XML External Entity Attacks

Dear E-Voting PIT Team

Glad to get in touch with you again, even though we try to not spam you.

Here we provide you with potential XXE vulnerabilites. (We checked for all the common java xml parsing classes)

TransformerFactory

/ evoting-solution/source-code/scytl-cryptolib/cryptolib-asymmetric/src/main/java/com/scytl/cryptolib/asymmetric/utils/DomUtils.java/com/scytl/cryptolib/asymmetric/utils/DomUtils.java/com/scytl/cryptolib/asymmetric/utils/DomUtils.java/com/scytl/cryptolib/asymmetric/utils/DomUtils.java/com/scytl/cryptolib/asymmetric/utils/DomUtils.java/com/scytl/cryptolib/asymmetric/utils/DomUtils.java/com/scytl/cryptolib/asymmetric/utils/DomUtils.java/com/scytl/cryptolib/asymmetric/utils/DomUtils.java/com/scytl/cryptolib/asymmetric/utils/DomUtils.java/com/scytl/cryptolib/asymmetric/utils/DomUtils.java/com/scytl/cryptolib/asymmetric/utils/DomUtils.java/com/scytl/cryptolib/asymmetric/utils/DomUtils.java/com/scytl/cryptolib/asymmetric/utils/DomUtils.java/com/scytl/cryptolib/asymmetric/utils/DomUtils.java/com/scytl/cryptolib/asymmetric/utils/DomUtils.java/com/scytl/cryptolib/asymmetric/utils/DomUtils.java/com/scytl/cryptolib/asymmetric/utils/DomUtils.java/com/scytl/cryptolib/asymmetric/utils/DomUtils/asymmetric/utils/DomUtils/asymmetric/utils/DomUtils/asymmetric/utils/DomUtils/asymmetric/utils/DomUtils/asymmetric/utils/DomUtils/asymmetric/utils/DomUtils/asymmetric/utils/asymmetri

There is an important setting missing:

```
transformerFactory.setAttribute(XMLConstants.ACCESS_EXTERNAL_STYLESHEET, "");
```

JAXB Unmarshaller

We are not sure about this one.

/home/muffinx/projects/evoting-solution/source-code/online-voting-secure-data-manager/secure-data-manager-backend/secure-data-manager-integration/src/main/java/com/scytl/products/ov/sdm/plugin/XmlObjectsLoader.java

```
private static Unmarshaller create(String schemaPath, Class<?> clazz) throws JAXBException, SAXException {
    Unmarshaller unmarshaller = JAXBContext.newInstance(clazz).createUnmarshaller();
    if (schemaPath != null) {
        Schema schema = getSchema(schemaPath, clazz);
        unmarshaller.setSchema(schema);
    }
    return unmarshaller;
}
```

Now there is a schema being set with anti-XXE settings, but still we are not sure about the effectiveness of this approach (pretty specific question):

```
public static Plugins unmarshal(Path xml) throws IOException, JAXBException, SAXException {
    return unmarshal(xml, "/xsd/plugins.xsd", Plugins.class);
}

@SuppressWarnings("unchecked")
public static <T> T unmarshal(final Path xml, final String schemaPath, Class<?> clazz)
    throws IOException, JAXBException {
```

1 von 3 21.02.2019, 20:32

```
try (InputStream is = Files.newInputStream(xml)) {
    return (T) create(schemaPath, clazz).unmarshal(is);
}

private static Unmarshaller create(String schemaPath, Class<?> clazz) throws JAXBException, SAXException {
    Unmarshaller unmarshaller = JAXBContext.newInstance(clazz).createUnmarshaller();
    if (schemaPath != null) {
        Schema schema = getSchema(schemaPath, clazz);
        unmarshaller.setSchema(schema);
    }
    return unmarshaller;
}
```

We include this to be 100% sure.

SAXReader

```
try {
    HttpsURLConnection con = (HttpsURLConnection)metadataUrl.openConnection();
    makeIgnoreCertificate(con);
    Document metadata = new SAXReader().read(con.getInputStream());
    this.ec2endpoint = readURLFromMetadata(metadata, "ec2");
    this.s3endpoint = readURLFromMetadata(metadata, "s3");
}
```

The class is directly initialized with missing settings:

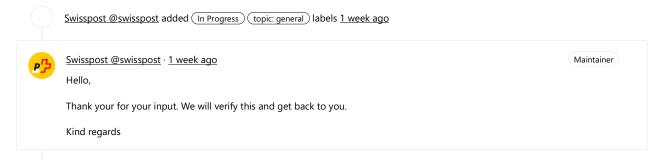
```
SAXBuilder builder = new SAXBuilder();
builder.setFeature("http://apache.org/xml/features/disallow-doctype-decl",true);
builder.setFeature("http://xml.org/sax/features/external-general-entities", false);
builder.setFeature("http://xml.org/sax/features/external-parameter-entities", false);
Document doc = builder.build(new File(fileName));
```

Take these serious as well, since if these work, an attacker could leak files and do SSRF attacks.

Thank you very much for reading our report.

Best Regards

Anthony Schneiter aka. muffinx Jannis Kirschner aka. xorkiwi



```
Swisspost @swisspost · 3 days ago
Hello Anthony,

Thank you for your observation. It is indeed a bug in the DomUtils.java class. We are going to fix it in a future version. Since this class is not used in the sVote solution (sVote does not work with XML for Cryptolib related operations), we consider it a minor bug.

Does this reply suffice to you?

Kind regards
```

2 von 3 21.02.2019, 20:32

 $\underline{\text{Swisspost}} \; \underline{\text{@swisspost}} \; \text{removed} \; \underline{\text{In Progress}} \; \text{label} \; \underline{\text{1 day ago}}$

 $\underline{\text{Swisspost @swisspost}} \text{ made the issue visible to everyone } \underline{\text{1 day ago}}$

3 von 3 21.02.2019, 20:32

XML External Entity Attacks (#18) · Issues · Swisspost / evoting-solut... https://gitlab.com/swisspost/evoting-solution/issues/18