

Open Source Security Tools for Information Technology Professionals

School of Professional Studies (SPS)

The City University of New York (CUNY)

Aron Trauring
Adjunct Professor
CEO, Zoteca

Class 12 December 12th, 2005

Class Agenda

6:30-6:45 Article of the Week

6:45-8:00 Network Vulnerability

8:00-8:15 Break

8:15-9:30 Nessus Lab

Instructor

Aron Trauring – Adjunct Professor, CUNY SPS / CEO, Zoteca

Email: atrauring@zoteca.com

Personal Website: <http://aronst.org/>

Zoteca Corporate Website: <http://www.zoteca.com/>

Network Vulnerability

Application Layer Vulnerability

- Firewall protects up to transport layer
- Greatest vulnerability at application layer
- Most attackers use well-known published attacks
- Tools are available to exploit vulnerabilities
- Old vulnerabilities still exploited
- Attacks on “zero-day” exploits or unpublished security holes are very rare
- Increasing barriers lowers vulnerability (criminals look for easy targets)

Barriers to Protecting Networks — Not Enough Time or Staff

- Staff cuts
- Outsourcing and off-shoring
- Information Week: Security’s Shaky State Dec. 5, 2005 http://www.informationweek.com/story/showArticle.jhtml?articleID=174900279&cid=RSSfeed_IWK_security

Barriers to Protecting Networks — Concerns About System Stability

- Some vendor patches might break stuff — e.g. Windows XP SP2
- Especially problematic for mission critical systems which need close to zero down time

Barriers to Protecting Networks — Overload

- Too many security patch notifications
- Particularly in larger organizations, cost of patching exceeds cost of software licensing and installation
- Even Microsoft servers get taken down even after they release patches

Barriers to Protecting Networks — Ignorance

- Windows automatic update might not cover all Microsoft software
- On GNU/Linux might have applications not installed through apt-get or other automated package system
- For software which is not part of some automated patching/notification system admin may not be aware of vulnerabilities

Router and Firewall Weaknesses

- Firewall set up is extremely complex — unless set up by expert likely to be mis-configured
- Time pressures and immediate access needs often lead to too much access (e.g. allow ftp and forget about it)
- Routers often run dangerous services — e.g. telnet, finger
- Routers often have default passwords which are unchanged
- Router/firewall web management interface can be exploited
- Lull into false sense of security

Web Server Exploits

- Built to provide files without authentication — high potential for exploitation
- As web servers expand functionality more likely to have vulnerabilities
- Modern web servers also execute code which is often insecure — ASP, PHP, ColdFusion etc.

Mail Server Exploits

- Exchange and Sendmail highly vulnerable
- SMTP is a simple and therefore vulnerable protocol
- Buffer overflow
- Executing shell commands via email server
- Server backdoors
- Stealing usernames and passwords

DNS Server Exploits

- Bind is a monolithic program
- Usually runs as root
- Often misconfigured
- Firewall settings for DNS often misconfigured
- DNS cache poisoning — fake DNS information stored in DNS server cache
- Can be used for phishing or delivering Trojan payloads

Database Exploits

- Web servers often interface with databases
- Authentication protocols may be insecure
- Code may be insecure and exploited via buffer overflows
- Specially crafted URLs can “inject” SQL code right into your system

The Never Expiring Password

- Database applications require authentication vis-a-vis the database
- Username and password embedded in application code
- Programmers have access to these
- In hosted settings staff at ISP has access to these
- Personnel leave but applications are never changed — far too much work

User and File Management

- Need to give user access while adhering to minimalist principles
- Laziness or lack of time leads to giving users too much access
- Windows has poor security by default, although improving in XP
- Former employee accounts are prime target for crackers — owner won't notice strange behavior
- Weak passwords another common vulnerability

Default Accounts in Hardware/Software Systems

- Routers, switches, firewalls, phone systems, alarm systems may have default passwords as well as back doors
- SNMP — Simple Management Network Protocol used in network management systems
- SNMP often comes with simple default passwords too
- `snmpwalk` — allows cracker to map network and take down devices
- SNMP buffer overflow exploits allows cracker to take over devices completely
- Default password lists can be found in seconds with a simple Google search
- Default Passwords <http://www.phenoelit.de/dpl/dpl.html>
- Default Passwords <http://www.cirt.net/cgi-bin/passwd.pl>

Blank or Weak Passwords

- No password or administrator/administrator
- Bad user passwords

Unneeded Services

- Violation of minimalist principle
- Legacy — `chargen`, `daytime`, `discard`, `echo`, `finger` and `quotd`
- “Personal” web servers
- Old code

Information Leaks

- Chatty operating systems (like Windows) gives out lot's of information about host and services
- Improperly configured DNS systems can expose your network topology
- Corporate information on public web servers expose information via Google
- Information can be exploited by cracker — e.g. username lists can simplify cracking passwords

Example System Crack

- DNS Stuff <http://www.dnsstuff.com/>
- Do DNS lookup on URL
- Click on IP to get a IPwhois — get corporate IP range
- Get sysadmin names (try brute force password attack)
- Port scan IP range (nmap)
- Vulnerability scan (nessus)
- e.g NETBIOS null sessions allowed on server — get all usernames, groups, machines, shares
- NETBIOS Null Sessions http://www.brown.edu/Research/SysAdmins/articles/netbios_null_sessions.html
- e.g. web server has buffer overflow vulnerability
- Download tool kit to exploit vulnerability (e.g. Attack Toolkit)
- Brute force attacks on passwords
- Social engineering to leverage information already have

General Principles for Limiting Vulnerabilities

- Minimalism — Install only what you need
- Minimalism — Keep only users that are currently live
- Vigilance — Install stable version on mission critical systems
- Vigilance — Automate patch installation process
- Vigilance — Firewall perimeter and hosts
- Vigilance — Use sniffers, scanners and data mining tools

Vulnerability Scanners

- Multiple and various points of attack
- Multiple vulnerabilities associated with these
- Vulnerability scanners check multiple points and their multiple weaknesses
- Requires constant signature updates

Considerations for Vulnerability Scanning

- Scan with permission
- Make sure you have current backups
- Time your scans when least disruptive
- Don't scan excessively
- Place scanner appropriately
- Use certificates so remote communication between client and server are encrypted

What Vulnerability Testing Doesn't Test

- Application logic error except for well known bugs in well known services
- Custom applications
- Social engineering attacks
- Undiscovered vulnerabilities
- Attacks that already happened

Nessus Lab

FOSS Vulnerability Scanners

- Nessus — Version 3 will be closed source
- OpenVAS — New FOSS version of Nessus
- ATK — Attack Tool Kit for Windows

Nessus Controversy

- Tenable Network Security commercial company behind Nessus
- “Customer and regulatory demands limit GPL usefulness” — multiple licensing is an option used in these cases; rarely problem
- “Want to make money” — SNORT had commercial support and signature subscription and is GPL and will remain so under Check Point ownership
- “Little contribution by community” — 50 developers have already signed up for OpenVAS project
- Past controversies — Sendmail, SSH, XFree86 have all been eclipsed by subsequent forks

Client / Server Architecture

- Server runs tests
- Client configures and controls the session
- Scanning server can sit outside your network yet be locally accessible
- Server can be at point in network which gives it maximum bandwidth or access to rest of network
- Multiple OSES supported by server regardless of client
- Web client available

Installation

- `apt-get install nessus nessusd`
- `nessus` — client
- `nessusd` — server
- Debian installation automatically creates certificate
- If not run: `nessus-mkcert`
- Set up user: `nessus-user`

Nessus User

- Can use rules to restrict what tests can run
- Can restrict which IPs can login from
- Need at least one user with no rules

Running Nessus

- `nessusd &` — start server
- `nessus` — start client

Nessus Login Page

- **Server:** use localhost if running on same host as server
- **Port:** 1241 (standard)
- **Login:** username created when setting up `nessus`
- **Password:** password created when setting up `nessus`
- If running server elsewhere provide IP address
- You can look at previous scans without being logged in

Feed Types for Plugins

- Direct Feed — paid subscription — 9692 plugins
- Registered Feed — non-GPL but free with 7 day delay — 9664 plugins
- GPL (non-registered) — 1051 plugins
- Debian/Ubuntu — 2000+ plugins; not dependent on Tenable

Plugin Categories

- AIX Local Security Checks
- Backdoors
- Brute force attacks
- CGI abuses
- CGI abuses : XSSCISCO
- Debian Local Security Checks
- Default Unix Accounts
- Denial of Service
- Fedora Local Security Checks
- Finger abuses
- Firewalls
- FreeBSD Local Security Checks
- FTP
- Gain a shell remotely

- Gain root remotely
- General
- Gentoo Local Security Checks
- HP-UX Local Security Checks
- MacOS X Local Security Checks
- Mandrake Local Security Checks
- Misc.
- Netware
- NIS
- Peer-To-Peer File Sharing
- Port scanners
- Red Hat Local Security Checks
- Remote file access
- RPC
- Service detection
- Settings
- Slackware Local Security Checks
- SMTP problems
- SNMP
- Solaris Local Security Checks
- SuSE Local Security Checks
- Useless services
- Web Servers
- Windows
- Windows : Microsoft Bulletins
- Windows : User management

Nessus Attack Scripting Language (NASL)

- Allows for writing custom security plugins
- Don't have to know internal workings of Nessus
- Allows for independent development of plugins

Plugins Configuration

- Can selectively enable or disable groups as well as individual plugins
- Click on category to see sub-category of plugins
- Plugins that can cause servers to crash are highlighted with triangular exclamation symbol
- Dangerous plugins off by default
- Can enable all, all but dangerous, disable all or load custom plugins
- Can filter by name, description, summary, author, id number or category

Preferences — Nessus Integration with Other Tools

- Can use Nmap for port scanning
- Nikto and Whisker — testing web servers
- Hydra — brute force password attacks
- CGI programs — tests your Web servers CGIs
- Use Nessus client to configure
- Not all tools available on all platforms

Preferences — Nmap Configuration

- Access to most of Nmap's builtin options
- Can use an existing Nmap results file so don't have to run new scan

Preferences — Login Configuration

- By default Nessus operates as if you are a "stranger"
- If want it to perform deeper tests can provide service passwords
- Can also give a specific HTTP login page with associated form fields and password
- If Hydra is available can configure for brute force login attempts (not an Ubuntu/Debian option)

Preferences — Service Configuration

- Used for testing SSL services
- Can specify certificates to get reports on level of encryption web server supports
- Can be used to see if your web server still accepts 40-bit certificates (not safe)

Preferences — News Server Configuration

- Checks for possibility for spamming or other misuse of NNTP servers

Scan Options — Port Range

- By default checks ports 1-15000 which covers most services
- Can have it check all 65, 535 ports to look for Trojans
- Consider unscanned ports as closed — allows for faster check

Scan Options — Number of Hosts / Checks to Test / Perform at the Same Time

- If do too many at once may actually slow down testing
- Set to about 10 on a normal server (instead of default of 20)
- Very fast server on large network may try more simultaneously

Scan Options — Miscellaneous

- Path to the CGI — can change path if CGIs in non-standard place
- Do reverse lookup — to get host names
- Optimize — Nessus won't perform tests that don't apply to a particular host
- Shut off optimize if want to apply every possible test regardless
- Safe checks — doesn't actually run tests that may crash a server
- Designate hosts by MAC address — useful in environment where IP is assigned by DHCP

Scan Options — Port Scanner

- May vary on different platforms
- SYN Scan — if don't use Nmap can use built in SYN scanner
- LaBrea tarpitted hosts — set up to catch port scanners; check in order to detect and avoid

Target Options — Target Selection

- Single IP — 192.168.1.100
- IPs separated by comma — 192.168.1.1,192.168.1.2
- IP range — 192.168.1.1-192.168.1.100
- CIDR — 192.168.1.0/24
- Hostname — www.cuny.edu
- Any combination of above
- Can also read from a text file

Target Options — Miscellaneous

- Perform a DNS zone transfer — for a DNS configured domain
- Save this session — so can be used again
- Save empty session — sessions with no live hosts
- Previous sessions — can reload in future

User Tab

- Shows all users and associated rules
- Can edit and add rules here

Knowledge Base

- Can store scan results in a database
- Can use results of past scans to make current scan more “intelligent”
- Can avoid doing a port scan each time run Nessus
- Only tests found hosts and open ports
- Shouldn't always use it as new hosts may be added and new ports opened up

Reporting

- Can save in multiple formats
- NBE is native format
- HTML provides well-formatted report