Network Security

Web Server

Vulnerability Scanning

Module 4

Keith A. Watson, CISSP, CISA
IA Research Engineer, CERIAS
kaw@cerias.purdue.edu
Web Server Vulnerability Scanning

Definition: Scanning a remote web server for specific and known configuration errors, vulnerabilities or weaknesses.
Web Scanning: Overview

- Web Server Vulnerability Scanning
- Common Web Vulnerabilities
- Nikto
  - Installation
  - Options
  - Scan, mutation, output modes
- Exercises
Web Scanning: Why scan?

- Web servers provide a static and external access point for attackers
  - Your web site is always on and “located” in the same place (i.e. FQDN is unchanged)
- Firewalls provide limited or no protection to poorly written web applications and misconfigured web servers
  - All traffic for web-based attacks goes through permitted HTTP/HTTPS connections
Web Scanning: Why scan?

- We want to see vulnerabilities as the attacker will see them
- We want to know what needs to be fixed before we go into production
- We want to periodically verify that the configuration is the same as last time
- We want to check for newly discovered vulnerabilities
Web Scanning: What do we scan?

1. Primary and external web sites and apps
2. Internal use only web sites and apps
3. Web applications under development
4. Network devices with web-based interfaces
5. Printers with web-based interfaces
6. Anything else that is web-based
Web Scanning: Tools available

- Nikto 2
- iPod Touch / iPhone tools
  - None discovered
- Various commercial (and expensive) tools
  - Milescan Web Security Auditor
  - SPI Dynamics WebInspect
  - Acunetix WVS
  - Rational AppScan
- BackTrack
  - Many tools available
Web Scanning: Common Web Vulnerabilities

- **SQL Injection**
  - Application does not correctly filter user input
  - User input is used in a SQL query
  - Attacker can manipulate the query to access sensitive information or alter the data
  - Has been used to capture username and password tables
Web Scanning:
Common Web Vulnerabilities

- Cross-Site Scripting (XSS)
  - Application does not correctly filter user input
  - User input can inject HTML, Javascript onto page
  - Other users view a trusted page with malicious code embedded on it
  - Common routes of attack comment forms, web forums, web submission systems, blogs
Web Scanning: Common Web Vulnerabilities

- PHP Include
  - Common mistake in PHP development
  - Programmer uses an include statement to load a page from a URL using an internal variable
  - The value of the variable can be changed to point to a remote site
  - The PHP interpreter loads the malicious remote page and executes it locally on web server
Web Scanning: Common Web Vulnerabilities

• Information Leaks
  • Common mistakes made by webmasters and administrators
  • Open lists of files in web server directories (index)
  • Username and password files downloadable
  • Scripts that list files outside of the document root
  • Old scripts with incorrect file extensions (like .old)
  • Excellent way to find out how the server is configured and ways to compromise it
Web Scanning: Common Web Vulnerabilities

● Privilege escalation
  • Exploiting vulnerabilities in code to raise privilege
  • A normal user can become the editor or admin
  • An anonymous user can gain registered user access
  • “Hidden” variable used to exploit problems in code
Web Scanning: Nikto 2 summary

- Nikto looks for each of the common web application security issues
- Open Source (GPL license)
- It is a command line tool
- Written in Perl 5
- Has a database of known issues
  - Can be updated from developer’s site
- Best used from BackTrack
  - Unless you want to install it and all its dependencies including Perl and libraries
Web Scanning: Trivia question

- Where does the word “Nikto” come from?

  (Bonus points if you can cite two sources.)
Web Scanning: Trivia answer

• First used in **The Day the Earth Stood Still** movie from 1951 (Director: Robert Wise)
  - “Klaatu Barada Nikto” was the phrase used to prevent Gort the robot from destroying the Earth

• Second time used in **The Army of Darkness** a B movie from 1992 (Director: Sam Raimi)
  - “Klaatu Barada Nikto” was the phrase used to “disable” the Necronomicon before picking it up
  - Ash (Bruce Campbell) bumbles the phrase and awakens the Army of Darkness
Web Scanning: Nikto 2 features

- Looks for more than 3,500 potentially dangerous files and CGI scripts
- Looks for over 900 web server versions
  - Looks for version-specific issues on over 250
- Has anti-IDS features
  - Can test your IDS for detection
- Fingerprint web servers through favicon.ico’s
- Does a 404 (file not found) check for various file types
Web Scanning: More Nikto 2 features

- Reduces false positives through multiple source of evidence
  - Looks at headers, page content, content hashing
- Creates reports in HTML, XML, CSV, TXT
  - Also has template engine for report customization
- Scan tuning for inclusion/exclusion of plugins
- Periodic updates with the latest plugins
Web Scanning: Nikto installation

• Can work on Windows, Mac OS X, Linux

• General requirements
  • Perl 5 (installed on Mac OS X, available on others)
  • Net::SSLeay Perl module for SSL (Mac OS X)
  • Command line use

• Windows
  • Perl 5 can be installed from ActiveState or Cygwin
  • Install Net::SSLeay using CPAN tool (needs compiler from Cygwin)

• Linux
  • Install Nikto through package manager
Web Scanning: Nikto operation

- Download and extract nikto archive
- Open a terminal or DOS window
- cd into the nikto directory
- execute commands from the nikto directory
  - $ perl nikto.pl -host <target>

- Or, run Nikto from BackTrack
Web Scanning: Nikto updates

- Periodically nikto plugins and databases are updated
- If you have a net connection you can update nikto manually or through nikto
- You can download updates from the nikto web site: http://cirt.net/nikto/UPDATES/2.1.0/
  - Look for “Plugins & DBs” link on the main site
- You can update using nikto
  - perl nikto.pl -update
Web Scanning: Nikto updates

- Updating from nikto:
  
  ```
  $ perl ./nikto.pl -update
  + Retrieving 'db_favicon'
  + Retrieving 'nikto_headers.plugin'
  + Retrieving 'db_headers'
  + Retrieving 'db_server_msgs'
  + Retrieving 'db_tests'
  + Retrieving 'nikto_subdomain.plugin'
  + Retrieving 'CHANGES.txt'
  + www.cirt.net message: Please submit your bugs!!
  ```
Web Scanning: Nikto basic mode

- For the default testing mode:
  - perl nikto.pl -host <target>
  - perl nikto.pl -host <URL>

- For a non-standard port:
  - perl nikto.pl -host <target> -p <port>
  - perl nikto.pl -host <target> -p 80,443,8000,8080

- Forcing an SSL check (tests only for SSL)
  - perl nikto.pl -host <target> -ssl
Web Scanning: Nikto special modes

• Find only (-findonly)
  • Checks for a web server and reports header information

• Configurable CGI scan (-Cgidirs <flag>)
  • “none” - do not scan CGI directories
  • “all” - scan all CGI directories (default)
  • list directories to scan: “/cgi-test/”

• Username and password (-id <id:password>)
  • Specifies the user and password for Basic Authentication protected web sites
Web Scanning:
Nikto special modes

- No 404 error checking (-no404)
  - Skips checks on 404 error pages
  - Speeds test on slow links
  - Generates more false positives

- Slow down checks (-Pause <seconds>)
  - Adds a delay between tests

- Virtual Host (-vhost)
  - Specify the virtual host to scan in the header
Web Scanning: 
Nikto mutations

- Mutations are away to expand testing
- Guesses are made in order to discover info
  - File and directory names
  - Usernames (/~user)
- Mutations can be expensive in number of requests, scanning system load, network traffic
- See the documentation for more info
Web Scanning: Nikto output options

- Write to a file (-output <file>)
- Display option (-Display <option>)
  - “1” - Show redirects
  - “2” - Show cookies received
  - “3” - Show 200/OK responses
  - “4” - Show URLs requiring authentication
  - “D” - debug output
  - “V” - verbose output
- Format option (-Format <type>); use -output
  - “csv”, “htm”, “xml” or “txt” (default)
Web Scanning: Nikto scan tuning

• Scan tuning selects specific tests to run
• Tuning option (-Tuning <options>)
  • “0” - File upload
  • “1” - Interesting files based on log data
  • “2” - Misconfiguration or default files
  • “3” - Information disclosure
  • “4” - Injection attacks (XSS, Scripts, HTML)
  • “5” - Remote file retrieval - inside web root
  • “6” - Denial of service
  • “7” - Remote file retrieval - server wide
Web Scanning: Nikto scan tuning

More Tuning options (-Tuning <options>)
- “8” - File upload
- “9” - SQL injection
- “a” - Authentication bypass
- “b” - Software identification
- “c” - Remote source inclusion
- “x” - reverse the option

Examples:
- -Tuning 12378
- -Tuning 58xb
Web Scanning: Nikto output example

$ perl ./nikto.pl -host npdn-infosec.cerias.purdue.edu
- Nikto v2.1.0/2.1.0

+ Target IP: 128.10.253.66
+ Target Hostname: npdn-infosec.cerias.purdue.edu
+ Target Port: 80

+ Server: Apache
+ No CGI Directories found (use '-C all' to force check all possible dirs)
+ 3582 items checked: 0 item(s) reported on remote host

+ 1 host(s) tested
Web Scanning:
more Nikto output

+ Server: Apache/2.2.11 (Unix) mod_ssl/2.2.11 OpenSSL/0.9.8i DAV/2 PHP/5.2.10
+ OSVDB-0: Allowed HTTP Methods: GET, HEAD, POST, OPTIONS, TRACE
+ OSVDB-877: HTTP TRACE method is active, suggesting the host is vulnerable to XST
+ OSVDB-0: mod_ssl/2.2.11 OpenSSL/0.9.8i DAV/2 PHP/5.2.10 - mod_ssl 2.8.7 and lower are vulnerable to a remote buffer overflow which may allow a remote shell (difficult to exploit). [http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2002-0082](http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2002-0082), OSVDB-756.
+ OSVDB-0: Apache/2.2.11 appears to be outdated (current is at least Apache/2.2.14). Apache 1.3.41 and 2.0.63 are also current.
+ OSVDB-0: mod_ssl/2.2.11 appears to be outdated (current is at least 2.8.31) (may depend on server version)
+ OSVDB-3092: /img/: This may be interesting...
+ 3582 items checked: 6 item(s) reported on remote host
+ End Time: 2009-11-11 13:05:03 (73 seconds)

+ 1 host(s) tested
Web Scanning: Exercise 1

- Start up BackTrack
- Find “nikto” in the BackTrack menu
- Use nikto to do a basic scan against 192.168.1.24
- What is the web server software and version?
- What issues are discovered?
- What issue might need further exploration?
Web Scanning: Exercise 2

- Using the web servers you discovered from nmap, use nikto to scan them for issues
- Include the wiki server and the WLAN router web configuration server too (192.168.1.1)
- What issues did you find?
- What servers did you find?
- What other interesting issues are there?
End of Module 4

• Questions?