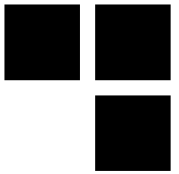
SYNACKTIV
DIGITAL SECURITY



UDP Just Opened

Quand Synacktiv devient le 5e opérateur

Présenté 16/06/2016

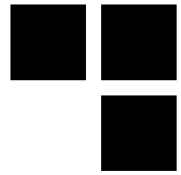
Pour BeeRumP 2016

Par Renaud Dubourguais

Situation initiale

- **Besoin d'Internet**
  - Mais
    - → Je ne suis pas chez moi
    - → Je n'ai pas mon téléphone sur moi
    - → Aucun Wi-Fi à l'horizon
  - Heureusement
    - → J'ai un PC portable
    - → Un dongle 3G
    - → Une carte SIM prépayée d'un opérateur bien connu
  - Mais malheur, je n'ai plus de forfait :(

Yolo, ça va passer !



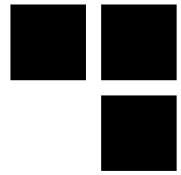
- **Montons la *data* et voyons ce qui se passe**

```
# wvdial --config /etc/wvdial.conf myoperator
[…]
--> Pid of pppd: 30747
--> Using interface ppp0
--> local  IP address 10.153.80.130
--> remote IP address 10.64.64.64
--> primary   DNS address 192.168.10.110
```

  - Étonnamment cela semble fonctionner

    → J’ai un DNS et une passerelle par défaut

Yolo, ça va passer !



- **Tentons un accès web pour confirmer**

```
$ GET -se http://www.google.com
200 OK
Connection: close
Server: Apache
Vary: Accept-Encoding
Content-Length: 0
Content-Type: text/html;charset=ISO-8859-1
Set-Cookie: operatorcookie=1015b9671f4e8b20928e5414f58b1ee7;
path=/; domain=.myoperator.fr
```

  - Bon ce n'est pas aussi simple, l'opérateur fait son travail et met en place un portail captif

Yolo, ça va passer !

- **Et si ce n'est pas du HTTP ?**

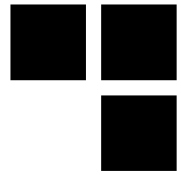
```
$ nc www.synacktiv.com 80
test
```

```
10.153…  62.210.…  TCP      76 46752 → 80 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=275427678 TSecr=0 WS=128
62.210…  10.153.…  TCP      80 80 → 46752 [SYN, ACK] Seq=0 Ack=1 Win=22400 Len=0 MSS=1400 WS=8 TSval=736301663 TSecr=2754…
10.153…  62.210.…  TCP      68 46752 → 80 [ACK] Seq=1 Ack=1 Win=29312 Len=0 TSval=275427804 TSecr=736301663
10.153…  62.210.…  TCP      73 46752 → 80 [PSH, ACK] Seq=1 Ack=1 Win=29312 Len=5 TSval=275429321 TSecr=736301663
10.153…  62.210.…  TCP      73 [TCP Retransmission] 46752 → 80 [PSH, ACK] Seq=1 Ack=1 Win=29312 Len=5 TSval=275429573 TSe…
10.153…  62.210.…  TCP      73 [TCP Retransmission] 46752 → 80 [PSH, ACK] Seq=1 Ack=1 Win=29312 Len=5 TSval=275429952 TSe…
10.153…  62.210.…  TCP      73 [TCP Retransmission] 46752 → 80 [PSH, ACK] Seq=1 Ack=1 Win=29312 Len=5 TSval=275430710 TSe…
```

Analyse post-yolo

- **Si trafic HTTP**
  - Portail captif et redirection vers une page « garage »
- **Sinon, filtrage douteux**
  - Tous les paquets PSH sont filtrés
  - Mais le handshake TCP est bien réalisé

TFO, épisode II



- **TFO → cf. rump Bee^CSSTIC 2014**
  - Ajoutons de la donnée dans le paquet SYN !
    - → Le kernel peut basculer en non-TFO sans vous notifier
    - → Peut amener quelques faux-positifs dans l'analyse
    - → Une approche manuelle avec Scapy est préférable
  - Après un *training* Scapy et une *pull request*
    - → Un SYN TFO sans donnée, ça passe
    - → Un SYN TFO avec donnée, ça ne passe pas

Analyse post-TFO

- **Un filtrage pas si douteux finalement**
  - Filtrage si le paquet TCP contient de la donnée

    → Quels que soient les *flags* TCP
  - Dommage, pas de rump sur *tfo2tcp*...

Et UDP ?

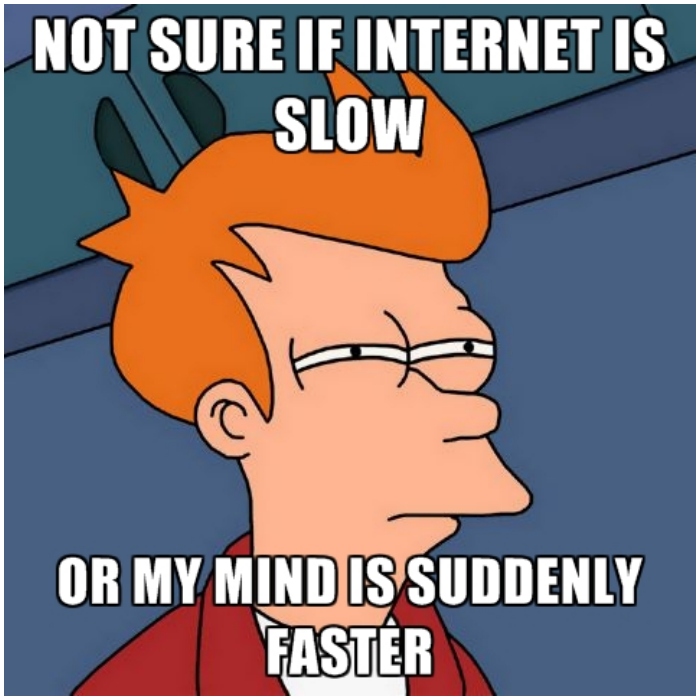
- **Même constat**
  - Filtrage dès que de la donnée est présente

```
# echo 'hello from client!' | nc -u <ip> 8080
(je ne reçois rien)
```

```
# echo 'hello from server!' | nc -uvnlp 8080
(je ne reçois rien)
```

Et dns2tcp ?

- **J’ai un serveur DNS mais je suis en 3G…**
  - Connexion très instable
  - Une latence parfois > 500ms



Scapy à la rescousse

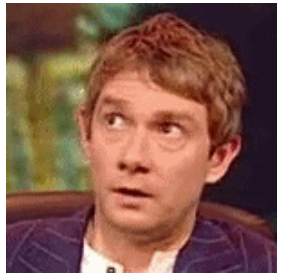
- **La magie noire de Scapy**
  - Côté client, j'envoi un paquet UDP standard

```
>>> sr(IP(dst="<ip>")/UDP(dport=8080)/'data')
```

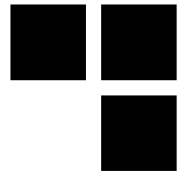
  - Côté serveur, je reçois la donnée ???

```
# nc -uvlnp 8080
data
```

  - Scapy contournerait donc le filtrage ?



Analyse post-Scapy



- **Utiliser Scapy c'est bien, le comprendre c'est mieux**
  - Si envoi d'un paquet TCP sans port source

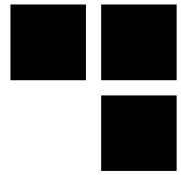
    → Utilisation du port source 20 par défaut

    → Mais **ne passe pas** le filtrage
  - Si envoi d'un paquet UDP sans port source

    → Utilisation du port source 53 par défaut

    → **Cette fois le paquet passe le filtrage**
  - Donc, pas de filtrage si le port source est 53 ?

Analyse post-Scapy



- Validons avec un *netcat*

```
# echo 'hello from client!' | nc -s 10.153.80.130 -p 53 -u <ip> 8080
hello from server!
```

```
# echo 'hello from server!' | nc -uvnlp 8080
hello from client!
```

À moi l'Internet convivial « gratuit »

- **Utilisons OpenVPN**

```
# openvpn --local 10.153.80.130 --lport 53 --config myvpn.conf
```

  - Et voilà, j'ai Internet illimité sans avoir de forfait
  - Latence équivalente à une latence 3G classique
  - Du coup, si on montait un opérateur du pauvre ?

Opérateur du pauvre / rentabilité

- **Business finalement pas si rentable**
  - Ne marche que chez un seul opérateur du marché
    → Peu probable que l'affaire soit pérenne
  - Nécessite d'acheter tous les 6 mois du forfait pour maintenir le numéro actif
    → Soit 10€ minimum tous les 6 mois / SIM
    → Soit un prix de vente de 1,6€ / mois / SIM + marges
    → Ça ne vous rappelle rien ?

Opérateur du pauvre / rentabilité

Possible en jouant sur les marges ou en obtenant un rabais sur les SIM

Utilise probablement le forfait de la carte prépayée renouvelée tous les 6 mois. Quand plus de forfait sur la carte, plus de service… it sounds familiar?

Mieux vaut brider à 50Mo pour éviter de se faire détecter par l'opérateur...

AVEZ-VOUS DES QUESTIONS ?

MERCI DE VOTRE ATTENTION

SYNACKTIV
DIGITAL SECURITY