Who We Are

- PaulDotCom Enterprises
  - PaulDotCom Security Weekly Podcast
  - Penetration Testing, Security Consulting, Device Testing

- PaulDotCom Community
  - Forum, IRC, Hack Naked TV, Wiki, Mailing List

- SANS Instructors & Certified Professionals
  - Upcoming courses all across the world!

http://pauldotcom.com/events/
The Challenge

- If you had to pick tools to take with you to test a web application for security, what would they be?
  - You are limited to web applications, no wireless, no client-side
  - You must test and identify common web app vulnerabilities (XSS, CSRF, SQL Injection)
  - You must show how you could penetrate systems, gain access, and keep that access to demonstrate risk
Best Of Web Application Testing Tools

1) **Nikto** - Web server enumeration
2) **Webscarab** - Proxy/Scanner
3) **w3af** - Web Exploit framework
4) **Firefox** - Multi-purpose
5) **Cenzic Hailstorm** - Deep testing
6) **Core IMPACT** - Web Exploitation

**Spotlight** - Samurai WTF
This Presentation Will Help Build Your Ninja Skills...

There is a network ninja in this picture....
Nikto

- Performs a quick analysis of the web server and applications
- Important to know which applications to attack (for “blind” tests)
- Nikto reports back on:
  - Common vulnerable CGI applications
  - Web server type, version, and add-ons
  - “Interesting” Files

Recently Updated! Check the web site: http://www.cirt.net/nikto2
Nikto

- After using Nmap for host discovery, run Nikto to scan the web servers
- This may spot vulnerabilities and other files of interest for attacking the application or even your target
  - Vulnerable Apache/IIS/PHP/.NET = PWNAGE
  - A file named “passwords.txt” is gold
  - Source code to the application or version if open source
Nikto

Step 1 - Run Nmap to find Web servers:

```
# nmap -PN -oA top1000 -T4 -n -sS --top-ports 1000 -iL targets.all
```

**NOTE:** This will not detect web servers running on alternate ports. To do this replace -sS with -sV. This will only do the top 1000 ports.

Integrate Nikto with Nessus:
Nikto

Step 2 - Parse the Nmap results into a format that Nikto can understand:

```
$ awk '/open/ {print $2 ":" $5}' top1000.gnmap | \ egrep '(80|443)' | cut -d/ -f1 > nmaphosts
```

This will produce a file with IP and Port:

```
1.1.1.1:80
2.2.2.2:443
```
Nikto

Step 3 - Run Nikto:

```
$ ./nikto.pl -h nmaphosts -o nmaphosts.out
```

Review the output:

+ Server: Apache/2.2.4 (Unix) mod_ssl/2.2.4 OpenSSL/0.9.8d
+ Root page / redirects to: https://portal.int.pauldotcom.com/
+ All CGI directories 'found', use '-C none' to test none
+ OSVDB-0: Non-standard header keep-alive returned by server, with contents: timeout=15, max=100
+ Apache/2.2.4 appears to be outdated (current is at least Apache/2.2.9). Apache 1.3.39 and 2.0.61 are also current.
<table>
<thead>
<tr>
<th>ID</th>
<th>Date</th>
<th>Method</th>
<th>Host</th>
<th>Path</th>
<th>Parameters</th>
<th>Status</th>
<th>Origin</th>
<th>Possible inj.</th>
<th>XSS</th>
<th>CRLF</th>
</tr>
</thead>
<tbody>
<tr>
<td>96</td>
<td>2008/08/26</td>
<td>GET</td>
<td><a href="http://example.com/">http://example.com/</a></td>
<td>/images/placehold.gif</td>
<td>?utmw=1</td>
<td>200 OK</td>
<td>Proxy</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>95</td>
<td>2008/08/26</td>
<td>GET</td>
<td><a href="http://example.com/">http://example.com/</a></td>
<td>/images/placehold.gif</td>
<td>?utmw=1</td>
<td>200 OK</td>
<td>Proxy</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>93</td>
<td>2008/08/26</td>
<td>GET</td>
<td><a href="http://example.com/">http://example.com/</a></td>
<td>/images/placehold.gif</td>
<td>?utmw=1=968</td>
<td>200 OK</td>
<td>Proxy</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>92</td>
<td>2008/08/26</td>
<td>GET</td>
<td><a href="http://example.com/">http://example.com/</a></td>
<td>/images/checkout.gif</td>
<td>?utmw=1=968</td>
<td>200 OK</td>
<td>Proxy</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>91</td>
<td>2008/08/26</td>
<td>GET</td>
<td><a href="http://example.com/">http://example.com/</a></td>
<td>/images/update_quantity.gif</td>
<td>?utmw=1=968</td>
<td>200 OK</td>
<td>Proxy</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>90</td>
<td>2008/08/26</td>
<td>GET</td>
<td><a href="http://example.com/">http://example.com/</a></td>
<td>/images/remove.gif</td>
<td>?utmw=1=968</td>
<td>200 OK</td>
<td>Proxy</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>89</td>
<td>2008/08/26</td>
<td>GET</td>
<td><a href="http://example.com/">http://example.com/</a></td>
<td>/images/save_cart.gif</td>
<td>?utmw=1=968</td>
<td>200 OK</td>
<td>Proxy</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>88</td>
<td>2008/08/26</td>
<td>GET</td>
<td><a href="http://example.com/">http://example.com/</a></td>
<td>/images/</td>
<td></td>
<td>200 OK</td>
<td>Proxy</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>87</td>
<td>2008/08/26</td>
<td>GET</td>
<td><a href="http://example.com/">http://example.com/</a></td>
<td>/Controls/images/clear.gif</td>
<td></td>
<td>200 OK</td>
<td>Proxy</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>86</td>
<td>2008/08/26</td>
<td>GET</td>
<td><a href="http://example.com/">http://example.com/</a></td>
<td>/shoppingcart.aspx</td>
<td></td>
<td>404 Not Found</td>
<td>Proxy</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

WebScarab Flags injection points!
Stops all requests/responses

Edit stuff here

Header Information

Page coming back from server to browser
Webscarab - Hidden Fields

Options:
- Reveal hidden fields in HTML pages: checked
- Prevent browser from caching content: checked
- Inject known cookies into requests: unchecked
- Get cookies from responses: checked

Screenshot shows a website with a pop-up menu for selecting one option. A red circle highlights a hidden field for email type.
About Mick

• 14+ years industry experience
• Sys Admin and developer roots
• Evil thoughts + curiosity + good deeds = Mick
• White hat hacker
You don't have any expensive tools – so what?

• Too poor (like me)
  - I work at a not-for-profit.

• Don't do this work enough to justify the $$$ tools
Ain't no school like the old school

- All 'sploits “in” the browser
- Back in my day we didn't have tool xyz! We pushed packets through the wire by hand!!
- – And we liked it that way!
Big gains for doing this this way

- Great way to truly learn
- Deeper understanding
- Almost zero false positives/negatives
- Easiest way to show examples
- Nearly 100% buy in from “the suits”
  - Hint: Works best with HTML injection ;-)
How do you do it?

- Don't re-invent the wheel
- Find resources that specialize in this method
  - A bit like drinking from the fire hose!
  - No he's not been doing this since 1970. ;-}
Remember: Some ways are better than others

- If you're truly hardcore, you can use telnet
- I like Firefox – YMMV
  - Some extensions can really help
  - Have to get creative with them though!
- (think evil thoughts)
Firefox extensions to checkout

- Firebug
- YSlow
- TestGen4Web
Firebug – Firefox extension

- Inline editing
- Great for
  - Hidden fields
  - Breaking forms
  - Messing with JavaScript
  - Making rogue sites and man-in-the-middle components (a talk for another day)
  - (Oh it's handy for web development too...)
YSlow – Firefox extension

- Requires firebug
- Recon like no other! Shows you:
  - Overall time
  - Gives breakdown of components
- Timing is typically THE best indicator!
  - Is it +/-?
  - By how much?
Firefox extension
TestGen4Web

• Allows record and playback
• You can modify your recordings
• Instant test system
• There's plenty of others out there.
• Experiment around!
So you want more of a system?

- It's OK. Web hacking by hand – or even extension isn't always the way to go.
- May we interest you in w3af?
w3af – the OSS web exploit system

• Look Ma! Version 1 is arriving!
• Impressive capabilities with high potential
• If you have a commercial tool – watch this one. Might disrupt this space.
• w3af could be an entire series webcasts!
• Thanks Seth for bashing me over the head with this tool!
w3af 101

- Vulnerability tester
- Manual test capabilities
- Extendable framework
- Request fuzzer
- Interceptor
- Quite nice GUI
- EXPLOIT SYSTEM!!
w3af additional resources

- Homepage: http://w3af.sourceforge.net/
Methodology

• Any tool you use should be geared towards helping you understand how a web application works.

• Commercial tools to help with that are:
  - Core Impact
  - Cenzic Hailstorm Professional

• Over the next few slides we will look at these tools further
Core Impact

• Yes, it is a network penetration testing tool…
• But they have been working on the web side of the house for about a year now
• Why??
• Because sometimes a web server is the best way to get a shell!
Site Structure
Why do we care about site structure?

- There are things that may indicate vulnerabilities
  - Process flow… Think CSRF
  - Robots.txt – What do they not want you to see?
  - Include.php – Configuration to a backend
- Does the Developer have a clue?
Why Core is on this list
Microsoft SQL Server 9.00.1399.06
sql> select loginid, employeekey from dimemployee
['adventure-works\cynthia0', '138']
['adventure-works\laural', '144']
['adventure-works\brandon0', '161']
['adventure-works\gordon0', '237']
['adventure-works\wendy0', '73']
['adventure-works\caro10', '171']
['adventure-works\jeano', '44']
['adventure-works\pete0', '104']
['adventure-works\douglas0', '77']
['adventure-works\jim0', '62']
['adventure-works\ovidiu0', '267']
['adventure-works\janeth0', '115']
['adventure-works\yvonne0', '42']
['adventure-works\fred0', '47']
['adventure-works\sylvester0', '225']
['adventure-works\brian2', '226']
['adventure-works\dan0', '105']
['adventure-works\jol', '264']
['adventure-works\mike0', '220']
['adventure-works\jae0', '291']
['adventure-works\david2', '55']
['adventure-works\michael3', '177']
['adventure-works\david0', '77']
['adventure-works\min0', '243']
['adventure-works\pengo', '43']
['adventure-works\robert0', '3']
['adventure-works\alexo', '159']
['adventure-works\george0', '64']
['adventure-works\david4', '89']

```
sql>
```
Cenzic Hailstorm Pro

- What if you are new to the field of web applications?
- What if you are a seasoned vet?
- Do you need two tools?
- No
Flexibility

• Lets be honest…
• Sometimes you are checking a box.
• Wouldn't you like to have the ability to check that box and do a rock’en assessment?
• Or, do you want to write your own checks?
Compliance Checks

Security Assessment

Run Assessment

1. Choose Assessment Type:
   - Best Practices
   - OWASP
   - GLBA
   - HIPAA
   - PCI
   - SOX
   - MCARD
   - Traverse Only

2. Specify Login Credentials:
   - Password: ___________________________
   - Confirm Password: __________________
   - Application Name: __________________

Run | Cancel | Help
BYOA = Build Your Own Attacks

SmartAttack Editor

<table>
<thead>
<tr>
<th>Param Name</th>
<th>Param Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>maxTraversalRetries</td>
<td>4</td>
</tr>
<tr>
<td>attacksToDelay</td>
<td>&quot;login&quot;</td>
</tr>
<tr>
<td>makeRequestsUsingSockets</td>
<td>true</td>
</tr>
<tr>
<td>startPolicyAfterTraversal</td>
<td>true</td>
</tr>
<tr>
<td>injectFormFields</td>
<td>true</td>
</tr>
<tr>
<td>injectCookies</td>
<td>true</td>
</tr>
<tr>
<td>injectAFileWaitTime</td>
<td>true</td>
</tr>
<tr>
<td>stopOnFirstFault</td>
<td>true</td>
</tr>
<tr>
<td>erroPageMatchExpr</td>
<td>&quot;***&quot;</td>
</tr>
<tr>
<td>URLsToSkip</td>
<td>[]</td>
</tr>
<tr>
<td>fieldsToSkip</td>
<td>[]</td>
</tr>
<tr>
<td>reportIndividualSuccesses</td>
<td>true</td>
</tr>
<tr>
<td>skipFirstNInjectableRequests</td>
<td>0</td>
</tr>
<tr>
<td>headersToInject</td>
<td>[]</td>
</tr>
<tr>
<td>skipEquivalentRequests</td>
<td>true</td>
</tr>
<tr>
<td>considerCookiesWhenSkippingEquivalentRequests</td>
<td>true</td>
</tr>
<tr>
<td>maxRequestsWithSameURL</td>
<td>15</td>
</tr>
<tr>
<td>ajaxRequestFormatFile</td>
<td>&quot;AjaxRequestFormat.txt&quot;</td>
</tr>
<tr>
<td>detectionValueFile</td>
<td>&quot;data_FD_ApplicationException.txt&quot;</td>
</tr>
<tr>
<td>injectionLength</td>
<td>[135, 527, 1855, 4135]</td>
</tr>
<tr>
<td>injectionContent</td>
<td>&quot;ë&quot;</td>
</tr>
</tbody>
</table>
Other tools to check out

• Accunetix
• HP WebInspect
• You should really try more then one before you buy
• Always go with the one you are most comfortable with
Samurai

• A very special mention

• Want a live CD with all of the best Open Source web testing tools pre-installed and ready to go?

• Check out Samurai!
  - http://samurai.inguardians.com/
Honorable Mentions

- **Spike Proxy** [http://www.immunitysec.com/resources-freesoftware.shtml](http://www.immunitysec.com/resources-freesoftware.shtml) - Highly customizable proxy written in Python

- **Paros Proxy** - Not updated since 2006

- **Grendel-scan** - Web site domain expired, but awesome vulnerability scanner

- **Nessus 4** - Has several plugins for CGI scanning, and a built-in CGI scanner that works great for identifying web apps
/** End */

- Presentations: [http://pauldotcom.com/presentations.html](http://pauldotcom.com/presentations.html)
  - Special category just for this webcast series!
- Email: psw@pauldotcom.com