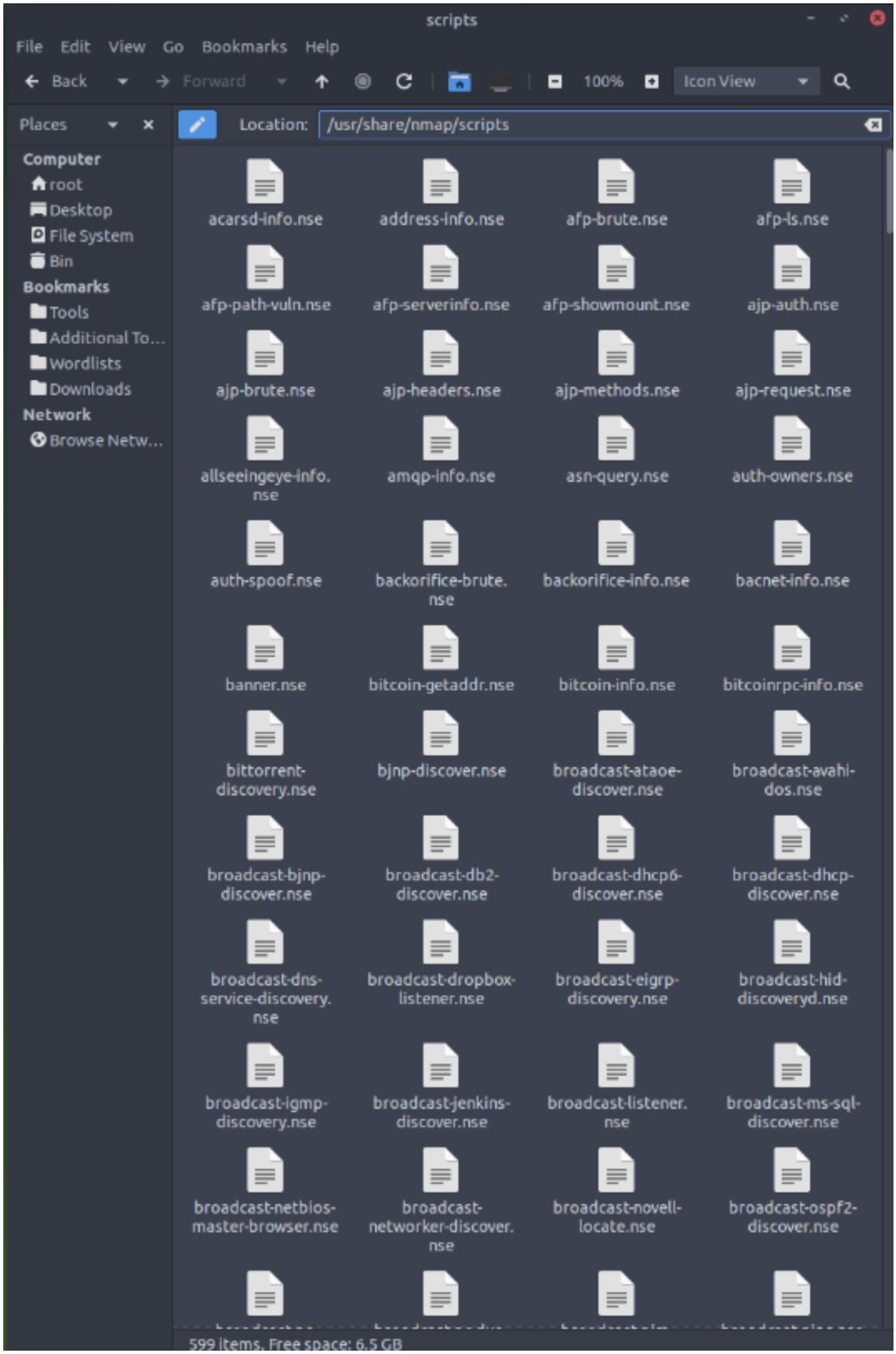
Nmap Scripting Engine (NSE)

A script is a piece of code that does not need to be compiled: **remains in its original human-readable format and does not need to be converted to machine language.**

There are many programs that provide additional functionality via scripts but the basis is scripts allow us to add custom functions that did not exist via the built-in commands.
NSE: Is a Lua interpreter that allows Nmap to execute Nmap scripts written in Lua Language. However we don't need Lua to make use of Nmap Scripts.

Nmap default installation can easily contain close to 600 scripts. Nmap installation folder if we check the files at /usr/share/nmap/scripts we will notice there are hundreds of scripts conveniently named starting with the protocol target.



We can specify to use any group of these installed scripts and could also expand on them / add more to use for scans. We can choose to run default scripts however by doing **--script=default**

or -sC categories include, **auth, brute, broadcast, default, discovery, dos, exploit, external, fuzzer, intrusive, malware, safe, version and vuln.**

Script Category	Description
`auth`	Authentication related scripts
`broadcast`	Discover hosts by sending broadcast messages
`brute`	Performs brute-force password auditing against logins
`default`	Default scripts, same as `-sC`
`discovery`	Retrieve accessible information, such as database tables and DNS names
`dos`	Detects servers vulnerable to Denial of Service (DoS)
`exploit`	Attempts to exploit various vulnerable services
`external`	Checks using a third-party service, such as Geoplugin and Virustotal
`fuzzer`	Launch fuzzing attacks
`intrusive`	Intrusive scripts such as brute-force attacks and exploitation
`malware`	Scans for backdoors
`safe`	Safe scripts that won't crash the target
`version`	Retrieve service versions
`vuln`	Checks for vulnerabilities or exploit vulnerable services

```
Pentester Terminal

pentester@TryHackMe$ sudo nmap -sS -sC 10.10.130.226

Starting Nmap 7.60 ( https://nmap.org ) at 2021-09-10 05:08 BST
Nmap scan report for ip-10-10-161-170.eu-west-1.compute.internal (10.10.161.170)
Host is up (0.0011s latency).
Not shown: 994 closed ports
PORT    STATE SERVICE
22/tcp  open  ssh
| ssh-hostkey:
|   1024 d5:80:97:a3:a8:3b:57:78:2f:0a:78:ae:ad:34:24:f4 (DSA)
|   2048 aa:66:7a:45:eb:d1:8c:00:e3:12:31:d8:76:8e:ed:3a (RSA)
|   256 3d:82:72:a3:07:49:2e:cb:d9:87:db:08:c6:90:56:65 (ECDSA)
|_  256 dc:f0:0c:89:70:87:65:ba:52:b1:e9:59:f7:5d:d2:6a (EdDSA)
25/tcp  open  smtp
|_smtp-commands: debra2.thm.local, PIPELINING, SIZE 10240000, VRFY, ETRN, STARTTLS, ENHANCEDSTATUSCODES, 8BITMIME, DSN,
| ssl-cert: Subject: commonName=debra2.thm.local
| Not valid before: 2021-08-10T12:10:58
|_Not valid after:  2031-08-08T12:10:58
|_ssl-date: TLS randomness does not represent time
80/tcp  open  http
|_http-title: Welcome to nginx on Debian!
110/tcp open  pop3
|_pop3-capabilities: RESP-CODES CAPA TOP SASL UIDL PIPELINING AUTH-RESP-CODE
111/tcp open  rpcbind
| rpcinfo:
|   program version   port/proto  service
|   100000  2,3,4        111/tcp  rpcbind
|   100000  2,3,4        111/udp  rpcbind
|   100024  1          38099/tcp  status
|_  100024  1          54067/udp  status
143/tcp open  imap
|_imap-capabilities: LITERAL+ capabilities IMAP4rev1 OK Pre-login ENABLE have LOGINDISABLEDA0001 listed SASL-IR ID more post-login LOGIN-R
MAC Address: 02:A0:E7:B5:B6:C5 (Unknown)

Nmap done: 1 IP address (1 host up) scanned in 2.21 seconds
```

The example shown above is doing a stealth SYN scan followed by the default script command -sC.

We can also specify the script name by doing **--script *SCRIPT NAME*** specified. Or pattern **--script "ftp" which would include ftp-brute.**

```
Pentester Terminal

pentester@TryHackMe$ sudo nmap -sS -n --script "http-date" 10.10.130.226

Starting Nmap 7.60 ( https://nmap.org ) at 2021-09-10 08:04 BST
Nmap scan report for 10.10.130.226
Host is up (0.0011s latency).
Not shown: 994 closed ports
PORT    STATE SERVICE
22/tcp  open  ssh
25/tcp  open  smtp
80/tcp  open  http
|_http-date: Fri, 10 Sep 2021 07:04:26 GMT; 0s from local time.
110/tcp open  pop3
111/tcp open  rpcbind
143/tcp open  imap
MAC Address: 02:44:87:82:AC:83 (Unknown)

Nmap done: 1 IP address (1 host up) scanned in 1.78 seconds
```

Finally, you might expand the functionality of Nmap beyond the official Nmap scripts; you can write your script or download Nmap scripts from the Internet. Downloading and using a Nmap script from the Internet holds a certain level of risk. So it is a good idea not to run a script from an author you don't trust.

Finally as shown above we could run the script **"http-date" Or**

nmap -sS -n --script “http-date” 10.10.130.226 As shown in the above example. Which gives the HTTP date and time! Very useful actually.

Conclusion

So what this means, IF **the target system is running the application HTTP eg on port 80** the HTTP service responds normally but the server also includes a date / time.
We could use this to check for **clock drift** or see multiple services on different IPs that have the same system time. This suggests they are on the same host.
What does this mean?

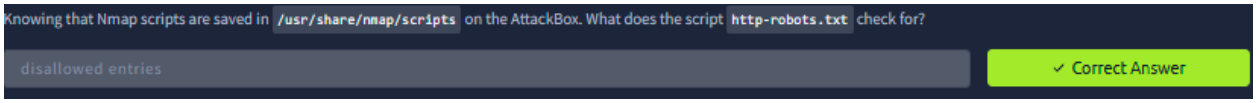
1. **Infer Location (Timezone)**
2. **Correlate Systems**
3. **Distinguish real vs fake /Decoys or zombies**

Blue Team:
Can find red team target if they are using decoys, or zombie hosts to launch idle scans. Comparing date value to system time. Or measure clock skew or time zones, use it to narrow down systems vs spoofed ones.

Red Team:
Compare across multiple hosts to see if multiple domains sit behind the same physical host (skew). Can guess geographical location/time zone, discover virtualised environments or containers. Find out if service is running in a sandbox or honeypot.

I have now customised and added this script to my own tool belt, because one day I can use it to my advantage. I tested it on myself and received the time and date of my current service / machine. I also used the netstat tool to see if anything malicious was on my computer, thankfully it wasn’t but it is an excellent way at reversing any attacks to try pinpoint malicious targets.

Questions:



We can see in the description on line 8 tells us what it checks for, which it checks for disallowed entries.

http-date.nse × http-robots.txt.nse ×

```
local http = require "http"
local nmap = require "nmap"
local shortport = require "shortport"
local strbuf = require "strbuf"
local table = require "table"

description = [[
Checks for disallowed entries in <code>/robots.txt</code> on a web server.

The higher the verbosity or debug level, the more disallowed entries are shown.
]]

---
--@output
-- 80/tcp  open   http    syn-ack
-- |  http-robots.txt: 156 disallowed entries (40 shown)
-- |  /news?output=xhtml& /search /groups /images /catalogs
-- |  /catalogues /news /nwshp /news?btcid=*& /news?btaid=*&
-- |  /setnewsprefs? /index.html? /? /addurl/image? /pagead/ /relpage/
-- |  /relcontent /sorry/ /imgres /keyword/ /u/ /univ/ /cobrand /custom
-- |  /advanced_group_search /googlesite /preferences /setprefs /swr /url /default
-- |  /m? /m/? /m/lcb /m/news? /m/setnewsprefs? /m/search? /wml?
-- |_ /wml/? /wml/search?
```

Can you figure out the name for the script that checks for the remote code execution vulnerability MS15-034 (CVE2015-1635)?

http-vuln-cve2015-1635 ✓ Correct Answer

```
root@ip-10-10-95-179:~# nmap -sC 10.10.180.218
Starting Nmap 7.80 ( https://nmap.org ) at 2025-07-27 12:24 BST
Nmap scan report for ip-10-10-180-218.eu-west-1.compute.internal (10.10.180.218)
Host is up (0.0029s latency).
Not shown: 991 closed ports
PORT    STATE SERVICE
22/tcp  open  ssh
25/tcp  open  smtp
|_smtp-commands: debra2.thm.local, PIPELINING, SIZE 10240000, VRFY, ETRN, STARTTLS, ENHANCEDSTATUSCODES, 8BITMIME, DSN, CHUNKING
,
| ssl-cert: Subject: commonName=debra2.thm.local
| Not valid before: 2021-08-10T12:10:58
|_Not valid after:  2031-08-08T12:10:58
|_ssl-date: TLS randomness does not represent time
53/tcp  open  domain
| dns-nsid:
|_  bind.version: 9.18.28-1~deb12u2-Debian
80/tcp  open  http
|_http-title: Welcome to nginx on Debian!
110/tcp open  pop3
|_pop3-capabilities: SASL CAPA PIPELINING AUTH-RESP-CODE TOP UIDL RESP-CODES STLS
| ssl-cert: Subject: commonName=debra2.thm.local
| Not valid before: 2021-08-10T12:10:58
|_Not valid after:  2031-08-08T12:10:58
111/tcp open  rpcbind
| rpcinfo:
|   program version    port/proto  service
|   100000  2,3,4        111/tcp   rpcbind
|   100000  2,3,4        111/udp   rpcbind
|   100000  3,4          111/tcp6  rpcbind
|   100000  3,4          111/udp6  rpcbind
```

Launch the AttackBox if you haven't already. After you ensure you have terminated the VM from Task 2, start the target machine for this task. On the AttackBox, run Nmap with the default scripts `-sC` against `10.10.180.218`. You will notice that there is a service listening on port 53. What is its full version value?

9.18.28-1~deb12u2-Debian ✓ Correct Answer

```
root@ip-10-10-95-179:~# nmap -script "ssh2-enum-algos" 10.10.180.218
Starting Nmap 7.80 ( https://nmap.org ) at 2025-07-27 12:27 BST
Nmap scan report for ip-10-10-180-218.eu-west-1.compute.internal (10.10.180.218)
Host is up (0.0069s latency).
Not shown: 991 closed ports
PORT    STATE SERVICE
22/tcp  open  ssh
| ssh2-enum-algos:
|   kex_algorithms: (11)
|       sntrup761x25519-sha512@openssh.com
|       curve25519-sha256
|       curve25519-sha256@libssh.org
|       ecdh-sha2-nistp256
|       ecdh-sha2-nistp384
|       ecdh-sha2-nistp521
|       diffie-hellman-group-exchange-sha256
|       diffie-hellman-group16-sha512
|       diffie-hellman-group18-sha512
|       diffie-hellman-group14-sha256
|       kex-strict-s-v00@openssh.com
|   server_host_key_algorithms: (4)
|       rsa-sha2-512
|       rsa-sha2-256
|       ecdsa-sha2-nistp256
|       ssh-ed25519
|   encryption_algorithms: (6)
|       chacha20-poly1305@openssh.com
|       aes128-ctr
|       aes192-ctr
|       aes256-ctr
|       aes128-gcm@openssh.com
|       aes256-gcm@openssh.com
|   mac_algorithms: (10)
|       umac-64-etm@openssh.com
|       umac-128-etm@openssh.com
|       hmac-sha2-256-etm@openssh.com
|       hmac-sha2-512-etm@openssh.com
|       hmac-sha1-etm@openssh.com
|       umac-64@openssh.com
|       umac-128@openssh.com
|       hmac-sha2-256
|       hmac-sha2-512
|       hmac-sha1
|   compression_algorithms: (2)
|       none
|       zlib@openssh.com
```

Based on its description, the script `ssh2-enum-algos` "reports the number of algorithms (for encryption, compression, etc.) that the target SSH2 server offers." What is the name of the server host key algorithm that relies on SHA2-512 and is supported by `10.10.180.218`?

rsa-sha2-512 ✓ Correct Answer 💡 Hint

Saving the file:

Whenever you use Nmap scan it is only reasonable to save the results to a file. Selecting and adopting a good naming convention is crucial. The number of files can grow quickly. 3 main formats are:

1. **Normal**
2. **Grepable (grep able)**
3. **XML**
4. **Script Kiddie (Not recommended)**

Normal:
Normal format similar to the output you get on the command prompt screen. **-oN filename to achieve this.**

```
pentester@TryHackMe$ cat MACHINE_IP_scan.nmap
# Nmap 7.60 scan initiated Fri Sep 10 05:14:19 2021 as: nmap -sS -sV -O -oN MACHINE_IP_scan 10.10.180.218
```

Grapable:
Command grep, meaning Global Regular Expression Printer, it makes filtering the scan output for specific keywords or terms efficient. **-oG filename to achieve this.**

```
pentester@TryHackMe$ cat MACHINE_IP_scan.gnmap
# Nmap 7.60 scan initiated Fri Sep 10 05:14:19 2021 as: nmap -sS -sV -O -oG MACHINE_IP_scan 10.10.180.218
```

XML:
Save results in XML format. **-oX filename to achieve this.** Or use **-oA filename to combine all 3 together.**

Script kiddie:
Useless format. Can use it to save the output of the scan, displaying the output filename.

```
pentester@TryHackMe$ cat MACHINE_IP_scan.kiddie

$tart!ng nMaP 7.60 ( httpz://nMap.0rG ) at 2021-09-10 05:17 B$T
Nmap scan rEp0rt f0r |p-10-10-161-170.EU-w3$t-1.C0mputE.intErnaL (10.10.161.170)
```

scp pentester@10.10.15.3:/home/pentester/*
Password: THM17577

scp = secure copy remote
pentester = username of target machine
10.10.15.3 = Target IP
/home/pentester/* = All files in users home directory
. = current directory on our attack box

scp pentester@10.10.15.3:/home/pentester/* .

```
root@ip-10-10-217-125:~# scp pentester@10.10.15.3:/home/pentester/* .
The authenticity of host '10.10.15.3 (10.10.15.3)' can't be established.
ECDSA key fingerprint is SHA256:tu3cN9tpWkjX8w+YCxcuYqURILsRv37ypGkfazubEUE.
Are you sure you want to continue connecting (yes/no/[fingerprint])?
```

Yes
I type the password: THM17577

```
Warning: Permanently added '10.10.15.3' (ECDSA) to the list of known hosts.
pentester@10.10.15.3's password:
scan_172_17_network.gnmap                     100%   13KB   5.8MB/s   00:00
scan_172_17_network.nmap                      100%   17KB   6.8MB/s   00:00
root@ip-10-10-217-125:~# ls
burp.json    Instructions  scan_172_17_network.gnmap  thinclient_drives
CTFBuilder   Pictures      scan_172_17_network.nmap   Tools
Desktop      Postman       Scripts
Downloads    Rooms         snap
root@ip-10-10-217-125:~#
```

```
root@ip-10-10-217-125:~# grep https scan_172_17_network.gnmap
Host: 172.17.0.215 ()   Ports: 22/closed/tcp//ssh///, 80/open/tcp//http///, 443/open/tcp//https///
Ignored State: filtered (997)
Host: 172.17.19.249 ()  Ports: 22/open/tcp//ssh///, 53/open/tcp//domain///, 80/open/tcp//http///, 443/op
en/tcp//https///        Ignored State: closed (996)
Host: 172.17.23.240 ()  Ports: 22/closed/tcp//ssh///, 80/open/tcp//http///, 443/open/tcp//https///
Ignored State: filtered (997)
```

Check the attached Nmap logs. How many systems are listening on the HTTPS port?

3 ✓ Correct Answer

```
root@ip-10-10-217-125:~# grep 8089 https scan_172_17_network.gnmap
grep: https: No such file or directory
scan_172_17_network.gnmap:Host: 172.17.20.147 ()        Ports: 22/open/tcp//ssh///, 8000/open/tc
p//http-alt///, 8089/open/tcp//unknown///       Ignored State: closed (997)
```

What is the IP address of the system listening on port 8089?

172.17.20.147 ✓ Correct Answer

Option	Meaning
`-sV`	determine service/version info on open ports
`-sV --version-light`	try the most likely probes (2)
`-sV --version-all`	try all available probes (9)
`-O`	detect OS
`--traceroute`	run traceroute to target
`--script=SCRIPTS`	Nmap scripts to run
`-sC` or `--script=default`	run default scripts
`-A`	equivalent to `-sV -O -sC --traceroute`
`-oN`	save output in normal format
`-oG`	save output in grepable format
`-oX`	save output in XML format
`-oA`	save output in normal, XML and Grepable formats