Hacking Intranet Websites from the Outside

"JavaScript malware just got a lot more dangerous"

Black Hat (USA) - Las Vegas

08.03.2006

Jeremiah Grossman (Founder and CTO)
T.C. Niedzialkowski (Sr. Security Engineer)

WhiteHat Security

WhiteHat Sentinel - Continuous Vulnerability Assessment and Management Service for Websites.

Jeremiah Grossman (Founder and CTO)

- Technology R&D and industry evangelist
- Co-founder of the Web Application Security Consortium (WASC)
- Former Yahoo Information Security Officer

T.C. Niedzialkowski (Sr. Security Engineer)

- Manages WhiteHat Sentinel service for enterprise customers
- extensive experience in web application security assessments
- key contributor to the design of WhiteHat's scanning technology.

WHITEHAT™ SECURITY

Assumptions of Intranet Security

Doing any of the following on the internet would be crazy, but on intranet...

- Leaving hosts unpatched
- Using default passwords
- Not putting a firewall in front of a host

Is OK because the perimeter firewalls block external access to internal devices.

WHITEHAT™ SECURITY

Assumptions of Intranet Security

WRONG!

WHITEHAT™ SECURITY

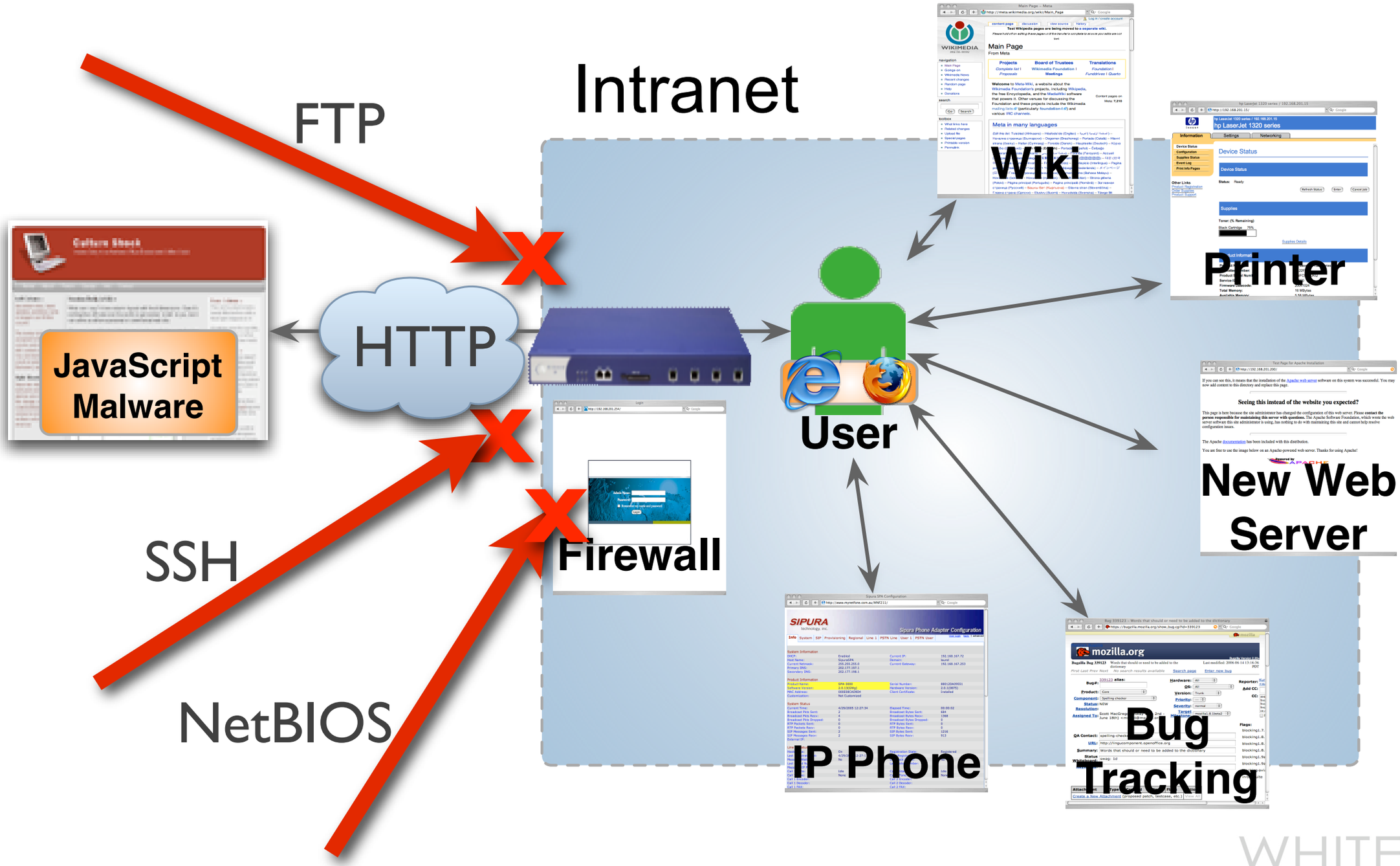
Everything is web-enabled

routers, firewalls, printers, payroll systems, employee directories, bug tracking systems, development machines, web mail, wikis, IP phones, web cams, host management, etc etc.

WHITEHAT™ SECURITY

Intranet users have access

To access intranet websites, control a user (or the browser) which is on the inside.



WHITEHAT™ SECURITY

Hacking the Intranet

JavaScript Malware

Gets behind the firewall to attack the intranet.

operating system and browser independent

special thanks to...
RSnake
http://ha.ckers.org/

WHITEHAT™ SECURITY

The following examples DO NOT use any well-known or un-patched web browser vulnerabilities. The code uses clever and sophisticated JavaScript, Cascading Style-Sheet (CSS), and Java Applet programming. Technology that is common to all popular web browsers. Example code is developed for Firefox 1.5, but the techniques should also apply to Internet Explorer.

Contracting JavaScript Malware

1. website owner embedded JavaScript malware.

2. web page defaced with embedded JavaScript malware.

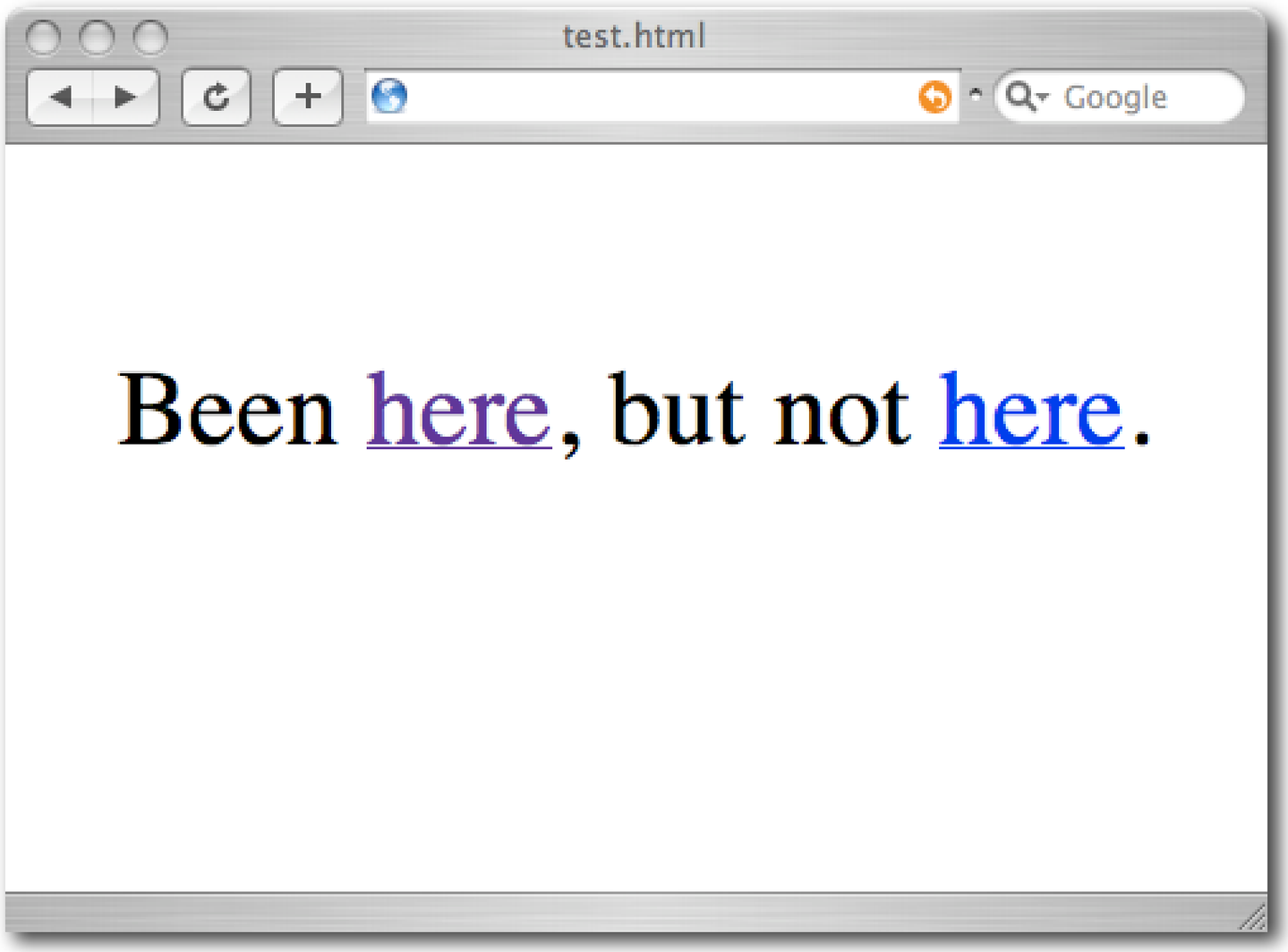
3. JavaScript Malware injected into into a public area of a website. (persistent XSS)

4. clicked on a specially-crafted link causing the website to echo JavaScript Malware. (non-persistent XSS)

WHITEHAT™ SECURITY

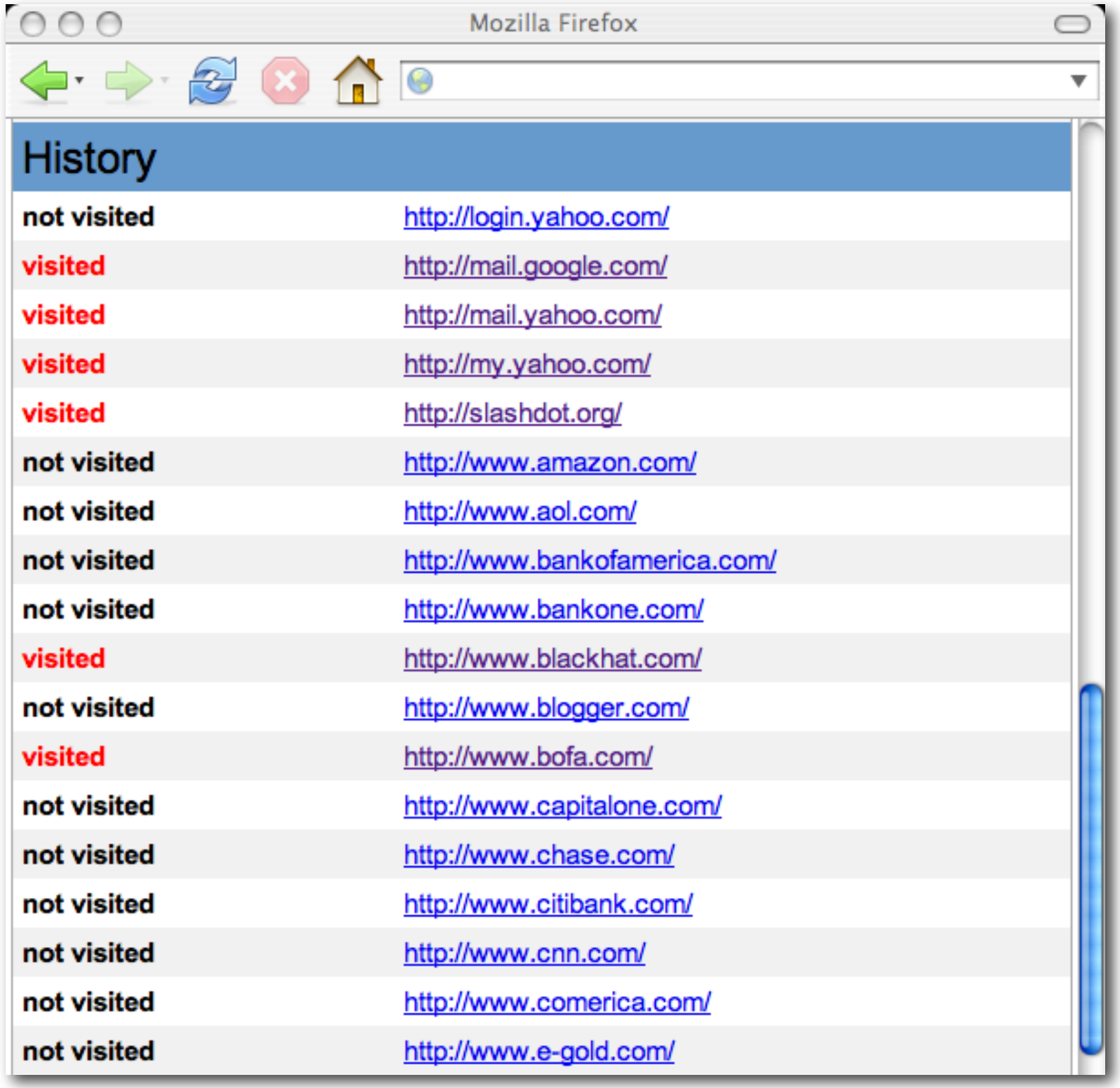
Stealing Browser History

JavaScript can make links and has access to CSS APIs



See the difference?

Cycle through the most popular websites



WHITEHAT™ SECURITY

NATed IP Address

IP Address Java Applet

This applet demonstrates that any server you visit can find out your real IP address if you enable Java, even if you're behind a firewall or use a proxy.

Lars Kindermann
http://reglos.de/myaddress/

Send internal IP address where JavaScript can access it

```
<APPLET CODE="MyAddress.class">
<PARAM NAME="URL" VALUE="demo.html?IP=">
</APPLET>
```

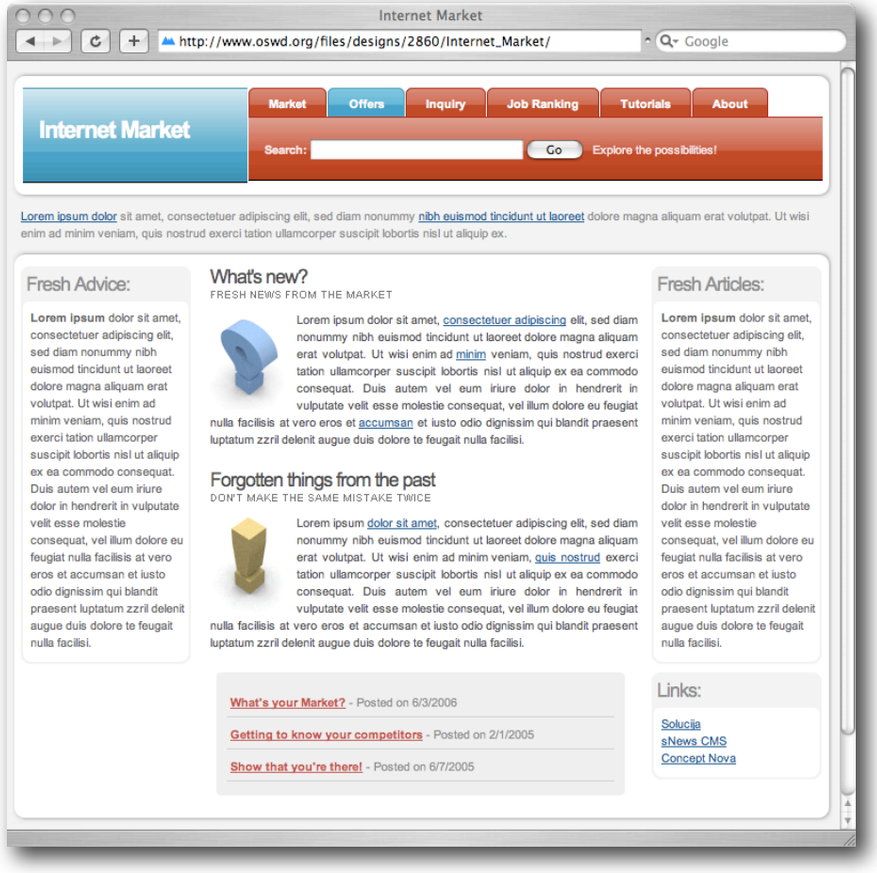
If we can get the internal subnet great, if not, we can still guess for port scanning...

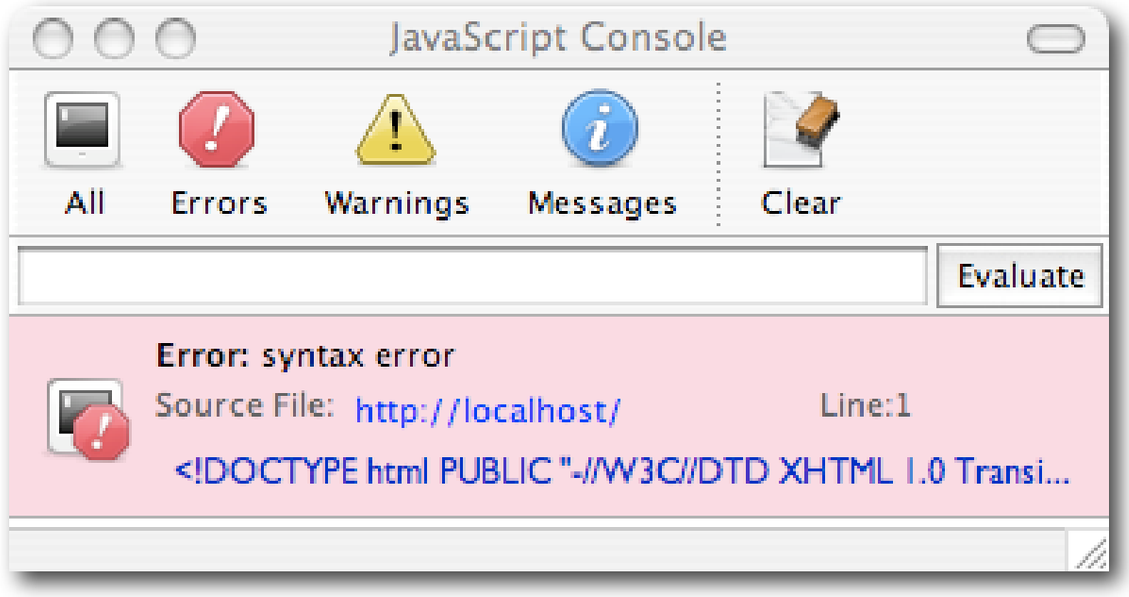
JavaScript Port Scanning

We can send HTTP requests to anywhere, but we can't access the response (same-origin policy). So how do we know if a connection is made?

<SCRIPT SRC="http://192.168.1.100/"></SCRIPT>

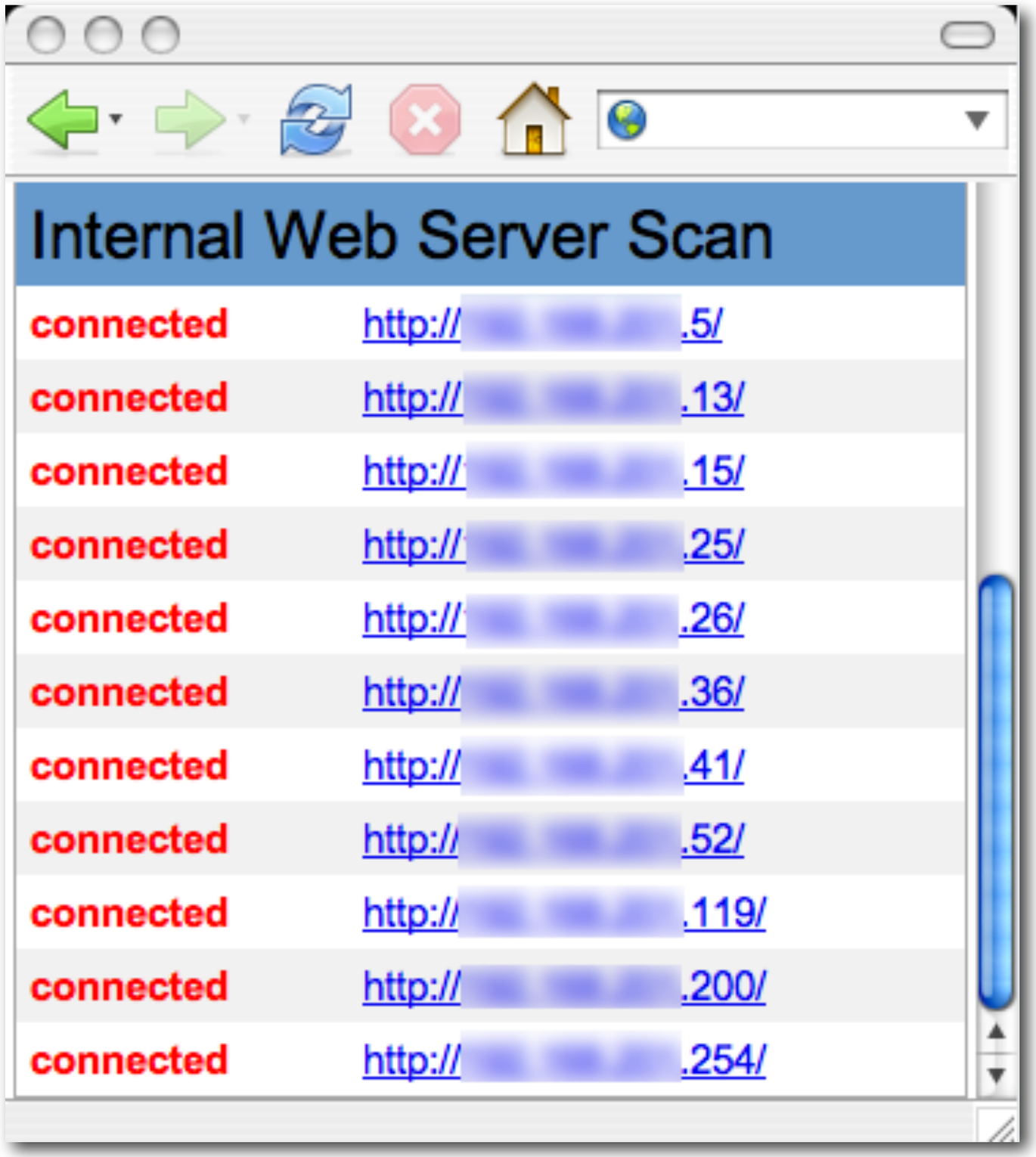
If a web server is listening on 192.168.1.100, HTML will be returned causing the JS interpreter to error.

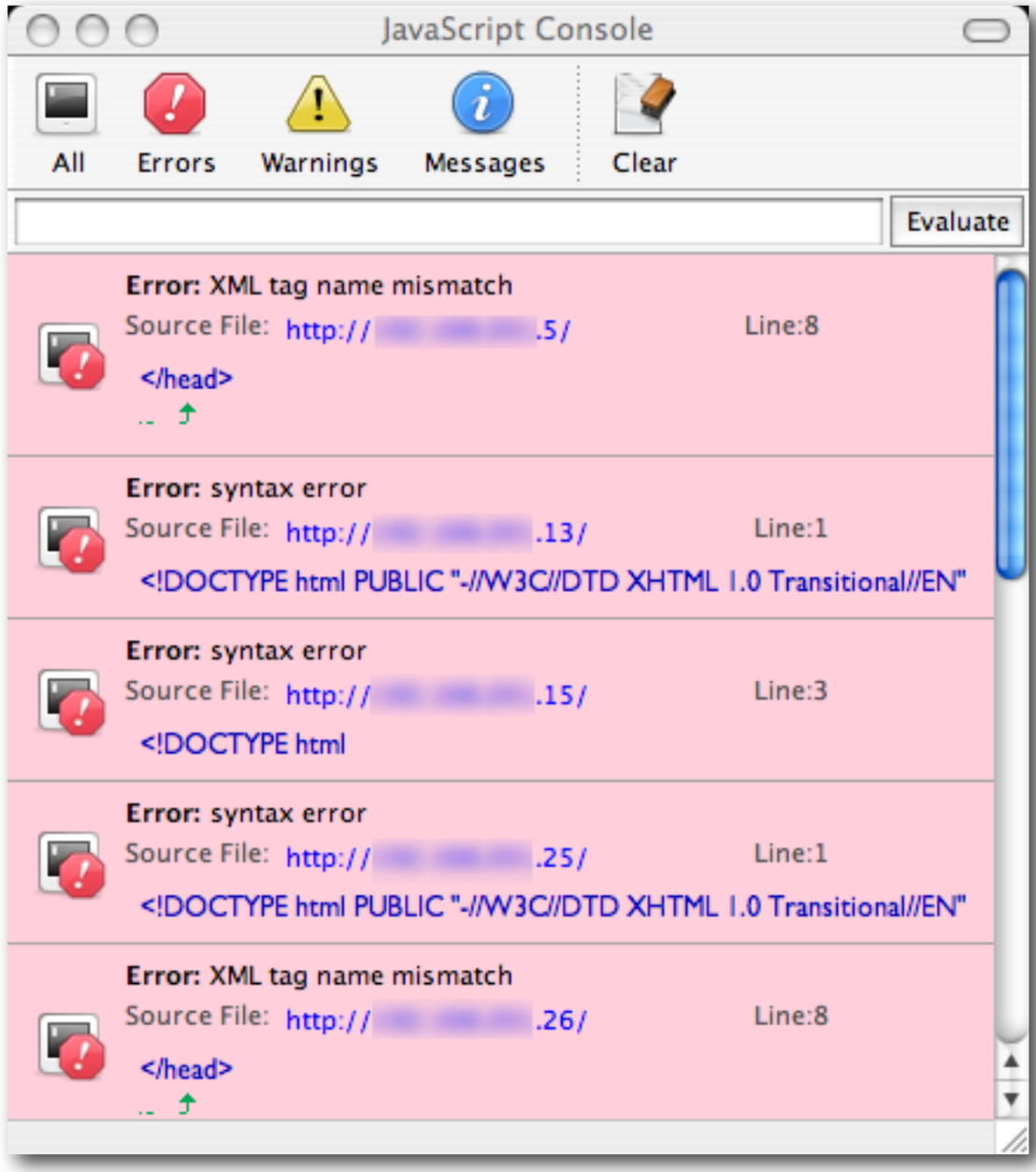




CAPTURE THE ERROR!

WHITEHAT™ SECURITY





WHITEHAT™ SECURITY

Blind URL Fingerprinting

There is a web server listening, but can't see the response, what is it?

Many web platforms have URL's to images that are unique.

Apache Web Server
/icons/apache_pb.gif

HP Printer
/hp/device/hp_invent_logo.gif

PHP Image Easter eggs
/?=PHPE9568F36-D428-11d2-A769-00AA001ACF42

Use OnError!

Cycle through unique URL's using Image DOM objects

If the onerror event does NOT execute, then it's the associated platform.

Technically, CSS and JavaScript pages can be used for fingerprinting as well.

WHITEHAT™ SECURITY

Browser Zombies

http://hacker/cgi-bin/zombie.pl?action=monitor

Search Results | Browser Zombies

Browser Zombies

Session	External IP	Internal IP
8898	209.11.127.13	192.168.201.204

Command	Re-direct Send
User-Agent	Mozilla/5.0 (Macintosh; U; PPC Mac O rv:1.8.0.4) Gecko/20060508 Firefox/1.
Screen: 1280x854 - Pixel: 32 - Color: 32	
Keystrokes	
Time	Tue Jun 27 09:14:29 2006

History

- http://www.blackhat.com/
- http://www.wellsfargo.com/
- http://mail.google.com/
- http://www.myspace.com/
- http://slashdot.org/
- http://www.yahoo.com/

Done

Browser Zombies

http://hacker/cgi-bin/zombie.pl?action=monitor

Search Results | Browser Zombies

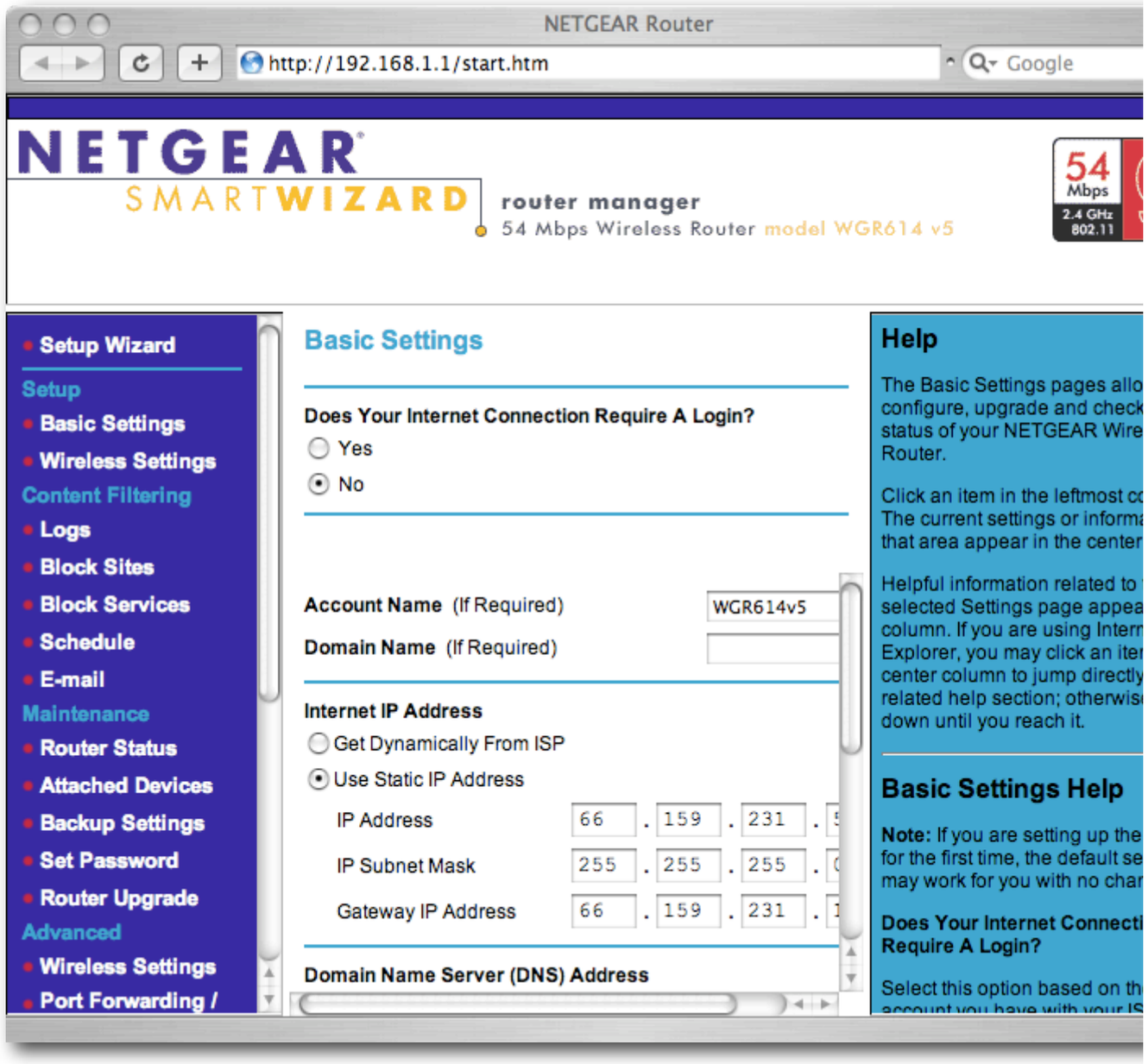
- http://www.amazon.com/
- http://www.cnn.com/
- http://mail.yahoo.com/
- http://www.myspace.com/
- http://www.usbank.com/
- http://www.bofa.com/

Internal Web Servers

- http://192.168.201.13/
- http://192.168.201.5/
- http://192.168.201.15/
- http://192.168.201.25/
- http://192.168.201.26/
- http://192.168.201.41/
- http://192.168.201.36/
- http://192.168.201.52/
- http://192.168.201.43/

DSL Wireless/Router Hacking

Login, if not already authenticated



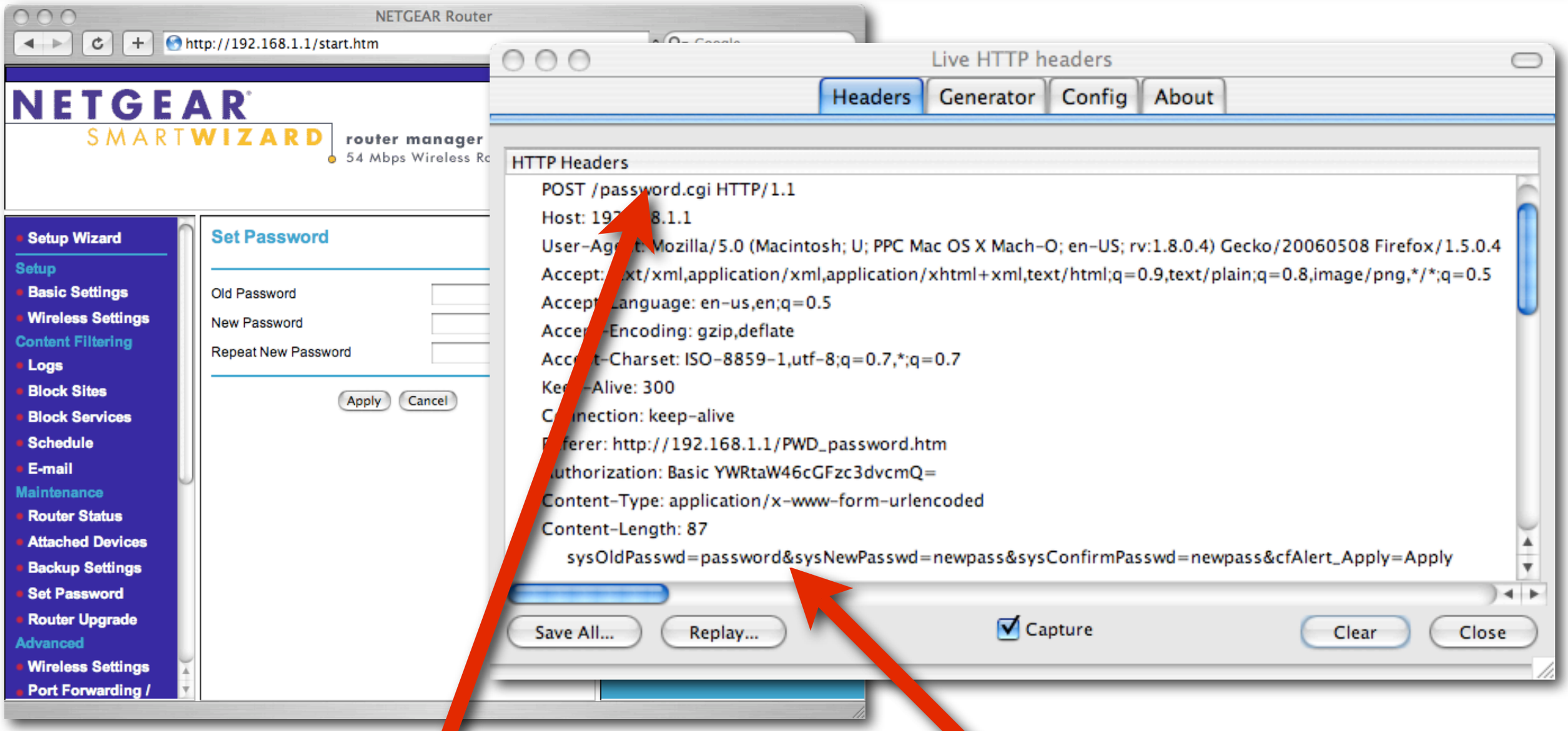
D-Link		
DI-514	admin	(blank)
DI-524	admin	(blank)
DI-614+	admin	(blank)
DI-624	admin	(blank)
DI-624+	admin	(blank)
DI-714	admin	(blank)
DI-724P+	admin	(blank)
DI-784	admin	(blank)
DWL-2100AP	admin	(blank)
DWL-G700AP	admin	(blank)
Dell		
TrueMobile 2300	admin	admin
Gateway		
WGR-200	admin	admin
WGR-250	admin	admin
Linksys		
BEFW11S4	(blank)	admin
WAP11	(blank)	admin
WAP54G	(blank)	admin
WRK54G	(blank)	admin
WRT54G	(blank)	admin
WRT54GS	(blank)	admin
WRT55AG	(blank)	admin
WRV54G	admin	admin
Microsoft		
MN-500	(blank)	admin

Factory defaults are handy!

http://admin:password@192.168.1.1/

WHITEHAT™ SECURITY

Change the password

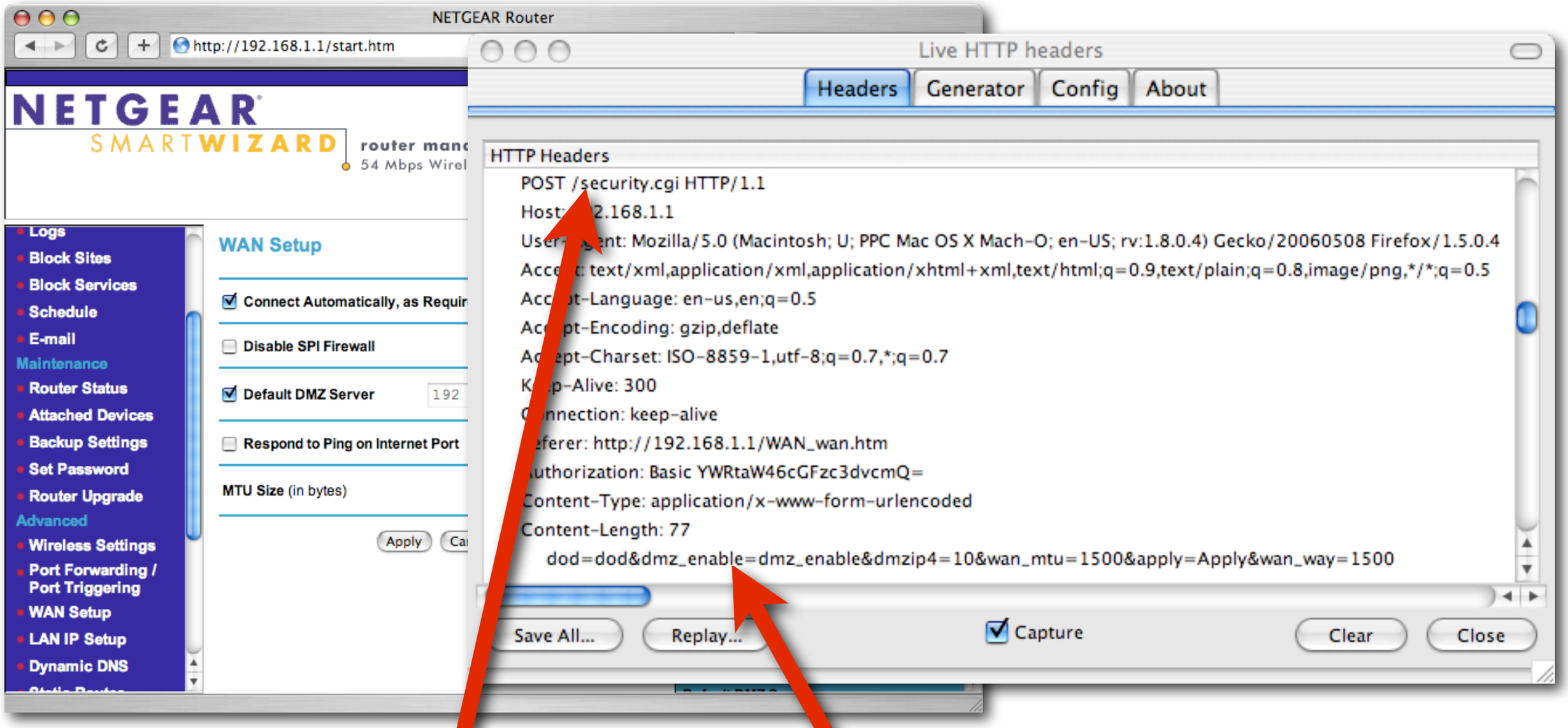


POST to GET

/password.cgi?
sysOldPasswd=password&sysNewPasswd=newpass&sysConfirmPasswd=newpass&cfAlert_Apply=Apply

WHITEHAT™ SECURITY

DMZ Hacking

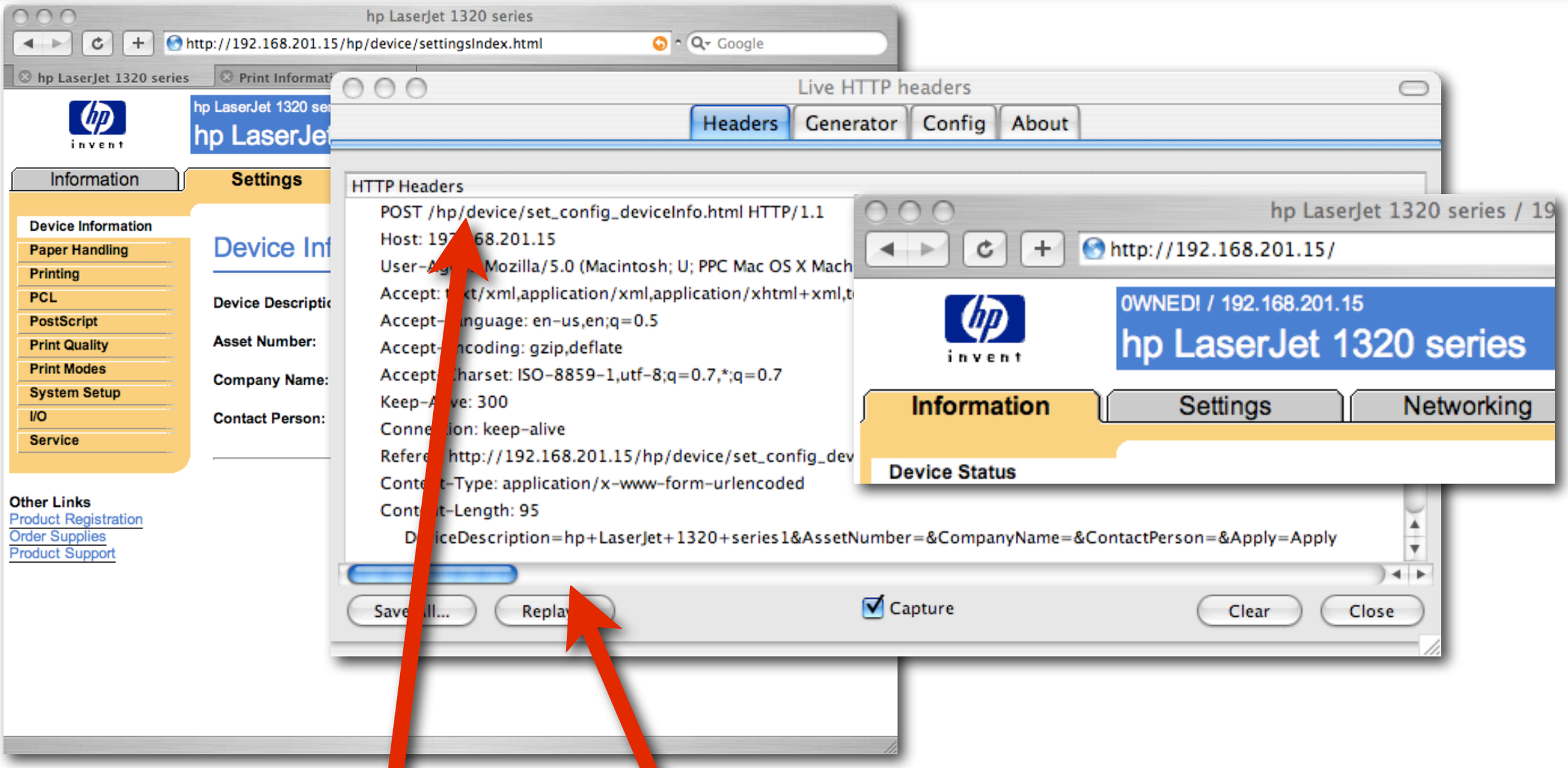


POST to GET

/security.cgi?
dod=dod&dmz_enable=dmz_enable&dmzip1=192&dmzip2=168&dmzip3=1&dmzip4=9&wan_mtu=1500&apply=Apply&wan_way=1500

WHITEHAT™ SECURITY

Network Printer Hacking



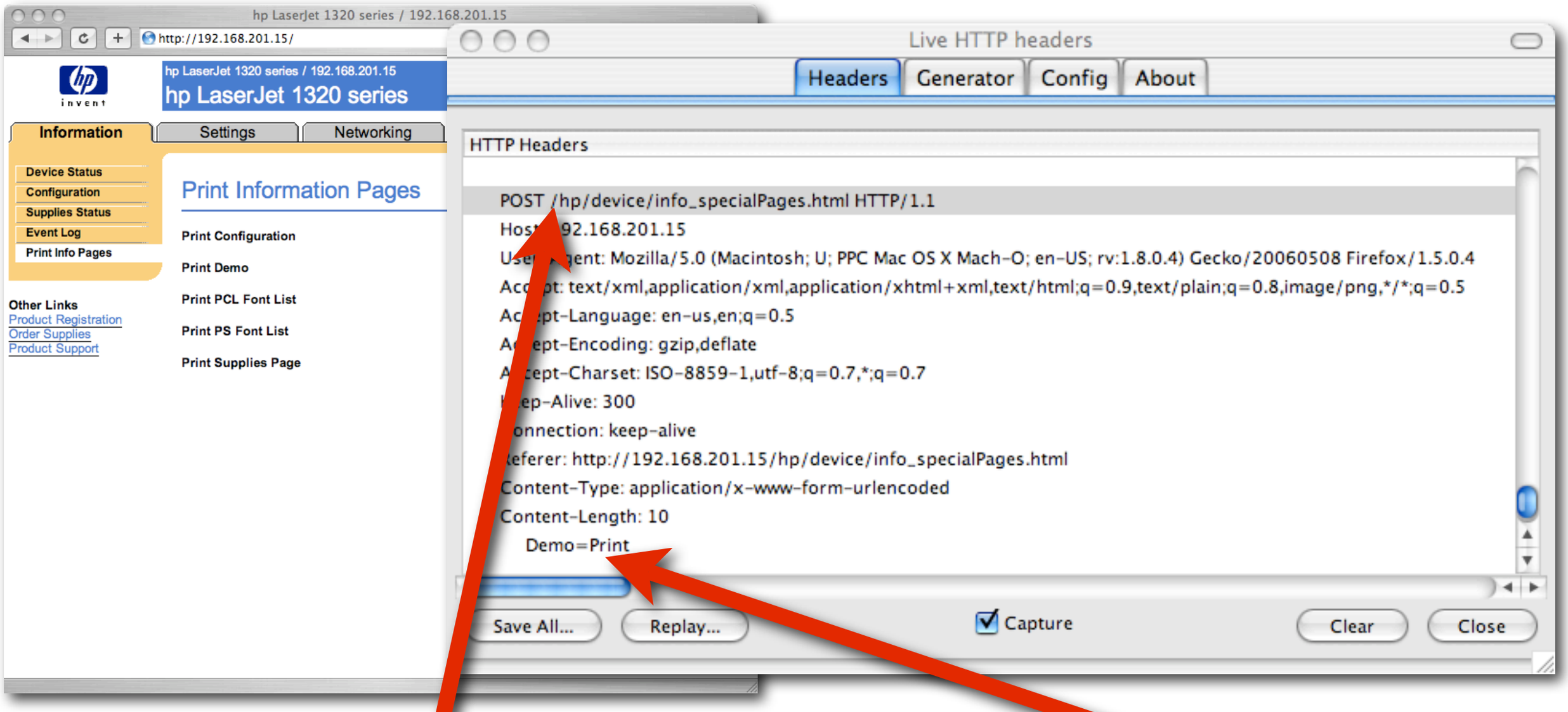
POST to GET

/hp/device/set_config_deviceInfo.html?DeviceDescription=0WNED!
&AssetNumber=&CompanyName=&ContactPerson=&Apply=Apply

WHITEHAT™ SECURITY

Network Printer Hacking

Auto-Fire Printer Test Pages



POST to GET /hp/device/info_specialPages.html?Demo=Print

WHITEHAT™ SECURITY

More Dirty Tricks

- black hat search engine optimization (SEO)
- Click-fraud
- Distributed Denial of Service
- Force access of illegal content
- Hack other websites (IDS sirens)
- Distributed email spam (Outlook Web Access)
- Distributed blog spam
- Vote tampering
- De-Anonymize people
- etc.

Once the browser closes there is little trace of the exploit code.

WHITEHAT™ SECURITY

Anybody can be a victim on any website

Trusted websites are hosting malware.

Cross-Site Scripting (XSS) and Cross-Site Request Forgery vulnerabilities amplify the problem.

WHITEHAT™ SECURITY

XSS Everywhere

Attacks the user of a website, not the website itself. The most common vulnerability.

SecurityFocus cataloged over 1,400 issues.

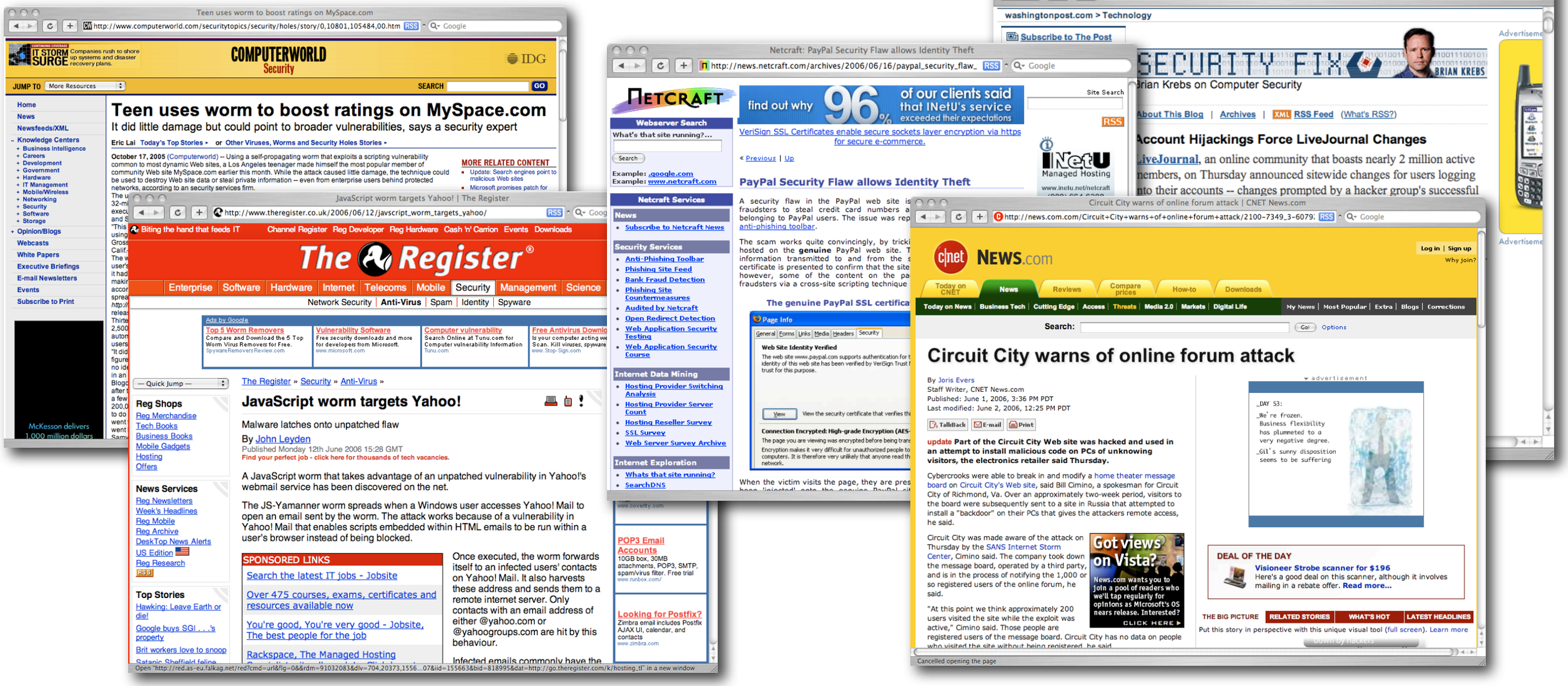
WhiteHat Security has Identified over 1,500 in custom web applications. 8 in 10 websites have XSS.

Tops the Web Hacking Incident Database (WHID)

http://www.webappsec.org/projects/whid/

WHITEHAT™ SECURITY

Exploited on popular websites



Exploitation Leads to website defacement, session hi-jacking, user impersonation, worms, phishing scams, browser trojans, and more...

WHITEHAT™ SECURITY

CSRF, even more widespread

A cross-site request forgery (CSRF or XSRF), although similar-sounding in name to cross-site scripting (XSS), is a very different and almost opposite form of attack. Whereas cross-site scripting exploits the trust a user has in a website, a **cross-site request forgery exploits the trust a website has in a user** by forging the enactor and making a request appear to come from a trusted user.

Wikipedia
http://en.wikipedia.org/wiki/Cross-site_request_forgery

No statistics, but the general consensus is **just about every piece of sensitive website functionality is vulnerable.**

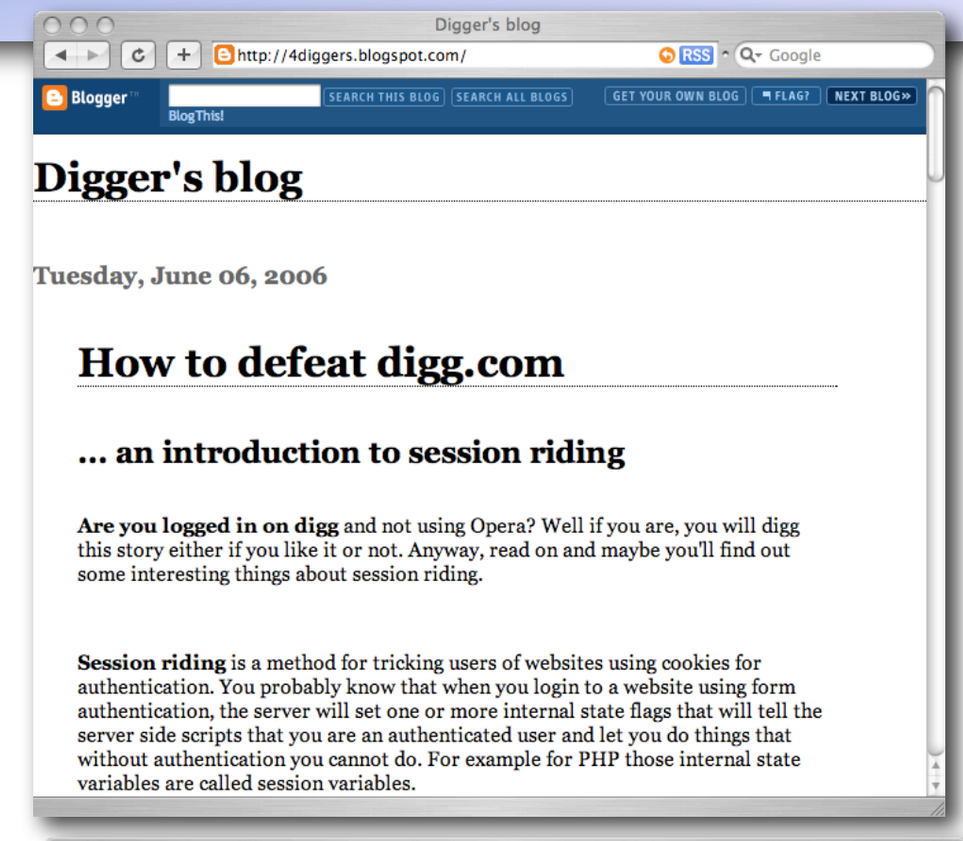
WHITEHAT™ SECURITY

CSRF HACK EXAMPLES

A STORY THAT DIGGS ITSELF

USERS LOGGED-IN TO DIGG.COM VISITING HTTP://4DIGGERS.BLOGSPOT.COM/ WILL AUTOMATICALLY DIGG THE STORY

http://ha.ckers.org/blog/20060615/a-story-that-diggs-itself/



COMPROMISING YOUR GMAIL CONTACT LIST

CONTACT LIST AVAILABLE IN JAVASCRIPT SPACE. <SCRIPT SRC=HTTP://MAIL.GOOGLE.COM/MAIL/?_URL_SCRUBBED>

http://www.webappsec.org/lists/websecurity/archive/2006-01/msg00087.html

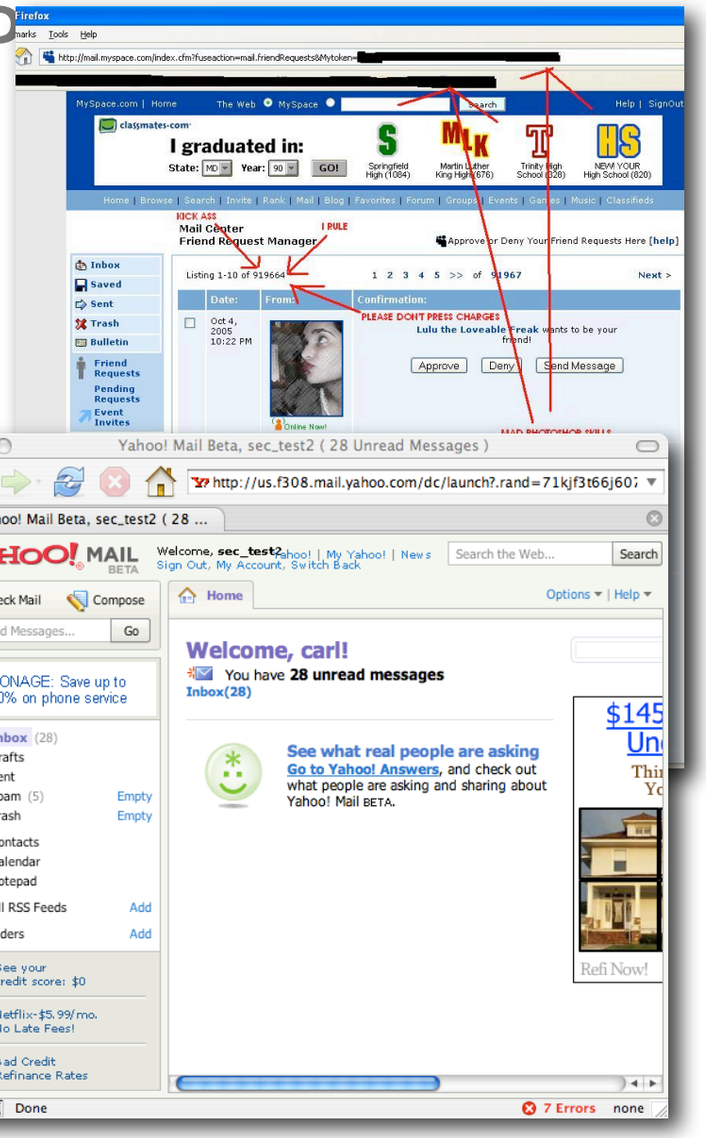
Worms

MySpace (Samy Worm) - first XSS worm

24 hours, 1 million users affected

- logged-in user views samys profile page, embedded JavaScript malware.
- Malware ads samy as their friend, updates their profile with "samy is my hero", and copies the malware to their profile.
- People visiting infected profiles are in turn infected causing exponential growth.

http://namb.la/popular/tech.html



Yahoo Mail (JS-Yamanner)

- User receives a email w/ an attachment embedded with JavaScript malware.
- User opens the attachment and malware harvesting @yahoo.com and @yahoogroups.com addresses from contact list.
- User is re-directed to another web page.

http://ha.ckers.org/blog/20060612/yahoo-xss-worm/

CROSS-SITE SCRIPTING WORMS AND VIRUSES
"The Impending Threat and the Best Defense"
http://www.whitehatsec.com/downloads/WHXSSThreats.pdf

WHITEHAT™ SECURITY

How to protect yourself

Or at least try

WHITEHAT™ SECURITY

Not going to work

Useful for other threats, but not against JavaScript malware.

~~Patching and anti-virus~~

~~Corporate Web Surfing Filters~~

~~Security Sockets Layer (SSL)~~

~~Two Factor Authentication~~

~~Stay away from questionable websites~~

WHITEHAT™ SECURITY

Better End-User Solutions

- Be suspicious of long links, especially those that look like they contain HTML code. When in doubt, type the domain name manually into your browser location bar.
- no web browser has a clear security advantage, but we prefer Firefox. For additional security, install browser add-ons such as NoScript (Firefox extension) or the Netcraft Toolbar.
- When in doubt, disable JavaScript, Java, and Active X prior to your visit.

WHITEHAT™ SECURITY

We Need More Browser Security

- Mozilla (Firefox), Microsoft and Opera development teams must begin formalizing and implementing **Content-Restrictions.**

Sites would define and serve content restrictions for pages which contained untrusted content which they had filtered. If the filtering failed, the content restrictions may still prevent malicious script from executing or doing damage.

Gervase Markham

http://www.gerv.net/security/content-restrictions/

- Mozilla (Firefox) developers, please implement **httpOnly**. It's been around for years!

Fixing XSS and CSRF

Preventing websites from hosting JavaScript Malware

- rock solid Input Validation. This includes URL's, query strings, headers, post data, etc.

filter HTML from output

```
$data =~ s/(<|>|\"|\'|\(|\)|:)/'&#'.ord($1).';'/sge;
or
$data =~ s/([^\w])/'&#'.ord($1).';'/sge;
```

Text

- Protect sensitive functionality from CSRF attack. Implement session tokens, CAPTCHAs and ~~HTTP Referer checking.~~

WHITEHAT™ SECURITY

Finding and Fixing

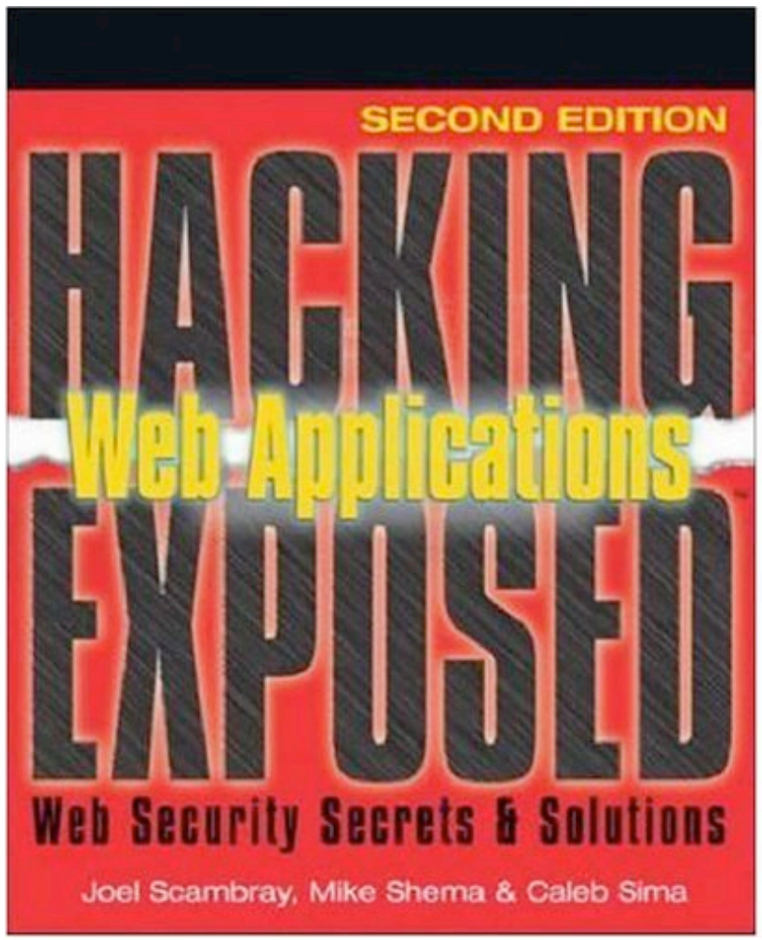
- Find your vulnerabilities before the bad guys do. Comprehensive assessments combine automated vulnerability scanning and expert-driven analysis.

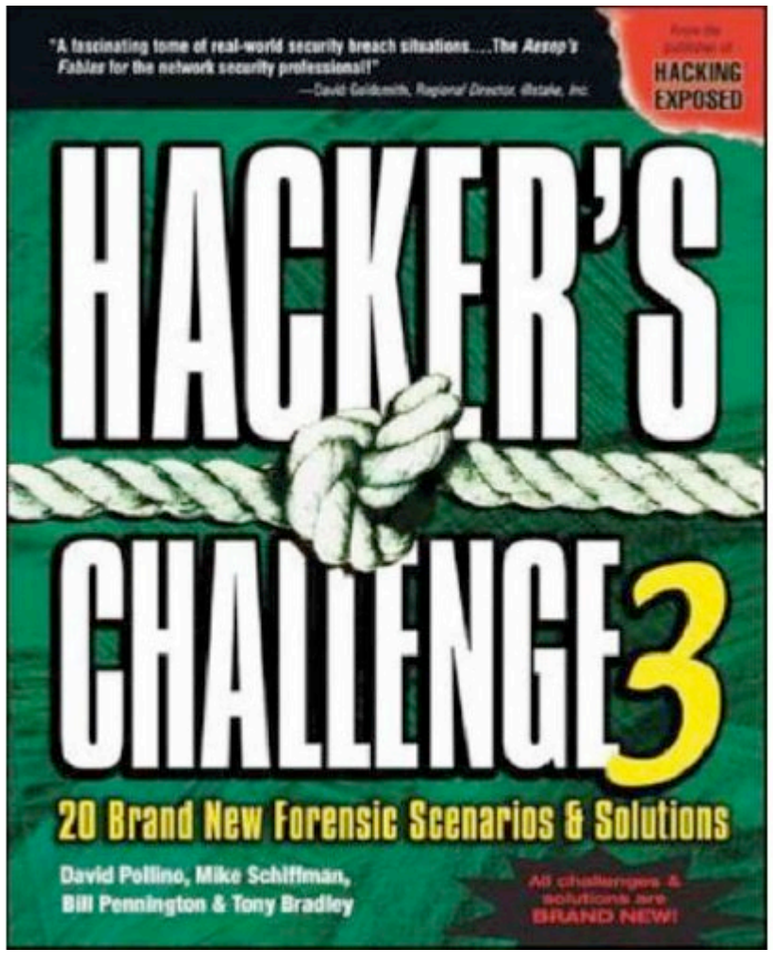
- When absolutely nothing can go wrong with your website, consider a web application firewall (WAF). Defense-in-Depth (mod_security, URL Scan, SecureIIS).

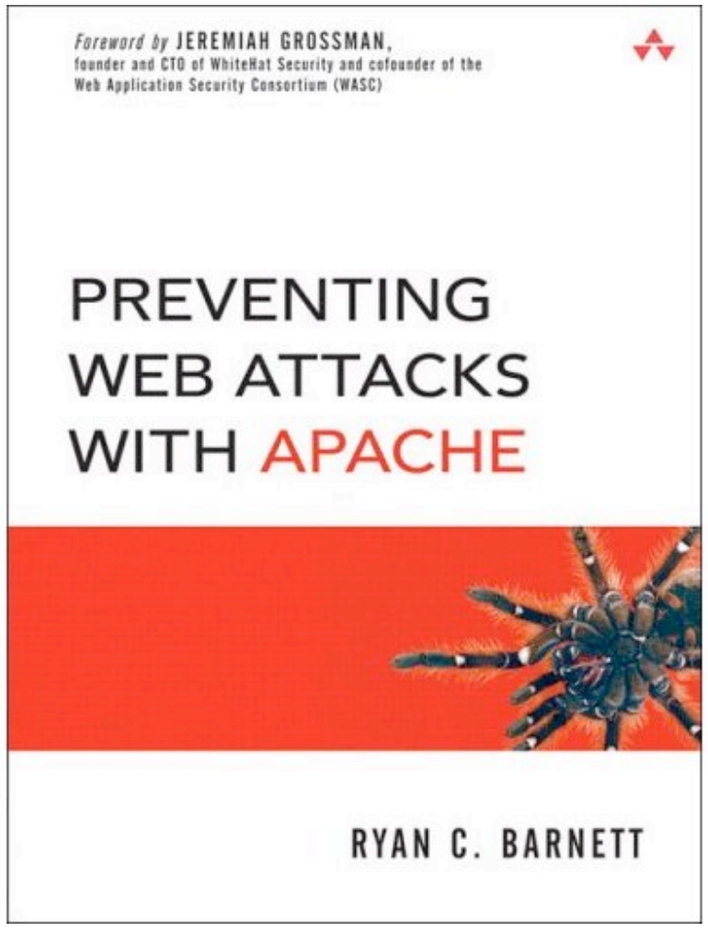
- harden the intranet websites. They are no longer out of reach. Patch and change default password.

WHITEHAT™ SECURITY

Recommended Reading







WHITEHAT™ SECURITY

THANK YOU!

Jeremiah Grossman
Founder and Chief Technology Officer
jeremiah@whitehatsec.com

T.C. Niedzialkowski
Sr. Security Engineer
tc@whitehatsec.com

For more information about WhiteHat Security,
please call 408.492.1817 or visit our website,
www.whitehatsec.com

WHITEHAT™ SECURITY