Starting to Write Nmap Scripts

A little Lua and NSE to get you started

About This Joker…

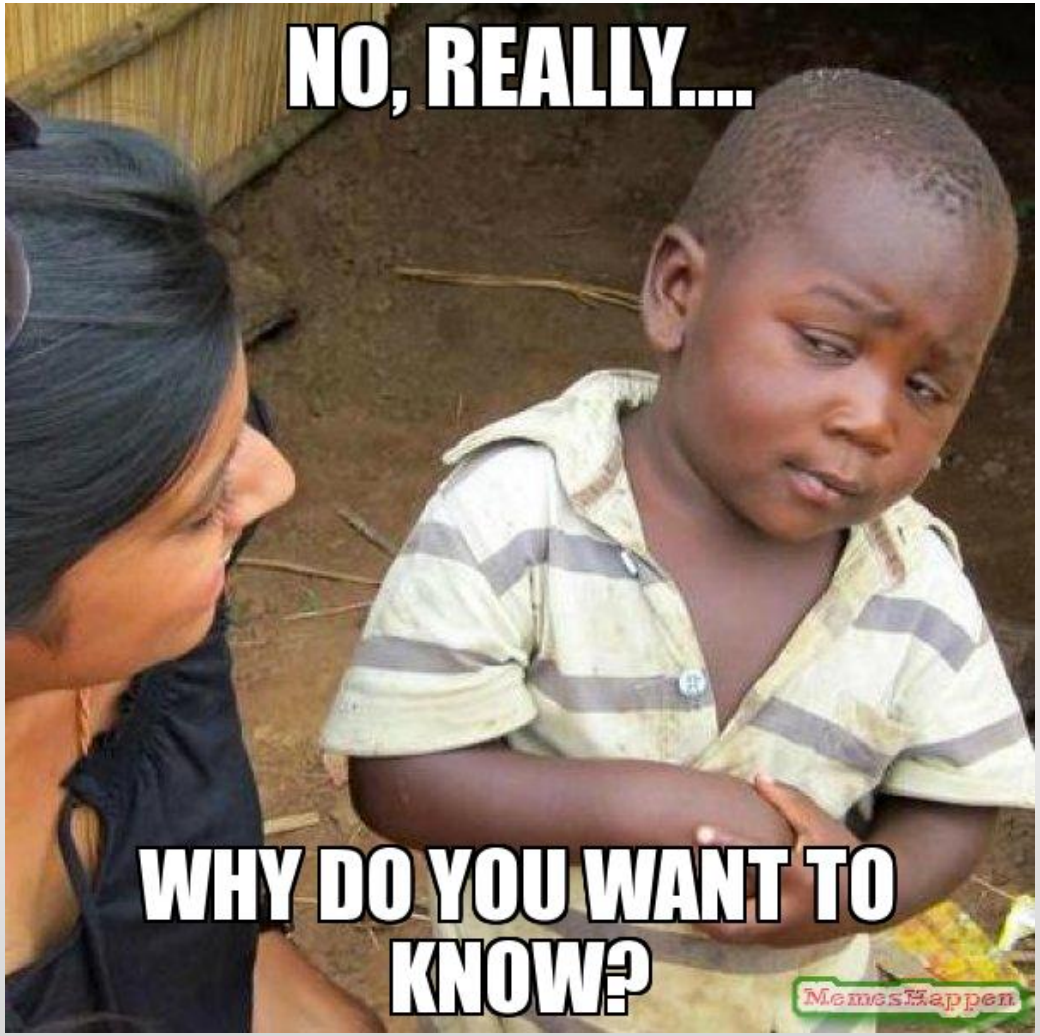
- Founder of Paladin Security
  - Web, mobile and network penetration testing
- Penetration Tester and Security Consultant
- Red Team Member @ Huge-Bank-That-Shall-Not-Be-Named
- CyberPatriot Mentor

- Scripts & Slides - https://github.com/tadaka/nse-scripts/

The Nmap Scripting Engine

- Built to allow users the ability to right their own checks
- Provide flexibility
- Doesn’t require you to know C/C++
- Avoid security issues like buffer overflows
- Uses the Lua scripting language
- Makes it way easier to add new checks
- 558 scripts currently available

Why Write NSE Scripts?



But first…. Lua

Lua

- First released 1993
- Lightweight, embeddable scripting language
- Well documented API
- Used in Wireshark, video games, etc
- Selected by Nmap Project instead of creating a new one

Comments

```
-- This is a comment


---
-- This is the convention for
-- multi line comment
---
```

Assignment

- Simple assignment

```
z = 10
```

- Overloaded assignment

```
a, b, c = 0, 1
print(a,b,c)
--> 0   1   nil
```

Variables

- Globals used outside of functions
- Locals limited to the block they are declared in

```
-- Global variable
foobar = 45
-- Local variable
local foobar = 45
```

Functions

- Simple to declare

```
function attackSomeHost(host, port)
    local exploit = 'ATTACK'
    -- do some stuff
    return exploit-result
end
```

Tables

- ALL data structures are tables
  - Arrays
  - Linked Lists
  - Matrices
  - And more!
    - (that I don't completely understand)

```
a = {}     -- new array
for i=1, 1000 do
  a[i] = 0
end
```

If, Then

```
if line > MAXLINES then
  showpage()
  line = 0
end

if op == "+" then
  r = a + b
elseif op == "-" then
  r = a – b
else
  error("invalid operation")
end
```

For Loops

```
for _, cookie in pairs(response.cookies) do
  print cookie
end
```

While Loops

```
local i = 1

while a[i] do
    print(a[i])
    i = i + 1
end
```

And now NSE!

NSE Structure

- The Head
- The Rule
- The Action

```
1  -- The Head
2  local shortport = require shortport
3
4  description = [[
5  Sean Jackson wuz here!!
6  ]]
7
8  categories = {"safe"}
9
10 --The Rule
11 portrule = shortport.http
12
13 -- The Action
14 action = function (host,port)
15   -- Do hacking
16   return stuff
17 end
```

Script Categories

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

NSE Script Requirement!

- Nmap is a port scanner
  - (at it's heart)
- All scripts are connected to an open port
- All results are associated with that port

```
PORT   STATE SERVICE
80/tcp open  http
| get-cookie:
|   Cookie: name=__cfduid; value=de380462ee3364089478de43bd03939cc1489035256; path=/
|   Cookie: name=PHPSESSID; value=6e6slmu5p3sefl9r23dfe23oh6; path=/
|_  Cookie: name=wfvt_809024886; value=58c0dff85f76a; path=/
```

NSE Libraries

- 128 NSE Libraries built into Nmap

afp	json	pcre	socks
brute	ldap	pgsql	ssh1
creds	msrpc	pop3	ssh2
datafiles	mssql	proxy	sslcert
datetime	mysql	rdp	sslv2
dns	netbios	**shortport**	**stdnse**
ftp	**nmap**	smb	tls
geoip	**nsedebug**	smbauth	unpwdb
http	openssl	smtp	vnc
imap	packet	snmp	vulns

Importing Libraries

- In the Head section

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

Building Portrules

```
portrule = function(host, port)
  local auth_port = { number=113, protocol="tcp" }
  local identd = nmap.get_port_state(host, auth_port)

  return identd ~= nil
    and identd.state == "open”
    and port.protocol == "tcp”
    and port.state == "open”
end
```

Portrules Simplified

```
-- Head
local shortport = require "shortport”
-- Rule
portrule = shortport.http
```

First NSE Scripts

Resources for NSE Goodness

- Programming in Lua book
  - https://www.lua.org/pil/contents.html
- Nmap book
  - https://nmap.org/book/nse.html
- NSE Documentation
  - https://nmap.org/nsedoc/
- Check existing scripts
- This presentation
  - https://github.com/tadaka/nse-scripts/

Questions?

Contact Info:

jason@paladinsec.com

@Jason_Wood

805-990-2555