Nmap scripting for sysadmins and network troubleshooting

Daniel Colson

Background/Intro

Background/Intro

“Nmap ("Network Mapper") is a free and open source utility for network discovery and security auditing.”

First released in 1997 by Gordon Lyon (Fyodor)

Scripting Engine released in 2006

Featured in at least 12 movies

Including The Matrix Reloaded, Die Hard 4, and The Bourne Ultimatum

https://nmap.org/

Scan Phases

1. Pre-scanning

    Scripts for adding targets or certain things like dhcp-discover run here.

2. Target Enumeration
3. Host Discovery (ping, etc)
4. Reverse-DNS Resolution
5. Port Scanning
6. Version Detection (if requested)

Scan Phases

7. OS Detection (if requested)
8. Traceroute (if requested)
9. Script Scanning (if requested)

   Most Nmap scripts run here at the end of the main scanning process

10. Output
11. Script Post Scanning

    Additional Nmap scripts may run here to process and display final results.

General Usage

Command structure:

```
nmap [ <Scan Type> ...] [ <Options> ] { <target specification> }
```

Examples:

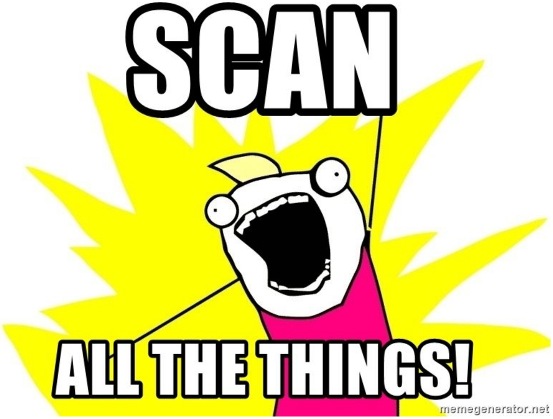
```
nmap -v scanme.nmap.org

nmap -sC -sV 192.168.1.1/24 -p 80,443,3389,3306

nmap –script discovery,safe 10.0.0.1-20 -p 1000-3000

nmap –script “http-*” -p 80 10.0.0.1,192.168.1.27

nmap –script /path/to/my-scripts scanme2.nmap.org
```



Examples

```
nmap -sC -sV scanme.nmap.org

PORT      STATE    SERVICE    VERSION
22/tcp    open     ssh        OpenSSH 6.6.1p1 Ubuntu 2ubuntu2.13 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   1024 ac:00:a0:1a:82:ff:cc:55:99:dc:67:2b:34:97:6b:75 (DSA)
|   2048 20:3d:2d:44:62:2a:b0:5a:9d:b5:b3:05:14:c2:a6:b2 (RSA)
|   256 96:02:bb:5e:57:54:1c:4e:45:2f:56:4c:4a:24:b2:57 (ECDSA)
|_  256 33:fa:91:0f:e0:e1:7b:1f:6d:05:a2:b0:f1:54:41:56 (ED25519)
80/tcp    open     http       Apache httpd 2.4.7 ((Ubuntu))
|_http-favicon: Nmap Project
|_http-title: Go ahead and ScanMe!
|_http-server-header: Apache/2.4.7 (Ubuntu)
646/tcp   filtered ldp
9929/tcp  open     nping-echo Nping echo
31337/tcp open     tcpwrapped
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel
```

Nmap Scripts

Nmap Scripts - Overview

Written in Lua

Execute in parallel

604 NSE scripts included with Nmap

139 NSE libraries included for scripts to use

Organized into 14 categories:

auth, broadcast, brute, default, discovery, dos, exploit, external, fuzzer, intrusive, malware, safe, version, vuln

Nmap Scripts - Usage

Examples

1. nmap –script http-title
2. nmap –script /path/to/myscript.nse
3. nmap –script “http-*”
4. nmap –script default,safe
5. nmap –script “not intrusive”
6. nmap --script mysql-users –script-args “mysqluser=’admin’, mysqlpass=’password’”

Nmap Scripts - Useful Scripts

- dhcp-discover (or broadcast-dhcp-discover)
  - Get details on DCHP configuration for a network
- firewalk
  - Tries to discover firewall rules using an IP TTL expiration technique known as firewalking.
- hostmap-crtsh, hostmap-robtex
  - Scripts for finding subdomains and hostnames for an IP using external data sources
- ip-geolocation-*
  - Use various online services to geolocate an IP address
- shodan-api
  - Search for information about the target host on Shodan (Internet-wide port/service scan database)
- ldap-search
  - Query an LDAP service on a host

Nmap Scripts - Useful Scripts

- mongodb-*, ms-sql-*, mysql-*
  - Run queries, security audits, and other operations on database servers
- http-*
  - Test the time a web server takes to return a page
  - Check Apache server status
  - Find RSS/Atom feeds
  - List HTTP Headers
  - Check if a host is on lists of known malicious web servers
  - Check version numbers for PHP, Wordpress, and other dev/service frameworks
- smb-*
  - Use Samba protocol to retrieve users, groups, processes, shares, registry, etc from a host.

Nmap Scripts - Useful Scripts

- ssh-run
    - Run a remote command on the server via SSH and return the output.
- ssl-enum-ciphers
    - Determines what SSL/TLS ciphers and compression methods a server accepts.
- targets-asn
    - Uses an external whois server to list all IP ranges for a given routing AS number
- vnc-info
    - Queries a VNC server for its protocol version and supported security types.
- dns-check-zone
    - Checks DNS zone configuration against best practices, including RFC 1912.
- whois-domain, whois-ip
    - Query whois database for details on the target

Nmap Scripts - Security Testing

- http-shellshock
  - Tests web applications for vulnerability to the Shellshock attack (CVE-2014-6271)
- http-vuln-cve2017-8917
  - Attempts an SQL injection on Joomla! web servers to test for this vulnerability.
- smb-brute
  - Runs a brute-force attack on an SMB server to try and find valid login credentials.
- http-wordpress-enum
  - Exploits an information disclosure vulnerability in certain older versions of Wordpress to get a list of all users on the site.
- http-csrf
  - Detect potential Cross-Site Request Forgery vulnerabilities in a web page

Nmap Script Development

https://github.com/Red5d/nse-scripts

Resources

Nmap Script Writing Tutorial - https://nmap.org/book/nse-tutorial.html

List of all built-in Nmap scripts - https://nmap.org/nsedoc/scripts/

List of all built-in Nmap libraries - https://nmap.org/nsedoc/lib/

Documentation and source code from each script/library web page

Scripting section of the Nmap book - https://nmap.org/book/nse.html

Other presentations on Nmap in general - https://nmap.org/presentations/

Main Concepts

Script Sections:

- **Head** - Script metadata (description, categories, etc) and library imports.

- **Rule** - Lua function which decides whether to skip or execute the script's action. Usually based on host/port info.
    - One of: portrule, hostrule, prerule, postrule

- **Action** - Function that contains the actual code that will be executed.

https://nmap.org/book/nse-tutorial.html

Main Concepts - Head

```
local http = require "http"
local shortport = require "shortport"
local stdnse = require "stdnse"

description = [[Southeast LinuxFest Example]]
author = "Red5d"
license = "Same as Nmap--See https://nmap.org/book/man-legal.html"
categories = {"discovery", "safe"}
```

Main Concepts - Rule

- prerule - Execute script before the scan
  - Collect service information, add new targets

- portrule - Execute script for all matching ports on each target during scan
  - Use this for scripts that run actions/queries/tests on discovered services

- postrule - Execute script after Nmap has scanned all targets
  - Use this for scripts that format output and use saved data from other scripts

- hostrule - Execute script once per host after Nmap has scanned all targets.
  - Used by the whois-ip script for looking up whois data on a host

Main Concepts - Rule - **pre**rule

```
prerule = function()
        if not nmap.is_privileged() then
                stdnse.verbose1("not running for lack of privileges.")
                return false
        end
        return true
end
```

```
prerule = function()
        return true
end
```

Main Concepts - Rule - **port**rule

```
portrule = shortport.http      (used in example script)
```

```
portrule = function(host, port)
   return port.protocol == "tcp"
          and port.number == 80
          and port.state == "open"
end
```

Main Concepts - Rule - **post**rule

– From ssh-hostkey.nse. If an ssh host key was found in a previous scan phase and stored in the registry, return true and run the action.

```
postrule = function()
        return (nmap.registry.sshhostkey ~= nil)
end
```

Main Concepts - Rule - **host**rule

– From whois-ip.nse. Checks if the target IP is routable on the Internet

```
hostrule = function( host )
        local is_private, err = ipOps.isPrivate( host.ip )
        if is_private == nil then
                stdnse.debug1("Error in Hostrule: %s.", err)
                return false
        end

        return not is_private
end
```

Main Concepts - Action (1)

```
action = function(host, port)
    -- Perform HTTP GET request
    resp = http.get(host, port, "/")

    -- Regex on the response body to find info
    local latest_post = resp.body:match("bookmark\">([^<]+)")
    local last_updated = resp.body:match("timestamp updated\">([^<]+)")
    local author = resp.body:match('posts by ([^"]+)')


...
```

Main Concepts - Action (2)

```
    -- Create an output table and load the info into it
    local output_tab = stdnse.output_table()
    output_tab.latest_post = latest_post
    output_tab.last_updated = last_updated
    output_tab.author = author

    -- Return the output table
    return output_tab
end
```

Example Script Results - scanned on June 1, 2022

Command: nmap --script selinuxfest southeastlinuxfest.org -p 443

```
PORT    STATE SERVICE
443/tcp open  https
| selinuxfest:
|   latest_post: SELF 2022:  Coming Down The Home Stretch
|   last_updated: May 21, 2022
|_  author: George P. Burdell
```

Example - External IP lookup

```
action = function(host, port)

    -- HTTP GET request to https://ifconfig.me/
    res = http.get("ifconfig.me", 443, "/")

    -- Regex to find the IP address
    local ipaddr = res.body:match("ip_address\">([^<]+)")

    -- Return a string with the IP address
    return "External IP Address: " .. ipaddr

end
```

Example - External IP lookup

Command: nmap --script externalip

```
Pre-scan script results:
|_externalip: External IP Address: <IP address>
WARNING: No targets were specified, so 0 hosts scanned.
```

Example - Vulnerability Scanner Script - CVE-2017-12542

```
res = http.get(host.ip, 443, "/xmldata?item=ALL")
local version = stdnse.strsplit("<FWRI>", res.body)[2]
local version = stdnse.strsplit("</FWRI>", version)[1]
output = {}
table.insert(output, "HP iLO Firmware Version: " .. version)
if 2.3 <= tonumber(version) and tonumber(version) <= 2.5 then
        table.insert(output, "Vulnerable: yes")
else
        table.insert(output, "Vulnerable: no")
end
return output
```

Useful NSE Libraries

Protocol

- http
- dns
- ftp
- imap
- irc
- bitcoin
- dhcp
- ipmi
- ldap
- msrpc
- mysql
- smb
- ssh2

Utility

vulns	Formatting for vuln check results
ipOps	Manipulate/compare IPs
httpspider	Basic HTTP spidering capability
pcre	Regular Expression matching
stdnse	Standard useful NSE functions
stringaux	String manipulation functions
target	Add targets to scan queue
Nmap	Interface with Nmap internals
url	URI parsing/composition

Custom Script Ideas

1. Internal asset information lookup using SQL/HTTP/LDAP libraries
2. Auditing for consistency or specific security requirements
3. Troubleshooting for common network/server issues
4. Quick version/vulnerability check across many servers
5. External data lookups (threat intel, DNS, network info, connection tests...)

Questions?

Github: **https://github.com/Red5d**

Matrix: **@red5d:dmatrix.duckdns.org**

Example NSE Scripts: **https://github.com/Red5d/nse-scripts**