

**Sink :**

**File** admin-backend/src/main/java/no/valg/eva/admin/backend/web/BackendStatusServlet.java 33-35 **Fact**  
Name: no.valg.eva.admin.backend.web.BackendStatusServlet.doGet

```

32|@Override
>33|protected void doGet(final HttpServletRequest req, final HttpServletResponse resp) throws ServletException,
IOException {
>34|    sendJsonTilbake(resp, lagStatus());
>35|}
36|
37|@Override
38|protected void doPost(final HttpServletRequest req, final HttpServletResponse resp) throws ServletException,
IOException {

```

**Issue B3B3C93FoC7C9C164E19BE8696A23D96:**

**Sink :**

**File** admin-backend/src/main/java/no/valg/eva/admin/backend/web/BackendStatusServlet.java 38-40 **Fact**  
Name: no.valg.eva.admin.backend.web.BackendStatusServlet.doPost

```

37|@Override
>38|protected void doPost(final HttpServletRequest req, final HttpServletResponse resp) throws ServletException,
IOException {
>39|    sendJsonTilbake(resp, lagStatus());
>40|}
41|
42|private Properties lagStatus() {
43|    return statusApplicationService.getStatusAndConfiguredVersionProperties();

```

## 6.153 FrontendStatusServlet.java

**Issue AACBE2CFBDAB2C32DD55189A2C4B5FE3:**

**Sink :**

**File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/servlets/FrontendStatusServlet.java 31-33 **Fact**  
Name: no.valg.eva.admin.frontend.servlets.FrontendStatusServlet.doGet

```

30|@Override
>31|public void doGet(final HttpServletRequest req, final HttpServletResponse resp) throws IOException {
>32|    sendJsonTilbake(resp, statusPropertiesProvider.getStatusProperties());
>33|}
34|
35|@Override
36|protected void doPost(final HttpServletRequest req, final HttpServletResponse resp) throws ServletException,
IOException {

```

**Issue D10AF2FF74D49B6B8BED6ADD42E87E90:**

**Sink :**

**File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/servlets/FrontendStatusServlet.java 36-38 **Fact**  
Name: no.valg.eva.admin.frontend.servlets.FrontendStatusServlet.doPost

```

35|    @Override
>36|    protected void doPost(final HttpServletRequest req, final HttpServletResponse resp) throws ServletException,
IOException {
>37|        sendJsonTilbake(resp, statusPropertiesProvider.getStatusProperties());

```

## 6.154 FrontendStatusServletOld.java

**Issue 9C464E570B5E53953A9AB9216C6B736B:**

**Sink :**

**File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/servlets/FrontendStatusServletOld.java 28-35  
**Fact** Name: no.valg.eva.admin.frontend.servlets.FrontendStatusServletOld.doGet

```

25|    private StatusService statusService;
26|
27|    @Override
>28|    public void doGet(final HttpServletRequest req, final HttpServletResponse res) throws IOException { >29|
try {
>30|        res.getWriter().write(statusService.getStatus());

```

```

>31|         } catch (final IllegalStateException ex) {
>32|             LOG.fatal("Unable to access EJB.", ex);
>33|             res.getWriter().write(ERROR);

```

## 6.155 KlonEndeligeTellingervServlet.java

### Issue C8142CB851C88A779C49DDF056ED796A:

**Sink :**

**File** admin-backend/src/main/java/no/valg/eva/admin/backend/web/KlonEndeligeTellingervServlet.java 79-85

**Fact Name:** no.valg.eva.admin.backend.web.KlonEndeligeTellingervServlet.doPost

```

78|@Override
>79|protected void doPost(final HttpServletRequest req, final HttpServletResponse resp) throws ServletException,
IOException {
>80|    if (erTjenestenTilkoblet) {
>81|        klonEndeligeTellingerv(req, resp);
>82|    } else {
>83|        sendNotFoundResponse(resp, ADVARSEL_IKKE_TILGANG);
>84|    }
>85|}
86|
87|private void klonEndeligeTellingerv(HttpServletRequest req, HttpServletResponse resp) throws IOException {
88|    ValghendelseSti valghendelseSti = new ValghendelseSti(req.getParameter(PARAMETER_VALGHENDELSE_ID));

```

### Issue DEAE421344707A20E6AC4E8E76BA2AB4:

**Sink :**

**File** admin-backend/src/main/java/no/valg/eva/admin/backend/web/KlonEndeligeTellingervServlet.java 58-64

**Fact Name:** no.valg.eva.admin.backend.web.KlonEndeligeTellingervServlet.doGet

```

57|@Override
>58|protected void doGet(HttpServletRequest req, HttpServletResponse resp) throws ServletException, IOException {
>59|    if (erTjenestenTilkoblet) {
>60|        sendHtmlTilbake(resp, lagGeografiSpesifikasjonForm());
>61|    } else {
>62|        sendNotFoundResponse(resp, ADVARSEL_IKKE_TILGANG);
>63|    }
>64|}
65|
66|private String lagGeografiSpesifikasjonForm() {
67|    return "<form method=\"POST\">"

```

## 6.156 LagGeografiSpesifikasjonServlet.java

### Issue 44374CAF9E067AAD495832315EF05332:

**Sink :**

**File** admin-backend/src/main/java/no/valg/eva/admin/backend/web/LagGeografiSpesifikasjonServlet.java 74-80

**Fact Name:** no.valg.eva.admin.backend.web.LagGeografiSpesifikasjonServlet.doPost

```

73|@Override
>74|protected void doPost(final HttpServletRequest req, final HttpServletResponse resp) throws ServletException,
IOException {
>75|    if (erTjenestenTilkoblet) {
>76|        lagGeografiSpesifikasjon(req, resp);
>77|    } else {
>78|        sendNotFoundResponse(resp, ADVARSEL_IKKE_TILGANG);
>79|    }
>80|}
81|
82|private void lagGeografiSpesifikasjon(HttpServletRequest req, HttpServletResponse resp) throws IOException {
String parameterEksporttype = req.getParameter(PARAMETER_EKSPORTTYPE);

```

### Issue E4D55A616FF539C2032396352DB9CBA4:

**Sink :**

**File** admin-backend/src/main/java/no/valg/eva/admin/backend/web/LagGeografiSpesifikasjonServlet.java 52-58

**Fact Name:** no.valg.eva.admin.backend.web.LagGeografiSpesifikasjonServlet.doGet

```

51|@Override
>52|protected void doGet(final HttpServletRequest req, final HttpServletResponse resp) throws ServletException,
IOException {
>53|    if (erTjenestenTilkoblet) {
>54|        sendHtmlTilbake(resp, lagGeografiSpesifikasjonForm());
>55|    } else {
>56|        sendNotFoundResponse(resp, ADVARSEL_IKKE_TILGANG);
>57|    }
>58|}
59|
60|private String lagGeografiSpesifikasjonForm() {
61|    return "<form method=\"POST\">"

```

## 6.157 LogoutServlet.java

**Issue** D1B25B673B7D0BEC43FBCBB4969BA271:

**Sink :**

**File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/servlets/LogoutServlet.java 20-31

**Fact Name:** no.valg.eva.admin.frontend.servlets.LogoutServlet.doGet

```

17|    private static final TmpLoginDetector TMP_LOGIN_DETECTOR = new TmpLoginDetector();
18|
19|    @Override
>20|    public void doGet(final HttpServletRequest request, final HttpServletResponse response) throws IOException {
>21|        HttpSession session = request.getSession(false);
>22|        if (session != null) {
>23|            session.invalidate();
>24|        }
>25|
>26|        if (TMP_LOGIN_DETECTOR.isTmpLoginEnabled()) {
>27|            response.sendRedirect(ADMIN_LOGOUT_LOCATION);
>28|        } else {
>29|            response.sendRedirect(SAML_LOGOUT_LOCATION);

```

## 6.158 ReduserGeografiServlet.java

**Issue** 2BE906AD6EB9A598FAC63AC310FD8434:

**Sink :**

**File** admin-backend/src/main/java/no/valg/eva/admin/backend/web/ReduserGeografiServlet.java 68-74

**Fact Name:** no.valg.eva.admin.backend.web.ReduserGeografiServlet.doGet

```

67|@Override
>68|protected void doGet(final HttpServletRequest req, final HttpServletResponse resp) throws ServletException,
IOException {
>69|    if (erTjenestenTilkoblet) {
>70|        sendHtmlTilbake(resp, buildReduserGeografiForm());
>71|    } else {
>72|        sendNotFoundResponse(resp, ADVARSEL_INGEN_TILGANG);
>73|    }
>74|}
75|
76|private String buildReduserGeografiForm() {
77|    return "<form method=\"POST\">"

```

**Issue** B2AB9D54D870E4FAD8A5ECA6B03DAC8C:

**Sink :**

**File** admin-backend/src/main/java/no/valg/eva/admin/backend/web/ReduserGeografiServlet.java 90-96

**Fact Name:** no.valg.eva.admin.backend.web.ReduserGeografiServlet.doPost

```

89|@Override
>90|protected void doPost(final HttpServletRequest req, final HttpServletResponse resp) throws ServletException,
IOException {
>91|    if (erTjenestenTilkoblet) {
>92|        reduserGeografi(req, resp);
>93|    } else {
>94|        sendNotFoundResponse(resp, ADVARSEL_INGEN_TILGANG);
>95|    }
>96|}
97|
98|private void reduserGeografi(HttpServletRequest req, HttpServletResponse resp) throws IOException {
99|    String parameterGeografispesifikasjon = req.getParameter(PARAMETER_GEOGRAFISPESIFIKASJON);

```

## 6.159 ReportContentDownloadServlet.java

**Issue** A844B6138F50603F3E965E67286225FC:

**Sink :**

**File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/reporting/ReportContentDownloadServlet.java 47-67

**Fact Name:** no.valg.eva.admin.frontend.reporting.ReportContentDownloadServlet.doGet

```

46|@Override
>47|protected void doGet(HttpServletRequest req, HttpServletResponse resp) throws ServletException, IOException {
>48|    String requestId = req.getParameter("requestId");
>49|    if (StringUtils.isBlank(requestId)) {
>50|        resp.setStatus(SC_BAD_REQUEST);
>51|        return;
>52|    }
>53|    ReportExecution reportExecution = getCache().getIfPresent(requestId);
>54|    if (reportExecution != null) {
>55|        try (ServletOutputStream out = resp.getOutputStream();) {
>56|            resp.setContentType(MIME_TYPES.get(reportExecution.getFormat()));
>57|            resp.addHeader("Content-Disposition", getContentDisposition(reportExecution.getFormat()) + ";
filename=\"\" + reportExecution.getFileName() + \"\"");
>58|            byte[] content = reportExecution.getContent();
>59|            resp.setContentLength(content.length);
>60|            out.write(content);
>61|            LOG.debug(format("Served %s (%d bytes) from report content cache", reportExecution.getFileName(),
content.length)); >62|        }
>63|    } else {
>64|        resp.setStatus(SC_NOT_FOUND);
>65|        LOG.error(format("Invalid report file requestId: %s", requestId));
>66|    }
>67|}
68|
69|private String getContentDisposition(String format) {
70|    return "html".equals(format) ? "inline" : "attachment";

```

## 6.160 ReportTemplateUploadStatusServlet.java

**Issue** 76536B1C22D63813EC4C87BFA7744FF6:

**Sink :**

**File** admin-backend/src/main/java/no/valg/eva/admin/backend/reporting/jasperserver/ReportTemplateUploadStatusServlet.java 31-39

**Fact Name:** no.valg.eva.admin.backend.reporting.jasperserver.ReportTemplateUploadStatusServlet.doGet

```

30|@Override
>31|protected void doGet(final HttpServletRequest req, final HttpServletResponse resp) throws ServletException,
IOException {
>32|    Writer w = getOutputWriter(resp);
>33|
>34|    if (isReportTemplatesUploaded()) {
>35|        w.append(wrapInHtmlHeaders("Reports are ready"));
>36|    } else {
>37|        resp.sendError(RETURN_CODE_UNAVAILABLE, "Reports are unavailable");
>38|    }
>39|}
40|
41|private Writer getOutputWriter(HttpServletResponse resp) throws IOException {
42|    Writer w = resp.getWriter();

```

## 6.161 ResourceNotFoundServlet.java

**Issue** EA3937B5E8809D5070020F80F91E07BF:

**Sink :**

**File** admin-frontend/src/main/java/no/evote/presentation/exceptions/ResourceNotFoundServlet.java 13-16 **Fact**

**Name:** no.evote.presentation.exceptions.ResourceNotFoundServlet.doGet

```

12|    @Override
>13|    public void doGet(final HttpServletRequest req, final HttpServletResponse res) {
>14|        String message = req.getAttribute("javax.servlet.error.message").toString();
>15|        ErrorPageRenderer.renderError(req, res, ErrorPageRenderer.Error.NOT_FOUND, message, message);

```

## 6.162 SystemDateTimeServlet.java

**Issue** 717A61BDE829BFB213C648A9523FD856:

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/common/web/SystemDateTimeServlet.java 93-99 **Fact**

**Name:** no.valg.eva.admin.common.web.SystemDateTimeServlet.doPost

```

92|@Override
> 93|protected void doPost(final HttpServletRequest req, final HttpServletResponse resp) throws ServletException,
IOException {

```

```

> 94|     if (canChangeTime()) {
> 95|         handleSettingOfDateTime(req, resp);
> 96|     } else {
> 97|         sendNotFoundResponse(resp);
> 98|     }
> 99| } 100|
101|private void handleSettingOfDateTime(HttpServletRequest req, HttpServletResponse resp) throws IOException { 102|
    Writer w = getOutputWriter(resp);

```

**Issue EC66A9C63F949B49EB16F067DB7FCD23:****Sink :****File** admin-common/src/main/java/no/valg/eva/admin/common/web/SystemDateTimeServlet.java 46-54**Fact** Name: no.valg.eva.admin.common.web.SystemDateTimeServlet.doGet

```

45|@Override
>46|protected void doGet(final HttpServletRequest req, final HttpServletResponse resp) throws ServletException,
IOException {
>47|     Writer w = getOutputWriter(resp);
>48|
>49|     if (canChangeTime()) {
>50|         w.append(buildSetDateTimeFormHtml());
>51|     } else {
>52|         sendNotFoundResponse(resp);
>53|     }
>54| }
55|
56|private Writer getOutputWriter(HttpServletResponse resp) throws IOException {
57|     Writer w = resp.getWriter();

```

**6.163 SystemPasswordServlet.java****Issue 0B1CD617AC095B7BC7A6B2513A7100A4:****Sink :****File** admin-backend/src/main/java/no/valg/eva/admin/backend/web/SystemPasswordServlet.java 28-35**Fact** Name: no.valg.eva.admin.backend.web.SystemPasswordServlet.doGet

```

27|@Override
>28|protected void doGet(final HttpServletRequest req, final HttpServletResponse resp) throws ServletException,
IOException {
>29|     if (!systemPasswordApplicationService.isPasswordSet()) {
>30|         sendHtmlTilbake(resp, "<form method=\"POST\">" + "System password: <input type=\"text\" name=\"password\"
/>" + "<br/>"
>31|             + "<input type=\"submit\" value=\"Set password\"/>" + "</form>");
>32|     } else {
>33|         sendHtmlTilbake(resp, "Password has already been set");
>34|     }
>35| }
36|
37|@Override
38|protected void doPost(final HttpServletRequest req, final HttpServletResponse resp) throws ServletException,
IOException {

```

**Issue 515ABE5F16AF34C8358A5F2A1CC62B67:****Sink :****File** admin-backend/src/main/java/no/valg/eva/admin/backend/web/SystemPasswordServlet.java 38-48**Fact** Name: no.valg.eva.admin.backend.web.SystemPasswordServlet.doPost

```

37|@Override
>38|protected void doPost(final HttpServletRequest req, final HttpServletResponse resp) throws ServletException,
IOException {
>39|     String password = req.getParameter("password");
>40|     if (!systemPasswordApplicationService.isPasswordSet() && systemPasswordIsCorrect(password)) {
>41|         systemPasswordApplicationService.setSystemPassword(password);
>42|         sendHtmlTilbake(resp, "Password is set");
>43|     } else if (!systemPasswordApplicationService.isPasswordSet() && !systemPasswordIsCorrect(password)) {
>44|         sendHtmlTilbake(resp, "The entered password was wrong!");
>45|     } else if (systemPasswordApplicationService.isPasswordSet()) {
>46|         sendHtmlTilbake(resp, "Password has already been set");
>47|     }
>48| }
49|
50|private boolean systemPasswordIsCorrect(final String password) {
51|     return systemPasswordApplicationService.isPasswordCorrect(password);

```

**6.164 TaskLoggerServlet.java****Issue 2257335BB42CoAD3D18B9D4DE5E3EACA:**

**Sink :****File** admin-backend/src/main/java/no/evote/service/util/TaskLoggerServlet.java 28-43**Fact Name:** no.evote.service.util.TaskLoggerServlet.doGet

```

25|     * @see HttpServlet#doGet(HttpServletRequest request, HttpServletResponse response)
26|     */
27|     @Override
>28|     protected void doGet(final HttpServletRequest request, final HttpServletResponse response) throws
ServletException, IOException {
>29|         response.setContentType("text/plain");
>30|         Writer writer = response.getWriter();
>31|
>32|         Map<String, DateTime> log = taskLogger.getLog();
>33|         if (!log.isEmpty()) {
>34|             writer.write("The following tasks have been logged: \n");
>35|             for (Map.Entry<String, DateTime> entry : log.entrySet()) {
>36|                 writer.write(entry.getValue() + ": " + entry.getKey() + "\n"); >37|             }
>38|         } else {
>39|             writer.write("No tasks have been logged yet. \n");
>40|         }
>41|
>42|         writer.flush();

```

## 6.165 TemplateUploadTriggerServlet.java

**Issue** A35ED936A9DABC178AE396D7F453E116:**Sink :****File** admin-backend/src/main/java/no/valg/eva/admin/backend/reporting/jasperserver/TemplateUploadTriggerServlet.java 22-29**Fact Name:** no.valg.eva.admin.backend.reporting.jasperserver.TemplateUploadTriggerServlet.doGet

```

19|     private Event<UploadReportTemplatesEvent> uploadReportTemplatesEventEvent;
20|
21|     @Override
>22|     protected void doGet(HttpServletRequest req, HttpServletResponse resp) throws ServletException, IOException {
>23|         if
(Boolean.valueOf(EvoteProperties.getProperty(EvoteProperties.JASPERSERVER_ENABLE_REPORT_TEMPLATES_UPLOAD_TRIGGER_SERVL
ET, "false"))) {
>24|             uploadReportTemplatesEventEvent.fire(new UploadReportTemplatesEvent());
>25|             Writer w = resp.getWriter();
>26|             resp.setContentType("text/html");
>27|             w.append("Uploading report templates...\n");

```

## 6.166 TmpLoginServlet.java

**Issue** 3AD0345B6746768ABA7986D077E5656D:**Sink :****File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/servlets/TmpLoginServlet.java 37-46**Fact Name:** no.valg.eva.admin.frontend.servlets.TmpLoginServlet.doPost

```

36| @Override
>37| protected void doPost(final HttpServletRequest req, final HttpServletResponse resp) throws IOException {
>38|
>39|     TmpLoginForm form = new TmpLoginForm(
>40|         req.getParameter("username"),
>41|         SecurityLevel.fromLevel(Integer.parseInt(req.getParameter("secLevel"))),
>42|         "true".equals(req.getParameter("scanning")));
>43|     req.getSession().setAttribute(TmpLoginForm.class.getName(), form);
>44|
>45|     resp.sendRedirect("/secure/selectRole.xhtml");
>46| }
47|
48| private String getFormAction(HttpServletRequest request) {
49|     boolean scanning = "true".equals(request.getParameter("scanning"));

```

**Issue** 8D93534A2B725168A558C9626E8AF700:**Sink :****File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/servlets/TmpLoginServlet.java 19-34**Fact Name:** no.valg.eva.admin.frontend.servlets.TmpLoginServlet.doGet



```

16| public class TmpLoginServlet extends HttpServlet {
17|
18|     @Override
>19|     protected void doGet(final HttpServletRequest req, final HttpServletResponse resp) throws IOException {
>20|         if (getTmpLoginDetector().isTmpLoginEnabled()) {
>21|             invalidateSessionIfExists(req);
>22|
>23|             Writer w = resp.getWriter();
>24|             resp.setContentType("text/html");
>25|
>26|             w.append("<html>" + "<head><link type=\"text/css\"
href=\"/javax.faces.resource/all.css.xhtml?ln=css&rv=4.0%20(90e94bd)\" "
>27|                 + "rel=\"stylesheet\"></head>" + "<body id=\"page-tmplogin\"> <form method=\"POST\" action=\""
+ getFormAction(req) + "\">"
>28|                 + "Bruker ID: <input type=\"text\" name=\"username\" autofocus=\"autofocus\"/> " + "<br/>"
>29|                 + "Security level: <input type=\"text\" name=\"secLevel\" value=\"3\"/> " + "<br/>" + "<input
type=\"submit\" value=\"Login\"/>"
>30|                 + "</form>" + "</body>" + "</html>");
>31|         } else {
>32|             resp.sendError(HttpServletResponse.SC_NOT_FOUND);
>33|         }
>34|     }
35|
36|     @Override
37|     protected void doPost(final HttpServletRequest req, final HttpServletResponse resp) throws IOException {

```

## [Medium] Unreleased Resource / Streams

Code Quality

**Analysis:** Reliability Issue

### Abstract

The program can potentially fail to release a system resource.

### Explanation

The program can potentially fail to release a system resource.

Resource leaks have at least two common causes:

- Error conditions and other exceptional circumstances.
- Confusion over which part of the program is responsible for releasing the resource.

Most unreleased resource issues result in general software reliability problems, but if an attacker can intentionally trigger a resource leak, the attacker may be able to launch a denial of service attack by depleting the resource pool.

**Example:** The following method never closes the file handle it opens. The `finalize()` method for `FileInputStream` eventually calls `close()`, but there is no guarantee as to how long it will take before the `finalize()` method will be invoked. In a busy environment, this can result in the JVM using up all of its file handles.

```

private void processFile(String fName) throws FileNotFoundException, IOException {
    FileInputStream fis = new FileInputStream(fName);
    int sz;
    byte[] byteArray = new byte[BLOCK_SIZE];
    while ((sz = fis.read(byteArray)) != -1) {
        processBytes(byteArray, sz);
    }
}

```

### Recommendation

1. Never rely on `finalize()` to reclaim resources. In order for an object's `finalize()` method to be invoked, the garbage collector must determine that the object is eligible for garbage collection. Because the garbage collector is not required to run unless the JVM is low on memory, there is no guarantee that an object's `finalize()` method will be invoked in an expedient fashion. When the

garbage collector finally does run, it may cause a large number of resources to be reclaimed in a short period of time, which can lead to "bursty" performance and lower overall system throughput. This effect becomes more pronounced as the load on the system increases.

Finally, if it is possible for a resource reclamation operation to hang (if it requires communicating over a network to a database, for example), then the thread that is executing the `finalize()` method will hang.

2. Release resources in a `finally` block. The code for the Example should be rewritten as follows:

```
public void processFile(String fName) throws FileNotFoundException, IOException {
    FileInputStream fis;
    try {
        fis = new FileInputStream(fName);
    }
    int sz;
    byte[] byteArray = new byte[BLOCK_SIZE];
    while ((sz = fis.read(byteArray)) != -1) {
        processBytes(byteArray, sz);
    }
    finally {
        if (fis != null) {
            safeClose(fis);
        }
    }
}

public static void safeClose(FileInputStream fis)
{
    if (fis != null) {
        try {
            fis.close();
        } catch (IOException e) {
            log(e);
        }
    }
}
```

This solution uses a helper function to log the exceptions that might occur when trying to close the stream. Presumably this helper function will be reused whenever a stream needs to be closed.

Also, the `processFile` method does not initialize the `fis` object to null. Instead, it checks to ensure that `fis` is not null before calling `safeClose()`. Without the null check, the Java compiler reports that `fis` might not be initialized. This choice takes advantage of Java's ability to detect uninitialized variables. If `fis` is initialized to null in a more complex method, cases in which `fis` is used without being initialized will not be detected by the compiler.

## 6.167 AntiSamyUtil.java

**Issue** 3AB04E6B164B5521F9FD7F594DD7319C:

**Source :**

**File** admin-common/src/main/java/no/evote/util/AntiSamyUtil.java 28

**Action** BranchTaken Branch taken: (policyFile == null)

```
25|private AntiSamyUtil() {
26|    try {
27|        String policyFile =
EvoteProperties.getProperty(EvoteProperties.NO_EVOTE_UTIL_ANTI_SAMY_FILTER_POLICY_FILE, true);
>28|        if (policyFile == null) {
29|            policy =
Policy.getInstance(this.getClass().getClassLoader().getResourceAsStream(POLICY_FILE_LOCATION));
30|            LOG.info("Using built-in policy file");
31|        } else {
```

**Sink :**

**File** admin-common/src/main/java/no/evote/util/AntiSamyUtil.java 29

**Action** Assign getResourceAsStream(...)

**Rule id** 74714BFC-EDF7-445B-8672-0996214D5845

```
26|try {
27|    String policyFile = EvoteProperties.getProperty(EvoteProperties.NO_EVOTE_UTIL_ANTI_SAMY_FILTER_POLICY_FILE,
true);
28|    if (policyFile == null) {
>29|        policy = Policy.getInstance(this.getClass().getClassLoader().getResourceAsStream(POLICY_FILE_LOCATION));
```

```
30|     LOG.info("Using built-in policy file");
31|   } else {
32|     policy = Policy.getInstance(new FileInputStream(policyFile));
```

**Issue DAC03675F08BoC161BFA9BCD2147A4EE:****Source :**

File admin-common/src/main/java/no/evote/util/AntiSamyUtil.java 28

Action BranchNotTaken Branch not taken: (policyFile != null)

```
25|private AntiSamyUtil() {
26|    try {
27|        String policyFile =
EvoteProperties.getProperty(EvoteProperties.NO_EVOTE_UTIL_ANTI_SAMY_FILTER_POLICY_FILE, true);
>28|        if (policyFile == null) {
29|            policy =
Policy.getInstance(this.getClass().getClassLoader().getResourceAsStream(POLICY_FILE_LOCATION));
30|            LOG.info("Using built-in policy file");
31|        } else {
```

**Sink :**

File admin-common/src/main/java/no/evote/util/AntiSamyUtil.java 32

Action InCall new FileInputStream(...)

Rule id 74714BFC-EDF7-445B-8672-0996214D5845

```
29|     policy = Policy.getInstance(this.getClass().getClassLoader().getResourceAsStream(POLICY_FILE_LOCATION));
30|     LOG.info("Using built-in policy file");
31|   } else {
>32|     policy = Policy.getInstance(new FileInputStream(policyFile));
33|     LOG.info("Using policy file from properties: " + policyFile);
34|   }
35|} catch (PolicyException | FileNotFoundException e) {
```

**6.168 ErrorPageRenderer.java****Issue C4DDD67A08B9E00EB2954396845E9A1E:****Source :**

File admin-frontend/src/main/java/no/evote/presentation/exceptions/ErrorPageRenderer.java 122 Action Assign is = getResourceAsStream(...)

Rule id 74714BFC-EDF7-445B-8672-0996214D5845

```
120|private static String getTemplate(Error error, String incidentCode) throws IOException { 121|
>122|    InputStream is = ErrorPageRenderer.class.getClassLoader().getResourceAsStream("errorpage.html");
123|    String title = error.getTitle();
124|    String message = error.getMessage();
125|    int statusCode = error.getCode();
```

**Sink :**

File admin-frontend/src/main/java/no/evote/presentation/exceptions/ErrorPageRenderer.java 122

Action Assign is = getResourceAsStream(...)

Rule id 74714BFC-EDF7-445B-8672-0996214D5845

```
120|private static String getTemplate(Error error, String incidentCode) throws IOException { 121|
>122|    InputStream is = ErrorPageRenderer.class.getClassLoader().getResourceAsStream("errorpage.html");
123|    String title = error.getTitle();
124|    String message = error.getMessage();
125|    int statusCode = error.getCode();
```

**6.169 EvaAdminServlet.java****Issue 0AB7CCEBB85D914410F69E8A79A4B6FB:****Source :**

File admin-common/src/main/java/no/valg/eva/admin/common/web/EvaAdminServlet.java 21

Action Assign w = getOutputWriter(...)

Rule id 74714BFC-EDF7-445B-8672-0996214D5845

```
18|protected abstract Logger getLogger();
19|
20|protected void sendHtmlTilbake(HttpServletResponse resp, String htmlBody) throws IOException {
>21|    Writer w = getOutputWriter(resp);
22|    w.append(wrapInHtmlHeaders(htmlBody));
```

**Sink :**

File admin-common/src/main/java/no/valg/eva/admin/common/web/EvaAdminServlet.java 21

Action Assign w = getOutputWriter(...)

Rule id 74714BFC-EDF7-445B-8672-0996214D5845

```

18|protected abstract Logger getLogger();
19|
20|protected void sendHtmlTilbake(HttpServletResponse resp, String htmlBody) throws IOException {
>21|     Writer w = getOutputWriter(resp);
22|     w.append(wrapInHtmlHeaders(htmlBody));

```

## 6.170 EvoteProperties.java

### Issue F2E5347A775E6B4C95DDC7936F73EA55:

#### Source :

File admin-common/src/main/java/no/evote/util/EvoteProperties.java 91

Action BranchNotTaken Branch not taken: (propertyFilePath != null)

```

88|loadPropertyFilePath();
89|
90|try {
>91|     if (propertyFilePath == null) {
92|         throw new IllegalArgumentException(EVOTE_PROPERTIES_ENV_VARIABLE + " is not defined"); 93|     } else {
94|         FileInputStream fi = new FileInputStream(propertyFilePath);

```

#### Sink :

File admin-common/src/main/java/no/evote/util/EvoteProperties.java 94

Action InCall fi = new FileInputStream(...)

Rule id 74714BFC-EDF7-445B-8672-0996214D5845

```

91|if (propertyFilePath == null) {
92|     throw new IllegalArgumentException(EVOTE_PROPERTIES_ENV_VARIABLE + " is not defined");
93|} else {
>94|     FileInputStream fi = new FileInputStream(propertyFilePath);
95|     properties.load(fi);
96|     fi.close();

```

## 6.171 ExportServiceBean.java

### Issue 2F056B36001B4131099FE2F112F6D138:

#### Source :

File admin-backend/src/main/java/no/evote/service/ExportServiceBean.java 511

Action Assign is = getResourceAsStream(...)

Rule id 74714BFC-EDF7-445B-8672-0996214D5845

```

509|private byte[] xslTransform(final String xslFilename, final Document doc) {
510|     InputStream is;
>511|     is = this.getClass().getClassLoader().getResourceAsStream("xsl" + File.separator + xslFilename);
512|
513|     if (is == null) {
514|         throw new EvoteException("Unable to find file: " + xslFilename);

```

#### Sink :

File admin-backend/src/main/java/no/evote/service/ExportServiceBean.java 511

Action Assign is = getResourceAsStream(...)

Rule id 74714BFC-EDF7-445B-8672-0996214D5845

```

509|private byte[] xslTransform(final String xslFilename, final Document doc) {
510|     InputStream is;
>511|     is = this.getClass().getClassLoader().getResourceAsStream("xsl" + File.separator + xslFilename);
512|
513|     if (is == null) {
514|         throw new EvoteException("Unable to find file: " + xslFilename);

```

## 6.172 FtpPregeneratedContentRetriever.java

### Issue BC3B6D82D1BAE391BFE398251BA5BE1A:

#### Source :

File admin-backend/src/main/java/no/valg/eva/admin/backend/reporting/jasperserver/api/FtpPregeneratedContentRetriever.java 38

Action Assign getInputStream()

Rule id 74714BFC-EDF7-445B-8672-0996214D5845

```

35|public byte[] tryPreGeneratedReport(String fileName) {
36|     try {
37|         URLConnection ftpConnection = new URL(ftpUrlPrefix + fileName).openConnection();
>38|         return toByteArray(ftpConnection.getInputStream());
39|     } catch (MalformedURLException e) {
40|         LOGGER.error(e.getMessage(), e);
41|     } catch (IOException e) {

```

#### Sink :

File admin-backend/src/main/java/no/valg/eva/admin/backend/reporting/jasperserver/api/FtpPregeneratedContentRetriever.java 38

Action Assign getInputStream()

Rule id 74714BFC-EDF7-445B-8672-0996214D5845

```

35|public byte[] tryPreGeneratedReport(String fileName) {
36|     try {

```

```

37|         URLConnection ftpConnection = new URL(ftpUrlPrefix + fileName).openConnection(); >38|         return
toByteArray(ftpConnection.getInputStream());
39|     } catch (MalformedURLException e) {
40|         LOGGER.error(e.getMessage(), e);
41|     } catch (IOException e) {

```

## 6.173 GenericCacheManager.java

### Issue oFD1EB1E5FBE993013DF4E1E474BAB23:

#### Source :

File admin-frontend/src/main/java/no/evote/presentation/cache/GenericCacheManager.java 30 Action Assign getResourceAsStream(...)

Rule id 74714BFC-EDF7-445B-8672-0996214D5845

```

27|     LOG.debug("init()");
28|
29|     LOG.debug("Bruker lokal cache (eneste typen støttet p.t.");
>30|     CacheManager.create(getClass().getClassLoader().getResourceAsStream("ehcache-local.xml")); 31|     manager =
CacheManager.getInstance();

```

#### Sink :

File admin-frontend/src/main/java/no/evote/presentation/cache/GenericCacheManager.java 30 Action Assign getResourceAsStream(...)

Rule id 74714BFC-EDF7-445B-8672-0996214D5845

```

27|     LOG.debug("init()");
28|
29|     LOG.debug("Bruker lokal cache (eneste typen støttet p.t.");
>30|     CacheManager.create(getClass().getClassLoader().getResourceAsStream("ehcache-local.xml")); 31|     manager =
CacheManager.getInstance();

```

## 6.174 IOUtil.java

### Issue 7DB37C785350A04975BE61A93FBD3FBC:

#### Source :

File admin-common/src/main/java/no/evote/util/IOUtil.java 52

Action InCall inputStream = new FileInputStream(...)

Rule id 74714BFC-EDF7-445B-8672-0996214D5845

```

51|public static byte[] getBytes(final File file) throws IOException { >52|     FileInputStream inputStream = new
FileInputStream(file);
53|     byte[] bytes = getBytes(inputStream);
54|     inputStream.close();
55|     return bytes;

```

#### Sink :

File admin-common/src/main/java/no/evote/util/IOUtil.java 52

Action InCall inputStream = new FileInputStream(...)

Rule id 74714BFC-EDF7-445B-8672-0996214D5845

```

51|public static byte[] getBytes(final File file) throws IOException { >52|     FileInputStream inputStream = new
FileInputStream(file);
53|     byte[] bytes = getBytes(inputStream);
54|     inputStream.close();
55|     return bytes;

```

## 6.175 IdPortenCredentialProducer.java

### Issue AD29030E0005FC97DFAB0A075CA81E4E:

#### Source :

File admin-rbac/src/main/java/no/valg/eva/admin/rbac/service/impl/IdPortenCredentialProducer.java 62

Action InCall metaDataInputStream = new FileInputStream(...)

Rule id 74714BFC-EDF7-445B-8672-0996214D5845

```

59|BasicParserPool ppMgr = new BasicParserPool();
60|ppMgr.setNamespaceAware(true);
61|
>62|InputStream metaDataInputStream = new
FileInputStream(EvoteProperties.getProperty(EvoteProperties.SAML_META_DATA_IDP_META));
63|Document metaDataDocument = ppMgr.parse(metaDataInputStream);
64|Element metaDataRoot = metaDataDocument.getDocumentElement();
65|metaDataInputStream.close();

```

**Sink :****File** admin-rbac/src/main/java/no/valg/eva/admin/rbac/service/impl/IdPortenCredentialProducer.java 62**Action** InCall metaDataInputStream = new FileInputStream(...)**Rule id** 74714BFC-EDF7-445B-8672-0996214D5845

```

59|BasicParserPool ppMgr = new BasicParserPool();
60|ppMgr.setNamespaceAware(true);
61|
>62|InputStream metaDataInputStream = new
FileInputStream(EvoteProperties.getProperty(EvoteProperties.SAML_META_DATA_IDP_META));
63|Document metaDataDocument = ppMgr.parse(metaDataInputStream);
64|Element metaDataRoot = metaDataDocument.getDocumentElement();
65|metaDataInputStream.close();

```

## 6.176 ImportElectoralRollUtil.java

**Issue** 2BCE4ABE23CEE6477A5E15B6A783FDB9:**Source :****File** admin-common/src/main/java/no/evote/util/ImportElectoralRollUtil.java 77**Action** BranchTaken Branch taken: (!isEmpty(filePath))

```

74|public static BufferedReader getBufferedReaderForFile(final String filePath) { 75|   InputStreamReader isr;
76|   FileInputStream fis;
>77|   if (!StringUtils.isEmpty(filePath)) {
78|       File file = new File(filePath);
79|       if (file.exists()) {
80|           try {

```

**Sink :****File** admin-common/src/main/java/no/evote/util/ImportElectoralRollUtil.java 82**Action** InCall isr = new InputStreamReader(fis, ...)**Rule id** 74714BFC-EDF7-445B-8672-0996214D5845

```

79|if (file.exists()) {
80|    try {
81|        fis = new FileInputStream(file);
>82|        isr = new InputStreamReader(fis, ISO_8859_1);
83|        LOGGER.info("FILE: " + isr.getEncoding());
84|        return new BufferedReader(isr);
85|    } catch (FileNotFoundException | UnsupportedEncodingException e) {

```

## 6.177 ManntallsimportMappingRepository.java

**Issue** 74851C920CCoCBF693DE2C56741C884A:**Source :****File** admin-configuration/src/main/java/no/valg/eva/admin/configuration/repository/ManntallsimportMappingRepository.java 60 **Action****InCall** new FileReader(...)**Rule id** 74714BFC-EDF7-445B-8672-0996214D5845

```

58|private ManntallsimportMapping lesMappingFraFil(File mappingfil) {
59|    try {
>60|        return new Gson().fromJson(new FileReader(mappingfil), ManntallsimportMapping.class);
61|    } catch (FileNotFoundException e) {
62|        throw new EvoteException("Mappingfil ble ikke funnet - rett etter at vi sjekket om den eksisterte. Dette
skal ikke kunne skje", e);

```

**Sink :****File** admin-configuration/src/main/java/no/valg/eva/admin/configuration/repository/ManntallsimportMappingRepository.java 60 **Action****InCall** new FileReader(...)**Rule id** 74714BFC-EDF7-445B-8672-0996214D5845

```

58|private ManntallsimportMapping lesMappingFraFil(File mappingfil) {
59|    try {
>60|        return new Gson().fromJson(new FileReader(mappingfil), ManntallsimportMapping.class);
61|    } catch (FileNotFoundException e) {
62|        throw new EvoteException("Mappingfil ble ikke funnet - rett etter at vi sjekket om den eksisterte. Dette
skal ikke kunne skje", e);

```

**Issue** 8BB7BF76DC21625AE8AC56B992D0280E:**Source :****File** admin-configuration/src/main/java/no/valg/eva/admin/configuration/repository/ManntallsimportMappingRepository.java 67 **Action****Assign** getResourceAsStream(...)**Rule id** 74714BFC-EDF7-445B-8672-0996214D5845

```

66|private ManntallsimportMapping lesMappingFraClasspath() {
>67|   InputStreamReader inputStreamReader = new
InputStreamReader(this.getClass().getClassLoader().getResourceAsStream(filnavn));
68|   return new Gson().fromJson(inputStreamReader, ManntallsimportMapping.class);

```

**Sink :**

**File** admin-configuration/src/main/java/no/valg/eva/admin/configuration/repository/ManntallsimportMappingRepository.java 67 **Action** InCall inputStreamReader = new InputStreamReader(this.getClass().getClassLoader().getResourceAsStream(this.filnavn))  
**Rule id** 74714BFC-EDF7-445B-8672-0996214D5845

```
66|private ManntallsimportMapping lesMappingFraClasspath() {
>67|   InputStreamReader inputStreamReader = new
InputStreamReader(this.getClass().getClassLoader().getResourceAsStream(filnavn));
68|   return new Gson().fromJson(inputStreamReader, ManntallsimportMapping.class);
```

## 6.178 OperatorAdminController.java

**Issue** 375A4B13052043F7BFE00F78EDB046B8:

**Source :**

**File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/rbac/ctrls/OperatorAdminController.java 120 **Action** Assign templateInputStream = getResourceAsStream(...)

**Rule id** 74714BFC-EDF7-445B-8672-0996214D5845

```
119|public StreamedContent getAdvanceVoteReceiversTemplate() {
>120|   InputStream templateInputStream =
getFacesContext().getExternalContext().getResourceAsStream(EARLY_VOTE_RECEIVER_TEMPLATE_PATH);
121|   return streamedContent(templateInputStream,
messageProvider.get("@rbac.uploadAdvanceVoteReceivers.templateFileName") + ".xlsx");
```

**Sink :**

**File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/rbac/ctrls/OperatorAdminController.java 120

**Action** Assign templateInputStream = getResourceAsStream(...)

**Rule id** 74714BFC-EDF7-445B-8672-0996214D5845

```
119|public StreamedContent getAdvanceVoteReceiversTemplate() {
>120|   InputStream templateInputStream =
getFacesContext().getExternalContext().getResourceAsStream(EARLY_VOTE_RECEIVER_TEMPLATE_PATH);
121|   return streamedContent(templateInputStream,
messageProvider.get("@rbac.uploadAdvanceVoteReceivers.templateFileName") + ".xlsx");
```

**Issue** E4B0AC9255DB905F9852C5CECE823D0B:

**Source :**

**File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/rbac/ctrls/OperatorAdminController.java 125 **Action** Assign templateInputStream = getResourceAsStream(...)

**Rule id** 74714BFC-EDF7-445B-8672-0996214D5845

```
124|public StreamedContent getVoteReceiverAndPollingPlaceResponsibleTemplate() {
>125|   InputStream templateInputStream =
getFacesContext().getExternalContext().getResourceAsStream(VOTE_RECEIVER_AND_PP_RESPONSIBLE_TEMPLATE_PATH);
126|   return streamedContent(templateInputStream,
messageProvider.get("@rbac.uploadVoteReceiversAndPollingPlaceResponsible.templateName") + ".xlsx");
```

**Sink :**

**File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/rbac/ctrls/OperatorAdminController.java 125

**Action** Assign templateInputStream = getResourceAsStream(...)

**Rule id** 74714BFC-EDF7-445B-8672-0996214D5845

```
124|public StreamedContent getVoteReceiverAndPollingPlaceResponsibleTemplate() {
>125|   InputStream templateInputStream =
getFacesContext().getExternalContext().getResourceAsStream(VOTE_RECEIVER_AND_PP_RESPONSIBLE_TEMPLATE_PATH);
126|   return streamedContent(templateInputStream,
messageProvider.get("@rbac.uploadVoteReceiversAndPollingPlaceResponsible.templateName") + ".xlsx");
```

## 6.179 ReduserGeografiServlet.java

**Issue** 8207A4C65F133234C8AA975B8C214B65:

**Source :**

**File** admin-backend/src/main/java/no/valg/eva/admin/backend/web/ReduserGeografiServlet.java 103

**Action** BranchNotTaken Branch not taken: (!this.feilIParametre(parameterGeografispesifikasjon, parameterValghendelseId))

```
100|String parameterValghendelseId = req.getParameter(PARAMETER_VALGHENDELSE_ID);
101|GeografiReduksjonsmodus parameterReduksjonsmodus = finnReduksjonsmodus(req);
102|
>103|if (feilIParametre(parameterGeografispesifikasjon, parameterValghendelseId)) {
104|   sendFeilmeldingTilbake(resp);
105|} else {
106|   reduserGeografiFor(parameterGeografispesifikasjon, parameterValghendelseId, parameterReduksjonsmodus);
```

**Sink :**

**File** admin-backend/src/main/java/no/valg/eva/admin/backend/web/ReduserGeografiServlet.java 107 **Action** Assign w = getOutputWriter(...)

**Rule id** 74714BFC-EDF7-445B-8672-0996214D5845

```

104|         sendFeilmeldingTilbake(resp);
105|     } else {
106|         reducerGeografiFor(parameterGeografispesifikasjon, parameterValghendelseId, parameterReduksjonsmodus);
>107|         Writer w = getOutputWriter(resp);
108|         w.append(wrapInHtmlHeaders("<p>Reduksjon av geografi er ferdigstilt. Se i loggen for å se hva som ble utført.</p>"));

```

**Issue E990F9D63FDF703AD77ECDBA9CE6CE46:****Source :**

**File** admin-backend/src/main/java/no/valg/eva/admin/backend/web/ReducerGeografiServlet.java 129 **Action** Assign w = getOutputWriter(...)

**Rule id** 74714BFC-EDF7-445B-8672-0996214D5845

```

128|private void sendFeilmeldingTilbake(HttpServletResponse resp) throws IOException {
>129|     Writer w = getOutputWriter(resp);
130|     w.append(wrapInHtmlHeaders(buildGeneriskFeilmelding() + buildReducerGeografiForm()));

```

**Sink :**

**File** admin-backend/src/main/java/no/valg/eva/admin/backend/web/ReducerGeografiServlet.java 129 **Action** Assign w = getOutputWriter(...)

**Rule id** 74714BFC-EDF7-445B-8672-0996214D5845

```

128|private void sendFeilmeldingTilbake(HttpServletResponse resp) throws IOException { >129|
Writer w = getOutputWriter(resp);
130|     w.append(wrapInHtmlHeaders(buildGeneriskFeilmelding() +
buildReducerGeografiForm()));

```

**6.180 ReportTemplateUploadStatusServlet.java****Issue 06F6A7B64D5EB644A80DFA43A2BBBF9F:****Source :**

**File** admin-backend/src/main/java/no/valg/eva/admin/backend/reporting/jasperserver/ReportTemplateUploadStatusServlet.java 32 **Action** Assign w = getOutputWriter(...)

**Rule id** 74714BFC-EDF7-445B-8672-0996214D5845

```

30|@Override
31|protected void doGet(final HttpServletRequest req, final HttpServletResponse resp) throws ServletException,
IOException {
>32|     Writer w = getOutputWriter(resp);
33|
34|     if (isReportTemplatesUploaded()) {
35|         w.append(wrapInHtmlHeaders("Reports are ready"));

```

**Sink :**

**File** admin-backend/src/main/java/no/valg/eva/admin/backend/reporting/jasperserver/ReportTemplateUploadStatusServlet.java 32 **Action** Assign w = getOutputWriter(...)

**Rule id** 74714BFC-EDF7-445B-8672-0996214D5845

```

30|@Override
31|protected void doGet(final HttpServletRequest req, final HttpServletResponse resp) throws ServletException,
IOException {
>32|     Writer w = getOutputWriter(resp);
33|
34|     if (isReportTemplatesUploaded()) {
35|         w.append(wrapInHtmlHeaders("Reports are ready"));

```

**6.181 SystemDateTimeServlet.java****Issue 952971AEA8D535731864B8E5B39D9E46:****Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/common/web/SystemDateTimeServlet.java 47  
**Action** Assign w = getOutputWriter(...)

**Rule id** 74714BFC-EDF7-445B-8672-0996214D5845

```

45|@Override
46|protected void doGet(final HttpServletRequest req, final HttpServletResponse resp) throws ServletException,
IOException {
>47|     Writer w = getOutputWriter(resp);
48|
49|     if (canChangeTime()) {
50|         w.append(buildSetDateTimeFormHtml());

```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/common/web/SystemDateTimeServlet.java 47

**Action** Assign w = getOutputWriter(...)

**Rule id** 74714BFC-EDF7-445B-8672-0996214D5845

```

45|@Override

```



```

46|protected void doGet(final HttpServletRequest req, final HttpServletResponse resp) throws ServletException,
IOException {
>47|    Writer w = getOutputWriter(resp);
48|
49|    if (canChangeTime()) {
50|        w.append(buildSetDateTimeFormHtml());

```

### Issue ACC4B1DA648EA74563E8E4B90327733F:

#### Source :

File admin-common/src/main/java/no/valg/eva/admin/common/web/SystemDateTimeServlet.java 102

Action Assign w = getOutputWriter(...)

Rule id 74714BFC-EDF7-445B-8672-0996214D5845

```

101|private void handleSettingOfDateTime(HttpServletRequest req, HttpServletResponse resp) throws IOException { >102|
Writer w = getOutputWriter(resp);
103|    String timeSetting = req.getParameter(PARAMETER_TIME_SETTING);
104|    String timeSettingForOutput = StringEscapeUtils.escapeHtml4(timeSetting); 105|    String fixedDateTimeString
= req.getParameter(PARAMETER_FIXED_TIME);

```

#### Sink :

File admin-common/src/main/java/no/valg/eva/admin/common/web/SystemDateTimeServlet.java 102 Action Assign w = getOutputWriter(...)

Rule id 74714BFC-EDF7-445B-8672-0996214D5845

```

101|private void handleSettingOfDateTime(HttpServletRequest req, HttpServletResponse resp) throws IOException { >102|
Writer w = getOutputWriter(resp);
103|    String timeSetting = req.getParameter(PARAMETER_TIME_SETTING);
104|    String timeSettingForOutput = StringEscapeUtils.escapeHtml4(timeSetting); 105|    String fixedDateTimeString
= req.getParameter(PARAMETER_FIXED_TIME);

```

## [Medium] Weak Cryptographic Hash

*Security Features*

**Analysis:** Suspicious

### Abstract

Weak cryptographic hashes cannot guarantee data integrity and should not be used in security-critical contexts.

### Explanation

MD2, MD4, MD5, RIPEMD-160, and SHA-1 are popular cryptographic hash algorithms often used to verify the integrity of messages and other data. However, as recent cryptanalysis research has revealed fundamental weaknesses in these algorithms, they should no longer be used within security-critical contexts.

Effective techniques for breaking MD and RIPEMD hashes are widely available, so those algorithms should not be relied upon for security. In the case of SHA-1, current techniques still require a significant amount of computational power and are more difficult to implement. However, attackers have found the Achilles' heel for the algorithm, and techniques for breaking it will likely lead to the discovery of even faster attacks.

### Recommendation

Discontinue the use of MD2, MD4, MD5, RIPEMD-160, and SHA-1 for data-verification in securitycritical contexts. Currently, SHA-224, SHA-256, SHA-384, SHA-512, and SHA-3 are good alternatives. However, these variants of the Secure Hash Algorithm have not been scrutinized as closely as SHA-1, so be mindful of future research that might impact the security of these algorithms.

## 6.182 DirectoryTreeDigestCalculator.java

**Issue** 392FFB7943AE975927D824B1D8D1D068:

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/util/DirectoryTreeDigestCalculator.java 100 **Action** InCall getInstance()

**Rule id** 2254C8BD-DABE-41CB-9D12-CAC442B24079

```

98|private static MessageDigest getMessageDigest() {
99|    try {
>100|        return MessageDigest.getInstance("MD5");
101|    } catch (NoSuchAlgorithmException e) {
102|        throw new RuntimeException("md5 not supported", e);

```

## 6.183 ErrorPageRenderer.java

**Issue** 6FFBCBDBCE207EC700BC177C1D045173:

**Sink :**

**File** admin-frontend/src/main/java/no/evote/presentation/exceptions/ErrorPageRenderer.java 51 **Action** InCall getInstance()

**Rule id** 2254C8BD-DABE-41CB-9D12-CAC442B24079

```

49|public static String md5(String message) {
50|    try {
>51|        MessageDigest m = MessageDigest.getInstance("MD5");
52|        m.update(message.getBytes(), 0, message.length());
53|
54|        return new BigInteger(1, m.digest()).toString(16).substring(0, 8);

```

## 6.184 InputStreamChecksumProvider.java

**Issue** A09D263B7B5CE090B5C94303D78503B3:

**Sink :**

**File** admin-database/src/main/java/no/valg/eva/admin/database/tools/InputStreamChecksumProvider.java 34 **Action** InCall getInstance()

**Rule id** 2254C8BD-DABE-41CB-9D12-CAC442B24079

```

32|private static MessageDigest getMessageDigest() {
33|    try {
>34|        return MessageDigest.getInstance("MD5");
35|    } catch (NoSuchAlgorithmException e) {
36|        throw new RuntimeException("md5 not supported", e);

```

### [Low] Code Correctness / Erroneous String Compare

Code Quality

**Analysis:** Reliability Issue

#### Abstract

Strings should be compared with the `equals()` method, not `==` or `!=`.

#### Explanation

This program uses `==` or `!=` to compare two strings for equality, which compares two objects for equality, not their values. Chances are good that the two references will never be equal.

**Example 1:** The following branch will never be taken.

```

if (args[0] == STRING_CONSTANT) {
    logger.info("miracle");
}

```

The `==` and `!=` operators will only behave as expected when they are used to compare strings contained in objects that are equal. The most common way for this to occur is for the strings to be interned, whereby the strings are added to a pool of objects maintained by the `String` class. Once a string is interned, all uses of that string will use the same object and equality operators will behave as expected. All string literals and string-valued constants are interned automatically. Other strings can be interned

manually be calling `String.intern()`, which will return a canonical instance of the current string, creating one if necessary.

## Recommendation

Use `equals()` to compare strings.

**Example 2:** The code in Example 1 could be rewritten in the following way:

```
if (STRING_CONSTANT.equals(args[0])) {
    logger.info("could happen");
}
```

## 6.185 JasperReportServiceBean.java

**Issue** 0D64F47BDB55BC3DD2FE07F8AB488AA0:

**Sink :**

**File** admin-backend/src/main/java/no/valg/eva/admin/backend/reporting/jasperserver/JasperReportServiceBean.java 514

```
511|    return values;
512|}
513|try {
>514|    switch (parameterId) {
515|
516|        case ELECTION_GROUP:
517|            getSelectableValuesForElectionGroup(reportParameter, values);
```

**Issue** 18B0BF5E32E6892C6B3C1084FB11601B:

**Sink :**

**File** admin-backend/src/main/java/no/valg/eva/admin/backend/reporting/jasperserver/JasperReportServiceBean.java 514

```
511|    return values;
512|}
513|try {
>514|    switch (parameterId) {
515|
516|        case ELECTION_GROUP:
517|            getSelectableValuesForElectionGroup(reportParameter, values);
```

**Issue** 304BF4C905333D5716FB86F3A1Fo6DE3:

**Sink :**

**File** admin-backend/src/main/java/no/valg/eva/admin/backend/reporting/jasperserver/JasperReportServiceBean.java 514

```
511|    return values;
512|}
513|try {
>514|    switch (parameterId) {
515|
516|        case ELECTION_GROUP:
517|            getSelectableValuesForElectionGroup(reportParameter, values);
```

**Issue** 4DB93A86FE8BB3787DBD17EAAEE07BE6:

**Sink :**

**File** admin-backend/src/main/java/no/valg/eva/admin/backend/reporting/jasperserver/JasperReportServiceBean.java 514

```
511|    return values;
512|}
513|try {
>514|    switch (parameterId) {
515|
516|        case ELECTION_GROUP:
517|            getSelectableValuesForElectionGroup(reportParameter, values);
```

**Issue** 53AEDE3CF05D12755FA21560EDA8BE0B:

**Sink :**

**File** admin-backend/src/main/java/no/valg/eva/admin/backend/reporting/jasperserver/JasperReportServiceBean.java 514

```
511|    return values;
512|}
513|try {
>514|    switch (parameterId) {
```

```
515|
516|     case ELECTION_GROUP:
517|         getSelectableValuesForElectionGroup(reportParameter, values);
```

**Issue 55100CFBC7541E1F308636AE3403D944:****Sink :****File** admin-backend/src/main/java/no/valg/eva/admin/backend/reporting/jasperserver/JasperReportServiceBean.java 514

```
511|     return values;
512| }
513| try {
>514|     switch (parameterId) {
515|
516|         case ELECTION_GROUP:
517|             getSelectableValuesForElectionGroup(reportParameter, values);
```

**Issue 639EAF35B6029AA94245D5DB05B2C145:****Sink :****File** admin-backend/src/main/java/no/valg/eva/admin/backend/reporting/jasperserver/JasperReportServiceBean.java 514

```
511|     return values;
512| }
513| try {
>514|     switch (parameterId) {
515|
516|         case ELECTION_GROUP:
517|             getSelectableValuesForElectionGroup(reportParameter, values);
```

**Issue 89F281DE8F668BB94FF6AD4A4402D022:****Sink :****File** admin-backend/src/main/java/no/valg/eva/admin/backend/reporting/jasperserver/JasperReportServiceBean.java 514

```
511|     return values;
512| }
513| try {
>514|     switch (parameterId) {
515|
516|         case ELECTION_GROUP:
517|             getSelectableValuesForElectionGroup(reportParameter, values);
```

**Issue 94542CCDB64D69440CFA79203A4979D0:****Sink :****File** admin-backend/src/main/java/no/valg/eva/admin/backend/reporting/jasperserver/JasperReportServiceBean.java 454

```
451|         "Report execution failed, received Response from server based on requestId %s with missing Content-
Type",
452|         reportInProgress.getRequestId());
453| }
>454| switch (contentType) {
455| case "application/status+xml":
456|     ReportExecutionStatus reportExecutionStatus = response.readEntity(ReportExecutionStatus.class);
457|     if (reportExecutionStatus != null) {
```

**Issue AC0CF5821D61B7A82CCE18E8D7958A79:****Sink :****File** admin-backend/src/main/java/no/valg/eva/admin/backend/reporting/jasperserver/JasperReportServiceBean.java 514

```
511|     return values;
512| }
513| try {
>514|     switch (parameterId) {
515|
516|         case ELECTION_GROUP:
517|             getSelectableValuesForElectionGroup(reportParameter, values);
```

**Issue ACB793377F7ECAF5465828814C4EF57B:****Sink :****File** admin-backend/src/main/java/no/valg/eva/admin/backend/reporting/jasperserver/JasperReportServiceBean.java 514

```
511|     return values;
512| }
513| try {
>514|     switch (parameterId) {
515|
516|         case ELECTION_GROUP:
517|             getSelectableValuesForElectionGroup(reportParameter, values);
```

**Issue E9381E056C1AA5EAE666EB3096AB7E1E:****Sink :**

File admin-backend/src/main/java/no/valg/eva/admin/backend/reporting/jasperserver/JasperReportServiceBean.java 514

```

511|     return values;
512| }
513| try {
>514|     switch (parameterId) {
515|
516|         case ELECTION_GROUP:
517|             getSelectableValuesForElectionGroup(reportParameter, values);

```

**Issue FC7304A08B47863803C82EE7E047FF4C:****Sink :**

File admin-backend/src/main/java/no/valg/eva/admin/backend/reporting/jasperserver/JasperReportServiceBean.java 514

```

511|     return values;
512| }
513| try {
>514|     switch (parameterId) {
515|
516|         case ELECTION_GROUP:
517|             getSelectableValuesForElectionGroup(reportParameter, values);

```

**[Low] Cross-Site Scripting / External Links***Input Validation and Representation***Analysis:** Suspicious**Abstract**

Sending unvalidated data to a web browser can result in the browser executing malicious code. Settings in the configuration can minimize and reduce the exposure to cross-site scripting

**Explanation**

Cross-site scripting (XSS) vulnerabilities occur when:

1. Data enters a web application through an untrusted source, most frequently a web request or database.
2. The data is included in dynamic content that is sent to a web user without being validated.

The malicious content sent to the web browser often takes the form of a segment of JavaScript, but may also include HTML, Flash or any other type of code that the browser may execute. The variety of attacks based on XSS is almost limitless, but they commonly include transmitting private data like cookies or other session information to the attacker, redirecting the victim to web content controlled by the attacker, or performing other malicious operations on the user's machine under the guise of the vulnerable site.

Since attacks against XSS vulnerabilities often involve communicating with or redirecting to a malicious site controlled by the attacker, the ability to inject references to content on other domains is integral to many exploits. AntiSamy can be configured to prevent links to external domains, which diminishes the damage an attacker may cause through an XSS attack. However, this protection is only a partial solution and does not address the overall threat posed by XSS vulnerabilities.

**Example 1:** The following AntiSamy configuration entry allows links to URLs outside of the domain on which the application is running.

```

<attribute name="href" onInvalid="filterTag">
<regexp-list>
<regexp name="onsiteURL"/>
<regexp name="offsiteURL"/>
</regexp-list>

```

```
</attribute>
```

## Recommendation

Whenever possible, prevent links from referencing external domains.

**Example 2:** The following AntiSamy configuration entry blocks links to URLs outside of the domain on which the application is running.

```
<attribute name="href" onInvalid="filterTag">
<regex-list>
<regex name="onsiteURL"/>
</regex-list> </attribute>
```

Since XSS vulnerabilities occur when an application includes malicious data in its output, one logical approach is to validate data immediately before it leaves the application. However, because web applications often have complex and intricate code for generating dynamic content, this method is prone to errors of omission (missing validation). An effective way to mitigate this risk is to also perform input validation for XSS.

Web applications must validate their input to prevent other vulnerabilities, such as SQL injection, so augmenting an application's existing input validation mechanism to include checks for XSS is generally relatively easy. Despite its value, input validation for XSS does not take the place of rigorous output validation. An application may accept input through a shared data store or other trusted source, and that data store may accept input from a source that does not perform adequate input validation. Therefore, the application cannot implicitly rely on the safety of this or any other data. This means the best way to prevent XSS vulnerabilities is to validate everything that enters the application and leaves the application destined for the user.

The most secure approach to validation for XSS is to create a whitelist of safe characters that are allowed to appear in HTTP content and accept input composed exclusively of characters in the approved set. For example, a valid username might only include alpha-numeric characters or a phone number might only include digits 0-9. However, this solution is often infeasible in web applications because many characters that have special meaning to the browser should still be considered valid input once they are encoded, such as a web design bulletin board that must accept HTML fragments from its users.

A more flexible, but less secure approach is known as blacklisting, which selectively rejects or escapes potentially dangerous characters before using the input. In order to form such a list, you first need to understand the set of characters that hold special meaning for web browsers. Although the HTML standard defines what characters have special meaning, many web browsers try to correct common mistakes in HTML and may treat other characters as special in certain contexts, which is why we do not encourage the use of blacklists as a means to prevent XSS. The CERT(R) Coordination Center at the Software Engineering Institute at Carnegie Mellon University provides the following details about special characters in various contexts [1]:

In the content of a block-level element (in the middle of a paragraph of text):

- "<" is special because it introduces a tag.
- "&" is special because it introduces a character entity.
- ">" is special because some browsers treat it as special, on the assumption that the author of the page intended to include an opening "<", but omitted it in error.

The following principles apply to attribute values:

- In attribute values enclosed with double quotes, the double quotes are special because they mark the end of the attribute value.

- In attribute values enclosed with single quote, the single quotes are special because they mark the end of the attribute value.
- In attribute values without any quotes, white-space characters, such as space and tab, are special.
- "&" is special when used with certain attributes, because it introduces a character entity.

In URLs, for example, a search engine might provide a link within the results page that the user can click to re-run the search. This can be implemented by encoding the search query inside the URL, which introduces additional special characters:

- Space, tab, and new line are special because they mark the end of the URL.
- "&" is special because it either introduces a character entity or separates CGI parameters.
- Non-ASCII characters (that is, everything above 128 in the ISO-8859-1 encoding) are not allowed in URLs, so they are considered to be special in this context.
- The "%" symbol must be filtered from input anywhere parameters encoded with HTTP escape sequences are decoded by server-side code. For example, "%" must be filtered if input such as "%68%65%6C%6C%6F" becomes "hello" when it appears on the web page in question.

Within the body of a <SCRIPT> </SCRIPT>:

- Semicolons, parentheses, curly braces, and new line characters should be filtered out in situations where text could be inserted directly into a pre-existing script tag.

Server-side scripts:

- Server-side scripts that convert any exclamation characters (!) in input to double-quote characters (") on output might require additional filtering.

Configuration files can be used to shrink the attack surface attackers may use to execute an attack.

**Example 1:** The following AntiSamy configuration file disallows to set an offsite url in an href.

```
<attribute name="href" onInvalid="filterTag">
<regexp-list>
<regexp name="onsiteURL"/>
</regexp-list>
</attribute>
```

## 6.186 antisamy-eva.xml

**Issue** 49CCDF7CFFA5FCD1DAD37046EA2D5063:

**Sink :**

**File** admin-common/src/main/resources/antisamy-eva.xml 199

```
196|<attribute name="href">
197|   <regexp-list>
198|     <regexp name="onsiteURL"/>
>199|     <regexp name="offsiteURL"/>
200|   </regexp-list>
201|</attribute>
```

### [Low] Denial of Service

*Input Validation and Representation*

**Analysis:** Reliability Issue

**Abstract**

An attacker could cause the program to crash or otherwise become unavailable to legitimate users.

## Explanation

Attackers may be able to deny service to legitimate users by flooding the application with requests, but flooding attacks can often be defused at the network layer. More problematic are bugs that allow an attacker to overload the application using a small number of requests. Such bugs allow the attacker to specify the quantity of system resources their requests will consume or the duration for which they will use them.

**Example 1:** The following code allows a user to specify the amount of time for which a thread will sleep. By specifying a large number, an attacker may tie up the thread indefinitely. With a small number of requests, the attacker may deplete the application's thread pool.

```
int usrSleepTime = Integer.parseInt(usrInput);
Thread.sleep(usrSleepTime);
```

**Example 2:** The following code reads a String from a zip file. Because it uses the `readLine()` method, it will read an unbounded amount of input. An attacker may take advantage of this code to cause an `OutOfMemoryException` or to consume a large amount of memory so that the program spends more time performing garbage collection or runs out of memory during some subsequent operation.

```
InputStream zipInput = zipFile.getInputStream(zipEntry);
Reader zipReader = new InputStreamReader(zipInput);
BufferedReader br = new BufferedReader(zipReader); String line = br.readLine();
```

## Recommendation

Validate user input to ensure that it will not cause inappropriate resource utilization.

**Example 3:** The following code allows a user to specify the amount of time for which a thread will sleep just as in Example 1, but only if the value is within reasonable bounds.

```
int usrSleepTime = Integer.parseInt(usrInput); if (usrSleepTime >= SLEEP_MIN && usrSleepTime <= SLEEP_MAX) {
    Thread.sleep(usrSleepTime);
} else {
    throw new
    Exception(
    "Inva
    lid
    sleep
    durat
    ion");
    ;
} }
```

**Example 4:** The following code reads a String from a zip file just as in Example 2, but the maximum string length it will read is `MAX_STR_LEN` characters.

```
InputStream zipInput = zipFile.getInputStream(zipEntry);
Reader zipReader = new InputStreamReader(zipInput);
BufferedReader br = new BufferedReader(zipReader);
StringBuffer sb = new StringBuffer();
int intC;
while ((intC = br.read()) != -1)
{ char c = (char) intC; if
(c == '\n') { break; }
if (sb.length() >= MAX_STR_LEN) {
throw new Exception("input too long");
}
sb.append(c);
}
String line = sb.toString();
```



## 6.187 CSVReader.java

**Issue** F29B3FD9CADFAB9534505ED89DA3545A:

**Sink :**

**File** admin-common/src/main/java/no/evote/util/CSVReader.java 39

**Action** InCall readLine()

**Rule id** 24023E22-D6C7-4D5C-Bo49-38B7EFC8B408

```
36|try {
37|    inputStreamReader = new InputStreamReader(inputStream, EvoteConstants.CHARACTER_SET);
38|    BufferedReader bufReader = new BufferedReader(inputStreamReader);
>39|    while ((line = bufReader.readLine()) != null) {
40|        LOG.debug("Reading line=" + line);
41|        regexCSVTokenizing(line, propertyMap);
```

## 6.188 CSVUtil.java

**Issue** 05D71DD0B3A435E6470B40844B9DCD73:

**Sink :**

**File** admin-common/src/main/java/no/evote/util/CSVUtil.java 60

**Action** InCall readLine()

**Rule id** 24023E22-D6C7-4D5C-Bo49-38B7EFC8B408

```
57|BufferedReader bufferedReader = new BufferedReader(reader);
58|int counter = 0;
59|String line;
>60|while ((line = bufferedReader.readLine()) != null) {
61|    if (counter == 0 && !line.matches("^@[\\w]+.*")) {
62|        line = line.substring(1);
```

## 6.189 ImportTablesWrapper.java

**Issue** 74F385B5B5ACA5FDE4F8E6DF3950C9F7:

**Sink :**

**File** admin-database/src/main/java/no/valg/eva/admin/database/tools/ImportTablesWrapper.java 108 **Action** InCall readLine()

**Rule id** 24023E22-D6C7-4D5C-Bo49-38B7EFC8B408

```
105|private void printOutputToConsole(Process process) throws IOException {
106|    BufferedReader input = new BufferedReader(new InputStreamReader(process.getInputStream())); // Using default
encoding to write to console
107|    String line;
>108|    while ((line = input.readLine()) != null) {
109|        System.out.println(line);
110|    }
111|    input.close();
```

## 6.190 SkdVoterFileParser.java

**Issue** 69E811FD1813554255771C71BEA5625C:

**Sink :**

**File** admin-voting/src/main/java/no/valg/eva/admin/voting/domain/electoralroll/SkdVoterFileParser.java 27 **Action** InCall readLine()

**Rule id** 24023E22-D6C7-4D5C-Bo49-38B7EFC8B408

```
25|public SkdVoterFileParser(final String filename) throws IOException {
26|    reader = ImportElectoralRollUtil.getBufferedReaderForFile(filename);
>27|    header = reader.readLine();
28|
29|    if (header == null || header.length() != RECORD_SIZE || !header().equals("HD")) {
30|        throw new IOException("Invalid or non-existent header in import file for electoral roll");
```

**Issue** BD226E9B1641282EFF7B9C73B357EA3C:

**Sink :**

**File** admin-voting/src/main/java/no/valg/eva/admin/voting/domain/electoralroll/SkdVoterFileParser.java 77 **Action** InCall readLine()

**Rule id** 24023E22-D6C7-4D5C-Bo49-38B7EFC8B408

```
74|public boolean hasNext() {
75|    try {
76|        if (nextRecord == null) {
>77|            nextRecord = reader.readLine();
78|        }
79|
```

```
80|         if (!isValidRecord(nextRecord)) {
```

**Issue F86DE3D901D224522FC97607DD2CE4B7:****Sink :****File** admin-voting/src/main/java/no/valg/eva/admin/voting/domain/electoralroll/SkdVoterFileParser.java 56 **Action** InCall readLine()**Rule id** 24023E22-D6C7-4D5C-Bo49-38B7EFC8B408

```
53|public SkdVoterRecord next() {
54|    try {
55|        if (nextRecord == null) {
>56|            nextRecord = reader.readLine();
57|        }
58|    }
59|    if (isValidRecord(nextRecord)) {
```

**6.191 VotingCardTextValidator.java****Issue 6174E93F9B2DBD72A6DAA1110EF5A11F:****Sink :****File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/validators/VotingCardTextValidator.java 51 **Action** InCall readLine()**Rule id** 24023E22-D6C7-4D5C-Bo49-38B7EFC8B408

```
48|         if (numChars > MAX_TEXT_LENGTH || numLines > MAX_LINES) {
49|             return false;
50|         }
>51|         line = reader.readLine();
52|     }
53| } catch (IOException ioe) {
54|     return false;
```

**Issue 6174E93F9B2DBD72A6DAA1110EF5A120:****Sink :****File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/validators/VotingCardTextValidator.java 44 **Action** InCall readLine()**Rule id** 24023E22-D6C7-4D5C-Bo49-38B7EFC8B408

```
41|int numLines = 0;
42|int numChars = 0;
43|try (LineNumberReader reader = new LineNumberReader(new StringReader(o.toString()))) {
>44|    String line = reader.readLine();
45|    while (line != null) {
46|        numLines++;
47|        numChars += line.length();
```

**[Low] Missing Check against Null***API Abuse***Class ID:** C3693353-E32E-4B40-964A-94186E0D63B3**Analysis:** Reliability Issue**Abstract**

The program can dereference a null pointer because it does not check the return value of a function that might return null.

**Explanation**

Just about every serious attack on a software system begins with the violation of a programmer's assumptions. After the attack, the programmer's assumptions seem flimsy and poorly founded, but before an attack many programmers would defend their assumptions well past the end of their lunch break.

Two dubious assumptions that are easy to spot in code are "this function call can never fail" and "it doesn't matter if this function call fails". When a programmer ignores the return value from a function, they implicitly state that they are operating under one of these assumptions.

**Example 1:** The following code does not check to see if the string returned by `getParameter()` is `null` before calling the member function `compareTo()`, potentially causing a null dereference.

```
String itemName = request.getParameter(ITEM_NAME);
if (itemName.compareTo(IMPORTANT_ITEM)) { ... }
...
```

**Example 2:** The following code shows a system property that is set to `null` and later dereferenced by a programmer who mistakenly assumes it will always be defined.

```
System.clearProperty("os.name");
...
String os = System.getProperty("os.name");
if (os.equalsIgnoreCase("Windows 95") )
System.out.println("Not supported");
```

The traditional defense of this coding error is:

"I know the requested value will always exist because.... If it does not exist, the program cannot perform the desired behavior so it doesn't matter whether I handle the error or simply allow the program to die dereferencing a null value."

But attackers are skilled at finding unexpected paths through programs, particularly when exceptions are involved.

## Recommendation

If a function can return an error code or any other evidence of its success or failure, always check for the error condition, even if there is no obvious way for it to occur. In addition to preventing security errors, many initially mysterious bugs have eventually led back to a failed method call with an unchecked return value.

Create an easy to use and standard way for dealing with failure in your application. If error handling is straightforward, programmers will be less inclined to omit it. One approach to standardized error handling is to write wrappers around commonly-used functions that check and handle error conditions without additional programmer intervention. When wrappers are implemented and adopted, the use of non-wrapped equivalents can be prohibited and enforced by using custom rules.

**Example 3:** The following code implements a wrapper around `getParameter()` that checks the return value of `getParameter()` against `null` and uses a default value if the requested parameter is not defined.

```
String safeGetParameter (HttpRequest request, String name)
{
    String value = request.getParameter(name);
    if (value == null) {
        return getDefaultValue(name)
    }
    return value;
}
```

## 6.192 FlywayDatabaseMigrator.java

### Issue D49A2008C64F9105A35AAB649FoCEB95:

**Source :**

**File** admin-database/src/main/java/no/valg/eva/admin/database/tools/FlywayDatabaseMigrator.java 35

**Action** Assign `getProperty(?) : System.getProperty` may return `NULL`

**Rule id** C3693353-E32E-4B40-964A-94186E0D63B3

```
34|private boolean isWindows() {
>35|     return System.getProperty("os.name").toLowerCase().contains("win");
36|}
37|
38|public void restoreBaseline() {
```

**Sink :**

**File** admin-database/src/main/java/no/valg/eva/admin/database/tools/FlywayDatabaseMigrator.java 35

**Action** Assign `getProperty(?) : System.getProperty` may return `NULL`

**Rule id** C3693353-E32E-4B40-964A-94186E0D63B3

```
34|private boolean isWindows() {
```

```

>35|     return System.getProperty("os.name").toLowerCase().contains("win");
36|}
37|
38|public void restoreBaseline() {

```

## [Low] Object Model Violation / Erroneous clone() Method

API Abuse

**Analysis:** Reliability Issue

### Abstract

A `clone()` method should call `super.clone()` to obtain the new object.

### Explanation

All implementations of `clone()` should obtain the new object by calling `super.clone()`. If a class fails to follow this convention, a subclass's `clone()` method will return an object of the wrong type.

**Example 1:** The following two classes demonstrate a bug introduced by failing to call `super.clone()`. Because of the way `Kibitzer` implements `clone()`, `FancyKibitzer`'s `clone` method will return an object of type `Kibitzer` instead of `FancyKibitzer`.

```

public class Kibitzer implements Cloneable {
    public Object clone() throws CloneNotSupportedException
    {
        Object returnMe = new Kibitzer();    ...
    }
}
public class FancyKibitzer extends Kibitzer
implements Cloneable {
    public Object clone() throws CloneNotSupportedException {
        Object returnMe = super.clone();
    }
}
...
}

```

### Recommendation

Always obtain the new object by calling `super.clone()`. The `java.lang.Object` implementation of `clone()` will always return an object of the correct type.

**Example 2:** The code in Example 1 could be rewritten in the following way:

```

public class Kibitzer implements Cloneable {
    public Object clone() throws CloneNotSupportedException
    {
        Object returnMe = super.clone();    ...
    }
}
public class FancyKibitzer extends Kibitzer
implements Cloneable {
    public Object clone() throws CloneNotSupportedException
    {
        Object returnMe = super.clone();    ...
    }
}

```

## 6.193 ReportParameter.java

**Issue** CC784FCA44ABADD772EF71B7B84E4783:

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/common/reporting/model/ReportParameter.java 108-110 **Fact**  
**Name:** no.valg.eva.admin.common.reporting.model.ReportParameter.clone

```

107|@Override
>108|protected ReportParameter clone() throws CloneNotSupportedException {
>109|    return new ReportParameter(id, label, description, inferred, type, defaultValue, getDependentParameters(),
parentValue, mandatory);
>110|}
111|
112|public void addDependentParameter(ReportParameter next) {
113|    if (dependentParameters == null) {

```

## 6.194 ReportTemplate.java

**Issue** 517F6987E4A349AA351659AAE3AE4C48:

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/common/reporting/model/ReportTemplate.java 221-233

**Fact Name:** no.valg.eva.admin.common.reporting.model.ReportTemplate.clone

```

220|@Override
>221|public ReportTemplate clone() throws CloneNotSupportedException {
>222|    return new ReportTemplate(reportUri, reportName, reportDescription, filenamePattern, new
ArrayList<>(Collections2.transform(parameters,
>223|        new Function<ReportParameter, ReportParameter>() {
>224|            @Override
>225|            public ReportParameter apply(ReportParameter input) {
>226|                try {
>227|                    return input.clone();
>228|                } catch (CloneNotSupportedException e) {
>229|                    throw new RuntimeException(e);
>230|                }
>231|            }
>232|        })), copyOf(fileFormats), areaLevel, areaPathMask, hidden, async, maybePreGenerated,
unselectableParameterValues);
>233|}
234|
235|@Override
236|public int compareTo(ReportTemplate other) {

```

**[Low] Object Model Violation / Just one of equals() and hashCode() Defined**

*API Abuse*

**Analysis:** Bad Practice

### Abstract

This class overrides only one of `equals()` and `hashCode()`.

### Explanation

Java objects are expected to obey a number of invariants related to equality. One of these invariants is that equal objects must have equal hashcodes. In other words, if `a.equals(b) == true` then `a.hashCode() == b.hashCode()`.

Failure to uphold this invariant is likely to cause trouble if objects of this class are stored in a collection. If the objects of the class in question are used as a key in a Hashtable or if they are inserted into a Map or Set, it is critical that equal objects have equal hashcodes.

**Example 1:** The following class overrides `equals()` but not `hashCode()`.

```

public class halfway() {
    public boolean equals(Object obj) {    ...
    }
}

```

## Recommendation

The FindBugs documentation recommends the following simple "starter" implementation of `hashCode()` [1]. It is highly inefficient, but it will produce correct results. If you do not believe that `hashCode()` is important for your program, consider using this implementation.

**Example 2:** The code in Example 1 could be rewritten in the following way:

```
public class halfway() {
    public boolean equals(Object obj) {
        ...
    }
    public int hashCode() {
        assert false : "hashCode not designed";
        return 42; // any arbitrary constant will do
    }
}
```

## 6.195 Municipality.java

**Issue** 4568FEA2F832C46FA7F3BE5F0649772A:

**Sink:**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Municipality.java 249-255 **Fact**  
**Name:** no.valg.eva.admin.configuration.domain.model.Municipality.equals

```
248|@Override
>249|public boolean equals(Object other) {
>250|     if (this == other) {
>251|         return true;
>252|     }
>253|
>254|     return super.equals(other);
>255|}
256|
257|@Override
258|public String toString() {
```

**[Low] Obsolete**

*Code Quality*

**Analysis:** Bad Practice

### Abstract

The use of deprecated or obsolete functions could indicate neglected code.

### Explanation

As programming languages evolve, methods occasionally become obsolete due to:

- Advances in the language
- Improved understanding of how operations should perform effectively and securely
- Changes in the conventions that govern certain operations

Methods that are removed from a language are usually replaced by newer counterparts that perform the same task in some different and hopefully better way.

**Example:** The following code constructs a string object from an array of bytes and a value that specifies the top 8 bits of each 16-bit Unicode character.

```
...
String name = new String(nameBytes, highByte); ...
```

In this example, the constructor may fail to correctly convert bytes to characters depending upon which charset is used to encode the string represented by `nameBytes`. Due to the evolution of the charsets used to encode strings, this constructor was deprecated and replaced by a constructor that accepts as one of its parameters the name of the `charset` used to encode the bytes for conversion.

Not all functions are deprecated or replaced because they pose a security risk. However, the presence of an obsolete function often indicates that the surrounding code has been neglected and may be in a state of disrepair. Software security has not been a priority, or even a consideration, for very long. If the program uses deprecated or obsolete functions, it raises the probability that there are security problems lurking nearby.

## Recommendation

Do not use deprecated or obsolete functions. Regardless of the direct security impact, replace such functions with modern counterparts. When you encounter an obsolete function, be aware that its presence raises the probability that the surrounding code contains security risks. Consider the securityrelevant assumptions that the application was developed under. Are they still valid? Does the presence of a particular obsolete function indicate a larger maintenance problem?

### 6.196 AbstractPickerController.java

#### Issue 2CEEB829CBF48F7BE4BD56956D698EF3:

**Sink :**

**File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/picker/ctrls/AbstractPickerController.java 76-78 **Fact**  
Name: no.valg.eva.admin.frontend.picker.ctrls.AbstractPickerController.getElectionCfg

```
73|         return cfg.getAreaCfg();
74|     }
75|
>76|     public ElectionCfg getElectionCfg() {
>77|         return cfg.getElectionCfg();
```

#### Issue 6B20FBEE8557802634F7EC413113796C:

**Sink :**

**File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/picker/ctrls/AbstractPickerController.java 72-74  
**Fact** Name: no.valg.eva.admin.frontend.picker.ctrls.AbstractPickerController.getAreaCfg

```
69|     return cfg;
70| }
71|
>72| public AreaCfg getAreaCfg() {
>73|     return cfg.getAreaCfg();
>74| }
75|
76| public ElectionCfg getElectionCfg() {
77|     return cfg.getElectionCfg();
```

#### Issue 6B8AEB42B1DE06F504445189E8B68817:

**Sink :**

**File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/picker/ctrls/AbstractPickerController.java 68-70 **Fact**  
Name: no.valg.eva.admin.frontend.picker.ctrls.AbstractPickerController.getCfg

```
>68| public ContextPickerCfg getCfg() {
>69|     return cfg;
>70| }
71|
72| public AreaCfg getAreaCfg() {
73|     return cfg.getAreaCfg();
```

### 6.197 AreaCfg.java

#### Issue 42A84A04AF59E0A77FE6EA39356678C6:

**Sink :**

**File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/picker/cfg/AreaCfg.java 58-61  
**Fact** Name: no.valg.eva.admin.frontend.picker.cfg.AreaCfg.setHideLeadingSingles

```

55|         return hideLeadingSingles;
56|     }
57|
>58|     public AreaCfg setHideLeadingSingles(boolean hideLeadingSingles) {
>59|         this.hideLeadingSingles = hideLeadingSingles;
>60|         return this;

```

**Issue D7A05AF2BD017C6A04CF797145EDC1B2:****Sink :****File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/picker/cfg/AreaCfg.java 21-24**Fact** Name: no.valg.eva.admin.frontend.picker.cfg.AreaCfg.addLevel

```

>21|public AreaCfg addLevel(AreaLevelEnum areaLevelEnum) {
>22|    areaLevels.add(areaLevelEnum);
>23|    return this;
>24|}
25|
26|public Set<AreaLevelEnum> getAreaLevels() {
27|    return areaLevels;

```

## 6.198 ContextPickerCfg.java

**Issue 786DA7B8CFD1CFFFB55650DC90148384:****Sink :****File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/picker/cfg/ContextPickerCfg.java 35-37**Fact** Name: no.valg.eva.admin.frontend.picker.cfg.ContextPickerCfg.getElectionCfg

```

32|     this.electionCfg = electionCfg;
33| }
34|
>35|public ElectionCfg getElectionCfg() {
>36|    return electionCfg;
>37|}
38|
39|public boolean isKeepRequestParams() {
40|    return !keepRequestParams.isEmpty();

```

**Issue 93C2B59C194E0FD9BBoA5A352607A852:****Sink :****File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/picker/cfg/ContextPickerCfg.java 23-25**Fact** Name: no.valg.eva.admin.frontend.picker.cfg.ContextPickerCfg.getAreaCfg

```

21|public abstract String getUri();
22|
>23|public AreaCfg getAreaCfg() {
>24|    return areaCfg;
>25|}
26|
27|public void setAreaCfg(AreaCfg areaCfg) {
28|    this.areaCfg = areaCfg;

```

**Issue C471F2490F79CAF233F315E547DF888A:****Sink :****File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/picker/cfg/ContextPickerCfg.java 27-29**Fact** Name: no.valg.eva.admin.frontend.picker.cfg.ContextPickerCfg.setAreaCfg

```

24|     return areaCfg;
25| }
26|
>27|public void setAreaCfg(AreaCfg areaCfg) {
>28|    this.areaCfg = areaCfg;
>29|}
30|
31|public void setElectionCfg(ElectionCfg electionCfg) {
32|    this.electionCfg = electionCfg;

```

**Issue EF3285DEE160F25EC1C8CEA91F2119C7:****Sink :****File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/picker/cfg/ContextPickerCfg.java 31-33**Fact** Name: no.valg.eva.admin.frontend.picker.cfg.ContextPickerCfg.setElectionCfg

```

28|     this.areaCfg = areaCfg;
29| }
30|

```



```

>31|public void setElectionCfg(ElectionCfg electionCfg) {
>32|    this.electionCfg = electionCfg;
>33|}
34|
35|public ElectionCfg getElectionCfg() {
36|    return electionCfg;

```

## 6.199 ElectionCfg.java

### Issue 628CB5B8794C930F5CE588BA753BACE2:

**Sink :**

**File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/picker/cfg/ElectionCfg.java 18-21

**Fact** Name: no.valg.eva.admin.frontend.picker.cfg.ElectionCfg.setLevel

```

15|    this.level = level;
16|}
17|
>18|public ElectionCfg setLevel(ElectionLevelEnum level) {
>19|    this.level = level;
>20|    return this;
>21|}
22|
23|public ElectionLevelEnum getElectionLevel() {
24|    return level;

```

## 6.200 VoteCount.java

### Issue 3916A05B003669ECACE79661AAF60E3E:

**Sink :**

**File** admin-counting/src/main/java/no/valg/eva/admin/counting/domain/model/VoteCount.java 468 **Fact**

**Location:** rhs.pollingDistrict

```

465|.appendSuper(super.equals(obj))
466|.append(this.voteCountStatus, rhs.voteCountStatus)
467|.append(this.voteCountCategory, rhs.voteCountCategory)
>468|.append(this.pollingDistrict, rhs.pollingDistrict)
469|.append(this.mvArea, rhs.mvArea)
470|.append(this.contestReport, rhs.contestReport)
471|.append(this.countQualifier, rhs.countQualifier)

```

### Issue 5B432C75F640F3B7C31FD845443C46BD:

**Sink :**

**File** admin-counting/src/main/java/no/valg/eva/admin/counting/domain/model/VoteCount.java 229 **Fact**

**Location:** this.pollingDistrict

```

227|@Deprecated
228|public void setPollingDistrict(final PollingDistrict pollingDistrict) {
>229|    this.pollingDistrict = pollingDistrict;
230|}
231|
232|@ManyToOne(fetch = LAZY)

```

### Issue 6124640ECE5F17B9FFF010ACAE2A3DEB:

**Sink :**

**File** admin-counting/src/main/java/no/valg/eva/admin/counting/domain/model/VoteCount.java 224 **Fact**

**Location:** this.pollingDistrict

```

221|@ManyToOne(fetch = LAZY)
222|@JoinColumn(name = "polling_district_pk", nullable = true)
223|public PollingDistrict getPollingDistrict() {
>224|    return pollingDistrict;
225|}
226|
227|@Deprecated

```

### Issue 63D1FE108C38AC5AC3CC9E8A93706C62:

**Sink :**

**File** admin-counting/src/main/java/no/valg/eva/admin/counting/domain/model/VoteCount.java 493 **Fact**

**Location:** this.pollingDistrict

```

490|.appendSuper(super.hashCode())
491|.append(voteCountStatus)
492|.append(voteCountCategory)

```

```
>493|.append(pollingDistrict)
494|.append(mvArea)
495|.append(contestReport)
496|.append(countQualifier)
```

**Issue 6A3A0CBD60B865E2DFCFA669B4E4232E:****Sink :**

**File** admin-counting/src/main/java/no/valg/eva/admin/counting/domain/model/VoteCount.java 406 **Fact**  
**Location:** this.pollingDistrict

```
403|@Override
404|public Long getAreaPk(final AreaLevelEnum level) {
405|    if (level.equals(AreaLevelEnum.POLLING_DISTRICT)) {
>406|        return pollingDistrict.getPk();
407|    } else {
408|        return null;
```

**Issue 6D1F57258005BAAF201CA803768D81C0:****Sink :**

**File** admin-counting/src/main/java/no/valg/eva/admin/counting/domain/model/VoteCount.java 518 **Fact**  
**Location:** this.pollingDistrict

```
515|.appendSuper(super.toString())
516|.append("voteCountStatus", voteCountStatus)
517|.append("voteCountCategory", voteCountCategory)
>518|.append("pollingDistrict", pollingDistrict)
519|.append("mvArea", mvArea)
520|.append("contestReport", contestReport)
521|.append("countQualifier", countQualifier)
```

**Issue 7624C32819C5BE2DA57EF9F8B73E31E1:****Sink :**

**File** admin-counting/src/main/java/no/valg/eva/admin/counting/domain/model/VoteCount.java 468 **Fact**  
**Location:** this.pollingDistrict

```
465|.appendSuper(super.equals(obj))
466|.append(this.voteCountStatus, rhs.voteCountStatus)
467|.append(this.voteCountCategory, rhs.voteCountCategory)
>468|.append(this.pollingDistrict, rhs.pollingDistrict)
469|.append(this.mvArea, rhs.mvArea)
470|.append(this.contestReport, rhs.contestReport)
471|.append(this.countQualifier, rhs.countQualifier)
```

**6.201 Voter.java****Issue 794DF46533A91C8E7A0A78DA8CADE9F3:****Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Voter.java 201

**Fact** **Location:** this.electionEvent

```
198|@ManyToOne(fetch = FetchType.LAZY)
199|@JoinColumn(name = SQLConstants.ELECTION_EVENT_PK, nullable = false)
200|public ElectionEvent getElectionEvent() {
>201|    return electionEvent;
202|}
203|
204|public void setElectionEvent(final ElectionEvent electionEvent) {
```

**Issue D7B3234D048DB4508FA8E508A9B33C78:****Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/Voter.java 205

**Fact** **Location:** this.electionEvent

```
204|public void setElectionEvent(final ElectionEvent electionEvent) {
>205|    this.electionEvent = electionEvent;
206|}
207|
208|@Column(name = "voter_id", nullable = false, length = 11)
```

**6.202 AbstractPickerController.java****Issue FB0638A95435AA31D8C44E2296685047:****Sink :**

**File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/picker/ctrls/AbstractPickerController.java 18

**Fact** **Name:** no.valg.eva.admin.frontend.picker.ctrls.AbstractPickerController.cfg

```

16|public abstract class AbstractPickerController extends ConversationScopedController implements Action {
17|
>18|   protected ContextPickerCfg cfg;
19|   protected Map<String, Object> requestParameterMap;
20|
21|   @PostConstruct

```

## 6.203 ContextPickerCfg.java

**Issue** 9445B58A18044FC3D4E6EDB8C2402CAE:

**Sink :**

**File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/picker/cfg/ContextPickerCfg.java 9

**Fact** Name: no.valg.eva.admin.frontend.picker.cfg.ContextPickerCfg.areaCfg

```

6|@Deprecated
7|public abstract class ContextPickerCfg {
8|
> 9|   protected AreaCfg areaCfg;
10|   protected ElectionCfg electionCfg;
11|   private List<String> keepRequestParams;

```

**Issue** AFB3105F6E7E3586B2AB99618E7DAD19:

**Sink :**

**File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/picker/cfg/ContextPickerCfg.java 10 **Fact**

Name: no.valg.eva.admin.frontend.picker.cfg.ContextPickerCfg.electionCfg

```

7|public abstract class ContextPickerCfg {
8|
> 9|   protected AreaCfg areaCfg;
10|   protected ElectionCfg electionCfg;
11|   private List<String> keepRequestParams;
12|
13|   public ContextPickerCfg() {

```

## 6.204 AreaImportChangesApplicationService.java

**Issue** 0B96DA38FA53928C8B5718C36CA899Fo:

**Sink :**

**File** admin-configuration/src/main/java/no/valg/eva/admin/configuration/application/AreaImportChangesApplicationService.java 138

**Fact** Name: no.valg.eva.admin.configuration.repository.MvAreaRepository.findByPathAndLevel

```

136|String pollingDistrictAreaPath = retrieveAreaPath(electionEvent, contentImportFile); 137|
>138|List<MvArea> mvAreas = mvAreaRepository.findByPathAndLevel(pollingDistrictAreaPath.substring(0, 24),
AreaLevelEnum.BOROUGH.getLevel());
139|
140|if (isEmptyMvAreaList(mvAreas)) {
141|   PollingDistrict newDistrict = newPollingDistrict(pollingDistrictId, pollingDistrictName,
mvAreas.get(0).getBorough());

```

**Issue** AoE60ABF457E192490FEBBF69A434FDB:

**Sink :**

**File** admin-configuration/src/main/java/no/valg/eva/admin/configuration/application/AreaImportChangesApplicationService.java 83

**Fact** Name: no.valg.eva.admin.configuration.repository.MvAreaRepository.findByPathAndLevel

```

80|ElectionEvent electionEvent = electionEventRepository.findByPk(userData.getElectionEventPk());
81|String countryAreaPath = retrieveAreaPath(electionEvent, null);
82|
>83|Collection<MvArea> allExistingPollingDistrictMvAreas = mvAreaRepository.findByPathAndLevel(countryAreaPath,
AreaLevelEnum.POLLING_DISTRICT.getLevel());
84|
85|int line = 1;

```

**Issue** E6CC9584FE4FED81DC813A153367DE91:

**Sink :**

**File** admin-configuration/src/main/java/no/valg/eva/admin/configuration/application/AreaImportChangesApplicationService.java 92 **Fact**

Name: no.valg.eva.admin.configuration.repository.MvAreaRepository.findByPathAndLevel

```

89|if (isEmptyLine(rowContent) && !isHeadingOfImportFile(rowContent)) {
90|
91|   String pollingDistrictAreaPath = retrieveAreaPath(electionEvent, rowContent);
>92|   List<MvArea> existingPollingDistrictMvAreas = mvAreaRepository.findByPathAndLevel(pollingDistrictAreaPath,
93|       AreaLevelEnum.POLLING_DISTRICT.getLevel());
94|
95|   if (isEmptyMvAreaList(existingPollingDistrictMvAreas)) {

```

## 6.205 AreaLevel.java

**Issue** 03B357D3672325D154801AB1372DAEE3:

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/AreaLevel.java 40

**Fact** Name: no.evotest.constants.AreaLevelEnum.getName

```
38|public AreaLevel(final AreaLevelEnum areaLevelEnum) {
39|    id = areaLevelEnum.getLevel();
>40|    name =
areaLevelEnum.getName(); 41|} 42|
43|@Column(name = "area_level_id", unique = true, nullable = false)
```

## 6.206 BallotCount.java

**Issue** 13CoEECD5641E206E530643BC474A347:

**Sink :**

**File** admin-counting/src/main/java/no/valg/eva/admin/counting/domain/model/BallotCount.java 178 **Fact**

Name: no.valg.eva.admin.counting.domain.model.VoteCount.getPollingDistrict

```
175|@Override
176|public Long getAreaPk(AreaLevelEnum level) {
177|    if (level.equals(AreaLevelEnum.POLLING_DISTRICT)) {
>178|        return voteCount.getPollingDistrict().getPk(); 179|    } else {
180|        return null;
```

## 6.207 BaseSettlementController.java

**Issue** EAD13C290DB7BE90A91C8ECB4Do2CFD4:

**Sink :**

**File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/settlement/ctrls/BaseSettlementController.java 137

**Fact** Name: no.evotest.service.configuration.MvElectionService.findSingleByPath

```
134|public String selectContest() {
135|    List<ContestArea> areas = contestAreaService.findContestAreasForContestPath(userData,
contestInfo.getElectionPath());
136|    contestArea = areas.get(0).getMvArea();
>137|    mvElection = mvElectionService.findSingleByPath(contestInfo.getElectionPath());
138|    setSettlementDone(settlementService.hasSettlementForContest(userData, contestInfo.getElectionPath()));
139|
140|    if (initSettlement()) {
```

## 6.208 BoroughController.java

**Issue** 1D31831F03864A068090959A4EDFD2EB:

**Sink :**

**File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/contexteditor/ctrls/BoroughController.java 95

**Fact** Name: no.evotest.presentation.util.MessageUtil.buildFacesMessage

```
93|if (!borough.getId().substring(0, 4).equalsIgnoreCase(parentMvArea.getMunicipality().getId())) {
94|    String[] summaryParams = { ID_LABEL, "@area_level[3].nummer", "@area.list_areas.message.id_four_first" };
>95|    MessageUtil.buildFacesMessage(CREATE_FORM, MessageUtil.AREA_ID_MUST_CONFORM, summaryParams,
FacesMessage.SEVERITY_ERROR);
96|} else {
97|    if (hasIdChanged(borough.getPk(), borough) && idExists(parentMvArea.getMunicipality(), borough)) {
98|        String[] summaryParams = { ID_LABEL, borough.getId(), parentMvArea.getMunicipalityName() };
```

**Issue** 253B5CADDD8E53468CABF2306504CDF:

**Sink :**

**File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/contexteditor/ctrls/BoroughController.java 59

**Fact** Name: no.evotest.presentation.util.MessageUtil.buildFacesMessage

```
56|} else {
57|    if (idExists(parentMvArea.getMunicipality(), newBorough)) {
58|        String[] summaryParams = { ID_LABEL, newBorough.getId(), parentMvArea.getMunicipalityName() };
>59|        MessageUtil.buildFacesMessage(CREATE_FORM, MessageUtil.CHOOSE_UNIQUE_ID, summaryParams,
FacesMessage.SEVERITY_ERROR);
60|    } else {
61|        newBorough.setMunicipality(parentMvArea.getMunicipality());
62|        if (execute() -> {
```

**Issue** 8150A9D0791EFDDDB217786206750D51F:

**Sink :**

**File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/contexteditor/ctrls/BoroughController.java 130 **Fact**  
Name: no.evote.presentation.util.MessageUtil.buildFacesMessage

```
128| } else {
129|     String[] summaryParams = { borough.getName() };
>130|     MessageUtil.buildFacesMessage(null, MessageUtil.REMOVE_NOT_ALLOWED_KEY, summaryParams,
FacesMessage.SEVERITY_INFO);
131|     LOGGER.error("Country " + borough.getName() + " with pk " + borough.getPk() + " is not deletable."); 132| }
133| getRequestContext().addCallbackParam("editBoroughHideDialog", hideDialog);
```

### Issue 852B72E98A131175A93859CB78B25816:

**Sink :**

**File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/contexteditor/ctrls/BoroughController.java 67

**Fact** Name: no.evote.presentation.util.MessageUtil.buildFacesMessage

```
64|     mvAreaPickerController.update(4, parentMvArea.getAreaPath() + "." + newBorough.getId());
65|     }, CREATE_FORM)) {
66|     String[] summaryParams = { newBorough.getName(), parentMvArea.getMunicipalityName() };
>67|     MessageUtil.buildFacesMessage(null, MessageUtil.CREATE_SUBLEVEL_SUCCESSFUL_KEY, summaryParams,
FacesMessage.SEVERITY_INFO);
68|     hideDialog = true;
```

### Issue AFD666EF8F26D43C46F1AB64B4F89347:

**Sink :**

**File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/contexteditor/ctrls/BoroughController.java 89

**Fact** Name: no.evote.presentation.util.MessageUtil.buildFacesMessage

```
86| List<MvArea> mvAreaList = mvAreaService.findByPathAndChildLevel(currentMvArea);
87| if (hasIdChanged(borough.getPk(), borough) && (mvAreaList != null && !mvAreaList.isEmpty())) {
88|     String[] summaryParams = { ID_LABEL, "@area_level[4].name" };
>89|     MessageUtil.buildFacesMessage(EDIT_FORM, MessageUtil.EDIT_ID_NOT_ALLOWED, summaryParams,
FacesMessage.SEVERITY_ERROR);
90|     return;
```

### Issue B0FF596FD48CEB328C9A8A1F1F7D1F69:

**Sink :**

**File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/contexteditor/ctrls/BoroughController.java 125 **Fact**

Name: no.evote.presentation.util.MessageUtil.buildFacesMessage

```
123|     ))) {
124|     String[] summaryParams = { borough.getName(), parentMvArea.getMunicipalityName() };
>125|     MessageUtil.buildFacesMessage(null, MessageUtil.DELETE_FROM_LEVEL_SUCCESSFUL_KEY, summaryParams,
FacesMessage.SEVERITY_INFO);
126|     hideDialog = true;
127|     }
128| } else {
```

### Issue BE8CE2C246862FD6381C87A0B85243EF:

**Sink :**

**File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/contexteditor/ctrls/BoroughController.java 55

**Fact** Name: no.evote.presentation.util.MessageUtil.buildFacesMessage

```
53| if (!newBorough.getId().substring(0, 4).equalsIgnoreCase(parentMvArea.getMunicipality().getId())) {
54|     String[] summaryParams = { ID_LABEL, "@area_level[3].nummer", "@area.list_areas.message.id_four_first" };
>55|     MessageUtil.buildFacesMessage(CREATE_FORM, MessageUtil.AREA_ID_MUST_CONFORM, summaryParams,
FacesMessage.SEVERITY_ERROR);
56| } else {
57|     if (idExists(parentMvArea.getMunicipality(), newBorough)) {
58|         String[] summaryParams = { ID_LABEL, newBorough.getId(), parentMvArea.getMunicipalityName() };
```

### Issue CFA115DCE5983DA70D343485E7C4EA2F:

**Sink :**

**File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/contexteditor/ctrls/BoroughController.java 106

**Fact** Name: no.evote.presentation.util.MessageUtil.buildFacesMessage

```
103|     mvAreaPickerController.update(4, null);
104|     }, EDIT_FORM)) {
105|     String[] summaryParams = { "" };
>106|     MessageUtil.buildFacesMessage(null, MessageUtil.UPDATE_SUCCESSFUL_KEY, summaryParams,
FacesMessage.SEVERITY_INFO);
107|     hideDialog = true;
```

### Issue FCD65CC50DABB1FF8756BE69882654DF:

**Sink :**

**File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/contexteditor/ctrls/BoroughController.java 99 **Fact**

Name: no.evote.presentation.util.MessageUtil.buildFacesMessage

```
96| } else {
```

```

97|     if (hasIdChanged(borough.getPk(), borough) && idExists(parentMvArea.getMunicipality(), borough)) {
98|         String[] summaryParams = { ID_LABEL, borough.getId(), parentMvArea.getMunicipalityName() };
> 99|         MessageUtil.buildFacesMessage(EDIT_FORM, MessageUtil.CHOOSE_UNIQUE_ID, summaryParams,
FacesMessage.SEVERITY_ERROR);
100|     } else {
101|         if (execute() -> {
102|             currentBorough = boroughService.update(userData, borough);

```

## 6.209 CandidateUploadController.java

**Issue** 2E70CFAF4F5734E3BF42236FAE9F7C6F:

**Sink :**

**File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/listproposal/ctrls/CandidateUploadController.java 227 **Fact**

**Name:** no.evote.service.configuration.ContestAreaService.findContestAreasForContest

```

225|private MvArea getMvAreaFromContest(final Contest contest) {
226|    if (contest.getElection().isSingleArea()) {
>227|        List<ContestArea> contestAreaList = contestAreaService.findContestAreasForContest(contest.getPk()); 228|
if (contestAreaList.size() == 1) {
229|        return contestAreaList.get(0).getMvArea();

```

## 6.210 ContestServiceBean.java

**Issue** 61CB2EE2A2D3974AC05B8855C4C6F8DD:

**Sink :**

**File** admin-backend/src/main/java/no/evote/service/configuration/ContestServiceBean.java 91

**Fact Name:** no.valg.eva.admin.configuration.repository.MvAreaRepository.findByPathAndLevel

```

90|public void createContestsForElection(UserData userData, Election election) {
>91|    List<MvArea> mvAreaList =
mvAreaRepository.findByPathAndLevel(election.getElectionGroup().getElectionEvent().getId(), election.getAreaLevel());
92|    for (MvArea mv : mvAreaList) {
93|        Contest contest = new Contest();
94|        contest.setElection(election);

```

## 6.211 CountingAreaDomainService.java

**Issue** 8070EE2452632B19F80441596ABED670:

**Sink :**

**File** admin-counting/src/main/java/no/valg/eva/admin/counting/domain/service/settlement/CountingAreaDomainService.java 71

**Fact Name:** no.valg.eva.admin.configuration.repository.MvAreaRepository.findByPathAndLevel

```

68|private List<MvArea> countingMvAreasByBorough(MvElection contestMvElection, MvArea contestMvArea, CountCategory
category) {
69|    CountingMode countingMode = voteCountService.countingMode(category, contestMvArea.getMunicipality(),
contestMvElection);
70|    if (countingMode == CENTRAL_AND_BY_POLLING_DISTRICT) {
>71|        return mvAreaRepository.findByPathAndLevel(contestMvArea.getPath(),
AreaLevelEnum.POLLING_DISTRICT.getLevel());
72|    }
73|    return singletonList(contestMvArea);

```

## 6.212 CountingImportServiceBean.java

**Issue** E55F2A2418D11753394EAC7952802CD9:

**Sink :**

**File** admin-backend/src/main/java/no/evote/service/counting/CountingImportServiceBean.java 1169

**Fact Name:** no.valg.eva.admin.counting.domain.model.VoteCount.setPollingDistrict

```

1166|    CountQualifier countQualifier, int ordinaryVotes, int rejectedVotes) {
1167|VoteCount voteCount = new VoteCount();
1168|voteCount.setContestReport(contestReport);
>1169|voteCount.setPollingDistrict(mvAreaForReporting.getPollingDistrict());
1170|voteCount.setMvArea(mvAreaForReporting);
1171|voteCount.setVoteCountCategory(votingCountCategory);
1172|voteCount.setCountQualifier(countQualifier);

```

**Issue EDAA55E011F31782E597E3803040D27C:****Sink :****File** admin-backend/src/main/java/no/evote/service/counting/CountingImportServiceBean.java 761**Fact Name:** no.valg.eva.admin.configuration.repository.ReportingUnitRepository.getReportingUnit

```

758|List<ContestRelArea> contestRelAreas = contestRelAreaRepository.findAllAllowed(operatorRole.getMvElection(),
operatorRole.getMvArea());
759|List<ReportingUnit> reportingUnits = new ArrayList<>();
760|for (ContestRelArea contestRelArea : contestRelAreas) {
>761|   reportingUnits.add(reportingUnitRepository.getReportingUnit(contestRelArea));
762|}
763|if (!reportingUnits.contains(reportingUnit)) {
764|   final String operatorRoleMvElection = operatorRole.getMvElection().toString();

```

**6.213 CountryController.java****Issue 0587696F2750A5B5360F7198112D8FCA:****Sink :****File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/contexteditor/ctrls/CountryController.java 104 **Fact****Name:** no.evote.presentation.util.MessageUtil.buildFacesMessage

```

102|} else {
103|   String[] summaryParams = { country.getName() };
>104|   MessageUtil.buildFacesMessage(null, MessageUtil.REMOVE_NOT_ALLOWED_KEY, summaryParams,
FacesMessage.SEVERITY_INFO);
105|   LOGGER.error("Country " + country.getName() + " with pk " + country.getPk() + " is not deletable."); 106|}
107|getRequestContext().addCallbackParam("editCountryHideDialog", hideDialog);

```

**Issue 0DF66D68CoE330C1C5ED9A916CC8CB04:****Sink :****File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/contexteditor/ctrls/CountryController.java 76**Fact Name:** no.evote.presentation.util.MessageUtil.buildFacesMessage

```

74|if (hasIdChanged(country.getPk(), country) && idExists(parentMvArea.getElectionEvent(), country)) {
75|   String[] summaryParams = { ID_LABEL, country.getId(), parentMvArea.getElectionEventName() };
>76|   MessageUtil.buildFacesMessage(EDIT_FORM, MessageUtil.CHOOSE_UNIQUE_ID, summaryParams,
FacesMessage.SEVERITY_ERROR);
77|} else {
78|   if (execute() -> {
79|       currentCountry = countryService.update(userData, country);

```

**Issue 54C46E83E216271FD9F5F09DFC85A1B8:****Sink :****File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/contexteditor/ctrls/CountryController.java 58**Fact Name:** no.evote.presentation.util.MessageUtil.buildFacesMessage

```

55|   }, CREATE_FORM)) {
56|       hideDialog = true;
57|       String[] summaryParams = { newCountry.getName(), parentMvArea.getElectionEventName() };
>58|       MessageUtil.buildFacesMessage(null, MessageUtil.CREATE_SUBLEVEL_SUCCESSFUL_KEY, summaryParams,
FacesMessage.SEVERITY_INFO);
59|   }
60|}
61|getRequestContext().addCallbackParam("createCountryHideDialog", hideDialog);

```

**Issue C9CC0B980FC0B9A805E20C27A2AA2BF8:****Sink :****File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/contexteditor/ctrls/CountryController.java 50**Fact Name:** no.evote.presentation.util.MessageUtil.buildFacesMessage

```

47|boolean hideDialog = false;
48|if (idExists(parentMvArea.getElectionEvent(), newCountry)) {
49|   String[] summaryParams = { ID_LABEL, newCountry.getId(), parentMvArea.getElectionEventName() };
>50|   MessageUtil.buildFacesMessage(CREATE_FORM, MessageUtil.CHOOSE_UNIQUE_ID, summaryParams,
FacesMessage.SEVERITY_ERROR);
51|} else {
52|   if (execute() -> {
53|       countryService.create(userData, newCountry);

```

**Issue E9B55753103F3D8A1ADFEA9F9F072626:****Sink :****File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/contexteditor/ctrls/CountryController.java 99 **Fact****Name:** no.evote.presentation.util.MessageUtil.buildFacesMessage

```

97|   ))) {
98|       String[] summaryParams = { country.getName(), parentMvArea.getElectionEventName() };
> 99|       MessageUtil.buildFacesMessage(null, MessageUtil.DELETE_FROM_LEVEL_SUCCESSFUL_KEY, summaryParams,
FacesMessage.SEVERITY_INFO);

```

```
100|         hideDialog = true;
101|     }
102|} else {
```

**Issue FEE6DF017CD346C42E23CE6CE0FFA6E8:****Sink :****File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/contexteditor/ctrls/CountryController.java 83**Fact** Name: no.evote.presentation.util.MessageUtil.buildFacesMessage

```
80|         mvAreaPickerController.update(1, null);
81|     }, EDIT_FORM)) {
82|         String[] summaryParams = { "" };
>83|         MessageUtil.buildFacesMessage(null, MessageUtil.UPDATE_SUCCESSFUL_KEY, summaryParams,
FacesMessage.SEVERITY_INFO);
84|         hideDialog = true;
```

**6.214 CountyController.java****Issue 48AAB7363277B5DE67D82AA8687425DF:****Sink :****File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/contexteditor/ctrls/CountryController.java 95**Fact** Name: no.evote.presentation.util.MessageUtil.buildFacesMessage

```
92|         mvAreaPickerController.update(2, null);
93|     }, EDIT_FORM)) {
94|         String[] summaryParams = { "" };
>95|         MessageUtil.buildFacesMessage(null, MessageUtil.UPDATE_SUCCESSFUL_KEY, summaryParams,
FacesMessage.SEVERITY_INFO);
96|         hideDialog = true;
```

**Issue 5847925C1E81C5CE6212567E19C7D485:****Sink :****File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/contexteditor/ctrls/CountryController.java 55**Fact** Name: no.evote.presentation.util.MessageUtil.buildFacesMessage

```
53|if (idExists(parentMvArea.getCountry(), newCounty)) {
54|    String[] summaryParams = { ID_LABEL, newCounty.getId(), parentMvArea.getCountryName() };
>55|    MessageUtil.buildFacesMessage(CREATE_FORM, MessageUtil.CHOOSE_UNIQUE_ID, summaryParams,
FacesMessage.SEVERITY_ERROR);
56|} else {
57|    newCounty.setCountry(parentMvArea.getCountry());
58|    if (execute() -> {
```

**Issue 6FC2778436ACDDB8A7C7C59A07338693:****Sink :****File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/contexteditor/ctrls/CountryController.java 110**Fact** Name: no.evote.presentation.util.MessageUtil.buildFacesMessage

```
107|         mvAreaPickerController.update(2, null);
108|     }) {
109|         String[] summaryParams = { county.getName(), parentMvArea.getCountryName() };
>110|         MessageUtil.buildFacesMessage(null, MessageUtil.DELETE_FROM_LEVEL_SUCCESSFUL_KEY, summaryParams,
FacesMessage.SEVERITY_INFO);
111|         hideDialog = true;
112|     }
113|} else {
```

**Issue 7C83D9CoDBABDFABF7D6B14661215BC8:****Sink :****File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/contexteditor/ctrls/CountryController.java 83**Fact** Name: no.evote.presentation.util.MessageUtil.buildFacesMessage

```
80|List<MvArea> mvAreaList = mvAreaService.findByPathAndChildLevel(currentMvArea);
81|if (hasIdChanged(county.getPk(), county) && (mvAreaList != null && !mvAreaList.isEmpty())) {
82|    String[] summaryParams = { ID_LABEL, "@area_level[2].name" };
>83|    MessageUtil.buildFacesMessage(EDIT_FORM, MessageUtil.EDIT_ID_NOT_ALLOWED, summaryParams,
FacesMessage.SEVERITY_ERROR);
84|    return;
85|}
86|if (hasIdChanged(county.getPk(), county) && idExists(parentMvArea.getCountry(), county)) {
```

**Issue A62EC851A0AAF5748065E8A10D9B5C2D:****Sink :****File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/contexteditor/ctrls/CountryController.java 63**Fact** Name: no.evote.presentation.util.MessageUtil.buildFacesMessage



```

60|         mvAreaPickerController.update(2, parentMvArea.getAreaPath() + "." + newCounty.getId());
61|     }, CREATE_FORM) {
62|         String[] summaryParams = { newCounty.getName(), parentMvArea.getCountryName() };
>63|         MessageUtil.buildFacesMessage(null, MessageUtil.CREATE_SUBLEVEL_SUCCESSFUL_KEY, summaryParams,
FacesMessage.SEVERITY_INFO);
64|         hideDialog = true;

```

**Issue D67A4B646F836F9DCFF4FD1108216A32:****Sink :****File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/contexteditor/ctrls/CountyController.java 88**Fact** Name: no.evote.presentation.util.MessageUtil.buildFacesMessage

```

86|if (hasIdChanged(county.getPk(), county) && idExists(parentMvArea.getCountry(), county)) {
87|    String[] summaryParams = { ID_LABEL, county.getId(), parentMvArea.getElectionEventName() };
>88|    MessageUtil.buildFacesMessage(EDIT_FORM, MessageUtil.CHOOSE_UNIQUE_ID, summaryParams,
FacesMessage.SEVERITY_ERROR);
89|} else {
90|    if (execute() -> {
91|        currentCounty = countyService.update(userData, county);

```

**Issue DF786692E7D55B95471353D87DE3507F:****Sink :****File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/contexteditor/ctrls/CountyController.java 115 **Fact****Name:** no.evote.presentation.util.MessageUtil.buildFacesMessage

```

113|} else {
114|    String[] summaryParams = { county.getName() };
>115|    MessageUtil.buildFacesMessage(null, MessageUtil.REMOVE_NOT_ALLOWED_KEY, summaryParams,
FacesMessage.SEVERITY_INFO);
116|    LOGGER.error("County " + county.getName() + " with pk " + county.getPk() + " is not deletable."); 117|}
118|getRequestContext().addCallbackParam("editCountyHideDialog", hideDialog);

```

**6.215 CountyVoteCountsController.java****Issue C12F3C6B024648A6B517CB8E6312B69F:****Sink :****File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/counting/ctrls/CountyVoteCountsController.java 42**Fact** Name: no.evote.service.configuration.MvElectionService.findSingleByPath

```

39|         userData.getOperatorAreaPath(), userData.getOperatorElectionPath(), null);
40|List<ContestInfo> finalContestInfoList = new ArrayList<>();
41|for (ContestInfo contest : contestInfoList) {
>42|    MvElection mvElection = mvElectionService.findSingleByPath(contest.getElectionPath());
43|    // Only interested in contest for county, or in the case of sami election contest on municipality level with
not single area
44|    if (!mvElection.getElection().isReferendum() && (contest.getAreaPath().isCountyLevel() ||
(contest.getAreaPath().isMunicipalityLevel()
45|        && !mvElection.getElection().isSingleArea())) {

```

**6.216 EndreVelgerManntallController.java****Issue 27E37C92296A791DE685E80C71ED42DC:****Sink :****File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/manntall/ctrls/EndreVelgerManntallController.java 76**Fact** Name: no.evote.service.configuration.MvElectionService.findByPathAndLevel

```

74|@PostConstruct
75|public void init() {
>76|    List<MvElection> mveList =
mvElectionService.findByPathAndLevel(userDataController.getElectionEvent().electionPath().path(),
ELECTION_GROUP.getLevel());
77|    if (mveList.size() == 1) {
78|        valgGruppe = mveList.get(0);
79|    } else {

```

**6.217 ForhandRegistreringController.java****Issue 79E2BECF347BD4B189AEF9692DFAF822:****Sink :**

**File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/stemmegivning/ctrls/registrering/ForhandRegistreringController.java 22  
**Fact** Name: no.evotefield.service.voting.VotingService.delete

```

19|public void slettForhandsstemme() {
20|    execute() -> {
21|        String navn = getNavn(getStemmegivning().getVoter());
>22|        votingService.delete(getUserData(), getStemmegivning().getPk());
23|        if (getStemmegivning().getVotingCategory().getId().equals(VotingCategory.FA.getId())) {
24|            buildDetailMessage("@voting.requestRemoveAdvanceVotingFA.response", new String[] { navn },
SEVERITY_INFO);
25|        } else {

```

## 6.218 FrontendStatusServletOld.java

**Issue** 6D881D5F7064447669C681D3B1DDCAA9:

**Sink** :

**File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/servlets/FrontendStatusServletOld.java 30 **Fact**  
**Name:** no.valg.eva.admin.common.application.service.StatusService.getStatus

```

27|@Override
28|public void doGet(final HttpServletRequest req, final HttpServletResponse res) throws IOException {
29|    try {
>30|        res.getWriter().write(statusService.getStatus());
31|    } catch (final IllegalStateException ex) {
32|        LOG.fatal("Unable to access EJB.", ex);
33|        res.getWriter().write(ERROR);

```

## 6.219 ImportOperatorServiceEjb.java

**Issue** 13D3A52C3936EB45547C2461B9EBoED9:

**Sink** :

**File** admin-rbac/src/main/java/no/valg/eva/admin/rbac/service/ImportOperatorServiceEjb.java 177  
**Fact** Name: no.valg.eva.admin.configuration.repository.MvAreaRepository.findByPathAndLevel

```

174|private MvArea retrieveMvArea(UserData userData, ImportOperatorRoleInfo operatorInfo) {
175|    if (!isAreaDefined(operatorInfo.getAreaId())) {
176|        AreaPath ap = new AreaPath(userData.getOperatorMvArea().getAreaPath());
>177|        return mvAreaRepository.findByPathAndLevel(ap.toMunicipalityPath().toString(),
AreaLevelEnum.MUNICIPALITY.getLevel()).get();
178|    }
179|
180|    if (operatorInfo.areaIsPollingDistrict()) {

```

## 6.220 KlondeligeTellingDomainService.java

**Issue** 431597EEFA7D93F86A74155D9FCF1BD3:

**Sink** :

**File** admin-counting/src/main/java/no/valg/eva/admin/counting/domain/service/KlondeligeTellingDomainService.java 293  
**Fact** Name: no.valg.eva.admin.counting.domain.model.VoteCount.getPollingDistrict

```

290|private void lagVoteCountForFylkesvalgstyret(ContestReport contestReportForFylkesvalgstyret, VoteCount
godkjentVoteCountForValgstyret,
291|                                           VoteCountStatus toSettlementStatus) {
292|    VoteCount voteCountTilValgoppgjorerForFylkesvalgstyret = new VoteCount();
>293|
voteCountTilValgoppgjorerForFylkesvalgstyret.setPollingDistrict(godkjentVoteCountForValgstyret.getPollingDistrict());
294|
voteCountTilValgoppgjorerForFylkesvalgstyret.setVoteCountCategory(godkjentVoteCountForValgstyret.getVoteCountCategory(
));
295|
voteCountTilValgoppgjorerForFylkesvalgstyret.setCountQualifier(godkjentVoteCountForValgstyret.getCountQualifier());
296|
voteCountTilValgoppgjorerForFylkesvalgstyret.setApprovedBallots(godkjentVoteCountForValgstyret.getApprovedBallots());

```

**Issue** 66C42963304AFDEAC8588E69B682A1A9:

**Sink** :

**File** admin-counting/src/main/java/no/valg/eva/admin/counting/domain/service/KlondeligeTellingDomainService.java 293  
**Fact** Name: no.valg.eva.admin.counting.domain.model.VoteCount.setPollingDistrict

```

290|private void lagVoteCountForFylkesvalgstyret(ContestReport contestReportForFylkesvalgstyret, VoteCount
godkjentVoteCountForValgstyret,
291|                                     VoteCountStatus toSettlementStatus) {
292|     VoteCount voteCountTilValgoppgjøerForFylkesvalgstyret = new VoteCount();
>293|
voteCountTilValgoppgjøerForFylkesvalgstyret.setPollingDistrict(godkjentVoteCountForValgstyret.getPollingDistrict());
294|
voteCountTilValgoppgjøerForFylkesvalgstyret.setVoteCountCategory(godkjentVoteCountForValgstyret.getVoteCountCategory()
); 295|
voteCountTilValgoppgjøerForFylkesvalgstyret.setCountQualifier(godkjentVoteCountForValgstyret.getCountQualifier());
296|
voteCountTilValgoppgjøerForFylkesvalgstyret.setApprovedBallots(godkjentVoteCountForValgstyret.getApprovedBallots());

```

## 6.221 LocaleConverter.java

### Issue 1A8061332CED1EF1A51E1CFE7A742E77:

#### Sink :

File admin-frontend/src/main/java/no/evote/converters/LocaleConverter.java 29

Fact Name: no.valg.eva.admin.frontend.LocaleController.getLocaleMap

```

28|     setLocaleController(context);
>29|     return localeController.getLocaleMap().get(value);
30| }
31|
32| @Override

```

## 6.222 MunicipalityController.java

### Issue 14B4E4EB6644F79740B80FD19626B0CB:

#### Sink :

File admin-frontend/src/main/java/no/valg/eva/admin/frontend/contexteditor/ctrls/MunicipalityController.java 118

Fact Name: no.evote.presentation.util.MessageUtil.buildFacesMessage

```

115|         mvAreaPickerController.update(AreaLevelEnum.MUNICIPALITY.getLevel(), null);
116|     }, EDIT_FORM)) {
117|         String[] summaryParams = {""};
>118|         MessageUtil.buildFacesMessage(null, MessageUtil.UPDATE_SUCCESSFUL_KEY, summaryParams,
FacesMessage.SEVERITY_INFO);
119|         hideDialog = true;

```

### Issue 49CA9A3C8E7EA4D5ADCA04EB7D031C83:

#### Sink :

File admin-frontend/src/main/java/no/valg/eva/admin/frontend/contexteditor/ctrls/MunicipalityController.java 75 Fact

Name: no.evote.presentation.util.MessageUtil.buildFacesMessage

```

72|         }, CREATE_FORM)) {
73|             hideDialog = true;
74|             String[] summaryParams = {newMunicipality.getName(), parentMvArea.getCountyName()};
>75|             MessageUtil.buildFacesMessage(null, MessageUtil.CREATE_SUBLEVEL_SUCCESSFUL_KEY, summaryParams,
FacesMessage.SEVERITY_INFO);

```

### Issue 512BA5832DF278C43412018D413A479A:

#### Sink :

File admin-frontend/src/main/java/no/valg/eva/admin/frontend/contexteditor/ctrls/MunicipalityController.java 61

Fact Name: no.evote.presentation.util.MessageUtil.buildFacesMessage

```

58| newMunicipality.setCounty(parentMvArea.getCounty());
59| if (!newMunicipality.getId().substring(0, 2).equalsIgnoreCase(parentMvArea.getCounty().getId())) {
60|     String[] summaryParams = {ID_LABEL, "@area_level[2].nummer", "@area.list_areas.message.id_two_first"};
>61|     MessageUtil.buildFacesMessage(CREATE_FORM, MessageUtil.AREA_ID_MUST_CONFORM, summaryParams,
FacesMessage.SEVERITY_ERROR);
62| } else {
63|     if (idExists(parentMvArea.getCounty(), newMunicipality)) {
64|         String[] summaryParams = {ID_LABEL, newMunicipality.getId(), parentMvArea.getCountyName()};

```

### Issue 598BC353EE328B365BA9272839F7DE1A:

#### Sink :

File admin-frontend/src/main/java/no/valg/eva/admin/frontend/contexteditor/ctrls/MunicipalityController.java 65 Fact

Name: no.evote.presentation.util.MessageUtil.buildFacesMessage

```

62| } else {
63|     if (idExists(parentMvArea.getCounty(), newMunicipality)) {
64|         String[] summaryParams = {ID_LABEL, newMunicipality.getId(), parentMvArea.getCountyName()};
>65|         MessageUtil.buildFacesMessage(CREATE_FORM, MessageUtil.CHOOSE_UNIQUE_ID, summaryParams,

```

```

FacesMessage.SEVERITY_ERROR);
66|     } else {
67|         newMunicipality.setCounty(parentMvArea.getCounty());
68|         newMunicipality.setElectronicMarkoffs(false);

```

**Issue A7406EA29A279F2BCF957134F06B879F:****Sink :****File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/contexteditor/ctrls/MunicipalityController.java 111 **Fact****Name:** no.evote.presentation.util.MessageUtil.buildFacesMessage

```

108| } else {
109|     if (hasIdChanged(municipality.getPk(), municipality) && idExists(parentMvArea.getCounty(), municipality)) {
110|         String[] summaryParams = {ID_LABEL, municipality.getId(), parentMvArea.getCountyName()};
>111|         MessageUtil.buildFacesMessage(EDIT_FORM, MessageUtil.CHOOSE_UNIQUE_ID, summaryParams,
FacesMessage.SEVERITY_ERROR);
112|     } else {
113|         if (execute() -> {
114|             currentMunicipality = municipalityService.update(userData, municipality);

```

**Issue D082F7AB52EEBE14DF8DC7269FFFE867:****Sink :****File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/contexteditor/ctrls/MunicipalityController.java 102**Fact Name:** no.evote.presentation.util.MessageUtil.buildFacesMessage

```

99|List<MvArea> mvAreaList = mvAreaService.findByPathAndChildLevel(currentMvArea);
100|if (hasIdChanged(municipality.getPk(), municipality) && (mvAreaList != null && !mvAreaList.isEmpty())) {
101|    String[] summaryParams = {ID_LABEL, "@area_level[3].name"};
>102|    MessageUtil.buildFacesMessage(EDIT_FORM, MessageUtil.EDIT_ID_NOT_ALLOWED, summaryParams,
FacesMessage.SEVERITY_ERROR);
103|    return;
104|}
105|if (!municipality.getId().substring(0, 2).equalsIgnoreCase(parentMvArea.getCounty().getId())) {

```

**Issue DA20FEE28B06755F2AF8111AD9BE9BA8:****Sink :****File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/contexteditor/ctrls/MunicipalityController.java 134**Fact Name:** no.evote.presentation.util.MessageUtil.buildFacesMessage

```

131| mvAreaPickerController.update(AreaLevelEnum.MUNICIPALITY.getLevel(), null);
132| )) {
133|     String[] summaryParams = {municipality.getName(), parentMvArea.getCountyName()};
>134|     MessageUtil.buildFacesMessage(null, MessageUtil.DELETE_FROM_LEVEL_SUCCESSFUL_KEY, summaryParams,
FacesMessage.SEVERITY_INFO);
135|     hideDialog = true;
136| }
137|} else {

```

**Issue DDFA47C3F62E4EAAF30983A7CCC4F6D7:****Sink :****File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/contexteditor/ctrls/MunicipalityController.java 139 **Fact****Name:** no.evote.presentation.util.MessageUtil.buildFacesMessage

```

137|} else {
138|     String[] summaryParams = {municipality.getName()};
>139|     MessageUtil.buildFacesMessage(null, MessageUtil.REMOVE_NOT_ALLOWED_KEY, summaryParams,
FacesMessage.SEVERITY_INFO);
140|     LOGGER.error("Municipality " + municipality.getName() + " with pk " + municipality.getPk() + " is not
deletable."); 141|}
142|getRequestContext().addCallbackParam("editMunicipalityHideDialog", hideDialog);

```

**Issue F7FFBCBBF1254E315938E27B10C11169:****Sink :****File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/contexteditor/ctrls/MunicipalityController.java 107**Fact Name:** no.evote.presentation.util.MessageUtil.buildFacesMessage

```

105|if (!municipality.getId().substring(0, 2).equalsIgnoreCase(parentMvArea.getCounty().getId())) {
106|    String[] summaryParams = {ID_LABEL, "@area_level[2].nummer", "@area.list_areas.message.id_two_first"};
>107|    MessageUtil.buildFacesMessage(CREATE_FORM, MessageUtil.AREA_ID_MUST_CONFORM, summaryParams,
FacesMessage.SEVERITY_ERROR);
108|} else {
109|    if (hasIdChanged(municipality.getPk(), municipality) && idExists(parentMvArea.getCounty(), municipality)) {
110|        String[] summaryParams = {ID_LABEL, municipality.getId(), parentMvArea.getCountyName()};

```

## 6.223 MvAreaPickerController.java

### Issue 7558C61AF71946B9A6636307B94DD6F3:

**Sink :**

**File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/area/ctrls/MvAreaPickerController.java 524

**Fact Name:** no.evote.service.configuration.MvAreaService.findSingleByPath

```
521|    contextPickerRoot = userData.getOperatorRole().getMvArea();
522|    if (contextPickerRoot.getAreaLevel() > includeAreasAboveMyLevel) {
523|        AreaPath areaPath =
AreaPath.from(contextPickerRoot.getAreaPath()).toAreaLevelPath(AreaLevelEnum.getLevel(includeAreasAboveMyLevel));
>524|    contextPickerRoot = mvAreaService.findSingleByPath(areaPath);
525|    }
526|} else {
527|    contextPickerRoot = userData.getOperatorRole().getMvArea();
```

### Issue BFE8D5C49EC8E78BC6DB4A618B459554:

**Sink :**

**File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/area/ctrls/MvAreaPickerController.java 240

**Fact Name:** no.evote.service.configuration.MvAreaService.findSingleByPath

```
237|int currentIndex = level - contextPickerRoot.getAreaLevel() - 1;
238|mvAreaPickerTables.get(currentIndex).setMvAreas(mvAreaService.findByPathAndChildLevel(parentMvArea)); 239|if
(!StringUtils.isEmpty(areaPath)) {
>240|    MvArea mvArea = mvAreaService.findSingleByPath(areaPath);
241|    if (mvArea != null && mvAreaPickerTables.get(currentIndex) != null &&
mvAreaPickerTables.get(currentIndex).getMvAreas() != null
242|        && mvAreaPickerTables.get(currentIndex).getMvAreas().contains(mvArea)) {
243|        mvAreaPickerTables.get(currentIndex).setSelectedMvArea(mvArea);
```

## 6.224 MvAreaRepository.java

### Issue 1F1B33BD877335C55BE6191CA0D83BF3:

**Sink :**

**File** admin-configuration/src/main/java/no/valg/eva/admin/configuration/repository/MvAreaRepository.java 46

**Fact Name:** no.valg.eva.admin.configuration.repository.MvAreaRepository.findByPathAndLevel

```
44|public List<MvArea> findByPathAndChildLevel(MvArea mvArea) {
45|    if (mvArea != null && mvArea.getElectionEvent() != null) {
>46|        return findByPathAndLevel(mvArea.getAreaPath(), mvArea.getAreaLevel() + 1); 47|    }
48|    return Collections.emptyList();
```

### Issue 39571599E3D41FoFoE3D2171CoC7CFoC:

**Sink :**

**File** admin-configuration/src/main/java/no/valg/eva/admin/configuration/repository/MvAreaRepository.java 60

**Fact Name:** no.valg.eva.admin.configuration.repository.MvAreaRepository.findByPathAndLevel

```
59|public List<MvArea> findByPathAndLevel(AreaPath areaPath, AreaLevelEnum areaLevelEnum) { >60|    return
findByPathAndLevel(areaPath.path(), areaLevelEnum.getLevel());
61|}
62|
63|@Deprecated
```

### Issue 50A1838A9D301B9F5C3FC08238580485:

**Sink :**

**File** admin-configuration/src/main/java/no/valg/eva/admin/configuration/repository/MvAreaRepository.java 52

**Fact Name:** no.valg.eva.admin.configuration.repository.MvAreaRepository.findByPathAndLevel

```
51|public List<MvArea> findByPathAndChildLevel(AreaPath areaPath) {
>52|    return findByPathAndLevel(areaPath.path(), areaPath.getLevel().getLevel() + 1);
53|}
54|
55|public List<MvArea> finnFor(ValggeografiSti valggeografiSti, AreaLevelEnum omraadenivaa) {
```

### Issue 95BDB625ACFD70B7DD00D89E52E12DCE:

**Sink :**

**File** admin-configuration/src/main/java/no/valg/eva/admin/configuration/repository/MvAreaRepository.java 153

**Fact Name:** no.valg.eva.admin.configuration.repository.MvAreaRepository.findSingleByPath

```

152|public MvArea findSingleByPath(AreaPath areaPath) {
>153|    return findSingleByPath(areaPath.path());
154|}
155|
156|public MvArea findSingleByPath(String electionEventId, AreaPath areaPath) {

```

## 6.225 MvAreaServiceEjb.java

**Issue** 5C485FA3DC43CBBE1640D401F2421CB9:

**Sink :**

**File** admin-backend/src/main/java/no/evote/service/configuration/MvAreaServiceEjb.java 75

**Fact** Name: no.valg.eva.admin.configuration.repository.MvAreaRepository.findByPathAndLevel

```

72|@Override
73|@SecurityNone
74|public List<MvArea> findByPathAndLevel(String path, int level) {
>75|    return mvAreaRepository.findByPathAndLevel(path, level); 76|}
77|
78|@Override

```

**Issue** 660B240C43CD481D5F5C54B64DB10576:

**Sink :**

**File** admin-backend/src/main/java/no/evote/service/configuration/MvAreaServiceEjb.java 88

**Fact** Name: no.valg.eva.admin.configuration.repository.MvAreaRepository.findSingleByPath

```

85|@SecurityNone
86|@Deprecated
87|public MvArea findSingleByPath(String path) {
>88|    return mvAreaRepository.findSingleByPath(path);
89|}
90|
91|@Override

```

## 6.226 MvElectionServiceEjb.java

**Issue** 18323D13E64126EA3825BEACC7130EF1:

**Sink :**

**File** admin-backend/src/main/java/no/evote/service/configuration/MvElectionServiceEjb.java 98 **Fact**

**Name:** no.evote.service.configuration.MvElectionServiceBean.findSingleByPathMinimal

```

95|@Override
96|@SecurityNone
97|public MvElectionMinimal findSingleByPathMinimal(UserData userData, String path) { > 98|    return
mvElectionService.findSingleByPathMinimal(userData, path);
99|}
100|
101|@Override

```

## 6.227 OperatorImportHelper.java

**Issue** 0BF57FF6D121FA0A0178A566179BDEBD:

**Sink :**

**File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/rbac/OperatorImportHelper.java 158 **Fact**

**Name:** no.evote.presentation.util.MessageUtil.buildFacesMessage

```

155|    try {
156|        return getInputStream();
157|    } catch (IOException e) {
>158|        MessageUtil.buildFacesMessage(null, e.getMessage(), null, FacesMessage.SEVERITY_ERROR); 159|
return null;

```

## 6.228 PollingDistrictController.java

**Issue** 082DB2A1797078EBD83BB7CA01501967:

**Sink :**

**File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/contexteditor/ctrls/PollingDistrictController.java 128

**Fact** Name: no.evote.service.configuration.LegacyPollingDistrictService.update

```

125|     MessageUtil.buildFacesMessage(EDIT_FORM, MessageUtil.CHOOSE_UNIQUE_ID, summaryParams,
FacesMessage.SEVERITY_ERROR);
126| } else {
127|     if (execute() -> {
>128|         pollingDistrict = pollingDistrictService.update(getUserData(), pollingDistrict);
129|         mvAreaPickerController.update(AreaLevelEnum.POLLING_DISTRICT.getLevel(), null); 130|     }, EDIT_FORM)) {
131|         String[] summaryParams = { "" };

```

**Issue 1EF66684E26528C669C91D4C6E893445:****Sink :****File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/contexteditor/ctrls/PollingDistrictController.java 101**Fact** Name: no.evote.presentation.util.MessageUtil.buildFacesMessage

```

98|         mvAreaPickerController.update(AreaLevelEnum.POLLING_DISTRICT.getLevel(), parentMvArea.getAreaPath() + "."
+ pollingDistrict.getId());
99|     }, CREATE_FORM)) {
100|         String[] summaryParams = { pollingDistrict.getName(), parentMvArea.getMunicipalityName() + " - " +
parentMvArea.getBoroughName() };
>101|         MessageUtil.buildFacesMessage(null, MessageUtil.CREATE_SUBLEVEL_SUCCESSFUL_KEY, summaryParams,
FacesMessage.SEVERITY_INFO);
102|         hideDialog = true;

```

**Issue 36724C131D91F9186489FA865E7C84F9:****Sink :****File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/contexteditor/ctrls/PollingDistrictController.java 147**Fact** Name: no.evote.presentation.util.MessageUtil.buildFacesMessage

```

144|         mvAreaPickerController.update(AreaLevelEnum.POLLING_DISTRICT.getLevel(), null);
145|     }) {
146|         String[] summaryParams = { pollingDistrict.getName(), parentMvArea.getMunicipalityName() + " - " +
parentMvArea.getBoroughName() };
>147|         MessageUtil.buildFacesMessage(null, MessageUtil.DELETE_FROM_LEVEL_SUCCESSFUL_KEY, summaryParams,
FacesMessage.SEVERITY_INFO);
148|         hideDialog = true;
149|     }
150| } else {

```

**Issue 4CFA000621299D6D76C0A6C4284E9FF1:****Sink :****File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/contexteditor/ctrls/PollingDistrictController.java 152 **Fact**

Name: no.evote.presentation.util.MessageUtil.buildFacesMessage

```

150| } else {
151|     String[] summaryParams = { pollingDistrict.getName() };
>152|     MessageUtil.buildFacesMessage(null, MessageUtil.REMOVE_NOT_ALLOWED_KEY, summaryParams,
FacesMessage.SEVERITY_INFO);
153|     LOGGER.error("Polling district " + pollingDistrict.getName() + " with pk " + pollingDistrict.getPk() + " is
not deletable."); 154| }
155| getRequestContext().addCallbackParam("editPollingDistrictHideDialog", hideDialog);

```

**Issue 65392881C6B24A03B055167585A9CF52:****Sink :****File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/contexteditor/ctrls/PollingDistrictController.java 132**Fact** Name: no.evote.presentation.util.MessageUtil.buildFacesMessage

```

129|         mvAreaPickerController.update(AreaLevelEnum.POLLING_DISTRICT.getLevel(), null); 130|     }, EDIT_FORM)) {
131|         String[] summaryParams = { "" };
>132|         MessageUtil.buildFacesMessage(null, MessageUtil.UPDATE_SUCCESSFUL_KEY, summaryParams,
FacesMessage.SEVERITY_INFO);
133|         hideDialog = true;

```

**Issue 92B731D2BBAB37DA56FCC86C98847BA2:****Sink :****File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/contexteditor/ctrls/PollingDistrictController.java 97 **Fact**

Name: no.evote.service.configuration.LegacyPollingDistrictService.create

```

95| pollingDistrict.setBorough(parentMvArea.getBorough());
96| if (execute() -> {
> 97|     pollingDistrictService.create(getUserData(), pollingDistrict);
98|     mvAreaPickerController.update(AreaLevelEnum.POLLING_DISTRICT.getLevel(), parentMvArea.getAreaPath() + "."
+ pollingDistrict.getId());
99| }, CREATE_FORM)) {
100|     String[] summaryParams = { pollingDistrict.getName(), parentMvArea.getMunicipalityName() + " - " +
parentMvArea.getBoroughName() };

```

**Issue A2A84D43FA5A007CE6833D964D25D8D0:****Sink :**

**File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/contexteditor/ctrls/PollingDistrictController.java 125 **Fact**  
**Name:** no.evote.presentation.util.MessageUtil.buildFacesMessage

```

122|boolean hideDialog = false;
123|if (hasIdChanged(pollingDistrict.getPk(), pollingDistrict) && idExists(parentMvArea.getBorough(),
pollingDistrict)) {
124|    String[] summaryParams = { ID_LABEL, pollingDistrict.getId(), parentMvArea.getMunicipalityName() + " - " +
parentMvArea.getBoroughName() };
>125|    MessageUtil.buildFacesMessage(EDIT_FORM, MessageUtil.CHOOSE_UNIQUE_ID, summaryParams,
FacesMessage.SEVERITY_ERROR);
126|} else {
127|    if (execute() -> {
128|        pollingDistrict = pollingDistrictService.update(getUserData(), pollingDistrict);

```

**Issue C691079322FE06FA590F6AC51DB896F6:****Sink :**

**File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/contexteditor/ctrls/PollingDistrictController.java 143 **Fact**  
**Name:** no.evote.service.configuration.LegacyPollingDistrictService.delete

```

140|boolean hideDialog = false;
141|if (getCurrentRemovable()) {
142|    if (execute() -> {
>143|        pollingDistrictService.delete(getUserData(), pollingDistrict);
144|        mvAreaPickerController.update(AreaLevelEnum.POLLING_DISTRICT.getLevel(), null); 145|    }) {
146|        String[] summaryParams = { pollingDistrict.getName(), parentMvArea.getMunicipalityName() + " - " +
parentMvArea.getBoroughName() };

```

**Issue F4D2DC699B3764D69E81A3A81E26CC57:****Sink :**

**File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/contexteditor/ctrls/PollingDistrictController.java 90  
**Fact** **Name:** no.evote.presentation.util.MessageUtil.buildFacesMessage

```

88|if (idExists(parentMvArea.getBorough(), pollingDistrict)) {
89|    String[] summaryParams = { ID_LABEL, pollingDistrict.getId(), parentMvArea.getBoroughName() };
>90|    MessageUtil.buildFacesMessage(CREATE_FORM, MessageUtil.CHOOSE_UNIQUE_ID, summaryParams,
FacesMessage.SEVERITY_ERROR);
91|} else {
92|    if (pollingDistrictType == POLLING_DISTRICT_TYPE_MUNICIPALITY && typeMunicipalityCanBeCreated) { 93|
pollingDistrict.setMunicipality(true);

```

## 6.229 RedigerListeforslagController.java

**Issue A0880AB711914ED862542993CCD9D70E:****Sink :**

**File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/listproposal/ctrls/RedigerListeforslagController.java 313  
**Fact** **Name:** no.evote.service.configuration.ContestAreaService.findContestAreasForContest

```

312|private Set<MvArea> getMvAreasFromContest() {
>313|    List<ContestArea> contestAreaList = contestAreaService.findContestAreasForContest(getContest().getPk());
314|    Set<MvArea> mvareas = new HashSet<>();
315|    for (ContestArea contestArea : contestAreaList) {
316|        mvareas.add(contestArea.getMvArea());

```

## 6.230 RejectedVotingsReportController.java

**Issue 3DC39B717C921B818B3C792A0239AF28:****Sink :**

**File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/stemmegivning/ctrls/RejectedVotingsReportController.java 62 **Fact**  
**Name:** no.evote.service.voting.VotingService.update

```

59|} else {
60|    selectedRejectedVotings.setVotingRejection(null);
61|    selectedRejectedVotings.setValidationTimestamp(null);
>62|    votingService.update(getUserData(), selectedRejectedVotings);
63|    showMessage(getMessageProvider().get("@voting.approveBallot.undoRejectionResponse",
selectedRejectedVotings.getVoter().getNameLine()),
64|        FacesMessage.SEVERITY_INFO);
65|    rejectedVotingsList = votingService.getRejectedVotingsByElectionGroupAndMunicipality(getUserData(),
valggruppeSti, kommuneSti);

```



## 6.231 ReportTemplateUploader.java

### Issue 4CCB2543BA053C21CCB3006BA2803FE2:

**Sink :**

**File** admin-backend/src/main/java/no/valg/eva/admin/backend/reporting/jasperserver/ReportTemplateUploader.java 271

**Fact** Name: no.valg.eva.admin.configuration.repository.MvAreaRepository.findByPathAndLevel

```
268|List<Map<String, String>> allMunicipalitiesParameters = new ArrayList<>();
269|List<Map<String, String>> allCountiesParameters = new ArrayList<>();
270|for (String electionEventId : nightJobsForElectionEvents) {
>271|     List<MvArea> mvAreasForMunicipalities = mvAreaRepository.findByPathAndLevel(electionEventId,
MUNICIPALITY.getLevel());
272|     allMunicipalitiesParameters.addAll(parametersForAllMvAreas(mvAreasForMunicipalities, MUNICIPALITY));
273|     List<MvArea> mvAreasForCounties = mvAreaRepository.findByPathAndLevel(electionEventId, COUNTY.getLevel());
274|     allCountiesParameters.addAll(parametersForAllMvAreas(mvAreasForCounties, COUNTY));
```

### Issue 5071E23A3921B1169A8EF900BE47D87A:

**Sink :**

**File** admin-backend/src/main/java/no/valg/eva/admin/backend/reporting/jasperserver/ReportTemplateUploader.java 273 **Fact**

Name: no.valg.eva.admin.configuration.repository.MvAreaRepository.findByPathAndLevel

```
270|for (String electionEventId : nightJobsForElectionEvents) {
271|     List<MvArea> mvAreasForMunicipalities = mvAreaRepository.findByPathAndLevel(electionEventId,
MUNICIPALITY.getLevel());
272|     allMunicipalitiesParameters.addAll(parametersForAllMvAreas(mvAreasForMunicipalities, MUNICIPALITY));
>273|     List<MvArea> mvAreasForCounties = mvAreaRepository.findByPathAndLevel(electionEventId, COUNTY.getLevel());
274|     allCountiesParameters.addAll(parametersForAllMvAreas(mvAreasForCounties, COUNTY));
275|}
276|reportsUploader
```

## 6.232 ReportingUnitServiceBean.java

### Issue 2CF66108BF44D5A36409DD42D6AFFC8E:

**Sink :**

**File** admin-backend/src/main/java/no/evote/service/configuration/ReportingUnitServiceBean.java 24

**Fact** Name: no.valg.eva.admin.configuration.repository.ReportingUnitRepository.getReportingUnit

```
21|public List<ReportingUnit> getAccessibleReportingUnits(MvElection mvElection, MvArea mvArea) {
22|     List<ReportingUnit> reportingUnits = new ArrayList<>();
23|     for (ContestRelArea cra : contestRelAreaRepository.findAllAllowed(mvElection, mvArea)) {
>24|         ReportingUnit ru = reportingUnitRepository.getReportingUnit(cra);
25|         if (ru != null && !reportingUnits.contains(ru)) {
26|             reportingUnits.add(ru);
```

## 6.233 RootLevelEdit.java

### Issue 498C5E58330448C62D1B34D0CC75FC1F:

**Sink :**

**File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/rbac/ctrls/RootLevelEdit.java 57 **Fact**

Name: no.evote.service.configuration.MvAreaService.findByPathAndLevel

```
54|         AreaPath path = AreaPath.from(getCounty());
55|         ctrl.getAreaOptions().init(path);
56|     } else {
>57|         municipalityList = ctrl.getMvAreaService().findByPathAndLevel(getCounty(),
AreaLevelEnum.MUNICIPALITY.getLevel());
```

### Issue D59FB6246EF575F7375E6BD6B433B8AD:

**Sink :**

**File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/rbac/ctrls/RootLevelEdit.java 19 **Fact**

Name: no.evote.service.configuration.MvAreaService.findByPathAndLevel

```
17|public RootLevelEdit(OperatorEditController ctrl) {
18|     this.ctrl = ctrl;
>19|     countyList = ctrl.getMvAreaService().findByPathAndLevel(
20|         ctrl.getUserData().getOperatorAreaPath().path(), AreaLevelEnum.COUNTY.getLevel());
```

## 6.234 SkrivUtValgkortController.java

**Issue** 88DF8A059D3A2AC2A10E9D5935C1D63A:

**Sink :**

**File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/manntall/ctrls/SkrivUtValgkortController.java 56 **Fact**  
Name: no.evote.service.configuration.MvAreaService.findSingleByPath

```
53|MvArea brukOmrade = omrade;
54|if (velger.getMvArea() != null && velger.getMvArea().getAreaLevel() == AreaLevelEnum.POLLING_DISTRICT.getLevel())
{
55|    AreaPath path = velger.getMvArea().areaPath().toMunicipalityPath();
>56|    brukOmrade = mvAreaService.findSingleByPath(path);
57|}
58|byte[] bytes = specialPurposeReportService.generateElectionCard(userData, velger.getPk(), brukOmrade, valgGruppe);
59|try {
```

## 6.235 StatisticReportingController.java

**Issue** D3CC2B846E4A328A53EB96435EEF2E4C:

**Sink :**

**File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/counting/ctrls/StatisticReportingController.java 113 **Fact**  
Name: no.evote.service.configuration.MvElectionService.findSingleByPath

```
111|private void initValgnattrapporteringMetadata() {
112|    if ((selectedArea != null || getUserData().isSameElectionCountyUser()) && contestInfo != null) {
>113|        mvElectionContest = getMvElectionService().findSingleByPath(contestInfo.getElectionPath()); 114|
contest = mvElectionContest.getContest();
115|
116|        if (contest != null && selectedArea.getMunicipality() != null &&
getUserDataController().getUserAccess().isResultatRapporterer()) {
```

## 6.236 StatusApplicationServiceApi.java

**Issue** DE0F9EDF2924F11CB2EDE310ABF10036:

**Sink :**

**File** admin-backend/src/main/java/no/valg/eva/admin/backend/application/service/StatusApplicationServiceApi.java 31 **Fact**  
Name: no.valg.eva.admin.backend.application.service.StatusApplicationService.getStatus

```
28|@SecurityNone
29|@Deprecated // Use getStatusProperties() instead 30|public String getStatus() {
>31|    return statusApplicationService.getStatus();
32|}
33|
34|@Override
```

## 6.237 ValgdistriktOpptellingskategoriOgValggeografiHolder.java

**Issue** 32F507E8982744F89F75D5EE8C7C9CE7:

**Sink :**

**File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/common/ValgdistriktOpptellingskategoriOgValggeografiHolder.java 69  
**Fact** Name: no.evote.service.configuration.MvAreaService.findSingleByPath

```
66|    String areaPaths = getRequestParameter("areaPath");
67|    if (areaPaths != null) {
68|        selectedAreaPath = AreaPath.from(areaPaths);
>69|        selectedMvArea = mvAreaService.findSingleByPath(selectedAreaPath);
```

**Issue** 92E885D808782D56D24B9EBCAA1A575A:

**Sink :**

**File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/common/ValgdistriktOpptellingskategoriOgValggeografiHolder.java 60  
**Fact** Name: no.evote.service.configuration.MvElectionService.findSingleByPath

```

57|if (contestPath != null) {
58|    selectedElectionPath = ElectionPath.from(contestPath);
59|    selectedContestInfo = contestInfoService.findContestInfoByPath(selectedElectionPath);
>60|    selectedMvElection = mvElectionService.findSingleByPath(selectedElectionPath);
61|}
62|String countCategoryId = getRequestParameter("category");
63|if (countCategoryId != null) {

```

## 6.238 VoteCountBuilder.java

### Issue FD420292E5105AC5D792CBC26FF6D77E:

Sink :

File admin-counting/src/main/java/no/valg/eva/admin/counting/builder/VoteCountBuilder.java 35

Fact Name: no.valg.eva.admin.counting.domain.model.VoteCount.setPollingDistrict

```

32|    voteCount.setMvArea(mvArea);
33|    // setter også polling district for å være bakoverkompatibel med f.eks. funksjon for å slette telldata
34|    // mvArea er pt. alltid polling district
>35|    voteCount.setPollingDistrict(mvArea.getPollingDistrict());
36|    return this;

```

## 6.239 VotingConfirmationVoterController.java

### Issue A3A016E6347AA04F84121E16A8F52352:

Sink :

File admin-frontend/src/main/java/no/valg/eva/admin/frontend/stemmegivning/ctrls/proving/VotingConfirmationVoterController.java 163

Fact Name: no.evote.service.voting.VotingService.update

```

160|execute() -> {
161|    rejectedVoting.setVotingRejection(null);
162|    rejectedVoting.setValidationTimestamp(null);
>163|    votingService.update(getUserData(), rejectedVoting);
164|    buildDetailMessage("@voting.approveBallot.undoRejectionResponse", new String[] { getVelger().getNameLine() },
SEVERITY_INFO);
165|    populateVotingListsForVoter(fetchVoterVotings());
166|});

```

### Issue E8D4A094ABF0516FFA69522F83CDA684:

Sink :

File admin-frontend/src/main/java/no/valg/eva/admin/frontend/stemmegivning/ctrls/proving/VotingConfirmationVoterController.java 152

Fact Name: no.evote.service.voting.VotingService.update

```

149|execute() -> {
150|    voting.setVotingRejection(findRejectionReason());
151|    voting.setValidationTimestamp(DateTime.now());
>152|    votingService.update(getUserData(), voting);
153|    rejectionReason = null;
154|    buildDetailMessage("@voting.approveBallot.votingRejected", new String[] {
voting.getVotingRejection().getName() }, SEVERITY_INFO); 155|    populateVotingListsForVoter(fetchVoterVotings());

```

## 6.240 VotingServiceBean.java

### Issue 94033EED3564CCBB9055250BF227128B:

Sink :

File admin-voting/src/main/java/no/valg/eva/admin/voting/service/VotingServiceBean.java 572

Fact Name: no.valg.eva.admin.voting.repository.VotingRepository.findAdvanceVotingCategories

```

569|private VelgerSomSkalStemme initVelgerMeldinger(StemmegivningsType stemmegivningsType, MvArea stemmested, Voter
velger) {
570|    List<VotingCategory> liste = new ArrayList<>();
571|    if (stemmegivningsType.isForhand() && isVelgerSammeKommune(stemmested.getMunicipality(), velger)) {
>572|        liste.addAll(votingRepository.findAdvanceVotingCategories());
573|    } else if (stemmegivningsType.isValgtingSentralt()) {
574|        if (stemmested.getMunicipality().isElectronicMarkoffs()) {
575|            liste.add(votingRepository.findVotingCategoryById(VB.getId()));

```

## 6.241 VotingServiceEjb.java

### Issue 08E96468233E4ABE47C41C632DEF81E2:

Sink :

**File** admin-backend/src/main/java/no/valg/eva/admin/backend/service/VotingServiceEjb.java 232 **Fact**  
**Name:** no.valg.eva.admin.voting.repository.VotingRepository.findAllVotingCategories

```
229|@Override
230|@Security(accesses = Beskyttet_Slett_Stemmegiving, type = READ)
231|public List<VotingCategory> findAllVotingCategories(UserData userData) {
>232|     return votingRepository.findAllVotingCategories();
233|}
234|
235|@Override
```

### Issue 109FC3D3F98F1BC14D3C91FBED338C84:

**Sink :**

**File** admin-backend/src/main/java/no/valg/eva/admin/backend/service/VotingServiceEjb.java 202 **Fact**  
**Name:** no.valg.eva.admin.voting.repository.VotingRepository.findAdvanceVotingCategories

```
199|@Override
200|@Security(accesses = Aggregert_Stemmegiving_Registrer, type = READ)
201|public List<VotingCategory> findAdvanceVotingCategories(UserData userData) { >202|     return
votingRepository.findAdvanceVotingCategories(); 203|}
204|
205|@Override
```

## 6.242 AbstractPickerController.java

### Issue 92F200191AE725D07CCF8F9999EB13F7:

**Sink :**

**File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/picker/ctrls/AbstractPickerController.java 35 **Fact**  
**Name:** ~tp~4

```
33|requestParameterMap = new HashMap<>();
34|if (cfg.isKeepRequestParams()) {
>35|     for (String parameterName : getCfg().getKeepRequestParams()) {
36|         String parameterValue = getRequestParameter(parameterName); 37|         if (parameterValue != null &&
!parameterValue.isEmpty()) {
38|             requestParameterMap.put(parameterName, parameterValue);
```

### Issue 9E57ADF74282376CD8E8A23B3D32A60D:

**Sink :**

**File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/picker/ctrls/AbstractPickerController.java 77 **Fact**  
**Name:** ~tp~0

```
76|     public ElectionCfg getElectionCfg() {
>77|         return cfg.getElectionCfg();
```

### Issue A7AA3360FC90E11BA1F58BBCF5B18829:

**Sink :**

**File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/picker/ctrls/AbstractPickerController.java 73 **Fact**  
**Name:** ~tp~0

```
72|public AreaCfg getAreaCfg() {
>73|     return cfg.getAreaCfg();
74|}
75|
76|public ElectionCfg getElectionCfg() {
```

## 6.243 AreaPickerController.java

### Issue 56DAC7E14C9F8776678FB387AF7EoAB9:

**Sink :**

**File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/picker/ctrls/AreaPickerController.java 32 **Fact**  
**Name:** ~tp~6

```
29|     appendRequestParam(result, "areaPath", areaPicker.getSelectedMvArea().getAreaPath());
30|}
31|
>32|if (getCfg().isKeepRequestParams()) { 33|     appendExistingRequestParams(result);
```

## 6.244 ContextPickerCfg.java

**Issue D7C9EDCFA23A82A9FBB43F2A357C7C89:**

**Sink :**

**File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/picker/cfg/ContextPickerCfg.java 14 **Fact**

Name: ~to

```
11|private List<String> keepRequestParams;
12|
13|public ContextPickerCfg() {
>14|     areaCfg = new AreaCfg();
15|     electionCfg = new ElectionCfg();
16|     keepRequestParams = new ArrayList<>();
```

**Issue D7C9EDCFA23A82A9FBB43F2A357C7C8A:**

**Sink :**

**File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/picker/cfg/ContextPickerCfg.java 15 **Fact**

Name: ~ti

```
13|public ContextPickerCfg() {
14|     areaCfg = new AreaCfg();
>15|     electionCfg = new ElectionCfg();
16|     keepRequestParams = new ArrayList<>();
```

## 6.245 ElectionPickerController.java

**Issue 61E96A7A0B0380491F84CD6ACA63736D:**

**Sink :**

**File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/picker/ctrls/ElectionPickerController.java 32 **Fact**

Name: ~tp~6

```
29|     appendRequestParam(result, "electionPath", electionPicker.getSelectedMvElection().getElectionPath());
30|}
31|
>32|if (getCfg().isKeepRequestParams()) {
33|     appendExistingRequestParams(result);
```

## 6.246 MvAreaPickerTable.java

**Issue 8C326595AB61E7FFBE1BFB433E9AF9BC:**

**Sink :**

**File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/area/MvAreaPickerTable.java 57 **Fact**

Name: filter

```
54|public void setMvAreas(final List<MvArea> mvAreas) {
55|     this.mvAreas = mvAreas;
56|     if (mvAreaFilters != null && this.mvAreas != null && !mvAreaFilters.isEmpty()) {
>57|         for (MvAreaFilter filter : mvAreaFilters) {
58|             this.mvAreas = filter.filter(this.mvAreas, level);
```

### [Low] Poor Error Handling / Overly Broad Catch

*Errors*

**Analysis:** Bad Practice

#### Abstract

The catch block handles a broad swath of exceptions, potentially trapping dissimilar issues or problems that should not be dealt with at this point in the program.

## Explanation

Multiple catch blocks can get ugly and repetitive, but "condensing" catch blocks by catching a high-level class like `Exception` can obscure exceptions that deserve special treatment or that should not be caught at this point in the program. Catching an overly broad exception essentially defeats the purpose of Java's typed exceptions, and can become particularly dangerous if the program grows and begins to throw new types of exceptions. The new exception types will not receive any attention.

**Example:** The following code excerpt handles three types of exceptions in an identical fashion.

```
try {
    doExchange();
}
catch (IOException e) {
    logger.error("doExchange failed", e);
} catch (InvocationTargetException e) {    logger.error("doExchange failed", e);
}
catch (SQLException e) {
    logger.error("doExchange failed", e); }
```

At first blush, it may seem preferable to deal with these exceptions in a single catch block, as follows:

```
try {
    doExchange();
}
catch (Exception e) {
    logger.error("doExchange failed", e); }
```

However, if `doExchange()` is modified to throw a new type of exception that should be handled in some different kind of way, the broad catch block will prevent the compiler from pointing out the situation. Further, the new catch block will now also handle exceptions derived from `RuntimeException` such as `ClassCastException`, and `NullPointerException`, which is not the programmer's intent.

## Recommendation

Do not catch broad exception classes like `Exception`, `Throwable`, `Error`, or `<RuntimeException>` except at the very top level of the program or thread.

## 6.247 AuditLogInterceptor.java

**Issue** 571E28DC48902958163ECDBB9A168BBC:

**Sink:**

**File** admin-backend/src/main/java/no/valg/eva/admin/backend/auditlog/AuditLogInterceptor.java 57-59

```
54|         if (!auditEvent.muteEvent()) {
55|             auditLogService.addToAuditTrail(auditEvent);
56|         }
>57|     } catch (Exception exceptionWhenAuditLogging) {
>58|         logger.error("Unable to audit log successful method invocation", exceptionWhenAuditLogging);
>59|     }
60|     return returnValue;
61| } catch (Exception e) {
62|     try {
```

**Issue** 571E28DC48902958163ECDBB9A168BBD:

**Sink:**

**File** admin-backend/src/main/java/no/valg/eva/admin/backend/auditlog/AuditLogInterceptor.java 65-67

```
62|     try {
63|         AbstractAuditEvent auditEvent = auditEventFactory.buildErrorAuditEvent(e.getMessage());
64|         auditLogService.addToAuditTrail(auditEvent);
>65|     } catch (Exception exceptionWhenAuditLogging) {
>66|         logger.error("Unable to audit log failed method invocation", exceptionWhenAuditLogging);
>67|     }
68|     throw e;
```

## 6.248 BaseController.java

**Issue** 182B813D82B935AF72994C6A642DAB96:

**Sink :**

**File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/BaseController.java 163-166

```
160|protected boolean execute(Service service, ErrorCodeHandler handler) {
161|    try {
162|        service.execute();
>163|    } catch (RuntimeException e) {
>164|        process(e, handler, null);
>165|        return false;
>166|    }
167|    return true;
```

**Issue** 2226D8D173AE953A71809542E233893C:

**Sink :**

**File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/BaseController.java 143-146

```
140|protected boolean execute(Service service) {
141|    try {
142|        service.execute();
>143|    } catch (RuntimeException e) {
>144|        process(e, null, null);
>145|        return false;
>146|    }
147|    return true;
```

**Issue** 9555EAECD580E117B4A4D264D9D55608:

**Sink :**

**File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/BaseController.java 153-156

```
150|protected boolean execute(Service service, String clientId) {
151|    try {
152|        service.execute();
>153|    } catch (RuntimeException e) {
>154|        process(e, null, clientId);
>155|        return false;
>156|    }
157|    return true;
```

## 6.249 BaseCountController.java

**Issue** 786525E9F830AEE028BC3A10FC59DBC9:

**Sink :**

**File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/counting/ctrls/BaseCountController.java 36-38

```
33|    try {
34|        process((RuntimeException) e);
35|        return;
>36|    } catch (Exception ee) {
>37|        LOGGER.trace(ee.getMessage());
>38|    }
39|}
40|if (e instanceof ValidateException) {
41|    LOGGER.debug(e.getMessage(), e);
```

## 6.250 BaseFinalCountController.java

**Issue** 9FBFoDA937199CoE56DD88F775E147C2:

**Sink :**

**File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/counting/ctrls/BaseFinalCountController.java 66-68

```
63|        if (message) {
64|            MessageUtil.buildDetailMessage(FacesMessage.SEVERITY_INFO,
getMessageProvider().get("@count.isSaved"));
65|        }
>66|    } catch (Exception e) {
>67|        logAndBuildDetailErrorMessage(e);
```

```

>68|     }
69|}
70|
71|@Override

```

## 6.251 BatchController.java

Issue B19A2C6DC43E73E820D421BA79F1532D:

Sink :

File admin-frontend/src/main/java/no/valg/eva/admin/frontend/counting/ctrls/BatchController.java 92-96

```

89|     buildFacesMessageAndLogException(e);
90| } catch (EvoteNoRollbackException e) {
91|     buildFacesMessageAndLogException(e);
>92| } catch (Exception e) {
>93|     String message = messageProvider.get("@count.error.upload.io") + ": " + e.getMessage();
>94|     LOGGER.error(message, e);
>95|     MessageUtil.buildDetailMessage(message, FacesMessage.SEVERITY_ERROR);
>96| } finally {
97|     if (stream != null) {
98|         try {
99|             stream.close();

```

## 6.252 BoroughsListProposalConfigurationController.java

Issue F9912326663FB060A11A7FF50C974F43:

Sink :

File admin-frontend/src/main/java/no/valg/eva/admin/frontend/configuration/ctrls/local/BoroughsListProposalConfigurationController.java 77-79

```

74| if (data == null) {
75|     try {
76|         data = loadData();
>77|     } catch (RuntimeException e) {
>78|         return false;
>79|     }
80|
81| }
82| return data != null && !data.getChildren().isEmpty() && data.isSingleArea();

```

## 6.253 CandidateJsonBuilder.java

Issue 42691D67DE8FB49A9BC24919022C1C49:

Sink :

File admin-common/src/main/java/no/valg/eva/admin/common/auditlog/auditevents/CandidateJsonBuilder.java 70-72

```

69|         return null;
>70|     } catch (Exception e) {
>71|         return null; // in case of uninitialized proxies
>72|     }
73| }
74|
75| private String getPartyId(Candidate candidate) {

```

Issue BE1AF8812D600D52E6D3123784F8357C:

Sink :

File admin-common/src/main/java/no/valg/eva/admin/common/auditlog/auditevents/CandidateJsonBuilder.java 83-85

```

82|         return null;
>83|     } catch (Exception e) {
>84|         return null; // in case of uninitialized proxies

```

## 6.254 CompareCountsController.java

Issue 5EDA03ABBBB8040680915FD17E6323F0:

Sink :

File admin-frontend/src/main/java/no/valg/eva/admin/frontend/counting/ctrls/CompareCountsController.java 143-145

```

140|         updateCountList(count);

```



```

141|         MessageUtil.buildDetailMessage(FacesMessage.SEVERITY_INFO,
getMessageProvider().get("@count.isApproved"));
142|         setupDefaultCompare();
>143|     } catch (Exception e) {
>144|         logAndBuildDetailErrorMessage(e);
>145|     }
146|}
147|
148|public void revoke(CompareCountsSelect selected) {

```

### Issue B73A1FAAE73A83EE5AB42BC050690936:

Sink :

File admin-frontend/src/main/java/no/valg/eva/admin/frontend/counting/ctrls/CompareCountsController.java 155-157

```

152|         updateCountList(count);
153|         MessageUtil.buildDetailMessage(FacesMessage.SEVERITY_INFO,
getMessageProvider().get("@count.isNotApprovedAnymore"));
154|         setupDefaultCompare();
>155|     } catch (Exception e) {
>156|         logAndBuildDetailErrorMessage(e);
>157|     }
158|}
159|
160|public Button getApproveButton(CompareCountsSelect selected) {

```

### Issue C0817A5B15463BB2F02F6A593CE06394:

Sink :

File admin-frontend/src/main/java/no/valg/eva/admin/frontend/counting/ctrls/CompareCountsController.java 127-129

```

124|         count = getCountingService().saveCount(userData, getCountContext(), count);
125|         updateCountList(count);
126|         MessageUtil.buildDetailMessage(FacesMessage.SEVERITY_INFO, getMessageProvider().get("@count.isSaved"));
>127|     } catch (Exception e) {
>128|         logAndBuildDetailErrorMessage(e);
>129|     }
130|}
131|
132|public void approve(CompareCountsSelect selected) {

```

## 6.255 ElectionDayPollingPlaceViewModel.java

### Issue 5AE799ECD21D7036DD1565DC9AD7E51F:

Sink :

File admin-frontend/src/main/java/no/valg/eva/admin/frontend/configuration/models/ElectionDayPollingPlaceViewModel.java 175-177

```

172|     try {
173|         validateOpeningHours();
174|         return true;
>175|     } catch (Exception e) {
>176|         return false;
>177|     }
178|}
179|
180|public void validateOpeningHours() {

```

## 6.256 ElectionEventResourceBundleProvider.java

### Issue B975BB6B5BB54E180E8F70CB60EFF51F:

Sink :

File admin-backend/src/main/java/no/valg/eva/admin/backend/reporting/jasperserver/ElectionEventResourceBundleProvider.java 75-77

```

72|     properties.load(reader);
73|} catch (IOException e) {
74|     LOG.error("IOException ved lasting av " + resourceName, e);
>75|} catch (Exception e) {
>76|     LOG.error("Feil ved lasting av " + resourceName, e);
>77|}
78|Map<String, String> globalTextAsMap = properties.keySet().stream().collect(Collectors.toMap(key -> (String) key,
key -> (String) properties.get(key)));
79|
80|List<Pair<String, byte[]>> resourceBundleForGlobalTexts = new ArrayList<>();

```

## 6.257 ElectionOptions.java

Issue 819DB3F2DF7491E71529B5DA1F41E0DD:

Sink :

File admin-frontend/src/main/java/no/valg/eva/admin/frontend/rbac/ctrls/ElectionOptions.java 127-129

```

124|         } else if (electionPath.getLevel() == CONTEST) {
125|             return findByPath(electionPath, contests);
126|         }
>127|     } catch (Exception e) {
>128|         return null;
>129|     }
130|     return null;

```

## 6.258 ElectionVoteCountCategoryController.java

Issue 41D33C3BB34718C741B9B673777A091B:

Sink :

File admin-frontend/src/main/java/no/valg/eva/admin/frontend/categories/ctrls/ElectionVoteCountCategoryController.java 183-186

```

180|         electionVoteCountCategoryService.update(userData, cats(elements));
181|         municipalityService.setRequiredProtocolCountForElectionEvent(userData,
selectedElectionGroup.getElectionEvent().getPk(), isRequiredProtocolCount());
182|         MessageUtil.buildDetailMessage(getFacesContext(), MessageUtil.UPDATE_SUCCESSFUL_KEY,
FacesMessage.SEVERITY_INFO);
>183|     } catch (RuntimeException e) {
>184|         String message = createErrorMessage(elements, e);
>185|         LOGGER.error(message, e);
>186|     }
187|     elements = electionVoteCountCategoryElements(userData, selectedElectionGroup);

```

## 6.259 EnumCache.java

Issue 549BBE30A8F1B42E929910DE1CDD7E59:

Sink :

File admin-frontend/src/main/java/no/evote/presentation/cache/EnumCache.java 40-42

```

37|         for (Enum<?> en : valueList) {
38|             propertCache.put(clazz.getSimpleName() + "." + en.name(), en); 39|         }
>40|     } catch (Exception e) {
>41|         log.error(clazz.getSimpleName(), e);

```

## 6.260 FullElectoralRollImporter.java

Issue 6C9DC5BD96DC3A62BE6608F49A868A5A:

Sink :

File admin-backend/src/main/java/no/evote/service/configuration/FullElectoralRollImporter.java 75-77

```

72|         success = true;
73|         recordCompletionOfImport(run);
74|         sendManntallsimportFullfortEvent(userData, electionEvent);
>75| } catch (Exception e) {
>76|     recordFailureOfImport(userData, filePath, e);
>77| } finally {
78|     finalizeImport(run, success, context);
79|     em.setFlushMode(OldFlushMode);

```

## 6.261 ImportBuypassSerialNumberController.java

Issue 24FF11BAC6271DD852A4698E663F6C63:

Sink :

File admin-frontend/src/main/java/no/valg/eva/admin/frontend/electoralroll/ctrls/ImportBuypassSerialNumberController.java 57-60

```

54|    try {
55|        operators = adminOperatorService.updateBuypassKeySerialNumbers(userData, importFile);
56|        MessageUtil.buildDetailMessage("@config.operator.import_buypass_number_success",
FacesMessage.SEVERITY_INFO);
>57|    } catch (Exception e) {
>58|        MessageUtil.buildDetailMessage("@config.operator.import_buypass_number_error",
FacesMessage.SEVERITY_FATAL);
>59|        LOGGER.error("Feil i import av buypassserienummer", e);
>60|    }
61|}
62|
63|public List<BuypassOperator> getOperators() {

```

## 6.262 IncrementalElectoralRollImporter.java

Issue 94D989EB00E4437E80883BAC5818C06F:

Sink :

File admin-backend/src/main/java/no/evote/service/configuration/IncrementalElectoralRollImporter.java 114-122

```

111|        finalizeImport(run, true, context);
112|        deleteDirectory(temporaryProcessingFolder); // The files viewed by admin are already copies, so the files
are actually archived outside admin. 113|        return true;
>114|    } catch (Exception e) {
>115|        LOGGER.error(format("Import of incremental electoral roll file %s failed with message %s in election
event %s",
>116|                            file.getName(), e.getMessage(), electionEvent.getId()), e);
>117|        lagreBakgrunnsjobbSomFeilet(userData, file.getName(), e.getMessage());
>118|        auditLogFailedFileImport(userData, file.getName(), e.getMessage());
>119|        finalizeImport(run, false, context);
>120|        deleteDirectory(temporaryProcessingFolder); // The files viewed by admin are already copies, so the files
are actually archived outside admin.
>121|        return false;
>122|    }
123|}
124|
125|private void validateFilename(File file) {

```

## 6.263 ListProposalConfigurationController.java

Issue 8F659A8C3138EA3F3B20B81D1B34242C:

Sink :

File admin-frontend/src/main/java/no/valg/eva/admin/frontend/configuration/ctrls/local/ListProposalConfigurationController.java 25-27

```

22|if (data == null) {
23|    try {
24|        data = loadData();
>25|    } catch (RuntimeException e) {
>26|        return false;
>27|    }
28|
29|}
30|return data != null && data.getChildren().isEmpty() && data.isSingleArea();

```

## 6.264 MapModelMaker.java

Issue 2596F8AB421F65059CD52B7A4E7BBA3E:

Sink :

File admin-frontend/src/main/java/no/valg/eva/admin/frontend/common/MapModelMaker.java 34-37

```

31|        double lat = parseDouble(splitLatLng[0]);
32|        double lng = parseDouble(splitLatLng[1]);
33|        return new LatLng(lat, lng);
>34|    } catch (Exception e) {
>35|        LOGGER.error("Exception parsing gps coordinates: " + e.getMessage(), e); >36|        return null;

```

## 6.265 OperatorEditController.java

**Issue** 76CA97DABF9DCB7D4056ADC668CC9BC8:

**Sink :**

**File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/rbac/ctrls/OperatorEditController.java 275-279

```

272|         MessageUtil.buildDetailMessage(getFacesContext(), name + " " +
messageProvider.get("@operation.deleted").toLowerCase(),
273|         FacesMessage.SEVERITY_INFO);
274|         resetState();
>275|     } catch (Exception e) {
>276|         LOGGER.warn("deleteOperator() failed", e);
>277|         MessageUtil.buildFacesMessage(getFacesContext(), null,
messageProvider.get("@common.message.delete.unsuccessful"), null,
>278|         FacesMessage.SEVERITY_ERROR);
>279|     }
280| }
281|
282| public Operator getOperator() {

```

## 6.266 ParentPollingDistrictConfigurationController.java

**Issue** 9B6755BFDDFB32D487893623D145EAA2:

**Sink :**

**File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/configuration/ctrls/local/ParentPollingDistrictConfigurationController.java 74-76

```

71|     try {
72|         ReportCountCategoriesConfigurationController countCtrl =
getController(ReportCountCategoriesConfigurationController.class);
73|         return countCtrl != null && !countCtrl.isValgtingsstemmerSentraltSamlet(); >74|     } catch
(RuntimeException e) {
>75|         return true;
>76|     }
77| }
78|
79| @Override

```

## 6.267 PollingDistrict.java

**Issue** 5C95C3767FE1ACB276A8CE731CF240DC:

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/common/configuration/model/local/PollingDistrict.java 49-51

```

46| public boolean isValid() {
47|     try {
48|         validateType();
>49|     } catch (RuntimeException e) {
>50|         return false;
>51|     }
52|     return super.isValid();

```

## 6.268 PreliminaryCountController.java

**Issue** 47115BDB41E5E93F7DE55CD87BA909E3:

**Sink :**

**File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/counting/ctrls/PreliminaryCountController.java 77-79

```

74|     try {
75|         setCount(countingService.revokeCount(userData, getCountContext(), getPreliminaryCount()));
76|         MessageUtil.buildDetailMessage(FacesMessage.SEVERITY_INFO,
getMessageProvider().get("@count.isNotApprovedAnymore"));
>77|     } catch (Exception e) {
>78|         logAndBuildDetailErrorMessage(e);
>79|     }
80| }
81|
82| @Override

```

**Issue** AFE69816757729A70393DD84828F02CF:

**Sink :**

File admin-frontend/src/main/java/no/valg/eva/admin/frontend/counting/ctrls/PreliminaryCountController.java 55-57

```

52|     preliminaryCount.validate();
53|     setCount(countingService.saveCount(userData, getCountContext(), preliminaryCount));
54|     MessageUtil.buildDetailMessage(FacesMessage.SEVERITY_INFO, getMessageProvider().get("@count.isSaved"));
>55| } catch (Exception e) {
>56|     logAndBuildDetailErrorMessage(e);
>57| }
58|}
59|
60|public void approveCount() {

```

**Issue D96E1FD4893736FAE21D06891B751A88:****Sink :**

File admin-frontend/src/main/java/no/valg/eva/admin/frontend/counting/ctrls/PreliminaryCountController.java 66-68

```

63|     getValidator().validate(preliminaryCount);
64|     setCount(countingService.approveCount(userData, getCountContext(), preliminaryCount));
65|     MessageUtil.buildDetailMessage(FacesMessage.SEVERITY_INFO, getMessageProvider().get("@count.isApproved"));
>66| } catch (Exception e) {
>67|     logAndBuildDetailErrorMessage(e);
>68| }
69|     getApproveDialog().closeAndUpdate("countingForm");

```

**6.269 ProposerAuditEvent.java****Issue 048649B7B54341057894DD8B914E9832:****Sink :**

File admin-common/src/main/java/no/valg/eva/admin/common/auditlog/auditevents/ProposerAuditEvent.java 103-105

```

102|     return null;
>103| } catch (Exception e) {
>104|     return null; // in case of uninitialized proxies >105| }
106|}
107|
108|private String getProposerRole() {

```

**Issue B233DE2C357EEE679A92E6CA28F1D323:****Sink :**

File admin-common/src/main/java/no/valg/eva/admin/common/auditlog/auditevents/ProposerAuditEvent.java 88-90

```

87|     return null;
>88| } catch (Exception e) {
>89|     return null; // in case of uninitialized proxies
>90| }
91|}
92|
93|private String getPartyId() {

```

**6.270 ProtocolAndPreliminaryCountController.java****Issue 46A012DEF2D8B3C7983873EDD659B20E:****Sink :**

File admin-frontend/src/main/java/no/valg/eva/admin/frontend/counting/ctrls/ProtocolAndPreliminaryCountController.java 50-52

```

47|     getCounts().setFirstProtocolCount(getProtocolAndPreliminaryCount().getProtocolCount());
48|     getCounts().setPreliminaryCount(getProtocolAndPreliminaryCount().getPreliminaryCount());
49|     MessageUtil.buildDetailMessage(FacesMessage.SEVERITY_INFO, getMessageProvider().get("@count.isApproved"));
>50| } catch (Exception e) {
>51|     logAndBuildDetailErrorMessage(e);
>52| }
53|     getApproveDialog().closeAndUpdate("countingForm");

```

**Issue 8FoF2DE71309499F6A6A9206E5994AD5:****Sink :**

File admin-frontend/src/main/java/no/valg/eva/admin/frontend/counting/ctrls/ProtocolAndPreliminaryCountController.java 38-40

```

35|     getCounts().setFirstProtocolCount(getProtocolAndPreliminaryCount().getProtocolCount());
36|     getCounts().setPreliminaryCount(getProtocolAndPreliminaryCount().getPreliminaryCount());
37|     MessageUtil.buildDetailMessage(FacesMessage.SEVERITY_INFO, getMessageProvider().get("@count.isSaved"));

```

```

>38|     } catch (Exception e) {
>39|         logAndBuildDetailErrorMessage(e);
>40|     }
41|}
42|
43|public void approveCount() {

```

**Issue BC68B2BED07BDFE62FCB1D9598B18CF2:****Sink :****File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/counting/ctrls/ProtocolAndPreliminaryCountController.java 63-65

```

60|         getCounts().setFirstProtocolCount(getProtocolAndPreliminaryCount().getProtocolCount());
61|         getCounts().setPreliminaryCount(getProtocolAndPreliminaryCount().getPreliminaryCount());
62|         MessageUtil.buildDetailMessage(FacesMessage.SEVERITY_INFO,
getMessageProvider().get("@count.isNotApprovedAnymore"));
>63|     } catch (Exception e) {
>64|         logAndBuildDetailErrorMessage(e);
>65|     }
66|}
67|
68|@Override

```

**6.271 ProtocolCountController.java****Issue C5C22CD3E3CD71640EB32BEB37EA581C:****Sink :****File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/counting/ctrls/ProtocolCountController.java 61-63

```

58|         protocolCount.validateForApproval();
59|         setCount(countingService.approveCount(userData, getCountContext(), protocolCount));
60|         MessageUtil.buildDetailMessage(FacesMessage.SEVERITY_INFO, getMessageProvider().get("@count.isApproved"));
>61|     } catch (Exception e) {
>62|         logAndBuildDetailErrorMessage(e);
>63|     }
64|     getApproveDialog().closeAndUpdate("countingForm");

```

**Issue DA1EC54965B36B2B25F8B28708435ADE:****Sink :****File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/counting/ctrls/ProtocolCountController.java 72-74

```

69|     try {
70|         setCount(countingService.revokeCount(userData, getCountContext(), getProtocolCount()));
71|         MessageUtil.buildDetailMessage(FacesMessage.SEVERITY_INFO,
getMessageProvider().get("@count.isNotApprovedAnymore"));
>72|     } catch (Exception e) {
>73|         logAndBuildDetailErrorMessage(e);
>74|     }
75|}
76|
77|@Override

```

**Issue E4BB516775C103362E2D400AF5BC7BAA:****Sink :****File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/counting/ctrls/ProtocolCountController.java 50-52

```

47|         protocolCount.validate();
48|         setCount(countingService.saveCount(userData, getCountContext(), protocolCount));
49|         MessageUtil.buildDetailMessage(FacesMessage.SEVERITY_INFO, getMessageProvider().get("@count.isSaved"));
>50|     } catch (Exception e) {
>51|         logAndBuildDetailErrorMessage(e);
>52|     }
53|}
54|
55|public void approveCount() {

```

**6.272 ReportLinksController.java****Issue 07A839138F8E852B90EB19E016D2E276:****Sink :****File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/reports/ctrls/ReportLinksController.java 111-114

```

108|         } else {

```

```

109|         return null;
110|     }
>111| } catch (RuntimeException e) {
>112|     process(e);
>113|     return null;
>114| }
115|}
116|
117|public void pollReportContent(String format) {

```

## 6.273 ReportTemplateUploadStatusServlet.java

### Issue E0894935197D0920CDDF93B205B53ACA:

#### Sink :

File admin-backend/src/main/java/no/valg/eva/admin/backend/reporting/jasperserver/ReportTemplateUploadStatusServlet.java 57-59

```

54|     try {
55|         jasperRestApiWithTimeout.getResources(reportUnit, "/reports/EVA");
56|         return true;
>57|     } catch (Exception e) {
>58|         return false;
>59|     }
60|}
61|
62|@Override

```

## 6.274 ReportTemplateUploader.java

### Issue 3ABF08E8FFD89B5DE9CF726AECCF6224:

#### Sink :

File admin-backend/src/main/java/no/valg/eva/admin/backend/reporting/jasperserver/ReportTemplateUploader.java 217-220

```

215|     } catch (NotFoundException e) {
216|         shouldUploadReportTemplates = true;
>217|     } catch (Exception e) {
>218|         LOG.error("Failed to obtain version info from JasperServer (" + baseUrl + context + ")");
>219|         shouldUploadReportTemplates = true;
>220|     }
221|     return shouldUploadReportTemplates;

```

### Issue 889891A7C752B91FF03F007721BE495F:

#### Sink :

File admin-backend/src/main/java/no/valg/eva/admin/backend/reporting/jasperserver/ReportTemplateUploader.java 236-238

```

233|         uploadRequested.set(false);
234|         runUploader(refreshResourceBundles, lastDatabaseLocaleTextTimeStamp); 235|     }
>236|     } catch (Exception e) {
>237|         LOG.error(e.getMessage(), e);
>238|     } finally {
239|         busy.set(false);
240|     }
241| } else {

```

### Issue BA00D6FB08177404D637AFCBCD623689:

#### Sink :

File admin-backend/src/main/java/no/valg/eva/admin/backend/reporting/jasperserver/ReportTemplateUploader.java 280-282

```

277|         .uploadAllTemplates(configuration, resourceBundles, refreshResourceBundles,
lastDatabaseLocaleTextTimeStamp, allMunicipalitiesParameters,
278|             allCountiesParameters);
279|         getUploadReportTemplatesDoneEvent().fire(new UploadReportTemplatesDoneEvent()); >280|     } catch
(Exception e) {
>281|         LOG.error(e.getMessage(), e);
>282|     }
283| } 284|
285|private List<Map<String, String>> parametersForAllMvAreas(List<MvArea> mvAreas, AreaLevelEnum finalLevel) {

```

## 6.275 RevocationListScheduler.java

**Issue** 16CDEC070CB6331D0512DF09AB405129:

**Sink :**

**File** admin-backend/src/main/java/no/valg/eva/admin/backend/application/schedule/RevocationListScheduler.java 36-38

```

33|     LOGGER.info("updateCertificateRevocationLists start");
34|     try {
35|         revocationListUpdater.updateIfNeeded();
>36|     } catch (Exception e) {
>37|         LOGGER.error(e.getMessage(), e);
>38|     }
39|     LOGGER.info("updateCertificateRevocationLists end");

```

## 6.276 ScheduledImportTimer.java

**Issue** C22E95939C9A9903EC7ABC054869CC6F:

**Sink :**

**File** admin-backend/src/main/java/no/valg/eva/admin/backend/application/schedule/ScheduledImportTimer.java 98-102

```

95|         LOGGER.info("no intitial import has been performed for '" + electionEvent.getName() +
APOSTROPHE);
96|         LOGGER.info("no scheduled import will be performed for '" + electionEvent.getName() +
APOSTROPHE);
97|     }
> 98|     } catch (RuntimeException | UnknownHostException e) {
> 99|         // In no way should any exception be propagated to the scheduled task, as it will cause the scheduled
task to be removed from the
>100|         // application server
>101|         LOGGER.error(e.getMessage(), e);
>102|     }
103| }
104| LOGGER.info("scheduledImportOfElectionEvents end");
105|} catch (RuntimeException e) {

```

**Issue** C22E95939C9A9903EC7ABC054869CC70:

**Sink :**

**File** admin-backend/src/main/java/no/valg/eva/admin/backend/application/schedule/ScheduledImportTimer.java 105-108

```

104|     LOGGER.info("scheduledImportOfElectionEvents end");
>105| } catch (RuntimeException e) {
>106|     // In no way should any exception be propagated to the scheduled task, as it will cause the scheduled
task to be removed from the application server >107|     LOGGER.error(e.getMessage(), e);

```

## 6.277 SessionListener.java

**Issue** DB6DC33A23EFAAD7AF861EFECA889970:

**Sink :**

**File** admin-frontend/src/main/java/no/evote/service/security/SessionListener.java 62-64

```

59|private UserData getUserDataIfPossible() {
60|    try {
61|        return userDataInstance.get();
>62|    } catch (RuntimeException e) {
>63|        LOGGER.trace("Unable to get UserData object while destroying context");
>64|    }
65|    return null;

```

## 6.278 SigningKeyController.java

**Issue** 1569A5377178C019CB79128A79D8C383:

**Sink :**

**File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/configuration/ctrls/SigningKeyController.java 108-111

```

105|    try (InputStream ins = file.getInputStream()) {
106|        Pkcs12Decoder pkcs12Decoder = getPkcs12Decoder();
107|        return pkcs12Decoder.validatorPassword(ins, password);
>108|    } catch (Exception e) {
>109|        LOGGER.error(e.getMessage(), e);
>110|        return false;
>111|    }

```



```
112|}
113|
114|Pkcs12Decoder getPkcs12Decoder() {
```

**Issue A29D165B6D29F24786B80D2069E8EF59:****Sink :****File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/configuration/ctrls/SigningKeyController.java 91-94

```
88|         byte[] bytes = ByteStreams.toByteArray(file.getInputStream());
89|         signingKey = signingKeyService.create(userData, signingKey, bytes, file.getFileName(), password,
currentElectionEvent);
90|         init();
>91|     } catch (Exception e) {
>92|         LOGGER.error(e.getMessage(), e);
>93|         MessageUtil.buildDetailMessage("@config.certificate_management.error.error", FacesMessage.SEVERITY_ERROR);
>94|     }
95|}
96|
97|private boolean validate() {
```

**6.279 StartCountingController.java****Issue 0A066AD8369CED14DCF2267247104B20:****Sink :****File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/counting/ctrls/StartCountingController.java 84-86

```
81|         if (isStartCountingXhtmlPath()) {
82|             initTabsAndRedirectToCountingXhtml();
83|         }
>84|     } catch (Exception e) {
>85|         logAndBuildDetailErrorMessage(e);
>86|     }
87|}
88|
89|public String getPageTitle() {
```

**6.280 StatusApplicationService.java****Issue E1E307B2FC6D4FD24F4EC04ADF88B173:****Sink :****File** admin-backend/src/main/java/no/valg/eva/admin/backend/application/service/StatusApplicationService.java 60-63

```
58|         throw new IllegalStateException("Empty result set - should never happen!");
59|     }
>60| } catch (RuntimeException | SQLException e) {
>61|     LOG.info("Backend is not connected to database", e);
>62|     return false;
>63| }
64|}
65|
66|public Properties getStatusAndConfiguredVersionProperties() {
```

**6.281 UserAgent.java****Issue D98380339057380BB8CBF1FA1FF8CoDA:****Sink :****File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/common/UserAgent.java 16-18

```
13|     this.familyName = readableUserAgent.getFamily().getName();
14|     try {
15|         this.versionMajor = Integer.parseInt(readableUserAgent.getVersionNumber().getMajor());
>16|     } catch (Exception e) {
>17|         ;
>18|     }
19|}
20|
21|public boolean isMSIE() {
```

**6.282 UserDataServiceEjb.java****Issue 01653B679D2E070F0D85C236A11583E3:**

**Sink :**

File admin-rbac/src/main/java/no/valg/eva/admin/rbac/service/UserDataServiceEjb.java 129-131

```

126|if (!operatorRepository.hasOperator(userData.getId())) {
127|    try {
128|        auditLogService.addToAuditTrail(new OperatorLoginAuditEvent(userData, Outcome.UnknownOperator));
>129|    } catch (Exception e) {
>130|        LOGGER.warn("addToAuditTrail() feilet: " + e.getMessage(), e);
>131|    }
132|    LOGGER.warn(userData.getClientAddress() + SPACE + userData.getId() + SPACE + new
Exception().getStackTrace()[0].getMethodName() + SPACE
133|        + this.getClass().getName() + SPACE +
USER_PASSED_ID_PORTEN_LOGIN_BUT_DID_NOT_HAVE_AN_OPERATOR_IN_THE_SYSTEM);
134|    throw new EvoteSecurityException(USER_PASSED_ID_PORTEN_LOGIN_BUT_DID_NOT_HAVE_AN_OPERATOR_IN_THE_SYSTEM + ":
" + userData.getId());

```

**6.283 ValgkortgrunnlagDomainService.java****Issue oD068C9EA77F0A331FE8DC3FC0C8653A:****Sink :**

File admin-configuration/src/main/java/no/valg/eva/admin/configuration/domain/service/ValgkortgrunnlagDomainService.java 122-126

```

119|        oppdaterBakgrunnsjobb(userData, bakgrunnsJobb, BATCH_STATUS_COMPLETED_ID);
120|        leggTilAuditlogg(userData, fil, GenererValgkortgrunnlagJobbFerdig, Success, null);
121|        valgkortgrunnlagStatistikk.toggStatistikkForGenerering();
>122|    } catch (Exception e) {
>123|        oppdaterBakgrunnsjobb(userData, bakgrunnsJobb, BATCH_STATUS_FAILED_ID);
>124|        leggTilAuditlogg(userData, fil, GenererValgkortgrunnlagJobbFerdig, GenericError, e.getMessage());
>125|        LOG.error("Generering av valgkortgrunnlag feilet", e);
>126|    }
127|    return fil;

```

**6.284 VoterServiceBean.java****Issue 98815CD141A37D407046AA8AA68AD8C3:****Sink :**

File admin-backend/src/main/java/no/evote/service/configuration/VoterServiceBean.java 86-89

```

84|        bakgrunnsjobbDomainService.oppdaterBakgrunnsjobb(userData, batch, BATCH_STATUS_COMPLETED_ID);
85|        LOGGER.info("delete voters - end");
>86|    } catch (Exception e) {
>87|        bakgrunnsjobbDomainService.oppdaterBakgrunnsjobb(userData, batch, BATCH_STATUS_FAILED_ID);
>88|        LOGGER.error("delete voters - error: ", e);
>89|    }
90|}
91|
92|public void prepareNewInitialLoad(UserData userData, MvElection mvElection, MvArea mvArea) {

```

**[Low] Poor Error Handling / Overly Broad Throws***Errors***Analysis:** Bad Practice**Abstract**

The method throws a generic exception making it harder for callers to do a good job of error handling and recovery.

**Explanation**

Declaring a method to throw `Exception` or `Throwable` makes it difficult for callers to do good error handling and error recovery. Java's exception mechanism is set up to make it easy for callers to anticipate what can go wrong and write code to handle each specific exceptional circumstance. Declaring that a method throws a generic form of exception defeats this system.

**Example:** The following method throws three types of exceptions.

```

public void doExchange()
    throws IOException, InvocationTargetException, SQLException { ...
}

```

While it might seem tidier to write

```
public void doExchange()
throws Exception { ...
}
```

doing so hampers the caller's ability to understand and handle the exceptions that occur. Further, if a later revision of `doExchange()` introduces a new type of exception that should be treated differently than previous exceptions, there is no easy way to enforce this requirement.

## Recommendation

Do not declare methods to throw `Exception` or `Throwable`. If the exceptions thrown by a method are not recoverable or should not generally be caught by the caller, consider throwing unchecked exceptions rather than checked exceptions. This can be accomplished by implementing exception classes that extend `RuntimeException` or `Error` instead of `Exception`, or add a try/catch wrapper in your method to convert checked exceptions to unchecked exceptions.

## 6.285 AuditLogInterceptor.java

Issue CC960231C9088BDD57670C8F95F6A05D:

Sink :

File admin-backend/src/main/java/no/valg/eva/admin/backend/auditlog/AuditLogInterceptor.java 35-70 **Fact**  
Name: no.valg.eva.admin.backend.auditlog.AuditLogInterceptor.intercept

```
34|@AroundInvoke
>35|public Object intercept(InvocationContext context) throws Exception {
>36|    AuditEventFactory auditEventFactory = null;
>37|    try {
>38|        auditEventFactory = createAuditEventFactory(context);
>39|    } catch (Exception e) {
>40|        logger.error("Unable to audit log method invocation - failed to create audit event.", e);
>41|        return context.proceed();
>42|    }
>43|
>44|    if (!auditEventFactory.isAuditedInvocation()) {
>45|        return context.proceed();
>46|    }
>47|
>48|    auditEventFactory.initialize();
>49|
>50|    try {
>51|        Object returnValue = context.proceed();
>52|        try {
>53|            AbstractAuditEvent auditEvent = auditEventFactory.buildSuccessfulAuditEvent(returnValue);
>54|            if (!auditEvent.muteEvent()) {
>55|                auditLogService.addToAuditTrail(auditEvent);
>56|            }
>57|        } catch (Exception exceptionWhenAuditLogging) {
>58|            logger.error("Unable to audit log successful method invocation", exceptionWhenAuditLogging);
>59|        }
>60|        return returnValue;
>61|    } catch (Exception e) {
>62|        try {
>63|            AbstractAuditEvent auditEvent = auditEventFactory.buildErrorAuditEvent(e.getMessage());
>64|            auditLogService.addToAuditTrail(auditEvent);
>65|        } catch (Exception exceptionWhenAuditLogging) {
>66|            logger.error("Unable to audit log failed method invocation", exceptionWhenAuditLogging);
>67|        }
>68|        throw e;
>69|    }
>70|}
71|
72|/**
73| * Creates an {@link AuditEventFactory}. May be overridden in tests, to provide a non-default implementation.
```

## 6.286 BootstrapFunctionalTestsImporter.java

Issue EC1454AF3BCB41FC37FAFE9058FDE2C8:

**Sink :****File** admin-database/src/main/java/no/valg/eva/admin/database/tools/BootstrapFunctionalTestsImporter.java 58-63**Fact Name:** no.valg.eva.admin.database.tools.BootstrapFunctionalTestsImporter.importIntoDatabase

```

55|     new BootstrapFunctionalTestsImporter(dbHost, dbPort, dbName).importIntoDatabase();
56| }
57|
>58| public void importIntoDatabase() throws Exception {
>59|     new ImportTablesWrapper(dbHost, dbPort, dbName)
>60|         .with("operator", data("operator.txt"))
>61|         .with("operator_role", data("operator_role.txt"))
>62|         .execute();
>63| }
64|
65| private File data(String filename) {
66|     return new File(dataDir, filename);

```

## 6.287 CertificateRevocationListService.java

### Issue 590EC46848E7C0BF96F08AD129478B5E:

**Sink :****File** admin-backend-common/src/main/java/no/valg/eva/admin/backend/common/port/adapter/service/buypass/CertificateRevocationListService.java 79-88**Fact Name:** no.valg.eva.admin.backend.common.port.adapter.service.buypass.CertificateRevocationListService.readCrlFromBuypass

```

>79| private X509CRL readCrlFromBuypass(URL url, PublicKey caPublicKey) throws Exception {
>80|     try (DataInputStream inStream = new DataInputStream(getURLConnection(url).getInputStream())) {
>81|
>82|         CertificateFactory cf = CertificateFactory.getInstance("X509");
>83|         X509CRL x509CRL = (X509CRL) cf.generateCRL(inStream);
>84|         x509CRL.verify(caPublicKey);
>85|
>86|         return x509CRL;
>87|     }
>88| }
89|
90| URLConnection getURLConnection(URL url) throws IOException {
91|     URLConnection urlConnection = url.openConnection();

```

## 6.288 ErrorInterceptor.java

### Issue 4BFBD2AB3F934425A97749D3AEEF8A79:

**Sink :****File** admin-backend/src/main/java/no/evote/service/util/ErrorInterceptor.java 26-52**Fact Name:** no.evote.service.util.ErrorInterceptor.cleanException

```

23|private static final Logger LOGGER = Logger.getLogger(ErrorInterceptor.class);
24|
25|@AroundInvoke
>26|public Object cleanException(InvocationContext ctx) throws Exception {
>27|    try {
>28|        return ctx.proceed();
>29|    } catch (PersistenceException e) {
>30|        Exception unwrapped = ExceptionUtil.unwrapThrowable(e, OptimisticLockException.class);
>31|        if (unwrapped != null) {
>32|            throw evoteException(ErrorCode.ERROR_CODE_0501_OPTIMISTIC_LOCK, unwrapped);
>33|        }
>34|        unwrapped = ExceptionUtil.unwrapThrowable(e, EntityNotFoundException.class);
>35|        if (unwrapped != null) {
>36|            throw evoteException(ErrorCode.ERROR_CODE_0502_ENTITY_NOT_FOUND, unwrapped);
>37|        }
>38|        unwrapped = ExceptionUtil.unwrapThrowable(e, ConstraintViolationException.class);
>39|        if (unwrapped != null) {
>40|            throw evoteException(ErrorCode.ERROR_CODE_0503_CONSTRAINT_VIOLATION, unwrapped);
>41|        }
>42|        unwrapped = ExceptionUtil.unwrapThrowable(e, StaleObjectStateException.class);
>43|        if (unwrapped != null) {
>44|            throw evoteException(ErrorCode.ERROR_CODE_0504_STALE_OBJECT, unwrapped);
>45|        }
>46|        LOGGER.error("Unexpected database error " + e.getMessage(), e);
>47|        throw evoteException(ErrorCode.ERROR_CODE_0500_UNEXPECTED, e);
>48|    } catch (Exception e) {
>49|        preventInvalidCause(e);
>50|        throw e;
>51|    }
>52|}
53|
54|private EvoteException evoteException(ErrorCode errorCode, Exception e) {
55|    EvoteException result = new EvoteException(errorCode, null, null);

```

## 6.289 ExportImportRolesController.java

Issue 4CF5BC13BB88A9D289D30FCD5C2B1400:

Sink :

File admin-frontend/src/main/java/no/valg/eva/admin/frontend/rbac/ctrls/ExportImportRolesController.java 58-64

Fact Name: no.valg.eva.admin.frontend.rbac.ctrls.ExportImportRolesController.fileUpload

```

55|    });
56|}
57|
>58|public void fileUpload(FileUploadEvent fileUploadEvent) throws Exception {
>59|    String fileContent = new String(IOUTil.getBytes(fileUploadEvent.getFile().getInputStream()), "UTF-8");
>60|    execute() -> {
>61|        int numberOfRolesImported = roleService.importRoles(userData, fileContent, isDeleteExistingRoles());
>62|        MessageUtil.buildDetailMessage("@rbac.roles.imported", new String[] {
String.valueOf(numberOfRolesImported) }, SEVERITY_INFO);
>63|    });
>64|}
65|
66|public boolean isDeleteExistingRoles() {
67|    return deleteExistingRoles;

```

## 6.290 ImportAreaHierarchyChangesController.java

Issue 497128A0B4BF856CF4114ABDB33A53B1:

Sink :

File admin-frontend/src/main/java/no/valg/eva/admin/frontend/electoralroll/ctrls/ImportAreaHierarchyChangesController.java 42-45

Fact Name: no.valg.eva.admin.frontend.electoralroll.ctrls.ImportAreaHierarchyChangesController.fileUpload

```

39|    this.areaImportChangesService = areaImportChangesService;
40|}
41|
>42|public void fileUpload(FileUploadEvent fileUploadEvent) throws Exception {
>43|    importFile = IOUTil.getBytes(fileUploadEvent.getFile().getInputStream());
>44|    FacesUtil.executeJS("PF('confirmationWidget').show()");
>45|}
46|
47|public void importUploadedFile() {
48|    response = null;

```

## 6.291 ImportAreaHierarchyController.java

Issue 6D8C17DC7D4959BFA31EC21DBBD14494:

Sink :

**File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/electionroll/ctrls/ImportAreaHierarchyController.java 39-42  
**Fact Name:** no.valg.eva.admin.frontend.electionroll.ctrls.ImportAreaHierarchyController.fileUpload

```

36|   this.areaImportService = areaImportService;
37| }
38|
>39| public void fileUpload(FileUploadEvent fileUploadEvent) throws Exception {
>40|     importFile = IOUtil.getBytes(fileUploadEvent.getFile().getInputStream());
>41|     FacesUtil.executeJS("PF('confirmationWidget').show()");
>42| }
43|
44| public void importUploadedFile() throws IOException {
45|     if (importFile == null) {

```

## 6.292 ImportBuypassSerialNumberController.java

**Issue** 7B25343DD15F2086EE91FC865647192B:

**Sink :**

**File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/electionroll/ctrls/ImportBuypassSerialNumberController.java 44-47

**Fact Name:** no.valg.eva.admin.frontend.electionroll.ctrls.ImportBuypassSerialNumberController.fileUpload

```

41|   this.adminOperatorService = adminOperatorService;
42| }
43|
>44| public void fileUpload(FileUploadEvent fileUploadEvent) throws Exception {
>45|     importFile = IOUtil.getBytes(fileUploadEvent.getFile().getInputStream());
>46|     FacesUtil.executeJS("PF('confirmationWidget').show()");
>47| }
48|
49| public void importUploadedFile() throws IOException {
50|     if (importFile == null) {

```

## 6.293 ImportOperatorsController.java

**Issue** 54C366DC74F6E1817BF60EC5A6DCoFD6:

**Sink :**

**File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/rbac/ctrls/ImportOperatorsController.java 42-48

**Fact Name:** no.valg.eva.admin.frontend.rbac.ctrls.ImportOperatorsController.fileUpload

```

39|   this.operatorRoleService = operatorRoleService;
40| }
41|
>42| public void fileUpload(FileUploadEvent fileUploadEvent) throws Exception {
>43|     byte[] importFile = IOUtil.getBytes(fileUploadEvent.getFile().getInputStream());
>44|     execute() -> {
>45|         List<ImportOperatorMessage> userMessages =
operatorRoleService.importOperatorRoles(userData, userData.getElectionEventPk(), importFile); >46|
handleUserMessages(userMessages);
>47|     });
>48| }
49|
50| private void handleUserMessages(List<ImportOperatorMessage> userMessages) {
51|     if (userMessages.isEmpty()) {

```

## 6.294 ImportTablesWrapper.java

**Issue** A1D9927037678CoAD2ED71B7D7F99CB6:

**Sink :**

**File** admin-database/src/main/java/no/valg/eva/admin/database/tools/ImportTablesWrapper.java 80-96

**Fact Name:** no.valg.eva.admin.database.tools.ImportTablesWrapper.execute

```

77|     return this;
78| }
79|
>80| public void execute() throws Exception {
>81|     ProcessBuilder pb = new ProcessBuilder("perl", "import-tables.pl", optionsBuilder.toString());
>82|     configureProcessBuilder(pb);
>83|
>84|     try {
>85|         Process process = pb.start();
>86|         printOutputToConsole(process);
>87|
>88|         int exitValue = process.waitFor();
>89|         if (exitValue != 0) {
>90|             throw new Exception("Command failed with status: " + exitValue);
>91|         }
>92|     } catch (Exception e) {
>93|         e.printStackTrace();
>94|         throw e;
>95|     }
>96| }
97|
98| private void configureProcessBuilder(ProcessBuilder pb) {
99|     pb.directory(new File(DirectoryLocator.adminDbDir(), "bin"));

```

## 6.295 JamonInterceptor.java

**Issue** 790E4B4D78E910382DE52EA52585AB37:

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/common/interceptor/JamonInterceptor.java 18-26

**Fact Name:** no.valg.eva.admin.common.interceptor.JamonInterceptor.invokeUnderTrace

```

15| public class JamonInterceptor implements Serializable {
16|
17|     @AroundInvoke
>18|     protected Object invokeUnderTrace(final InvocationContext ctx) throws Exception {
>19|         String name = createInvocationTraceName(ctx.getMethod());
>20|         Monitor monitor = MonitorFactory.start(name);
>21|         try {
>22|             return ctx.proceed();
>23|         } finally {
>24|             monitor.stop();
>25|         }
>26|     }
27|
28|     private String createInvocationTraceName(final Method method) {
29|         StringBuilder sb = new StringBuilder();

```

## 6.296 RoleAccessMigration.java

**Issue** 2725F9678B723261A2C2890CEE807E04:

**Sink :**

**File** admin-database/src/main/java/no/valg/eva/admin/database/migrations/RoleAccessMigration.java 71-80

**Fact Name:** no.valg.eva.admin.database.migrations.RoleAccessMigration.insert

```

>71| private void insert(Connection conn, Election election, RoleAccess role, String path, long rolePk, long accessPk)
throws Exception {
>72|     try (Statement stmt = conn.createStatement()) {
>73|         stmt.execute("INSERT INTO admin.role_access(role_pk, access_pk) VALUES(" + rolePk + ", " + accessPk +
")");
>74|     } catch (Exception e) {
>75|         // Ignore duplicate key (already exists is OK)
>76|         if (!e.getMessage().contains("nk_role_access")) {
>77|             throw e;
>78|         }
>79|     }
>80| }
81|
82| private List<RoleAccess> getRoleAccesses() throws Exception {
83|     ReferenceDataMigrationContext ctx = new ReferenceDataMigrationContext();

```

**Issue** 4737A6C904EDF06A2512B7D386EDE630:

**Sink :**

**File** admin-database/src/main/java/no/valg/eva/admin/database/migrations/RoleAccessMigration.java 107-119

**Fact Name:** no.valg.eva.admin.database.migrations.RoleAccessMigration.getElections

```

104|     return result;
105|}
106|
>107|private List<Election> getElections(Connection conn) throws Exception {
>108|     List<Election> result = new ArrayList<>();
>109|     try (Statement stmt = conn.createStatement();
>110|         ResultSet rs = stmt.executeQuery(
>111|             "SELECT election_event_pk, election_event_name FROM admin.election_event ORDER
BY election_event_id ASC")) { >112|         while (rs.next()) {
>113|             Election election = new Election(rs.getLong(1), rs.getString(2));
>114|             addRolesForElection(conn, election);
>115|             result.add(election);
>116|         }
>117|     }
>118|     return result;
>119|}
120|
121|private void addRolesForElection(Connection conn, Election election) throws Exception {
122|     ResultSet rs = null;

```

**Issue 4EC143424BA0B1DFCFF1CDCB486B3A6B:****Sink :****File** admin-database/src/main/java/no/valg/eva/admin/database/migrations/RoleAccessMigration.java 138-148**Fact** Name: no.valg.eva.admin.database.migrations.RoleAccessMigration.getAccesses

```

135|     election.setRoleMap(roleMap);
136|}
137|
>138|private Map<String, Long> getAccesses(Connection conn) throws Exception {
>139|     Map<String, Long> result = new HashMap<>();
>140|     try (Statement stmt = conn.createStatement();
>141|         ResultSet rs = stmt.executeQuery(
>142|             "SELECT access_pk, access_path FROM admin.access")) {
>143|         while (rs.next()) {
>144|             result.put(rs.getString(2), rs.getLong(1));
>145|         }
>146|     }
>147|     return result;
>148|}
149|
150|private static class Election {
151|     private long pk;

```

**Issue AFF90BE8F6DFF6B76CA4841FB0A213EF:****Sink :****File** admin-database/src/main/java/no/valg/eva/admin/database/migrations/RoleAccessMigration.java 65-69**Fact** Name: no.valg.eva.admin.database.migrations.RoleAccessMigration.deleteAllRoleAccesses

```

>65|private void deleteAllRoleAccesses(Connection conn) throws Exception {
>66|     try (Statement stmt = conn.createStatement()) {
>67|         stmt.executeUpdate("DELETE FROM admin.role_access");
>68|     }
>69|}
70|
71|private void insert(Connection conn, Election election, RoleAccess role, String path, long rolePk, long accessPk)
throws Exception {
72|     try (Statement stmt = conn.createStatement()) {

```

**Issue DEEAF74947B3ED73121773EDCDDF1380:****Sink :****File** admin-database/src/main/java/no/valg/eva/admin/database/migrations/RoleAccessMigration.java 121-136**Fact** Name: no.valg.eva.admin.database.migrations.RoleAccessMigration.addRolesForElection

```

118|     return result;
119|}
120|
>121|private void addRolesForElection(Connection conn, Election election) throws Exception {
>122|     ResultSet rs = null;
>123|     Map<String, Long> roleMap = new HashMap<>();

```



```

>124|    try (PreparedStatement stmt = conn.prepareStatement("SELECT role_pk, role_id FROM admin.role WHERE
election_event_pk = ?")) {
>125|        stmt.setLong(1, election.getPk());
>126|        rs = stmt.executeQuery();
>127|        while (rs.next()) {
>128|            roleMap.put(rs.getString(2), rs.getLong(1));
>129|        }
>130|    } finally {
>131|        if (rs != null) {
>132|            rs.close();
>133|        }
>134|    }
>135|    election.setRoleMap(roleMap);
>136|}
137|
138|private Map<String, Long> getAccesses(Connection conn) throws Exception {
139|    Map<String, Long> result = new HashMap<>();

```

**Issue F741658449E885240AD877F16831F98A:****Sink :****File** admin-database/src/main/java/no/valg/eva/admin/database/migrations/RoleAccessMigration.java 82-105**Fact Name:** no.valg.eva.admin.database.migrations.RoleAccessMigration.getRoleAccesses

```

> 82|private List<RoleAccess> getRoleAccesses() throws Exception {
> 83|    ReferenceDataMigrationContext ctx = new ReferenceDataMigrationContext();
> 84|    registerMigrationElements(ctx);
> 85|    File tempDir = MigrationFileUtils.createTempDirectory();
> 86|    File csvFile = MigrationFileUtils.createTempFileFromClasspath(REFERENCE_DATA_PATH +
ctx.getMigrationElements().get(0).getReferenceDataFilename(),
> 87|        tempDir);
> 88|    List<RoleAccess> result = new ArrayList<>();
> 89|    RoleAccess current = null;
> 90|    try (Scanner scanner = new Scanner(csvFile)) {
> 91|        while (scanner.hasNextLine()) {
> 92|            String[] line = scanner.nextLine().split(";");
> 93|            String roleId = line[0].trim();
> 94|            String accessPath = line[1].trim();
> 95|            if (current == null || !current.roleId.equals(roleId)) {
> 96|                current = new RoleAccess(roleId);
> 97|                result.add(current);
> 98|            }
> 99|            current.addPath(accessPath);
>100|        }
>101|    } finally {
>102|        MigrationFileUtils.deleteTempDir(tempDir);
>103|    }
>104|    return result;
>105|}
106|
107|private List<Election> getElections(Connection conn) throws Exception {
108|    List<Election> result = new ArrayList<>();

```

**6.297 SQLCollectorInterceptor.java****Issue B04876A25A27BEE29D63FC844573CE38:****Sink :****File** admin-backend/src/main/java/no/evote/service/util/SQLCollectorInterceptor.java 26-42**Fact Name:** no.evote.service.util.SQLCollectorInterceptor.aroundInvoke

```

25|@AroundInvoke
>26|public Object aroundInvoke(final InvocationContext ctx) throws Exception {
>27|    if (disabled) {
>28|        return ctx.proceed();
>29|    }
>30|
>31|    configureLogger();
>32|
>33|    sqlCollectorAppender.setContext(ctx);
>34|    try {
>35|        return ctx.proceed();
>36|    } finally {
>37|        log.debug("[Method: " + sqlCollectorAppender.getCalledMethod() + "]);
>38|        if (sqlCollectorAppender.hasSQL()) {
>39|            log.debug("[SQL: " + sqlCollectorAppender.getSQLReport() + "]);
>40|        }
>41|    }
>42|}
43|

```

```
44|private void configureLogger() {
45|    Logger hibernateSQLLogger = Logger.getLogger("org.hibernate.SQL");
```

## 6.298 SecuredObjectsInterceptor.java

**Issue** BC48D4C45BD77F452F610F003A899403:

**Sink :**

**File** admin-backend/src/main/java/no/valg/eva/admin/backend/auditlog/SecuredObjectsInterceptor.java 42-60

**Fact** Name: no.valg.eva.admin.backend.auditlog.SecuredObjectsInterceptor.logCall

```
41|@AroundInvoke
>42|public Object logCall(final InvocationContext ctx) throws Exception {
>43|    Class<?> calledClass = ctx.getTarget().getClass();
>44|    Method calledMethod = ctx.getMethod();
>45|    Object[] parameters = ctx.getParameters();
>46|
>47|    if (skipLogging(calledClass, calledMethod)) {
>48|        return ctx.proceed();
>49|    }
>50|
>51|    UserData userData = findUserDataAsFirstParameter(parameters);
>52|    try {
>53|        Object result = ctx.proceed();
>54|        logInfo(userData, calledClass, calledMethod, parameters);
>55|        return result;
>56|    } catch (Exception e) {
>57|        logError(userData, calledClass, calledMethod, parameters, e);
>58|        throw e;
>59|    }
>60|}
61|
62|private String createInfo(Object[] parameters) {
63|    if (parameters != null) {
```

## 6.299 SecurityInterceptor.java

**Issue** FAC603D4BD73F89D2E0660DB170C0630:

**Sink :**

**File** admin-backend/src/main/java/no/evote/service/security/SecurityInterceptor.java 81-87 **Fact**

**Name:** no.evote.service.security.SecurityInterceptor.intercept

```
78|/**
79| * Interception method that controls that the user has access to the right securable objects and has the right
election and area context
80| */
>81|public Object intercept(InvocationContext ctx) throws Exception {
>82|    if (shouldCheckSecurableObjects(ctx.getMethod())) {
>83|        checkAccess(ctx);
>84|        controlContext(ctx);
>85|    }
>86|    return ctx.proceed();
>87|}
88|
89|private void checkAccess(InvocationContext ctx) {
90|    Security security = getAnnotation(ctx, Security.class);
```

## 6.300 ServiceInvocationHandler.java

**Issue** 118E63FE77EC4ED83B43C6F9CD5C0244:

**Sink :**

**File** admin-frontend/src/main/java/no/evote/service/cache/ServiceInvocationHandler.java 45-108 **Fact**

**Name:** no.evote.service.cache.ServiceInvocationHandler.invoke

```
42| * "Grunnlagsdata" i.e. data that is never changed, is marked by the @Cacheable annotation. In addition there are
methods marked with @CacheInvalidate,
43| * which invalidates the entity cache, a separate cache for non-static entities that are frequently used.
44| */
> 45|public Object invoke(final Object proxy, final Method method, final Object[] args, final boolean retry) throws
Throwable {
> 46|    if (impl == null) {
> 47|        impl = lookupImpl();
```

```

> 48|     }
> 49|
> 50|     // Check whether result is in cache
> 51|     String cacheKey = null;
> 52|     boolean cacheable = isCacheable(method);
> 53|
> 54|     if (cacheable) {
> 55|         Element result = null;
> 56|         cacheKey = generateCacheKey(method, args);
> 57|         if (serviceInvocationCache.isKeyInCache(cacheKey)) {
> 58|             result = serviceInvocationCache.get(cacheKey);
> 59|         }
> 60|
> 61|         if (result != null) {
> 62|             return result.getObjectValue();
> 63|         }
> 64|     }
> 65|
> 66|     // Check whether this invocation should invalidate an entry in the eni
> 67|     if (isCacheInvalidator(method)) {
> 68|         CacheInvalidate annotation = method.getAnnotation(CacheInvalidate.class);
> 69|         Class<?> entityClass = annotation.entityClass();
> 70|         Object entityParam = args[annotation.entityParam()];
> 71|         Long invalidateKey = null;
> 72|
> 73|         if (entityParam instanceof BaseEntity) {
> 74|             invalidateKey = ((BaseEntity) entityParam).getPk();
> 75|         } else if (entityParam instanceof Long) {
> 76|             invalidateKey = (Long) entityParam;
> 77|         }
> 78|         entityCache.remove(entityClass, invalidateKey);
> 79|     }
> 80|
> 81|     try {
> 82|         // Invoke method and put result in cache if it is cacheable
> 83|         Object result = method.invoke(impl, args);
> 84|         if (cacheable) {
> 85|             serviceInvocationCache.put(new Element(cacheKey, result));
> 86|         }
> 87|         return result;
> 88|     } catch (final InvocationTargetException ex) {
> 89|         Throwable targetException = ex.getTargetException();
> 90|
> 91|         if (targetException instanceof NoSuchEJBException
> 92|             || (targetException instanceof EJBException && ((EJBException)
targetException).getCausedByException() instanceof MarshalException)) {
> 93|             if (retry) {
> 94|                 // Force re-lookup of the EJB and try calling the method again
> 95|                 LOG.warn("Detected stale instance of " + lookupClass.toString() + ", attempting to get new and
retrying", ex);
> 96|                 impl = null;
> 97|                 return invoke(proxy, method, args, false);
> 98|             } else {
> 99|                 // Error was unrecoverable, set impl to null so that we look it up again on next invocation
>100|                 LOG.warn("Failed retrying on " + lookupClass.toString() + ", giving up", ex);
>101|                 impl = null;
>102|             }
>103|         }
>104|
>105|         throw targetException;
>106|     }
>107|
>108| }
109|
110| @Override
111| public Object invoke(final Object proxy, final Method method, final Object[] args) throws Throwable {

```

## 6.301 Small2007DatasetImporter.java

### Issue 35652B96921E2B2C03DB8CEFF1AB89AD:

#### Sink:

File admin-database/src/main/java/no/valg/eva/admin/database/tools/Small2007DatasetImporter.java 30-133  
 Fact Name: no.valg.eva.admin.database.tools.Small2007DatasetImporter.importIntoDatabase

```

27|     new Small2007DatasetImporter().importIntoDatabase();
28| }
29|

```

```
> 30|void importIntoDatabase() throws Exception {  
> 31|    new ImportTablesWrapper()  
> 32|        // Bootstrap data for Valg 2007
```

```

> 33| .with("certificate_revocation_list", bootstrapFile("certificate_revocation_list.txt"))
> 34| .with("election_event", bootstrapFile("election_event.txt"))
> 35| .with("election_event_locale", bootstrapFile("election_event_locale.txt"))
> 36| .with("election_day", bootstrapFile("election_day.txt"))
> 37|
> 38| .with("binary_data", bootstrapFile("binary_data.txt"))
> 39| .with("signing_key", bootstrapFile("signing_key.txt"))
> 40|
> 41| .with("country", bootstrapFile("country.txt"))
> 42| .with("county", bootstrapFile("county.txt"))
> 43| .with("municipality", bootstrapFile("municipality.txt"))
> 44| .with(BOROUGH, bootstrapFile("borough.txt"))
> 45| .with("county", bootstrapFile("county-postal.txt"))
> 46| .with("municipality", bootstrapFile("municipality-postal.txt"))
> 47| .with(BOROUGH, bootstrapFile("borough-municipality.txt"))
> 48| .with(BOROUGH, bootstrapFile("borough-municipality-postal.txt"))
> 49|
> 50| // Extra bootstrap data for Valg 2007 (Svalbard, other missing data)
> 51| .with("municipality", bootstrapFile("municipality-extra.txt"))
> 52| .with(BOROUGH, bootstrapFile("borough-extra.txt"))
> 53| .with("polling_district", bootstrapFile("polling_district-extra.txt"))
> 54|
> 55| .with("election_group", bootstrapFile("election_group.txt"))
> 56| .with("election", bootstrapFile("election.txt"))
> 57| .with("contest", bootstrapFile("contest.txt"))
> 58| .with("contest_area", bootstrapFile("contest_area.txt"))
> 59|
> 60| .with("text_id", bootstrapFile("text_id.txt"))
> 61| .with("locale_text", bootstrapFile("locale_text.txt"))
> 62| .with("election_event_text", bootstrapFile("election_event_text.txt"))
> 63| .with("election_text", bootstrapFile("election_text.txt"))
> 64| .with("contest_text", bootstrapFile("contest_text.txt"))
> 65|
> 66| // Data converted from Valg 2007
> 67| .with("text_id", dataFile("text_id-STD-PAR.txt"))
> 68| .with("locale_text", dataFile("locale_text-STD-PAR.txt"))
> 69| .with("party", dataFile("party-STD-PAR.txt"))
> 70| .with("party_number", dataFile("party_number.txt"))
> 71| .with("party_contest_area", dataFile("party_contest_area.txt"))
> 72| .with("polling_district", dataFile("polling_district-SAM.txt"))
> 73| .with("polling_district", dataFile("polling_district-TEL-KRT.txt"))
> 74| .with("polling_place", dataFile("polling_place-STE.txt"))
> 75| .with("polling_place", dataFile("polling_place-KRT.txt"))
> 76| .with("opening_hours", dataFile("opening_hours-KRT.txt"))
> 77|
> 78| // Bootstrap data for RBAC
> 79| .with("role", bootstrapFile("role.txt"))
> 80| .with("role_access", bootstrapFile("role_access.txt"))
> 81| .with("role_include", bootstrapFile("role_include.txt"))
> 82|
> 83| .execute();
> 84|
> 85| triggerDatabaseCopy("role", "000000", "200701");
> 86|
> 87| new ImportTablesWrapper()
> 88| // // Bootstrap data for RBAC (continues..)
> 89| .with("operator", bootstrapFile("operator.txt"))
> 90| .with("operator", bootstrapFile("operator-admin.txt"))
> 91| .with("operator_role", bootstrapFile("operator_role.txt"))
> 92| .with("operator_role", bootstrapFile("operator_role-admin.txt"))
> 93|
> 94| // Extra bootstrap data for Valg2007 (Svalbard, other missing data)
> 95| .with("polling_place", bootstrapFile("polling_place-extra.txt"))
> 96| .with("voter", bootstrapFile("voter.txt"))
> 97|
> 98| .with("reporting_unit", dataFile("reporting_unit-KOM.txt"))
> 99| .with("reporting_unit", dataFile("reporting_unit-TEL-KRT.txt"))
>100| .with("responsible_officer", dataFile("responsible_officer-KOM.txt"))
>101| .with("responsible_officer", dataFile("responsible_officer-VST.txt"))
>102| .with("report_count_category", dataFile("report_count_category-KOM.txt"))
>103| .with("mv_election_reporting_units", bootstrapFile("mv_election_reporting_units.txt"))
>104|
>105| .with("ballot", dataFile("ballot-KOM-PAR.txt"))
>106| .with("proposer", dataFile("proposer-PAR.txt"))
>107| .with("proposer", dataFile("proposer-PAR-LIS-KAL.txt"))
>108| .with("affiliation", dataFile("affiliation-PAR.txt"))
>109| .with("candidate", dataFile("candidate-PAR-KAN-anon.txt"))
>110|

```

```
>111|         .with("voter", dataFile("voter-FLK-anon-test.txt"))
>112|         .with("voter", dataFile("voter-Hvaler.txt"))
>113|         .with("voter", dataFile("voter-SorOdal.txt"))
>114|         .with("voting", dataFile("voting-FLK-anon-test.txt"))
```

```

>115|         .with("voting", dataFile("voting-FLK-FOR-anon-test.txt"))
>116|         .with("voting", dataFile("voting-FLK-FRM-anon-test.txt"))
>117|         .with("voting", dataFile("voting-Hvaler.txt"))
>118|         .with("voting", dataFile("voting-SorOdal.txt"))
>119|
>120|         .with("contest_report", dataFile("contest_report-KOM.txt"))
>121|         .with("vote_count", dataFile("vote_count-TVI-STM.txt"))
>122|         .with("ballot_count", dataFile("ballot_count-PAR-TVI-STM.txt"))
>123|         .with("cast_vote", dataFile("cast_vote-PAR-RET-test.txt"))
>124|         .with("candidate_vote", dataFile("candidate_vote-PAR-KAN-RET-anon-test.txt"))
>125|         .with("cast_vote", dataFile("cast_vote-PAR-EST-SLE-test.txt"))
>126|         .with("candidate_vote", dataFile("candidate_vote-PAR-KAN-EST-anon-test.txt"))
>127|         .with("candidate_vote", dataFile("candidate_vote-PAR-KAN-SLE-anon-test.txt"))
>128|         .with("manual_contest_voting", dataFile("manual_contest_voting.txt"))
>129|
>130|         .execute();
>131|
>132|     copyToValg2008();
>133|}
134|
135|private void copyToValg2008() {
136|    String fromElectionEventId = "200701";

```

## [Low] Poor Error Handling / Program Catches NullPointerException

*Errors*

**Analysis:** Bad Practice

### Abstract

It is generally a bad practice to catch `NullPointerException`.

### Explanation

Programmers typically catch `NullPointerException` under three circumstances:

1. The program contains a null pointer dereference. Catching the resulting exception was easier than fixing the underlying problem.
2. The program explicitly throws a `NullPointerException` to signal an error condition.
3. The code is part of a test harness that supplies unexpected input to the classes under test.

Of these three circumstances, only the last is acceptable.

**Example:** The following code mistakenly catches a `NullPointerException`.

```

try {
    mysteryMethod();
} catch (NullPointerException npe) {
}

```

### Recommendation

The program should not dereference null pointers. If you cannot eliminate the cause of the null pointer dereference, you must carefully review the code to make sure that the exception is handled in such a way that the program does not enter into an unexpected or illegal state.

If the `NullPointerException` is being thrown explicitly, change the program to throw an exception derived from `RuntimeException` or `Error` instead.

## 6.302 CountingImportServiceBean.java

**Issue** 77A473AB27407F48187E91715B9B5D41:

**Sink :**

**File** admin-backend/src/main/java/no/evote/service/counting/CountingImportServiceBean.java 266-269

```

263|} catch (EvoteException e) {
264|    LOGGER.error(e.getMessage(), e);
265|    throw e;
>266|} catch (NullPointerException e) {
>267|    LOGGER.error(e.getMessage(), e);
>268|    throw new EvoteException("Missing content in EML.", e);
>269|} catch (Exception e) {
270|    LOGGER.error(e.getMessage(), e);
271|    throw new EvoteException(e.getMessage(), e);
272|} finally {

```

## [Low] Poor Logging Practice / Logger Not Declared Static Final

*Encapsulation*

**Analysis:** Bad Practice

### Abstract

Declare loggers to be static and final.

### Explanation

It is good programming practice to share a single logger object between all of the instances of a particular class and to use the same logger for the duration of the program.

**Example 1:** The following statement errantly declares a non-static logger.

```

private final Logger logger =
    Logger.getLogger(MyClass.class);

```

### Recommendation

Declare loggers to be static and final.

**Example 2:** The code in Example 1 could be rewritten in the following way:

```

private final static Logger logger =
    Logger.getLogger(MyClass.class);

```

## 6.303 AuditLogInterceptor.java

**Issue** 765EB7E952C7EE149DA6A4BA344567A0:

**Sink :**

**File** admin-backend/src/main/java/no/valg/eva/admin/backend/auditlog/AuditLogInterceptor.java 21

**Fact** Name: no.valg.eva.admin.backend.auditlog.AuditLogInterceptor.logger

```

18| * The implementation is robust, so that if audit logging fails, the business operation will still proceed as
normal. 19| */
20|public class AuditLogInterceptor implements Serializable {
>21|    private final Logger logger = Logger.getLogger(AuditLogInterceptor.class);
22|
23|    @Inject
24|    private AuditLogServiceBean auditLogService;

```

## 6.304 AuditLogServiceBean.java

**Issue** B7C7F8F451B0A51919743464B985D33F:

**Sink :**

**File** admin-backend-common/src/main/java/no/valg/eva/admin/backend/common/auditlog/AuditLogServiceBean.java 15 **Fact**

**Name:** no.valg.eva.admin.backend.common.auditlog.AuditLogServiceBean.auditLog

```

12|@Singleton
13|public class AuditLogServiceBean {
14|    private static final String AUDIT_LOGGER_NAME = "AuditLogger";
>15|    private final Logger auditLog;
16|
17|    public AuditLogServiceBean() {
18|        auditLog = Logger.getLogger(AUDIT_LOGGER_NAME);

```



## 6.305 EnumCache.java

Issue 8E4ECD21076C171D656F08E2C92D116B:

Sink :

File admin-frontend/src/main/java/no/evote/presentation/cache/EnumCache.java 18

Fact Name: no.evote.presentation.cache.EnumCache.log

```
15|private final Map<String, Object> propertCache = new HashMap<>();
16|private final Map<String, Class<?>> baseCache = new HashMap<>();
17|private static EnumCache staticEnumCache = null;
>18|private final Logger log = Logger.getLogger(EnumCache.class);
19|
20|public static EnumCache instance() {
21|    if (staticEnumCache == null) {
```

## 6.306 EqualsHashCodeUtil.java

Issue 25E9EA764DC39E2E6ECBB2D25AE5B91A:

Sink :

File admin-common/src/main/java/no/evote/util/EqualsHashCodeUtil.java 14

Fact Name: no.evote.util.EqualsHashCodeUtil.log

```
11|public final class EqualsHashCodeUtil {
12|
13|    private static final int PRIME = 31;
>14|    private static Logger log = Logger.getLogger(EqualsHashCodeUtil.class);
15|
16|    private EqualsHashCodeUtil() {
```

## 6.307 ExportServiceBean.java

Issue 3F4706D49EAF0672E9F6816B114AB3BC:

Sink :

File admin-backend/src/main/java/no/evote/service/ExportServiceBean.java 283

Fact Name: no.evote.service.ExportServiceBean.logger

```
280|private static final String TEXT_ID = "text_id";
281|// @formatter:on
282|private static final String ELECTION_EVENT = "election_event";
>283|private final Logger logger = Logger.getLogger(ExportServiceBean.class);
284|@Inject
285|private CountryServiceBean countryService;
286|@Inject
```

## 6.308 ImportOperatorServiceEjb.java

Issue 99019C050DC7BEE8B06C2D344B2197E3:

Sink :

File admin-rbac/src/main/java/no/valg/eva/admin/rbac/service/ImportOperatorServiceEjb.java 46

Fact Name: no.valg.eva.admin.rbac.service.ImportOperatorServiceEjb.logger

```
43|private static final String EARLY_VOTE_RECEIVER = "stemmemottak_forhånd";
44|private static final String OPPELLING_STEMMEKRETS = "ansvarlig_valglokale";
45|private static final String VOTE_RECEIVER = "stemmemottak_valgting";
>46|private final Logger logger = Logger.getLogger(ImportOperatorServiceEjb.class);
47|@Inject
48|private RoleRepository roleRepository;
49|@Inject
```

## 6.309 LifecycleFilter.java

Issue 9E1938C3AC8A81350EA2AA37CB6ED5B6:

Sink :

File admin-frontend/src/main/java/no/evote/lifecycle/LifecycleFilter.java 22

Fact Name: no.evote.lifecycle.LifecycleFilter.log

```
20|public class LifecycleFilter implements Filter {
21|
>22|    private final Logger log = Logger.getLogger(LifecycleFilter.class);
23|
24|    @Inject
25|    private SystemPasswordService passwordService;
```

## 6.310 LifecycleListener.java

**Issue** BCA5AEE81A0BC0648DA65DAA6E96F922:

**Sink :**

**File** admin-frontend/src/main/java/no/evote/lifecycle/LifecycleListener.java 14 **Fact**  
**Name:** no.evote.lifecycle.LifecycleListener.log

```
11|import org.opensaml.xml.ConfigurationException;
12|
13|public class LifecycleListener implements ServletContextListener {
>14|     private Logger log;
15|
16|     @Override
17|     public void contextDestroyed(final ServletContextEvent contextEvent) {
```

## 6.311 RBACTestFixture.java

**Issue** 7C3702281741B432070227444DC05CBA:

**Sink :**

**File** admin-backend-testtools/src/main/java/no/evote/service/backendmock/RBACTestFixture.java 55  
**Fact** Name: no.evote.service.backendmock.RBACTestFixture.log

```
52|public static final String OPERATOR_ID_0905861112 = "0905861112";
53|public static final String ROLE_NAME_VOTING_COUNT_ELECTION_DAY_ALL = "vtngCntElctnDyAll";
54|public static final String ROLE_NAME_VOTING_COUNT = "roleVotingCount";
>55|private final Logger log = Logger.getLogger(RBACTestFixture.class);
56|private Operator operator;
57|private Operator operatorRoot;
58|private Operator operator1;
```

## 6.312 SQLCollectorInterceptor.java

**Issue** 53D94B1D6CF761092AE35DBA40F87B23:

**Sink :**

**File** admin-backend/src/main/java/no/evote/service/util/SQLCollectorInterceptor.java 16 **Fact**  
**Name:** no.evote.service.util.SQLCollectorInterceptor.log

```
13|import org.apache.log4j.Logger;
14|
15|public class SQLCollectorInterceptor implements Serializable {
>16|     private final Logger log = Logger.getLogger(SQLCollectorInterceptor.class);
17|     private final SQLCollectorAppender sqlCollectorAppender = new SQLCollectorAppender();
18|     private boolean disabled = true;
```

## 6.313 SecuredObjectsInterceptor.java

**Issue** 8B40AA0F33AAD2275B9495A89487EC8C:

**Sink :**

**File** admin-backend/src/main/java/no/valg/eva/admin/backend/auditlog/SecuredObjectsInterceptor.java 29  
**Fact** Name: no.valg.eva.admin.backend.auditlog.SecuredObjectsInterceptor.logger

```
26|private static final String DELIMITER = ", ";
27|private static final char PARAMETER_SEPARATOR = ',';
28|
>29|private final Logger logger = Logger.getLogger(LOGGER_NAME);
30|
31|private ErrorCodesMapper errorCodesMapper = new ErrorCodesMapper();
```

## 6.314 SecurityInterceptor.java

**Issue** 24A11D9640BAEA6A9F9474C5E99927FF:

**Sink :**

**File** admin-backend/src/main/java/no/evote/service/security/SecurityInterceptor.java 57 **Fact**  
**Name:** no.evote.service.security.SecurityInterceptor.log

```
54|public static final String USER_DATA_KEY = "userData";
55|private static final long serialVersionUID = 1L;
56|private static final String ACCESS_CACHE_VERIFIED = "accessCacheVerified";
>57|private final transient Logger log = Logger.getLogger(SecurityInterceptor.class);
58|
```

```
59|@Resource
60|private TransactionSynchronizationRegistry registry;
```

## 6.315 SystemDateTimeServlet.java

**Issue** 3253BC8E289C657F517561C7B0E778E4:

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/common/web/SystemDateTimeServlet.java 34

**Fact Name:** no.valg.eva.admin.common.web.SystemDateTimeServlet.logger

```
31|static final String VALUE_TIME_SETTING_FIXED = "fixed";
32|static final String VALUE_TIME_SETTING_SYSTEM = "system";
33|
>34|private final Logger logger = Logger.getLogger(SystemDateTimeServlet.class);
35|private final Boolean canChangeTime;
36|
37|public SystemDateTimeServlet() {
```

**[Low] Poor Style / Confusing Naming**

*Code Quality*

**Analysis:** Bad Practice

### Abstract

The class contains a field and a method with the same name.

### Explanation

It is confusing to have a member field and a method with the same name. It makes it easy for a programmer to accidentally call the method when attempting to access the field or vice versa.

### Example 1:

```
public class Totaller {
    private int total;
    public int total() {
        ...
    }
}
```

### Recommendation

Rename either the method or the field. If the method returns the field, consider following the standard getter/setter naming convention.

**Example 2:** The code in Example 1 could be rewritten in the following way:

```
public class Totaller {    private int total;    public int getTotal() {    ...
    }
}
```

## 6.316 AbstractAuditEvent.java

**Issue** 30ED082F5EF8037CAAEC09CF3DE8A3C4:

**Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/common/auditlog/auditevents/AbstractAuditEvent.java 13 **Fact**

**Name:** no.valg.eva.admin.common.auditlog.auditevents.AbstractAuditEvent.electionEventId

```
11|public abstract class AbstractAuditEvent implements Serializable {
12|    private final String uid;
>13|    private final String electionEventId;
14|    private final String roleId;
15|    private final AreaPath roleAreaPath;
16|    private final ElectionPath roleElectionPath;
```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/common/auditlog/auditevents/AbstractAuditEvent.java 13 **Fact**  
**Name:** no.valg.eva.admin.common.auditlog.auditevents.AbstractAuditEvent.electionEventId

```
11|public abstract class AbstractAuditEvent implements Serializable {
12|    private final String uid;
>13|    private final String electionEventId;
14|    private final String roleId;
15|    private final AreaPath roleAreaPath;
16|    private final ElectionPath roleElectionPath;
```

**Issue 58B1775FB30EB9D746770797D5023C93:****Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/common/auditlog/auditevents/AbstractAuditEvent.java 17  
**Fact Name:** no.valg.eva.admin.common.auditlog.auditevents.AbstractAuditEvent.clientIpAddress

```
14|private final String roleId;
15|private final AreaPath roleAreaPath;
16|private final ElectionPath roleElectionPath;
>17|private final InetAddress clientIpAddress;
18|
19|public AbstractAuditEvent(String uid, String electionEventId, String roleId, AreaPath roleAreaPath, ElectionPath
roleElectionPath,
20|    InetAddress clientIpAddress) {
```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/common/auditlog/auditevents/AbstractAuditEvent.java 17  
**Fact Name:** no.valg.eva.admin.common.auditlog.auditevents.AbstractAuditEvent.clientIpAddress

```
14|private final String roleId;
15|private final AreaPath roleAreaPath;
16|private final ElectionPath roleElectionPath;
>17|private final InetAddress clientIpAddress;
18|
19|public AbstractAuditEvent(String uid, String electionEventId, String roleId, AreaPath roleAreaPath, ElectionPath
roleElectionPath,
20|    InetAddress clientIpAddress) {
```

**Issue 5B19AFF57536FF13C65CAFEAF81B9C60:****Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/common/auditlog/auditevents/AbstractAuditEvent.java 16  
**Fact Name:** no.valg.eva.admin.common.auditlog.auditevents.AbstractAuditEvent.roleElectionPath

```
13|private final String electionEventId;
14|private final String roleId;
15|private final AreaPath roleAreaPath;
>16|private final ElectionPath roleElectionPath;
17|private final InetAddress clientIpAddress; 18|
19|public AbstractAuditEvent(String uid, String electionEventId, String roleId, AreaPath roleAreaPath, ElectionPath
roleElectionPath,
```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/common/auditlog/auditevents/AbstractAuditEvent.java 16  
**Fact Name:** no.valg.eva.admin.common.auditlog.auditevents.AbstractAuditEvent.roleElectionPath

```
13|private final String electionEventId;
14|private final String roleId;
15|private final AreaPath roleAreaPath;
>16|private final ElectionPath roleElectionPath;
17|private final InetAddress clientIpAddress; 18|
19|public AbstractAuditEvent(String uid, String electionEventId, String roleId, AreaPath roleAreaPath, ElectionPath
roleElectionPath,
```

**Issue 69DF89F74AC2241E2F7E85D797F1E4BC:****Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/common/auditlog/auditevents/AbstractAuditEvent.java 12  
**Fact Name:** no.valg.eva.admin.common.auditlog.auditevents.AbstractAuditEvent.uid

```
9|import no.valg.eva.admin.common.ElectionPath;
10|
11|public abstract class AbstractAuditEvent implements Serializable {
>12|    private final String uid;
13|    private final String electionEventId;
14|    private final String roleId;
15|    private final AreaPath roleAreaPath;
```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/common/auditlog/auditevents/AbstractAuditEvent.java 12  
**Fact Name:** no.valg.eva.admin.common.auditlog.auditevents.AbstractAuditEvent.uid

```

9|import no.valg.eva.admin.common.ElectionPath;
10|
11|public abstract class AbstractAuditEvent implements Serializable {
>12|     private final String uid;
13|     private final String electionEventId;
14|     private final String roleId;
15|     private final AreaPath roleAreaPath;

```

**Issue CFF39CD22523E910FB79143B584F4970:****Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/common/auditlog/auditevents/AbstractAuditEvent.java 14

**Fact Name:** no.valg.eva.admin.common.auditlog.auditevents.AbstractAuditEvent.roleId

```

11|public abstract class AbstractAuditEvent implements Serializable {
12|     private final String uid;
13|     private final String electionEventId;
>14|     private final String roleId;
15|     private final AreaPath roleAreaPath;
16|     private final ElectionPath roleElectionPath;
17|     private final InetAddress clientIpAddress;

```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/common/auditlog/auditevents/AbstractAuditEvent.java 14

**Fact Name:** no.valg.eva.admin.common.auditlog.auditevents.AbstractAuditEvent.roleId

```

11|public abstract class AbstractAuditEvent implements Serializable { 12|     private final String uid;
13|     private final String electionEventId;
>14|     private final String roleId;
15|     private final AreaPath roleAreaPath;
16|     private final ElectionPath roleElectionPath;
17|     private final InetAddress clientIpAddress;

```

**Issue DC7BoA96214CDC95DBF6EDB2374346CE:****Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/common/auditlog/auditevents/AbstractAuditEvent.java 15 **Fact**

**Name:** no.valg.eva.admin.common.auditlog.auditevents.AbstractAuditEvent.roleAreaPath

```

12|private final String uid;
13|private final String electionEventId;
14|private final String roleId;
>15|private final AreaPath roleAreaPath;
16|private final ElectionPath roleElectionPath;
17|private final InetAddress clientIpAddress;

```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/common/auditlog/auditevents/AbstractAuditEvent.java 15

**Fact Name:** no.valg.eva.admin.common.auditlog.auditevents.AbstractAuditEvent.roleAreaPath

```

12|private final String uid;
13|private final String electionEventId;
14|private final String roleId;
>15|private final AreaPath roleAreaPath;
16|private final ElectionPath roleElectionPath; 17|private final InetAddress clientIpAddress;

```

## 6.317 Accesses.java

**Issue BoE6D3A80B6882269ED4CB7C53D49F74:****Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/common/rbac/Accesses.java 271

**Fact Name:** no.valg.eva.admin.common.rbac.Accesses.paths

```

269|private Access access;
270|private Accesses[] aggregated;
>271|private String[] paths;
272|
273|Accesses(Access access) {
274|     this.access = access;

```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/common/rbac/Accesses.java 271

**Fact Name:** no.valg.eva.admin.common.rbac.Accesses.paths

```

269|private Access access;
270|private Accesses[] aggregated;
>271|private String[] paths;
272|
273|Accesses(Access access) {

```

```
274| this.access = access;
```

## 6.318 ApprovedCastBallotRefForApprovedFinalCount.java

**Issue** 3EF71B46697F5F48811A49D9B1D27DBC:

**Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/common/counting/model/ApprovedCastBallotRefForApprovedFinalCount.java 13

**Fact** Name: no.valg.eva.admin.common.counting.model.ApprovedCastBallotRefForApprovedFinalCount.ballotId

```
10|import org.apache.commons.lang3.builder.ToStringBuilder;
11|
12|public final class ApprovedCastBallotRefForApprovedFinalCount extends CastBallotRefForApprovedFinalCount { >13|
private final BallotId ballotId;
14|
15| public ApprovedCastBallotRefForApprovedFinalCount(
16|     ReportingUnitTypeId reportingUnitTypeId, CountContext countContext, AreaPath countingAreaPath,
CastBallotId castBallotId, BallotId ballotId) {
```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/common/counting/model/ApprovedCastBallotRefForApprovedFinalCount.java 13

**Fact** Name: no.valg.eva.admin.common.counting.model.ApprovedCastBallotRefForApprovedFinalCount.ballotId

```
10|import org.apache.commons.lang3.builder.ToStringBuilder;
11|
12|public final class ApprovedCastBallotRefForApprovedFinalCount extends CastBallotRefForApprovedFinalCount {
>13| private final BallotId ballotId;
14|
15| public ApprovedCastBallotRefForApprovedFinalCount(
16|     ReportingUnitTypeId reportingUnitTypeId, CountContext countContext, AreaPath countingAreaPath,
CastBallotId castBallotId, BallotId ballotId) {
```

## 6.319 ApprovedFinalCountRef.java

**Issue** 1D9A2F7616BEED50797361B56B2B66AF:

**Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/common/counting/model/ApprovedFinalCountRef.java 15

**Fact** Name: no.valg.eva.admin.common.counting.model.ApprovedFinalCountRef.reportingUnitTypeId

```
12|import org.apache.commons.lang3.builder.ToStringBuilder; 13|
14|public class ApprovedFinalCountRef implements Serializable {
>15| private final ReportingUnitTypeId reportingUnitTypeId;
16| private final CountContext countContext;
17| private final AreaPath countingAreaPath;
```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/common/counting/model/ApprovedFinalCountRef.java 15

**Fact** Name: no.valg.eva.admin.common.counting.model.ApprovedFinalCountRef.reportingUnitTypeId

```
12|import org.apache.commons.lang3.builder.ToStringBuilder; 13|
14|public class ApprovedFinalCountRef implements Serializable {
>15| private final ReportingUnitTypeId reportingUnitTypeId;
16| private final CountContext countContext;
17| private final AreaPath countingAreaPath;
```

**Issue** 772D8943D8AFA0A156084A5BBE86E98A:

**Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/common/counting/model/ApprovedFinalCountRef.java 16 **Fact**

Name: no.valg.eva.admin.common.counting.model.ApprovedFinalCountRef.countContext

```
14|public class ApprovedFinalCountRef implements Serializable {
15| private final ReportingUnitTypeId reportingUnitTypeId;
>16| private final CountContext countContext;
17| private final AreaPath countingAreaPath; 18|
19| public ApprovedFinalCountRef(ReportingUnitTypeId reportingUnitTypeId, CountContext countContext, AreaPath
countingAreaPath) {
```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/common/counting/model/ApprovedFinalCountRef.java 16 **Fact**

Name: no.valg.eva.admin.common.counting.model.ApprovedFinalCountRef.countContext

```
14|public class ApprovedFinalCountRef implements Serializable {
15| private final ReportingUnitTypeId reportingUnitTypeId;
>16| private final CountContext countContext;
17| private final AreaPath countingAreaPath;
18|
19| public ApprovedFinalCountRef(ReportingUnitTypeId reportingUnitTypeId, CountContext countContext, AreaPath
countingAreaPath) {
```

**Issue 932993BE632002DF14AC56991ED81BE8:****Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/common/counting/model/ApprovedFinalCountRef.java 17 **Fact**  
**Name:** no.valg.eva.admin.common.counting.model.ApprovedFinalCountRef.countingAreaPath

```
14|public class ApprovedFinalCountRef implements Serializable {
15|     private final ReportingUnitTypeId reportingUnitTypeId;
16|     private final CountContext countContext;
>17|     private final AreaPath countingAreaPath; 18|
19|     public ApprovedFinalCountRef(ReportingUnitTypeId reportingUnitTypeId, CountContext countContext, AreaPath
countingAreaPath) {
20|         this.reportingUnitTypeId = reportingUnitTypeId;
```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/common/counting/model/ApprovedFinalCountRef.java 17 **Fact**  
**Name:** no.valg.eva.admin.common.counting.model.ApprovedFinalCountRef.countingAreaPath

```
14|public class ApprovedFinalCountRef implements Serializable {
15|     private final ReportingUnitTypeId reportingUnitTypeId;
16|     private final CountContext countContext;
>17|     private final AreaPath countingAreaPath; 18|
19|     public ApprovedFinalCountRef(ReportingUnitTypeId reportingUnitTypeId, CountContext countContext, AreaPath
countingAreaPath) {
20|         this.reportingUnitTypeId = reportingUnitTypeId;
```

## 6.320 AreaPath.java

**Issue 8C99005F110F6A8BC12ED61ECoE49623:****Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/common/AreaPath.java 37  
**Fact Name:** no.valg.eva.admin.common.AreaPath.path

```
34|private static final int GROUP_NO_POLLING_STATION = 15;
35|private static final Pattern AREA_PATH_PATTERN = Pattern.compile(AREA_PATH_REGEX);
36|private static final String SAMI_COUNTY_ID = "00";
>37|private final String path;
38|private final AreaLevelEnum level;
39|private final String electionEventId;
40|private final String countryId;
```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/common/AreaPath.java 37  
**Fact Name:** no.valg.eva.admin.common.AreaPath.path

```
34|private static final int GROUP_NO_POLLING_STATION = 15;
35|private static final Pattern AREA_PATH_PATTERN = Pattern.compile(AREA_PATH_REGEX);
36|private static final String SAMI_COUNTY_ID = "00";
>37|private final String path;
38|private final AreaLevelEnum level;
39|private final String electionEventId;
40|private final String countryId;
```

## 6.321 AuditEvent.java

**Issue 0D2AC57977C8B002B072DA44584AC042:****Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/common/auditlog/auditevents/AuditEvent.java 31  
**Fact Name:** no.valg.eva.admin.common.auditlog.auditevents.AuditEvent.timestamp

```
28|public abstract class AuditEvent extends AbstractAuditEvent implements Serializable { 29|     public static final
String OBJECT_CLASSES_METHOD_NAME = "objectClasses";
30|
>31|     private final DateTime timestamp;
32|     private final String detail;
33|     private final AuditEventType eventType;
34|     private final Process process;
```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/common/auditlog/auditevents/AuditEvent.java 31  
**Fact Name:** no.valg.eva.admin.common.auditlog.auditevents.AuditEvent.timestamp

```
28|public abstract class AuditEvent extends AbstractAuditEvent implements Serializable { 29|     public static final
String OBJECT_CLASSES_METHOD_NAME = "objectClasses";
30|
```

```

>31| private final DateTime timestamp;
32| private final String detail;
33| private final AuditEventType eventType;
34| private final Process process;

```

**Issue 5AFAC2F336DD321FE1F570846B059649:****Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/common/auditlog/auditevents/AuditEvent.java 32

**Fact Name:** no.valg.eva.admin.common.auditlog.auditevents.AuditEvent.detail

```

29|public static final String OBJECT_CLASSES_METHOD_NAME = "objectClasses";
30|
31|private final DateTime timestamp;
>32|private final String detail;
33|private final AuditEventType eventType;
34|private final Process process;
35|private final Outcome outcome;

```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/common/auditlog/auditevents/AuditEvent.java 32

**Fact Name:** no.valg.eva.admin.common.auditlog.auditevents.AuditEvent.detail

```

29|public static final String OBJECT_CLASSES_METHOD_NAME = "objectClasses";
30|
31|private final DateTime timestamp;
>32|private final String detail;
33|private final AuditEventType eventType;
34|private final Process process;
35|private final Outcome outcome;

```

**Issue 5F71DBECACC441E6B1F6C8D2E9A7C1B2:****Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/common/auditlog/auditevents/AuditEvent.java 34

**Fact Name:** no.valg.eva.admin.common.auditlog.auditevents.AuditEvent.process

```

31|private final DateTime timestamp;
32|private final String detail;
33|private final AuditEventType eventType;
>34|private final Process process;
35|private final Outcome outcome;
36|
37|protected AuditEvent(String uid, String electionEventId, String roleId, AreaPath roleAreaPath, ElectionPath
roleElectionPath,

```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/common/auditlog/auditevents/AuditEvent.java 34

**Fact Name:** no.valg.eva.admin.common.auditlog.auditevents.AuditEvent.process

```

31|private final DateTime timestamp;
32|private final String detail;
33|private final AuditEventType eventType;
>34|private final Process process;
35|private final Outcome outcome; 36|
37|protected AuditEvent(String uid, String electionEventId, String roleId, AreaPath roleAreaPath, ElectionPath
roleElectionPath,

```

**Issue 82B45A62F8188D2D4A36D4E85B6AEDD9:****Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/common/auditlog/auditevents/AuditEvent.java 33

**Fact Name:** no.valg.eva.admin.common.auditlog.auditevents.AuditEvent.eventType

```

31|private final DateTime timestamp;
32|private final String detail;
>33|private final AuditEventType eventType;
34|private final Process process;
35|private final Outcome outcome;

```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/common/auditlog/auditevents/AuditEvent.java 33

**Fact Name:** no.valg.eva.admin.common.auditlog.auditevents.AuditEvent.eventType

```

31|private final DateTime timestamp;
32|private final String detail;
>33|private final AuditEventType eventType;
34|private final Process process;
35|private final Outcome outcome;

```

**Issue DEF51A16BA6FC3FED512EC3E896DB7F9:****Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/common/auditlog/auditevents/AuditEvent.java 35



**Fact Name:** no.valg.eva.admin.common.auditlog.auditevents.AuditEvent.outcome

```

32|private final String detail;
33|private final AuditEventType eventType;
34|private final Process process;
>35|private final Outcome outcome;
36|
37|protected AuditEvent(String uid, String electionEventId, String roleId, AreaPath roleAreaPath, ElectionPath
roleElectionPath,
38|     InetAddress clientIpAddress, DateTime timestamp, AuditEventType eventType, Process process, Outcome
outcome, String detail) {

```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/common/auditlog/auditevents/AuditEvent.java 35

**Fact Name:** no.valg.eva.admin.common.auditlog.auditevents.AuditEvent.outcome

```

32|private final String detail;
33|private final AuditEventType eventType;
34|private final Process process;
>35|private final Outcome outcome; 36|
37|protected AuditEvent(String uid, String electionEventId, String roleId, AreaPath roleAreaPath, ElectionPath
roleElectionPath,
38|     InetAddress clientIpAddress, DateTime timestamp, AuditEventType eventType, Process process, Outcome
outcome, String detail) {

```

## 6.322 BallotId.java

**Issue** 59681AA54FF6CED065AFA2BCCD8E365C:

**Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/common/counting/model/BallotId.java 8

**Fact Name:** no.valg.eva.admin.common.counting.model.BallotId.id

```

5|import org.apache.commons.lang3.builder.EqualsBuilder; 6|
7|public final class BallotId implements Serializable {
> 8|     private final String id;
9|
10|     public BallotId(String id) {
11|         this.id = id;

```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/common/counting/model/BallotId.java 8

**Fact Name:** no.valg.eva.admin.common.counting.model.BallotId.id

```

5|import org.apache.commons.lang3.builder.EqualsBuilder;
6|
7|public final class BallotId implements Serializable {
> 8|     private final String id;
9|
10|     public BallotId(String id) {
11|         this.id = id;

```

## 6.323 BallotRejectionId.java

**Issue** 89B574E06C220E5F7441139CB09E727A:

**Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/common/counting/model/BallotRejectionId.java 8

**Fact Name:** no.valg.eva.admin.common.counting.model.BallotRejectionId.id

```

5|import org.apache.commons.lang3.builder.EqualsBuilder;
6|
7|public final class BallotRejectionId implements Serializable {
> 8|     private final String id;
9|
10|     public BallotRejectionId(String id) {
11|         this.id = id;

```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/common/counting/model/BallotRejectionId.java 8

**Fact Name:** no.valg.eva.admin.common.counting.model.BallotRejectionId.id

```

5|import org.apache.commons.lang3.builder.EqualsBuilder;
6|
7|public final class BallotRejectionId implements Serializable {
> 8|     private final String id;
9|
10|     public BallotRejectionId(String id) {
11|         this.id = id;

```

## 6.324 BaseApproveRejectedCountController.java

### Issue 7A981DF489F72D514594374F328FE993:

#### Source :

File admin-frontend/src/main/java/no/valg/eva/admin/frontend/counting/ctrls/BaseApproveRejectedCountController.java 56

Fact Name: no.valg.eva.admin.frontend.counting.ctrls.BaseApproveRejectedCountController.reportingUnitTypeId

```

54|protected final Count finalCount;
55|protected List<BallotCount> newBallotCounts;
>56|protected ReportingUnitTypeId reportingUnitTypeId;
57|@Inject
58|private PageTitleMetaBuilder pageTitleMetaBuilder; 59|@Inject

```

#### Sink :

File admin-frontend/src/main/java/no/valg/eva/admin/frontend/counting/ctrls/BaseApproveRejectedCountController.java 56

Fact Name: no.valg.eva.admin.frontend.counting.ctrls.BaseApproveRejectedCountController.reportingUnitTypeId

```

54|protected final Count finalCount;
55|protected List<BallotCount> newBallotCounts;
>56|protected ReportingUnitTypeId reportingUnitTypeId;
57|@Inject
58|private PageTitleMetaBuilder pageTitleMetaBuilder;
59|@Inject

```

## 6.325 CastBallotBinaryData.java

### Issue 67206C8E40777C82D2138B2800712A3F:

#### Source :

File admin-common/src/main/java/no/valg/eva/admin/common/counting/model/CastBallotBinaryData.java 11

Fact Name: no.valg.eva.admin.common.counting.model.CastBallotBinaryData.mimeType

```

9|public final class CastBallotBinaryData implements Serializable {
10|     private final String fileName;
>11|     private final String mimeType;
12|     private final byte[] bytes;
13|
14|     public CastBallotBinaryData(String fileName, String mimeType, byte[] bytes) {

```

#### Sink :

File admin-common/src/main/java/no/valg/eva/admin/common/counting/model/CastBallotBinaryData.java 11

Fact Name: no.valg.eva.admin.common.counting.model.CastBallotBinaryData.mimeType

```

9|public final class CastBallotBinaryData implements Serializable {
10|     private final String fileName;
>11|     private final String mimeType;
12|     private final byte[] bytes;
13|
14|     public CastBallotBinaryData(String fileName, String mimeType, byte[] bytes) {

```

### Issue 919B1E77B012C80AE6F75B2345D24F90:

#### Source :

File admin-common/src/main/java/no/valg/eva/admin/common/counting/model/CastBallotBinaryData.java 12

Fact Name: no.valg.eva.admin.common.counting.model.CastBallotBinaryData.bytes

```

9|public final class CastBallotBinaryData implements Serializable {
10|     private final String fileName;
11|     private final String mimeType;
>12|     private final byte[] bytes;
13|
14|     public CastBallotBinaryData(String fileName, String mimeType, byte[] bytes) { 15|         this.fileName =
fileName;

```

#### Sink :

File admin-common/src/main/java/no/valg/eva/admin/common/counting/model/CastBallotBinaryData.java 12

Fact Name: no.valg.eva.admin.common.counting.model.CastBallotBinaryData.bytes

```

9|public final class CastBallotBinaryData implements Serializable {
10|     private final String fileName;
11|     private final String mimeType;

```

```

>12| private final byte[] bytes;
13|
14| public CastBallotBinaryData(String fileName, String mimeType, byte[] bytes) { 15| this.fileName =
fileName;

```

**Issue 9AA7E9A72E1878BAAEA869EF71CC6846:****Source :****File** admin-common/src/main/java/no/valg/eva/admin/common/counting/model/CastBallotBinaryData.java 10**Fact Name:** no.valg.eva.admin.common.counting.model.CastBallotBinaryData.fileName

```

7|import org.apache.commons.lang3.builder.ToStringBuilder;
8|
9|public final class CastBallotBinaryData implements Serializable {
>10| private final String fileName;
11| private final String mimeType;
12| private final byte[] bytes;

```

**Sink :****File** admin-common/src/main/java/no/valg/eva/admin/common/counting/model/CastBallotBinaryData.java 10**Fact Name:** no.valg.eva.admin.common.counting.model.CastBallotBinaryData.fileName

```

7|import org.apache.commons.lang3.builder.ToStringBuilder;
8|
9|public final class CastBallotBinaryData implements Serializable {
>10| private final String fileName;
11| private final String mimeType;
12| private final byte[] bytes;

```

**6.326 CastBallotId.java****Issue 8224AA23134684661B7873621412A0EC:****Source :****File** admin-common/src/main/java/no/valg/eva/admin/common/counting/model/CastBallotId.java 8**Fact Name:** no.valg.eva.admin.common.counting.model.CastBallotId.id

```

5|import org.apache.commons.lang3.builder.EqualsBuilder; 6|
7|public final class CastBallotId implements Serializable {
> 8| private final String id;
9|
10| public CastBallotId(String id) {
11| this.id = id;

```

**Sink :****File** admin-common/src/main/java/no/valg/eva/admin/common/counting/model/CastBallotId.java 8**Fact Name:** no.valg.eva.admin.common.counting.model.CastBallotId.id

```

5|import org.apache.commons.lang3.builder.EqualsBuilder; 6|
7|public final class CastBallotId implements Serializable {
> 8| private final String id;
9|
10| public CastBallotId(String id) {
11| this.id = id;

```

**6.327 CastBallotRefForApprovedFinalCount.java****Issue 2436F8DACEF5CEA39FD672DFEA80BA5E:****Source :****File** admin-common/src/main/java/no/valg/eva/admin/common/counting/model/CastBallotRefForApprovedFinalCount.java 15**Fact Name:** no.valg.eva.admin.common.counting.model.CastBallotRefForApprovedFinalCount.reportingUnitTypeId

```

12|import org.apache.commons.lang3.builder.ToStringBuilder;
13|
14|public abstract class CastBallotRefForApprovedFinalCount implements Serializable {
>15| private final ReportingUnitTypeId reportingUnitTypeId;
16| private final CountContext countContext;
17| private final AreaPath countingAreaPath;
18| private final CastBallotId castBallotId;

```

**Sink :****File** admin-common/src/main/java/no/valg/eva/admin/common/counting/model/CastBallotRefForApprovedFinalCount.java 15**Fact Name:** no.valg.eva.admin.common.counting.model.CastBallotRefForApprovedFinalCount.reportingUnitTypeId

```

12|import org.apache.commons.lang3.builder.ToStringBuilder;
13|
14|public abstract class CastBallotRefForApprovedFinalCount implements Serializable {
>15| private final ReportingUnitTypeId reportingUnitTypeId;
16| private final CountContext countContext;
17| private final AreaPath countingAreaPath;
18| private final CastBallotId castBallotId;

```

**Issue 50E20DC54F3B3DF64CoC79C3CBFE3FD8:****Source :**

File admin-common/src/main/java/no/valg/eva/admin/common/counting/model/CastBallotRefForApprovedFinalCount.java 17

Fact Name: no.valg.eva.admin.common.counting.model.CastBallotRefForApprovedFinalCount.countingAreaPath

```

14|public abstract class CastBallotRefForApprovedFinalCount implements Serializable {
15|    private final ReportingUnitTypeId reportingUnitTypeId;
16|    private final CountContext countContext;
>17|    private final AreaPath countingAreaPath;
18|    private final CastBallotId castBallotId;
19|
20|    public CastBallotRefForApprovedFinalCount(

```

**Sink :**

File admin-common/src/main/java/no/valg/eva/admin/common/counting/model/CastBallotRefForApprovedFinalCount.java 17

Fact Name: no.valg.eva.admin.common.counting.model.CastBallotRefForApprovedFinalCount.countingAreaPath

```

14|public abstract class CastBallotRefForApprovedFinalCount implements Serializable {
15|    private final ReportingUnitTypeId reportingUnitTypeId;
16|    private final CountContext countContext;
>17|    private final AreaPath countingAreaPath;
18|    private final CastBallotId castBallotId;
19|
20|    public CastBallotRefForApprovedFinalCount(

```

**Issue 7EF2B94B6E6D823D6DD732063E82EEC6:****Source :**

File admin-common/src/main/java/no/valg/eva/admin/common/counting/model/CastBallotRefForApprovedFinalCount.java 16

Fact Name: no.valg.eva.admin.common.counting.model.CastBallotRefForApprovedFinalCount.countContext

```

14|public abstract class CastBallotRefForApprovedFinalCount implements Serializable {
15|    private final ReportingUnitTypeId reportingUnitTypeId;
>16|    private final CountContext countContext;
17|    private final AreaPath countingAreaPath;
18|    private final CastBallotId castBallotId;

```

**Sink :**

File admin-common/src/main/java/no/valg/eva/admin/common/counting/model/CastBallotRefForApprovedFinalCount.java 16

Fact Name: no.valg.eva.admin.common.counting.model.CastBallotRefForApprovedFinalCount.countContext

```

14|public abstract class CastBallotRefForApprovedFinalCount implements Serializable {
15|    private final ReportingUnitTypeId reportingUnitTypeId;
>16|    private final CountContext countContext;
17|    private final AreaPath countingAreaPath;
18|    private final CastBallotId castBallotId;

```

**Issue C52276FAB5C74E1818F671E8645D4381:****Source :**

File admin-common/src/main/java/no/valg/eva/admin/common/counting/model/CastBallotRefForApprovedFinalCount.java 18

Fact Name: no.valg.eva.admin.common.counting.model.CastBallotRefForApprovedFinalCount.castBallotId

```

15|private final ReportingUnitTypeId reportingUnitTypeId;
16|private final CountContext countContext;
17|private final AreaPath countingAreaPath;
>18|private final CastBallotId castBallotId;
19|
20|public CastBallotRefForApprovedFinalCount(
21|    ReportingUnitTypeId reportingUnitTypeId, CountContext countContext, AreaPath countingAreaPath,
    CastBallotId castBallotId) {

```

**Sink :**

File admin-common/src/main/java/no/valg/eva/admin/common/counting/model/CastBallotRefForApprovedFinalCount.java 18

Fact Name: no.valg.eva.admin.common.counting.model.CastBallotRefForApprovedFinalCount.castBallotId

```

15|private final ReportingUnitTypeId reportingUnitTypeId;
16|private final CountContext countContext;
17|private final AreaPath countingAreaPath;
>18|private final CastBallotId castBallotId;
19|
20|public CastBallotRefForApprovedFinalCount(
21|    ReportingUnitTypeId reportingUnitTypeId, CountContext countContext, AreaPath countingAreaPath,
    CastBallotId castBallotId) {

```

**6.328 ContestStatus.java****Issue 270265E374CA0A5F6003BDE7500AA000:****Source :**

File admin-common/src/main/java/no/valg/eva/admin/common/configuration/status/ContestStatus.java 17

Fact Name: no.valg.eva.admin.common.configuration.status.ContestStatus.id

```

>17|private int id;
18|
19|ContestStatus(int id) {
20|    this.id = id;

```

**Sink :****File** admin-common/src/main/java/no/valg/eva/admin/common/configuration/status/ContestStatus.java 17**Fact Name:** no.valg.eva.admin.common.configuration.status.ContestStatus.id

```

>17|private int id;
18|
19|ContestStatus(int id) {
20|    this.id = id;

```

## 6.329 CountingOverviewController.java

**Issue** 1113964EBBBADBF9E2FoFF52E7E82991:**Source :****File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/counting/ctrls/CountingOverviewController.java 67**Fact Name:** no.valg.eva.admin.frontend.counting.ctrls.CountingOverviewController.pickerAreaPath

```

65|private FromRequest fromRequest;
66|private Kontekst data;
>67|private AreaPath pickerAreaPath;
68|private List<CountingOverviewTabModel> tabs;
69|private int activeTabIndex;
70|private List<CountingOverviewPanelModel> panels;

```

**Sink :****File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/counting/ctrls/CountingOverviewController.java 67**Fact Name:** no.valg.eva.admin.frontend.counting.ctrls.CountingOverviewController.pickerAreaPath

```

65|private FromRequest fromRequest;
66|private Kontekst data;
>67|private AreaPath pickerAreaPath;
68|private List<CountingOverviewTabModel> tabs;
69|private int activeTabIndex;
70|private List<CountingOverviewPanelModel> panels;

```

**Issue** 12E2C7E6FFCF69105CoBFBE04B64C726:**Source :****File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/counting/ctrls/CountingOverviewController.java 71**Fact Name:** no.valg.eva.admin.frontend.counting.ctrls.CountingOverviewController.activePanelIndeces

```

68|private List<CountingOverviewTabModel> tabs;
69|private int activeTabIndex;
70|private List<CountingOverviewPanelModel> panels;
>71|private String activePanelIndeces;
72|private AreaPath areaPath;
73|private String tekstForRapporteringsknapp;
74|private Boolean altErRapportert;

```

**Sink :****File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/counting/ctrls/CountingOverviewController.java 71**Fact Name:** no.valg.eva.admin.frontend.counting.ctrls.CountingOverviewController.activePanelIndeces

```

68|private List<CountingOverviewTabModel> tabs;
69|private int activeTabIndex;
70|private List<CountingOverviewPanelModel> panels;
>71|private String activePanelIndeces;
72|private AreaPath areaPath;
73|private String tekstForRapporteringsknapp;
74|private Boolean altErRapportert;

```

**Issue** 4037A2F31253AE5DE0E130FC4ED1AF44:**Source :****File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/counting/ctrls/CountingOverviewController.java 65 **Fact****Name:** no.valg.eva.admin.frontend.counting.ctrls.CountingOverviewController.fromRequest

```

62|@Inject
63|protected MessageProvider messageProvider;
64|
>65|private FromRequest fromRequest; 66|private Kontekst data;
67|private AreaPath pickerAreaPath;
68|private List<CountingOverviewTabModel> tabs;

```

**Sink :****File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/counting/ctrls/CountingOverviewController.java 65 **Fact****Name:** no.valg.eva.admin.frontend.counting.ctrls.CountingOverviewController.fromRequest

```

62|@Inject
63|protected MessageProvider messageProvider;
64|
>65|private FromRequest fromRequest;
66|private Kontekst data;

```

```
67|private AreaPath pickerAreaPath;
68|private List<CountingOverviewTabModel> tabs;
```

**Issue 6B6772535D2BFBF3EDE6BE89656A3C99:****Source :**

**File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/counting/ctrls/CountingOverviewController.java 70

**Fact Name:** no.valg.eva.admin.frontend.counting.ctrls.CountingOverviewController.panels

```
67|private AreaPath pickerAreaPath;
68|private List<CountingOverviewTabModel> tabs;
69|private int activeTabIndex;
>70|private List<CountingOverviewPanelModel> panels;
71|private String activePanelIndeces;
72|private AreaPath areaPath;
73|private String tekstForRapporteringsknapp;
```

**Sink :**

**File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/counting/ctrls/CountingOverviewController.java 70

**Fact Name:** no.valg.eva.admin.frontend.counting.ctrls.CountingOverviewController.panels

```
67|private AreaPath pickerAreaPath;
68|private List<CountingOverviewTabModel> tabs;
69|private int activeTabIndex;
>70|private List<CountingOverviewPanelModel> panels;
71|private String activePanelIndeces;
72|private AreaPath areaPath;
73|private String tekstForRapporteringsknapp;
```

**Issue C7E8123788BC54DE724BABC7B6512E63:****Source :**

**File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/counting/ctrls/CountingOverviewController.java 75

**Fact Name:** no.valg.eva.admin.frontend.counting.ctrls.CountingOverviewController.antallRapporterbare

```
72|private AreaPath areaPath;
73|private String tekstForRapporteringsknapp;
74|private Boolean altErRapportert;
>75|private Long antallRapporterbare;
76|private Boolean ingenTellingForDetteOmraadet;
77|
78|@Override
```

**Sink :**

**File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/counting/ctrls/CountingOverviewController.java 75

**Fact Name:** no.valg.eva.admin.frontend.counting.ctrls.CountingOverviewController.antallRapporterbare

```
72|private AreaPath areaPath;
73|private String tekstForRapporteringsknapp;
74|private Boolean altErRapportert;
>75|private Long antallRapporterbare;
76|private Boolean ingenTellingForDetteOmraadet;
77|
78|@Override
```

**Issue D2324372D1EE435B487F744969E7876F:****Source :**

**File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/counting/ctrls/CountingOverviewController.java 68

**Fact Name:** no.valg.eva.admin.frontend.counting.ctrls.CountingOverviewController.tabs

```
65|private FromRequest fromRequest;
66|private Kontekst data;
67|private AreaPath pickerAreaPath;
>68|private List<CountingOverviewTabModel> tabs;
69|private int activeTabIndex;
70|private List<CountingOverviewPanelModel> panels;
71|private String activePanelIndeces;
```

**Sink :**

**File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/counting/ctrls/CountingOverviewController.java 68

**Fact Name:** no.valg.eva.admin.frontend.counting.ctrls.CountingOverviewController.tabs

```
65|private FromRequest fromRequest;
66|private Kontekst data;
67|private AreaPath pickerAreaPath;
>68|private List<CountingOverviewTabModel> tabs;
69|private int activeTabIndex;
70|private List<CountingOverviewPanelModel> panels;
71|private String activePanelIndeces;
```

**Issue DE88AA6170BF405869AF0FC7D319A4DB:****Source :**

**File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/counting/ctrls/CountingOverviewController.java 69

**Fact** Name: no.valg.eva.admin.frontend.counting.ctrls.CountingOverviewController.activeTabIndex

```
66|private Kontekst data;  
67|private AreaPath pickerAreaPath;  
68|private List<CountingOverviewTabModel> tabs;  
>69|private int activeTabIndex;  
70|private List<CountingOverviewPanelModel> panels;  
71|private String activePanelIndeces;  
72|private AreaPath areaPath;
```

**Sink :**

**File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/counting/ctrls/CountingOverviewController.java 69

**Fact** Name: no.valg.eva.admin.frontend.counting.ctrls.CountingOverviewController.activeTabIndex

```
66|private Kontekst data;  
67|private AreaPath pickerAreaPath;  
68|private List<CountingOverviewTabModel> tabs;  
>69|private int activeTabIndex;  
70|private List<CountingOverviewPanelModel> panels;  
71|private String activePanelIndeces;  
72|private AreaPath areaPath;
```

## 6.330 CountyStatusEnum.java

**Issue** 3766026035A73E7936CE75F4CCF5FA68:

**Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/common/configuration/status/CountyStatusEnum.java 19

**Fact** Name: no.valg.eva.admin.common.configuration.status.CountyStatusEnum.id

```
>19|private int id;  
20|  
21|CountyStatusEnum(int id) {  
22|    this.id = id;
```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/common/configuration/status/CountyStatusEnum.java 19

**Fact** Name: no.valg.eva.admin.common.configuration.status.CountyStatusEnum.id

```
>19|private int id;  
20|  
21|CountyStatusEnum(int id) {  
22|    this.id = id;
```

## 6.331 ElectionEventStatusEnum.java

### Issue F56579039FD93EBEEA0BBFB35FC7D626:

**Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/common/configuration/status/ElectionEventStatusEnum.java 21 **Fact**

```
>21|private int id;
22|
23|private ElectionEventStatusEnum(int id) {
24|     this.id = id;
```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/common/configuration/status/ElectionEventStatusEnum.java 21 **Fact**

**Name:** no.valg.eva.admin.common.configuration.status.ElectionEventStatusEnum.id

```
>21|private int id;
22|
23|private ElectionEventStatusEnum(int id) {
24|     this.id = id;
```

## 6.332 ElectionPath.java

### Issue CD40A857FA0978893A6396A7AEAB9240:

**Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/common/ElectionPath.java 31

**Fact Name:** no.valg.eva.admin.common.ElectionPath.path

```
28|public static final int GROUP_NO_ELECTION = 5;
29|public static final int GROUP_NO_CONTEST = 7;
30|private static final Pattern ELECTION_PATH_PATTERN = Pattern.compile(ELECTION_PATH_REGEX);
>31|private final String path;
32|private final ElectionLevelEnum level;
33|private final String electionEventId;
34|private final String electionGroupId;
```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/common/ElectionPath.java 31

**Fact Name:** no.valg.eva.admin.common.ElectionPath.path

```
28|public static final int GROUP_NO_ELECTION = 5;
29|public static final int GROUP_NO_CONTEST = 7;
30|private static final Pattern ELECTION_PATH_PATTERN = Pattern.compile(ELECTION_PATH_REGEX);
>31|private final String path;
32|private final ElectionLevelEnum level;
33|private final String electionEventId;
34|private final String electionGroupId;
```

## 6.333 ImportOperatorMessage.java

### Issue DEBE3DCF123B838681723E522661D174:

**Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/common/rbac/ImportOperatorMessage.java 7

**Fact Name:** no.valg.eva.admin.common.rbac.ImportOperatorMessage.line

```
5|public class ImportOperatorMessage extends UserMessage {
6|
> 7|     private final int line;
8|
9|     public ImportOperatorMessage(int line, String message, Object... args) {
10|         super(message, args);
```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/common/rbac/ImportOperatorMessage.java 7

**Fact Name:** no.valg.eva.admin.common.rbac.ImportOperatorMessage.line



```

5|public class ImportOperatorMessage extends UserMessage {
6|
> 7|   private final int line;
8|
9|   public ImportOperatorMessage(int line, String message, Object... args) {
10|       super(message, args);

```

## 6.334 LevelingSeatFactory.java

**Issue** 41E0063A1FBB1BA6583B66CE438E5CCB:

**Source :**

**File** admin-settlement/src/main/java/no/valg/eva/admin/settlement/domain/model/factory/LevelingSeatFactory.java 38

**Fact Name:** no.valg.eva.admin.settlement.domain.model.factory.LevelingSeatFactory.levelingSeatSummaryMap

```

35|private final Map<CandidateSeatsKey, SortedSet<CandidateSeat>> candidateSeatsMap = new HashMap<>();
36|private final List<LevelingSeatQuotient> levelingSeatQuotients = new ArrayList<>();
37|private ElectionSettlement electionSettlement;
>38|private Map<Party, LevelingSeatSummary> levelingSeatSummaryMap;
39|private Set<Contest> leveledContests = new HashSet<>();
40|private Map<Party, Integer> partySeatsMap = new HashMap<>();
41|private int lastRankNumber;

```

**Sink :**

**File** admin-settlement/src/main/java/no/valg/eva/admin/settlement/domain/model/factory/LevelingSeatFactory.java 38

**Fact Name:** no.valg.eva.admin.settlement.domain.model.factory.LevelingSeatFactory.levelingSeatSummaryMap

```

35|private final Map<CandidateSeatsKey, SortedSet<CandidateSeat>> candidateSeatsMap = new HashMap<>();
36|private final List<LevelingSeatQuotient> levelingSeatQuotients = new ArrayList<>();
37|private ElectionSettlement electionSettlement;
>38|private Map<Party, LevelingSeatSummary> levelingSeatSummaryMap;
39|private Set<Contest> leveledContests = new HashSet<>();
40|private Map<Party, Integer> partySeatsMap = new HashMap<>();
41|private int lastRankNumber;

```

## 6.335 MunicipalityStatusEnum.java

**Issue** E4AF278FC95528D7282DC7BFD4CB1E88:

**Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/common/configuration/status/MunicipalityStatusEnum.java 19

**Fact Name:** no.valg.eva.admin.common.configuration.status.MunicipalityStatusEnum.id

```

>19|private int id;
20|
21|MunicipalityStatusEnum(int id) {
22|   this.id = id;

```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/common/configuration/status/MunicipalityStatusEnum.java 19

**Fact Name:** no.valg.eva.admin.common.configuration.status.MunicipalityStatusEnum.id

```

>19|private int id;
20|
21|MunicipalityStatusEnum(int id) {
22|   this.id = id;

```

## 6.336 MvArea.java

**Issue** 9631C40821F810C497BA3290BC90975E:

**Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/MvArea.java 319 **Fact**

**Name:** no.valg.eva.admin.configuration.domain.model.MvArea.areaPath

```

316|@Setter private Municipality municipality; 317|@Setter private Borough borough;
318|@Setter private PollingPlace pollingPlace;
>319|@Setter private String areaPath;
320|@Setter private int areaLevel;
321|@Setter private String electionEventId;
322|@Setter private String countryId;

```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/MvArea.java 319 **Fact**  
**Name:** no.valg.eva.admin.configuration.domain.model.MvArea.areaPath

```
316|@Setter private Municipality municipality; 317|@Setter private Borough borough;
318|@Setter private PollingPlace pollingPlace;
>319|@Setter private String areaPath;
320|@Setter private int areaLevel;
321|@Setter private String electionEventId;
322|@Setter private String countryId;
```

**6.337 MvAreaDigest.java****Issue** E29F73B071CF1E1F2C6F4AF0EE65BB0C:**Source :**

**File** admin-configuration/src/main/java/no/valg/eva/admin/configuration/domain/model/MvAreaDigest.java 61 **Fact**  
**Name:** no.valg.eva.admin.configuration.domain.model.MvAreaDigest.areaPath

```
58|         resultClass = MvAreaDigest.class) })
59|public class MvAreaDigest extends BaseEntity {
60|
>61|    private String areaPath;
62|    private int areaLevel;
63|    private String electionEventName;
64|    private String countryName;
```

**Sink :**

**File** admin-configuration/src/main/java/no/valg/eva/admin/configuration/domain/model/MvAreaDigest.java 61 **Fact**  
**Name:** no.valg.eva.admin.configuration.domain.model.MvAreaDigest.areaPath

```
58|         resultClass = MvAreaDigest.class) })
59|public class MvAreaDigest extends BaseEntity {
60|
>61|    private String areaPath;
62|    private int areaLevel;
63|    private String electionEventName;
64|    private String countryName;
```

**6.338 MvElection.java****Issue** 98DBA2BD44D9ED3DFB3FAB410743C615:**Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/MvElection.java 159  
**Fact** **Name:** no.valg.eva.admin.configuration.domain.model.MvElection.electionPath

```
156|private Contest contest;
157|private ElectionGroup electionGroup;
158|private Election election;
>159|private String electionPath;
160|private int electionLevel;
161|private String electionEventId;
162|private String electionGroupId;
```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/configuration/domain/model/MvElection.java 159  
**Fact** **Name:** no.valg.eva.admin.configuration.domain.model.MvElection.electionPath

```
156|private Contest contest;
157|private ElectionGroup electionGroup;
158|private Election election;
>159|private String electionPath;
160|private int electionLevel;
161|private String electionEventId;
162|private String electionGroupId;
```

**6.339 MvElectionDigest.java****Issue** 3E3E5F0B081921B0C8799C714B430BC3:**Source :**

**File** admin-configuration/src/main/java/no/valg/eva/admin/configuration/domain/model/MvElectionDigest.java 74 **Fact**  
Name: no.valg.eva.admin.configuration.domain.model.MvElectionDigest.electionPath

```
71|         resultClass = MvElectionDigest.class)
```

```
72|}}  
73|public class MvElectionDigest extends BaseEntity {  
>74|     private String electionPath;  
75|     private int electionLevel;  
76|     private String electionEventName;  
77|     private String electionGroupName;
```

**Sink :**

**File** admin-configuration/src/main/java/no/valg/eva/admin/configuration/domain/model/MvElectionDigest.java 74 **Fact**  
Name: no.valg.eva.admin.configuration.domain.model.MvElectionDigest.electionPath

```
71|         resultClass = MvElectionDigest.class)  
72|}}  
73|public class MvElectionDigest extends BaseEntity {
```

```

>74| private String electionPath;
75| private int electionLevel;
76| private String electionEventName;
77| private String electionGroupName;

```

## 6.340 RejectedCastBallotRefForApprovedFinalCount.java

**Issue** 2ABEEB7CCA5CBE9FE1923DCB5C06149C:

**Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/common/counting/model/RejectedCastBallotRefForApprovedFinalCount.java 13

**Fact** Name: no.valg.eva.admin.common.counting.model.RejectedCastBallotRefForApprovedFinalCount.ballotRejectionId

```

10| import org.apache.commons.lang3.builder.ToStringBuilder;
11|
12| public final class RejectedCastBallotRefForApprovedFinalCount extends CastBallotRefForApprovedFinalCount {
>13|     private final BallotRejectionId ballotRejectionId;
14|
15|     public RejectedCastBallotRefForApprovedFinalCount(
16|         ReportingUnitTypeId reportingUnitTypeId, CountContext countContext, AreaPath countingAreaPath,

```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/common/counting/model/RejectedCastBallotRefForApprovedFinalCount.java 13

**Fact** Name: no.valg.eva.admin.common.counting.model.RejectedCastBallotRefForApprovedFinalCount.ballotRejectionId

```

10| import org.apache.commons.lang3.builder.ToStringBuilder;
11|
12| public final class RejectedCastBallotRefForApprovedFinalCount extends CastBallotRefForApprovedFinalCount { >13|
private final BallotRejectionId ballotRejectionId;
14|
15|     public RejectedCastBallotRefForApprovedFinalCount(
16|         ReportingUnitTypeId reportingUnitTypeId, CountContext countContext, AreaPath countingAreaPath,

```

## 6.341 ResponsibleOfficer.java

**Issue** 28243D9A73B1A63F155E90FED48F5B54:

**Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/common/configuration/model/local/ResponsibleOfficer.java 18 **Fact**

**Name:** no.valg.eva.admin.common.configuration.model.local.ResponsibleOfficer.displayOrder

```

15| private Long pk;
16| private AreaPath areaPath;
17| private ResponsibilityId responsibilityId;
>18| private int displayOrder;
19| private String firstName;
20| private String middleName;
21| private String lastName;

```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/common/configuration/model/local/ResponsibleOfficer.java 18 **Fact**

**Name:** no.valg.eva.admin.common.configuration.model.local.ResponsibleOfficer.displayOrder

```

15| private Long pk;
16| private AreaPath areaPath;
17| private ResponsibilityId responsibilityId;
>18| private int displayOrder;
19| private String firstName;
20| private String middleName;
21| private String lastName;

```

## 6.342 SertifikatOgNøkkel.java

**Issue** 415B8D08C40BA151186D589130C7489E:

**Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/crypto/SertifikatOgNøkkel.java 15 **Fact**

**Name:** no.valg.eva.admin.crypto.SertifikatOgNøkkel.sertifikatkjede

```

12| */
13| public class SertifikatOgNøkkel {
14|     private final X509Certificate sertifikat;
>15|     private final Certificate[] sertifikatkjede;
16|     private final PrivateKey nøkkel; 17|
18|     SertifikatOgNøkkel(X509Certificate sertifikat, Certificate[] sertifikatkjede, PrivateKey nøkkel) {

```

**Sink :**

File admin-common/src/main/java/no/valg/eva/admin/crypto/SertifikatOgNøkkel.java 15 **Fact**  
 Name: no.valg.eva.admin.crypto.SertifikatOgNøkkel.sertifikatkjede

```
12| */
13|public class SertifikatOgNøkkel {
14|   private final X509Certificate sertifikat;
>15|   private final Certificate[] sertifikatkjede;
16|   private final PrivateKey nøkkel; 17|
18|   SertifikatOgNøkkel(X509Certificate sertifikat, Certificate[] sertifikatkjede, PrivateKey nøkkel) {
```

### Issue 6920BB8214D49FAEEF8A21E8E0155459:

#### Source :

File admin-common/src/main/java/no/valg/eva/admin/crypto/SertifikatOgNøkkel.java 16 **Fact**  
 Name: no.valg.eva.admin.crypto.SertifikatOgNøkkel.nøkkel

```
13|public class SertifikatOgNøkkel {
14|   private final X509Certificate sertifikat;
15|   private final Certificate[] sertifikatkjede;
>16|   private final PrivateKey nøkkel; 17|
18|   SertifikatOgNøkkel(X509Certificate sertifikat, Certificate[] sertifikatkjede, PrivateKey nøkkel) { 19|
this.sertifikatkjede = sertifikatkjede;
```

#### Sink :

File admin-common/src/main/java/no/valg/eva/admin/crypto/SertifikatOgNøkkel.java 16 **Fact**  
 Name: no.valg.eva.admin.crypto.SertifikatOgNøkkel.nøkkel

```
13|public class SertifikatOgNøkkel {
14|   private final X509Certificate sertifikat;
15|   private final Certificate[] sertifikatkjede;
>16|   private final PrivateKey nøkkel;
17|
18|   SertifikatOgNøkkel(X509Certificate sertifikat, Certificate[] sertifikatkjede, PrivateKey nøkkel) { 19|
this.sertifikatkjede = sertifikatkjede;
```

### Issue B855688599BCC5F7380178C114ED7120:

#### Source :

File admin-common/src/main/java/no/valg/eva/admin/crypto/SertifikatOgNøkkel.java 14 **Fact**  
 Name: no.valg.eva.admin.crypto.SertifikatOgNøkkel.sertifikat

```
11| * Privat nøkkel må behandles som en hemmelighet.
12| */
13|public class SertifikatOgNøkkel {
>14|   private final X509Certificate sertifikat;
15|   private final Certificate[] sertifikatkjede;
16|   private final PrivateKey nøkkel;
```

#### Sink :

File admin-common/src/main/java/no/valg/eva/admin/crypto/SertifikatOgNøkkel.java 14 **Fact**  
 Name: no.valg.eva.admin.crypto.SertifikatOgNøkkel.sertifikat

```
11| * Privat nøkkel må behandles som en hemmelighet.
12| */
13|public class SertifikatOgNøkkel {
>14|   private final X509Certificate sertifikat;
15|   private final Certificate[] sertifikatkjede;
16|   private final PrivateKey nøkkel;
```

## 6.343 SimpleAuditEvent.java

### Issue 926EB132E251935FFDDAE6831354A752:

#### Source :

File admin-common/src/main/java/no/valg/eva/admin/common/auditlog/auditevents/SimpleAuditEvent.java 31 **Fact**  
 Name: no.valg.eva.admin.common.auditlog.auditevents.SimpleAuditEvent.objectType

```
28| */
29|public class SimpleAuditEvent extends AuditEvent {
30|
>31|   private Class objectType;
32|   private Map<String, Optional<String>> jsonProperties; 33|
34|   public SimpleAuditEvent(String uid, String electionEventId, String roleId, AreaPath roleAreaPath,
```

#### Sink :

File admin-common/src/main/java/no/valg/eva/admin/common/auditlog/auditevents/SimpleAuditEvent.java 31 **Fact**  
 Name: no.valg.eva.admin.common.auditlog.auditevents.SimpleAuditEvent.objectType

```
28| */
29|public class SimpleAuditEvent extends AuditEvent {
30|
>31|   private Class objectType;
32|   private Map<String, Optional<String>> jsonProperties;
```

```
33|
34| public SimpleAuditEvent(String uid, String electionEventId, String roleId, AreaPath roleAreaPath,
```

## 6.344 SimpleAuditEventType.java

### Issue C64B9E2268D3BA4B923F9E051E06370E:

#### Source :

File admin-common/src/main/java/no/valg/eva/admin/common/auditlog/SimpleAuditEventType.java 16

Fact Name: no.valg.eva.admin.common.auditlog.SimpleAuditEventType.process

```
13|SubConfigEvent(Process.CENTRAL_CONFIGURATION, setOf(Outcome.Success)),
14|AccessDeniedInBackend(Process.AUTHORIZATION, setOf(Outcome.GenericError));
15|
>16|private final Process process;
17|private final Set<Outcome> validOutcomes;
18|
19|SimpleAuditEventType(Process process, Set<Outcome> validOutcomes) {
```

#### Sink :

File admin-common/src/main/java/no/valg/eva/admin/common/auditlog/SimpleAuditEventType.java 16

Fact Name: no.valg.eva.admin.common.auditlog.SimpleAuditEventType.process

```
13|SubConfigEvent(Process.CENTRAL_CONFIGURATION, setOf(Outcome.Success)),
14|AccessDeniedInBackend(Process.AUTHORIZATION, setOf(Outcome.GenericError));
15|
>16|private final Process process;
17|private final Set<Outcome> validOutcomes;
18|
19|SimpleAuditEventType(Process process, Set<Outcome> validOutcomes) {
```

## 6.345 SkdVoterFileParser.java

### Issue 3B8086B0591C2A847BC203CD34A75B3A:

#### Source :

File admin-voting/src/main/java/no/valg/eva/admin/voting/domain/electoratroll/SkdVoterFileParser.java 19

Fact Name: no.valg.eva.admin.voting.domain.electoratroll.SkdVoterFileParser.header

```
16|private static final Logger LOGGER = Logger.getLogger(SkdVoterFileParser.class);
17|public static final int RECORD_SIZE = 509;
18|
>19|private final String header;
20|private final BufferedReader reader;
21|private int rowNum = 0;
```

#### Sink :

File admin-voting/src/main/java/no/valg/eva/admin/voting/domain/electoratroll/SkdVoterFileParser.java 19

Fact Name: no.valg.eva.admin.voting.domain.electoratroll.SkdVoterFileParser.header

```
16|private static final Logger LOGGER = Logger.getLogger(SkdVoterFileParser.class);
17|public static final int RECORD_SIZE = 509;
18|
>19|private final String header;
20|private final BufferedReader reader;
21|private int rowNum = 0;
```

## 6.346 SkdVoterRecord.java

### Issue 323D9DC933C66C8B558B5CA8DD1DAD59:

#### Source :

File admin-voting/src/main/java/no/valg/eva/admin/voting/domain/electoratroll/SkdVoterRecord.java 27

Fact Name: no.valg.eva.admin.voting.domain.electoratroll.SkdVoterRecord.kjoren

```
24|private static final String SKD_TIMESTAMP_FORMAT = "yyyy-MM-dd-HH.mm.ss.SSS"; 25|
26|private String record; >27|private String kjoren;
28|
29|SkdVoterRecord(String record, String kjoren) {
30|    this.record = record;
```

#### Sink :

File admin-voting/src/main/java/no/valg/eva/admin/voting/domain/electoratroll/SkdVoterRecord.java 27

Fact Name: no.valg.eva.admin.voting.domain.electoratroll.SkdVoterRecord.kjoren

```
24|private static final String SKD_TIMESTAMP_FORMAT = "yyyy-MM-dd-HH.mm.ss.SSS"; 25|
26|private String record;
>27|private String kjoren;
28|
29|SkdVoterRecord(String record, String kjoren) {
```

```
30| this.record = record;
```

## 6.347 Stemmekrets.java

### Issue 3290B5CD19344B50D836EAD2D05CoEEF:

**Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/felles/valggeografi/model/StemmeKrets.java 9

**Fact** Name: no.valg.eva.admin.felles.valggeografi.model.StemmeKrets.bydelNavn

```
6|private final boolean kommuneStemmeKrets;
7|private final String fylkeskommuneNavn;
8|private final String kommuneNavn;
> 9|private final String bydelNavn; 10|
11|public StemmeKrets(StemmeKretsSti sti, String navn, boolean kommuneStemmeKrets, String fylkeskommuneNavn, String
kommuneNavn, String bydelNavn) {
12|    super(sti, navn);
```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/felles/valggeografi/model/StemmeKrets.java 9

**Fact** Name: no.valg.eva.admin.felles.valggeografi.model.StemmeKrets.bydelNavn

```
6|private final boolean kommuneStemmeKrets;
7|private final String fylkeskommuneNavn;
8|private final String kommuneNavn;
> 9|private final String bydelNavn; 10|
11|public StemmeKrets(StemmeKretsSti sti, String navn, boolean kommuneStemmeKrets, String fylkeskommuneNavn, String
kommuneNavn, String bydelNavn) {
12|    super(sti, navn);
```

### Issue 61BoF850A511FA34F3AA4798C3F41A39:

**Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/felles/valggeografi/model/StemmeKrets.java 8

**Fact** Name: no.valg.eva.admin.felles.valggeografi.model.StemmeKrets.kommuneNavn

```
5|public class StemmeKrets extends Valggeografi<StemmeKretsSti> {
6|    private final boolean kommuneStemmeKrets;
7|    private final String fylkeskommuneNavn;
> 8|    private final String kommuneNavn;
9|    private final String bydelNavn; 10|
11|    public StemmeKrets(StemmeKretsSti sti, String navn, boolean kommuneStemmeKrets, String fylkeskommuneNavn,
String kommuneNavn, String bydelNavn) {
```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/felles/valggeografi/model/StemmeKrets.java 8

**Fact** Name: no.valg.eva.admin.felles.valggeografi.model.StemmeKrets.kommuneNavn

```
5|public class StemmeKrets extends Valggeografi<StemmeKretsSti> {
6|    private final boolean kommuneStemmeKrets;
7|    private final String fylkeskommuneNavn;
> 8|    private final String kommuneNavn;
9|    private final String bydelNavn; 10|
11|    public StemmeKrets(StemmeKretsSti sti, String navn, boolean kommuneStemmeKrets, String fylkeskommuneNavn,
String kommuneNavn, String bydelNavn) {
```

### Issue 988C5E74CF75468070D322B6CA009739:

**Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/felles/valggeografi/model/StemmeKrets.java 7

**Fact** Name: no.valg.eva.admin.felles.valggeografi.model.StemmeKrets.fylkeskommuneNavn

```
5|public class StemmeKrets extends Valggeografi<StemmeKretsSti> {
6|    private final boolean kommuneStemmeKrets;
>7|    private final String fylkeskommuneNavn;
8|    private final String kommuneNavn;
9|    private final String bydelNavn;
```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/felles/valggeografi/model/StemmeKrets.java 7

**Fact** Name: no.valg.eva.admin.felles.valggeografi.model.StemmeKrets.fylkeskommuneNavn

```
5|public class StemmeKrets extends Valggeografi<StemmeKretsSti> {
6|    private final boolean kommuneStemmeKrets;
>7|    private final String fylkeskommuneNavn;
8|    private final String kommuneNavn; 9|    private final String bydelNavn;
```

## 6.348 Sti.java

### Issue 73B03ACFD7ADC11B1BF2EBDC6F9CEACE:

**Source :****File** admin-common/src/main/java/no/valg/eva/admin/felles/sti/Sti.java 15**Fact Name:** no.valg.eva.admin.felles.sti.Sti.forelderSti

```

12|private static final String STI_FORMAT = "%s.%s";
13|
14|private Pattern patternForValideringAvId;
>15|private F forelderSti;
16|private String sisteId;
17|
18|protected Sti(Pattern patternForValideringAvId, F forelderSti, String sisteId) {

```

**Sink :****File** admin-common/src/main/java/no/valg/eva/admin/felles/sti/Sti.java 15**Fact Name:** no.valg.eva.admin.felles.sti.Sti.forelderSti

```

12|private static final String STI_FORMAT = "%s.%s";
13|
14|private Pattern patternForValideringAvId;
>15|private F forelderSti;
16|private String sisteId;
17|
18|protected Sti(Pattern patternForValideringAvId, F forelderSti, String sisteId) {

```

### Issue F226E7BC4A33037A2EAE3D1679E9B9D9:

**Source :****File** admin-common/src/main/java/no/valg/eva/admin/felles/sti/Sti.java 16**Fact Name:** no.valg.eva.admin.felles.sti.Sti.sisteId

```

14|private Pattern patternForValideringAvId;
15|private F forelderSti;
>16|private String sisteId;
17|
18|protected Sti(Pattern patternForValideringAvId, F forelderSti, String sisteId) {
19|    this.patternForValideringAvId = patternForValideringAvId;

```

**Sink :****File** admin-common/src/main/java/no/valg/eva/admin/felles/sti/Sti.java 16**Fact Name:** no.valg.eva.admin.felles.sti.Sti.sisteId

```

14|private Pattern patternForValideringAvId;
15|private F forelderSti;
>16|private String sisteId;
17|
18|protected Sti(Pattern patternForValideringAvId, F forelderSti, String sisteId) {
19|    this.patternForValideringAvId = patternForValideringAvId;

```

## 6.349 Styretype.java

### Issue 2183710203BBFDC4C2E071831A9F28B5:

**Source :****File** admin-common/src/main/java/no/valg/eva/admin/felles/konfigurasjon/model/Styretype.java 27**Fact Name:** no.valg.eva.admin.felles.konfigurasjon.model.Styretype.navn

```

24|STEMMESTYRET(6, VALGHENDELSE, STEMMEKRETS);
25|
26|private final int id;
>27|private final String navn;
28|private final ValghierarkiNivaa valghierarkiNivaa;
29|private final ValggeografiNivaa valggeografiNivaa;

```

**Sink :****File** admin-common/src/main/java/no/valg/eva/admin/felles/konfigurasjon/model/Styretype.java 27**Fact Name:** no.valg.eva.admin.felles.konfigurasjon.model.Styretype.navn

```

24|STEMMESTYRET(6, VALGHENDELSE, STEMMEKRETS);
25|
26|private final int id;
>27|private final String navn;
28|private final ValghierarkiNivaa valghierarkiNivaa;
29|private final ValggeografiNivaa valggeografiNivaa;

```

### Issue 8BF9B606981738E3CF2C07B207C12F1C:

**Source :****File** admin-common/src/main/java/no/valg/eva/admin/felles/konfigurasjon/model/Styretype.java 26 **Fact****Name:** no.valg.eva.admin.felles.konfigurasjon.model.Styretype.id

```

23|BYDELSVALGSTYRET(5, VALG, BYDEL),
24|STEMMESTYRET(6, VALGHENDELSE, STEMMEKRETS);
25|
>26|private final int id;
27|private final String navn;
28|private final ValghierarkiNivaa valghierarkiNivaa;

```



```
29|private final ValggeografiNivaa valggeografiNivaa;
```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/felles/konfigurasjon/model/Styretype.java 26 **Fact**  
**Name:** no.valg.eva.admin.felles.konfigurasjon.model.Styretype.id

```
23|BYDELSVALGSTYRET(5, VALG, BYDEL),
24|STEMMESTYRET(6, VALGHENDELSE, STEMMEKRETS);
25|
>26|private final int id;
27|private final String navn;
28|private final ValghierarkiNivaa valghierarkiNivaa;
29|private final ValggeografiNivaa valggeografiNivaa;
```

**Issue C46E5FB783255C5FA284E81715Bo8C5B:****Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/felles/konfigurasjon/model/Styretype.java 29  
**Fact** Name: no.valg.eva.admin.felles.konfigurasjon.model.Styretype.valggeografiNivaa

```
26|private final int id;
27|private final String navn;
28|private final ValghierarkiNivaa valghierarkiNivaa;
>29|private final ValggeografiNivaa valggeografiNivaa;
30|
31|Styretype(int id, ValghierarkiNivaa valghierarkiNivaa, ValggeografiNivaa valggeografiNivaa) { 32|    this.id =
id;
```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/felles/konfigurasjon/model/Styretype.java 29  
**Fact** Name: no.valg.eva.admin.felles.konfigurasjon.model.Styretype.valggeografiNivaa

```
26|private final int id;
27|private final String navn;
28|private final ValghierarkiNivaa valghierarkiNivaa;
>29|private final ValggeografiNivaa valggeografiNivaa;
30|
31|Styretype(int id, ValghierarkiNivaa valghierarkiNivaa, ValggeografiNivaa valggeografiNivaa) { 32|    this.id =
id;
```

**Issue D05F6DA21D837D915BDE599A8EE56192:****Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/felles/konfigurasjon/model/Styretype.java 28  
**Fact** Name: no.valg.eva.admin.felles.konfigurasjon.model.Styretype.valghierarkiNivaa

```
26|private final int id;
27|private final String navn;
>28|private final ValghierarkiNivaa valghierarkiNivaa;
29|private final ValggeografiNivaa valggeografiNivaa;
30|
31|Styretype(int id, ValghierarkiNivaa valghierarkiNivaa, ValggeografiNivaa valggeografiNivaa) {
```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/felles/konfigurasjon/model/Styretype.java 28  
**Fact** Name: no.valg.eva.admin.felles.konfigurasjon.model.Styretype.valghierarkiNivaa

```
26|private final int id;
27|private final String navn;
>28|private final ValghierarkiNivaa valghierarkiNivaa;
29|private final ValggeografiNivaa valggeografiNivaa;
30|
31|Styretype(int id, ValghierarkiNivaa valghierarkiNivaa, ValggeografiNivaa valggeografiNivaa) {
```

**6.350 SubjectSerialNumber.java****Issue 976966D346C77D6A86004667299D9FA6:****Source :**

**File** admin-backend-common/src/main/java/no/valg/eva/admin/backend/common/application/bypass/SubjectSerialNumber.java 18  
**Fact** Name: no.valg.eva.admin.backend.common.application.bypass.SubjectSerialNumber.serialNumber

```
15|static final String SERIAL_NUMBER_IDENTIFIER = "OID.2.5.4.5";
16|
17|private String serialNumberStr;
>18|private BigInteger serialNumber;
19|
20|private SubjectSerialNumber(String serialNumberStr) {
21|    this.serialNumberStr = requireNonNull(serialNumberStr);
```

**Sink :**

**File** admin-backend-common/src/main/java/no/valg/eva/admin/backend/common/application/bypass/SubjectSerialNumber.java 18  
**Fact** Name: no.valg.eva.admin.backend.common.application.bypass.SubjectSerialNumber.serialNumber

```

15|static final String SERIAL_NUMBER_IDENTIFIER = "OID.2.5.4.5";
16|
17|private String serialNumberStr;
>18|private BigInteger serialNumber;
19|
20|private SubjectSerialNumber(String serialNumberStr) {
21|     this.serialNumberStr = requireNonNull(serialNumberStr);

```

## 6.351 Valg.java

### Issue 4AFD4FF1B3B366B5EA3295AE53BCAD95:

#### Source :

File admin-common/src/main/java/no/valg/eva/admin/felles/valghierarki/model/Valg.java 11 Fact  
Name: no.valg.eva.admin.felles.valghierarki.model.Valg.valggeografiNivaa

```

8|import org.apache.commons.lang3.builder.ToStringBuilder;
9|
10|public class Valg extends Valghierarki<ValgSti> {
>11|     private final ValggeografiNivaa valggeografiNivaa;
12|     private final boolean enkeltOmrade;
13|     private final String valggruppeNavn;

```

#### Sink :

File admin-common/src/main/java/no/valg/eva/admin/felles/valghierarki/model/Valg.java 11 Fact  
Name: no.valg.eva.admin.felles.valghierarki.model.Valg.valggeografiNivaa

```

8|import org.apache.commons.lang3.builder.ToStringBuilder;
9|
10|public class Valg extends Valghierarki<ValgSti> {
>11|     private final ValggeografiNivaa valggeografiNivaa;
12|     private final boolean enkeltOmrade;
13|     private final String valggruppeNavn;

```

### Issue 72F6FBDF83DFDF3566F1C04E773C3E30:

#### Source :

File admin-common/src/main/java/no/valg/eva/admin/felles/valghierarki/model/Valg.java 13 Fact  
Name: no.valg.eva.admin.felles.valghierarki.model.Valg.valggruppeNavn

```

10|public class Valg extends Valghierarki<ValgSti> {
11|     private final ValggeografiNivaa valggeografiNivaa;
12|     private final boolean enkeltOmrade;
>13|     private final String valggruppeNavn; 14|
15|     public Valg(ValgSti sti, String navn, ValggeografiNivaa valggeografiNivaa, boolean enkeltOmrade, String
valggruppeNavn) {
16|         super(sti, navn);

```

#### Sink :

File admin-common/src/main/java/no/valg/eva/admin/felles/valghierarki/model/Valg.java 13 Fact  
Name: no.valg.eva.admin.felles.valghierarki.model.Valg.valggruppeNavn

```

10|public class Valg extends Valghierarki<ValgSti> {
11|     private final ValggeografiNivaa valggeografiNivaa;
12|     private final boolean enkeltOmrade;
>13|     private final String valggruppeNavn; 14|
15|     public Valg(ValgSti sti, String navn, ValggeografiNivaa valggeografiNivaa, boolean enkeltOmrade, String
valggruppeNavn) {
16|         super(sti, navn);

```

## 6.352 Valgdistrikt.java

**Issue** 2103A5B6519E1099FB93C48914E3496C:

**Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/felles/valghierarki/model/Valgdistrikt.java 7

**Fact Name:** no.valg.eva.admin.felles.valghierarki.model.Valgdistrikt.valggeografiNivaa

```

4|import no.valg.eva.admin.felles.valggeografi.model.ValggeografiNivaa;
5|
6|public class Valgdistrikt extends Valghierarki<ValgdistriktSti> {
> 7|     private final ValggeografiNivaa valggeografiNivaa;
8|
9|     public Valgdistrikt(ValgdistriktSti sti, String navn, ValggeografiNivaa valggeografiNivaa) {
10|         super(sti, navn);

```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/felles/valghierarki/model/Valgdistrikt.java 7

**Fact Name:** no.valg.eva.admin.felles.valghierarki.model.Valgdistrikt.valggeografiNivaa

```

4|import no.valg.eva.admin.felles.valggeografi.model.ValggeografiNivaa;
5|
6|public class Valgdistrikt extends Valghierarki<ValgdistriktSti> {
> 7|     private final ValggeografiNivaa valggeografiNivaa;
8|
9|     public Valgdistrikt(ValgdistriktSti sti, String navn, ValggeografiNivaa valggeografiNivaa) {
10|         super(sti, navn);

```

## 6.353 Valggeografi.java

**Issue** 22072C976817A0D3EBDC97BD9A4A5CF1:

**Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/felles/valggeografi/model/Valggeografi.java 15

**Fact Name:** no.valg.eva.admin.felles.valggeografi.model.Valggeografi.navn

```

13|public abstract class Valggeografi<S extends ValggeografiSti> implements Serializable {
14|     private final S sti;
>15|     private final String navn;
16|
17|     public Valggeografi(S sti, String navn) {
18|         this.sti = sti;

```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/felles/valggeografi/model/Valggeografi.java 15

**Fact Name:** no.valg.eva.admin.felles.valggeografi.model.Valggeografi.navn

```

13|public abstract class Valggeografi<S extends ValggeografiSti> implements Serializable {
14|     private final S sti;
>15|     private final String navn;
16|
17|     public Valggeografi(S sti, String navn) {
18|         this.sti = sti;

```

**Issue** 837D1E2346A4BCF35DAD4FC6FBC2B247:

**Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/felles/valggeografi/model/Valggeografi.java 14

**Fact Name:** no.valg.eva.admin.felles.valggeografi.model.Valggeografi.sti

```

11|import org.apache.commons.lang3.builder.ToStringBuilder;
12|
13|public abstract class Valggeografi<S extends ValggeografiSti> implements Serializable {
>14|     private final S sti;
15|     private final String navn;
16|
17|     public Valggeografi(S sti, String navn) {

```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/felles/valggeografi/model/Valggeografi.java 14

**Fact Name:** no.valg.eva.admin.felles.valggeografi.model.Valggeografi.sti

```

11| import org.apache.commons.lang3.builder.ToStringBuilder;
12|
13| public abstract class Valggeografi<S extends ValggeografiSti> implements Serializable {
>14|     private final S sti;
15|     private final String navn;
16|
17|     public Valggeografi(S sti, String navn) {

```

## 6.354 ValggeografiNivaa.java

**Issue** 63D7452C8DD8E6EAEC8731833BB2A5BA:

**Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/felles/valggeografi/model/ValggeografiNivaa.java 35

**Fact** Name: no.valg.eva.admin.felles.valggeografi.model.ValggeografiNivaa.nivaa

```

>35| private final int nivaa;
36|
37| ValggeografiNivaa(int nivaa) {
38|     this.nivaa = nivaa;

```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/felles/valggeografi/model/ValggeografiNivaa.java 35

**Fact** Name: no.valg.eva.admin.felles.valggeografi.model.ValggeografiNivaa.nivaa

```

>35| private final int nivaa;
36|
37| ValggeografiNivaa(int nivaa) {
38|     this.nivaa = nivaa;

```

## 6.355 ValggeografiPanel.java

**Issue** 96C49709BCE86042E0F578B3C062D1F1:

**Source :**

**File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/kontekstvelger/panel/ValggeografiPanel.java 66

**Fact** Name: no.valg.eva.admin.frontend.kontekstvelger.panel.ValggeografiPanel.selectableElectionGeoLevels

```

63| private OpptellingskategoriPanel opptellingskategoriPanel;
64| private ValggeografiTjeneste valggeografiTjeneste = ValggeografiTjeneste.DEFAULT;
65| private List<ValggeografiNivaa> electionGeoLevels;
>66| private Set<ValggeografiNivaa> selectableElectionGeoLevels;
67| private ValggeografiFilter valggeografiFilter = ValggeografiFilter.DEFAULT;
68| private ValggeografiVariant valggeografiVariant = ValggeografiVariant.STANDARD; 69| private
ValggeografiNivaa selectedElectionGeoLevel;

```

**Sink :**

**File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/kontekstvelger/panel/ValggeografiPanel.java 66

**Fact** Name: no.valg.eva.admin.frontend.kontekstvelger.panel.ValggeografiPanel.selectableElectionGeoLevels

```

63| private OpptellingskategoriPanel opptellingskategoriPanel;
64| private ValggeografiTjeneste valggeografiTjeneste = ValggeografiTjeneste.DEFAULT;
65| private List<ValggeografiNivaa> electionGeoLevels;
>66| private Set<ValggeografiNivaa> selectableElectionGeoLevels;
67| private ValggeografiFilter valggeografiFilter = ValggeografiFilter.DEFAULT;
68| private ValggeografiVariant valggeografiVariant = ValggeografiVariant.STANDARD; 69| private
ValggeografiNivaa selectedElectionGeoLevel;

```

## 6.356 ValggeografiTabell.java

**Issue** 2DB715DCC80C05639D74A1D7DoC4996C:

**Source :**

**File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/kontekstvelger/valggeografi/ValggeografiTabell.java 26

**Fact** Name: no.valg.eva.admin.frontend.kontekstvelger.valggeografi.ValggeografiTabell.countCategory

```

23| extends KontekstvelgerTabell<ValggeografiPanel, ValggeografiRad<S>, ValggeografiNivaa> {
24|private final ValggeografiNivaa nivaa;
25|private ValggeografiFilter filter;

```

```

>26|private CountCategory countCategory; 27|
28|public ValggeografiTabell(ValggeografiPanel panel, ValggeografiNivaa nivaa, CountCategory countCategory) {
29| super(panel, true);

```

**Sink :**

**File** admin-frontend/src/main/java/no/valg/eva/admin/frontend/kontekstvelger/valggeografi/ValggeografiTabell.java 26 **Fact**  
**Name:** no.valg.eva.admin.frontend.kontekstvelger.valggeografi.ValggeografiTabell.countCategory

```

23| extends KontekstvelgerTabell<ValggeografiPanel, ValggeografiRad<S>, ValggeografiNivaa> {
24|private final ValggeografiNivaa nivaa;
25|private ValggeografiFilter filter;
>26|private CountCategory countCategory; 27|
28|public ValggeografiTabell(ValggeografiPanel panel, ValggeografiNivaa nivaa, CountCategory countCategory) {
29| super(panel, true);

```

## 6.357 Valghierarki.java

### Issue 44CB7951DC4C543CC1AA66F5EC70CB7F:

**Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/felles/valghierarki/model/Valghierarki.java 16  
**Fact** **Name:** no.valg.eva.admin.felles.valghierarki.model.Valghierarki.navn

```

14|public abstract class Valghierarki<S extends ValghierarkiSti> implements Serializable {
15| private final S sti;
>16| private final String navn;
17|
18| public Valghierarki(S sti, String navn) {
19| this.sti = sti;

```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/felles/valghierarki/model/Valghierarki.java 16  
**Fact** **Name:** no.valg.eva.admin.felles.valghierarki.model.Valghierarki.navn

```

14|public abstract class Valghierarki<S extends ValghierarkiSti> implements Serializable {
15| private final S sti;
>16| private final String navn;
17|
18| public Valghierarki(S sti, String navn) {
19| this.sti = sti;

```

### Issue 4AC8ABCB4B34FE1F6BC1107536CoF459:

**Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/felles/valghierarki/model/Valghierarki.java 15  
**Fact** **Name:** no.valg.eva.admin.felles.valghierarki.model.Valghierarki.sti

```

12|import org.apache.commons.lang3.builder.ToStringBuilder;
13|
14|public abstract class Valghierarki<S extends ValghierarkiSti> implements Serializable {
>15| private final S sti;
16| private final String navn;
17|
18| public Valghierarki(S sti, String navn) {

```

**Sink :**

**File** admin-common/src/main/java/no/valg/eva/admin/felles/valghierarki/model/Valghierarki.java 15  
**Fact** **Name:** no.valg.eva.admin.felles.valghierarki.model.Valghierarki.sti

```

12|import org.apache.commons.lang3.builder.ToStringBuilder;
13|
14|public abstract class Valghierarki<S extends ValghierarkiSti> implements Serializable {
>15| private final S sti;
16| private final String navn;
17|
18| public Valghierarki(S sti, String navn) {

```

## 6.358 ValghierarkiNivaa.java

### Issue 41C8847512F8B60ED10CBA1E743C97DF:

**Source :**

**File** admin-common/src/main/java/no/valg/eva/admin/felles/valghierarki/model/ValghierarkiNivaa.java 20 **Fact**  
**Name:** no.valg.eva.admin.felles.valghierarki.model.ValghierarkiNivaa.nivaa

```
17|VALG(2),
18|VALGDISTRIKT(3);
19|
>20|private final int nivaa;
21|
22|private static final Map<Integer, ValghierarkiNivaa> ELECTION_HIERARCHY_LEVEL_MAP = new HashMap<>();
```

**Sink :****File** admin-common/src/main/java/no/valg/eva/admin/felles/valghierarki/model/ValghierarkiNivaa.java 20**Fact** Name: no.valg.eva.admin.felles.valghierarki.model.ValghierarkiNivaa.nivaa

```
17|VALG(2),
18|VALGDISTRIKT(3);
19|
>20|private final int nivaa;
21|
22|private static final Map<Integer, ValghierarkiNivaa> ELECTION_HIERARCHY_LEVEL_MAP = new HashMap<>();
```

**[Low] Weak Cryptographic Hash / Insecure PBE Iteration Count***Security Features***Analysis:** Suspicious**Abstract**

The iteration count used by a password-based key derivation function is too low.

**Explanation**

A key derivation function is used to derive a key from a base key and other parameters. In a password-based key derivation function, the base key is a password and the other parameters are a salt value and an iteration count. An iteration count has traditionally served the purpose of increasing the cost of generating keys from a password. If the iteration count is too low, the feasibility of an attack increases as an attacker may compute "rainbow tables" for the application and more easily determine the hashed password values.

**Example 1:** The following code uses an iteration count of 50:

```
... final int
iterationCount=50;
PBEPParameterSpec pbeps=new PBEPParameterSpec(salt,iterationCount);
...
```

Applications that use a low iteration count for password-based encryption are vulnerable to trivial dictionary-based attacks -- exactly the type of attack that password-based encryption schemes were designed to protect against.

**PBE (Password-Based Encryption) schemes**

Best practices for use of `PBEPParameterSpec` include: - use a non-constant salt - use at least 1000 iterations

Best practices for use of `PBEKeySpec` include: - use a non-constant salt - use at least 10000 iterations

See "An Empirical Study of Cryptographic Misuse in Android Applications" for a more thorough explanation.

This finding is from research found in "An Empirical Study of Cryptographic Misuse in Android Applications". [http://www.cs.ucsb.edu/~chris/research/doc/ccs13\\_cryptolint.pdf](http://www.cs.ucsb.edu/~chris/research/doc/ccs13_cryptolint.pdf) and recommendations in "NIST Special Publication 800-132" <http://csrc.nist.gov/publications/nistpubs/800-132/nist-sp800-132.pdf>

**Recommendation**

When using password-based key derivation functions, the iteration count should be at least 1,000. This will significantly increase the cost of an exhaustive search for passwords without a noticeable impact on the cost of deriving individual keys. NIST SP 800-132 recommends using an iteration count of as high as 10,000,000 for critical keys or very powerful systems.

**Example 2:** The following code uses an iteration count of 100,000:

```
...
final int iterationCount=100000;
PBEPParameterSpec pbeps=new PBEPParameterSpec(salt,iterationCount); ...
```

**6.359 CryptoServiceBean.java****Issue** 7DB7E37BE48EFAB50866261B67842E41:

**Sink :****File** admin-backend-common/src/main/java/no/evote/service/CryptoServiceBean.java 161**Action** InCall PBEKeySpec()**Rule id** B2C6376E-AD55-434D-AD71-33D341AA3A1C

```
158|byte[] salt = new byte[8];
159|// Salt is first 8 bytes of stored encrypted value
160|System.arraycopy(dataToDecrypt, 8, salt, 0, 8);
>161|PBEKeySpec keySpec = new PBEKeySpec(password.toCharArray(), salt, 1, 256); 162|
163|key = generateSecretKeyBC(PBEWITHMD5AND256BITAES_CBC_OPENSSL, keySpec);
```

**Issue F66933AA80A4601D86FED932A9DDCD46:****Sink :****File** admin-backend-common/src/main/java/no/evote/service/CryptoServiceBean.java 114**Action** InCall PBEKeySpec()**Rule id** B2C6376E-AD55-434D-AD71-33D341AA3A1C

```
111|random.nextBytes(salt);
112|
113|
>114|PBEKeySpec keySpec = new PBEKeySpec(password.toCharArray(), salt, 1, 256); 115|
116|key = generateSecretKeyBC(PBEWITHMD5AND256BITAES_CBC_OPENSSL, keySpec);
```



## Valgdirektoratets oppfølging av koderevisjonen

### Innledning

Valgdirektoratet arbeider kontinuerlig med å forbedre sine systemer, koderevisjoner er en del av denne forbedringsprosessen og gir viktige innspill for bedring av kodekvalitet og økt sikkerhet. Våren 2018 bestilte Valgdirektoratet koderevisjon for EVA applikasjonene EVA Admin, EVA Skanning og EVA Resultat fra en ekstern markedsaktør.

### Målsetning

Målsetningen for oppdraget var å få en uavhengig vurdering av kildekode og systemarkitektur for EVA-porteføljen der svakheter belyses langs to akser:

- *Sikkerhet* – Med sikkerhet menes feil eller mangler i applikasjonskode og/eller arkitektur som vil kunne ha konsekvenser for valgresultatet i den forstand at kode og/eller arkitektur åpner for/ikke tilstrekkelig utelukker 3. parts manipulasjon.
- *Kvalitet* – Med kvalitet menes feil eller mangler i kode og/eller arkitektur som er potensielle feilkilder og/eller som reduserer forståelse, testbarhet og vedlikeholbarhet. Typiske kandidater:
  - o Kodekompleksitet
  - o Kodeorganisering
  - o Feil eller u hensiktsmessig bruk av biblioteker / rammeverk etc.

Vurderingene skulle brukes i forbedringsarbeidet av hele porteføljen for EVA

### Koderevisjonsprosess

Leverandør og Valgdirektoratet gjennomførte oppstartsmøte med innføring i applikasjoner og domene før leverandør gjorde sin revisjon og analyse. Ved oppdragets avslutning gikk leverandør gjennom funnrapporter med Valgdirektoratet.

Funnene fra leverandør ble så gjennomgått og vurdert av Valgdirektoratet, der Valgdirektoratet gjorde sin vurdering basert på fra alvorlighetsgrad angitt av leverandør. Utfallet av analysen var tredelt:

- Enkelte funn var falske positive, dvs. at funnene ikke var reelle funn når kontekst for funnene ble vurdert, disse ble avvist
- De mest alvorlige funnene ble prioritert og rettet før produksjonssetting
- Mindre alvorlige funn ble prioritert ned og inngår i den generelle forbedringsprosessen for EVA produktene. Dette er funn som ikke er direkte sårbarheter, og som vurderes og prioriteres i videre utviklingsløp.

Valgdirektoratet er med denne prosessen trygg på at EVA-applikasjonene holder god kvalitet og er tilstrekkelig sikret for valget i 2019.

## Avviksrapport – EVA Admin

### Anbefalinger – Kodekvalitet og arkitektur

Anbefaling 1 Se over eventuelle avhengigheter mellom applikasjons- og testkode

Vurderes i videre utviklingsløp, behovet for bygging utenfor Valgdirektoratets rammeverk er begrenset knyttet til forvaltning og produksjon.

Anbefaling 2 Alle del-prosjekter (repositories) bør være avgrenset til logiske blokker som gir mening og en konsekvent oppdeling av koden

Restrukturering og kodeorganiseringen etter gjeldene mål bilde (se systemdokumentasjon) er en del av pågående arbeid.

Anbefaling 3 Java “package”-strukturen bør gjenspeile strukturen i koden (repositories)

Restrukturering og kodeorganiseringen etter gjeldene mål bilde (se systemdokumentasjon) er en del av pågående arbeid.

Anbefaling 4 Fullfør rydding av evote-spesifikk funksjonalitet

Restrukturering og kodeorganiseringen etter gjeldene mål bilde (se systemdokumentasjon) er en del av pågående arbeid

## Anbefalinger – kryptorelaterte funn

Anbefaling 5 Se på muligheten for å benytte mer moderne krypto-primitiver

Funn knyttet til systempassord i EVA Admin. Systempassord er knyttet til den tidligere e-valgløsningen og er vurdert fjernet, men endelig analyse er ikke ferdigstilt. Autentisering og integritet ved frontend-, backend- og databasetilgang er mitigert ved sikkerhetsmekanismer i nettverk, se systemdokumentasjon «Sikring av infrastruktur».

Anbefaling 6 Benytt operativsystemets mekanismer for sikker slumpallgenerering

Funn knyttet til systempassord i EVA Admin. Systempassord er knyttet til den tidligere e-valgløsningen og er vurdert fjernet, men endelig analyse er ikke ferdigstilt. Autentisering og integritet ved frontend-, backend- og databasetilgang er mitigert ved sikkerhetsmekanismer i nettverk, se systemdokumentasjon «Sikring av infrastruktur».

Anbefaling 7 Sett antall iterasjoner for passordbasert kryptering til en høyere verdi

Funn knyttet til systempassord i EVA Admin. Systempassord er knyttet til den tidligere e-valgløsningen og er vurdert fjernet, men endelig analyse er ikke ferdigstilt. Autentisering og integritet ved frontend-, backend- og databasetilgang er mitigert ved sikkerhetsmekanismer i nettverk, se systemdokumentasjon «Sikring av infrastruktur».

Anbefaling 8 Verifikasjon av kryptografiske signaturer må ikke tillate usikre signeringsmekanismer (f.eks. ved å validere parametere satt i CMS SignerInfo)

Knyttet til opplasting av telleresultater og verifikasjon buypass signatur. Planlagt testet før eventuelle tiltak.

## Manuelle funn

### Direktory traversal

Se tabell «Avvikslogg - utbedrede funn» - «Zip-slip»

## Verktøybaserte funn

### Komponenter med sårbarheter

Sårbarhetene beskrevet i rapporten er utbedret der nyere versjon finnes, eller der oppgradering til nyere versjon ikke hindres (se reelle funn under). Valgdirektoratet har automatisert owasp i CI/CD byggpipeline som anbefalt av Mnemonic. Owasp-funn vurderes løpende i utviklingsprosessen som følger:

- Falske positiver / ikke direkte sårbarheter: Sårbarheter i biblioteker som brukes av biblioteker som brukes av EVA Admin, men der underliggende biblioteks funksjonalitet ikke brukes
- Reelle funn:
  - o Oppgraderes direkte dersom det finnes nyere versjon tilgjengelig og dersom konsekvensene av oppgradering er kontrollerbare
  - o Planlegges dersom det finnes nyere versjon tilgjengelig, men der konsekvensene av oppgradering ikke er kontrollerbare
  - o Legges på vent dersom nyere versjon ikke er tilgjengelig

## Verktøybaserte funn

### Kildekodefunn

#### Avvikslogg – Utbedrede funn

Avvik	Kommentar
<p>Cross-Site Scripting / External Links</p> <p>Analysis: Suspicious Abstract: Sending unvalidated data to a web browser can result in the browser executing malicious code. Settings in the configuration can minimize and reduce the exposure to cross-site scripting</p>	<p>Endret i antiSamy-konfigurasjon lenker er ikke tillatt i http-innhold</p>
<p>Weak Cryptographic Hash / Insecure PBE Iteration Count</p> <p>Analysis: Suspicious Abstract: The iteration count used by a password-based key derivation function is too low.</p>	<p>Økt antall iterasjoner</p>
<p>ZIP-slip (rapportert i etterkant)</p> <p>Denne sårbarheten kan oppstå når man prøver å pakke ut ziparkiver uten å validere filnavnene som pakkes ut.</p>	<p>Validerer at zip-fil ikke har filer med relative filstier til andre steder i filsystemet</p>
<p>Manglende automatisert kontroll på utdaterte bibliotek (OWASP)</p>	<p>Introdusert OWASP-jobb i bygg-pipeline</p>
<p>End of life for SAML2-protokollen for Apache Shibboleth - Id porten integrasjon - autentisering</p>	<p>Erstattet Shibboleth og SAML2 med OpenId connect for autentisering mot Id-porten</p>

#### Avvikslogg – False positiver / ikke relevante funn.

Avvik	Kommentar
<p>[High] Password Management / Hardcoded Password Security Features, Analysis: Suspicious</p>	<p>Vi ville egentlig ha en løsning uten pålogging. Det var imidlertid enklere å implementere den med pålogging. Passordet er derfor satt likt brukernavnet.</p>
<p>[Medium] J2EE Misconfiguration / Excessive Session Timeout</p>	<p>Det er krav fra brukerne om å ha så lang timeout.</p>

	Environment, Analysis: Suspicious	
	[Medium] J2EE Misconfiguration / Incomplete Error Handling Environment, Analysis: Bad Practice	Håndteres på annen måte, implementert i BaseController.java og RequestBroker.js
	[Medium] Weak Cryptographic Hash Security Features, Analysis: Suspicious	Brukes ikke for faktisk kryptering, men for å få en hash i stedet for tekst som kunne stått i klartekst
	[Low] Code Correctness / Erroneous String Compare Code Quality, Analysis: Reliability Issue	Alle tilfellene er ved bruk i switch-statement, og i følge Java-dokumentasjonen benyttes equals for sammenligningen. Vi anser funnene som falske positive. Se Oracle dokumentasjon
	Privacy Violation / Heap Inspection Analysis: Suspicious Abstract: Storing sensitive data in a String object makes it impossible to reliably purge the data from memory.	Passordet må ligge i minnet for å kunne bruke p12 sertifikatene. Denne sårbarheten er mitigert. Koden eksekveres på en backend server som ikke er tilgjengelig fra noen andre servere enn EVA Admin frontend serveren. Det å tappe minnet på denne maskinen er derfor meget vanskelig.

### Avvikslogg – Restanseliste mindre kritiske funn

Avvik	Kommentar
XML Entity Expansion Injection Analysis: Suspicious Abstract: Using XML parsers configured to not prevent nor limit Document Type Definition (DTD) entity resolution can expose the parser to an XML Entity Expansion injection	Backlog Merk, dette gjelder genering av XML basert på data i EVA Admin, ikke mottak av XML fra eksternt system
XML External Entity Injection Analysis: Suspicious Abstract: Using XML parsers configured to not prevent nor limit external entities resolution can expose the parser to an XML External Entities attack	Merk, dette gjelder genering av XML basert på data i EVA Admin, ikke mottak av XML fra eksternt system
Code Correctness / Constructor Invokes Overridable Function Analysis: Bad Practice Abstract: A constructor of the class calls a function that can be overridden.	Backlog
Missing Check against Null Analysis: Reliability Issue Abstract: The program can dereference a null pointer because it does not check the return value of a	Backlog

	function that might return null.	
	Missing Check for Null Parameter Analysis: Bad Practice Abstract: This function violates the contract that it must compare its parameter with null.	Backlog
	Poor Error Handling / Throw Inside Finally Analysis: Bad Practice Abstract: Using a throw statement inside a finally block breaks the logical progression through the try-catch-finally.	Backlog
	Redundant Null Check Analysis: Bad Practice Abstract: The program can potentially dereference a null pointer, thereby causing a null pointer exception.	Backlog
	System Information Leak / HTML Comment in JSP Analysis: Bad Practice Abstract: Any information revealed in an HTML comment might help an adversary learn about the system and form a plan of attack.	Backlog
	Unreleased Resource / Streams Analysis: Reliability Issue Abstract: The program can potentially fail to release a system resource.	Backlog
	Denial of Service Analysis: Reliability Issue Abstract: An attacker could cause the program to crash or otherwise become unavailable to legitimate users.	Backlog
	Missing Check against Null Analysis: Reliability Issue Abstract: The program can dereference a null pointer because it does not check the return value of a function that might return null.	Backlog
	Object Model Violation / Erroneous clone() Method Analysis: Reliability Issue Abstract: A clone() method should call super.clone() to obtain the new object.	Backlog
	Object Model Violation / Just one of equals() and hashCode() Defined Analysis: Bad Practice Abstract: This class overrides only one of equals() and hashCode().	Backlog

<p>Obsolete          Analysis: Bad Practice          Abstract: The use of deprecated or obsolete functions could indicate neglected code.</p>	<p>Backlog</p>
<p>Poor Error Handling / Overly Broad Catch          Analysis: Bad Practice          Abstract: The catch block handles a broad swath of exceptions, potentially trapping dissimilar issues or problems that should not be dealt with at this point in the program.</p>	<p>Backlog</p>
<p>Poor Error Handling / Overly Broad Throws          Analysis: Bad Practice          Abstract: The method throws a generic exception making it harder for callers to do a good job of error handling and recovery.</p>	<p>Backlog</p>
<p>Poor Error Handling / Program Catches NullPointerException          Analysis: Bad Practice          Abstract: It is generally a bad practice to catch NullPointerException.</p>	<p>Backlog</p>
<p>Poor Logging Practice / Logger Not Declared Static Final          Analysis: Bad Practice          Abstract: Declare loggers to be static and final.</p>	<p>Backlog</p>
<p>Poor Style / Confusing Naming          Analysis: Bad Practice          Abstract: The class contains a field and a method with the same name.</p>	<p>Backlog</p>