Testing for vulnerabilities

Service enumeration

PURPOSE

- Identify active services on systems
  - Also find out the role of the server
- Identify the apps that offer those services
- Identify app version
  - To search for specific vulnerabilities
- Identification of vulnerabilities or configuration errors, in order to exploit
  - Further information can be obtained
  - Can gain access to the target

PURPOSE

- Vulnerabilities
  - An error in the application that can be exploited by an attacker
  - Buffer overflows, Use after free, command injection, etc.
  - Usually, they can be exploited no matter how well the service is configured
  - Easy to discover – e.g. google "application 1.4.7 vulnerability"
- Configuration errors
  - Ports left open, default credentials (root:root), accessible configuration files, accessible installation files, etc.
  - Usually, they can be exploited no matter how well the application is written
  - Harder to discover
  - Read the manual/documentation

METHODOLOGY

- Nmap identifies the service, application, version & operating system
- The attacker looks for vulnerabilities or configuration errors for that version of the application and for that operating system
- We have 4 cases:

1. We find vulnerabilities that can be exploited
2. We find vulnerabilities that don’t have a public exploit
3. Configuration errors are discovered
4. We don't find anything

METHODOLOGY

1. Vulnerabilities that can be exploited
   - Metasploit – point and click
   - Exploit-db, packetstorm, etc. – source code, scripts, step-by-step descriptions
2. Vulnerabilities that don’t have a public exploit
   - We develop the exploit ourselves
3. Configuration errors
   - Default credentials, backdoor accounts, installation scripts, etc.
4. We find nothing
   - Fuzzing
   - Code Audit (Open Source)
   - As a rule, if you don't discover any vulnerabilities or configuration errors, you can assume that the service is secure

FTP

- FTP – File Transfer Protocol
  - Protocol used for file transfer
  - Very common on servers
  - FTP services can be run on both Windows and Linux
    - Some FTP servers are specific (Linux/Windows)
    - CrushFTP, ProFTPD, SFTPPlus, Pure-FTPd, vsftpd, Wu-ftp, Cerberus, FileZilla, IIS
- Vulnerabilities and attacks
  - Remote Code Execution
    - Buffer overflow, use-after-free, command injection, etc.
  - Auth Bypass
    - Backdoor accounts, default credentials, etc.
  - Information Disclosure
    - Directory traversal, format-string, etc.
  - Brute Force
  - Denial-of-Service

HTTP

- HTTP – Hyper-Text Transfer Protocol
  - Web Protocol
  - HTTP servers present on most servers
  - Linux and Windows
  - Apache, IIS, lighthttpd, nginx, etc.
- Generally, the vulnerabilities are in the exposed web services, not in the server
- Vulnerabilities and attacks
  - Remote Code Execution
    - Buffer overflow, use-after-free, command injection, etc.
  - Information Disclosure
    - Traversal directory, accessible config files, phpinfo, etc.
  - Web Service-Specific Vulnerabilities
    - Local File Inclusion, Remote File Inclusion, Directory Traversal, Cross Site Request Firgery, Cross Site Scripting, Command Injection, Authentication
    - Bypass, Default Credentials, Weak-Credentials, SQL Injection, Brute-Force, etc.
  - Denial-of-Service

HTTP

- example

```
https://10.x.x.x/section.php?page=alldocs
```

- attack

```
https://10.x.x.x/section.php?page=../../../../../../../../../../../../../etc/passwd%00
```

SMTP

- SMTP – Simple Mail Transfer Protocol
  - Linux & Windows:
  - Exchange, OpenSMTPD, Postfix, qmail, sendmail, etc.
- Vulnerabilities and attacks
  - Remote Code Execution
    - Buffer overflow, use-after-free, command injection, etc.
  - Authentication bypass
  - Information Disclosure
    - Existing users (VRFY), etc.
  - Denial-of-Service

SMTP

- example

```
root@kali# nc -nv 10.x.x.x 25
(UNKNOWN) [10.x.x.x] 25 (smtp) open
vrfy root
expn
220 barry ESMTP Sendmail 8.11.6/8.11.6; Wed, 4 Jan 20
250 2.1.5 root <root@barry>
501 5.5.2 Argument required
```

SSH

- SSH – Secure Shell
  - Allows running services on the network in an secured manner
  - Best known for remote-login
- Specific to Linux, but there are also variants for Windows
  - OpenSSH
- Vulnerabilities & attacks
  - Remote Code Execution
  - Authentication Bypass
  - Brute Force
  - Information Disclosure
  - Denial-of-Service

SMB

- SMB – Samba
  - resource sharing (mainly Windows)

OTHER CASES

- Common Library Vulnerabilities
  - Hearttleed – OpenSSL
    - OpenSSL is used by HTTP, SSH, FTP servers, etc.
    - Any service that uses the vulnerable version is affected
- Vulnerabilities in the operating system
  - ShellShock
    - Vulnerability in bash
    - Affected services: HTTP (CGI), DHCP, Qmail
- Services taken individually may not be vulnerable
- At first glance, each service appears to be set up correctly
- A configuration error in one or more services can mean that a system is compromised

Conclusions

- Vulnerabilities
- Configuration errors
- It is critical to identify the application, version, etc.
- Some services are easily exploitable
- Sometimes, we need to use vulnerabilities from multiple services
- In this phase – google is your best friend
  - exploit-db.com
  - packetstormsecurity.com
  - cvedetails.com
  - cve.mitre.gov
  - nvd.nist.gov
- Sometimes, you won’t find ready-made exploits
- Sometimes, you won't find vulnerabilities
- Sometimes, you won't even find configuration errors