



# Penetration Testing & Ethical Hacking

## Information Gathering

### Parte 2

Arcangelo Castiglione  
arcastiglione@unisa.it

# Outline

---

- Concetti Preliminari
- Raccolta di Informazioni da Risorse Web-Based
- Raccolta delle Informazioni di Registrazione
- **Raccolta delle Informazioni di Routing**
- Raccolta di Informazioni dai Record DNS
- Raccolta di Informazioni mediante Crawler
- Raccolta di informazioni dal Dark Web
- Altri Strumenti e Servizi per Raccogliere Informazioni

# Raccolta Informazioni di Routing

---

- Raccogliere («tracciare») le informazioni di routing (*tracerouting*) permette di
  - Identificare gli host presenti tra l'host del pentester e l'host target
  - Raccogliere informazioni sul funzionamento della rete e su come il traffico viene instradato tra l'host del pentester e l'host target
  - Determinare se esistono eventuali «barriere» intermedie tra l'host del pentester e l'host target
    - Firewall
    - Server proxy
    - Etc

# Raccolta Informazioni di Routing

---

## ➤ Osservazione

- Il risultato dell'esecuzione di tali comandi può dipendere da molteplici fattori
  - Politiche di filtro in atto sulla rete dell'host sorgente
  - Politiche di filtro in atto sulla rete dell'host di destinazione (target)
  - Politiche di filtro in atto sulle reti degli host intermedi
  - Politiche di routing
  - Configurazione di rete dell'host sorgente e dell'host target

# Raccolta Informazioni di Routing

---

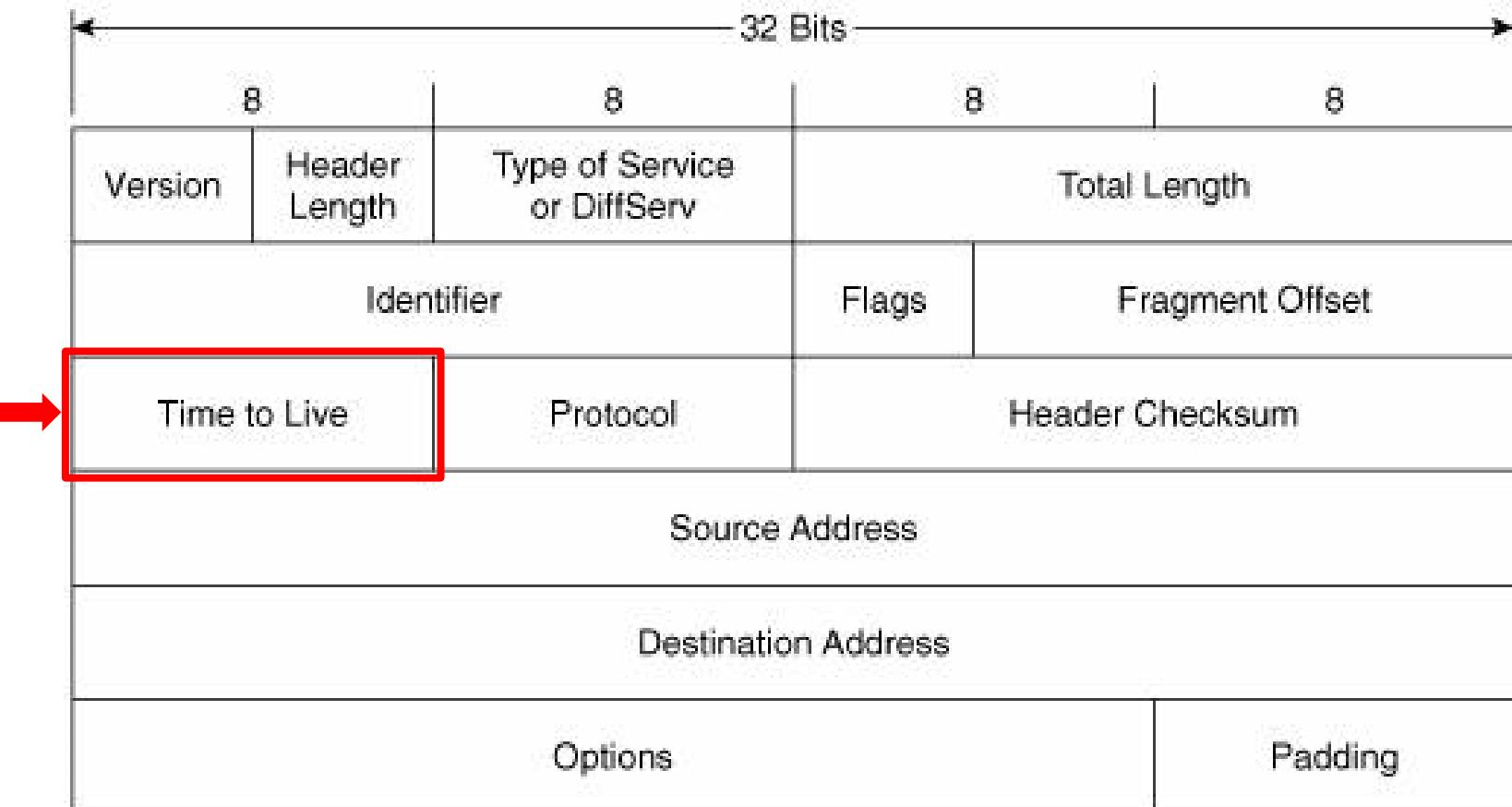
## ➤ Osservazione

- Il risultato dell'esecuzione di tali comandi può dipendere da molteplici fattori
  - Politiche di filtro in atto sulla rete dell'host sorgente
  - Politiche di filtro in atto sulla rete dell'host di destinazione (target)
  - Politiche di filtro in atto sulle reti degli host intermedi
  - Politiche di routing
  - Configurazione di rete dell'host sorgente e dell'host target

**N.B. Quando tali comandi sono eseguiti su sistemi in macchina virtuale,  
potrebbe essere necessario modificare alcune proprietà di rete su tali sistemi**

# Traceroute

Concetti alla base: Header IPv4 – Time To Live (TTL)



# Traceroute

Concetti alla base: Header IPv4 – Time To Live (TTL)

---

- Limite superiore al «**tempo di vita**» di un **pacchetto (datagramma) IP** sulla rete Internet
- Il valore del campo TTL è **impostato dall'host mittente** del pacchetto ed è **decrementato da ogni router intermedio (hop)** lungo la rotta verso la destinazione
- Se il campo **TTL** raggiunge il **valore 0** prima che il pacchetto arrivi alla sua destinazione
  1. Il **pacchetto** viene **scartato**
  2. Viene inviato al mittente un pacchetto contenente un messaggio di **errore «ICMP Time Exceeded»** (ICMP Type: 11)
    - Tale pacchetto conterrà l'**indirizzo IP dell'host che ha generato l'errore**

# Traceroute

## Concetti alla base – Idea di Funzionamento

---

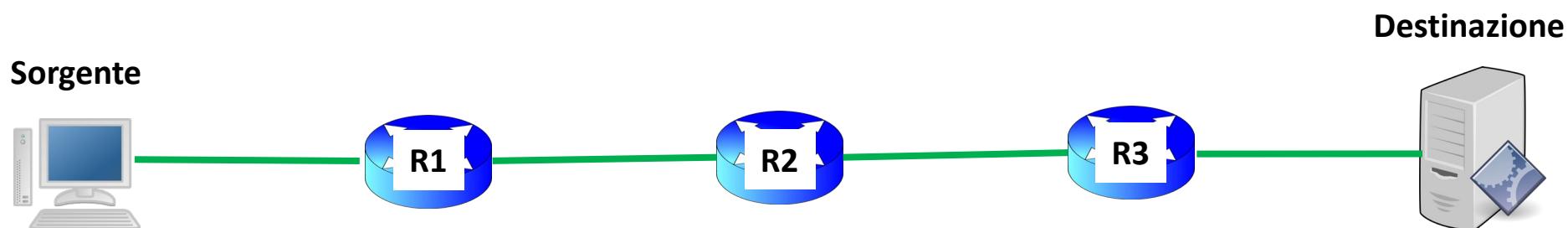
- **traceroute** invia pacchetti *UDP* (in Linux e sistemi Unix-like) oppure «*ICMP Echo Request*» (in Windows) verso l'host di destinazione
  - Il campo *Time To Live (TTL)* del pacchetto è inizialmente impostato ad 1
  - Tale campo è poi di volta in volta incrementato di 1 ad ogni host (*hop*) intermedio raggiunto lungo il percorso di routing verso l'host di destinazione
  - Fino ad un valore massimo prefissato, dipendente dal sistema operativo in uso

# Traceroute

Concetti alla base – Logica di Funzionamento

Sorgente-

Routing path



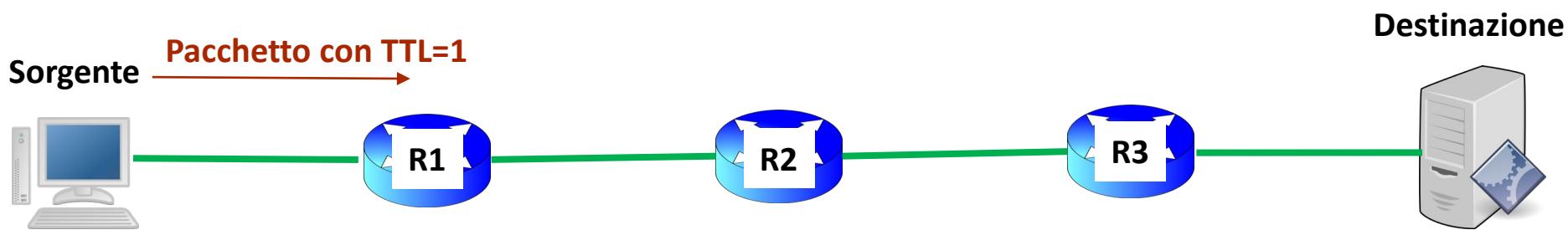
Information Gathering

# Traceroute

Concetti alla base – Logica di Funzionamento

Sorgente-

Routing path

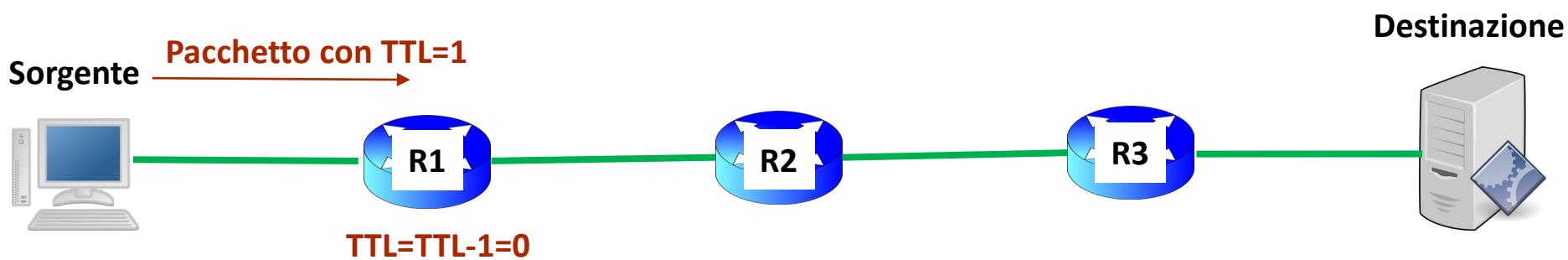


# Traceroute

Concetti alla base – Logica di Funzionamento

Sorgente-

Routing path

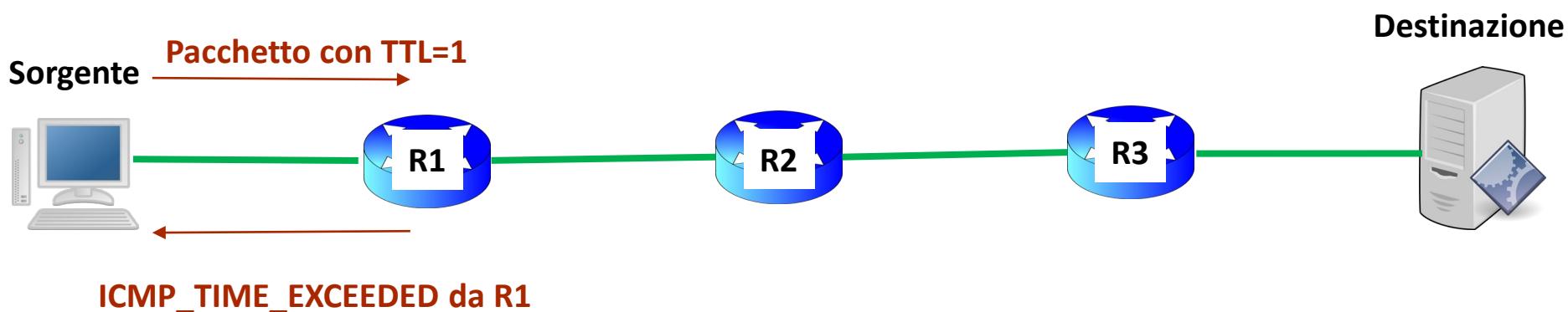


# Traceroute

Concetti alla base – Logica di Funzionamento

Sorgente-R1

Routing path

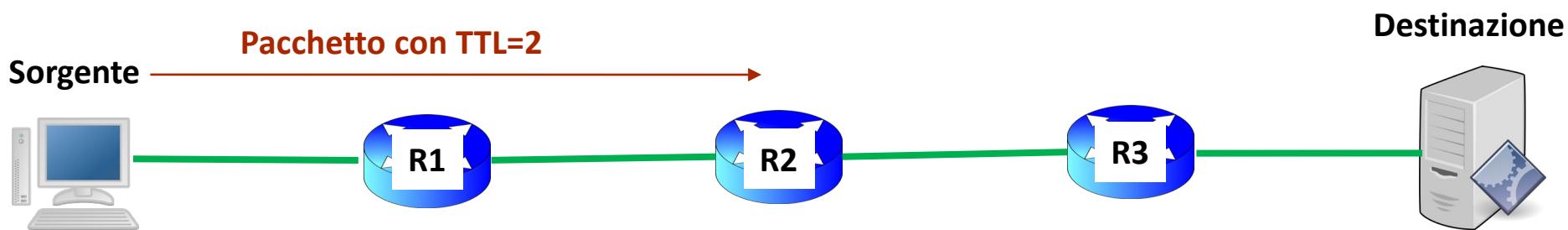


# Traceroute

Concetti alla base – Logica di Funzionamento

Sorgente-R1

Routing path



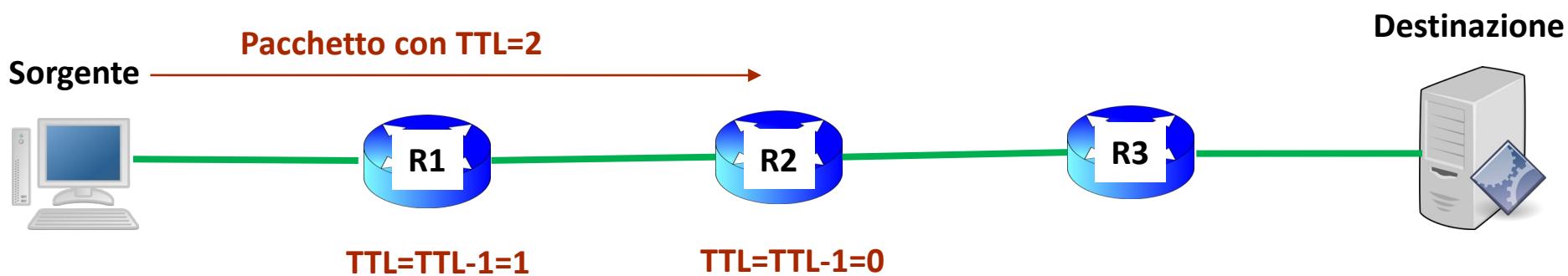
Information Gathering

# Traceroute

Concetti alla base – Logica di Funzionamento

Sorgente-R1

Routing path



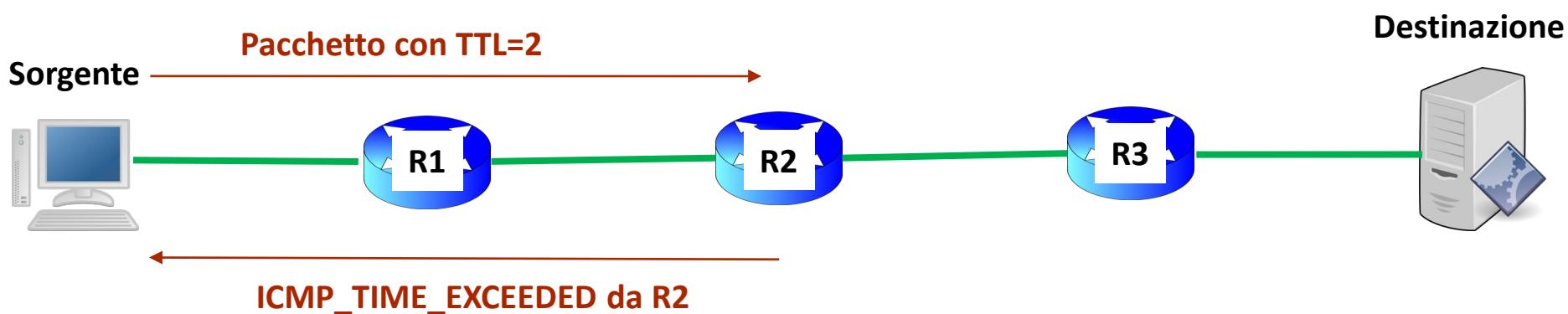
Information Gathering

# Traceroute

Concetti alla base – Logica di Funzionamento

Sorgente-R1-R2

Routing path

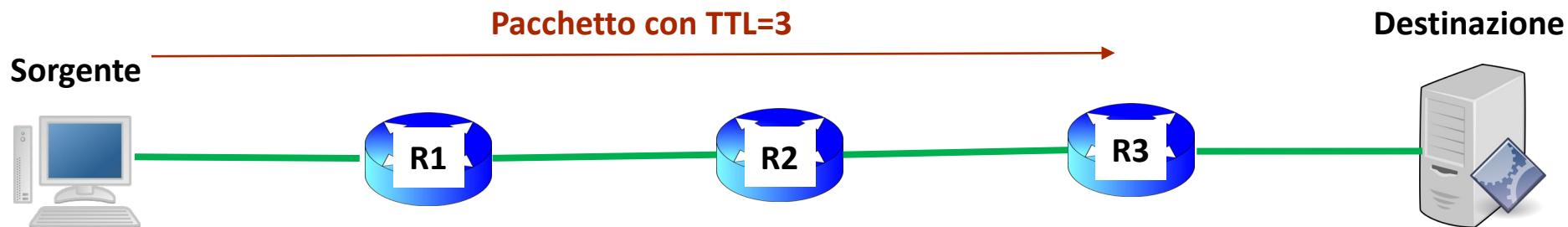


# Traceroute

Concetti alla base – Logica di Funzionamento

Sorgente-R1-R2

Routing path



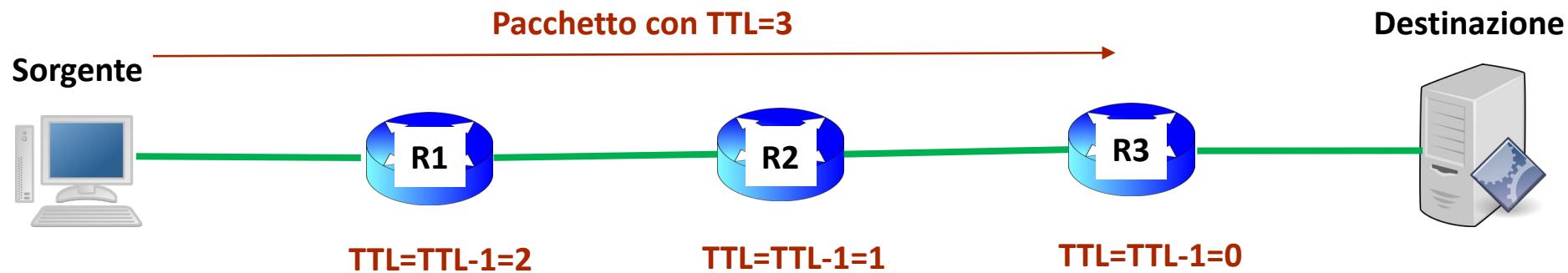
Information Gathering

# Traceroute

Concetti alla base – Logica di Funzionamento

Sorgente-R1-R2

Routing path



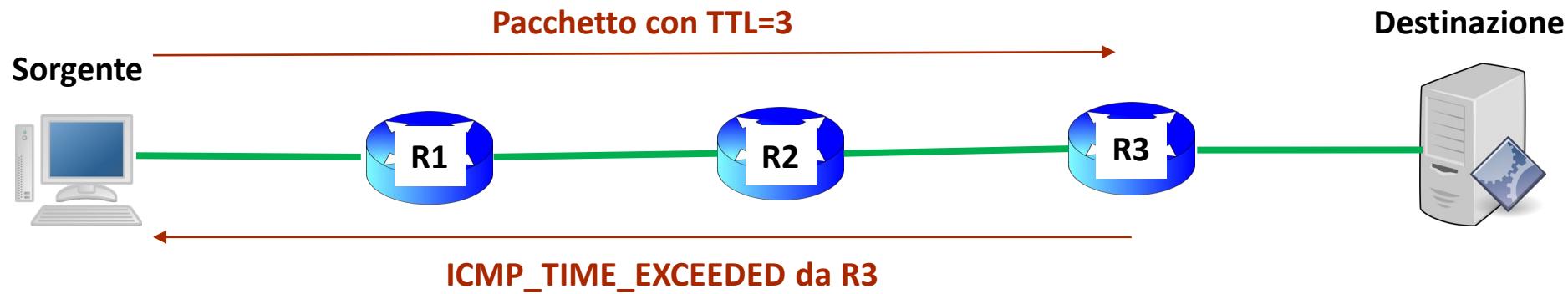
Information Gathering

# Traceroute

Concetti alla base – Logica di Funzionamento

Sorgente-R1-R2-R3

Routing path

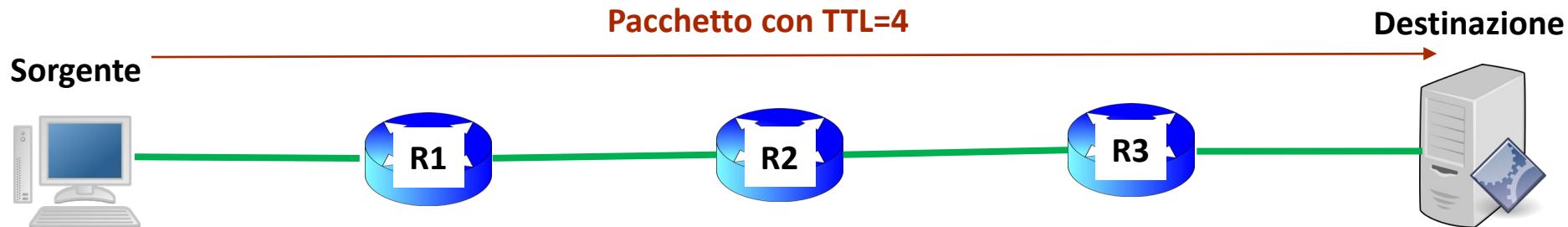


# Traceroute

Concetti alla base – Logica di Funzionamento

Sorgente-R1-R2-R3

Routing path



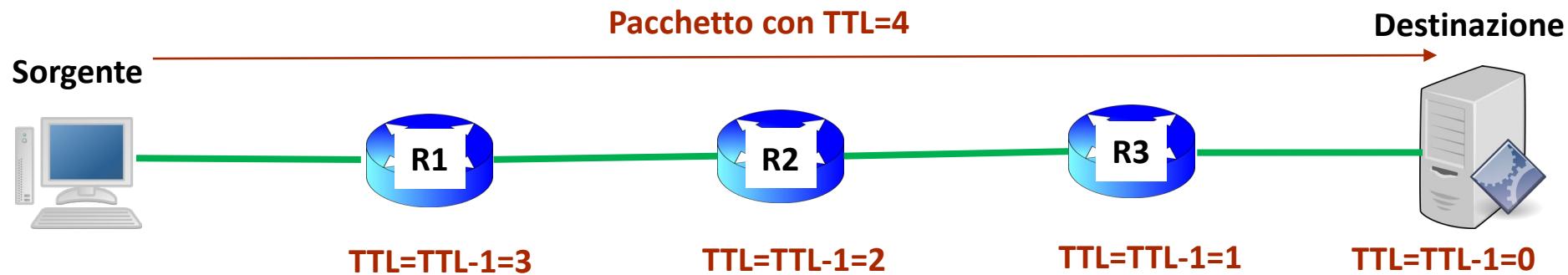
Information Gathering

# Traceroute

Concetti alla base – Logica di Funzionamento

Sorgente-R1-R2-R3

Routing path

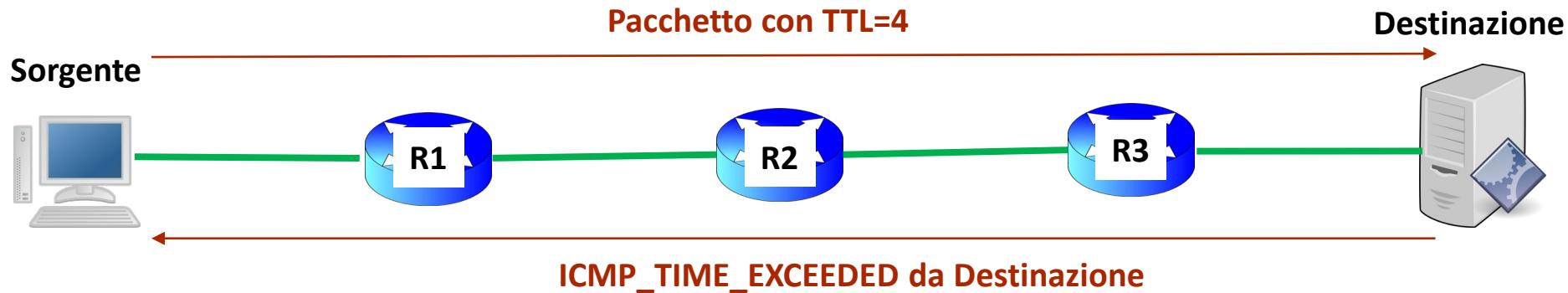


Information Gathering

# Traceroute

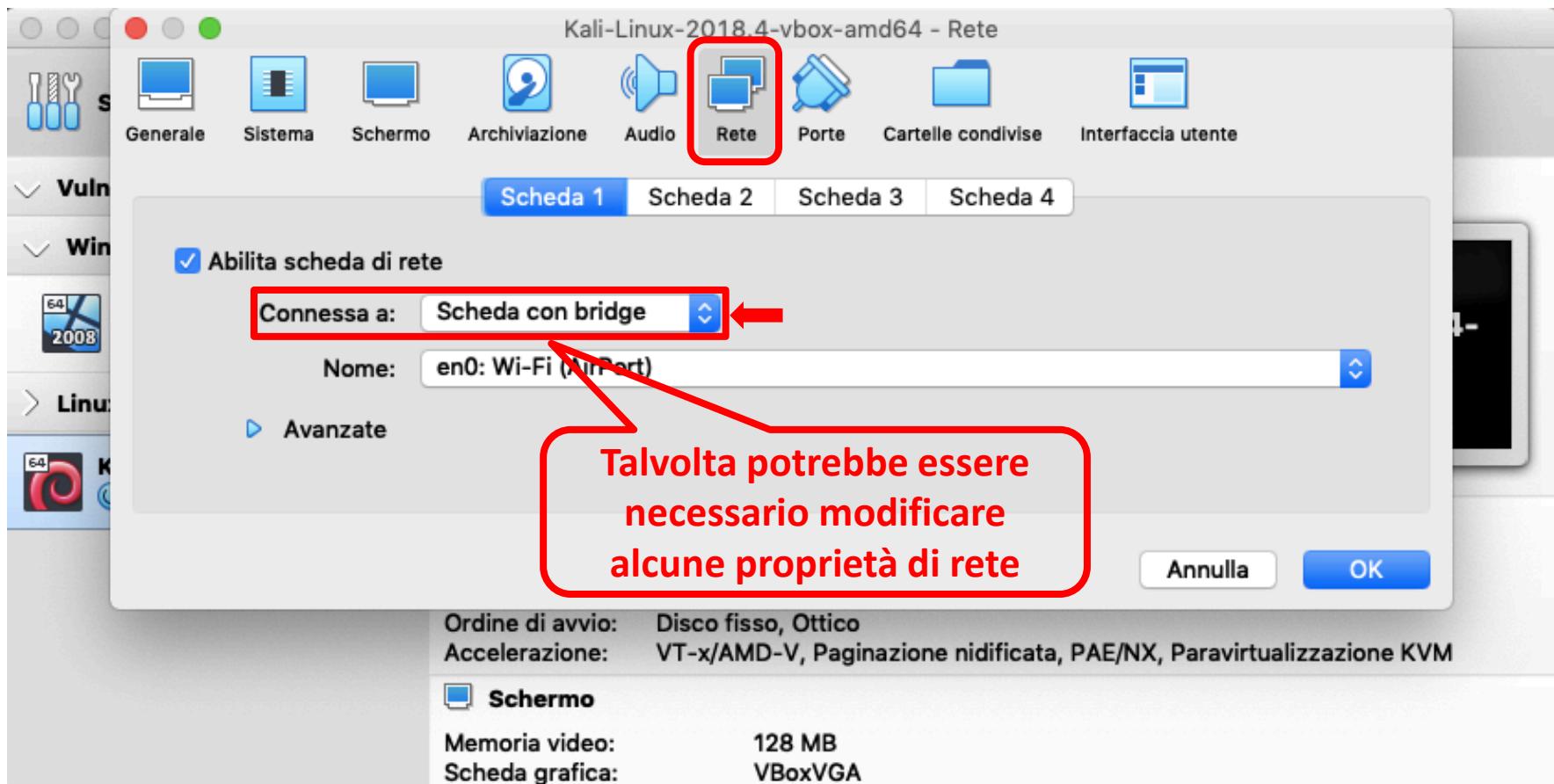
Concetti alla base – Logica di Funzionamento

Sorgente-R1-R2-R3-Destinazione      Routing path



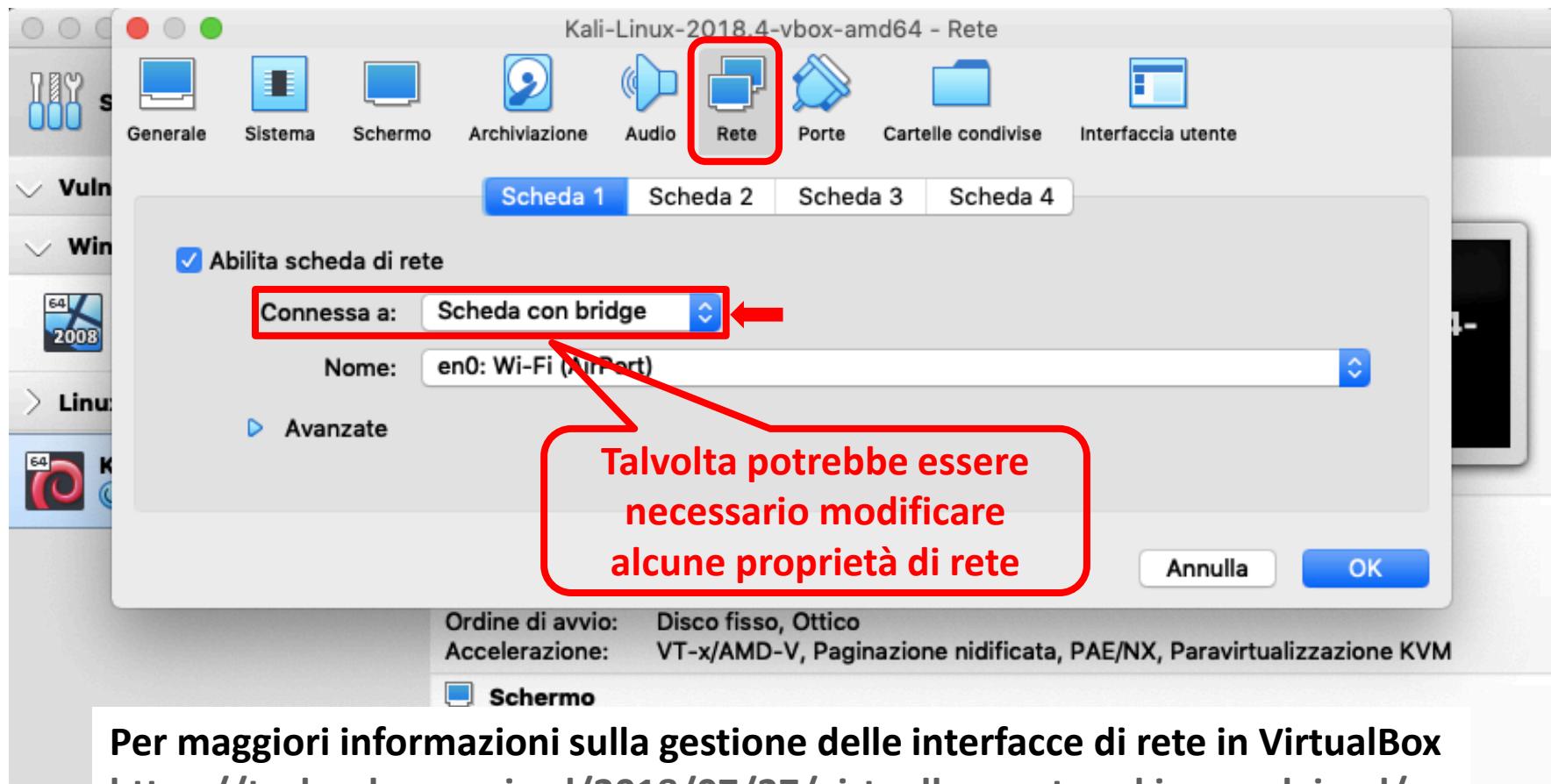
# Raccolta Informazioni di Routing

## Configurazione Preliminare della VM



# Raccolta Informazioni di Routing

## Configurazione Preliminare della VM



# Raccolta Informazioni di Routing

---

- Kali Linux fornisce vari strumenti per ottenere le informazioni di routing (*tracerouting*)
  - **traceroute**
  - **tcptraceroute**
  - **tctrace**
  - Etc

# Raccolta Informazioni di Routing

## Comando traceroute

- **traceroute** di default invia pacchetti *UDP* verso l'host di destinazione
- Per maggiori informazioni sul comando **man traceroute**

```
TRACEROUTE(1)           Traceroute For Linux          TRACEROUTE(1)

NAME
    traceroute - print the route packets trace to network host

SYNOPSIS
    traceroute [-46dFITUnreAV] [-f first_ttl] [-g gate, ... ]
                [-i device] [-m max_ttl] [-p port] [-s src_addr]
                [-q nqueries] [-N squeries] [-t tos]
                [-l flow_label] [-w waittimes] [-z sendwait] [-UL] [-D]
                [-P proto] [--sport=port] [-M method] [-O mod_options]
                [--mtu] [--back]
                host [packet_len]
    traceroute6 [options]
    tcptraceroute [options]
    lft [options]

DESCRIPTION
    traceroute tracks the route packets taken from an IP network on
    their way to a given host. It utilizes the IP protocol's time to
    live (TTL) field and attempts to elicit an ICMP TIME_EXCEEDED re-
    sponse from each gateway along the path to the host.

    traceroute6 is equivalent to traceroute -6
    tcptraceroute is equivalent to traceroute -T
```

# Raccolta Informazioni di Routing

## Comando traceroute

- **traceroute** di default invia pacchetti *UDP* verso l'host di destinazione
- Logica di funzionamento definita dalla *man page* del comando

```
This program attempts to trace the route an IP packet would follow to some internet host by launching probe packets with a small ttl (time to live) then listening for an ICMP "time exceeded" reply from a gateway. We start our probes with a ttl of one and increase by one until we get an ICMP "port unreachable" (or TCP reset), which means we got to the "host", or hit a max (which defaults to 30 hops). Three probes (by default) are sent at each ttl setting and a line is printed showing the ttl, address of the gateway and round trip time of each probe. The address can be followed by additional information when requested. If the probe answers come from different gateways, the address of each responding system will be printed. If there is no response within a certain timeout, an "*" (asterisk) is printed for that probe.
```

# Comando traceroute

Concetti alla base – Logica di Funzionamento in Linux

---

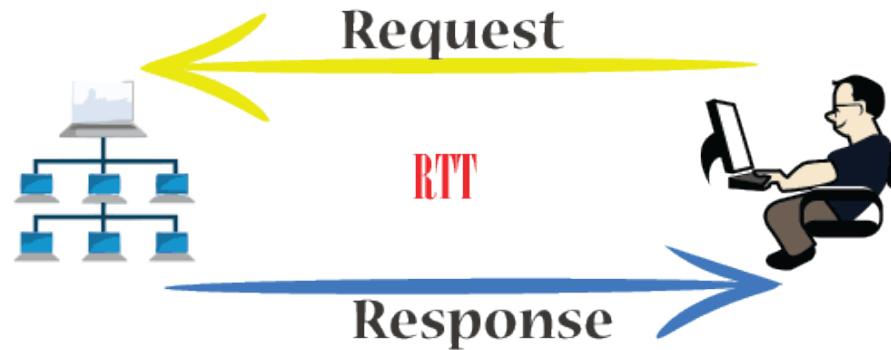
## ➤ Il comando **traceroute**

- Invia tre richieste per ciascun valore del TTL (Time To Live)
- Stampa una riga per ciascun valore del TTL. Tale riga include
  - Valore del TTL
  - Hostname e/o Indirizzo IP del router che ha risposto alla richiesta
  - *Round-Trip Time (RTT)* relativo a ciascuna richiesta
- Per ciascuna richiesta, se non c'è una risposta entro un certo periodo di *timeout*, viene stampato il simbolo asterisco (\*)

# Comando traceroute

Concetti alla base – Logica di Funzionamento in Linux

- **Round-Trip Time (RTT)** o **Round-Trip Delay (RTD)**: tempo richiesto da un pacchetto per viaggiare da una specifica sorgente ad una specifica destinazione e viceversa



# Comando traceroute

## Esempio 1

➤ **traceroute www.unisa.it**

Output parziale – 1/3

```
root@kali:~# traceroute www.unisa.it
traceroute to www.unisa.it (193.205.160.20), 30 hops max, 60 byte packets
 1 _gateway (192.168.1.1)  1.748 ms  6.011 ms  5.997 ms
 2 * * *
 3 172.18.22.78 (172.18.22.78)  51.804 ms 172.18.18.62 (172.18.18.62)  5
 4.268 ms 172.18.22.64 (172.18.22.64)  57.245 ms
 5 172.18.19.226 (172.18.19.226)  64.270 ms
 6 172.17.4.229 (172.17.4.229)  70.873 ms 172.17.4.241 (172.17.4.241)  7
 3.819 ms 172.17.4.229 (172.17.4.229)  77.358 ms
 7 r-rm156-vl4.ipnet.interbusiness.it (151.99.29.208)  78.492 ms r-rm156
 -vl3.ipnet.interbusiness.it (151.99.29.144)  45.162 ms r-rm156-vl4.ipnet.
 interbusiness.it (151.99.29.208)  48.527 ms
 8 172.17.5.206 (172.17.5.206)  50.917 ms  52.817 ms  55.753 ms
 9 garr-nap.namex.it (193.201.28.15)  58.726 ms  61.916 ms  64.323 ms
 10 rx2-rm2-rx2-na1.na1.garr.net (90.147.80.30)  73.057 ms  74.994 ms  76
 .282 ms
```

# Comando traceroute

## Esempio 1

➤ **traceroute www.unisa.it**

Output parziale – 1/3

```
root@kali:~# traceroute www.unisa.it
traceroute to www.unisa.it (193.205.160.20), 30 hops max, 60 byte packets
1 _gateway (192.168.1.1)  1.748 ms  6.011 ms  5.997 ms
2 * * *
3 172.18.22.78 (172.18.22.78)  51.804 ms 172.18.18.62 (172.18.18.62)  5
4.268 ms 172.18.22.64 (172.18.22.64)  57.245 ms
4 * * 172.18.19.226 (172.18.19.226)  64.270 ms
5 172.17.4.229 (172.17.4.229)  70.873 ms 172.17.4.241 (172.17.4.241)  7
3.819 ms 172.17.4.229 (172.17.4.229)  77.358 ms
6 r-rm156-vl4.ipnet.interbusiness.it (151.99.29.208)  78.492 ms r-rm156
-vl3.ipnet.interbusiness.it (151.99.29.144)  45.162 ms r-rm156-vl4.ipnet.
interbusiness.it (151.99.29.208)  48.527 ms
7 172.17.5.206 (172.17.5.206)  50.917 ms  52.817 ms  55.753 ms
8 garr-nap.namex.it (193.201.28.15)  58.726 ms  61.916 ms  64.323 ms
9 rx2-rm2-rx2-na1.na1.garr.net (90.147.80.30)  73.057 ms  74.994 ms  76
.82 ms
```

Valori del TTL

# Comando traceroute

## Esempio 1

➤ **traceroute www.unisa.it**

Output parziale – 1/3

```
root@kali:~# traceroute www.unisa.it
traceroute to www.unisa.it (193.205.160.20), 30 hops max, 60 byte packets
1 _gateway (192.168.1.1) 1.748 ms 6.011 ms 5.997 ms
2 * * *
3 172.18.22.78 (172.18.22.78) 51.804 ms 172.18.18.62 (172.18.18.62) 5
4.268 ms 172.18.22.64 (172.18.22.64) 57.245 ms
4 * * 172.18.19.226 (172.18.19.226) 64.270 ms
5 172.17.4.229 (172.17.4.229) 70.873 ms 172.1
3.819 ms 172.17.4.229 (172.17.4.229) 77.358 ms
6 r-rm156-vl4.ipnet.interbusiness.it (151.99.29.208) 78.492 ms r-rm156
-vl3.ipnet.interbusiness.it (151.99.29.144) 45.162 ms r-rm156-vl4.ipnet.
interbusiness.it (151.99.29.208) 48.527 ms
7 172.17.5.206 (172.17.5.206) 50.917 ms 52.817 ms 55.753 ms
8 garr-nap.namex.it (193.201.28.15) 58.726 ms 61.916 ms 64.323 ms
9 rx2-rm2-rx2-na1.na1.garr.net (90.147.80.30) 73.057 ms 74.994 ms 76
.82 ms
```

Valori del TTL

RTT delle 3 richieste con TTL=1

# Comando traceroute

## Esempio 1

➤ **traceroute www.unisa.it**

Output parziale – 1/3

```
root@kali:~# traceroute www.unisa.it
traceroute to www.unisa.it (193.205.160.20), 30 hops max, 60 byte packets
 1 gateway (192.168.1.1)  1.748 ms  6.011 ms  5.997 ms
 2 * * *
 3 172.18.22.7 (172.18.22.7)  4.268 ms  172.18.22.7  18.62 (172.18.18.62)  5
 4 * * 172.18.19.2
 5 172.17.4.229 (172.17.4.229)  70.875 ms 172.17.4.241 (172.17.4.241)  7
 3.819 ms 172.17.4.229 (172.17.4.229)  77.358 ms
 6 r-rm156-vl4.ipnet.interbusiness.it (151.99.29.208)  78.492 ms r-rm156
-vl3.ipnet.interbusiness.it (151.99.29.144)  45.162 ms r-rm156-vl4.ipnet.
interbusiness.it (151.99.29.208)  48.527 ms
 7 172.17.5.206 (172.17.5.206)  50.917 ms  52.817 ms  55.753 ms
 8 garr-nap.namex.it (193.201.28.15)  58.726 ms  61.916 ms  64.323 ms
 9 rx2-rm2-rx2-na1.na1.garr.net (90.147.80.30)  73.057 ms  74.994 ms  76
.82 ms
```

Le 3 richieste con TTL=2 sono  
andate in timeout

Valori del TTL

# Comando traceroute

## Esempio 1

➤ **traceroute www.unisa.it**

Output parziale – 1/3

```
root@kali:~# traceroute www.unisa.it
traceroute to www.unisa.it (193.205.160.20), 30 hops max, 60 byte packets
 1 _gateway (192.168.1.1)  1.748 ms  6.011 ms  5.997 ms
 2 * * *
 3 172.18.22.78 (172.18.22.78)  51.804 ms 172.18.18.62 (172.18.18.62)  5
 4.268 ms 172.18.22.64 (172.18.22.64)  57.245 ms
 4 * * 172.18.19.226 (172.18.19.226)  64.270 ms
 5 172.17.4.229 (172.17.4.229)  70.272 ms 172.17.4.241 (172.17.4.241)  7
 3.819 ms 172.1.
 6 r-rm156-vl4.1
-vl3.ipnet.interb
interbusiness.it
 7 172.17.5.206
 8 garr-nap.namex.it (193.201.28.15)  58.726 ms  61.916 ms  64.323 ms
 9 rx2-rm2-rx2-na1.na1.garr.net (90.147.80.30)  73.057 ms  74.994 ms  76
.82 ms
```

Le 3 richieste con TTL=3 hanno  
avuto risposta da 3 router  
distinti e ciascuna risposta  
presenta il proprio RTT

Valori del TTL

# Comando traceroute

## Esempio 1

➤ **traceroute www.unisa.it**

Output parziale – 1/3

```
root@kali:~# traceroute www.unisa.it
traceroute to www.unisa.it (193.205.160.20), 30 hops max, 60 byte packets
 1 _gateway (192.168.1.1)  1.748 ms  6.011 ms  5.997 ms
 2 * * *
 3 172.18.22.78 (172.18.22.78)  51.804 ms 172.18.18.62 (172.18.18.62)  5
 4.268 ms 172.18.22.64 (172.18.22.64)  57.245 ms
 4 * * 172.18.19.226 (172.18.19.226)  64.270 ms
 5 172.17.4.229 (172.17.4.229)  70.873 ms 172.17.4.241 (172.17.4.241)  7
 3.819 ms 172.17.4.229 172.17.4.241
 6 r-rm156-vl4.ipnet.in
 -vl3.ipnet.interbusiness.it
 interbusiness.it (151.99.29.1)
 7 172.17.5.206 (172.17.5.206)
 8 garr-nap.namex.it (193.205.160.20)
 9 rx2-rm2-rx2-na1.nal.garr.net (90.147.80.30)  73.057 ms  74.994 ms  76
 .82 ms
```

Le prime 2 richieste con TTL=4

sono andate in timeout, mentre

la terza ha ricevuto risposta in  
un tempo di 64270 ms

Valori del TTL

# Comando traceroute

## Esempio 1

➤ **traceroute www.unisa.it**

Output parziale – 2/3

```
10 rx2-nal-rx1-na2.na2.garr.net (90.147.82.126) 48.055 ms 49.662 ms rx  
2-nal-rx1-nal.nal.garr.net (90.147.80.169) 54.764 ms  
11 rx1-na2-rx1-sa.sa.garr.net (90.147.82.146) 53.674 ms 52.077 ms 52.  
191 ms  
12 * * *  
13 * * *  
14 * * *  
15 * * *  
16 * * *  
17 * * *  
18 * * *  
19 * * *  
20 * * *  
21 * * *  
22 * * *  
23 * * *  
24 * * *
```

- A partire dall'hop 12 non ci sono informazioni disponibili
- Ciò indica/suggerisce la presenza di dispositivi di filtering (ad es., firewall) tra l'host del pentester e l'host target

Valori del TTL

# Comando traceroute

## Esempio 1

➤ **traceroute www.unisa.it**

Output parziale – 3/3

```
25 * * *
26 * * *
27 * * *
28 * * *
29 * * *
30 * * *
```

➤ Di default, nella versione corrente di Kali,  
viene impostato TTL=30, quindi possono  
essere tracciati 30 hop

↑  
Valori del TTL

# Comando traceroute

## Esempio 2

➤ **traceroute 8.8.8.8**

```
└─(root㉿kali)-[~]
└─# traceroute 8.8.8.8
traceroute to 8.8.8.8 (8.8.8.8), 30 hops max, 60 byte packets
 1  modemtim (192.168.1.1)  10.333 ms  9.970 ms  9.928 ms
 2  * * *
 3  172.17.161.108 (172.17.161.108)  14.287 ms  14.563 ms  172.17.161.100 (172
.17.161.100)  15.576 ms
 4  172.17.160.104 (172.17.160.104)  17.665 ms  172.17.160.148 (172.17.160.148
)  17.371 ms  172.17.160.128 (172.17.160.128)  17.333 ms
 5  172.19.177.40 (172.19.177.40)  21.148 ms  172.19.177.48 (172.19.177.48)  2
5.589 ms  172.19.177.40 (172.19.177.40)  25.362 ms
 6  172.19.177.8 (172.19.177.8)  32.978 ms  25.959 ms *
 7  ae49.milano11.mil.seabone.net (195.22.205.98)  24.810 ms  ae49.milano50.mi
l.seabone.net (195.22.205.116)  31.781 ms  ae49.milano11.mil.seabone.net (195.
22.205.98)  25.887 ms
 8  142.250.165.98 (142.250.165.98)  27.607 ms  74.125.146.168 (74.125.146.168
)  30.389 ms  72.14.243.190 (72.14.243.190)  32.378 ms
 9  * * *
10  dns.google (8.8.8.8)  26.805 ms  23.737 ms  25.765 ms
```

In questo caso la destinazione è stata raggiunta

# Comando tcptraceroute

---

- Estende le funzionalità fornite dal comando **traceroute**
- **tcptraceroute** potrebbe essere utilizzato anche in presenza di firewall tra l'host del pentester e l'host di destinazione
  - I firewall sono spesso configurati per filtrare il traffico *UDP* e *ICMP* associato al comando **traceroute**
  - Le informazioni restituite da tale comando potrebbero quindi risultare parziali o inattendibili
- **tcptraceroute** usa pacchetti *TCP SYN* e richiede i privilegi di root per poter essere eseguito

# Comando tcptraceroute

---

- **tcptraceroute** permette di effettuare il tracerouting usando connessioni TCP su una specifica porta
  - Di default, la porta utilizzata è la 80
  - Viene usato il TCP *Three-Way Handshake*
    - Se la porta è *aperta* viene ricevuto un pacchetto *SYN/ACK*
    - Se la porta è *chiusa* viene ricevuto un pacchetto *RST*
- **N.B.** I firewall di solito non bloccano tali connessioni
  - Consentendo di tracciare (*enumerare*) il percorso di routing tra due host anche attraverso meccanismi di filtering

# Comando **tcptraceroute**

---

- Per ottenere informazioni sul comando **tcptraceroute** basta digitarlo
  
- Il suo utilizzo più semplice è quello di invocarlo seguito da un nome di dominio
  
- **tcptraceroute** è equivalente al comando **traceroute -T**

```
root@kali:~# tcptraceroute
Usage: /usr/sbin/tcptraceroute [-hvFnSAE] [-i dev] [-f furst_ttl] [-l length]
                               [-q nqueries] [-t tos] [-m max_ttl] [-p src_port] [-s src_addr]
                               [-w wait_time] host [dest_port] [length]
root@kali:~#
```

# Comando tcptraceroute

## Esempio 1

➤ **tcptraceroute www.unisa.it**

```
└─(root㉿kali)-[~]
# tcptraceroute www.unisa.it
Running:
traceroute -T -O info www.unisa.it
traceroute to www.unisa.it (193.205.185.20), 30 hops max, 60 byte packets
 1 modemtim (192.168.1.1)  41.116 ms  41.073 ms  41.052 ms
 2 * * *
 3 172.17.161.108 (172.17.161.108)  44.627 ms  45.105 ms  172.17.161.100 (172
.17.161.100)  44.987 ms
 4 172.17.160.104 (172.17.160.104)  46.399 ms  48.283 ms  48.602 ms
 5 172.19.177.40 (172.19.177.40)  54.111 ms  55.190 ms  55.167 ms
 6 garr-nap.namex.it (193.201.28.15)  55.143 ms  16.795 ms  18.754 ms
 7 rx2-rm2-rx2-na1.na1.garr.net (90.147.80.30)  21.361 ms  21.864 ms  22.201
ms
 8 rx2-na1-rx1-na2.na2.garr.net (90.147.82.126)  21.982 ms rx2-na1-rx1-na1.n
a1.garr.net (90.147.80.169)  22.180 ms rx2-na1-rx1-na2.na2.garr.net (90.147.8
2.126)  22.051 ms
 9 rx1-na1-rx1-sa.sa.garr.net (90.147.81.26)  29.086 ms  24.507 ms  23.343 m
s
10 193.204.219.202 (193.204.219.202)  23.304 ms  23.285 ms  19.726 ms
11 193.205.175.253 (193.205.175.253)  31.947 ms  30.490 ms  41.130 ms
12 193.205.177.247 (193.205.177.247)  21.021 ms  21.952 ms  25.214 ms
13 193.205.185.20 (193.205.185.20) <syn,ack>  30.342 ms  23.449 ms  22.538 m
s
```

# Comando tcptraceroute

## Esempio 2

- **tcptraceroute hackthissite.org**

```
root@kali:~# tcptraceroute hackthissite.org
Running:
traceroute -T -O info hackthissite.org
traceroute to hackthissite.org (137.74.187.100), 30 hops max, 60 byte pa
ckets
 1 _gateway (192.168.1.1)  5.929 ms  6.480 ms  10.871 ms
 2 * * *
 3 172.18.22.66 (172.18.22.66)  64.055 ms 172.18.22.74 (172.18.22.74)
 61.361 ms 172.18.19.204 (172.18.19.204)  59.965 ms
 4 * * *
 5 * * *
 6 * * *
 7 etrunk49.milan01.mil.seabone.net (195.22.205.98)  55.743 ms etrunk49
.milano50.mil.seabone.net (195.22.205.116)  63.369 ms etrunk49.milan01.r
il.seabone.net (195.22.205.98)  62.729 ms
 8 ae11.milano58.mil.seabone.net (195.22.208.79)  71.778 ms ae10.milano
58.mil.seabone.net (195.22.208.117)  56.209 ms  54.144 ms
 9 be100-152.mil-5-a9.it.eu (91.121.131.62)  56.327 ms  49.695 ms  55.7
16 ms
```

Output parziale

# Comando tcptraceroute

## Esempio 3

➤ **tcptraceroute 8.8.8.8 53**

```
root@kali:~# tcptraceroute 8.8.8.8 53
Running:
        traceroute -T -O info -p 53 8.8.8.8
traceroute to 8.8.8.8 (8.8.8.8), 30 hops max, 60 byte packets
  1 _gateway (192.168.1.1)  3.312 ms  6.420 ms  6.344 ms
  2 * * *
  3 172.18.22.66 (172.18.22.66)  51.680 ms 172.18.18.62 (172.18.18.62)
  54.818 ms 172.18.22.74 (172.18.22.74)  57.785 ms
  4 * * *
  5 * * *
  6 * * *
  7 etrunk49.milano50.mil.seabone.net (195.22.205.116)  65.413 ms  67.63
  5 ms or 70.561 ms
  8/see 72.14.221.64 (72.14.221.64)  68.773 ms 72.14.195.206 (72.14.195.206)
      55.156 ms 72.14.209.236 (72.14.209.236)  57.429 ms
  9 108.170.245.65 (108.170.245.65)  64.017 ms 108.170.245.81 (108.170.2
  45.81)  61.064 ms 72.14.239.144 (72.14.239.144)  61.728 ms
 10 209.85.248.201 (209.85.248.201)  58.121 ms 209.85.248.163 (209.85.24
  8.163)  57.995 ms 209.85.249.67 (209.85.249.67)  62.058 ms
 11 google-public-dns-a.google.com (8.8.8.8) <syn,ack>  68.597 ms  67.78
  0 ms  61.281 ms
```

# Comando tcptraceroute

## Esempio 3

➤ **tcptraceroute 8.8.8.8 53**

```
root@kali:~# tcptraceroute 8.8.8.8 53
Running:
traceroute -T -O info -p 53 8.8.8.8
traceroute to 8.8.8.8 (8.8.8.8), 30 hops
  1 _gateway (192.168.1.1)  3.312 ms  6.413 ms
  2 * * *
  3 172.18.22.66 (172.18.22.66)  51.54.818 ms 172.18.22.74 (172.18.22.74)
  4 * * *
  5 * * *
  6 * * *
  7 etrunk49.milano50.mil.seabone.net (195.22.205.116)  65.413 ms  67.635 ms
  8 72.14.221.64 (72.14.221.64)  68.773 ms 72.14.195.206 (72.14.195.206)
    55.156 ms 72.14.209.236 (72.14.209.236)  57.429 ms
  9 108.170.245.65 (108.170.245.65)  64.017 ms 108.170.245.81 (108.170.245.81)  61.064 ms 72.14.239.144 (72.14.239.144)  61.728 ms
 10 209.85.248.201 (209.85.248.201)  58.121 ms 209.85.248.163 (209.85.248.163)  57.995 ms 209.85.249.67 (209.85.249.67)  62.058 ms
 11 google-public-dns-a.google.com (8.8.8.8) <syn,ack>  68.597 ms  67.780 ms  61.281 ms
```

Effettuo il traceroute verso un server  
DNS di Google (8.8.8.8),  
interrogandolo sulla porta (53)  
attraverso cui il servizio DNS è erogato

# Comando tctrace

---

- Non presente di default in Kali Linux, va quindi installato
  - `apt-get install irpas`
- Logica di funzionamento molto simile a quella del comando **tcptraceroute**
  - **tctrace** invia un pacchetto SYN ad un host specifico e se la risposta è un SYN/ACK, la porta è considerata aperta
- Sintassi del comando
  - `tctrace -i <interfacciaDiRete> -d <targethost>`

Per maggiori informazioni **man tctrace**

# Comando tctrace

## Esempio

➤ **tctrace -i eth0 -d www.unisa.it**

```
root@kali:~# tctrace -i eth0 -d www.unisa.it
 1(1)  [192.168.1.1]
 2(all) Timeout
 3(1)  [172.18.22.66]
 4(all) Timeout
 5(1)  [172.17.4.229]
 6(1)  [151.99.29.144]
 7(1)  [172.17.5.206]
 8(1)  [193.201.28.15]
 9(1)  [90.147.80.30]
10(1)  [90.147.80.169]
11(1)  [90.147.81.26]
12(1)  [193.204.219.202]
13(1)  [193.205.175.253]
14(1)  [193.205.177.247]
15(1)  [193.205.160.20] (reached; open)
```

# Outline

---

- Concetti Preliminari
- Raccolta di Informazioni da Risorse Web-Based
- Raccolta delle Informazioni di Registrazione
- Raccolta delle Informazioni di Routing
- **Raccolta di Informazioni dai Record DNS**
- Raccolta di Informazioni mediante Crawler
- Raccolta di informazioni dal Dark Web
- Altri Strumenti e Servizi per Raccogliere Informazioni

# Analisi dei Record DNS

## Obiettivi

---

- **Caratterizzare l'asset in termini volumetrici**
  
- **Rispondere ad importanti domande per caratterizzare un asset**
  - Quanti sono gli indirizzi IP «appartenenti» all'asset?
  - Quali sono gli indirizzi IP «appartenenti» all'asset?
  - Quali sono i nomi di dominio/sottodomini appartenenti all'asset?
  - Etc



# Analisi dei Record DNS

## Obiettivi

---

- L'obiettivo degli strumenti appartenenti a questa categoria è quello di raccogliere informazioni sui server DNS ed i relativi record
- Tali strumenti permettono anche di ottenere informazioni su tutti gli indirizzi IP e gli hostname associati ad un determinato dominio



# Analisi dei Record DNS

Intro – Cos'è il DNS?

---

- **DNS - Domain Name System (o Server)**
- Database globalmente distribuito, scalabile ed affidabile
- Il DNS si occupa di
  - **Forward Mapping (o Lookup):** conversione da nomi di dominio a indirizzi IP
  - **Reverse Mapping (o Lookup):** conversione da indirizzi IP a nomi di dominio

# Analisi dei Record DNS

## Intro – Cos'è il DNS?

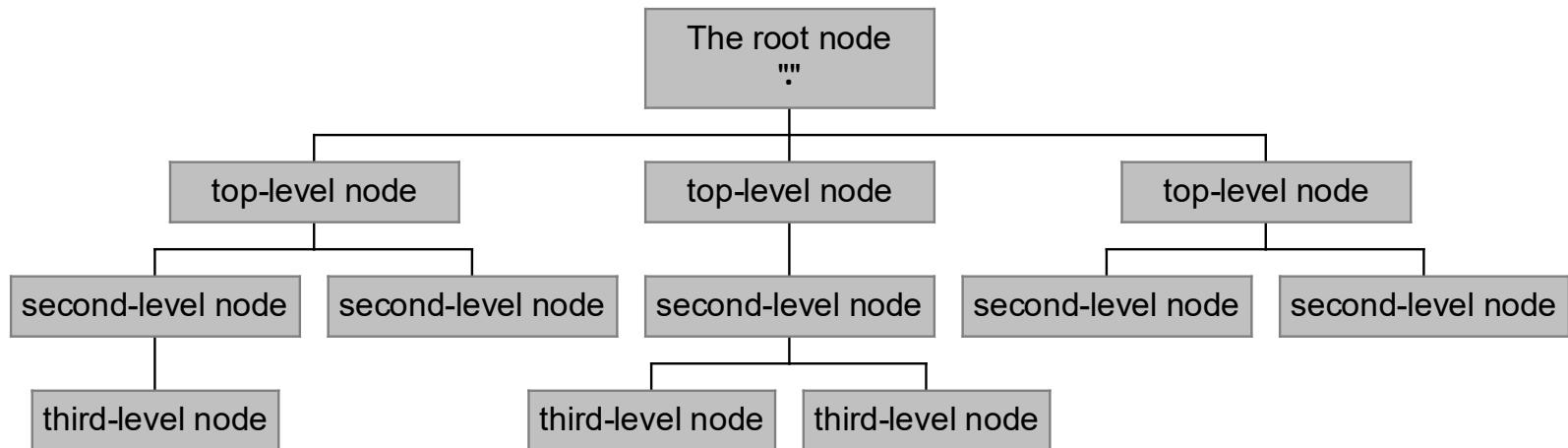
---

- Il DNS si basa su tre componenti principali
  - Spazio dei nomi (**Name Space**)
  - Server (**Name Server**) che rendono disponibile lo spazio dei nomi
  - Client (**Resolver**) che interrogano i server riguardo allo spazio dei nomi

# Analisi dei Record DNS

## Intro – Name Space

- Il Name Space rappresenta la struttura del DNS
  - **Albero invertito**, con il nodo «radice» in cima
- Ciascun nodo ha un'etichetta
  - Il nodo root ha l'etichetta “.”

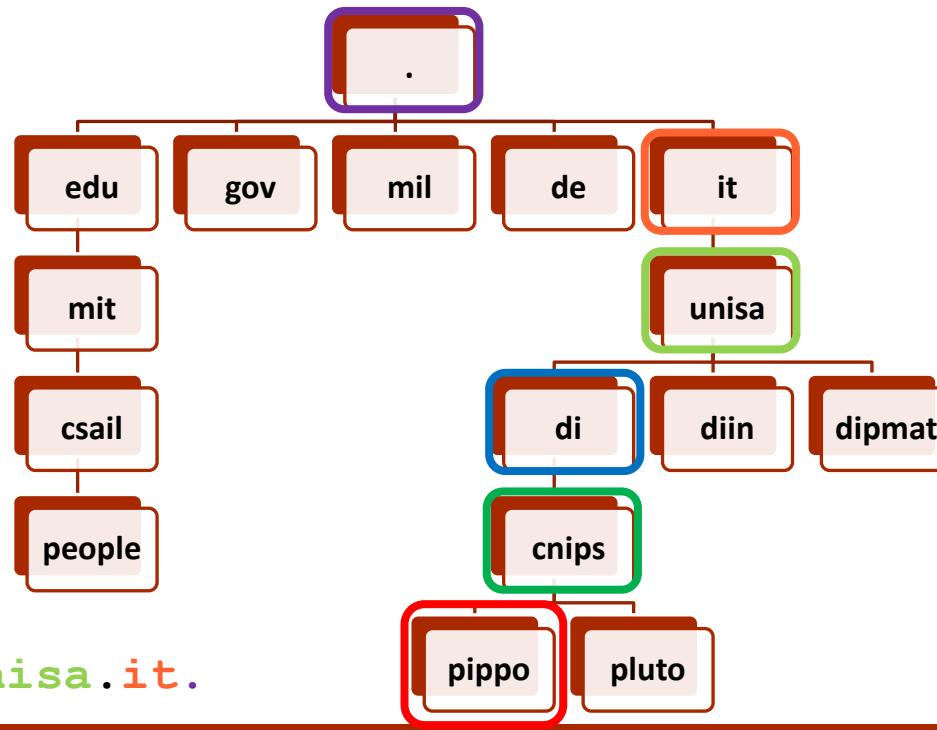


# Analisi dei Record DNS

## Intro – Fully Qualified Domain Name - FQDN

- Un nome di dominio (*Fully Qualified Domain Name - FQDN*) è la sequenza di etichette (lette da sx a dx) da un nodo verso la radice
  - Nome di dominio completo, che specifica la posizione esatta di un host su Internet

Esempio



# Analisi dei Record DNS

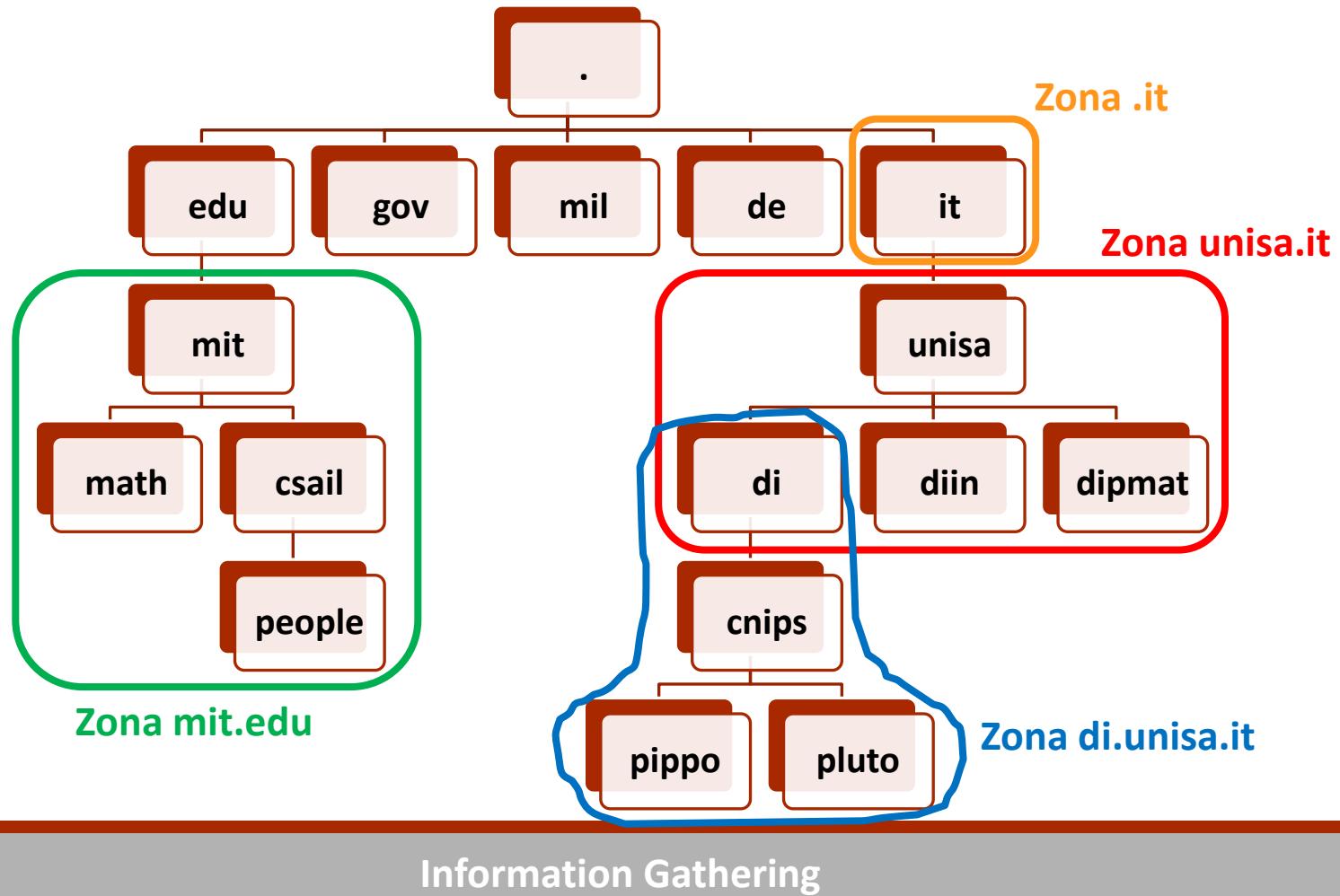
## Intro – Delega

---

- L'amministratore di un dominio può **delegare** la responsabilità della gestione di un suo sottodominio a qualcun altro
  
- Ogni volta che un amministratore effettua tale operazione, viene creata una nuova **unità amministrativa**, chiamata «**zona**»
  - Il sottodominio ed il suo dominio «padre» possono essere amministrati in modo indipendente
  - Il confine tra le zone è chiamato «**punto di delega**»

# Analisi dei Record DNS

Intro – Delega – Esempio



# Analisi dei Record DNS

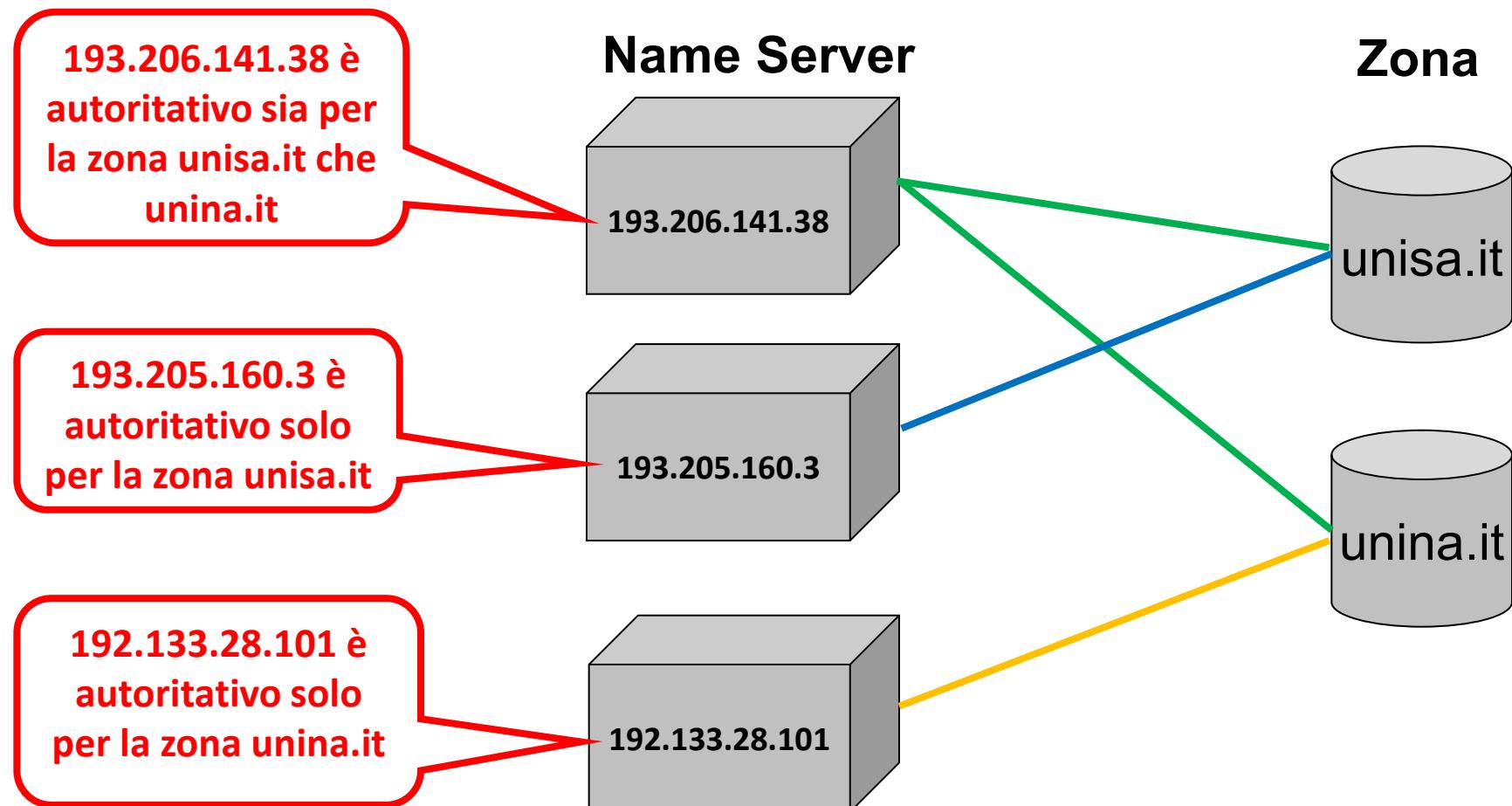
## Intro – Name Server e Zone

---

- I server DNS (*Name Server*) che memorizzano le informazioni relative ad un'intera zona sono detti «**autoritativi**» per tale zona
- Di solito, più di un Name Server è autoritativo per la stessa zona
  - Questo assicura ridondanza e bilanciamento del carico
- Un singolo Name Server può essere autoritativo per più zone

# Analisi dei Record DNS

## Intro – Name Server e Zone – Esempio



# Analisi dei Record DNS

## Intro – Tipi di Name Server

---

- Due tipi di Name Server
  - **Name Server Autoritativi (o Name Server Primari)**: memorizzano informazioni relative ad un'intera zona e possono essere
    - **Master**: dove i dati sono inseriti o modificati
    - **Slave**: dove i dati sono replicati
  - **Name Server di Caching (o Name Server Secondari)**: memorizzano i dati ottenuti da un Name Server Autoritativo
    - Tipicamente ciò è fatto per fini prestazionali

# Analisi dei Record DNS

## Intro – Tipi di Name Server

---

- Due tipi di Name Server
  - **Name Server Autoritativi (o Name Server Primari)**: memorizzano informazioni relative ad un'intera zona e possono essere
    - **Master**: dove i dati sono inseriti o modificati
    - **Slave**: dove i dati sono replicati
  - **Name Server di Caching (o Name Server Secondari)**: memorizzano i dati ottenuti da un Name Server Autoritativo
    - Tipicamente ciò è fatto per fini prestazionali
- Per poter interoperare tra loro, i Name Server utilizzano le informazioni memorizzate nei **Record DNS**

# Analisi dei Record DNS

## Comando host

---

- Mediante il comando **host** è possibile ottenere gli indirizzi IP associati ad un determinato hostname (e viceversa)
  
- Se il parametro passato al comando **host** è un **hostname**, tale comando permette di realizzare un **forward mapping** (o **forward lookup**)
  
- Se il parametro passato al comando **host** è un **indirizzo IP**, tale comando permette di realizzare un **reverse mapping** (o **reverse lookup**)

# Analisi dei Record DNS

## Comando host

- Per maggiori informazioni sul comando **host**

- **man host**

```
HOST(1) pippo.txt BIND9 HOST(1)
NAME
    host - DNS lookup utility

SYNOPSIS
    host [-aCdlnrsTUwv] [-c class] [-N ndots] [-R number] [-t type] [-W wait]
          [-m flag] [[-4] | [-6]] [-v] [-V] {name} [server]

DESCRIPTION
    host is a simple utility for performing DNS lookups. It is normally used to
    convert names to IP addresses and vice versa. When no arguments or options
    are given, host prints a short summary of its command line arguments and
    options.

    name is the domain name that is to be looked up. It can also be a
    dotted-decimal IPv4 address or a colon-delimited IPv6 address, in which
    case host will by default perform a reverse lookup for that address.
    server is an optional argument which is either the name or IP address of
    the name server that host should query instead of the server or servers
    listed in /etc/resolv.conf.
```

# Analisi dei Record DNS

## Comando host – Esempio Forward Mapping

- Mediante il comando **host** è possibile ottenere gli indirizzi IP associati ad un determinato hostname

```
root@kali:~# host hackthissite.org
hackthissite.org has address 137.74.187.104
hackthissite.org has address 137.74.187.100
hackthissite.org has address 137.74.187.103
hackthissite.org has address 137.74.187.101
hackthissite.org has address 137.74.187.102
hackthissite.org has IPv6 address 2001:41d0:8:ccd8:137:74:187:100
hackthissite.org has IPv6 address 2001:41d0:8:ccd8:137:74:187:102
hackthissite.org has IPv6 address 2001:41d0:8:ccd8:137:74:187:103
hackthissite.org has IPv6 address 2001:41d0:8:ccd8:137:74:187:101
hackthissite.org has IPv6 address 2001:41d0:8:ccd8:137:74:187:104
hackthissite.org mail is handled by 10 aspmx.l.google.com.
hackthissite.org mail is handled by 20 alt1.aspmx.l.google.com.
hackthissite.org mail is handled by 20 alt2.aspmx.l.google.com.
hackthissite.org mail is handled by 30 aspmx2.googlemail.com.
hackthissite.org mail is handled by 30 aspmx3.googlemail.com.
hackthissite.org mail is handled by 30 aspmx4.googlemail.com.
hackthissite.org mail is handled by 30 aspmx5.googlemail.com.
root@kali:~#
```

host hackthissite.org

# Analisi dei Record DNS

## Comando host – Esempio Forward Mapping

- Mediante il comando **host** è possibile ottenere gli indirizzi IP associati ad un determinato hostname

```
root@kali:~# host hackthissite.org
hackthissite.org has address 137.74.187.104
hackthissite.org has address 137.74.187.100
hackthissite.org has address 137.74.187.103
hackthissite.org has address 137.74.187.101
hackthissite.org has address 137.74.187.102
hackthissite.org has IPv6 address 2001:41d0:8:ccd8:137:74:187:100
hackthissite.org has IPv6 address 2001:41d0:8:ccd8:137:74:187:102
hackthissite.org has IPv6 address 2001:41d0:8:ccd8:137:74:187:103
hackthissite.org has IPv6 address 2001:41d0:8:ccd8:137:74:187:101
hackthissite.org has IPv6 address 2001:41d0:8:ccd8:137:74:187:104
hackthissite.org mail is handled by 10 aspmx.l.google.com.
hackthissite.org mail is handled by 20 alt1.aspmx.l.google.com.
hackthissite.org mail is handled by 20 alt2.aspmx.l.google.com.
hackthissite.org mail is handled by 30 aspmx2.googlemail.com.
hackthissite.org mail is handled by 30 aspmx3.googlemail.com.
hackthissite.org mail is handled by 30 aspmx4.googlemail.com.
hackthissite.org mail is handled by 30 aspmx5.googlemail.com.
root@kali:~#
```

hackthissite.org  
Host di testing che permette  
di esercitarsi con strumenti  
per l'ethical hacking

# Analisi dei Record DNS

## Comando host – Esempio Forward Mapping

- Mediante il comando **host** è possibile ottenere i record associati ad un determinato hostname

Di default, il comando **host** restituisce i campi (*record*) A, AAAA ed MX di un determinato dominio

```
root@kali:~# host hackthissite.org
hackthissite.org has address 137.74.187.104
hackthissite.org has address 137.74.187.100
hackthissite.org has address 137.74.187.103
hackthissite.org has address 137.74.187.101
hackthissite.org has address 137.74.187.102
hackthissite.org has IPv6 address 2001:41d0:8:ccd8:137:74:187:100
hackthissite.org has IPv6 address 2001:41d0:8:ccd8:137:74:187:102
hackthissite.org has IPv6 address 2001:41d0:8:ccd8:137:74:187:103
hackthissite.org has IPv6 address 2001:41d0:8:ccd8:137:74:187:101
hackthissite.org has IPv6 address 2001:41d0:8:ccd8:137:74:187:104
hackthissite.org mail is handled by 10 aspmx.l.google.com.
hackthissite.org mail is handled by 20 alt1.aspmx.l.google.com.
hackthissite.org mail is handled by 20 alt2.aspmx.l.google.com.
hackthissite.org mail is handled by 30 aspmx2.googlemail.com.
hackthissite.org mail is handled by 30 aspmx3.googlemail.com.
hackthissite.org mail is handled by 30 aspmx4.googlemail.com.
hackthissite.org mail is handled by 30 aspmx5.googlemail.com.
root@kali:~#
```

# Analisi dei Record DNS

## Comando host – Esempio Forward Mapping

- Mediante il comando **host** è possibile ottenere gli indirizzi IP associati ad un determinato hostname

```
root@kali:~# host hackthissite.org
hackthissite.org has address 137.74.187.104
hackthissite.org has address 137.74.187.100
hackthissite.org has address 137.74.187.103
hackthissite.org has address 137.74.187.101
hackthissite.org has address 137.74.187.102
hackthissite.org has IPv6 address 2001:41d0:8:ccd8:137:74:187:100
hackthissite.org has IPv6 address 2001:41d0:8:ccd8:137:74:187:102
hackthissite.org has IPv6 address 2001:41d0:8:ccd8:137:74:187:103
hackthissite.org has IPv6 address 2001:41d0:8:ccd8:137:74:187:101
hackthissite.org has IPv6 address 2001:41d0:8:ccd8:137:74:187:104
hackthissite.org mail is handled by 10 aspmx.l.google.com.
hackthissite.org mail is handled by 20 alt1.aspmx.l.google.com.
hackthissite.org mail is handled by 20 alt2.aspmx.l.google.com.
hackthissite.org mail is handled by 30 aspmx2.googlemail.com.
hackthissite.org mail is handled by 30 aspmx3.googlemail.com.
hackthissite.org mail is handled by 30 aspmx4.googlemail.com.
hackthissite.org mail is handled by 30 aspmx5.googlemail.com.
root@kali:~#
```

Indirizzi IPv4 associati  
ad **hackthissite.org**  
(Record A)

# Analisi dei Record DNS

## Comando host – Esempio Forward Mapping

- Mediante il comando **host** è possibile ottenere gli indirizzi IP associati ad un determinato hostname

```
root@kali:~# host hackthissite.org
hackthissite.org has address 137.74.187.104
hackthissite.org has address 137.74.187.100
hackthissite.org has address 137.74.187.103
hackthissite.org has address 137.74.187.101
hackthissite.org has address 137.74.187.102
hackthissite.org has IPv6 address 2001:41d0:8:ccd8:137:74:187:100
hackthissite.org has IPv6 address 2001:41d0:8:ccd8:137:74:187:102
hackthissite.org has IPv6 address 2001:41d0:8:ccd8:137:74:187:103
hackthissite.org has IPv6 address 2001:41d0:8:ccd8:137:74:187:101
hackthissite.org has IPv6 address 2001:41d0:8:ccd8:137:74:187:104
hackthissite.org mail is handled by 10 aspmx.l.google.com.
hackthissite.org mail is handled by 20 alt1.aspmx.l.google.com.
hackthissite.org mail is handled by 20 alt2.aspmx.l.google.com.
hackthissite.org mail is handled by 30 aspmx2.googlemail.com.
hackthissite.org mail is handled by 30 aspmx3.googlemail.com.
hackthissite.org mail is handled by 30 aspmx4.googlemail.com.
hackthissite.org mail is handled by 30 aspmx5.googlemail.com.
root@kali:~#
```

**Indirizzi IPv6 associati ad  
hackthissite.org (Record AAAA)**

# Analisi dei Record DNS

## Comando host – Esempio Forward Mapping

- Mediante il comando **host** è possibile ottenere gli indirizzi IP associati ad un determinato hostname

```
root@kali:~# host hackthissite.org
hackthissite.org has address 137.74.187.104
hackthissite.org has address 137.74.187.100
hackthissite.org has address 137.74.187.103
hackthissite.org has address 137.74.187.101
hackthissite.org has address 137.74.1
hackthisite.org has IPv6 address 200
hackthisite.org mail is handled by 10 aspmx.l.google.com.
hackthisite.org mail is handled by 20 alt1.aspmx.l.google.com.
hackthisite.org mail is handled by 20 alt2.aspmx.l.google.com.
hackthisite.org mail is handled by 30 aspmx2.googlemail.com.
hackthisite.org mail is handled by 30 aspmx3.googlemail.com.
hackthisite.org mail is handled by 30 aspmx4.googlemail.com.
hackthisite.org mail is handled by 30 aspmx5.googlemail.com.
root@kali:~#
```

Server responsabili della gestione delle e-mail relative al dominio **hackthisite.org** (Record MX)

# Analisi dei Record DNS

## Comando host – Esempio Forward Mapping

- Di default il comando **host** restituisce i record A, AAAA ed MX di un dominio
- Tale comando può essere invocato per far restituire **tutti i record** di un determinato dominio

➤ **host -a hackthissite.org**

```
root@kali:~# host -a hackthissite.org
Trying "hackthissite.org"
Trying "hackthissite.org"
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 9167
;; flags: qr rd ra; QUERY: 1, ANSWER: 28, AUTHORITY: 0, ADDITIONAL: 0

;; QUESTION SECTION:
;hackthissite.org.      IN      ANY

;; ANSWER SECTION:
hackthissite.org.    3599   IN      SOA     c.ns.buddyns.com. admin.hackthissite.
org. 2019030401 3600 900 604800 86400
hackthissite.org.    3599   IN      CAA     0 iodef "mailto:admin@hackthissite.or
g"
hackthissite.org.    3599   IN      CAA     0 issue "letsencrypt.org"
hackthissite.org.    3599   IN      CAA     0 issue "sectigo.com"
hackthissite.org.    3599   IN      MX      10 aspmx.l.google.com.
hackthissite.org.    3599   IN      MX      20 alt1.aspmx.l.google.com.
hackthissite.org.    3599   IN      MX      20 alt2.aspmx.l.google.com.
hackthissite.org.    3599   IN      MX      30 aspmx2.googlemail.com.
hackthissite.org.    3599   IN      MX      30 aspmx3.googlemail.com.
hackthissite.org.    3599   IN      MX      30 aspmx4.googlemail.com.
hackthissite.org.    3599   IN      MX      30 aspmx5.googlemail.com.
hackthissite.org.    3599   IN      NS      c.ns.buddyns.com.
```

Output parziale

# Analisi dei Record DNS

## Comando host – Esempio Reverse Mapping

- Mediante il comando **host** è anche possibile ottenere l'hostname associato ad un determinato indirizzo IP  
**(Reverse Lookup)**

- **host 137.74.187.102**

```
root@kali:~# host 137.74.187.102
102.187.74.137.in-addr.arpa domain name pointer hackthissite.org.
root@kali:~#
```

# Analisi dei Record DNS

## Comando host

- Mediante il comando **host** è anche possibile controllare se un determinato dominio utilizza un *Content Delivery Network (CDN)*
  - Akamai, Cloudflare, etc

```
host www.nike.com
```

```
root@kali:~# host www.nike.com
www.nike.com is an alias for www-geo.nike.com.akadns.net.
www-geo.nike.com.akadns.net is an alias for www.nike.com.akadns.net.
www.nike.com.akadns.net is an alias for ev-cn.nike.com.edgekey.net.
ev-cn.nike.com.edgekey.net is an alias for ev-cn.nike.com.edgekey.net.globalredir.akadns.net.
ev-cn.nike.com.edgekey.net.globalredir.akadns.net is an alias for e2785.x.akamaiedge.net.
e2785.x.akamaiedge.net has address 2.22.19.97
root@kali:~#
```

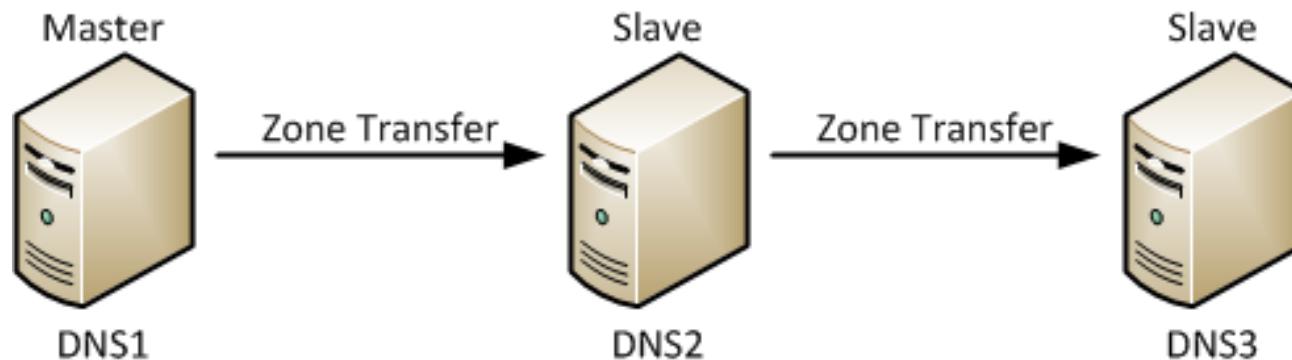
```
host www.microsoft.com
```

```
root@kali:~# host www.microsoft.com
www.microsoft.com is an alias for www.microsoft.com-c-3.edgekey.net.
www.microsoft.com-c-3.edgekey.net is an alias for www.microsoft.com-c-3.edgekey.net.globalredir.akadns.net.
www.microsoft.com-c-3.edgekey.net.globalredir.akadns.net is an alias for e13678.dspb.akamaiedge.net.
e13678.dspb.akamaiedge.net has address 2.23.106.83
e13678.dspb.akamaiedge.net has IPv6 address 2001:41a8:26:1a3::356e
e13678.dspb.akamaiedge.net has IPv6 address 2001:41a8:26:187::356e
root@kali:~#
```

# Analisi dei Record DNS

## Comando host – Zone Transfer (ZT)

- Mediante il comando **host** è anche possibile effettuare un'operazione di **DNS Zone Transfer (ZT)**
  - Meccanismo usato per replicare un database DNS da un *Master Name Server* verso uno *Slave Name Server*
  - Senza questo meccanismo, gli amministratori dei server DNS dovrebbero aggiornare ciascun server DNS separatamente



# Analisi dei Record DNS

## Comando host – Zone Transfer (ZT)

- Se un'operazione di ZT va a buon fine, un utente malevolo potrebbe
  - Conoscere host che non sono pubblicamente disponibili
  - Raccogliere altre informazioni potenzialmente utili sull'asset
- Improbabile trovare server DNS che permettano di effettuare ZT a seguito di richieste da parte di host arbitrari (non autenticati)
  - Le informazioni che tale meccanismo permette di scambiare potrebbero essere sensibili o comunque critiche
- **N.B.** Se un server DNS permette a chiunque di effettuare operazioni di ZT senza alcuna limitazione
  - Tale server è stato mal configurato oppure presenta dei bug



# Analisi dei Record DNS

## Comando host – Zone Transfer (ZT) – Esempio

---

- Per simulare un'operazione di ZT, utilizziamo **zonetransfer.me**, un servizio DNS vulnerabile «by design» allo ZT

1. Individuiamo i *Master Name Server* associati al dominio **zonetransfer.me**

- **host -t ns zonetransfer.me**

```
root@kali:~# host -t ns zonetransfer.me
zonetransfer.me name server nsztm1.digi.ninja.
zonetransfer.me name server nsztm2.digi.ninja.
```

- I *Master Name Server* associati al dominio **zonetransfer.me** sono
  - **nsztm1.digi.ninja**
  - **nsztm2.digi.ninja**

# Analisi dei Record DNS

## Comando host – Zone Transfer (ZT) – Esempio

- Per simulare un'operazione di ZT, utilizziamo **zonetransfer.me**, un servizio DNS vulnerabile «by design» allo ZT

2. Richiediamo di effettuare uno ZT, simulando di essere uno *Slave Name Server*

- **host -l zonetransfer.me nsztm1.digi.ninja.**

```
root@kali:~# host -l zonetransfer.me nsztm1.digi.ninja
Using domain server:
Name: nsztm1.digi.ninja
Address: 34.225.33.2#53
Aliases:

zonetransfer.me has address 217.147.177.157
zonetransfer.me name server nsztm1.digi.ninja.
zonetransfer.me name server nsztm2.digi.ninja.
157.177.147.217.IN-ADDR.ARPA.zonetransfer.me domain name pointer www.zonetransfer.me.
asfdbbox.zonetransfer.me has address 127.0.0.1
canberra-office.zonetransfer.me has address 202.14.81.230
dc-office.zonetransfer.me has address 143.228.181.132
deadbeef.zonetransfer.me has IPv6 address dead:beaf::
```

Output parziale

# Analisi dei Record DNS

## Comando host – Zone Transfer (ZT) – Esempio

- Per simulare un'operazione di ZT, utilizziamo **zonetransfer.me**, un servizio DNS vulnerabile «by design» allo ZT

2. Richiediamo di effettuare uno ZT, simulando di essere uno *Slave Name Server*

- **host -l zonetransfer.me nsztml.digi.ninja.**

```
asfdbbox.zonetransfer.me has address 127.0.0.1
canberra-office.zonetransfer.me has address 202.14.81.230
dc-office.zonetransfer.me has address 143.228.181.132
deadbeef.zonetransfer.me has IPv6 address dead:beaf::
email.zonetransfer.me has address 74.125.206.26
home.zonetransfer.me has address 127.0.0.1
internal.zonetransfer.me name server intns1.zonetransfer.me.
internal.zonetransfer.me name server intns2.zonetransfer.me.
intns1.zonetransfer.me has address 81.4.108.41
intns2.zonetransfer.me has address 167.88.42.94
office.zonetransfer.me has address 4.23.39.254
ipv6actnow.org.zonetransfer.me has IPv6 address 2001:67c:2e8:11::c100:1332
owa.zonetransfer.me has address 207.46.197.32
alltcpportsopen.firewall.test.zonetransfer.me has address 127.0.0.1
vpn.zonetransfer.me has address 174.36.59.154
www.zonetransfer.me has address 5.196.105.14
```

Output parziale

# Analisi dei Record DNS

## Comando `dig`

- Mediante il comando `dig` è possibile effettuare interrogazioni DNS
  - È più flessibile rispetto al comando `host`
- Per maggiori informazioni sul comando `dig`
  - `man dig`

```
DIG(1)          pippo.txt          BIND9          DIG(1)

NAME
    dig - DNS lookup utility

SYNOPSIS
    dig [@server] [-b address] [-c class] [-f filename] [-k filename] [-m]
        [-p port#] [-q name] [-t type] [-v] [-x addr] [-y [hmac:]name:key]
        [[-4] | [-6]] [name] [type] [class] [queryopt...]

    dig [-h]

    dig [global-queryopt...] [query...]

DESCRIPTION
    dig is a flexible tool for interrogating DNS name servers. It performs DNS
    lookups and displays the answers that are returned from the name server(s)
    that were queried. Most DNS administrators use dig to troubleshoot DNS
    problems because of its flexibility, ease of use and clarity of output.
    Other lookup tools tend to have less functionality than dig.

    Although dig is normally used with command-line arguments, it also has a
    batch mode of operation for reading lookup requests from a file. A brief
    summary of its command-line arguments and options is printed when the -h
    option is given. Unlike earlier versions, the BIND 9 implementation of dig
    allows multiple lookups to be issued from the command line.
```

# Analisi dei Record DNS

## Comando dig – Esempio

```
root@kali:~# dig hackthissite.org

; <>> DiG 9.11.5-P1-1-Debian <>> hackthissite.org
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 18851
;; flags: qr rd ra; QUERY: 1, ANSWER: 5, AUTHORITY: 5, ADDITIONAL: 11

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
;; QUESTION SECTION:
;hackthissite.org.      IN      A

;; ANSWER SECTION:
hackthissite.org.    3426    IN      A      137.74.187.104
hackthissite.org.    3426    IN      A      137.74.187.101
hackthissite.org.    3426    IN      A      137.74.187.103
hackthissite.org.    3426    IN      A      137.74.187.100
hackthissite.org.    3426    IN      A      137.74.187.102
```

1/2

**dig hackthissite.org**

```
;; AUTHORITY SECTION:
hackthissite.org.    3426    IN      NS      h.ns.buddyns.com.
hackthissite.org.    3426    IN      NS      j.ns.buddyns.com.
hackthissite.org.    3426    IN      NS      c.ns.buddyns.com.
hackthissite.org.    3426    IN      NS      f.ns.buddyns.com.
hackthissite.org.    3426    IN      NS      g.ns.buddyns.com.

;; ADDITIONAL SECTION:
c.ns.buddyns.com.   56936   IN      A      88.198.106.11
c.ns.buddyns.com.   56936   IN      AAAA   2a01:4f8:d12:d01::10:4
f.ns.buddyns.com.   56945   IN      A      103.6.87.125
f.ns.buddyns.com.   56945   IN      AAAA   2403:2500:4000::f3e
g.ns.buddyns.com.   102509  IN      A      107.181.178.180
g.ns.buddyns.com.   102509  IN      AAAA   2607:f7a0:a:23::3
h.ns.buddyns.com.   56936   IN      A      119.252.20.56
h.ns.buddyns.com.   56936   IN      AAAA   2401:1400:1:1201:0:1:7853:1a5
j.ns.buddyns.com.   56945   IN      A      185.34.136.178
j.ns.buddyns.com.   56945   IN      AAAA   2a00:dcc7:d3ff:88b2::1

;; Query time: 3 msec
;; SERVER: 193.205.160.3#53(193.205.160.3)
;; WHEN: mar feb 19 11:09:01 EST 2019
;; MSG SIZE  rcvd: 439

root@kali:~#
```

2/2

# Analisi dei Record DNS

## Comando dnsenum

---

- Permette di raccogliere varie informazioni su un dominio
  - Indirizzi IP associati ad un dominio
  - Server DNS associati ad un dominio
  - Record MX associati ad un dominio
  - Altri record associati ad un dominio
  - Etc

# Analisi dei Record DNS

## Comando dnsenum

---

- Permette anche di ottenere i nomi dei sottodomini (hostname)
  - Tramite tecniche di **Brute Forcing**, usando una lista di nomi fornita in input
  - Kali fornisce già due file contenenti liste di nomi (*wordlist*) dei sottodomini
    - **dns.txt** che contiene circa **1480** nomi di sottodominio
    - **dns-big.txt** che contiene circa **266930** nomi di sottodominio

# Analisi dei Record DNS

## Comando dnsenum

---

- Oltre ad essere utilizzato per ottenere informazioni sul DNS, **dnsenum** fornisce anche altre funzionalità
  - Effettuare Zone Transfer (ZT)
  - Individuare i blocchi di rete /24 appartenenti al dominio
  - Effettuare reverse lookup sugli indirizzi IP appartenenti a tali blocchi
  - Usare i thread per processare differenti query
  - Etc

# Analisi dei Record DNS

## Comando dnsenum

- Per maggiori informazioni sul comando **dnsenum**
  - **man dnsenum**

```
NAME
    dnsenum -- multithread script to enumerate information on a domain
    and to discover non-contiguous IP blocks

VERSION
    dnsenum version 1.2.6

SYNOPSIS
    dnsenum [options] <domain> -f dns.txt

DESCRIPTION
    Supported operations: nslookup, zonetransfer, google scraping,
    domain brute force (support also recursion), whois ip and reverse
    lookups.

    Operations:
        • 1) Get the host's address (A record).
        • 2) Get the nameservers (threaded).
        • 3) Get the MX record (threaded).
        • 4) Perform AXFR queries on nameservers (threaded).
```

# Analisi dei Record DNS

## Comando dnsenum – Esempio 1

➤ **dnsenum zonetransfer.me**

root@kali:~# dnsenum zonetransfer.me
Smartmatch is experimental at /usr/bin/dnsenum line 698.
Smartmatch is experimental at /usr/bin/dnsenum line 698.
dnsenum VERSION:1.2.4
----- zonetransfer.me -----
<b>Host's addresses:</b>
zonetransfer.me. 6691 IN A 5.196.105.14
<b>Name Servers:</b>
nsztm1.digi.ninja. 10299 IN A 81.4.108.41 nsztm2.digi.ninja. 10799 IN A 34.225.33.2

**Indirizzo IP associato al dominio**

**Name Server associati al dominio**

# Analisi dei Record DNS

## Comando dnsenum – Esempio 1

➤ **dnsenum zonettransfer.me**

Mail (MX) Servers:

Server responsabili della  
gestione delle e-mail  
relative al dominio

ASPMX5.GOOGLEMAIL.COM.	292	IN	A	173.194.202.26
ASPMX3.GOOGLEMAIL.COM.	292	IN	A	172.217.194.27
ASPMX4.GOOGLEMAIL.COM.	292	IN	A	108.177.97.27
ALT1.ASPMX.L.GOOGLE.COM.	292	IN	A	209.85.233.27
ALT2.ASPMX.L.GOOGLE.COM.	292	IN	A	172.217.194.27
ASPMX.L.GOOGLE.COM.	292	IN	A	64.233.166.27
ASPMX2.GOOGLEMAIL.COM.	292	IN	A	209.85.233.27

Trying Zone Transfers and getting Bind Versions:

Trying Zone Transfer for zonettransfer.me  
zonetransfer.me.

Tentativo di ZT e rilevazione delle  
versioni dei server DNS in uso

SOA	IN	MINIMO	CASIO
300	IN	MINIMO	CASIO
301	IN	TXT	(
7200	IN	MX	0
7200	IN	MX	10
7200	IN	MX	10
7200	IN	MX	20

Output parziale ZT

# Analisi dei Record DNS

Comando dnsenum – Esempio 1

➤ **dnsenum zonettransfer.me**

In questo caso lo ZT va a buon fine

Output parziale ZT

Info.zonettransfer.me.	7200	IN	TXT	(
internal.zonettransfer.me.	300	IN	NS	intns1.zonetran
sfer.me.				
internal.zonettransfer.me.	300	IN	NS	intns2.zonetran
sfer.me.				
intns1.zonettransfer.me.	300	IN	A	81.4.108.41
intns2.zonettransfer.me.	300	IN	A	167.88.42.94
office.zonettransfer.me.	7200	IN	A	4.23.39.254
ipv6actnow.org.zonettransfer.me.	7200	IN	AAAA	2001:67c:2e8:11
::c100:1332				
owa.zonettransfer.me.	7200	IN	A	207.46.197.32
robinwood.zonettransfer.me.	302	IN	TXT	"Robin
rp.zonettransfer.me.	321	IN	RP	(
sip.zonettransfer.me.	3333	IN	NAPTR	(
sqliz.zonettransfer.me.	300	IN	TXT	"
sshock.zonettransfer.me.	7200	IN	TXT	"()
staging.zonettransfer.me.	7200	IN	CNAME	www.sydneyopera
house.com.				
alltcpportsopen.firewall.test.zonettransfer.me.	301	IN	A	127.0.0.
1				
testing.zonettransfer.me.	301	IN	CNAME	www.zonetra
r.me.				
vpn.zonettransfer.me.	4000	IN	A	174.36.59.154

# Analisi dei Record DNS

Comando dnsenum – Esempio 1

➤ **dnsenum zonetransfer.me**

Output parziale

```
Brute forcing with /usr/share/dnsenum/dns.txt:  
  
zonetransfer.me class C netranges:  
  
4.23.39.0/24  
5.196.105.0/24  
52.91.28.0/24  
74.125.206.0/24  
81.4.108.0/24  
143.228.181.0/24  
167.88.42.0/24  
174.36.59.0/24  
202.14.81.0/24  
207.46.197.0/24  
  
N. N. N. H.
```

**Brute-force** mediante una **wordlist** per individuare indirizzi IP ed hostname

10 blocchi /24, ciascun blocco è costituito da 256 indirizzi IP

N. = Network  
H. = Host

# Analisi dei Record DNS

Comando dnsenum – Esempio 1

➤ **dnsenum zonetransfer.me**

```
Performing reverse lookup on 2560 ip addresses:  
-----  
0 results out of 2560 IP addresses.  
  
zonetransfer.me ip blocks:  
-----  
  
done.  
root@kali:~#
```

**Tentativo di *reverse lookup* sugli indirizzi IP appartenenti al dominio**

# Analisi dei Record DNS

Comando dnsenum – Esempio 2

---

- **Obiettivo:** ricavare informazioni di rete sull'asset **unisa.it**
  - Indirizzi IP
  - Name Server
  - Sottodomini (Hostname)
  - Etc



# Analisi dei Record DNS

Comando dnsenum – Esempio 2

➤ **dnsenum unisa.it**

unisa.it				
dnsenum VERSION:1.2.6				
Host's addresses:				
-----				
unisa.it.	299	IN	A	193.205.185.20
Name Servers:				
ns1.garr.net.	21102	IN	A	193.206.141.38
dns-001.unisa.it.	70	IN	A	193.205.160.139
ns.unisa.it.	299	IN	A	193.205.160.3
Mail (MX) Servers:				
ASPMX.L.GOOGLE.COM.	292	IN	A	64.233.167.26

# Analisi dei Record DNS

Comando dnsenum – Esempio 2

➤ **dnsenum unisa.it**

```
Mail (MX) Servers:  
-----  
ASPMX.L.GOOGLE.COM.          292   IN   A    64.233.167.26  
ALT1.ASPMX.L.GOOGLE.COM.      292   IN   A    209.85.233.26  
ALT2.ASPMX.L.GOOGLE.COM.      292   IN   A    172.253.118.26  
ALT3.ASPMX.L.GOOGLE.COM.      292   IN   A    108.177.97.26  
ALT4.ASPMX.L.GOOGLE.COM.      292   IN   A    74.125.28.26  
  
Trying Zone Transfers and getting Bind Versions:  
-----  
Trying Zone Transfer for unisa.it on ns1.garr.net ...  
AXFR record query failed: REFUSED  
  
Trying Zone Transfer for unisa.it on dns-001.unisa.it ...  
AXFR record query failed: REFUSED  
  
Trying Zone Transfer for unisa.it on ns.unisa.it ...  
AXFR record query failed: REFUSED
```

**Server responsabili della gestione delle e-mail relative al dominio**

**Tentativo di ZT e rilevazione delle versioni dei server DNS in uso**

**In questo caso lo ZT non va a buon fine**

# Analisi dei Record DNS

Comando dnsenum – Esempio 2

➤ **dnsenum unisa.it**

Sottodomini di  
unisa.it ricavati  
in base alla  
wordlist  
**dns.txt**

Brute forcing with /usr/share/dnsenum/dns.txt:				
abc.unisa.it.	299	IN	CNAME	srv-002.unisa.it
t.				
srv-002.unisa.it.	299	IN	A	193.205.160.14
beta.unisa.it.	299	IN	CNAME	www.unisa.it.
www.unisa.it.	299	IN	CNAME	www3.unisa.it.
www3.unisa.it.	299	IN	A	193.205.185.20
di.unisa.it.	21599	IN	A	192.41.218.193
forum.unisa.it.	299	IN	A	193.205.184.105
ftp.unisa.it.	299	IN	A	193.205.184.63
intranet.unisa.it.	299	IN	A	193.205.167.63

Output parziale

# Analisi dei Record DNS

## Comando dnsenum – Esempio 2

➤ **dnsenum unisa.it**

```
unisa.it class C netranges:
```

```
unisatfiles  
192.41.218.0/24  
193.205.160.0/24  
193.205.167.0/24  
193.205.171.0/24  
193.205.176.0/24  
193.205.184.0/24  
193.205.185.0/24
```

7 blocchi /24, ciascun blocco  
composto da 256 indirizzi IP

N. N. N. H.

**N. = Network**

**H. = Host**

# Analisi dei Record DNS

Comando dnsenum – Esempio 2

➤ **dnsenum unisa.it**

```
unisa.it class C netranges:
```

```
unisatfiles  
192.41.218.0/24  
193.205.160.0/24  
193.205.167.0/24  
193.205.171.0/24  
193.205.176.0/24  
193.205.184.0/24  
193.205.185.0/24
```

1792 indirizzi IP

N. N. N. H.

**N. = Network**

**H. = Host**

# Analisi dei Record DNS

Comando dnsenum – Esempio 2

➤ **dnsenum unisa.it**

```
unisa.it class C netranges:
```

```
unisatfiles  
192.41.218.0/24  
193.205.160.0/24  
193.205.167.0/24  
193.205.171.0/24  
193.205.176.0/24  
193.205.184.0/24  
193.205.185.0/24
```



1792 indirizzi IP

N. N. N. H.

**N. = Network**

**H. = Host**

# Analisi dei Record DNS

Comando dnsenum – Esempio 2

➤ **dnsenum unisa.it**

Performing reverse lookup on 1792 ip addresses:				
3.160.205.193.in-addr.arpa.	86400	IN	PTR	ns.unisa.it.
9.160.205.193.in-addr.arpa.	86400	IN	PTR	esa-mx-1.unisa.it.
10.160.205.193.in-addr.arpa.	86400	IN	PTR	smtp1.unisa.it.
11.160.205.193.in-addr.arpa.	86400	IN	PTR	esa-mx-2.unisa.it.
12.160.205.193.in-addr.arpa.	86400	IN	PTR	cluster-idm.unisa.it.
13.160.205.193.in-addr.arpa.	86400	IN	PTR	cluster-db.unisa.it.
14.160.205.193.in-addr.arpa.	86400	IN	PTR	srv-002.unisa.it.
15.160.205.193.in-addr.arpa.	86400	IN	PTR	newweb.unisa.it.
20.160.205.193.in-addr.arpa.	86400	IN	PTR	www3.unisa.it.
23.160.205.193.in-addr.arpa.	86400	IN	PTR	lbip.unisa.it.
24.160.205.193.in-addr.arpa.	86400	IN	PTR	ilo-lb01.unisa.it.
25.160.205.193.in-addr.arpa.	86400	IN	PTR	ilo-lb02.unisa.it.
59.160.205.193.in-addr.arpa.	86400	IN	PTR	dmzsw.unisa.it.
60.160.205.193.in-addr.arpa.	86400	IN	PTR	dmz1.unisa.it.
61.160.205.193.in-addr.arpa.	86400	IN	PTR	dmz2.unisa.it.
62.160.205.193.in-addr.arpa.	86400	IN	PTR	6000r.unisa.it.
129.160.205.193.in-addr.arpa.	86400	IN	PTR	virtual.unisa.it.
130.160.205.193.in-addr.arpa.	86400	IN	PTR	esse3web.unisa.it.

Reverse lookup per  
ciascun indirizzo IP

Output parziale

# Analisi dei Record DNS

Comando dnsenum – Esempio 2

➤ **dnsenum unisa.it**

Indirizzi IP su cui è  
stato possibile  
effettuare il  
reverse lookup

**unisa.it ip blocks:**

```
193.205.160.3/32
193.205.160.9/32
193.205.160.10/31
193.205.160.12/30
193.205.160.20/32
193.205.160.23/32
193.205.160.24/31
193.205.160.59/32
193.205.160.60/31
```

Output parziale

<https://www.ipaddressguide.com/cidr>

# Analisi dei Record DNS

Comando dnsenum – Esempio 2

➤ **dnsenum unisa.it**

Indirizzi IP su cui è  
stato possibile  
effettuare il  
reverse lookup

```
193.205.184.252/31
193.205.184.254/32
193.205.185.1/32
193.205.185.2/31
193.205.185.5/32
193.205.185.6/32
193.205.185.8/31
193.205.185.11/32
193.205.185.12/30
193.205.185.16/30
193.205.185.23/32
193.205.185.24/32
```

```
done.
```

```
root@kali:~#
```

<https://www.ipaddressguide.com/cidr>

Output parziale

# Analisi dei Record DNS

## Comando fierce

---

- Strumento che utilizza diverse tecniche per trovare indirizzi IP e sottodomini (hostname) di un determinato dominio
  
- Per trovare i nomi dei sottodomini utilizza una *wordlist* (dizionario) fornita in input dall'utente
  
- Permette di individuare spazi di indirizzamento IP non contigui

# Analisi dei Record DNS

# Comando fierce

- Per maggiori informazioni sul funzionamento di **fierce**
  - **man fierce**

**FIERCE(1)** General Commands Manual **FIERCE(1)**

**NAME**  
fierce - DNS scanner that helps locate non-contiguous IP space and hostnames against specified domains.

**SYNOPSIS**  
**fierce** [-h] [--domain DOMAIN] [--connect] [--wide] [--traverse TRA-VERSE] [--search SEARCH [SEARCH ...]] [--range RANGE] [--delay DELAY] [--subdomains SUBDOMAINS [SUBDOMAINS ...]] | --subdomain-file SUBDO-MAIN\_FILE] [--dns-servers DNS\_SERVERS [DNS\_SERVERS ...]] | --dns-file DNS\_FILE] [--tcp]

**DESCRIPTION**  
Fierce is a semi-lightweight scanner that helps locate non-contiguous IP space and hostnames against specified domains. It's really meant as a pre-cursor to nmap, OpenVAS, nikto, etc, since all of those require that you already know what IP space you are looking for. This does not perform exploitation and does not scan the whole internet indiscriminately. It is meant specifically to locate likely targets both inside and outside a corporate network. Because it uses DNS primarily you will often find mis-configured networks that leak internal address space. That's especially useful in targeted malware. Originally written by RSnake along with others at <http://ha.ckers.org/>. This is sim-

Manual page fierce(1) line 1 (press h for help or q to quit)

# Analisi dei Record DNS

## Comando **fierce** – Esempio

- Esempio di utilizzo di **fierce** sul dominio **unisa.it**
- **fierce --domain unisa.it**

```
root@kali:~# fierce --wide --domain unisa.it
NS: ns1.garr.net. dns-001.unisa.it. ns.unisa.it
SOA: ns.unisa.it. (193.205.160.3)
Zone: failure
Wildcard: failure
Found: abc.unisa.it. (193.205.160.14)
Nearby:
{'193.205.160.10': 'smtp1.unisa.it.',
 '193.205.160.11': 'esa-mx-2.unisa.it.',
 '193.205.160.12': 'cluster-idm.unisa.it.',
 '193.205.160.129': 'virtual.unisa.it.',
 '193.205.160.13': 'cluster-db.unisa.it.',
 '193.205.160.130': 'esse3web.unisa.it.',
 '193.205.160.131': 'testwebradio.unisa.it.',
 '193.205.160.133': 'dns-002.unisa.it.',
 '193.205.160.134': 'pino.unisa.it.',
 '193.205.160.135': 'gre-pino.unisa.it.',
 '193.205.160.136': 'baobab.unisa.it.',
 '193.205.160.137': 'papaja.unisa.it.',
 '193.205.160.138': 'dns-003.unisa.it.',
 '193.205.160.139': 'dns-001.unisa.it.',
 '193.205.160.14': 'srv-002.unisa.it.',
 '193.205.160.140': 'test3.unisa.it.',
 '193.205.160.141': 'cedro.unisa.it.',
 '193.205.160.142': 'test1.unisa.it.'}
```

Output Parziale

DNS per **unisa.it**

Tentativo fallito di  
DNS Zone Transfer

Sottodomini di  
**unisa.it**

# Analisi dei Record DNS

## Comando **fierce** – Esempio

- Esempio di utilizzo di **fierce** sul dominio **unisa.it**
- **fierce --domain unisa.it**

Output Parziale

```
Found: antivirus.unisa.it. (193.205.160.152)
Found: auth.unisa.it. (193.205.185.6)
Nearby:
{'193.205.185.1': 'portaleappalti.unisa.it.',
 '193.205.185.10': 'www.edisu.sa.it.',
 '193.205.185.11': 'webmail-2.unisa.it.',
 '193.205.185.12': 'hd.unisa.it.',
 '193.205.185.13': 'traced.unisa.it.',
 '193.205.185.14': 'tracedlinux.unisa.it.',
 '193.205.185.15': 'openproject.unisa.it.',
 '193.205.185.16': 'universiade.unisa.it.',
 '193.205.185.17': 'adisu-sistemi.unisa.it.',
 '193.205.185.18': 'www4.unisa.it.',
 '193.205.185.19': 'wrstreaming.unisa.it.',
 '193.205.185.2': 'appalti-bo.unisa.it.',
 '193.205.185.23': 'elearning.test.unisa.it.',
 '193.205.185.24': 'bproxy.unisa.it.',
 '193.205.185.3': 'archibus.unisa.it.',
 '193.205.185.5': 'musa-as.unisa.it.',
 '193.205.185.6': 'auth.unisa.it.',
 '193.205.185.8': 'simu.unisa.it.',
 '193.205.185.9': 'auth-test.unisa.it.'}
Found: beta.unisa.it. (193.205.185.20)
Found: cd.unisa.it. (193.205.185.20)
Found: cert.unisa.it. (192.41.218.27)
Nearby:
{'192.41.218.1': 'udsab.dia.unisa.it.',
 '192.41.218.10': 'capri.dia.unisa.it.',
```

Sottodomini di  
unisa.it

# Analisi dei Record DNS

DMitry

---

- *Deepmagic Information Gathering Tool*
- Strumento multifunzione per la raccolta di informazioni su un determinato dominio
- Permette di ottenere varie informazioni
  - Informazioni di registrazione (WHOIS) relative ad un dominio
  - Informazioni sul dominio raccolte da Netcraft.com
  - Sottodomini
  - Indirizzi e-mail «associati» al dominio
  - Etc

# Analisi dei Record DNS

DMitry

- **Osservazione:** DMitry permette di ottenere mediante un singolo strumento informazioni che potrebbero essere altrimenti ottenute utilizzando diversi strumenti
- Per maggiori informazioni su DMitry
  - `man dmitry`

```
DMitry(1)                                General Commands Manual      DMitry(1)

NAME
    DMitry - Deepmagic Information Gathering Tool

SYNOPSIS
    dmitry [Options] host

DESCRIPTION
    DMitry (Deepmagic Information Gathering Tool) is a UNIX/(GNU)Linux command line application with the ability to gather as much information as possible about a host.

    Basic functionality of DMitry allows for information to be gathered about a target host from a simple whois lookup on the target to uptime reports and TCP portscans.

    The application is considered a tool to assist in information gathering when information is required quickly by removing the need to enter multiple commands and the timely process of searching through data from multiple sources.
```

Information Gathering

# Analisi dei Record DNS

## DMitry – Esempio

---

➤ **Esempio:** useremo DMitry per effettuare le seguenti operazioni sul dominio **unisa.it**

- Effettuare un Whois lookup
- Otttenere informazioni da Netcraft.com
- Cercare tutti i possibili sottodomini
- Cercare tutte le possibili e-mail relative a tale dominio

# Analisi dei Record DNS

## DMitry – Esempio

➤ `dmitry -iwnse unisa.it`

```
root@kali:~# dmitry -iwnse unisa.it
Deepmagic Information Gathering Tool
"There be some deep magic going on"

HostIP:193.205.185.20
HostName:unisa.it

Gathered Inet-whois information for 193.205.185.20
_____
inetnum:          193.205.160.0 - 193.205.191.255
netname:          UNISA-CAMPUS-NET
descr:            Università degli Studi di Salerno
country:          IT
admin-c:          FR7236-RIPE
tech-c:           GL965-RIPE
tech-c:           SF12137-RIPE
PA
remarks:          This prefix is statically assigned
remarks:          To notify abuse mailto: cert@garr.it
remarks:          GARR - Italian academic and research network
mnt-by:           GARR-LIR
```

Risultato del  
whois lookup

Output parziale

# Analisi dei Record DNS

## DMitry – Esempio

Informazioni sul dominio

```
Gathered Inic-whois information for unisa.it

Domain:          unisa.it
Status:          ok
Signed:          no
Created:         1996-01-29 00:00:00
Last Update:    2020-02-14 00:56:42
Expire Date:   2021-01-29

Registrant
Organization:   Universita' di Salerno
Address:        Universita' di Salerno
                Baronissi
                84081
                SA
                IT
Created:        2007-03-01 10:47:03
e:              2011-03-24 11:01:07

Admin Contact
Name:            Giuseppe Cattaneo
Address:         Universita' di Salerno
```

Output parziale

# Analisi dei Record DNS

DMitry – Esempio

Alcuni sottodomini  
di unisa.it

```
Gathered Netcraft information for unisa.it
_____
Retrieving Netcraft.com information for unisa.it
Netcraft.com Information gathered
_____
Gathered Subdomain information for unisa.it
_____
Searching Google.com:80 ...
HostName:www.unisa.it
HostIP:193.205.185.20
HostName:pec.unisa.it
HostIP:127.0.0.1
HostName:web.unisa.it
HostIP:193.205.185.20
HostName:www.di.unisa.it
HostIP:193.205.185.20
HostName:Di.unisa.it
HostIP:192.41.218.193
HostName:www.cla.unisa.it
```

Output parziale

# Analisi dei Record DNS

## DMitry – Esempio

Output parziale

```
Gathered E-Mail information for unisa.it
_____
Searching Google.com:80 ...
lpassegg@unisa.it
pcapuano@unisa.it
plongo@unisa.it
prorettore@unisa.it
auletta@unisa.it
neri@unisa.it
[REDACTED]
ecaracciolo@unisa.it
llionetti@unisa.it
vvitale@unisa.it
fbasile@unisa.it
rettore@unisa.it
l.rizzo@unisa.it
```

Alcune e-mail relative ad unisa.it

# Analisi dei Record DNS

## DMitry – Esempio

Output parziale

```
dericca@unisa.it
gmarsico@unisa.it
segrrett@unisa.it
amarabotti@unisa.it
luchini@unisa.it
caip2019@unisa.it
robttag@unisa.it
rorroco@unisa.it
cdipietr@unisa.it
alieto@unisa.it
raffaele@unisa.it
direttoreclla@unisa.it
auletta@dia.unisa.it
Searching Altavista.com:80 ...
Found 34 E-Mail(s) for host unisa.it, Searched 0 pages containing 0 results

All scans completed, exiting
root@kali:~#
```

**Alcune e-mail relative ad unisa.it**

# Outline

---

- Concetti Preliminari
- Raccolta di Informazioni da Risorse Web-Based
- Raccolta delle Informazioni di Registrazione
- Raccolta delle Informazioni di Routing
- Raccolta di Informazioni dai Record DNS
- **Raccolta di Informazioni mediante Crawler**
- Raccolta di informazioni dal Dark Web
- Altri Strumenti e Servizi per Raccogliere Informazioni

# Utilizzo di Crawler

---

- Strumenti che permettono di raccogliere una grande quantità di informazioni, generalmente appartenenti all'*Open Source INTelligence (OSINT)*
  
- Tali informazioni possono
  - Provenire da varie fonti
    - Motori di Ricerca, Social Network , etc
  - Essere di vario tipo
    - Informazioni di dominio, indirizzi di posta elettronica, informazioni personali, documenti, metadati relativi ai documenti, etc

# Utilizzo di Crawler

---

- **Osservazione:** Talvolta, alcuni di questi strumenti potrebbero fornire risultati parziali o non fornirne affatto
  - A causa di «blocchi» attuati dai motori di ricerca utilizzati da tali strumenti
- Alcune possibili soluzioni
  - Ripetere l'esecuzione di tali strumenti dopo qualche minuto (o qualche ora)
  - Eseguire tali strumenti utilizzando *proxy chain* in modalità *round-robin* o altri meccanismi per lo *spoofing dell'indirizzo IP*
  - Diminuire il numero e la frequenza di query parallele effettuate da tali strumenti



# Utilizzo di Crawler

---

- **Osservazione:** Il risultato fornito dagli strumenti che vedremo è difficilmente riproducibile e dipende da molteplici fattori
  - Ripetendo più volte l'esecuzione di tali strumenti sul medesimo asset, anche utilizzando gli stessi parametri, si potrebbero ottenere risultati diversi ad ogni nuova esecuzione degli strumenti stessi



# Utilizzo di Crawler

Surface Mapping ed Asset Discovery – theHarvester

---

- Permette di raccogliere informazioni OSINT da varie fonti
  - Yahoo
  - ZoomEye
  - Bing
  - Baidu
  - DuckDuckGo
  - RocketReach
  - Etc



# Utilizzo di Crawler

# theHarvester

- Per ottenere informazioni sull'utilizzo di questo comando è sufficiente digitarlo seguito dal parametro **-h**
    - **theHarvester -h**



# Utilizzo di Crawler

## theHarvester

- Per ottenere informazioni sull'utilizzo di questo comando è sufficiente digitarlo seguito dal parametro **-h**

- **theHarvester -h**

```
options:
  -h, --help            show this help message and exit
  -d DOMAIN, --domain DOMAIN
                        Company name or domain to search.
  -l LIMIT, --limit LIMIT
                        Limit the number of search results, default=500.
  -S START, --start START
                        Start with result number X, default=0.
  -p, --proxies         Use proxies for requests, enter proxies in
                        proxies.yaml.
  -s, --shodan          Use Shodan to query discovered hosts.
  --screenshot SCREENSHOT
                        Take screenshots of resolved domains specify output
                        directory: --screenshot output_directory
  -v, --virtual-host   Verify host name via DNS resolution and search for
                        virtual hosts.
  -e DNS_SERVER, --dns-server DNS_SERVER
                        DNS server to use for lookup.
  -r, --take-over      Check for takeovers.
  -n, --dns-lookup     Enable DNS server lookup, default False.
  -c, --dns-brute      Perform a DNS brute force on the domain.
```



# Utilizzo di Crawler

## theHarvester

- Per ottenere informazioni sull'utilizzo di questo comando è sufficiente digitarlo seguito dal parametro **-h**

- **theHarvester -h**

```
options:  
  -h, --help            show this help message and exit  
  -d DOMAIN, --domain DOMAIN  
                        Company name or domain to search.  
  -l LIMIT, --limit LIMIT  
                        Limit the number of search results, default=500.  
  -S START, --start START  
                        Start with result number X, default=0.  
  -p, --proxies         Use proxies for requests, enter proxies in  
                        proxies.yaml.  
  -s, --shodan          Use Shodan to query discovered hosts.  
  --screenshot SCREENSHOT  
                        Take screenshots of resolved domains specify output  
                        directory: --screenshot output_directory  
  -v, --virtual-host   Verify host name via DNS resolution and search for  
                        virtual hosts.  
  -e DNS_SERVER, --dns-server DNS_SERVER  
                        DNS server to use for lookup.  
  -r, --take-over      Check for takeovers.  
  -n, --dns-lookup     Enable DNS server lookup, default False.  
  -c, --dns-brute     Perform a DNS brute force on the domain.
```



# Utilizzo di Crawler

## theHarvester

- Per ottenere informazioni sull'utilizzo di questo comando è sufficiente digitarlo seguito dal parametro **-h**

- **theHarvester -h**

```
options:  
  -h, --help            show this help message and exit  
  -d DOMAIN, --domain DOMAIN  
                        Company name or domain to search.  
  -l LIMIT, --limit LIMIT  
                        Limit the number of search results, default=500.  
  -S START, --start START  
                        Start with result number X, default=0.  
  -p, --proxies         Use proxies for requests, enter proxies in  
                        proxies.yaml.  
  -s, --shodan          Use Shodan to query discovered hosts.  
  --screenshot SCREENSHOT  
                        Take screenshots of resolved domains specify output  
                        directory: --screenshot output_directory  
  -v, --virtual-host   Verify host name via DNS resolution and search for  
                        virtual hosts.  
  -e DNS_SERVER, --dns-server DNS_SERVER  
                        DNS server to use for lookup.  
  -r, --take-over      Check for takeovers.  
  -n, --dns-lookup     Enable DNS server lookup, default False.  
  -c, --dns-brute     Perform a DNS brute force on the domain.
```



# Utilizzo di Crawler

## theHarvester

- Per ottenere informazioni sull'utilizzo di questo comando è sufficiente digitarlo seguito dal parametro **-h**

- **theHarvester -h**

```
options:  
  -h, --help            show this help message and exit  
  -d DOMAIN, --domain DOMAIN  
                        Company name or domain to search.  
  -l LIMIT, --limit LIMIT  
                        Limit the number of search results, default=500.  
  -S START, --start START  
                        Start with result number X, default=0.  
  -p, --proxies         Use proxies for requests, enter proxies in  
                        proxies.yaml.  
  -s, --shodan          Use Shodan to query discovered hosts.  
  --screenshot Screenshot  
                        Take screenshots of resolved domains specify output  
                        directory: --screenshot output_directory  
  -v, --virtual-host   Verify host name via DNS resolution and search for  
                        virtual hosts.  
  -e DNS_SERVER, --dns-server DNS_SERVER  
                        DNS server to use for lookup.  
  -r, --take-over      Check for takeovers.  
  -n, --dns-lookup     Enable DNS server lookup, default False.  
  -c, --dns-brute     Perform a DNS brute force on the domain.
```



# Utilizzo di Crawler

## theHarvester

- Per ottenere informazioni sull'utilizzo di questo comando è sufficiente digitarlo seguito dal parametro **-h**

- **theHarvester -h**

```
-f FILENAME, --filename FILENAME
                                Save the results to an XML and JSON file.
-b SOURCE, --source SOURCE
                                anubis, baidu, bevigil, binaryedge, bing, bingapi,
                                bufferoverun, censys, certspotter, crtsh,
                                dnsdumpster, duckduckgo, fullhunt, github-code,
                                hackertarget, hunter, intelx, omnisint, otx,
                                pentesttools, projectdiscovery, qwant, rapiddns,
                                rocketreach, securityTrails, sublist3r,
                                threatcrowd, threatminer, urlscan, virustotal,
                                yahoo, zoomeye
```



# Utilizzo di Crawler

## theHarvester

- Per ottenere informazioni sull'utilizzo di questo comando è sufficiente digitarlo seguito dal parametro **-h**

➤ **theHarvester -h**

```
-f FILENAME, --filename FILENAME
                                Save the results to an XML and JSON file.
-b SOURCE, --source SOURCE
                                anubis, baidu, bevigil, binaryedge, bing, bingapi,
                                bufferoverun, censys, certspotter, crtsh,
                                dnsdumpster, duckduckgo, fullhunt, github-code,
                                hackertarget, hunter, intelx, omnisint, otx,
                                pentesttools, projectdiscovery, qwant, rapiddns,
                                rocketreach, securityTrails, sublist3r,
                                threatcrowd, threatminer, urlscan, virustotal,
                                yahoo, zoomeye
```



# Utilizzo di Crawler

theHarvester – Fonti Attive di Informazione

---

- Consente di effettuare *Passive Information Gathering* ed *Active Information Gathering* a seconda delle fonti di informazione utilizzate
  - Alcune Fonti «Attive» di Informazione
    - **DNS Brute Force:** raccolta di informazioni sui record DNS basata su *brute force* (utilizzo di una *wordlist*)
    - **DNS Reverse Lookup:** reverse lookup di IP scoperti, per trovare hostname
    - Etc



# Utilizzo di Crawler

theHarvester – Fonti Passive di Informazione

---

- Consente di effettuare *Passive Information Gathering* ed *Active Information Gathering* a seconda delle fonti di informazione utilizzate
  - Alcune Fonti «Passive» di Informazione
    - **Bing:** Motore di ricerca offerto da Microsoft
    - **Yahoo:** Motore di ricerca offerto da Yahoo
    - **Virustotal:** Motore di ricerca offerto da Virustotal
    - Etc



# Utilizzo di Crawler

theHarvester – Fonti Passive di Informazione

---

- TheHarvester usa fonti di informazioni (*moduli*) provenienti da terze parti
  - Alcune richiedono *una licenza (API keys)* per poter funzionare
  - Prima di poter utilizzare theHarvester su tali fonti è necessario registrarsi presso di esse



# Utilizzo di Crawler

## theHarvester – Fonti Passive di Informazione

---

- Alcune fonti (*moduli*) richiedono una licenza (*API keys*) per poter funzionare
  - **bing**: Motore di ricerca di Microsoft, accesso tramite API
  - **hunter**: Motore di ricerca di Hunter
  - **intelx**: Motore di ricerca di Intelx
  - **securityTrails**: Motore di ricerca di Security Trails (il più grande archivio al mondo di dati DNS storici)
  - Etc
  
- La licenza (*API keys*) per ciascun modulo deve essere impostata nel file **api-keys.yaml**



# Utilizzo di Crawler

## theHarvester – Fonti Passive di Informazione

---

- Alcune fonti (*moduli*) richiedono una licenza (*API keys*) per poter funzionare
  - **bing**: Motore di ricerca di Microsoft, accesso tramite API
  - **hunter**: Motore di ricerca di Hunter
  - **intelx**: Motore di ricerca di Intelx
  - **securityTrails**: Motore di ricerca di Security Trails (il più grande archivio al mondo di dati DNS storici)
  - Etc
  
- La licenza (*API keys*) per ciascun modulo deve essere impostata nel file **api-keys.yaml**

**N.B. Maggiore sarà il numero di API keys impostate, maggiori informazioni verranno restituite dallo strumento**



# Utilizzo di Crawler

## theHarvester – api-keys.yaml

- Esempio di contenuto del file **api-keys.yaml**

```
apikeys:  
  bing:  
    key:  
  
  github:  
    key:  
  
  hunter:  
    key:  
  
  intelx:  
    key: 9df61df0-84f7-4dc7-b34c-8ccfb8646ace  
  
  securityTrails:  
    key:
```



# Utilizzo di Crawler

theHarvester – Esempio

➤ theHarvester -d unisa.it -b all

## Output Parziale



# Utilizzo di Crawler

## theHarvester – Esempio

➤ **theHarvester -d unisa.it -b all**

Output Parziale

```
[!] Missing API key for binaryedge.  
[!] Missing API key for Censys ID and/or Secret.  
[!] Missing API key for fullhunt.  
[!] Missing API key for Github.  
[!] Missing API key for Hunter.  
[!] Missing API key for Intelx.  
[!] Missing API key for PentestTools.  
[!] Missing API key for ProjectDiscovery.  
[!] Missing API key for RocketReach.  
[!] Missing API key for Securitytrail.  
[!] Missing API key for virustotal.  
[!] Missing API key for zoomeye.
```

**Vengono innanzitutto segnalate tutte le API Keys mancanti**



# Utilizzo di Crawler

## theHarvester – Esempio

➤ **theHarvester -d unisa.it -b all**

Output Parziale

```
[*] ASNs found: 1
-----
AS137

[*] InterestingUrls found: 11
-----
http://biograph2014.unisa.it/
http://proxy.unisa.it/
https://auth.syntonia.unisa.it/carbon/admin/login.jsp
https://discourse.di.unisa.it/login
https://edu.diin.unisa.it/
https://mathedu.diem.unisa.it/
https://web.unisa.it/
https://web.unisa.it/didattica/master/bandi?anno=2021&bando=4611
https://web.unisa.it/didattica/master/bandi?anno=2021&bando=4611
https://www.giovani.unisa.it/
https://www.unisa.it/
```

Autonomous Systems (AS) ed URL «potenzialmente interessanti» rilevati



# Utilizzo di Crawler

## theHarvester – Esempio

➤ **theHarvester -d unisa.it -b all**

Output Parziale

```
[*] IPs found: 92
3.124.95.236
13.57.92.51
35.157.109.54
35.158.75.116
50.18.241.247
52.31.164.244
54.184.58.211
81.88.52.159
89.46.105.52
94.130.24.43
94.177.175.11
130.186.6.8
130.186.7.111
130.186.7.114
130.186.12.53
130.186.27.50
130.186.27.89
130.186.29.4
```

**Indirizzi IP rilevati**



# Utilizzo di Crawler

## theHarvester – Esempio

➤ **theHarvester -d unisa.it -b all**

Output Parziale

```
[*] Emails found: 49
aabate@unisa.it
aajwad@unisa.it
aarcamone@unisa.it
adibartolomeo@unisa.it
adicrescenz@unisa.it
antonioadicrescenz@unisa.it
apopol@unisa.it
apuca@unisa.it
caso@diima.unisa.it
cdemaio@unisa.it
cguarnaccia@unisa.it
concilio@unisa.it
ctepedino@unisa.it
dbaldantoni@unisa.it
delmal@dia.unisa.it
desiena@physics.unisa.it
didattica.cla@unisa.it
eabdurrahman@unisa.it
fbasile@unisa.it
fdecaro@unisa.it
furban@unisa.it
```

E-mail rilevate



# Utilizzo di Crawler

## theHarvester – Esempio

➤ **theHarvester -d unisa.it -b all**

Output Parziale

```
[*] Hosts found: 5420
2048.k8s.unisa.it:193.205.185.27
50anni.informatica.unisa.it:192.41.218.22
6000r.faceco.unisa.it:193.205.172.79
aagg1.edisu.unisa.it:193.205.170.131
abate.edisu.unisa.it:193.205.170.152
abate117.diciv.unisa.it:193.205.178.52
abbagnale.dssp.unisa.it:193.205.171.125
abbondanza.dsss.unisa.it:193.205.172.160
abelarda.dimec.unisa.it:193.205.183.14
abellinumtour.centroictbc.unisa.it:193.205.190.220
acastagnola.seda.unisa.it:193.205.168.201
acastagnola.seda.unisa.it
accessocampus.unisa.it:www3.unisa.it
accessocampus.unisa.it:www.unisa.it
```

Host rilevati



# Utilizzo di Crawler

## Surface Mapping ed Asset Discovery – OWASP Maryam

---

- Framework modulare scritto in Python, che permette di raccogliere, in maniera veloce e comoda, numerose informazioni sia in maniera passiva che attiva
  
- Consente di
  - Ottener
    - Indirizzi e-mail, documenti, metadati, informazioni presenti sui social network, etc, grazie all'utilizzo di motori di ricerca
    - Informazioni su componenti Web-based di un determinato asset
      - Visitando pagine Web e ricercando informazioni all'interno di esse (file, collegamenti, etc)
      - Identificando WebApp, Web Application Firewall (WAF), file interessanti ed importanti
  - Enumerare sottodomini, indirizzi IP ed altre informazioni riguardanti il DNS (*record*)
  - Ottener report in vari formati
  - Etc

# Utilizzo di Crawler

## OWASP Maryam – Installazione ed Help

- È necessario installarlo su Kali
  - `apt install maryam`
- Per avviarlo è sufficiente digitare il seguente comando da Terminale
  - `maryam`
- Dalla consolle di Maryam, digitando il comando `help` è possibile conoscerne l'utilizzo

```
OWASP Maryam(v2.5.0): Open-source Intelligence Framework.
To show the framework help, run 'help' command.
[maryam][default] > help

Commands (type [help|?] <topic>):
exit           Exits the framework
help            Displays this menu
reload          Reloads all modules
report          Get report from the Gathers and save it to the other formats
search          Searches available modules
set             Sets module options
shell            Executes shell commands
show             Shows various framework items
unset            Unsets module options
update          Update modules via module name
web              Manage web/api interface
workspaces       Manages workspaces

[maryam][default] > █
```

# Utilizzo di Crawler

## OWASP Maryam – Moduli

- Maryam fornisce numerosi moduli, raggruppati in quattro categorie, ciascuna delle quali permette di raccogliere una determinata categoria di informazioni

- **show modules**

Footprint	Osint	Search
tldbrute	onion_search	piratebay
entry_points	phone_number_search	wikileaks
crawl_pages	image_search	yahoo
dnsbrute	github_leaks	google
filebrute	article_search	etools
<b>Iris</b>		
cluster	famous_person	instagram
topicmodeling	email_pwned	arxiv
iris_cluster	email_search	tiktok
sentiment	dns_search	sanctionsearch
iris	username_search	stackoverflow
	crawler	discord
	cloud_storage	trello
	social_nets	quora
	docs_search	github
	suggest	twitter
	cve_search	bing
	dark_web_crawler	
	domain_reputation	
	tweet_search	

# Utilizzo di Crawler

## OWASP Maryam – Modulo dnsbrute

- Supponiamo di voler utilizzare il modulo **dnsbrute** appartenente alla categoria *Footprint*

- Per conoscerne la sua sintassi, è sufficiente digitare il nome di tale modulo

```
[maryam][default] > dnsbrute
usage: dnsbrute [-h] -d DOMAIN [-c COUNT] [-w WORDLIST] [-t THREAD] [-l]
                 [-i] [--output] [--api] [--format]

dnsbrute 1.1(Saeeddqn) - description: DNS brute force attack, supports
concurrency.

options:
  -h, --help            show this help message and exit
  -d, --domain DOMAIN  Domain name without https://
  -c, --count COUNT    Number of payloads len(max=count of payloads).
                        default is max
  -w, --wordlist WORDLIST
                        wordlist address. default is dnsnames.txt in data
                        folder
  -t, --thread THREAD  The number of links that open per round(default=8)
  -l, --wordlists       List of most common DNS wordlists
  -i, --ips             Show ip addresses
  --output, --output     Save the output to the workspace
  --api, --api           Show results in the JSON format
  --format, --format     Beautifying JSON output if --api is used
Comments: The wordlist option can be an url
```

# Utilizzo di Crawler

## OWASP Maryam – Modulo dnsbrute

- Supponiamo di voler utilizzare il modulo **dnsbrute** appartenente alla categoria *Footprint*
  - Tramite il parametro **-l** del comando **dnsbrute** possiamo visualizzare le sue *wordlist*

```
[*] common DNS wordlists
+-----+
| | list | scale | +-----+
+-----+
| https://raw.githubusercontent.com/danielmiessler/SecLists/master/Discovery/DNS/subdomains-top1million-5000.txt | small |
| https://raw.githubusercontent.com/danielmiessler/SecLists/master/Discovery/DNS/fierce-hostlist.txt | small |
| https://raw.githubusercontent.com/danielmiessler/SecLists/master/Discovery/DNS/namelist.txt | small |
| https://raw.githubusercontent.com/danielmiessler/SecLists/master/Discovery/DNS/subdomains-top1million-20000.txt | medium |
| https://raw.githubusercontent.com/danielmiessler/SecLists/master/Discovery/DNS/subdomains-top1million-110000.txt | large |
| https://raw.githubusercontent.com/danielmiessler/SecLists/master/Discovery/DNS/deepmagic.com-prefixes-top50000.txt | large |
| https://raw.githubusercontent.com/danielmiessler/SecLists/master/Discovery/DNS/shubs-subdomains.txt | large |
+-----+
```

# Utilizzo di Crawler

## OWASP Maryam – Modulo dnsbrute

- Supponiamo di voler utilizzare il modulo **dnsbrute** appartenente alla categoria *Footprint*
  - Scegliamo di utilizzare la *wordlist subdomains.txt* sul dominio **unisa.it**

```
[maryam][default] > dnsbrute -d unisa.it -w subdomains.txt
```

```
_____  
STARTING DNS BRUTE FORCE WITH 484699 PAYLOAD  
_____
```

```
[*] wiki.unisa.it  
[*] www.unisa.it  
[*] www2.unisa.it  
[*] web.unisa.it  
[*] cs.unisa.it  
[*] www3.unisa.it  
[*] auth.unisa.it  
[*] beta.unisa.it  
[*] www.cs.unisa.it  
[*] www4.unisa.it  
[*] test.unisa.it  
[*] cl.unisa.it
```

# Utilizzo di Crawler

## OWASP Maryam – Modulo dnsbrute

- Supponiamo di voler utilizzare il modulo **dnsbrute** appartenente alla categoria *Footprint*
  - Scegliamo di utilizzare la *wordlist subdomains.txt* sul dominio **unisa.it**

**Output Parziale**

```
[*] ua.apps.unisa.it
[*] natura.di.unisa.it
[*] atip-aiprp.apps.unisa.it
[*] iwad-dc5.apps.unisa.it
[*] besthack.site.unisa.it
[*] omap.prod.unisa.it
[*] icxt.di.unisa.it
[*] sicurezzaonline.di.unisa.it
[*] psico.unisa.it
[*] static.apps.unisa.it
[*] devicenotices.site.unisa.it
[*] reustice.site.unisa.it
[*] info.apps.unisa.it
```

# Utilizzo di Crawler

## OWASP Maryam – Modulo Crawler

- Supponiamo di voler utilizzare il modulo **crawler** appartenente alla categoria *OSINT*
  - Per conoscerne la sintassi, è sufficiente digitare il nome di tale modulo

```
Osint
_____
cve_search
phone_number_search
article_search
github_leaks
docs_search
username_search
social_nets
email_pwned
suggest
famous_person
email_search
reddit_search
onion_search
bing_mobile_view
dns_search
dark_web_crawler
crawler
tweet_search
cloud_storage
domain_reputation
```

Output Parziale

# Utilizzo di Crawler

## OWASP Maryam – Modulo Crawler

- Modulo che esegue la scansione di un dominio per trovare link, file JS, e-mail, file multimediali, username, commenti, contatti telefonici ed altre informazioni di potenziale interesse

- **crawler**

```
[maryam][default] > crawler
usage: crawler [-h] -d DOMAIN [--debug] [-l LIMIT] [-t THREAD] [--output]
                [--api] [--format]

crawler 0.5(Saeed) - description: Crawl web pages to find links, JS Files,
CSS files, Comments and everything else interesting, supports concurrency.

options:
  -h, --help            show this help message and exit
  -d DOMAIN, --domain DOMAIN
                        Domain string
  --debug, --debug      debug the scraper
  -l LIMIT, --limit LIMIT
                        Scraper depth level
  -t THREAD, --thread THREAD
                        The number of links that open per round
  --output, --output    Save the output to the workspace
  --api, --api          Show results in the JSON format
  --format, --format    Beautifying JSON output if --api is used

Examples:
  crawler -d <DOMAIN>
  crawler -d <DOMAIN> -l 10 -t 3 --output --debug
```

# Utilizzo di Crawler

## OWASP Maryam – Modulo Crawler – Esempio

- Modulo che esegue la scansione di un dominio per trovare link, file JS, e-mail, file multimediali, username, commenti, contatti telefonici ed altre informazioni di potenziale interesse

➤ `crawler -d owasp.org -l 4 --output --debug`

Output Parziale

```
[maryam][default] > crawler -d owasp.org -l 4 --output --debug
[*] https://owasp.org/supporters
[*] https://owasp.org/store
[*] https://owasp.org/search
[*] https://owasp.org/www-policy/operational/general-disclaimer.html
[*] https://owasp.org/donate?reponame=owasp.github.io
[*] https://owasp.org/www--site-theme/favicon.ico
[*] https://owasp.org/membership
[*] https://owasp.org/sitemap/
[*] https://owasp.org/blog/2023/03/10/strategic-plan-open-letter-update.html
[*] js: https://owasp.org/www--site-theme/assets/js/yaml.min.js
[*] https://owasp.org
[*] https://owasp.org/chapters/
[*] https://owasp.org/sitemap
[*] https://owasp.org/about/
[*] js: https://owasp.org/www--site-theme/assets/js/kjua.min.js
[*] https://owasp.org/slack/invite
[*] https://owasp.org/contact/
[*] https://owasp.org/donate
[*] https://owasp.org/events/
```

# Utilizzo di Crawler

## OWASP Maryam – Modulo Crawler – Esempio

- Modulo che esegue la scansione di un dominio per trovare link, file JS, e-mail, file multimediali, username, commenti, contatti telefonici ed altre informazioni di potenziale interesse

➤ `crawler -d owasp.org -l 4 --output --debug`

Output Parziale

```
[*] js(32)
[*]   https://owasp.org/www--site-theme/assets/js/yaml.min.js
[*]   https://owasp.org/www--site-theme/assets/js/kjua.min.js
[*]   https://owasp.org/www--site-theme/assets/js/jquery-3.4.1.min.js
[*]   https://owasp.org/www--site-theme/assets/js/luxon.min.js
[*]   https://owasp.org/www--site-theme/assets/js/js.cookie.js
[*]   https://owasp.org/www--site-theme/assets/js/util.js
[*]   https://owasp.org/cdn-cgi/scripts/5c5dd728/cloudflare-static/email-de
code.min.js
[*]   https://owasp.org/www-project-node.js-goat/
[*]   https://owasp.org/assets/js/page--src--pages--index-vue.9046d380.js
[*]   https://owasp.org/assets/js/page--src--templates--markdown-page-vue.a
2e4fad8.js
[*]   https://owasp.org/assets/js/app.882b7d80.js
[*]   https://owasp.org/assets/js/search.46db29eb.js
[*]   https://owasp.org/assets/js/page--src--pages--bom-maturity-model-vue.
dcc7a792.js
```

**File JavaScript (js)**

# Utilizzo di Crawler

## OWASP Maryam – Modulo Crawler – Esempio

- Modulo che esegue la scansione di un dominio per trovare link, file JS, e-mail, file multimediali, username, commenti, contatti telefonici ed altre informazioni di potenziale interesse

➤ `crawler -d owasp.org -l 4 --output --debug`

Output Parziale

```
[*] cdn(33)
[*] //www.google-analytics.com/analytics.js
[*] //www.mediawiki.org/
[*] //www.mediawiki.org/wiki/Special:MyLanguage/Help:Categories
[*] //tools.ietf.org/html/rfc4998
[*] //tools.ietf.org/html/rfc6283
[*] //www.youtube.com/embed/CDbWvEwBBxo?
[*] //www.youtube.com/embed/pypTPaU7mM?
[*] //www.youtube.com/embed/zEV3HOuM_Vw?
[*] //www.youtube.com/embed/_Z9RQSnf8-g?
[*] //http://www.rafael.co.il/
[*] //www.youtube.com/embed/videoseries?list=PLiooNakZQW8qeeXxYp0tRBL3EtjyJnm3L&
[*] //tools.ietf.org/html/rfc2616
[*] //tools.ietf.org/html/rfc6797
[*] //www.youtube.com/embed/videoseries?list=PLpr-xdpM8wG-ma2GOBmdpGGfnVPVwFFQd&
[*] //www.youtube.com/embed/videoseries?list=PLpr-xdpM8wG8jz9QpzQeLeB0914Ysq-Cl&
```

**Content Delivery Network (cdn)**

# Utilizzo di Crawler

## OWASP Maryam – Modulo Crawler – Esempio

- Modulo che esegue la scansione di un dominio per trovare link, file JS, e-mail, file multimediali, username, commenti, contatti telefonici ed altre informazioni di potenziale interesse

➤ `crawler -d owasp.org -l 4 --output --debug`

Output Parziale

```
[*] getlinks(3993)
[*]   https://owasp.org/donate?reponame=owasp.github.io
[*]   https://owasp.org/donate?reponame=www-policy
[*]   https://owasp.org/donate?reponame=www-chapter-central-university-of-r
ajasthan&title=OWASP+Central+University+of+Rajasthan+-+Student+Chapter
[*]   https://owasp.org/donate/?reponame=www-project-purpleteam&title=O
WASP+purpleteam
[*]   https://owasp.org/donate?reponame=www-project-purpleteam&title=OWASP+
PurpleTeam
[*]   https://owasp.org/donate?reponame=www-chapter-lovely-professional-u
niversity&title=OWASP+Lovely+Professional+University+-+Student+Chapter
[*]   https://owasp.org/donate?reponame=www-project-domain-protect&title=O
WASP+Domain+Protect
[*]   https://owasp.org/donate?reponame=www-project-go-secure-coding-practi
ces-guide&title=OWASP+Go+Secure+Coding+Practices+Guide
[*]   https://owasp.org/donate?reponame=www-project-data-security-top-10&t
itle=OWASP+Data+Security+Top+10
[*]   https://owasp.org/donate?reponame=www-project-top-10-ci-cd-security-r
isks&title=OWASP+Top+10+CI%2FCD+Security+Risks
```

**getlinks**

# Utilizzo di Crawler

## OWASP Maryam – Modulo Crawler – Esempio

- Modulo che esegue la scansione di un dominio per trovare link, file JS, e-mail, file multimediali, username, commenti, contatti telefonici ed altre informazioni di potenziale interesse

➤ `crawler -d owasp.org -l 4 --output --debug`

Output Parziale

```
[*] comments(0)
[*]
[*] emails(991)
[*]     recaptcha@1.3.0
[*]     miya@owasp.org.cn
[*]     barbosa@owasp.org
[*]     flores@owasp.org
[*]     purecss@2.0.5
[*]     chapter@owasp.org
[*]     seedorff@owasp.org
[*]     haddix@owasp.org
[*]     meissler@owasp.org
[*]     carter@owasp.org
[*]     milan@owasp.org
[*]     butler@owasp.org
[*]     paco@owasp.org
[*]     pannell@owasp.org
```

Comments ed e-mail

# Utilizzo di Crawler

## OWASP Maryam – Modulo Crawler – Esempio

- Modulo che esegue la scansione di un dominio per trovare link, file JS, e-mail, file multimediali, username, commenti, contatti telefonici ed altre informazioni di potenziale interesse

➤ `crawler -d owasp.org -l 4 --output --debug`

Output Parziale

```
[*] phones(0)
[*]
[*] ..
[*] media(4948)
[*]     https://owasp.org/assets/images/AppSec_DC_2023_Banner_1200x300_V01.jpg
eg
[*]     https://owasp.org/assets/sitedata/banner-data.yml
[*]     https://owasp.org/assets/images/logo.png
[*]     https://owasp.org/assets/sitedata/popup-data.yml
[*]     https://owasp.org/assets/images/content/featured_project_aiseccpriv.jpg
g
[*]     https://owasp.org/assets/sitedata/corp_members.yml
[*]     https://owasp.org/assets/images/people/staff_andrew.jpg
[*]     https://owasp.org/assets/images/web/global-conference.png
[*]     https://owasp.org/assets/sitedata/events.yml
[*]     https://owasp.org/assets/images/people/board-grant.png
[*]     https://owasp.org/assets/images/people/leader_springett.png
[*]     https://owasp.org/assets/images/web//members-header.png
[*]     https://owasp.org/assets/images/web/chaper-wide.jpg
```

**Phones e media**

# Utilizzo di Crawler

## OWASP Maryam – Modulo Crawler – Esempio

- Modulo che esegue la scansione di un dominio per trovare link, file JS, e-mail, file multimediali, username, commenti, contatti telefonici ed altre informazioni di potenziale interesse

➤ `crawler -d owasp.org -l 4 --output --debug`

Output Parziale

```
[*] usernames(15)
[*]     Instagram
[*]         instagram.com/owasp_lpu
[*]         instagram.com/owasp_vadodara
[*]         instagram.com/owaspbh
[*]         instagram.com/owasp_nitr
[*]         instagram.com/carlos.crowsec
[*]         instagram.com/owasp_jp
[*]         instagram.com/owasp.mec
[*]         instagram.com/owasp.gauhati.university
[*]         instagram.com/owasp_bhubaneswar
[*]         instagram.com/owaspvitbhopal
[*]         instagram.com/owasp_bsacist
[*]         instagram.com/owasp_vellore
[*]         instagram.com/owaspindore
```

**Username Instagram**

# Utilizzo di Crawler

## OWASP Maryam – Modulo Crawler – Esempio

- Modulo che esegue la scansione di un dominio per trovare link, file JS, e-mail, file multimediali, username, commenti, contatti telefonici ed altre informazioni di potenziale interesse

➤ `crawler -d owasp.org -l 4 --output --debug`

Output Parziale

```
[*] Facebook
[*]   facebook.com/OWASPFoundation
[*]   facebook.com/csp
[*]   facebook.com/OWASPReading
[*]   facebook.com/OWASPTGU
[*]   facebook.com/OwaspCusco
[*]   facebook.com/owaspireland
[*]   facebook.com/OWASPIreland
[*]   facebook.com/OWASPSanJac
[*]   facebook.com/owasp
[*]   facebook.com/OwaspHongKongChapter
[*]   facebook.com/owaspBristol
[*]   facebook.com/OWASPBristolChapter
[*]   facebook.com/groups
[*]   facebook.com/owaspid
```

**Username Facebook**

# Utilizzo di Crawler

## OWASP Maryam – Modulo Crawler – Esempio

- Modulo che esegue la scansione di un dominio per trovare link, file JS, e-mail, file multimediali, username, commenti, contatti telefonici ed altre informazioni di potenziale interesse

➤ `crawler -d owasp.org -l 4 --output --debug`

Output Parziale

```
[*]          Twitter
[*]          twitter.com/owasp
[*]          twitter.com/OWASPPurpleTeam
[*]          twitter.com/widgets.js
[*]          twitter.com/owasp_lpu
[*]          twitter.com/domain_protect
[*]          twitter.com/OvidiuCical
[*]          twitter.com/Dkrivelev
[*]          twitter.com/omer_gil
[*]          twitter.com/iiamit
[*]          twitter.com/claudijd
[*]          twitter.com/_mwc
[*]          twitter.com/travismcpeak
[*]          twitter.com/tysbano
[*]          twitter.com/astha_singhal
[*]          twitter.com/rung
```

**Username Twitter**

# Utilizzo di Crawler

## OWASP Maryam – Modulo Crawler – Esempio

- Modulo che esegue la scansione di un dominio per trovare link, file JS, e-mail, file multimediali, username, commenti, contatti telefonici ed altre informazioni di potenziale interesse
  - `crawler -d owasp.org -l 4 --output --debug`

Output Parziale

```
[*]      Github
[*]      github.com/OWASP
[*]      github.com/owasp
[*]      github.com/owasp-change
[*]      github.com/CycloneDX
[*]      github.com/purpleteam-labs
[*]      github.com/binarymist
[*]      github.com/domain-protect
[*]      github.com/cider-security-research
[*]      github.com/oshp
[*]      github.com/ovh
[*]      github.com/orgs
[*]      github.com/adamaveray
[*]      github.com/twitter
[*]      github.com/w3c
[*]      github.com/xsleaks
[*]      github.com/zaproxy
[*]      github.com/riramar
[*]      github.com/rfc-st
```

**Username Github**

# Utilizzo di Crawler

## OWASP Maryam – Modulo Crawler – Esempio

- Modulo che esegue la scansione di un dominio per trovare link, file JS, e-mail, file multimediali, username, commenti, contatti telefonici ed altre informazioni di potenziale interesse
  - `crawler -d owasp.org -l 4 --output --debug`

Output Parziale

```
[*]      Github site
[*]      owasp.github.io
[*]      buttons.github.io
[*]      owasp-change.github.io
[*]      w3c.github.io
[*]      stedolan.github.io
[*]      api.github.com
[*]      gist.github.com
[*]      uploads.github.com
[*]      translator.github.com
[*]      alive.github.com
[*]      identicons.github.com
[*]      customer-stories-feed.github.com
[*]      spotlights-feed.github.com
[*]      docs.github.com
[*]      nets4geeks.github.io
[*]      owtf.github.io
[*]      securityrat.github.io
[*]      webgoat.github.io
[*]      bbva.github.io
```

**Github site**

# Utilizzo di Crawler

## OWASP Maryam – Modulo Crawler – Esempio

- Modulo che esegue la scansione di un dominio per trovare link, file JS, e-mail, file multimediali, username, commenti, contatti telefonici ed altre informazioni di potenziale interesse

➤ `crawler -d owasp.org -l 4 --output --debug`

Output Parziale

```
[*]     Telegram
[*]         telegram.me/secbsb
[*]     Youtube user
[*]         youtube.com/user/OWASPGLOBAL
[*]         youtube.com/user/owtfproject
[*]         youtube.com/user/GhazalGah
[*]         youtube.com/user/webpwnized
[*]         youtube.com/user/owaspomaha
[*]         youtube.com/user/owaspgbg
[*]     Youtube channel
[*]         youtube.com/channel/UCg70aYUEoBV_pn-skdFcA-Q
[*]         youtube.com/channel/UCbOuqKOuGOLLpENey0TTYqQ
[*]         youtube.com/channel/UC5hhduxXVgrk-cudJlUxKTA
[*]         youtube.com/channel/UCFhikRaW1zu5wkKqqqaIe8Q
[*]         youtube.com/c/OWASPStockholm
[*]         youtube.com/channel/UCitrDIOsVjayy6GrQ2LuzKA
[*]         youtube.com/c/OWASP_DevSlop
[*]         youtube.com/channel/UCDwRks28thuvwICPM5VgmSQ
```

**Telegram e YouTube**

# Utilizzo di Crawler

## OWASP Maryam – Modulo Crawler – Esempio

- Modulo che esegue la scansione di un dominio per trovare link, file JS, e-mail, file multimediali, username, commenti, contatti telefonici ed altre informazioni di potenziale interesse

➤ `crawler -d owasp.org -l 4 --output --debug`

Output Parziale

```
[*]     Linkedin company
[*]         linkedin.com/company/owasp
[*]         linkedin.com/company/owasp-lpu
[*]         linkedin.com/company/raspina-net-pars
[*]         linkedin.com/company/owasp-nit-rourkela
[*]         linkedin.com/company/owasp-mahendra-engineering-college
[*]         linkedin.com/company/keyes-security
[*]         linkedin.com/company/owasphonduras
[*]         linkedin.com/company/owasp-gauhati-university
[*]         linkedin.com/company/owasp-devslop
[*]         linkedin.com/company/owaspkusco
[*]         linkedin.com/company/owasp-orange-county
[*]         linkedin.com/company/owasp-vit-bhopal-university
[*]         linkedin.com/company/owasp-sofia
[*]         linkedin.com/company/owasp-muscat
[*]         linkedin.com/company/owasp-bsacist
[*]         linkedin.com/company/owasp-baku
[*]         linkedin.com/company/owasp-lucknow-chapter
[*]         linkedin.com/company/owasp-maine
[*]         linkedin.com/company/owaspiitd
```

**LinkedIn company**

# Utilizzo di Crawler

## OWASP Maryam – Modulo Crawler – Esempio

- Modulo che esegue la scansione di un dominio per trovare link, file JS, e-mail, file multimediali, username, commenti, contatti telefonici ed altre informazioni di potenziale interesse
  - `crawler -d owasp.org -l 4 --output --debug`

Output Parziale

```
[*]      Linkedin individual
[*]          linkedin.com/in/jmanico
[*]          linkedin.com/in/purabparihar
[*]          linkedin.com/in/balalikhithkanigolla
[*]          linkedin.com/in/ovidiuclical
[*]          linkedin.com/in/daniel-krivelevich
[*]          linkedin.com/in/omer-gil
[*]          linkedin.com/in/alireza-mostame-29970b242
[*]          linkedin.com/in/rezataba
[*]          linkedin.com/in/amirmahdi-nowbakht-3b8865200
[*]          linkedin.com/in/raphael-hagi
[*]          linkedin.com/in/eduardo-bellis-92482534
[*]          linkedin.com/in/bbarbosa85
[*]          linkedin.com/in/devpauloasilva
[*]          linkedin.com/in/rspro
[*]          linkedin.com/in/michaelbargury
```

**Linkedin individual**

# Utilizzo di Crawler

## OWASP Maryam – Modulo Crawler – Esempio

- Modulo che esegue la scansione di un dominio per trovare link, file JS, e-mail, file multimediali, username, commenti, contatti telefonici ed altre informazioni di potenziale interesse

➤ `crawler -d owasp.org -l 4 --output --debug`

Output Parziale

```
[*] Googleplus
[*] plus.google.com/communities
[*] plus.google.com/112137101792593443873
[*] plus.google.com/104775568539346911434
[*] plus.google.com/events
[*] plus.google.com/+Pawe
[*] plus.google.com/+Bj
[*] WordPress
[*] files.wordpress.com
[*] redblueteam.wordpress.com
[*] standbywordpress.wordpress.com
[*] securitythoughts.wordpress.com
[*] teom.wordpress.com
[*] support.wordpress.com
[*] gaurangkp.wordpress.com
[*] owasporizon.wordpress.com
[*] globalprojectscommittee.wordpress.com
[*] cagataycivici.wordpress.com
```

**Googleplus e WordPress**

# Utilizzo di Crawler

## OWASP Maryam – Modulo Crawler – Esempio

- Modulo che esegue la scansione di un dominio per trovare link, file JS, e-mail, file multimediali, username, commenti, contatti telefonici ed altre informazioni di potenziale interesse

➤ `crawler -d owasp.org -l 4 --output --debug`

Output Parziale

```
[*]      Blogger
[*]      owasp.blogspot.com
[*]      sharingsec.blogspot.com
[*]      sectooladdict.blogspot.com
[*]      respectxss.blogspot.com
[*]      hussaina-begum.blogspot.com
[*]      abhisharma404.blogspot.com
[*]      dummy2dummies.blogspot.com
[*]      sg6-labs.blogspot.com
[*]      myappsecurity.blogspot.com
[*]      michael-coates.blogspot.com
[*]      jeremiahgrossman.blogspot.com
[*]      homakov.blogspot.com
[*]      tacticalwebappsec.blogspot.com
[*]      seguridad-agile.blogspot.com
[*]      googleprojectzero.blogspot.com
```

Blogger