Payback on Web Attacker: Web Honeypots
ME?

VULNEX:  www.vulnex.com

Blog:  www.simonroses.com

Twitter:  @simonroses / @vulnexsl

Former: Microsoft, PwC & @Stake
CISSP, CEH & CSSLP
Honeypots can add another line of defense in corporate networks

Plenty of honeypots but still immature solutions

Web Honeypots Fun
AGENDA

1. HONEYPOT OVERVIEW
2. CARMEN DESIGN
3. ATTACK MADNESS
4. CONCLUSIONS
1. Honeypot Overview
“The new NATO Computer Incident Response Capability (NCIRC) will be designed around a defense scheme that may well include such components as traps for attackers known as honeypots, among other proactive defense tools, one of the experts behind the program said July 11 at the Farnborough International Airshow.

The NCIRC contract was awarded to a Northrop Grumman/Finmeccanica team in February, and the program has passed its proof of concept testing, company representatives said at the press conference.

NCIRC is scheduled to be fully operational by the end of 2012.

Dennis McCallam, director of information systems cyber security at Northrop Grumman, said the system is being built around an “active defense” approach. The term is used to describe everything from cyber situational awareness to operations that include responding to an attacker with disruptive actions.

“I think a key component of this is the ability to have an adaptive, agile defense, and all those components that go with it to allow you to respond quickly and decisively,” McCallam said. McCallam said the system may include honeypots, which are fake, seemingly desirable targets designed to lure attackers into revealing their activity.

“If you Google active defense, honeypots will be in there,” he said. “It’s all part of the game. Whatever you can do to understand what’s going on and be able to adapt to it.”

Cyber experts continue to debate the value of honeypots, as the ability of a network administrator to use the information gained from the trap is disputed.

Company representatives said they were not permitted to discuss potential offensive cyber capability built into the design, but did not dispute that the system would be outfitted accordingly.”

1. HONEYPOT OVERVIEW

• Old concept but still plenty to do

• Honeypots no widely deployed except for:
  – Security / Malware research
  – Intelligence gathering

• Defense-in-Depth approach
1. OPEN SOURCE HONEYPOTS

- Commercial vs. Open Source solutions

- Plenty of honeypots options:
  - Network & systems
    - Honeyd
    - Dionaea
  - SMTP
    - Smtp-honeypot
  - SSH
    - Kippo
    - Kojoney
  - Web
    - Phpot
    - GHH – The Google Hack Honeypot
    - Glastopf
1. WEB HONEYPOTS

• Web Honeypots are not new either

• Usually runs on top of well-known web servers

• Main focus on analyzing attacker techniques / patterns

• Simple form of a web honeypot is just a well-known insecure app

• The truth is web honeypots are not trivial
2. Carmen Design
2. WEB HONEYPOT DESIRED FEATURES

• Should be:
  – Simple to install / use / run
  – Cross platform
  – Small Performance impact
  – Allow more than just analyzing attackers techniques
  – Easy to interact / extend with
2. CARMEN FEATURES

- 100% Python
- Cross platform
- Multithreading
- Easy to use, easy to run
- Simulate well-known web servers
- Basic Auth
- Cookie Generation / Session ID
- CGI
- Plug-in
2. SIMPLICITY

- Carmen uses XML files for configuration:
  - Set IP and ports to use
  - Define Web Server behavior
  - What HTTP Methods to accept
  - Define the HTTP OPTIONS
  - Define Cookies / Session ID

- Can’t be easier to run:
  - Win: Use start.bat
  - Linux: # carmen.py
2. RUNNING (I)

```
root@bt: ~/carmenV2
root@bt:~/carmenV2# python carmen.py

CARMEN

[+] Carmen Web Server V 2.0 Running at 192.168.1.36 : 80
[+] VULNEX
[+] http://www.vulnex.com
```
2. RUNNING (II)

Web Application Honeypot

Alea Jacta Est

VULNEX
(C) 2012
2. WEB SERVER SIMULATION

• Carmen can simulate well-known web servers:
  – Apache
  – IIS
  – Netscape
  – Carmen (Apache behavior)

• You can mix behaviors, “customization mode” 😊
2. CARMEN BEHAVIORS

APACHE
- HEADERS
- BODY
- CGI
- APPS

IIS
- HEADERS
- BODY
- CGI
- APPS

Default

APACHE
- HEADERS
- BODY
- CGI
- APPS

IIS
- HEADERS
- BODY
- CGI
- APPS

Custom
2. WEB FINGERPRINT EVASION

- Uses same “give-out” information techniques as well-known web servers:
  - HTTP Header banners
  - HTTP OPTION
  - HTTP Error codes / messages
  - Technology identification / correlation
2. **COOKIE / SESSION ID GENERATION**

- By default when you set a well-known web server it will use its known session ID type:
  - id_session=
  - carmen_cookie=
  - ASPSESSIONx=
  - Apache=
  - ID=
2. **SESSION ID ALGORITHMS**

- Uses 8 algorithms for the Session ID (random):

1. Uppercase Letters
2. Uppercase / Lowercase Letters
3. Letters and Numbers
4. Random Letters with MD5
5. Client IP + Letters
6. Client IP + Letters with XOR
7. Client IP + Numbers
8. Client IP + Numbers with XOR
2. BRUTE FORCE AD INFINITUM (I)

- Currently HTTP BASIC Auth only.

- You can configure what directories to protect.
2. **BRUTE FORCE AD INFINITUM (II)**

- Currently HTTP BASIC Auth only.
- You can configure what directories to protect.
- The joy:
  - Carmen never authenticates.
- Capture attack password list
3. Attack Madness
3. ATTACK DEMOS

1. Web Server Fingerprint
2. Session ID Analysis
3. Directory Listing
4. Password Brute Force
5. Cross Site Scripting (XSS)
6. PHP CGI (CVE-2012-1823)
3. ATTACK DEMO: WEB SERVER FINGERPRINT (I)

• Before launching an attack, attackers will identify:
  – Web Server Version
  – Extension, Modules, etc.
  – Application Server Version
  – HTTP Methods Allowed

• Popular fingerprint tools are:
  – HTTPRecon
  – Httprint
  – Hmap
  – WebserverFP
3. ATTACK DEMO: WEB SERVER FINGERPRINT (II)

[Images of terminal output and software interface showing the fingerprinting of a web server with details about the server version and the HTTP request responses.]
3. ATTACK DEMO: SESSION ID ANALYSIS (I)

• Well-known security tools incorporate session ID analysis capabilities.

• What we do is generate “junk” so attackers spend time / waste resources trying to analyze data.
3. ATTACK DEMO: SESSION ID ANALYSIS (II)

<table>
<thead>
<tr>
<th>Date</th>
<th>Value</th>
<th>Numeric</th>
<th>Difference</th>
</tr>
</thead>
<tbody>
<tr>
<td>2012/10/22 12:16:47.442</td>
<td>AADIPFDHNNQGABO2JAGTV</td>
<td>375971.99715019834966395</td>
<td>-1940442300115238470628191</td>
</tr>
<tr>
<td>2012/10/22 12:16:42.917</td>
<td>AAEVRDSPOZ0W3VUTBM0</td>
<td>54798458972149302124136</td>
<td>-176866753026387015987742352</td>
</tr>
<tr>
<td>2012/10/22 12:16:40.906</td>
<td>AAQTZGKJXZQWJLWUMY</td>
<td>190126610794797036056108</td>
<td>-1971567655287352742043859675</td>
</tr>
<tr>
<td>2012/10/22 12:16:36.182</td>
<td>AAYQIXCAEIDIZOUSA</td>
<td>278610557145286550121508</td>
<td>-19083786163292621760705825170</td>
</tr>
<tr>
<td>2012/10/22 12:15:41.336</td>
<td>ABFAFOZFKNKNXMWM</td>
<td>35158091055432537591073618</td>
<td>-14920436474719194910050929429</td>
</tr>
<tr>
<td>2012/10/22 12:16:52.558</td>
<td>ABKAAQLCTGXRGGMKMMS</td>
<td>41213321756112473882121574</td>
<td>-589977110107985717550240407</td>
</tr>
<tr>
<td>2012/10/22 12:18:11.63</td>
<td>ABQHAPVMXDKFCRLY6HF</td>
<td>4792702934594202810108663</td>
<td>-1901819121298420923731300</td>
</tr>
<tr>
<td>2012/10/22 12:15:33.807</td>
<td>ABSSKAECSXDNHUXKL</td>
<td>509956528035858638768524</td>
<td>-163971917640127142712578353</td>
</tr>
<tr>
<td>2012/10/22 12:18:16.891</td>
<td>ACDUEPNTFWSJCMVVLDR</td>
<td>6071158132430949296729499</td>
<td>-1421800443293705775824240545</td>
</tr>
<tr>
<td>2012/10/22 12:16:57.777</td>
<td>ACBTAAGGZCFFDTWUYCU</td>
<td>609218736907313794762344</td>
<td>-175198632140462160030988385279</td>
</tr>
<tr>
<td>2012/10/22 12:16:25.165</td>
<td>ACCGWH-BXAPGSZBYXNA</td>
<td>615288893785228583133054</td>
<td>-1435657634541932230621718282</td>
</tr>
<tr>
<td>2012/10/22 12:16:22.722</td>
<td>ACPTGCTDRGBRZJNTMXY</td>
<td>7702414195088055642451263</td>
<td>-99775943551024922824060301</td>
</tr>
<tr>
<td>2012/10/22 12:16:22.712</td>
<td>ACREVFRTISRZTFVRNMOY</td>
<td>7844114220228193881261661</td>
<td>-1631027425099978903554146549</td>
</tr>
<tr>
<td>2012/10/22 12:17:29.851</td>
<td>ACUFKHKJNIOBDQJAT</td>
<td>8186602199812030338855328</td>
<td>-21261032353323782758814313126</td>
</tr>
<tr>
<td>2012/10/22 12:17:38.898</td>
<td>ADJFDYTBMADMRPCTKCGWF</td>
<td>107885903620821701017226409</td>
<td>-1026559887857855158484062186531</td>
</tr>
<tr>
<td>2012/10/22 12:16:51.147</td>
<td>ADVGHECQHIHUKZKJTYR</td>
<td>112522590659493429814505981</td>
<td>-3879382253083957829212482400</td>
</tr>
<tr>
<td>2012/10/22 12:16:08.452</td>
<td>AEIQRNPMPHURCSQDFCPP</td>
<td>13784545006998657591827823</td>
<td>-168526133005457118554260738685</td>
</tr>
<tr>
<td>2012/10/22 12:17:48.954</td>
<td>AEVDCHPBGVSJFQICANTT</td>
<td>141894629545740843443396473</td>
<td>-6657511486525425174413752824</td>
</tr>
<tr>
<td>2012/10/22 12:16:49.343</td>
<td>AEWXZPFCWZPCWGOEDEE</td>
<td>1434677055338720388010690</td>
<td>-188642370983141447575262683</td>
</tr>
<tr>
<td>2012/10/22 12:18:30.593</td>
<td>AEZJWI2RHLADVDKYPRUA</td>
<td>14667046407635735215695522</td>
<td>-11827946539407737664437932</td>
</tr>
<tr>
<td>2012/10/22 12:15:43.944</td>
<td>AEFNZJRRLPFKFNNAVRX</td>
<td>15254118377353591045986961</td>
<td>-109872333533821478896611413617</td>
</tr>
<tr>
<td>2012/10/22 12:17:37.593</td>
<td>AGBKKRZCFZPSAMMZFYQN</td>
<td>1882763874261254706321599</td>
<td>-18758529262457297139612812</td>
</tr>
<tr>
<td>2012/10/22 12:14:17.565</td>
<td>AGKICNPKOKS5SDDHC</td>
<td>18857500562529113944001116</td>
<td>-158050263497585234150253078767</td>
</tr>
<tr>
<td>2012/10/22 12:17:05.410</td>
<td>AGKITYFQUYIBSINCNL</td>
<td>188605723398334217998415623</td>
<td>-19423932151725898174704936866</td>
</tr>
<tr>
<td>2012/10/22 12:16:42.37</td>
<td>AGXQCEMSM-HFTHCGUEI</td>
<td>203659731077577043868240124</td>
<td>-17240058342481505852408893</td>
</tr>
<tr>
<td>2012/10/22 12:16:37.690</td>
<td>AGZLUKBYGVNUGWROHUEVI</td>
<td>205736635336819273364203632</td>
<td>-1664360919973757927676048078766</td>
</tr>
<tr>
<td>2012/10/22 12:16:02.321</td>
<td>AGHCRKDPQXEZHMMZTCLPS</td>
<td>222345274649224313099517017228</td>
<td>-287325775367522995785454851119</td>
</tr>
</tbody>
</table>

Minimum: 35180010534521335971079316
Maximum: 19884726134858012248286586845
Range: 1.9829566E28

Used 32.93 of 455.12MB
3. ATTACK DEMO: SESSION ID ANALYSIS (III)
3. ATTACK DEMO: SESSION ID ANALYSIS (IV)

**WebScarab**

<table>
<thead>
<tr>
<th>Date</th>
<th>Value</th>
<th>Numeric</th>
<th>Difference</th>
</tr>
</thead>
<tbody>
<tr>
<td>2012/10/22</td>
<td>12:33:15</td>
<td>192.168.1.35</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>12:33:15</td>
<td>192.168.1.35.3</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>12:33:15</td>
<td>192.168.1.35.5</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>12:33:16</td>
<td>192.168.1.35.3</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>12:33:16</td>
<td>192.168.1.35.4</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>12:33:16</td>
<td>192.168.1.35.0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>12:33:16</td>
<td>192.168.1.35.7</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>12:33:16</td>
<td>192.168.1.35.0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>12:33:16</td>
<td>192.168.1.35.6</td>
<td>6</td>
</tr>
<tr>
<td></td>
<td>12:33:16</td>
<td>192.168.1.35.8</td>
<td>8</td>
</tr>
<tr>
<td></td>
<td>12:33:16</td>
<td>192.168.1.35.4</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td>12:33:16</td>
<td>192.168.1.35.5</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>12:33:16</td>
<td>192.168.1.35.1</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>12:33:16</td>
<td>192.168.1.35.2</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>12:33:17</td>
<td>192.168.1.35.6</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td>12:33:17</td>
<td>192.168.1.35.7</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>12:33:17</td>
<td>192.168.1.35.7</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>12:33:17</td>
<td>192.168.1.35.6</td>
<td>6</td>
</tr>
<tr>
<td></td>
<td>12:33:17</td>
<td>192.168.1.35.9</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>12:33:17</td>
<td>192.168.1.35.6</td>
<td>6</td>
</tr>
<tr>
<td></td>
<td>12:33:17</td>
<td>192.168.1.35.5</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>12:33:17</td>
<td>192.168.1.35.3</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>12:33:17</td>
<td>192.168.1.35.3</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>12:33:17</td>
<td>192.168.1.35.0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>12:33:17</td>
<td>192.168.1.35.9</td>
<td>9</td>
</tr>
<tr>
<td></td>
<td>12:33:17</td>
<td>192.168.1.35.9</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>12:33:18</td>
<td>192.168.1.35.2</td>
<td>2</td>
</tr>
</tbody>
</table>

**Min:** 0  **Max:** 9  **Range:** 9

Used 22.76 of 455.12MB
3. ATTACK DEMO: SESSION ID ANALYSIS (V)
3. ATTACK DEMO: DIRECTORY LISTING (I)

• We created directories that allow directory listing and put fake documents

• Ideally for OSINT confusion
• We created directories that ask for authentication

• It never gets authenticated

• Ideally to grab password lists
3. ATTACK DEMO: PASSWORD BRUTE FORCE (II)

![Authentication Required dialog box with user name and password fields]

VULNEX
(C) 2012
Authorization Required
3. ATTACK DEMO: CROSS SITE SCRIPTING - XSS (I)

- A simple CGI that simulates a web app to post message into a forum

- As a defense it performs a black list approach against XSS attacks
3. ATTACK DEMO: CROSS SITE SCRIPTING - XSS (II)
3. ATTACK DEMO: CROSS SITE SCRIPTING - XSS (III)

Message: HELLO OWASP APPSEC USA 2012
3. ATTACK DEMO: CROSS SITE SCRIPTING - XSS (IV)
3. ATTACK DEMO: CROSS SITE SCRIPTING - XSS (V)

Message: (XSS Attack Stopped) <script>alert("evil XSS")</script>
3. ATTACK DEMO: CROSS SITE SCRIPTING - XSS (VI)
3. ATTACK DEMO: CROSS SITE SCRIPTING - XSS (VII)
3. ATTACK DEMO: PHP CGI (CVE-2012-1823) (I)

- Plenty of fun bug this year
- CGI Web app that simulates bug
- Attacker get fake PHP code
3. ATTACK DEMO: PHP CGI (CVE-2012-1823) (II)
3. ATTACK DEMO: PHP CGI (CVE-2012-1823) (II)
3. ATTACK DEMO: PHP CGI (CVE-2012-1823) (II)

```php
<?php
include("config.php");
session_start();
if($_SERVER["REQUEST_METHOD"] == "POST")
{
    // username and password sent from Form
    $myusername=addslashes($_POST['username']);
    $mypassword=addslashes($_POST["password"]);

    $sql="SELECT id FROM admin WHERE username='$myusername' and passcode='$mypassword';
    $result=mysql_query($sql);
    $row=mysql_fetch_array($result);
    $active=$row['active'];
    $count=mysql_num_rows($result);

    // If result matched $myusername and $mypassword, table row must be 1 row
    if($count==1)
    {
```

**PHP CGI Bug "-s"**
3. IDEAS FOR WEB TRAPS / FUTURE MODULES

– External Web Honeypot
  • Fake PDFs (for OSINT / metadata confusion)
  • Crawler / Spiders mess-up

– Internal Web Honeypot
  • CMS
  • Document repository
  • Old systems
4. Conclusions
4. HONEYPOT CAVEATS

- Custom development is resource consuming

- Honeypots require attention (as wife does :)

- No “execute and forget” approach

- You are adding a vulnerable system into network
4. CONCLUSIONS

• Honeypots have a place in corporate defenses

• Can help to get intelligence on attacker against employees / company
4. HOW CAN YOU HELP?

- Development
- Testing
- Real Deployment Data
- Moral support 😊
5. Q&A

• Thanks!