

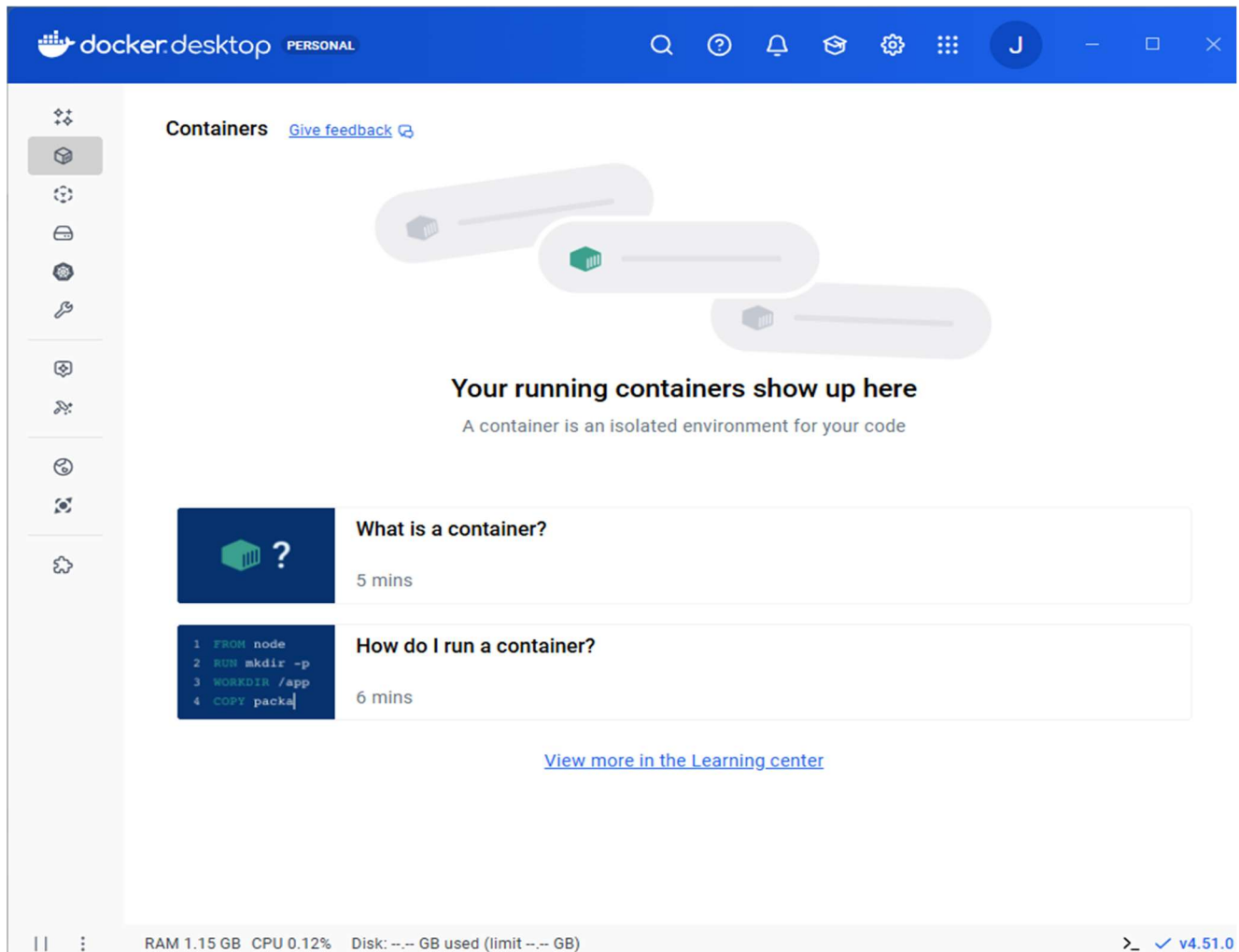
TP1

CYBERSECURITE

AGBENONZAN KOSSIVI JACQUES JUNIOR
SANOGO SOULEYMANE

1) Lancer l'environnement

- Lancez Docker Desktop



- Ouvrez un terminal dans ce dossier.
- Lancez : **`docker compose up -d`**

- Nous avons ouvert docker, puis via notre invite de commande windows cmd nous avons lancer `docker compose up -d` pour lancer le conteneur.

```
C:\WINDOWS\system32\cmd. X + v
Microsoft Windows [Version 10.0.26220.7070]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Junio>cd desktop\tp1\tp1-cyber-lab

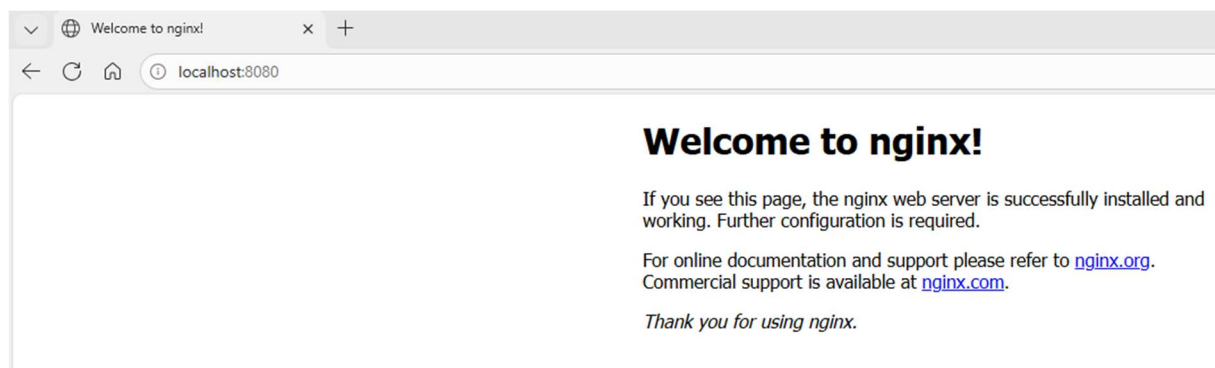
C:\Users\Junio\Desktop\TP1\tp1-cyber-lab>docker-compose up -d
[+] Running 4/4
 ✓ Network tp1-cyber-lab_labnet Created 0.1s
 ✓ Container tp1_attacker Started 0.8s
 ✓ Container tp1_web Started 0.9s
 ✓ Container tp1_target Started 1.0s

C:\Users\Junio\Desktop\TP1\tp1-cyber-lab>
```

-

- Vérifiez :

- **HTTP : ouvrez `http://localhost:8080`**

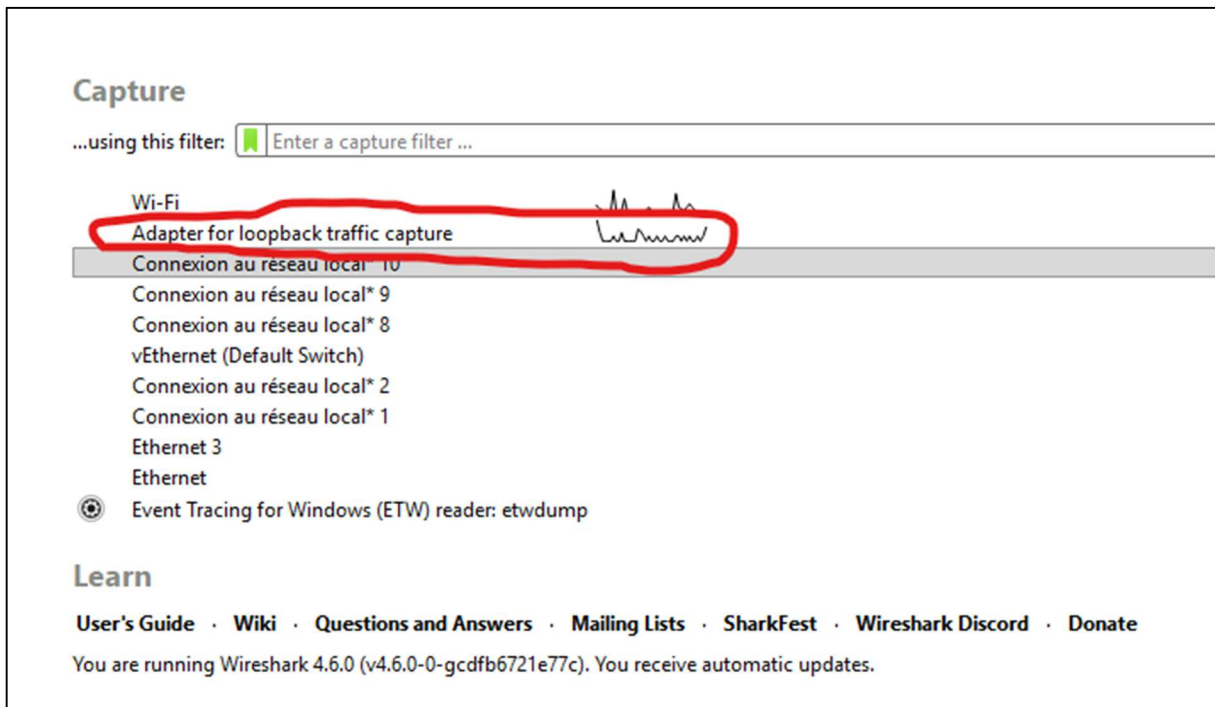


- (Optionnel) SSH : `ssh -p 2222 student@localhost` (mdp `student`)
- Nous nous sommes connecté à la machine distante via : -
- (Optionnel) SSH : `ssh -p 2222 student@localhost` (mdp `student`)

```
C:\Users\Junio\Desktop\TP1\tp1-cyber-lab>ssh -p 2222 student@localhost
The authenticity of host '[localhost]:2222 ([::1]:2222)' can't be established.
ED25519 key fingerprint is SHA256:nRVJ85RkaBnZr1GaNmOnrBcL6NXw+WAtw9wma5GLHko.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '[localhost]:2222' (ED25519) to the list of known hosts.
student@localhost's password:
Welcome to OpenSSH Server
0385de44b81d:~$ |
```

2. Démarrer la capture pour les échanges avec le Port 8080

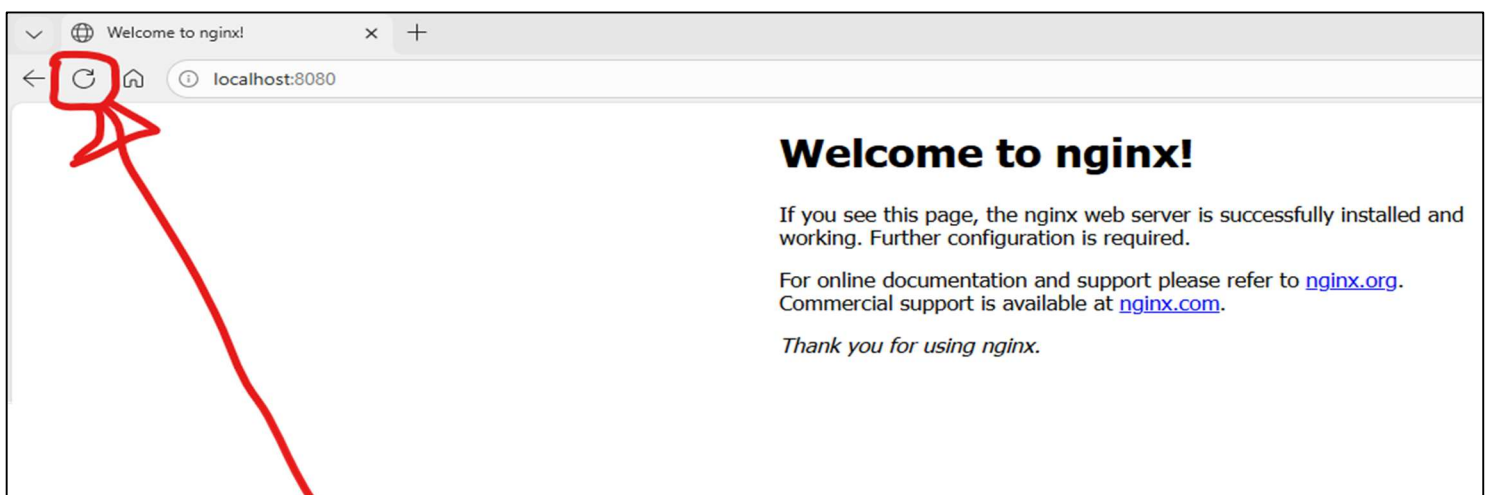
- Ouvrir WireShark
- Double-cliquez sur **"Adapter for loopback traffic capture"**



- La capture devrait démarrer automatiquement

3. Générer du trafic

- Ouvrez votre navigateur
- Allez sur : <http://localhost:8080>
- Actualisez la page 2-3 fois



3. Observer dans Wireshark

Baseline.pcapng

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

Apply a display filter ... <Ctrl-/>

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	127.0.0.1	127.0.0.1	TCP	56	64561 → 49350 [SYN] Seq=0 Win=65535 Len=0 MSS=65495 WS=256 SACK_PERM
2	0.000073	127.0.0.1	127.0.0.1	TCP	56	49350 → 64561 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=65495 WS=256 SACK_PERM
3	0.000119	127.0.0.1	127.0.0.1	TCP	44	64561 → 49350 [ACK] Seq=1 Ack=1 Win=65280 Len=0
4	0.000191	127.0.0.1	127.0.0.1	TCP	41132	64561 → 49350 [PSH, ACK] Seq=1 Ack=1 Win=65280 Len=41088
5	0.000277	127.0.0.1	127.0.0.1	TCP	44	49350 → 64561 [ACK] Seq=1 Ack=1089 Win=130816 Len=0
6	0.000313	127.0.0.1	127.0.0.1	TCP	44	64561 → 49350 [FIN, ACK] Seq=41089 Ack=1 Win=65280 Len=0
7	0.000335	127.0.0.1	127.0.0.1	TCP	44	49350 → 64561 [ACK] Seq=1 Ack=41090 Win=130816 Len=0
8	0.000580	127.0.0.1	127.0.0.1	TCP	56	64562 → 49350 [SYN] Seq=0 Win=65535 Len=0 MSS=65495 WS=256 SACK_PERM
9	0.000628	127.0.0.1	127.0.0.1	TCP	56	49350 → 64562 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=65495 WS=256 SACK_PERM
10	0.000655	127.0.0.1	127.0.0.1	TCP	44	64562 → 49350 [ACK] Seq=1 Ack=1 Win=65280 Len=0
11	0.000698	127.0.0.1	127.0.0.1	TCP	41132	64562 → 49350 [PSH, ACK] Seq=1 Ack=1 Win=65280 Len=41088
12	0.000761	127.0.0.1	127.0.0.1	TCP	44	49350 → 64562 [ACK] Seq=1 Ack=41089 Win=130816 Len=0
13	0.000797	127.0.0.1	127.0.0.1	TCP	44	64562 → 49350 [FIN, ACK] Seq=41089 Ack=1 Win=65280 Len=0
14	0.000817	127.0.0.1	127.0.0.1	TCP	44	49350 → 64562 [ACK] Seq=1 Ack=41090 Win=130816 Len=0
15	0.028153	127.0.0.1	127.0.0.1	TCP	44	49350 → 64561 [FIN, ACK] Seq=1 Ack=41090 Win=130816 Len=0
16	0.028317	127.0.0.1	127.0.0.1	TCP	44	64561 → 49350 [ACK] Seq=41090 Ack=2 Win=65280 Len=0
17	0.060664	127.0.0.1	127.0.0.1	TCP	44	49350 → 64560 [FIN, ACK] Seq=1 Ack=1 Win=511 Len=0
18	0.060815	127.0.0.1	127.0.0.1	TCP	44	64560 → 49350 [ACK] Seq=1 Ack=2 Win=255 Len=0
19	1.997752	127.0.0.1	127.0.0.1	TCP	56	64563 → 49350 [SYN] Seq=0 Win=65535 Len=0 MSS=65495 WS=256 SACK_PERM
20	1.997865	127.0.0.1	127.0.0.1	TCP	56	49350 → 64563 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=65495 WS=256 SACK_PERM
21	1.997929	127.0.0.1	127.0.0.1	TCP	44	64563 → 49350 [ACK] Seq=1 Ack=1 Win=65280 Len=0
22	1.998031	127.0.0.1	127.0.0.1	TCP	41132	64563 → 49350 [PSH, ACK] Seq=1 Ack=1 Win=65280 Len=41088
23	1.998118	127.0.0.1	127.0.0.1	TCP	44	49350 → 64563 [ACK] Seq=1 Ack=41089 Win=130816 Len=0
24	1.998150	127.0.0.1	127.0.0.1	TCP	44	64563 → 49350 [FIN, ACK] Seq=41089 Ack=1 Win=65280 Len=0
25	1.998171	127.0.0.1	127.0.0.1	TCP	44	49350 → 64563 [ACK] Seq=1 Ack=41090 Win=130816 Len=0
26	1.998277	127.0.0.1	127.0.0.1	TCP	56	64564 → 49350 [SYN] Seq=0 Win=65535 Len=0 MSS=65495 WS=256 SACK_PERM
27	1.998706	127.0.0.1	127.0.0.1	TCP	56	49350 → 64564 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=65495 WS=256 SACK_PERM
28	1.998754	127.0.0.1	127.0.0.1	TCP	44	64564 → 49350 [ACK] Seq=1 Ack=1 Win=65280 Len=0
29	1.998862	127.0.0.1	127.0.0.1	TCP	41132	64564 → 49350 [PSH, ACK] Seq=1 Ack=1 Win=65280 Len=41088
30	1.998914	127.0.0.1	127.0.0.1	TCP	44	49350 → 64564 [ACK] Seq=1 Ack=41089 Win=130816 Len=0
31	1.998960	127.0.0.1	127.0.0.1	TCP	44	64564 → 49350 [FIN, ACK] Seq=41089 Ack=1 Win=65280 Len=0
32	1.998980	127.0.0.1	127.0.0.1	TCP	44	49350 → 64564 [ACK] Seq=1 Ack=41090 Win=130816 Len=0
33	1.999317	127.0.0.1	127.0.0.1	TCP	56	64565 → 49350 [SYN] Seq=0 Win=65535 Len=0 MSS=65495 WS=256 SACK_PERM
34	1.999415	127.0.0.1	127.0.0.1	TCP	56	49350 → 64565 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=65495 WS=256 SACK_PERM
35	1.999467	127.0.0.1	127.0.0.1	TCP	44	64565 → 49350 [ACK] Seq=1 Ack=1 Win=65280 Len=0
36	1.999564	127.0.0.1	127.0.0.1	TCP	41132	64565 → 49350 [PSH, ACK] Seq=1 Ack=1 Win=65280 Len=41088
37	1.999628	127.0.0.1	127.0.0.1	TCP	44	49350 → 64565 [ACK] Seq=1 Ack=41089 Win=130816 Len=0
38	1.999655	127.0.0.1	127.0.0.1	TCP	44	64565 → 49350 [FIN, ACK] Seq=41089 Ack=1 Win=65280 Len=0

> Frame 91: Packet, 76 bytes on wire (608 bits), 76 bytes captured (608 bits) on interface \Device\NPF_{...} Loopback, id 0

> Null/Loopback

> Internet Protocol Version 6, Src: ::1, Dst: ::1

> Transmission Control Protocol, Src Port: 49252, Dst Port: 8080, Seq: 0, Len: 0

0000 18 00 00 00 00 0f da 42 00 20 06 00 00 00 00:B.....
0010 00 00 00 00 00 00 00 00 00 00 01 00 00 00:d.....
0020 00 00 00 00 00 00 00 00 00 00 01 c0 64 1f 90x:.....f:~...
0030 78 a9 b1 c4 00 00 00 00 00 02 ff ff 6a 9c 00 00
0040 02 04 ff c3 01 03 03 08 01 01 04 02:

Baseline.pcapng | Packets: 379 - Dropped: 0 (0.0%) | Profile: Default

Nous voyons :

- **Paquets TCP** avec des adresses **127.0.0.1**
- **Handshakes SYN** → SYN-ACK → ACK
- **Requêtes HTTP GET** vers le port 8080
- **Patientons 30 secondes à 1 minute et arrêtons la capture**

4. Arrêter et sauvegarder

ANALYSE POUR "ADAPTER FOR LOOPBACK TRAFFIC CAPTURE"

Notes à prendre pendant la capture

Observation	À noter
Nombre de paquets capturés	379
Protocoles identifiés	TCP, http, MDNS,UDP,ICMP,SSDP

1- TCP (Transmission Control Protocol)

On remarque que lorsqu'on applique un filtre « **tcp.port == 8080** »

On observe la séquence suivante entre le **Port 8080** et le **Port 49252**,
aussi entre le **Port 8080** et le **Port 59503** :

- [SYN] → [SYN-ACK] → [ACK]

Baseline.pcapng

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

tcp.port == 8080

No.	Time	Source	Destination	Protocol	Length	Info
91	6.667928	:::1	:::1	TCP	76	49252 → 8080 [SYN] Seq=0 Win=65535 Len=0 MSS=65475 WS=256 SACK_PERM
92	6.668036	:::1	:::1	TCP	76	8080 → 49252 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=65475 WS=256 SACK_PERM
93	6.668081	:::1	:::1	TCP	64	49252 → 8080 [ACK] Seq=1 Ack=1 Win=65280 Len=0
94	6.675040	:::1	:::1	HTTP	904	GET / HTTP/1.1
95	6.675081	:::1	:::1	TCP	64	8080 → 49252 [ACK] Seq=1 Ack=841 Win=64512 Len=0
96	6.676833	:::1	:::1	HTTP	244	HTTP/1.1 304 Not Modified
97	6.676878	:::1	:::1	TCP	64	49252 → 8080 [ACK] Seq=841 Ack=181 Win=65280 Len=0
98	6.681302	:::1	:::1	TCP	76	59503 → 8080 [SYN] Seq=0 Win=65535 Len=0 MSS=65475 WS=256 SACK_PERM
99	6.681386	:::1	:::1	TCP	76	8080 → 59503 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=65475 WS=256 SACK_PERM
100	6.681430	:::1	:::1	TCP	64	59503 → 8080 [ACK] Seq=1 Ack=1 Win=65280 Len=0
101	8.293674	:::1	:::1	HTTP	904	GET / HTTP/1.1
102	8.293722	:::1	:::1	TCP	64	8080 → 49252 [ACK] Seq=181 Ack=1681 Win=63744 Len=0
103	8.294843	:::1	:::1	HTTP	244	HTTP/1.1 304 Not Modified
104	8.294875	:::1	:::1	TCP	64	49252 → 8080 [ACK] Seq=1681 Ack=361 Win=65024 Len=0
105	9.280503	:::1	:::1	HTTP	904	GET / HTTP/1.1
106	9.280558	:::1	:::1	TCP	64	8080 → 49252 [ACK] Seq=361 Ack=2521 Win=62976 Len=0
107	9.281628	:::1	:::1	HTTP	244	HTTP/1.1 304 Not Modified
108	9.281681	:::1	:::1	TCP	64	49252 → 8080 [ACK] Seq=2521 Ack=541 Win=64768 Len=0

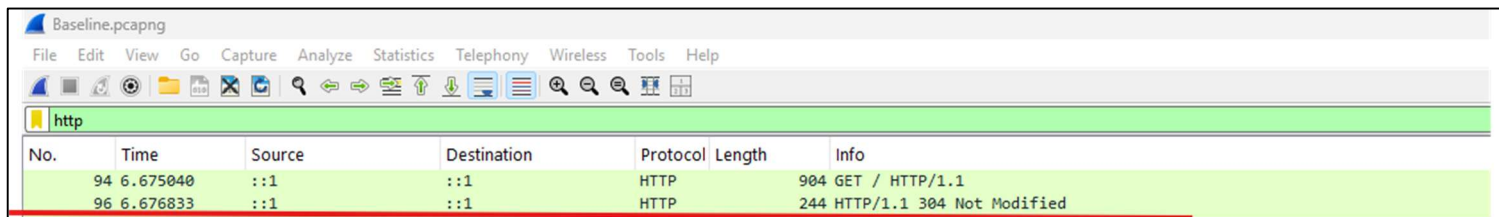
D'après le Cours quand on a la séquence :

- [SYN] → [SYN-ACK] → [ACK] alors le Port 8080 est **Ouvert** Pour TCP
- [RST] → [ACK] : Le port est **Fermé** Pour TCP

Conclusion : le port 8080 est Ouvert pour TCP dans notre cas

2- HTTP (HyperText Transport Protocol)

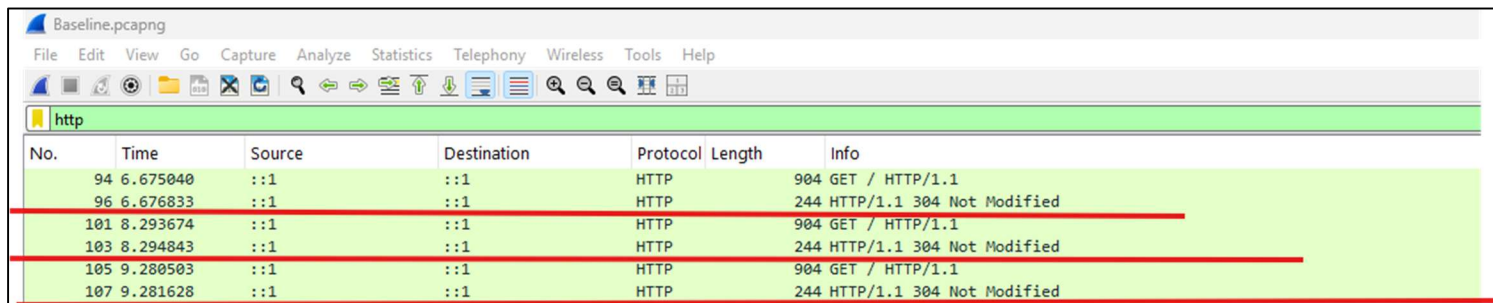
On appliquant le filtre à **http** On observe la séquence suivante :



The screenshot shows the Wireshark interface with the filter 'http' applied. The packet list table contains two entries:

No.	Time	Source	Destination	Protocol	Length	Info
94	6.675040	:::1	:::1	HTTP	904	GET / HTTP/1.1
96	6.676833	:::1	:::1	HTTP	244	HTTP/1.1 304 Not Modified

En Actualisant la page [Welcome to nginx!](#) Trois Fois, On observe la même séquence en boucle



The screenshot shows the Wireshark interface with the filter 'http' applied. The packet list table contains six entries, showing a repeating sequence of requests and responses:

No.	Time	Source	Destination	Protocol	Length	Info
94	6.675040	:::1	:::1	HTTP	904	GET / HTTP/1.1
96	6.676833	:::1	:::1	HTTP	244	HTTP/1.1 304 Not Modified
101	8.293674	:::1	:::1	HTTP	904	GET / HTTP/1.1
103	8.294843	:::1	:::1	HTTP	244	HTTP/1.1 304 Not Modified
105	9.280503	:::1	:::1	HTTP	904	GET / HTTP/1.1
107	9.281628	:::1	:::1	HTTP	244	HTTP/1.1 304 Not Modified

- Etant donnée que nous travaillons en local, Un client Web (nous) **d'adresse ip :::1** interroge un serveur Web **d'adresse ip :::1** (Encore nous) pour obtenir des Informations sur la page web **HTTP/1.1 (N°96) avec GET/http/1.1**
- Le Serveur lui reponds, **http/1.1 304 Not Modified** : Qui veut dire Chers Clients Vous êtes déjà sur la page que vous demandez (**http/1.1**) pas besoin de recharger toute la page (**304 Not Modified**)
- Ce qui se Comprend vu que nous actualisons simplement la page.

En appliquant le filtre pour MDNS on a les paquets suivants :

Baseline.pcapng

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

No.	Time	Source	Destination	Protocol	Length	Info
172	16.077662	192.168.56.1	224.0.0.251	MDNS	459	Standard query response 0x0000 PTR Juroot._dovsc._tcp.local SRV 0 6800 Juroot.local TXT
173	16.078062	192.168.100.13	224.0.0.251	MDNS	459	Standard query response 0x0000 PTR Juroot._dovsc._tcp.local SRV 0 6800 Juroot.local TXT
174	16.078330	172.31.16.1	224.0.0.251	MDNS	459	Standard query response 0x0000 PTR Juroot._dovsc._tcp.local SRV 0 6800 Juroot.local TXT
175	16.078555	172.30.32.1	224.0.0.251	MDNS	459	Standard query response 0x0000 PTR Juroot._dovsc._tcp.local SRV 0 6800 Juroot.local TXT
176	16.078832	fe80::9964:2a60:e9b...f02::fb	224.0.0.251	MDNS	479	Standard query response 0x0000 PTR Juroot._dovsc._tcp.local SRV 0 6800 Juroot.local TXT
177	16.079085	fe80::d107:337a:f66...f02::fb	224.0.0.251	MDNS	479	Standard query response 0x0000 PTR Juroot._dovsc._tcp.local SRV 0 6800 Juroot.local TXT
178	16.079335	fe80::30c:8d8a:2ced...f02::fb	224.0.0.251	MDNS	479	Standard query response 0x0000 PTR Juroot._dovsc._tcp.local SRV 0 6800 Juroot.local TXT
179	16.079570	fe80::1a4f:289c:538...f02::fb	224.0.0.251	MDNS	479	Standard query response 0x0000 PTR Juroot._dovsc._tcp.local SRV 0 6800 Juroot.local TXT
180	16.080139	192.168.56.1	224.0.0.251	MDNS	74	Standard query 0x0000 ANY Juroot._dovsc._tcp.local, "QM" question
181	16.080270	192.168.100.13	224.0.0.251	MDNS	74	Standard query 0x0000 ANY Juroot._dovsc._tcp.local, "QM" question
182	16.080405	172.31.16.1	224.0.0.251	MDNS	74	Standard query 0x0000 ANY Juroot._dovsc._tcp.local, "QM" question
183	16.080531	172.30.32.1	224.0.0.251	MDNS	74	Standard query 0x0000 ANY Juroot._dovsc._tcp.local, "QM" question
184	16.080662	fe80::9964:2a60:e9b...f02::fb	224.0.0.251	MDNS	94	Standard query 0x0000 ANY Juroot._dovsc._tcp.local, "QM" question
185	16.080827	fe80::d107:337a:f66...f02::fb	224.0.0.251	MDNS	94	Standard query 0x0000 ANY Juroot._dovsc._tcp.local, "QM" question
186	16.080993	fe80::30c:8d8a:2ced...f02::fb	224.0.0.251	MDNS	94	Standard query 0x0000 ANY Juroot._dovsc._tcp.local, "QM" question
187	16.081119	fe80::1a4f:289c:538...f02::fb	224.0.0.251	MDNS	94	Standard query 0x0000 ANY Juroot._dovsc._tcp.local, "QM" question
188	16.336163	192.168.56.1	224.0.0.251	MDNS	74	Standard query 0x0000 ANY Juroot._dovsc._tcp.local, "QM" question
189	16.336929	192.168.100.13	224.0.0.251	MDNS	74	Standard query 0x0000 ANY Juroot._dovsc._tcp.local, "QM" question
190	16.337668	172.31.16.1	224.0.0.251	MDNS	74	Standard query 0x0000 ANY Juroot._dovsc._tcp.local, "QM" question
191	16.338277	172.30.32.1	224.0.0.251	MDNS	74	Standard query 0x0000 ANY Juroot._dovsc._tcp.local, "QM" question
192	16.339040	fe80::9964:2a60:e9b...f02::fb	224.0.0.251	MDNS	94	Standard query 0x0000 ANY Juroot._dovsc._tcp.local, "QM" question
193	16.339853	fe80::d107:337a:f66...f02::fb	224.0.0.251	MDNS	94	Standard query 0x0000 ANY Juroot._dovsc._tcp.local, "QM" question
194	16.340585	fe80::30c:8d8a:2ced...f02::fb	224.0.0.251	MDNS	94	Standard query 0x0000 ANY Juroot._dovsc._tcp.local, "QM" question
195	16.341377	fe80::1a4f:289c:538...f02::fb	224.0.0.251	MDNS	94	Standard query 0x0000 ANY Juroot._dovsc._tcp.local, "QM" question
196	16.594061	192.168.56.1	224.0.0.251	MDNS	74	Standard query 0x0000 ANY Juroot._dovsc._tcp.local, "QM" question
197	16.594923	192.168.100.13	224.0.0.251	MDNS	74	Standard query 0x0000 ANY Juroot._dovsc._tcp.local, "QM" question
198	16.595766	172.31.16.1	224.0.0.251	MDNS	74	Standard query 0x0000 ANY Juroot._dovsc._tcp.local, "QM" question
199	16.596520	172.30.32.1	224.0.0.251	MDNS	74	Standard query 0x0000 ANY Juroot._dovsc._tcp.local, "QM" question
200	16.597382	fe80::9964:2a60:e9b...f02::fb	224.0.0.251	MDNS	94	Standard query 0x0000 ANY Juroot._dovsc._tcp.local, "QM" question
201	16.598169	fe80::d107:337a:f66...f02::fb	224.0.0.251	MDNS	94	Standard query 0x0000 ANY Juroot._dovsc._tcp.local, "QM" question
202	16.599341	fe80::30c:8d8a:2ced...f02::fb	224.0.0.251	MDNS	94	Standard query 0x0000 ANY Juroot._dovsc._tcp.local, "QM" question
203	16.600270	fe80::1a4f:289c:538...f02::fb	224.0.0.251	MDNS	94	Standard query 0x0000 ANY Juroot._dovsc._tcp

Portion 1 (Requête)

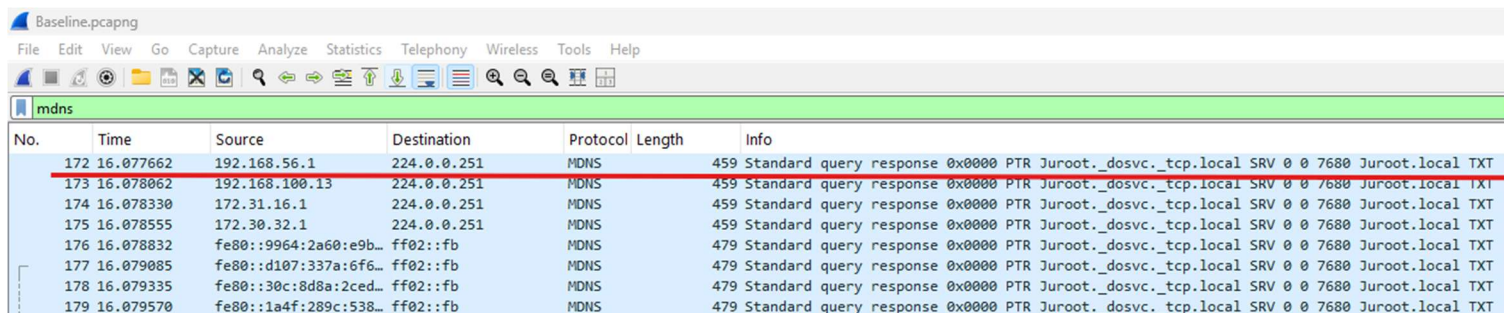
180	16.080139	192.168.56.1	224.0.0.251	MDNS	74	Standard query	0x0000	ANY	Juroot._dosvc._tcp.local,	"QM" question
181	16.080270	192.168.100.13	224.0.0.251	MDNS	74	Standard query	0x0000	ANY	Juroot._dosvc._tcp.local,	"QM" question
182	16.080405	172.31.16.1	224.0.0.251	MDNS	74	Standard query	0x0000	ANY	Juroot._dosvc._tcp.local,	"QM" question
183	16.080531	172.30.32.1	224.0.0.251	MDNS	74	Standard query	0x0000	ANY	Juroot._dosvc._tcp.local,	"QM" question
184	16.080682	fe80::9964:2a60:e9b...	ff02::fb	MDNS	94	Standard query	0x0000	ANY	Juroot._dosvc._tcp.local,	"QM" question
185	16.080827	fe80::d107:337a:6f6...	ff02::fb	MDNS	94	Standard query	0x0000	ANY	Juroot._dosvc._tcp.local,	"QM" question
186	16.080983	fe80::30c:8d8a:2ced...	ff02::fb	MDNS	94	Standard query	0x0000	ANY	Juroot._dosvc._tcp.local,	"QM" question
187	16.081119	fe80::1a4f:289c:538...	ff02::fb	MDNS	94	Standard query	0x0000	ANY	Juroot._dosvc._tcp.local,	"QM" question
188	16.336163	192.168.56.1	224.0.0.251	MDNS	74	Standard query	0x0000	ANY	Juroot._dosvc._tcp.local,	"QM" question
189	16.336929	192.168.100.13	224.0.0.251	MDNS	74	Standard query	0x0000	ANY	Juroot._dosvc._tcp.local,	"QM" question
190	16.337668	172.31.16.1	224.0.0.251	MDNS	74	Standard query	0x0000	ANY	Juroot._dosvc._tcp.local,	"QM" question
191	16.338277	172.30.32.1	224.0.0.251	MDNS	74	Standard query	0x0000	ANY	Juroot._dosvc._tcp.local,	"QM" question
192	16.339040	fe80::9964:2a60:e9b...	ff02::fb	MDNS	94	Standard query	0x0000	ANY	Juroot._dosvc._tcp.local,	"QM" question
193	16.339853	fe80::d107:337a:6f6...	ff02::fb	MDNS	94	Standard query	0x0000	ANY	Juroot._dosvc._tcp.local,	"QM" question
194	16.340585	fe80::30c:8d8a:2ced...	ff02::fb	MDNS	94	Standard query	0x0000	ANY	Juroot._dosvc._tcp.local,	"QM" question
195	16.341377	fe80::1a4f:289c:538...	ff02::fb	MDNS	94	Standard query	0x0000	ANY	Juroot._dosvc._tcp.local,	"QM" question
196	16.594061	192.168.56.1	224.0.0.251	MDNS	74	Standard query	0x0000	ANY	Juroot._dosvc._tcp.local,	"QM" question
197	16.594923	192.168.100.13	224.0.0.251	MDNS	74	Standard query	0x0000	ANY	Juroot._dosvc._tcp.local,	"QM" question
198	16.595786	172.31.16.1	224.0.0.251	MDNS	74	Standard query	0x0000	ANY	Juroot._dosvc._tcp.local,	"QM" question
199	16.596520	172.30.32.1	224.0.0.251	MDNS	74	Standard query	0x0000	ANY	Juroot._dosvc._tcp.local,	"QM" question
200	16.597382	fe80::9964:2a60:e9b...	ff02::fb	MDNS	94	Standard query	0x0000	ANY	Juroot._dosvc._tcp.local,	"QM" question
201	16.598169	fe80::d107:337a:6f6...	ff02::fb	MDNS	94	Standard query	0x0000	ANY	Juroot._dosvc._tcp.local,	"QM" question
202	16.599341	fe80::30c:8d8a:2ced...	ff02::fb	MDNS	94	Standard query	0x0000	ANY	Juroot._dosvc._tcp.local,	"QM" question
203	16.600270	fe80::1a4f:289c:538...	ff02::fb	MDNS	94	Standard query	0x0000	ANY	Juroot._dosvc._tcp.local,	"QM" question

Ici (1) un appareil d'adresse IPV6 « **fe80::d107:337a:6f6d:6cb7** » émet une requête Multicast (Requête à tout les composant sur le réseau) grâce à l'IPv6 Multicast « **ff02::fb** » via MDNS, pour demander toutes les informations sur le service « **Juroot. dosvc. tcp.local** », il en est de même pour (2) mais avec

source une adresse IPv4 source et une adresse IPv6 de diffusion ou encore dit Multicast.

NB : MDNS se loge généralement sur le **Port** : 5353.

Portion 2 (Réponse)



The image shows a Wireshark packet capture of MDNS traffic. The filter is 'mdns'. The table below represents the data shown in the packet list pane.

No.	Time	Source	Destination	Protocol	Length	Info
172	16.077662	192.168.56.1	224.0.0.251	MDNS	459	Standard query response 0x0000 PTR Juroot._dosvc._tcp.local SRV 0 0 7680 Juroot.local TXT
173	16.078062	192.168.100.13	224.0.0.251	MDNS	459	Standard query response 0x0000 PTR Juroot._dosvc._tcp.local SRV 0 0 7680 Juroot.local TXT
174	16.078330	172.31.16.1	224.0.0.251	MDNS	459	Standard query response 0x0000 PTR Juroot._dosvc._tcp.local SRV 0 0 7680 Juroot.local TXT
175	16.078555	172.30.32.1	224.0.0.251	MDNS	459	Standard query response 0x0000 PTR Juroot._dosvc._tcp.local SRV 0 0 7680 Juroot.local TXT
176	16.078832	fe80::9964:2a60:e9b...	ff02::fb	MDNS	479	Standard query response 0x0000 PTR Juroot._dosvc._tcp.local SRV 0 0 7680 Juroot.local TXT
177	16.079085	fe80::d107:337a:6f6...	ff02::fb	MDNS	479	Standard query response 0x0000 PTR Juroot._dosvc._tcp.local SRV 0 0 7680 Juroot.local TXT
178	16.079335	fe80::30c:8d8a:2ced...	ff02::fb	MDNS	479	Standard query response 0x0000 PTR Juroot._dosvc._tcp.local SRV 0 0 7680 Juroot.local TXT
179	16.079570	fe80::1a4f:289c:538...	ff02::fb	MDNS	479	Standard query response 0x0000 PTR Juroot._dosvc._tcp.local SRV 0 0 7680 Juroot.local TXT

Ici Plusieurs appareil d'adresse **IPv4 (192.168.56.1, 192.168.100.13, 172.31.16.1)** et **IPv6 (fe80::d107:337a:6f6d:6cb7)** émettent des Réponses vers des adresses IPv4 et IPv6 Multicast (donc vers toutes les autres machines du réseau) pour dire « j'ai des Informations sur **Juroot._dosvc._tcp.local** voici tout ce que je sais sur lui dans ce txt (Juroot.local.TXT) ».

4- SSDP (Simple Service Discovery Protocol)

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help						
ssdp						
No.	Time	Source	Destination	Protocol	Length	Info
95	15.449240	192.168.56.1	239.255.255.250	SSDP	202	M-SEARCH * HTTP/1.1
96	15.449601	10.0.77.143	239.255.255.250	SSDP	202	M-SEARCH * HTTP/1.1
97	15.449937	172.30.32.1	239.255.255.250	SSDP	202	M-SEARCH * HTTP/1.1
122	16.451129	192.168.56.1	239.255.255.250	SSDP	202	M-SEARCH * HTTP/1.1
123	16.451344	10.0.77.143	239.255.255.250	SSDP	202	M-SEARCH * HTTP/1.1
124	16.451523	172.30.32.1	239.255.255.250	SSDP	202	M-SEARCH * HTTP/1.1
125	17.452769	192.168.56.1	239.255.255.250	SSDP	202	M-SEARCH * HTTP/1.1
126	17.453138	10.0.77.143	239.255.255.250	SSDP	202	M-SEARCH * HTTP/1.1
127	17.453501	172.30.32.1	239.255.255.250	SSDP	202	M-SEARCH * HTTP/1.1
131	18.453488	192.168.56.1	239.255.255.250	SSDP	202	M-SEARCH * HTTP/1.1
132	18.453610	10.0.77.143	239.255.255.250	SSDP	202	M-SEARCH * HTTP/1.1
133	18.453729	172.30.32.1	239.255.255.250	SSDP	202	M-SEARCH * HTTP/1.1

> Frame 97: Packet, 202 bytes on wire (1616 bits), 202 bytes captured (1616 bits) on interface \Device\NPF_Loopback, i

> Null/Loopback

> Internet Protocol Version 4, Src: 172.30.32.1, Dst: 239.255.255.250

> User Datagram Protocol, Src Port: 53181, Dst Port: 1900

> Simple Service Discovery Protocol

Ici plusieurs machines d'adresse IPv4 (192.168.56.1, 10.0.77.143,...) émettent des messages **Multicast ou de Diffusion** (Message à tous les appareils sur le réseau) via l'adresse **Multicast (239.255.255.250)** pour demander des services web, précisément sur **http/1.1**

NB : SDNS est principalement logé sur le port 1900

REPONSES	SIGNIFICATION
M-SEARCH	Ce sont des requêtes envoyées par un client pour découvrir des services.
NOTIFY	Ce sont des annonces envoyées par des périphériques pour dire "je suis là".
Si on a M-SEARCH et NOTIFY	Le Port est actif

Conclusion : Dans notre cas le port 1900 est fermé dû à l'absence de NOTIFY

5- UDP (User Data Protocol)

icmp						
No.	Time	Source	Destination	Protocol	Length	Info
29	0.730965	10.0.77.143	10.0.77.143	ICMP	84	Destination unreachable (Host unreachable)
50	4.730632	10.0.77.143	10.0.77.143	ICMP	84	Destination unreachable (Host unreachable)
94	12.731105	10.0.77.143	10.0.77.143	ICMP	84	Destination unreachable (Host unreachable)

REPONSES	SIGNIFICATION
IMCP → Destination unreachable (Host unreachable)	Port UDP Fermé
---(Silence)	Port UDP Ouvert

Dans notre cas le port UDP est fermé

Étape 2 : Accéder au conteneur attaquant

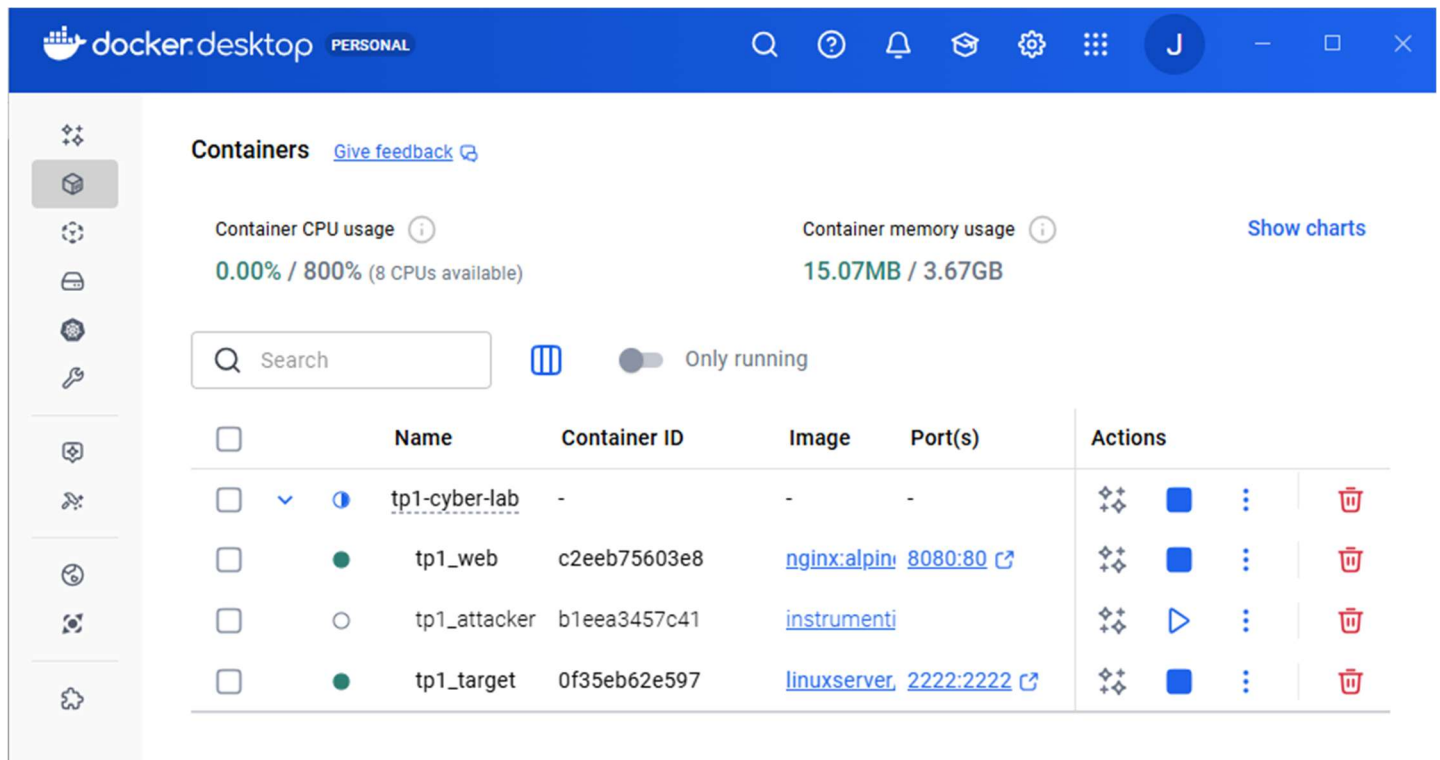
Ouvrez un nouveau terminal/CMD :

Via le Terminal Tapez : **docker compose exec attacker sh**, puis entrez
nmap -sS -p 1-1000 target

```
C:\Users\Junio\Desktop\TP1\tp1-cyber-lab>docker compose exec attacker sh
/ #
C:\Users\Junio\Desktop\TP1\tp1-cyber-lab>docker compose exec attacker sh
/ # nmap -sS -p 1-1000 target
Starting Nmap 7.95 ( https://nmap.org ) at 2025-11-15 12:36 UTC
Nmap scan report for target (172.18.0.4)
Host is up (0.000017s latency).
rDNS record for 172.18.0.4: tp1_target.tp1-cyber-lab_labnet
All 1000 scanned ports on target (172.18.0.4) are in ignored states.
Not shown: 1000 closed tcp ports (reset)
MAC Address: 32:9F:7A:76:8B:5E (Unknown)

Nmap done: 1 IP address (1 host up) scanned in 0.25 seconds
/ #
C:\Users\Junio\Desktop\TP1\tp1-cyber-lab>
```

Ou encore lancer la machine via docker



The screenshot shows the Docker Desktop interface. The top bar is blue with the Docker logo and 'PERSONAL' label. The left sidebar contains navigation icons. The main area is titled 'Containers' and shows system usage: 'Container CPU usage 0.00% / 800% (8 CPUs available)' and 'Container memory usage 15.07MB / 3.67GB'. Below this is a search bar and a filter toggle for 'Only running'. A table lists four containers: 'tp1-cyber-lab', 'tp1_web', 'tp1_attacker', and 'tp1_target'. Each row includes a checkbox, status icon, name, ID, image, ports, and a set of action icons (restart, stop, delete, etc.).

	Name	Container ID	Image	Port(s)	Actions
☐	tp1-cyber-lab	-	-	-	[Restart] [Stop] [Delete]
☐	tp1_web	c2eeb75603e8	nginx:alpine	8080:80	[Restart] [Stop] [Delete]
☐	tp1_attacker	b1eea3457c41	instrumenti		[Restart] [Stop] [Delete]
☐	tp1_target	0f35eb62e597	linuxserver	2222:2222	[Restart] [Stop] [Delete]

Dans WireShark Double-Cliquez sur vEthernet (WSL(Hyper-V firewall))



Laisser WireShark tourne pendant 30 secondes à une minute et stoppé la capture

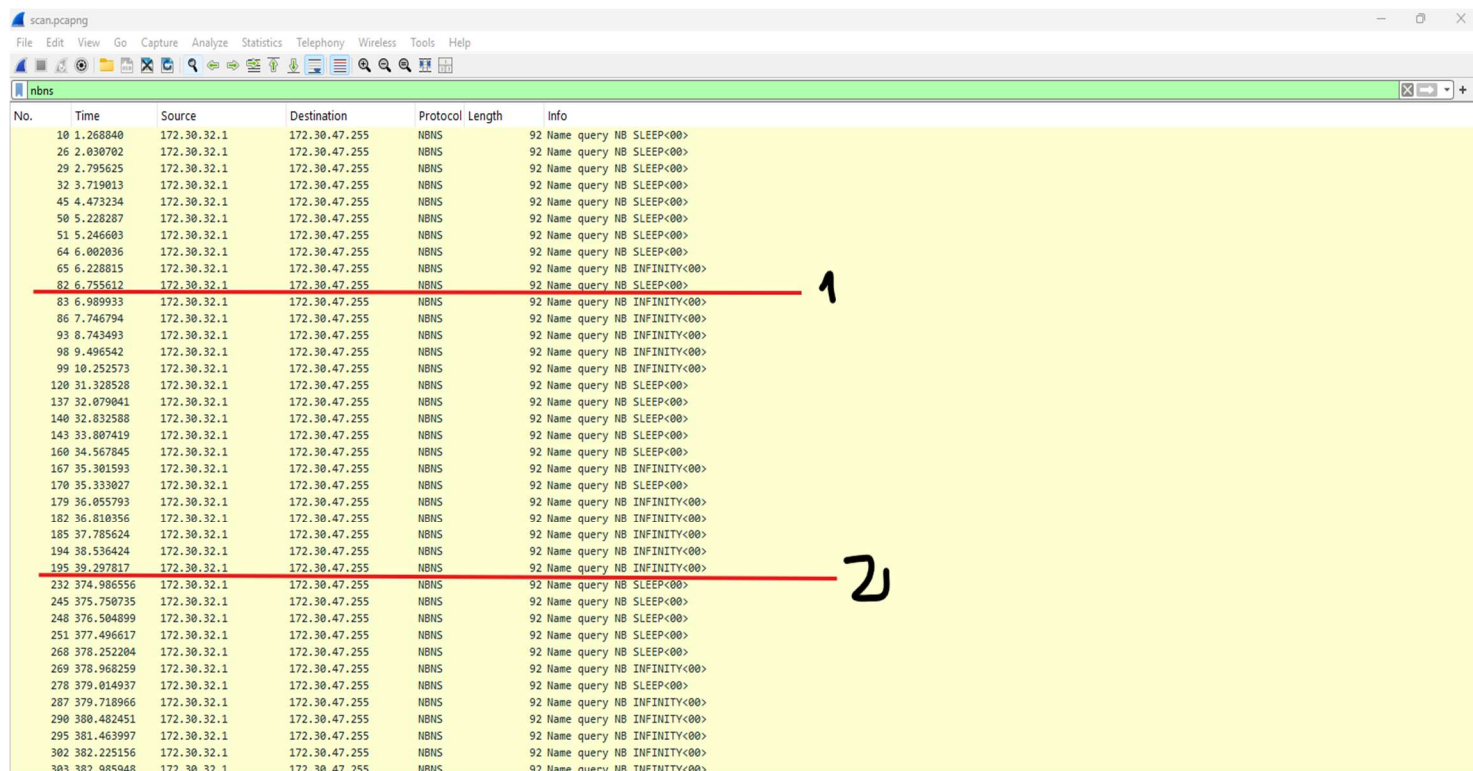
No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	172.30.32.1	224.0.0.251	MDNS	71	Standard query 0x0000 A sleep.local, "QM" question
2	0.003546	fe80::1a4f:289c:538...	ff02::fb	MDNS	91	Standard query 0x0000 A sleep.local, "QM" question
3	0.006722	172.30.32.1	224.0.0.251	MDNS	71	Standard query 0x0000 A sleep.local, "QM" question
4	0.010465	fe80::1a4f:289c:538...	ff02::fb	MDNS	91	Standard query 0x0000 A sleep.local, "QM" question
5	0.017338	172.30.32.1	224.0.0.251	MDNS	71	Standard query 0x0000 A sleep.local, "QM" question
6	0.018421	fe80::1a4f:289c:538...	ff02::fb	MDNS	91	Standard query 0x0000 A sleep.local, "QM" question
7	0.032605	172.30.32.1	224.0.0.251	MDNS	71	Standard query 0x0000 A sleep.local, "QM" question
8	0.033924	fe80::1a4f:289c:538...	ff02::fb	MDNS	91	Standard query 0x0000 A sleep.local, "QM" question
9	1.268821	172.30.32.1	224.0.0.251	MDNS	65	Standard query 0x0000 AAAA sleep, "QM" question
10	1.268840	172.30.32.1	172.30.47.255	NBNS	92	Name query NB SLEEP<00>
11	1.269588	fe80::1a4f:289c:538...	ff02::fb	MDNS	85	Standard query 0x0000 AAAA sleep, "QM" question
12	1.270234	172.30.32.1	224.0.0.251	MDNS	65	Standard query 0x0000 A sleep, "QM" question
13	1.271022	fe80::1a4f:289c:538...	ff02::fb	MDNS	85	Standard query 0x0000 A sleep, "QM" question
14	1.285439	172.30.32.1	224.0.0.251	MDNS	65	Standard query 0x0000 AAAA sleep, "QM" question
15	1.286120	fe80::1a4f:289c:538...	ff02::fb	MDNS	85	Standard query 0x0000 AAAA sleep, "QM" question
16	1.286827	172.30.32.1	224.0.0.251	MDNS	65	Standard query 0x0000 A sleep, "QM" question
17	1.287499	fe80::1a4f:289c:538...	ff02::fb	MDNS	85	Standard query 0x0000 A sleep, "QM" question
18	1.685687	172.30.32.1	224.0.0.251	MDNS	65	Standard query 0x0000 A sleep, "QM" question
19	1.686323	fe80::1a4f:289c:538...	ff02::fb	MDNS	85	Standard query 0x0000 A sleep, "QM" question
20	1.687049	172.30.32.1	224.0.0.251	MDNS	65	Standard query 0x0000 AAAA sleep, "QU" question
21	1.687705	fe80::1a4f:289c:538...	ff02::fb	MDNS	85	Standard query 0x0000 AAAA sleep, "QU" question
22	1.704091	172.30.32.1	224.0.0.251	MDNS	65	Standard query 0x0000 A sleep, "QM" question
23	1.707192	fe80::1a4f:289c:538...	ff02::fb	MDNS	85	Standard query 0x0000 A sleep, "QM" question
24	1.709543	172.30.32.1	224.0.0.251	MDNS	65	Standard query 0x0000 AAAA sleep, "QU" question
25	1.711507	fe80::1a4f:289c:538...	ff02::fb	MDNS	85	Standard query 0x0000 AAAA sleep, "QU" question
26	2.030702	172.30.32.1	172.30.47.255	NBNS	92	Name query NB SLEEP<00>
27	2.463081	172.30.32.1	224.0.0.251	MDNS	71	Standard query 0x0000 A sleep.local, "QM" question
28	2.466341	fe80::1a4f:289c:538...	ff02::fb	MDNS	91	Standard query 0x0000 A sleep.local, "QM" question
29	2.795625	172.30.32.1	172.30.47.255	NBNS	92	Name query NB SLEEP<00>
30	2.878665	172.30.32.1	224.0.0.251	MDNS	71	Standard query 0x0000 A sleep.local, "QM" question
31	2.880733	fe80::1a4f:289c:538...	ff02::fb	MDNS	91	Standard query 0x0000 A sleep.local, "QM" question
32	3.719013	172.30.32.1	172.30.47.255	NBNS	92	Name query NB SLEEP<00>
33	3.719164	172.30.32.1	224.0.0.251	MDNS	65	Standard query 0x0000 AAAA sleep, "QM" question
34	3.719936	fe80::1a4f:289c:538...	ff02::fb	MDNS	85	Standard query 0x0000 AAAA sleep, "QM" question
35	3.720528	172.30.32.1	224.0.0.251	MDNS	65	Standard query 0x0000 A sleep, "QM" question
36	3.721098	fe80::1a4f:289c:538...	ff02::fb	MDNS	85	Standard query 0x0000 A sleep, "QM" question
37	3.963037	172.30.32.1	224.0.0.251	MDNS	71	Standard query 0x0000 A sleep.local, "QM" question
38	3.966609	fe80::1a4f:289c:538...	ff02::fb	MDNS	91	Standard query 0x0000 A sleep.local, "QM" question
39	4.130473	172.30.32.1	224.0.0.251	MDNS	65	Standard query 0x0000 A sleep, "QM" question
40	4.141813	fe80::1a4f:289c:538...	ff02::fb	MDNS	85	Standard query 0x0000 A sleep, "QM" question
41	4.143865	172.30.32.1	224.0.0.251	MDNS	65	Standard query 0x0000 AAAA sleep, "QU" question
42	4.146014	fe80::1a4f:289c:538...	ff02::fb	MDNS	85	Standard query 0x0000 AAAA sleep, "QU" question
43	4.378287	172.30.32.1	224.0.0.251	MDNS	71	Standard query 0x0000 A sleep.local, "QM" question
44	4.379061	fe80::1a4f:289c:538...	ff02::fb	MDNS	91	Standard query 0x0000 A sleep.local, "QM" question
45	4.473234	172.30.32.1	172.30.47.255	NBNS	92	Name query NB SLEEP<00>
46	4.975649	172.30.32.1	224.0.0.251	MDNS	74	Standard query 0x0000 A infinity.local, "QM" question
47	4.979016	fe80::1a4f:289c:538...	ff02::fb	MDNS	94	Standard query 0x0000 A infinity.local, "QM" question
48	4.982115	172.30.32.1	224.0.0.251	MDNS	74	Standard query 0x0000 A infinity.local, "QM" question

Notes à prendre pendant la capture

Observation	À noter
Nombre de paquets capturés	315
Protocoles identifiés	MDNS, NBNS

1- NBNS (NetBIOS Name Service).

C'est un protocole ancien utilisé par Windows pour résoudre des noms d'hôtes sur un réseau local, avant que DNS ne soit généralisé.



No.	Time	Source	Destination	Protocol	Length	Info
10	1.268840	172.30.32.1	172.30.47.255	NBNS	92	Name query NB SLEEP<00>
26	2.030702	172.30.32.1	172.30.47.255	NBNS	92	Name query NB SLEEP<00>
29	2.795625	172.30.32.1	172.30.47.255	NBNS	92	Name query NB SLEEP<00>
32	3.719013	172.30.32.1	172.30.47.255	NBNS	92	Name query NB SLEEP<00>
45	4.473234	172.30.32.1	172.30.47.255	NBNS	92	Name query NB SLEEP<00>
50	5.228287	172.30.32.1	172.30.47.255	NBNS	92	Name query NB SLEEP<00>
51	5.246603	172.30.32.1	172.30.47.255	NBNS	92	Name query NB SLEEP<00>
64	6.002036	172.30.32.1	172.30.47.255	NBNS	92	Name query NB SLEEP<00>
65	6.228815	172.30.32.1	172.30.47.255	NBNS	92	Name query NB INFINITY<00>
82	6.755612	172.30.32.1	172.30.47.255	NBNS	92	Name query NB SLEEP<00>
83	6.989933	172.30.32.1	172.30.47.255	NBNS	92	Name query NB INFINITY<00>
86	7.746794	172.30.32.1	172.30.47.255	NBNS	92	Name query NB INFINITY<00>
93	8.743493	172.30.32.1	172.30.47.255	NBNS	92	Name query NB INFINITY<00>
98	9.496542	172.30.32.1	172.30.47.255	NBNS	92	Name query NB INFINITY<00>
99	10.252573	172.30.32.1	172.30.47.255	NBNS	92	Name query NB INFINITY<00>
120	31.328528	172.30.32.1	172.30.47.255	NBNS	92	Name query NB SLEEP<00>
137	32.079041	172.30.32.1	172.30.47.255	NBNS	92	Name query NB SLEEP<00>
140	32.832588	172.30.32.1	172.30.47.255	NBNS	92	Name query NB SLEEP<00>
143	33.007419	172.30.32.1	172.30.47.255	NBNS	92	Name query NB SLEEP<00>
160	34.567845	172.30.32.1	172.30.47.255	NBNS	92	Name query NB SLEEP<00>
167	35.301593	172.30.32.1	172.30.47.255	NBNS	92	Name query NB INFINITY<00>
170	35.333027	172.30.32.1	172.30.47.255	NBNS	92	Name query NB SLEEP<00>
179	36.055793	172.30.32.1	172.30.47.255	NBNS	92	Name query NB INFINITY<00>
182	36.810356	172.30.32.1	172.30.47.255	NBNS	92	Name query NB INFINITY<00>
185	37.785624	172.30.32.1	172.30.47.255	NBNS	92	Name query NB INFINITY<00>
194	38.536424	172.30.32.1	172.30.47.255	NBNS	92	Name query NB INFINITY<00>
195	39.297817	172.30.32.1	172.30.47.255	NBNS	92	Name query NB INFINITY<00>
232	374.986556	172.30.32.1	172.30.47.255	NBNS	92	Name query NB SLEEP<00>
245	375.750735	172.30.32.1	172.30.47.255	NBNS	92	Name query NB SLEEP<00>
248	376.504899	172.30.32.1	172.30.47.255	NBNS	92	Name query NB SLEEP<00>
251	377.496617	172.30.32.1	172.30.47.255	NBNS	92	Name query NB SLEEP<00>
268	378.252204	172.30.32.1	172.30.47.255	NBNS	92	Name query NB SLEEP<00>
269	378.968259	172.30.32.1	172.30.47.255	NBNS	92	Name query NB INFINITY<00>
278	379.014937	172.30.32.1	172.30.47.255	NBNS	92	Name query NB SLEEP<00>
287	379.718966	172.30.32.1	172.30.47.255	NBNS	92	Name query NB INFINITY<00>
290	380.482451	172.30.32.1	172.30.47.255	NBNS	92	Name query NB INFINITY<00>
295	381.463997	172.30.32.1	172.30.47.255	NBNS	92	Name query NB INFINITY<00>
302	382.225156	172.30.32.1	172.30.47.255	NBNS	92	Name query NB INFINITY<00>
303	382.985948	172.30.32.1	172.30.47.255	NBNS	92	Name query NB INFINITY<00>

En (1) une machine avec pour adresse IPv4 « **172.30.32.1** » en un **message BroadCast** (Message à tous les appareils du sous réseau) via **l'adresse BroadCast** « **172.30.47.255** » pour obtenir des infos sur SLEEP, Pour faire simple il demande « Qui est SLEEP ? », si une machine avec ce nom est sur le sous reseaux elle repondra avec son adresse IP.

Il en est de même en (2) avec la machine INFINITY

NB : NBNS est généralement sur Port **UDP 137**

Tableau d'analyse des ports

Port	Protocole	État	Justification (Séquence observée)
80	TCP	OUVERT	SYN trame 45 → SYN-ACK trame 48
2222	TCP	OUVERT	SYN trame 52 → SYN-ACK trame 55
8080	TCP	OUVERT	SYN trame 60 → SYN-ACK trame 63
22	TCP	FERMÉ	SYN trame 120 → RST-ACK trame 121
443	TCP	FERMÉ	SYN trame 125 → RST-ACK trame 126
9999	TCP	FILTRÉ	SYN trame 200 → Pas de réponse
53	UDP	FERMÉ	UDP trame 305 → ICMP Port Unreachable tram
128	UDP	FERMÉ	UDP trame 305 → ICMP Port Unreachable tram
1900	SSDP	FERME	Absence de NOTIFY

Nettoyage de l'environnement

Étape 1 : Arrêt des conteneurs

```
bash
```

```
docker compose down
```

Étape 2 (optionnel) : Nettoyage complet

```
bash
```

```
docker compose down -v
```

```
C:\Users\Junio\Desktop\TP1\tp1-cyber-lab>docker compose down -v
[+] Running 4/4
✔Container tp1_attacker      Removed    0.1s
✔Container tp1_web          Removed    0.9s
✔Container tp1_target       Removed    1.5s
✔Network tp1-cyber-lab_labnet Removed    0.4s
Activate Windows
Go to Settings to activate Windows.
```

Checklist de fin de TP

Ce que nous devons avoir :

- **Capture baseline** (baseline.pcapng)
- **Capture scan** (scan.pcapng)
- **Analyses Wireshark** complétées
- **Tableau des ports** avec états déduits
- **Notes** sur les séquences TCP,UDP,... observées

 Aspect éthique - RAPPEL IMPORTANT

-  Toutes les manipulations sont restées **dans le lab fourni**
-  **Aucun test** en dehors du périmètre autorisé
-  Environnement maintenant **nettoyé**