



PREPARAZIONE DI UN ATTACCO IN RETE: “SCANNING”

Corso di Laurea Magistrale in Ingegneria Informatica

A.A. 2015/2016

Prof. Simon Pietro Romano

sromano@unina.it



COME SI PREPARA UN ATTACCO DI RETE?

- Concetti vitali per chiunque si voglia preparare, con cognizione di causa, a sferrare un attacco in una rete di calcolatori:
 - footprinting:
 - l'arte di raccogliere informazioni in rete
 - la cosiddetta “network reconnaissance”
 - scanning:
 - ispezione minuziosa del “perimetro” di attacco, alla ricerca di potenziali punti di ingresso
 - enumeration:
 - ‘probing’ dei servizi identificati, al fine di identificare potenziali vulnerabilità



SCANNING vs FOOTPRINTING

- Footprinting:
 - studio dell'ambiente per raccogliere informazioni ad ampio spettro
- Scanning:
 - perlustrazione meticolosa del “perimetro” di attacco al fine di individuare potenziali punti di accesso ai sistemi dell’organizzazione target



SCANNING: OBIETTIVI

- Verificare quali dei sistemi rilevati in fase di footprinting sono “alive”, vale a dire ‘in ascolto’ di eventuale traffico in ingresso
- Aggirare eventuali firewall per effettuare la ricognizione di sistemi protetti da regole di filtraggio
- Mantenere l’anonimato (e ridurre al minimo il livello di intrusività) ricorrendo a tecniche cosiddette di “*passive scanning*”



RILEVARE SISTEMI “ALIVE”

- Tecnica base: *ping sweep*
 - invio di traffico di uno specifico tipo verso un host target e successiva analisi dei risultati
 - NB:
 - Il termine “ping” è storicamente associato all’invio di messaggi ICMP (Internet Control Message Protocol) di tipo *Echo Request*, seguito dalla ricezione di messaggi ICMP di tipo *Echo Reply*
 - Oggi questo termine si è evoluto e sta ad indicare, genericamente, l’invio di messaggi appartenenti, oltre ad ICMP, a protocolli quali ARP (Address Resolution protocol), TCP (Transmission Control Protocol) ed UDP (Universal Datagram Protocol)



arp-scan: SCOPERTA DI HOST SU RETE LOCALE

- Utile quando l'attaccante si trova sulla stessa rete locale del target
- Restituisce informazioni sugli host attivi:
 - indirizzo IP
 - indirizzo MAC
 - produttore della scheda di rete locale

```
root@kali:~# arp-scan 143.225.28.128/25
Interface: eth0, datalink type: EN10MB (Ethernet)
Starting arp-scan 1.9 with 128 hosts (http://www.nta-monitor.com/tools/arp-scan/)

143.225.28.130 00:19:cb:46:ec:da ZyXEL Communications Corporation
143.225.28.136 18:03:73:b2:4f:7f Dell Inc
143.225.28.134 00:1a:4b:0d:69:bb Hewlett-Packard Company
143.225.28.135 00:22:64:b2:16:96 Hewlett-Packard Company
143.225.28.139 10:dd:b1:ef:47:21 Apple
143.225.28.148 00:1f:f3:3e:3e:2d Apple, Inc.
143.225.28.157 00:19:99:cc:91:77 Fujitsu Technology Solutions
143.225.28.167 98:5a:eb:d1:73:ee (Unknown)
143.225.28.169 40:6c:8f:3c:31:e3 Apple, Inc.
143.225.28.175 68:5b:35:98:51:58 Apple inc
143.225.28.177 b8:ca:3a:77:b8:fc Dell PCBA Test
143.225.28.179 9c:93:4e:5e:0a:e0 Xerox Corporation
143.225.28.178 00:09:aa:9e:34:c9 XEROX CORPORATION
143.225.28.180 08:60:6e:48:69:37 ASUSTek COMPUTER INC. type: Ethernet
143.225.28.187 5c:f9:dd:ea:4b:7a Dell Inc
143.225.28.192 00:3f:49:ac:d9:ec (Unknown) hardware size: 0
143.225.28.193 00:24:8c:03:0d:e4 ASUSTek COMPUTER INC. size: 4
143.225.28.196 4c:72:b9:da:98:e6 Pegatron Corporation request (?)
143.225.28.197 00:08:9b:d9:85:f8 ICP Electronics Inc. MAC address: CadmusCo_7f:fb:60
143.225.28.200 00:0c:29:ea:93:ae VMware, Inc. Sender IP address: 172.17.0.123 (172.17
143.225.28.201 28:92:4a:38:f6:c2 Hewlett Packard Target MAC address: 00:00:00:00:00:00
143.225.28.202 28:92:4a:2d:59:93 Hewlett Packard Target IP address: 172.17.0.123 (172.17
143.225.28.203 00:0c:29:6f:54:0c VMware, Inc.
143.225.28.204 c8:cb:08:ce:6f:36 Hewlett Packard
143.225.28.208 68:5b:35:97:c8:ac Apple inc
143.225.28.210 20:c9:d0:11:34:60 Apple Inc
143.225.28.217 90:72:40:00:f2:29 Apple
143.225.28.219 68:5b:35:8d:9b:fb Apple inc
143.225.28.220 00:15:6d:4f:a8:31 Ubiquiti Networks Inc.
143.225.28.229 00:15:f2:69:01:1a ASUSTek COMPUTER INC.
143.225.28.233 c8:60:00:8b:6c:e2 ASUSTek COMPUTER INC.
143.225.28.234 54:04:a6:46:fa:76 ASUSTek COMPUTER INC.
143.225.28.237 b8:ca:3a:9d:d6:f9 Dell PCBA Test
143.225.28.240 08:00:37:cb:6c:dd FUJI-XEROX CO. LTD. with a printer's name: Xerox WorkCentre 5000
143.225.28.244 00:04:00:49:cc:0b LEXMARK INTERNATIONAL, INC.
143.225.28.250 00:17:c8:07:88:4e KYOCERA Document Solutions Inc.
143.225.28.249 00:21:5a:e5:ba:7e Hewlett-Packard Company
143.225.28.254 00:17:df:b3:c4:00 CISCO SYSTEMS, INC.

38 packets received by filter, 0 packets dropped by kernel
Ending arp-scan 1.9: 128 hosts scanned in 1.636 seconds (78.24 hosts/sec). 38 responded
```



ARP SCANNING CON “nmap”

- nmap: Network Mapper
 - un programma molto potente per la scoperta di topologie di rete
 - capace di “mappare” nodi di rete e relativi servizi
 - supporta lo “scanning” con protocollo ARP tramite l’opzione “-PR”
 - si limita alla sola “scoperta” degli host tramite l’opzione “-sn”
 - ...in realtà può fare molto più di questo (cfr. slide successive)

```
root@kali:~# nmap -sn -PR 143.225.28.128/25
Starting Nmap 6.49BETA4 ( https://nmap.org ) at 2015-09-28 11:08 EDT
Nmap scan report for 143.225.28.130
Host is up (0.0018s latency).
MAC Address: 00:19:CB:46:EC:DA (ZyXEL Communications)
Nmap scan report for 143.225.28.134
Host is up (0.00066s latency).
MAC Address: 00:1A:4B:8D:69:BB (Hewlett-Packard Company)
Nmap scan report for 143.225.28.135
Host is up (0.00045s latency).
MAC Address: 00:22:64:B2:16:96 (Hewlett-Packard Company)
Nmap scan report for 143.225.28.136
Host is up (0.00037s latency).
MAC Address: 18:03:73:B2:4F:7F (Dell)
Nmap scan report for 143.225.28.139
Host is up (0.00026s latency).
MAC Address: 10:DD:B1:EF:47:21 (Apple)
Nmap scan report for 143.225.28.140
Host is up (0.0011s latency).
MAC Address: 00:1F:F3:E3:2D (Apple)
Nmap scan report for 143.225.28.157
Host is up (0.00059s latency).
MAC Address: 00:19:99:CC:91:77 (Fujitsu Technology Solutions)
Nmap scan report for 143.225.28.167
Host is up (-0.10s latency).
MAC Address: 98:5A:EB:D1:73:EE (Apple)
Nmap scan report for 143.225.28.169
Host is up (-0.10s latency).
MAC Address: 40:6C:8F:3C:31:E3 (Apple)
Nmap scan report for 143.225.28.175
Host is up (-0.10s latency).
MAC Address: 68:58:35:98:51:58 (Apple)
Nmap scan report for 143.225.28.177
Host is up (-0.10s latency).
MAC Address: B8:CA:3A:77:88:FC (Dell)
Nmap scan report for 143.225.28.178
Host is up (-0.099s latency).
MAC Address: 00:00:AA:9E:34:CF (Xerox)
Nmap scan report for 143.225.28.179
Host is up (-0.100s latency).
MAC Address: 9C:93:4E:5E:0A:E0 (Xerox)
Nmap scan report for 143.225.28.180
Host is up (-0.10s latency).
MAC Address: 00:60:6E:48:69:37 (Asustek Computer)
Nmap scan report for 143.225.28.187
Host is up (-0.100s latency).
MAC Address: 5C:F9:DD:EA:4B:7A (Dell)
Nmap scan report for 143.225.28.192
Host is up (0.00032s latency).
```



SCOPERTA DI HOST REMOTI

- ARP funziona SOLO su rete locale
- Nel caso di host remoti, si ricorre a protocolli di più alto livello:
 - ICMP
 - TCP/UDP
- Anche in caso di host remoti, i tool disponibili abbondano...



ICMP HOST DISCOVERY

- Il più classico dei programmi di utilità: *ping*
 - invio di un messaggio *ICMP Echo request...*
 - ...ricezione di un messaggio di tipo *ICMP Echo Reply*

```
root@kali:~# ping -c 4 143.225.28.167
PING 143.225.28.167 (143.225.28.167) 56(84) bytes of data.
64 bytes from 143.225.28.167: icmp_seq=1 ttl=64 time=0.245 ms
64 bytes from 143.225.28.167: icmp_seq=2 ttl=64 time=0.171 ms
64 bytes from 143.225.28.167: icmp_seq=3 ttl=64 time=0.198 ms
64 bytes from 143.225.28.167: icmp_seq=4 ttl=64 time=0.192 ms

--- 143.225.28.167 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 2997ms
rtt min/avg/max/mdev = 0.171/0.201/0.245/0.030 ms
```



IL SOLITO *nmap*...

- Opzioni da usare:
 - “**-sn**” → “no port scan”
 - “**-PE**” → invia un messaggio ICMP Echo Request
 - “**--send-ip**” → non inviare pacchetti ARP
- NB: in assenza di tali opzioni (e se eseguito come utente “root”), nmap farebbe anche le seguenti cose:
 - ARP ping, invio di un messaggio ICMP Timestamp request, TCP pinging sulle porte 80 e 443!

```
root@kali:~# nmap -sn -PE --send-ip 143.225.28.167
Starting Nmap 6.49BETA4 ( https://nmap.org ) at 2015-09-28 13:39 EDT
Nmap scan report for 143.225.28.167
Host is up (0.00015s latency).
MAC Address: 98:5A:EB:D1:73:EE (Apple)
Nmap done: 1 IP address (1 host up) scanned in 13.21 seconds
```



nping: IL PING DEGLI HACKER!

- Consente lo “spoofing” dell’indirizzo MAC sorgente, dell’indirizzo IP sorgente e di qualsiasi altro campo del pacchetto
- Può essere configurato per inviare specifici messaggi ICMP (ad esempio, “Timestamp” request)...

```
root@kali:~# nping -c 4 --icmp --icmp-type time 143.225.28.254

Starting Nping 0.6.49BETA4 ( http://nmap.org/nping ) at 2015-09-28 13:54 EDT
SENT (0.0022s) ICMP [143.225.28.168 > 143.225.28.254 Timestamp request (type=13/code=0) id=32993 seq=1 orig=0 recv=0 trans=0] IP [ttl=64 id=18556 iplen=40 ]
RCVD (0.1983s) ICMP [143.225.28.254 > 143.225.28.168 Timestamp reply (type=14/code=0) id=32993 seq=1 orig=0 recv=64525994 trans=64525994] IP [ttl=255 id=18556 iplen=40 ]
SENT (1.0028s) ICMP [143.225.28.168 > 143.225.28.254 Timestamp request (type=13/code=0) id=32993 seq=2 orig=0 recv=0 trans=0] IP [ttl=64 id=18556 iplen=40 ]
RCVD (1.2031s) ICMP [143.225.28.254 > 143.225.28.168 Timestamp reply (type=14/code=0) id=32993 seq=2 orig=0 recv=64526995 trans=64526995] IP [ttl=255 id=18556 iplen=40 ]
SENT (2.0033s) ICMP [143.225.28.168 > 143.225.28.254 Timestamp request (type=13/code=0) id=32993 seq=3 orig=0 recv=0 trans=0] IP [ttl=64 id=18556 iplen=40 ]
RCVD (2.2025s) ICMP [143.225.28.254 > 143.225.28.168 Timestamp reply (type=14/code=0) id=32993 seq=3 orig=0 recv=64527995 trans=64527995] IP [ttl=255 id=18556 iplen=40 ]
SENT (3.0045s) ICMP [143.225.28.168 > 143.225.28.254 Timestamp request (type=13/code=0) id=32993 seq=4 orig=0 recv=0 trans=0] IP [ttl=64 id=18556 iplen=40 ]
RCVD (3.2031s) ICMP [143.225.28.254 > 143.225.28.168 Timestamp reply (type=14/code=0) id=32993 seq=4 orig=0 recv=64528996 trans=64528996] IP [ttl=255 id=18556 iplen=40 ]

Max rtt: 200.074ms | Min rtt: 196.240ms | Avg rtt: 198.479ms
Raw packets sent: 4 (160B) | Rcvd: 4 (184B) | Lost: 0 (0.00%)
Nping done: 1 IP address pinged in 3.20 seconds
```



TCP/UDP HOST DISCOVERY

- Utile nei casi in cui il protocollo ICMP risulti, per motivi di sicurezza, filtrato
 - Un firewall che protegge un server web potrebbe filtrare i pacchetti ICMP ad esso indirizzati...
 - ...ma dovrebbe necessariamente lasciare passare i segmenti TCP diretti alla porta 80
 - un hacker può quindi effettuare il “probing” contattando TCP (sulla porta 80) per determinare se l’host in questione è “alive”
 - Ovviamente non è facile “indovinare” a priori quali servizi siano attivi su un host di cui si vuole conoscere lo stato:
 - invio “cieco” di segmenti TCP indirizzati a numeri di porta variabili sull’host target
 - attività dispendiosa in termini di tempo e tutt’altro che “silenziosa”:
 - il traffico generato difficilmente sfugge ad un sistema di Intrusion Detection ben configurato!



PORT PROBING CON *nmap*

- Con l'opzione “***-Pn***”, nmap effettua il probing su ben 1000 porte di potenziale interesse!

```
root@kali:~# nmap -Pn 143.225.28.169
Starting Nmap 6.49BETA4 ( https://nmap.org ) at 2015-09-28 14:09 EDT
Nmap scan report for 143.225.28.169
Host is up (0.00077s latency).
Not shown: 969 closed ports, 26 filtered ports
PORT      STATE SERVICE
80/tcp    open  http
111/tcp   open  rpcbind
443/tcp   open  https
1023/tcp  open  netvenuechat
2049/tcp  open  nfs
MAC Address: 40:6C:8F:3C:31:E3 (Apple)

Nmap done: 1 IP address (1 host up) scanned in 97.63 seconds
```



PROBING DI UNA SINGOLA PORTA

- Soluzione maggiormente “scalabile”
 - In *nmap*:
 - impiego dell’opzione:
“-sS -p [#porta] –open”
- Realizzabile anche con tool quali *nping* o *SuperScan*
 - <http://www.foundstone.com/>

```
root@kali:~# nmap -Pn -sS -p 22 --open 143.225.28.128/25
Starting Nmap 6.49BETA4 ( https://nmap.org ) at 2015-09-28 14:16 EDT
Nmap scan report for 143.225.28.139
Host is up (-0.076s latency).
PORT      STATE SERVICE
22/tcp    open  ssh
MAC Address: 10:DD:B1:EF:47:21 (Apple)

Nmap scan report for 143.225.28.187
Host is up (-0.077s latency).
PORT      STATE SERVICE
22/tcp    open  ssh
MAC Address: 5C:F9:DD:EA:4B:7A (Dell)
```

```
Nmap scan report for 143.225.28.220
Host is up (-0.076s latency).
PORT      STATE SERVICE
22/tcp    open  ssh
MAC Address: 00:15:6D:4F:AE:31 (Ubiquiti Networks)

Nmap scan report for 143.225.28.254
Host is up (-0.070s latency).
PORT      STATE SERVICE
22/tcp    open  ssh
MAC Address: 00:17:DF:B3:C4:00 (Cisco Systems)

Nmap done: 128 IP addresses (37 hosts up) scanned in 42.32 seconds
```



PING SWEEP: DETECTION

- In rete:
 - impiego di sistemi di Intrusion Detection network-based
 - es: *Snort*: www.snort.org
- Sugli host:
 - impiego di tool per il rilevamento (ed il “logging”) di attività sospette indirizzate al nodo
 - es: *scanlogd*, *ippl*, *Protolog*



PING SWEEP: PREVENZIONE

- Filtrare il tipo di messaggi ICMP consentiti:
 - es: solo Echo Request, Echo Reply, Host Unreachable e Time Exceeded e solo se indirizzati ad (alcuni) host della DMZ (“DeMilitarized Zone”)
 - Access Control Lists (ACL) per consentire l’impiego di ICMP solo ad un insieme predefinito di nodi esterni
- Usare ICMP come applicativo utente (“userland daemon”)
 - es: *pingd* → gestisce i messaggi ICMP al livello host (cioè fuori dal kernel)
- Attenzione!
 - ICMP, su un host “compromesso”, può diventare una “back door” a livello di sistema operativo:
 - “covert channel” per il tunneling di dati generici all’interno di pacchetti ICMP:
 - es: *loki2* → <http://phrack.org/issues/51/6.html>

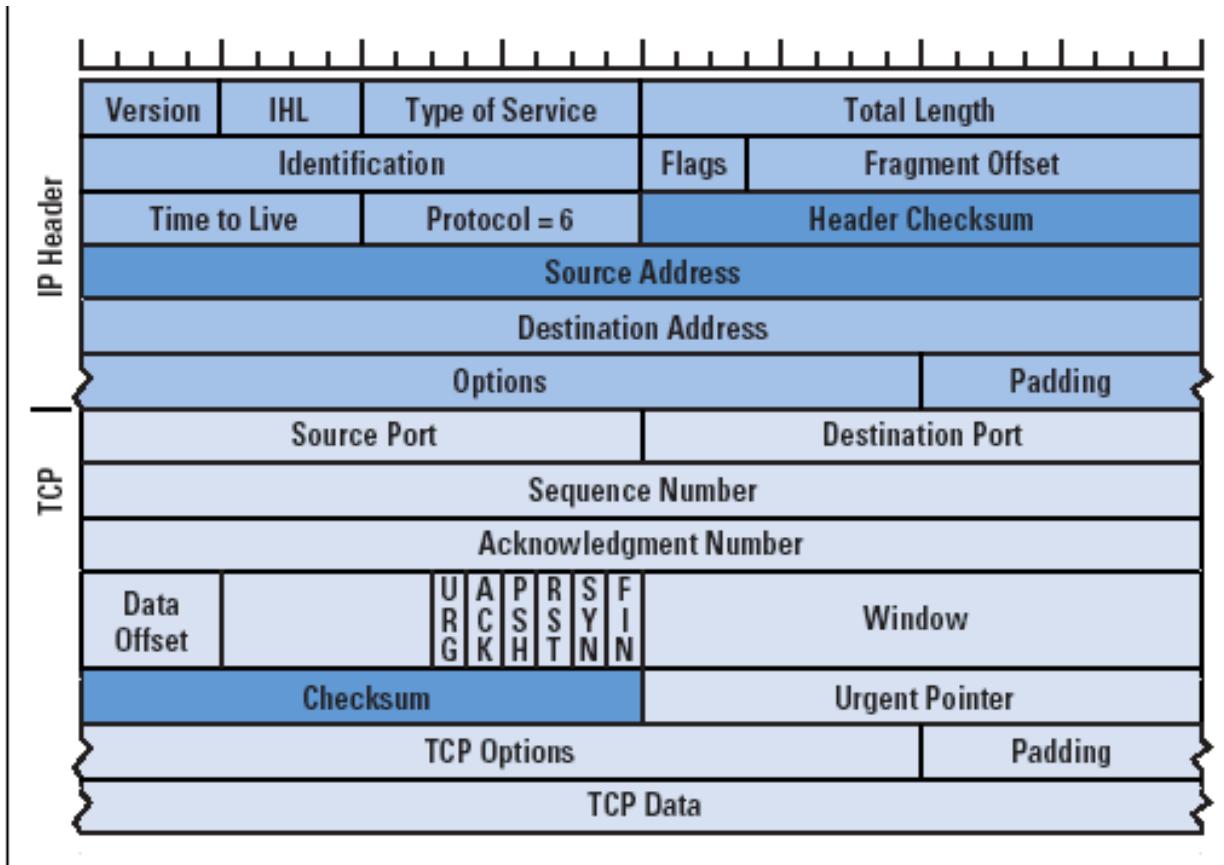


ALLA SCOPERTA DEI SERVIZI ATTIVI

- Port scanning
 - attività di “probing” dei nodi di rete al fine di determinare servizi in esecuzione o in ascolto su di essi
 - tipicamente realizzata tramite l’invio di pacchetti indirizzati alle porte TCP ed UDP maggiormente diffuse
 - cruciale per determinare, a partire dai servizi, le potenziali vulnerabilità di un nodo di rete
 - utile anche per determinare altri tipi di informazioni:
 - tipo e versione del Sistema Operativo
 - applicazioni in uso...



RINFRESCHIAMOCI LA MEMORIA...





TIPI DI SCANSIONE (1/3)

- TCP connect scan:
 - effettua l'intero ciclo del 3-way handshake (SYN, SYN+ACK, ACK)
 - utile quando la scansione deve essere effettuata come utente non privilegiato
 - più lento e più facilmente tracciabile rispetto alle altre soluzioni
- TCP SYN scan:
 - non completa l'handshake: invia SYN e attende SYN+ACK, senza inviare ACK finale
 - utile per determinare la presenza di servizi “in ascolto”
 - servizio assente (su quella porta) se ricevo RST + ACK dal nodo remoto
 - più robusta (“stealthy”) rispetto alla precedente
 - spesso non catturata nei log di sistema del target
 - NB: potenziale causa di una condizione di Denial of Service sul target:
 - elevato numero di connessioni “mezze aperte”!



TIPI DI SCANSIONE (2/3)

- TCP FIN scan:
 - invia un segmento FIN ad una specifica porta dell'host target
 - RFC 793 → il sistema target 'dovrebbe' rispondere con un segmento RST se la porta in questione è chiusa
 - NB: di solito questa tecnica funziona solo per stack TCP/IP Unix-based
- TCP Xmas Tree scan:
 - Invia segmento con bit SYN, FIN, URG e PUSH alti
 - RFC 793 → il sistema target 'dovrebbe' rispondere con un segmento RST
- TCP Null scan:
 - Invia un segmento TCP con tutti i flag a 0
 - RFC 793 → il sistema target 'dovrebbe' rispondere con un segmento RST
- TCP ACK scan:
 - invia un segmento TCP con il bit ACK alto
 - utile per identificare firewall che implementano semplici regole di filtraggio
 - dati appartenenti a connessioni stabilite (ACK = 1) sono ammessi, senza ulteriore 'ispezione' del pacchetto in ingresso



TIPI DI SCANSIONE (3/3)

- TCP Window scan:
 - utile solo per sistemi tipo FreeBSD o AIX
 - sfrutta un'anomalia nel modo di riportare la dimensione della finestra
 - invia un ACK e si aspetta di ricevere un RST:
 - RST con WIN = 0 → porta chiusa
 - RST con WIN > 0 → porta aperta!
- TCP RPC scan:
 - specifico per sistemi UNIX
 - rileva ed identifica servizi del tipo Remote Procedure Call (RPC)
- UDP scan:
 - invia un pacchetto UDP ad una specifica porta sul target:
 - se riceve un “ICMP Port Unreachable” → porta chiusa
 - molto meno affidabile rispetto alle tecniche basate su TCP, a causa della elevata probabilità che il traffico UDP sia filtrato dalla rete/host destinazione



PORT SCANNING CON *nmap*

- TCP SYN scan...

```
root@kali:~# nmap -sS 143.225.28.169

Starting Nmap 6.49BETA4 ( https://nmap.org ) at 2015-09-29 05:45 EDT
Nmap scan report for 143.225.28.169
Host is up (0.00073s latency).
Not shown: 929 closed ports, 66 filtered ports
PORT      STATE SERVICE
80/tcp    open  http
111/tcp   open  rpcbind
443/tcp   open  https
1023/tcp  open  netvenuechat
2049/tcp  open  nfs
MAC Address: 40:6C:8F:3C:31:E3 (Apple)

Nmap done: 1 IP address (1 host up) scanned in 94.74 seconds
```

- TCP SYN scan con salvataggio dell'output su file

```
root@kali:~# nmap -sF 143.225.28.169 -oX pippozzo.xml

Starting Nmap 6.49BETA4 ( https://nmap.org ) at 2015-09-29 05:56 EDT
Nmap scan report for 143.225.28.169
Host is up (0.00013s latency).
All 1000 scanned ports on 143.225.28.169 are open|filtered
MAC Address: 40:6C:8F:3C:31:E3 (Apple)

Nmap done: 1 IP address (1 host up) scanned in 21.54 seconds
```



SCANNING CON 'DIVERSIVO'...

- Impiego dell'opzione cosiddetta di "decoy":

The -D option allows you to specify Decoys. This option makes it look like those decoys are scanning the target network. It does not hide your own IP, but it makes your IP one of a torrent of others supposedly scanning the victim at the same time. This not only makes the scan look more scary, but reduces the chance of you being traced from your scan (difficult to tell which system is the "real" source).

- Richiede "spoofing" di un indirizzo IP valido, per evitare di inondare di segmenti SYN il sistema target
- Con questo tipo di scan, per il sistema target risulta difficile identificare il reale artefice della scansione, i cui dati si confondono con quelli del nodo utilizzato come 'diversivo'

Sorgente della scansione

```
root@kali:~# nmap -sS 143.225.28.167 -D 143.225.28.169
Starting Nmap 6.49BETA4 ( https://nmap.org ) at 2015-09-29 09:46 EDT
Nmap scan report for 143.225.28.167
Host is up (0.00023s latency).
All 1000 scanned ports on 143.225.28.167 are closed
MAC Address: 98:5A:EB:D1:73:EE (Apple)

Nmap done: 1 IP address (1 host up) scanned in 69.42 seconds
```

Host "esca"

Source	Destination	Protocol	Length	Info
143.225.28.167	143.225.28.169	TCP	60	6565..36606 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
143.225.28.167	143.225.28.169	TCP	60	100..36607 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
143.225.28.167	143.225.28.169	TCP	60	50001..36606 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
143.225.28.167	143.225.28.169	TCP	60	3390..36606 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
143.225.28.167	143.225.28.169	TCP	60	5102..36606 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
143.225.28.167	143.225.28.169	TCP	60	16993..36606 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
143.225.28.167	143.225.28.169	TCP	60	5102..36607 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
143.225.28.167	143.225.28.169	TCP	60	5679..36606 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
143.225.28.167	143.225.28.169	TCP	60	5999..36606 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
143.225.28.167	143.225.28.169	TCP	60	5405..36606 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
143.225.28.167	143.225.28.169	TCP	60	3001..36606 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
143.225.28.167	143.225.28.169	TCP	60	5405..36607 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
143.225.28.167	143.225.28.169	TCP	60	4005..36606 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0



UN ALTRO TOOL: *netcat*

- Il “coltellino svizzero” della sicurezza
- Utile quando si intende ridurre al minimo le proprie tracce in un sistema compromesso
- Scanning basato su TCP o su UDP (opzione “-u”)
- Moltissime opzioni di configurazione...

```
root@kali:~# netcat -help
[vl.10-41]
connect to somewhere: nc [-options] hostname port[s] [ports] ...
listen for inbound:   nc -l -p port [-options] [hostname] [port]
options:
  -c shell commands
  -e filename
  -b
  -g gateway
  -G num
  -h
  -i secs
  -k
  -l
  -n
  -o file
  -p port
  -r
  -q secs
  -s addr
  -T tos
  -t
  -u
  -v
  -w secs
  -C
  -z
as '-e'; use /bin/sh to exec [dangerous!!]
program to exec after connect [dangerous!!]
allow broadcasts
source-routing hop point[s], up to 8
source-routing pointer: 4, 8, 12, 16...
this crust
delay interval for lines sent, ports scanned
set keepalive option on socket
listen mode, for inbound connects
numeric-only IP addresses, no DNS
hex dump of traffic
local port number
randomize local and remote ports
quit after EOF on stdin and delay of secs
local source address
set Type Of Service
answer TELNET negotiation
UDP mode
verbose [use twice to be more verbose]
timeout for connects and final net reads
Send CRLF as line-ending
zero-I/O mode [used for scanning]
port numbers can be individual or ranges: lo-hi [inclusive];
hyphens in port names must be backslash escaped (e.g. 'ftp\-\data').ents Down
```



PORT SCANNING: CONTROMISURE

- Al solito:
 - Detection:
 - impiego di un Network-based Intrusion Detection System alla Snort (www.snort.org)
 - impiego di tool per l'analisi dei log file: es: scanlogd
 - NB: attenzione al fatto che gli indirizzi IP dei supposti attaccanti sono tipicamente forgiati ad arte (spoofing)...
 - ...prendere contromisure contro tali indirizzi quasi sempre equivale a prendersela con la "persona" sbagliata!
 - Prevention:
 - Qui c'è veramente poco da fare (difficile convincere un hacker della inutilità di effettuare una scansione dei nostri nodi di rete)
 - Unico consiglio utile: disattivare TUTTI i servizi ritenuti non necessari!



SCOPERTA DEL TIPO DI SISTEMA OPERATIVO

- Dalle tecniche più semplici...
 - es: "banner-grabbing" → cfr. lezione sul *footprinting*
- ...a quelle più avanzate:
 - il cosiddetto "stack fingerprinting":
 - ricerca, a partire dai servizi disponibili su un nodo (cfr. port scanning), di segni evidenti della presenza di un determinato tipo di Sistema Operativo
 - Esempi:
 - porte 135 (Endpoint Mapper), 139 (NetBIOS) e 445 (Active Directory) attive → Windows!
 - porte 22 (ssh), 111 (SUN RPC), 2049 (NFS) attive → elevata probabilità che si tratti di un sistema Unix-based!
 - basato sull'impiego sia di tecniche attive che di tecniche passive



ACTIVE STACK FINGERPRINTING

- Insieme di tecniche volte a determinare in maniera veloce ed affidabile il tipo di Sistema Operativo (SO) di cui è dotato un nodo di rete
- Impiego di conoscenze dettagliate sulle specifiche implementazioni dello stack TCP/IP standard di Internet (RFC) da parte dei vari produttori di SO
- Necessità di fare affidamento su ALMENO una porta aperta sul nodo per ottenere stime affidabili

```
---[ Phrack Magazine  Volume 8, Issue 54 Dec 25th, 1998, article 09 of 12

-----[ Remote OS detection via TCP/IP Stack FingerPrinting

-----[ Fyodor <fyodor@dhp.com> (www.insecure.org) October 18, 1998

----[ ABSTRACT

This paper discusses how to glean precious information about a host by querying its TCP/IP stack. I first present some of the "classical" methods of determining host OS which do not involve stack fingerprinting. Then I describe the current "state of the art" in stack fingerprinting tools. Next comes a description of many techniques for causing the remote host to leak information about itself. Finally I detail my (nmap) implementation of this, followed by a snapshot gained from nmap which discloses what OS is running on many popular Internet sites.
```



PROBE PER STACK FINGERPRINTING (1/2)

- FIN probe:
 - invio di un segmento FIN verso una porta aperta:
 - RFC793 → NON rispondere!
 - Alcune implementazioni di Windows (7, 200X, Vista) → invia FIN+ACK ☹
- Bogus flag probe:
 - configurazione di un flag non definito nell'header di un segmento TCP SYN
 - Linux → ricopia il flag in questione nel segmento di risposta (SYN+ACK) ☹
- “Don’t Fragment bit” monitoring:
 - alcuni SO ‘settano’ tale bit (nell’header IP) di default, per incrementare le prestazioni
- TCP initial window size:
 - alcune implementazioni di TCP associano un valore costante alla finestra di ricezione
- ACK value:
 - RFC793 → “Sequence # + 1”
 - Alcune implementazioni di TCP → “Sequence #” ☹



PROBE PER STACK FINGERPRINTING (2/2)

- ICMP message quoting:
 - diversi SO “ricopiano” parti diverse del messaggio originario, quando costruiscono un messaggio ICMP di errore
- Type of Service (TOS):
 - analisi del TOS dei pacchetti ICMP “Port Unreachable” ricevuti (dovrebbe essere ‘0’, ma alcuni SO lo configurano in maniera diversa...)
- Fragmentation Handling:
 - diversi SO implementano in maniera diversa la ricostruzione di datagrammi IP a partire da frammenti “sovraposti”
- TCP Options:
 - RFC1323 → definisce NUOVE opzioni per TCP (“no operation”, “window scale”, ecc)...
 - ...il che ci consente di distinguere stack più recenti da stack meno recenti (compatibili solo con RFC793)



OS DETECTION CON *nmap*

- Impiego dell'opzione “**-O**”, basata sull'utilizzo della maggior parte delle tecniche di stack fingerptinting menzionate

```
root@kali:/etc/snort/rules# nmap -O 143.225.28.169
Starting Nmap 6.49BETA4 ( https://nmap.org ) at 2015-09-30 02:42 EDT
Nmap scan report for 143.225.28.169
Host is up (0.00081s latency).
Not shown: 995 closed ports
PORT      STATE SERVICE
80/tcp    open  http
111/tcp   open  rpcbind
443/tcp   open  https
1023/tcp  open  netvenuechat
2049/tcp  open  nfs
MAC Address: 40:6C:8F:3C:31:E3 (Apple)
Device type: general purpose|media device|phone
Running: Apple Mac OS X 10.7.X|10.9.X|10.8.X, Apple iOS 4.X|5.X|6.X
OS CPE: cpe:/o:apple:mac_os_x:10.7 cpe:/o:apple:mac_os_x:10.9 cpe:/o:apple:mac_os_x:10.8 cpe:/o:apple:iphone_os:4
cpe:/a:apple:apple_tv:4 cpe:/o:apple:iphone_os:5 cpe:/o:apple:iphone_os:6
OS details: Apple Mac OS X 10.7.0 (Lion) - 10.10 (Yosemite) or iOS 4.1 - 8.1.2 (Darwin 10.0.0 - 14.0.0)
Network Distance: 1 hop

OS detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 51.39 seconds
```



OS DETECTION: CONTROMISURE

- Detection:
 - i “consigli” relativi al rilevamento di attività generiche di port scanning valgono anche in questo caso
 - scansioni con particolari opzioni configurate nei pacchetti sonda (es: SYN flag in TCP) sono un buon indicatore della presenza di attività di “stack fingerprinting”
- Prevention:
 - Molto complicata!
 - prevenire, in questo caso, richiede di modificare il comportamento del Sistema Operativo (a livello di codice sorgente o di parametri di configurazione)
 - L'unica vera forma di prevenzione consiste nel fare affidamento sulla presenza di firewall e/o proxy “robusti” a protezione delle risorse di rete dei propri sistemi



PASSIVE STACK FINGERPRINTING

- Tecniche concepite per rendere l'attività di fingerprinting più robusta rispetto alla possibilità di rilevamento da parte di eventuali IDS
- Evitano di inviare, in maniera proattiva, pacchetti sonda verso i nodi target
- Fanno affidamento solo sul monitoraggio e sull'analisi del traffico di rete
- Richiedono che l'attaccante:
 - sia posizionato in un punto "centrale" della rete
 - sia capace di "ascoltare" il traffico su una porta che consenta la cattura dei pacchetti ("mirrored port")
- Alcuni esempi di progetti e di tool per il fingerprinting passivo:
 - progetto "*honeynet*" → <http://honeynet.org/>
 - tool "*sypphon*" → port mapping passivo, OS identification, topology discovery



PASSIVE SIGNATURES

- Identificazione di specifiche “caratteristiche” del traffico monitorato
 - Time To Live (TTL)
 - differenti SO usano differenti valori di default
 - TCP Window size
 - differenti SO usano differenti valori di default per la finestra “iniziale” annunciata da un ricevitore TCP
 - Don’t Fragment (DF) bit
 - alcuni SO lo configurano alto di default, altri no
- Analizzando il traffico e comparando i risultati dell’analisi con specifiche “signature” presenti in una base di dati costruita ad hoc, è possibile identificare il SO dei nodi che hanno generato i dati sottoposti a monitoraggio



CONSERVARE ED ELABORARE I RISULTATI

- Attività fondamentale per riuscire ad analizzare in maniera strutturata tutte le informazioni raccolte durante le fasi di scanning
- Tipicamente realizzata tramite importazione dei dati delle scansioni in una apposita base di dati
- Il database delle scansioni diventa una repository preziosa di dati su cui effettuare elaborazioni utili:
 - alla estrapolazione di un profilo completo del sistema target
 - alla individuazione di potenziali vulnerabilità tramite attività di inferenza e di correlazione delle informazioni
 - alla preparazione delle successive fasi di attacco tramite stesura di un piano strutturato di azioni “offensive”



GESTIONE DEI DATI CON “Metasploit”

- Una piattaforma completa per la messa a punto di attacchi di rete
- Un'impressionante collezione di:
 - tool di fingerptinting
 - “payload” di attacco
 - “exploit” di sistemi di rete
- Capace di importare i dati in un database sul quale effettuare query specifiche per:
 - lo studio organico dei sistemi target
 - l'individuazione di potenziali pattern di attacco



IL DATABASE Metasploit

```
A database appears to be already configured, skipping initialization
[*] The initial module cache will be built in the background, this can take 2-5 minutes...

# cowsay++
< metasploit >
-----
 \   ^__^
  )  ooo\
  (  ^--)
   ||----|
   ||     |

Payload caught by AV? Fly under the radar with Dynamic Payloads in
Metasploit Pro -- learn more on http://rapid7.com/metasploit

      =[ metasploit v4.11.4-2015071403           ]
+ -- --=[ 1467 exploits - 840 auxiliary - 232 post      ]
+ -- --=[ 432 payloads - 37 encoders - 8 nops        ]
+ -- --=[ Free Metasploit Pro trial: http://r-7.co/trymsp  ]

msf > db_nmap 143.225.28.128/25
[*] Nmap: Starting Nmap 6.49BETA4 ( https://nmap.org ) at 2015-09-30 03:59 EDT
```



Metasploit: ANALISI DEI DATI

Terminata la scansione...

```
[*] Nmap: 9103/tcp open jetdirect          143.225.1
[*] Nmap: MAC Address: 00:17:C8:07:88:4E (Kyocera Document Solutions) 143.225.1
[*] Nmap: Nmap scan report for 143.225.28.254 143.225.1
[*] Nmap: Host is up (0.00072s latency). 143.225.1
[*] Nmap: Not shown: 998 closed ports 143.225.1
[*] Nmap: PORT STATE SERVICE 143.225.1
[*] Nmap: 22/tcp open ssh 143.225.1
[*] Nmap: 23/tcp open telnet 143.225.1
[*] Nmap: MAC Address: 00:17:DF:B3:C4:00 (Cisco Systems) 143.225.1
[*] Nmap: Nmap scan report for 143.225.28.168 143.225.1
[*] Nmap: Host is up (0.0000010s latency). 143.225.1
[*] Nmap: All 1900 scanned ports on 143.225.28.168 are closed. 143.225.1
[*] Nmap: Nmap done: 128 IP addresses (38 hosts up) scanned in 7023.90 seconds 143.225.1
msf > 
```

msf > services					
Services					
host	port	proto	name	state	info
143.225.28.134	21	tcp	ftp	open	
143.225.28.134	80	tcp	http	open	
143.225.28.134	631	tcp	ipp	open	
143.225.28.134	7627	tcp	soap-http	open	
143.225.28.134	14000	tcp	scotty-ft	open	
143.225.28.134	280	tcp	http-mgmt	open	
143.225.28.134	9100	tcp	jetdirect	open	
143.225.28.134	443	tcp	https	open	
143.225.28.134	515	tcp	printer	open	
143.225.28.134	23	tcp	telnet	open	
143.225.28.135	135	tcp	msrpc	open	
143.225.28.135	139	tcp	netbios-ssn	open	
143.225.28.135	21	tcp	ftp	open	
143.225.28.135	3389	tcp	ms-wbt-server	open	
143.225.28.135	1947	tcp	sentinelrm	open	
143.225.28.135	445	tcp	microsoft-ds	open	
143.225.28.135	1723	tcp	pptp	open	
143.225.28.136	3389	tcp	ms-wbt-server	open	

...i dati sono nel DB, pronti per essere analizzati!



DOMANDE?

