# Christopher D Hopkins

(Hydroplane)

https://github.com/cdhop/nmap101

A security scanner originally written by Gordon Lyon (Fyodor Vaskovich) used to discover hosts and services on a computer network, thus creating a "map" of the network.

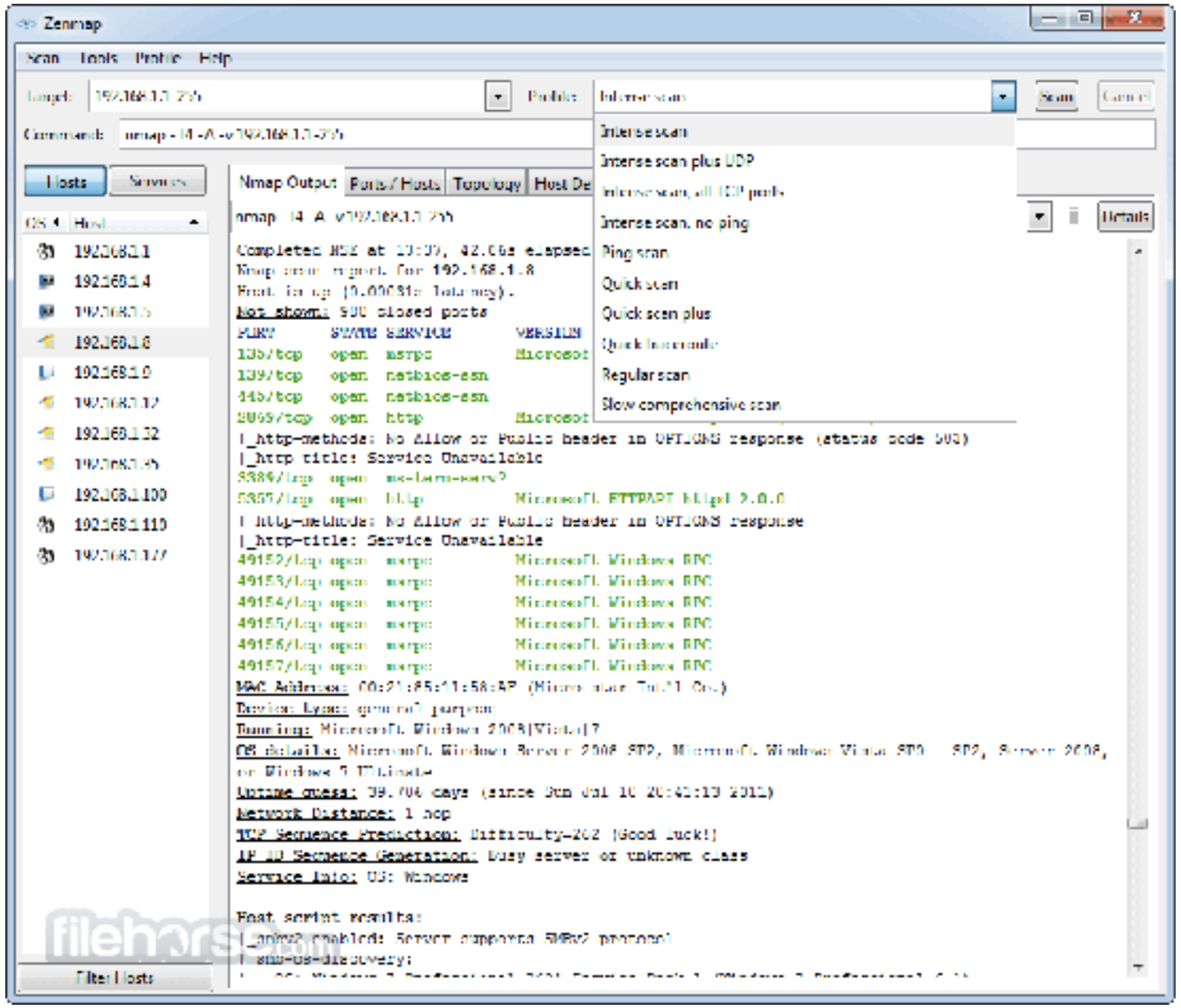
— Wikipedia

- Host Discovery
- Port Scanner
- Version/OS Detection
- Additional functionality through NMAP Scripting Engine (NSE)

- I am not a lawyer
- I am definitely not your lawyer
- To be safe, only scan your own systems, or systems that you have explicit authorization to scan.
- Test Host: **scanme.nmap.org**

- Package Management
- Binaries (Windows/Macintosh)
- Source Code

# GUI (Zenmap)


Zenmap
Scan Tools Profile Help
Target:
Profile:
Intense scan
Scan
Cancel
Command:
nmap -T4 -A -v
Intense scan
Intense scan plus UDP
Intense scan, no ping
Ping scan
Quick scan
Quick scan plus
Regular scan
Slow comprehensive scan
Hosts
Services
Nmap Output
Ports / Hosts
Topology
Host Details
OS
Host
Details
Filter Hosts
Completed NSE at
Not shown: 991 closed ports
PORT STATE SERVICE VERSION
135/tcp open msrpc Microsoft
139/tcp open netbios-ssn
445/tcp open netbios-ssn
2869/tcp open http Microsoft
|_http-methods: No Allow or Public header in OPTIONS response (status code 503)
|_http-title: Service Unavailable
5357/tcp open http Microsoft HTTPAPI httpd 2.0
|_http-methods: No Allow or Public header in OPTIONS response
|_http-title: Service Unavailable
49152/tcp open msrpc Microsoft Windows RPC
49153/tcp open msrpc Microsoft Windows RPC
49154/tcp open msrpc Microsoft Windows RPC
49155/tcp open msrpc Microsoft Windows RPC
49156/tcp open msrpc Microsoft Windows RPC
49157/tcp open msrpc Microsoft Windows RPC
MAC Address:
Device type: general purpose
Running: Microsoft Windows 2008|Vista|7
OS details: Microsoft Windows Server 2008 SP2, Microsoft Windows Vista SP0 - SP2, Server 2008,
Uptime guess:
Network Distance: 1 hop
TCP Sequence Prediction: Difficulty=262 (Good luck!)
IP ID Sequence Generation: Busy server or unknown class
Service Info: OS: Windows
Host script results:
|_smb2-time: Protocol negotiation failed (SMB2)
| smb-os-discovery:

A port is an end point/interface for communication on a system/host available over a network.

Specific port numbers are often used to identify specific services (for example: 80 http, 22 ssh, 443 https, etc)

Ports 1-1024 are considered ‘well-known’ and usually require root/administrator privileges.

There are 65536 (0-65535/16 bits) possible ports.

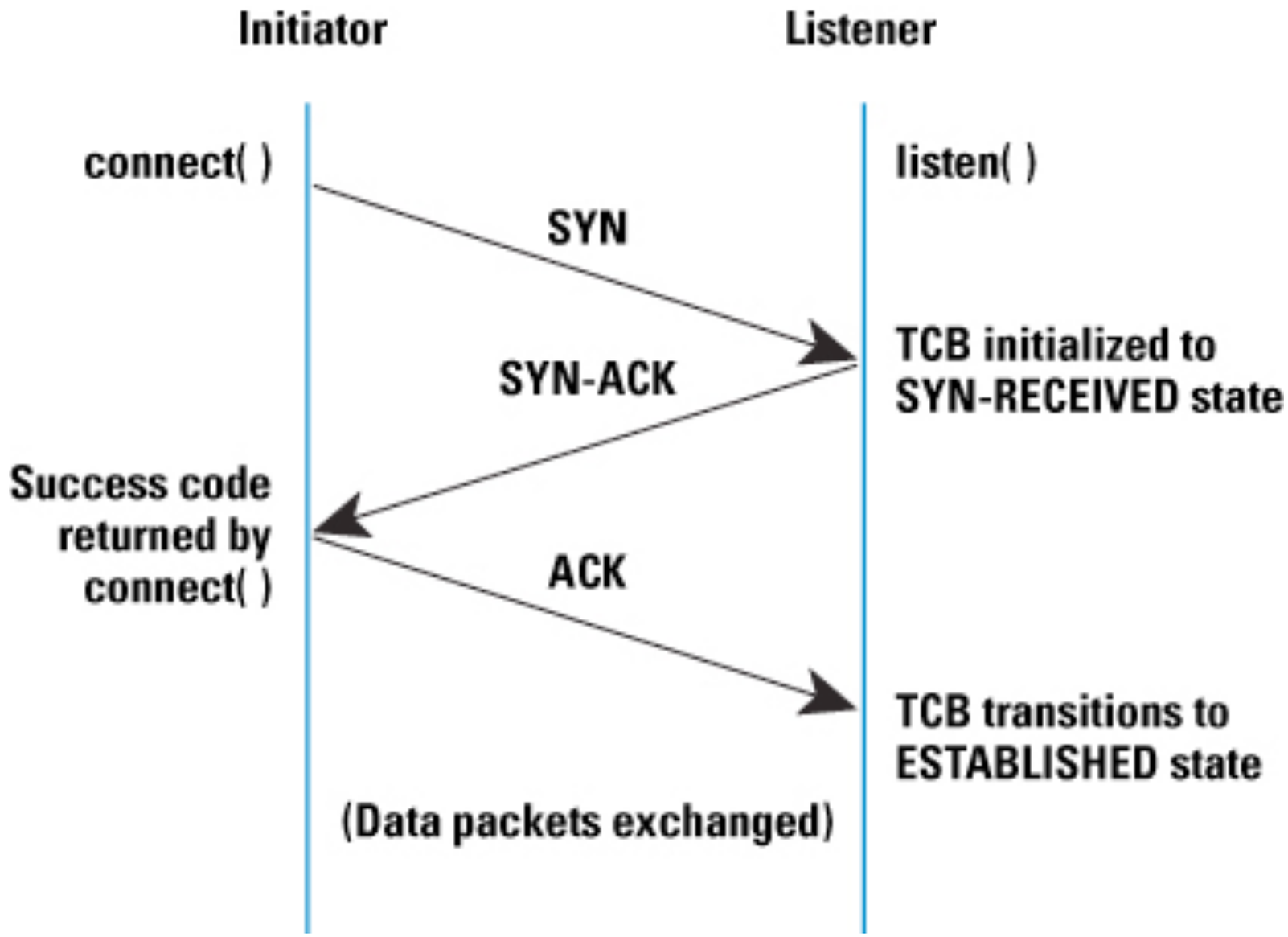
A port that accepts connections is considered to be ‘OPEN’. Conversely, a port that does not accept connections is considered to be ‘CLOSED’.

It may be difficult to conclusively determine the status of a port.

- **Transmission Control Protocol (TCP)**
  ‘guarantees’ delivery of data/packets
  (Examples: http, ssh, smtp).

- **User Datagram Protocol (UDP)**
  provides ‘best effort’ delivery of data/packets
  (Examples: dns, snmp, ntp).

# TCP Three Way Handshake



# Scan Phases

1. Target Enumeration
2. Host Discovery
3. Reverse DNS Resolution
4. Port Scanning
5. Version Detection
6. OS Detection
7. Traceroute
8. Script Scanning
9. Output

**-sn** find hosts that respond to ICMP, http, and/or https (No port scan)

**-Pn** skips Nmap discovery stage altogether (No ping)

**-PR** low-level local network host discovery (ARP ping)

- Parameter: **-sT**
- Can be used by an unprivileged user
- Completes the TCP Three Way Handshake
- Example: **nmap -sT scanme.nmap.org**

- Parameter: **-sS**
- Must be a privileged user
- Sends the SYN packet, then waits for the SYN/ACK
- Faster than Connect Scan
- Example: **nmap -sS scanme.nmap.org**

- Parameter: **-sU**
- Must be a privileged user
- Only way to scan UDP Ports
- Recommend using **-sUV** in order to get more valuable results
- Example: **nmap -sUV scanme.nmap.org**

- Parameters: **-sX** | **-sN**
- Must be a privileged user
- Exploits standards/RFCs
- Usually doesn't work against Windows
- Examples: **nmap -sX scanme.nmap.org**

- Parameter: **-sI**
- To the target it appears that the idle host is performing the port scan
- Stealth Scan
- Recommend disabling host discovery
- Example: **nmap -Pn -sI patsy.host target.host**

- Parameter: **-sV**
- Grabs and displays the service banners
- Increases the confidence in the identification of services
- Example: **nmap -sV scanme.nmap.org**

- Parameter: **-O**
- Scans the target and attempts to detect the OS by comparing it to Nmap's OS fingerprint profiles
- Example: **nmap -O scanme.nmap.org**

- Parameter: **-T(1-5)**
- Increase/Decrease scan speeds
- Faster scans may be unreliable
- Default speed is 3
- Might try slow scan speeds to ‘hide’ a port scan
- Example: **nmap -T4 scanme.nmap.org**

- Parameter: **--host-timeout 1m**
- Helpful with coping with latency
- Example: **nmap scanme.nmap.org --host-timeout 1m**

- Parameters: **-oN** | **-oX** | **-oS** | **-oG** | **-oA**
- Available Formats: Normal, XML, Script Kiddie, Grepable
- Can be used to feed other tools
- Example: **nmap -oN target.nmap target.host**

- An arbitrary scripting framework that allows users to trigger additional checks/actions based on certain open ports or services

- Added to NMAP through a Google Summer of Code in 2006

- There are over 500+ scripts included

- Example: **nmap -p443 --script=ssl-enum-ciphers scanme.nmap.org**

# NSE: Categories

- NSE scripts define a list of categories they belong to.
- Currently defined categories are **auth**, **broadcast**, **brute**, **default**, **discovery**, **dos**, **exploit**, **external**, **fuzzer**, **intrusive**, **malware**, **safe**, **version**, and **vuln**.
- Scripts can/usually belong to more that one category

- Scripts are written in LUA
- Generally have three sections: **head**, **rule**, **action**
- Using the -d option flag can be useful when writing/debuging scripts
- Over 120 standard libraries available

```
description = [[ A simple example NSE script ]]

---
--@output
-- 22/tcp  open  ssh
-- |_simple-example: Open!

author = "hydroplane"
license = "Same as Nmap--See http://nmap.org./book/man-legal.html"
categories = {'safe'}

portrule = function(host, port)
  return port.state == 'open'
end

action = function(host, port)
  return 'Open!'
end
```

# Practical Example

- Homepage: **https://nmap.org**

- **Nmap Network Scanning**: The Official Nmap Project Guide to Network Discovery and Security Scanning (ISBN: 978-0979958717)

- **Nmap Essentials** (ISBN: 978-1783554065)

- **SANS Nmap Cheat Sheet**: https://blogs.sans.org/pen-testing/files/2013/10/NmapCheatSheetv1.0.pdf

# Questions?

# Trinity Uses NMAP