

Networking

G.Filosofi 2018

- Blockchain: a network distributed database of peer-to-peer transactions (of any kind, e.g. a payment, a google search, voting, etc.) where all the participants share a consensus. Any new transaction added at any point of the network gets spreaded over the entire network. It cannot be removed or changed anymore.
- SEO (Search Engine Optimisation): the art of making your web content more relevant in google search. An important factor is the number of input links, i.e. links to your web pages found in other web sites
- Telnet: client-server protocol used for bidirectional interactive text-oriented communication. For secure use SSH is preferred. Every time users 'hit' a key a packet is sent. Users authenticate with a clear-text username and password. Exposed on TCP port ????.
`host $ telnet <ipaddr>`
- FTP (File Transfer Protocol): client-server protocol used for file transfer. Users authenticate with a clear-text username and password. For secure transmission FTP is often secured with SSL/TLS (FTPS) or replaced with SSH File Transfer Protocol (SFTP)
- SSH (Secure Shell): client-server remote terminal for operating network services securely over an unsecured network. The best known example application is for access to computer systems by remote users. Username and password are sent encrypted over the network. SSH is a recommended alternative to Telnet because it is encrypted. However, it is susceptible to sniffing using MITM techniques
- SMTP (Simplified Mail Transport Protocol): the Internet standard for email. The secure version is S/SMTP. SMTP is a clear-text protocol. In addition to that, emails inherently do not have any authentication mechanism. In other words, the protocol does not provide a secure way of authenticating whether a received email is indeed from the listed sender
- The Domain Name System (DNS) is a translation service that translates hostnames to IP addresses
- The Pup (PARC Universal Packet) format created for Alto (1973) influenced the TCP/IP protocol suite development
- IPv4 sono 32 bit, IPv6 sono 128 bit. Questo significa tantissimi ip address per ogni metro quadro sulla terra. Oggi il traffico IPv6 è al 14%, in crescita. La latenza di IPv6 è inferiore a IPv4
- url: ns2.ntc.com
- IPv4: 205.251.198.137
- IPv6: 2600:9000:5306:8900:0:0:0:1
- IPv6: an IP address consisting of 8 groups of 16-bit hexadecimal numbers (for a total of 128 bits). If several groups in a row are all zero, you can omit those groups and replace them with a double colon (but only once per IP address). For example, the IPv6 address for example.com is 2001:500:88:200::10.
- pfSense: piccola distribuzione FreeBSD con un sw che funge da router, firewall e proxy. Si può creare una VPN con la quale ci si può collegare alla rete domestica dall'esterno, si può creare una rete ospiti nella rete domestica, impostare traffic shaping, fare logging, monitorare l'utilizzo di banda dei vari device connessi, ecc.
- ntop: comando di sistema per avere statistiche di rete
- Tunneling SSH: Una volta stabilita una connessione sicura SSH con un server remoto, si può stabilire un tunnel per accedere ad alcune risorse presenti nella rete locale del server remoto ma che non hanno indirizzo pubblico
- VPN (Virtual Private Network): it is the client-server protocol of choice to establish a secure connection over the internet
- Switch: Layer 2 device that routes MAC addresses
- Router: Layer 3 device that routes IP addresses
- Proxy: Layer 4 server that performs access to web sites on clients behalf. A client makes

a query to the Proxy he want to visit a specific URL. The Proxy establishes a brand new connection toward that URL. This will guarantee anonymization for the client. Moreover the Proxy caches the web pages, which will reduce internet traffic for repeated browsing sessions.

- WAN: è la porta di connessione alla internet pubblica
- RAID: *Redundant Array of Indipendent Disks*
- NAS: Network Attached Storage
- NAT/PAT: Network/Port Address Translation. The majority of NATs map multiple private hosts to one publicly exposed IP address. In a typical configuration, a local network uses one of the designated IP subnets. A router on that network has a private address in that address space and a "public" WAN address assigned by an ISP. As traffic passes from the local network to the Internet, all the source private addresses in each packet are translated on the fly by the router to its public address. For different source addresses the router overwrites different source ports. The router tracks basic data about each active connection. When a reply returns to the router, it uses the connection tracking data it stored during the outbound phase (source address and port) to determine the private address and port on the internal network to which to forward the reply. NAT/PAT are considered Level 3 protocols
- Firewall: a wall of fire that protects a local network form incoming connections
- If you go to the WhatIsMyIPAddress.com home page, you can see your public IP address. Now is 87.18.216.169
- Wireshark is a powerful free packet sniffer for monitoring and analyzing network traffic
- nettop is a shell command to monitor network activity
- netsynt: a shell command to ..
- TTL (Time To Live): numero massimo di hops di un pacchetto IP prima che venga eliminato
- 3-way Handshake: the TCP message exchange to initiate any TCP connection (SSH, Telnet, FTP, HTTP,..). It is what qualifies TCP a "connection oriented" or "reliable" protocol

1. Client -> SYN -> Server
server is open)
2. Client <- SYN+ACK <- Server
3. Client -> ACK -> Server

(with this message the client ask/infer if the

SYN and ACK are bit of TCP header

The same sequence is repeated when the connection is closing

Source	Destination	Protocol	Length	Info
192.168.0.37	192.168.0.142	TCP	66	59580 - 22 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
192.168.0.142	192.168.0.37	TCP	66	59580 - [SYN, ACK] Seq=1 Win=29200 Len=0 MSS=1460 SACK_PERM=1 WS=128
192.168.0.37	192.168.0.142	TCP	60	59580 - [ACK] Seq=1 Ack=1 Win=2102272 Len=0
192.168.0.142	192.168.0.37	SSHv2	95	Server: Protocol (SSH-2.0-OpenSSH_7.6p1 Ubuntu-4ubuntu0.3)
192.168.0.37	192.168.0.142	TCP	60	59580 - 22 [ACK] Seq=1 Ack=42 Win=2102272 Len=0
192.168.0.37	192.168.0.142	TCP	88	Client: Protocol (SSH-2.0-TSSSH/2.69 Win32)
192.168.0.142	192.168.0.37	TCP	54	22 - 59580 [ACK] Seq=42 Ack=27 Win=29312 Len=0
192.168.0.142	192.168.0.37	SSHv2	1134	Server: Key Exchange Init
192.168.0.37	192.168.0.142	TCP	68	59580 - 22 [ACK] Seq=27 Ack=1122 Win=2101248 Len=8
192.168.0.37	192.168.0.142	SSHv2	1079	Client: Key Exchange Init
192.168.0.142	192.168.0.37	TCP	54	22 - 59580 [ACK] Seq=1043 Ack=1043 Win=32128 Len=8
192.168.0.37	192.168.0.142	SSHv2	134	Client: Elliptic Curve Diffie-Hellman Key Exchange Init
192.168.0.142	192.168.0.37	TCP	54	22 - 59580 [ACK] Seq=1122 Ack=1123 Win=32128 Len=8
192.168.0.142	192.168.0.37	SSHv2	366	Server: Elliptic Curve Diffie-Hellman Key Exchange Reply, New Keys
192.168.0.37	192.168.0.142	SSHv2	78	Client: New Keys
192.168.0.142	192.168.0.37	TCP	54	22 - 59580 [ACK] Seq=1434 Ack=1139 Win=32128 Len=8
192.168.0.37	192.168.0.142	SSHv2	150	Client: Encrypted packet (len=96)
192.168.0.142	192.168.0.37	TCP	54	22 - 59580 [ACK] Seq=1434 Ack=1235 Win=32128 Len=8
192.168.0.37	192.168.0.142	SSHv2	159	Server: Encrypted packet (len=96)
192.168.0.142	192.168.0.37	TCP	54	22 - 59580 [ACK] Seq=1434 Ack=1235 Win=32128 Len=8
192.168.0.37	192.168.0.142	SSHv2	166	Client: Encrypted packet (len=112)
192.168.0.142	192.168.0.37	SSHv2	166	Server: Encrypted packet (len=112)
192.168.0.37	192.168.0.142	SSHv2	182	Client: Encrypted packet (len=128)
192.168.0.142	192.168.0.37	SSHv2	134	Server: Encrypted packet (len=80)
Padding Length: 7				
▼ Key Exchange				
Message Code: Key Exchange Init (20)				
▶ Algorithms				
Padding String: e1030f3851f2d6				
0030	29 10 0b f1 08 08 00 00 83 f4 07 14 bb 24 bb 5c		 \$ \n
0040	21 5a 38 67 06 c9 c1 11 34 c7 d3 a9 08 00 00 b7			!28g- . . 4-
0050	65 63 64 68 2d 73 68 61 32 2d 66 69 73 74 78 32			ecdh-sha_2-nistp2
0060	35 36 2c 65 63 64 68 2d 73 68 61 32 2d 66 69 73			56, ecdh- sha2-nis
0070	74 70 33 38 34 2c 65 63 64 68 2d 73 68 61 32 2d			tp384_ec_dh-sha2-
0080	66 69 73 74 70 35 32 31 2c 64 69 66 66 65 65 60			nistp521_.diffie-
0090	68 65 6c 6c 6d 61 6e 2d 67 72 6f 75 78 2d 65 78			hellman_group-ex
00a0	63 68 61 66 67 65 2d 73 68 61 32 35 36 2c 64 69			change-s_h256_di
00b0	66 66 69 65 2d 68 65 6c 6c 6d 61 6e 2d 67 72 6f			ffie-hel_1man-gro
00c0	75 70 2d 65 78 63 68 61 6e 67 65 2d 73 68 61 31			up-excha_nge-sha1
00d0	2c 64 69 66 66 69 65 2d 68 65 6c 6c 6d 61 6e 2d			,diffie_ hellman-
00e0	67 72 6f 75 70 31 34 2d 73 68 61 31 2c 64 69 65			group14_ sha1,dif
00f0	66 69 65 2d 68 65 6c 6c 6d 61 6e 2d 67 72 6f 75			fie-hell man-grou
0100	70 31 2d 73 68 61 31 00 00 00 57 65 63 64 73 61			p1-sha1_ -Wedcsa
0110	2d 73 68 61 32 2d 66 69 73 74 70 32 35 36 2c 65			-sha2-ni stp256,e
0120	63 64 73 61 2d 73 68 61 32 2d 6e 69 73 74 70 33			crsa-sha_2-nistp3

How To

- To inspect incoming network connections
\$ w
- To change the port of SSH from 22 to 2233
\$ sudo vim /etc/ssh/sshd_config
\$ sudo service ssh restart
- To configure static IP and routes
Consider two networks connected through the Internet

Pentester deve consegnare a Backbone tutti i pacchetti non destinati alla sottorete di appartenenza

Pentester \$ ifconfig eth0 210.100.1.1 netmask 255.255.255.0 up

Pentester \$ route add default gw 210.100.1.2

Backbone ha due interfacce di rete

Backbone \$ ifconfig eth0 210.100.1.2 netmask 255.255.255.0 up

Backbone \$ ifconfig eth1 211.100.1.1 netmask 255.255.255.0 up

Quello che è destinato alla sottorete 212.100.1.x deve essere inoltrato a Router

Backbone \$ sysctl -w net.ipv4.ip forward=1

Backbone \$ route add -net 212.100.1.0/24 gw 211.100.1.2

Router ha due interfacce di rete

Router \$ ifconfig eth0 212.100.1.1 netmask 255.255.255.0 up

Router \$ ifconfig eth1 211.100.1.2 netmask 255.255.255.0 up

Quello che è destinato alla sottorete 210.100.1.x deve essere inoltrato a Backbone

Router \$ sysctl -w net.ipv4.ip forward=1

Router \$ route add -net 210.100.1.0/24 gw 211.100.1.1

Server Web deve consegnare a Router tutti i pacchetti non destinati alla sottorete di appartenenza

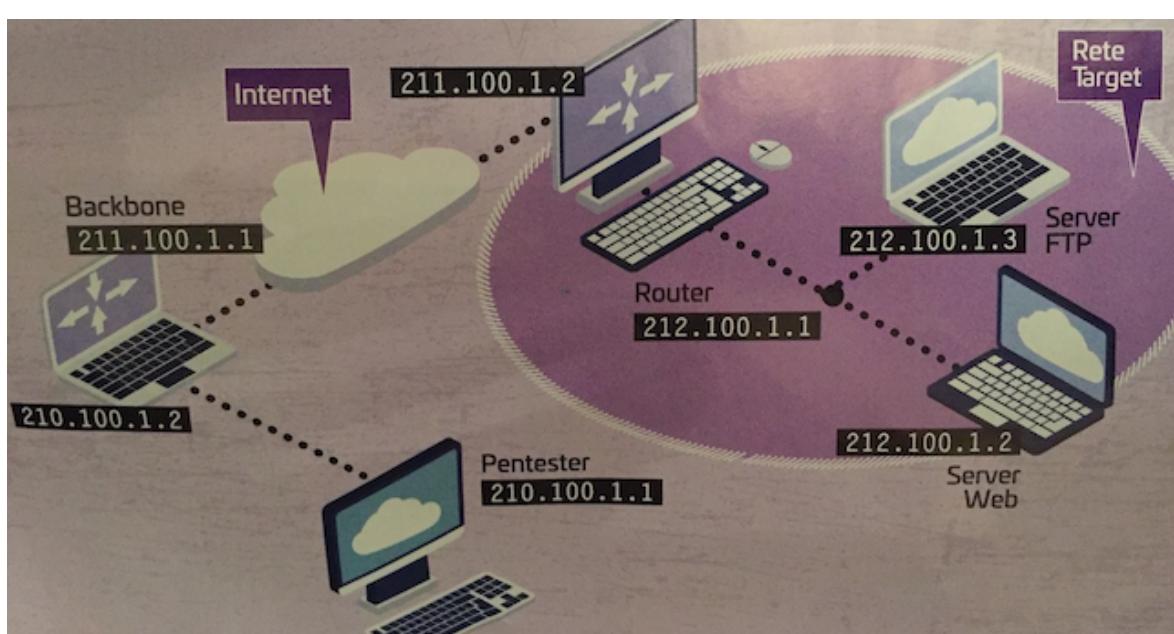
ServerWeb \$ ifconfig eth0 212.100.1.2 netmask 255.255.255.0 up

ServerWeb \$ route add default gw 212.100.1.1

Server FTP deve consegnare a Router tutti i pacchetti non destinati alla sottorete di appartenenza

ServerFTP \$ ifconfig eth0 212.100.1.3 netmask 255.255.255.0 up

ServerFTP \$ route add default gw 212.100.1.1



nc

A command to open TCP/UDP connections and listen. It works with IPv4/6

- Example: using nmap over 8.8.8.8 host you find out there an open port 53/tcp open tcpwrapped

Then try to connect to the port
client \$ nc -v 8.8.8.8 53

```
[gfmacbook:myNotes gabrielefilosofi$ nc -v 8.8.8.8 53
found 0 associations
found 1 connections:
  1: flags=82<CONNECTED,PREFERRED>
    outif en0
    src 192.168.1.57 port 51634
    dst 8.8.8.8 port 53
    rank info not available
    TCP aux info available
```

Connection to 8.8.8.8 port 53 [tcp/domain] succeeded!

-v : shows not only the open ports, but also the ports that were checked and found to not be open

- With -e option you can make any process a network server

Example: on host computer with IP add <hostip>

```
host $ nc -l -p 1234 -e /bin/sh
```

Then from remote client connect to the host

```
client $ nc <hostip> 1234
```

- With -u option you can work with UDP

```
host $ nc -l -u -p 1234
```

```
client $ nc -u <hostip> 1234
```

hping2/3

A command to send custom TCP packets and display replies

- To send <n> TCP messages to a server <ipaddr> port <port>

```
host $ hping3 -S -p <port> -c <n> <ipaddr>
```

-S only the SYN flag is active

-SA both SYN and ACK flags active

-A only the ACK flag is active

-t <m> sets TTL to <m> 1, quindi dopo il primo hop il messaggio viene scartato

-T invia pacchetti con TTL crescenti

-tr-stop ferma l'invio dei pacchetti alla ricezione di una risposta

- \$ hping3 -S -p <port> -c <n> <serverip>. se TTL 64 allora é Unix/Linux, se 128 Win

- To install on macOS

```
host $ brew install hping
```

Add /usr/local/sbin/ to PATH

```
host $ export PATH="/usr/local/sbin:$PATH"
```

tcpdump

A monitor for TCP/IP traffic

- To grab incoming ping

```
host $ tcpdump -i en0 icmp
```

nmap

A network scanning tool and created in 1997 by *Gordon Lyon* (aka Fyodor)

- To install on macOS

```
host $ brew install nmap
```

- Use

- \$ nmap [scan type] [options] [target]
- Examples


```
$ sudo nmap -sS -Pn -sV <ip>
```

 Some Options:
 - sS SYN scan
 - sV enables services versions detection using banner grabbing
 - Pn bypasses ping blocking
 - O enables OS detection
 - A enables OS and service version detection
 - p 1,80 check ports 1 and 80
- To display also SSL/TLS certificates and RSA public keys


```
$ sudo nmap -n -vv -A <ip> --min-parallelism=50 --max-parallelism=150 -PN -T2 -oA <ip>
```
- To scan ports from 1 to 1000


```
$ sudo nmap -A <ip>
```
- To scan protocol versions active on UDP ports from 1 to 65535 of any target IP 192.168.0.1 to 192.168.0.3


```
$ nmap -n -sU -sV 192.168.0.1-3 -p 1-65535
```
- To search for directory structures and files on a http server


```
$ nmap -sV -p 80 <serverip> --script=http-enum
```

 Example:
 On target host run a HTTP server

```
$ python -m SimpleHTTPServer 9000
```

 Serving HTTP on 0.0.0.0 port 9000 ...
 From another host, run nmap

dirb

```
$ dirb http://<serverweb>
$ dirb http://<serverweb> <filelist>
```

Example:
 Activate Apache2 web server on Kali Linux
 kali \$ /etc/init.d/apache2 start
 kali \$ dirb http://<serverweb>

```
root@kali:~# dirb http://192.168.0.14

-----
DIRB v2.22
By The Dark Raver
-----

START TIME: Wed Jul 31 15:06:33 2019
URL BASE: http://192.168.0.14/
WORDLIST FILES: /usr/share/dirb/wordlists/common.txt
-----


GENERATED WORDS: 4612
---- Scanning URL: http://192.168.0.14/ ----
+ http://192.168.0.14/index.html (CODE:200|SIZE:10707)
=> DIRECTORY: http://192.168.0.14/javascript/
+ http://192.168.0.14/server-status (CODE:200|SIZE:5038)

---- Entering directory: http://192.168.0.14/javascript/ ----
=> DIRECTORY: http://192.168.0.14/javascript/jquery/

---- Entering directory: http://192.168.0.14/javascript/jquery/ ----
+ http://192.168.0.14/javascript/jquery/jquery (CODE:200|SIZE:271809)
-----


END TIME: Wed Jul 31 15:06:36 2019
DOWNLOADED: 13836 - FOUND: 3
```

From remote station download identified file javascript/jquery/jquery
 host \$ curl -O <http://192.168.0.14/javascript/jquery/jquery>

hacking SmartTV

Let's make some practice with the Smart TV

Name: LGwebOSTV

Model: LG 55UH850V

OS: webOS 3.0

<https://www.displayspecifications.com/en/model/a38c46c>

```
host $ sudo nmap -sS -Pn -sV <LGsmartTV_ipaddr>
      3000/tcp open  http  LG smart TV http service
      3001/tcp open  ssl/http LG smart TV http service
      9998/tcp open  http  Google Chromecast httpd
```

We want to interact with HTTP server

```
host $ printf "GET / HTTP/1.0|r|n|r|n" | nc -n -i 1 192.168.1.5 3000
```

```
gfmacbook:myNotes gabrielefilosofi$ printf "GET / HTTP/1.0\r\n\r\n" | nc 192.168.1.5 3000
HTTP/1.1 200 OK
Date: Tue, 23 Jul 2019 22:00:05 GMT
Connection: close
Hello world
```

```
host $ curl -v http://192.168.1.5:3000
```

where -v is the verbose option

```
gfmacbook:Documents gabrielefilosofi$ curl -v http://192.168.1.5:3000
* Rebuilt URL to: http://192.168.1.5:3000/
* Trying 192.168.1.5... w
* TCP_NODELAY set
* Connected to 192.168.1.5 (192.168.1.5) port 3000 (#0)
> GET / HTTP/1.1 WHAT
> Host: 192.168.1.5:3000
> User-Agent: curl/7.54.0 bash
> Accept: */*
< HTTP/1.1 200 OK
< Date: Tue, 23 Jul 2019 22:50:37 GMT
< Connection: keep-alive
< Transfer-Encoding: chunked
<
Hello world
* Connection #0 to host 192.168.1.5 left intact
```

```
host $ printf "GET / HTTP/1.0|r|n|r|n" | nc -n -i 1 192.168.1.5 9998
```

```

gfmacbook:myNotes gabrielefilosofi$ printf "GET / HTTP/1.0\r\n\r\n" | nc -n -i 1 192.168.1.5 9998
HTTP/1.1 200 OK
Content-Length:1500
Content-Type:text/html; charset=UTF-8
Yesterday ...LGwebO...
<html>
<head>
<title>Cast shell remote debugging</title>
<style> ...LGPL (Lesser GPL); IL...
</style>
<script>
<body> 2019 ...LG In...
function onLoad() {
    var tabs_list_request = new XMLHttpRequest();
    tabs_list_request.open("GET", "/json/list?t=" + new Date().getTime(), true);
    tabs_list_request.onreadystatechange = onReady;
    tabs_list_request.send();
}
Computational Ne...
function onReady() {
    if(this.readyState == 4 && this.status == 200) {
        if(this.response != null) {
            var responseJSON = JSON.parse(this.response);
            for (var i = 0; i < responseJSON.length; ++i) {
                appendItem(responseJSON[i]);
            }
        }
    }
}
I migliori sistemi per realtà...
23/08/2016 ...and LG-G3 o qu...
} Attachments
function appendItem(item_object) {
    var frontend_ref;
    if (item_object.devtoolsFrontendUrl) {
        frontend_ref = document.createElement("a");
        frontend_ref.href = item_object.devtoolsFrontendUrl;
        frontend_ref.title = item_object.title;
    } else {
        frontend_ref = document.createElement("div");
        frontend_ref.title = "The tab already has active debugging session";
    }
    var text = document.createElement("div");
    if (item_object.title)
        text.innerText = item_object.title;
    else
        text.innerText = "(untitled tab)";
    text.style.cssText = "background-image:url(" + item_object.faviconUrl + ")";
    frontend_ref.appendChild(text);

    var item = document.createElement("p");
    item.appendChild(frontend_ref);

    document.getElementById("items").appendChild(item);
}
</script>
</head>
<body onload='onLoad()'>
    <div id='caption'>Inspectable WebContents</div>
    <div id='items'></div>
</body>
</html>

```

Open TCP Port: 1775
 Open TCP Port: 3000 hbc
 Open TCP Port: 3001 redwood-b
 Open TCP Port: 9955
 Open TCP Port: 9998 distinct32
 Open TCP Port: 18181 opsec-cvp
 Open TCP Port: 36866

Port Gear has completed...

\$ sudo nmap -sS -Pn -sV <LGsmartTV.ipc
 3000/tcp open http LG smart TV http se
 15/16/2019 ...open ssl/http LG smart TV http se
 9998/tcp open http Google Chromecast

Last login: Tue Jul 23 22:07:58 on ttys001
 gfmacbook:~ gabrielefilosofi\$ w

USER	TTY	FROM	LOGIN@	IDLE	WHAT
gabrielefilosofi	pts/0	console	-	21:43	2:11 -
				21:43	curl -v http
				23:54	- w
				23:54	-
				21:43	2:11 -
				21:43	- bash
				23:54	- w

A utility for downloading files from the Internet.

- To download all files and subfolders in directory:

host \$ wget -r -np -nH --cut-dirs=3 -

-r : recursively

host \$ nmap -sV -p 9998 192.168.1.5 --script=http-enum

```

gfmacbook:~ gabrielefilosofi$ nmap -sV -Pn -p 9998 192.168.1.5 --script=http-enum
Starting Nmap 7.70 ( https://nmap.org ) at 2019-07-30 19:30 CEST
Nmap scan report for LGweb0STV.homenet.telecomitalia.it (192.168.1.5)
Host is up (0.026s latency).

PORT      STATE SERVICE VERSION
9998/tcp  open  http  Google Chromecast httpd
| http-enum:
|   /images/Safeword_Token.jpg: Citrix
|   /_vti_txt/: Frontpage file or folder
|   /wp-admin/upgrade.php: Wordpress login page.
|   /infusions/avatar_studio/avatar_studio.php: PHP-Fusion Mod avatar_studio
|_  /nfsvrlets/servlet/SPSRouterServlet/: netForensics
Service Info: Device: media device

```

```

PORT      STATE SERVICE VERSION
9998/tcp  open  http  Google Chromecast httpd
| http-enum:
|   /images/Safeword_Token.jpg: Citrix
|   /_vti_txt/: Frontpage file or folder
|   /wp-admin/upgrade.php: Wordpress login page.
|   /infusions/avatar_studio/avatar_studio.php: PHP-Fusion Mod avatar_studio
|_  /nfsvrlets/servlet/SPSRouterServlet/: netForensics
Service Info: Device: media device

```

hacking router

```

host $ sudo nmap -sS -Pn -sV 192.168.1.1
      53/tcp  open  domain   (generic dns response: NOTIMP)
      80/tcp  open  http     Mbedthis-Appweb (Thomson Technicolor broadband router
          http admin)
      443/tcp open  ssl/http Mbedthis-Appweb (Thomson Technicolor broadband router
          http admin)
      5001/tcp open  commplex-link?

```

wget

A utility for downloading files from the Internet

- To download all files and subfolders in ddd directory of a HTTP web site with index.html


```
host $ wget -r -np -nH --cut-dirs=3 -R index.html http://hostname/aaa/bbb/ccc/
            ddd/
      -r : recursively
      -np : not going to upper directories, like ccc/...
      -nH : not saving files to hostname folder
      --cut-dirs=3 : but saving it to ddd by omitting first 3 folders aaa, bbb, ccc
      -R index.html : excluding index.html files
```

dig

A tool for interrogating DNS servers

- To ask for a specific record

```
host $ dig @<dnsserver> <recordname> <querytype> +<option>
```

If no <dnsserver> is provided, it is supplied by /etc/resolv.conf

<querytype> can be ANY, A (default), MX, SIG, TXT, SOA, MS, etc.

<option> can be trace, norecurse, etc.

Example:

```
gabriele@fub:~$ dig @8.8.8.8 cosmed.com ANY
; <>> DiG 9.11.3-1ubuntu1.8-Ubuntu <>> @8.8.8.8 cosmed.com ANY
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<- opcode: QUERY, status: NOERROR, id: 41819
;; flags: qr rd ra; QUERY: 1, ANSWER: 6, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 512
; QUESTION SECTION:
;cosmed.com.      IN  ANY

;; ANSWER SECTION:
cosmed.com.    899  IN  A   185.56.219.165
cosmed.com.    899  IN  NS  ns1.register.it.
cosmed.com.    899  IN  NS  ns2.register.it.
cosmed.com.    899  IN  SOA ns1.register.it. hostmaster.register.it. 2019051702 10800 3600 604800 86400
cosmed.com.    899  IN  MX  0  cosmed-com.mail.protection.outlook.com.
cosmed.com.    899  IN  TXT "v=spf1 include:spf.protection.outlook.com -all"

;; Query time: 86 msec
;; SERVER: 8.8.8#53(8.8.8.8)
;; WHEN: Tue Jul  9 12:50:16 CEST 2019
;; MSG SIZE  rcvd: 259
```

- To ask for IP corresponding to a domain
host \$ dig @192.168.0.241 google.com
- To ask for reverse DNS lookup of a given IP
host \$ dig -x <ipaddr>