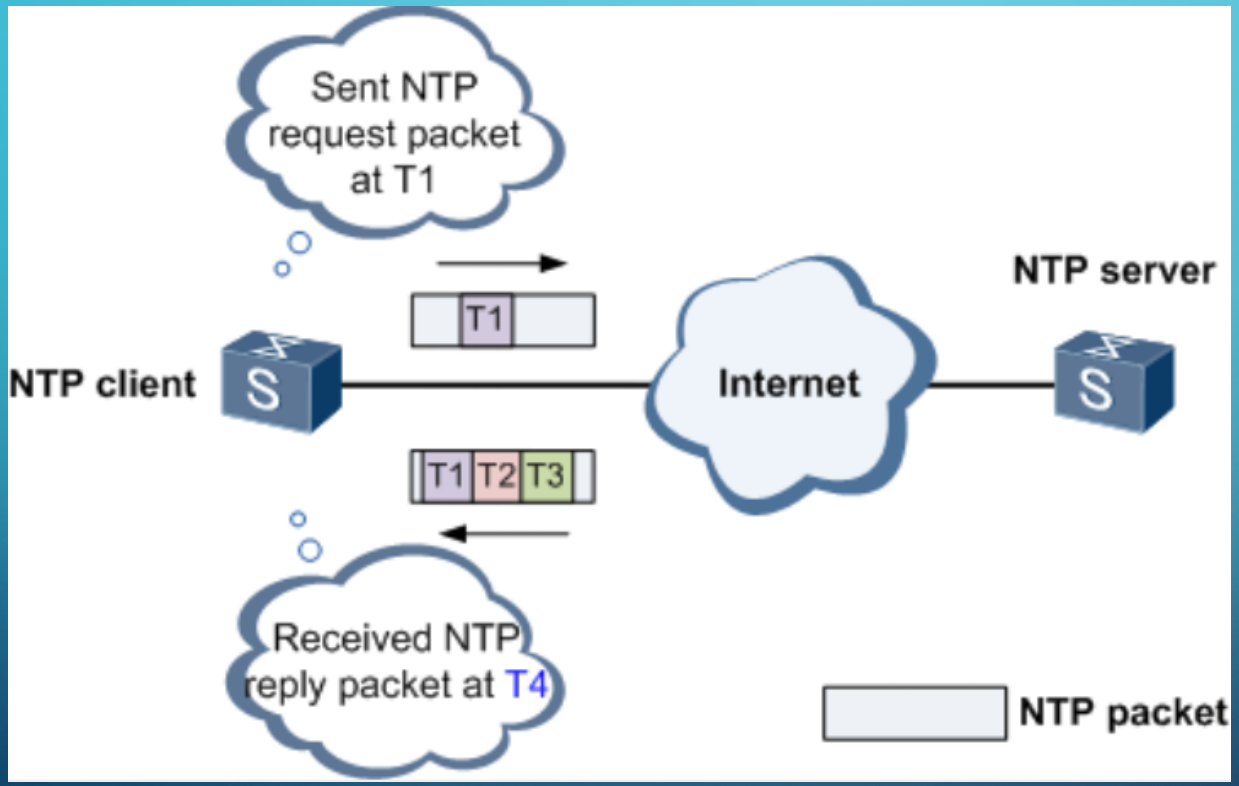
HACKATHON@AIS2018

NETWORK TIME PROTOCOL DRAFT

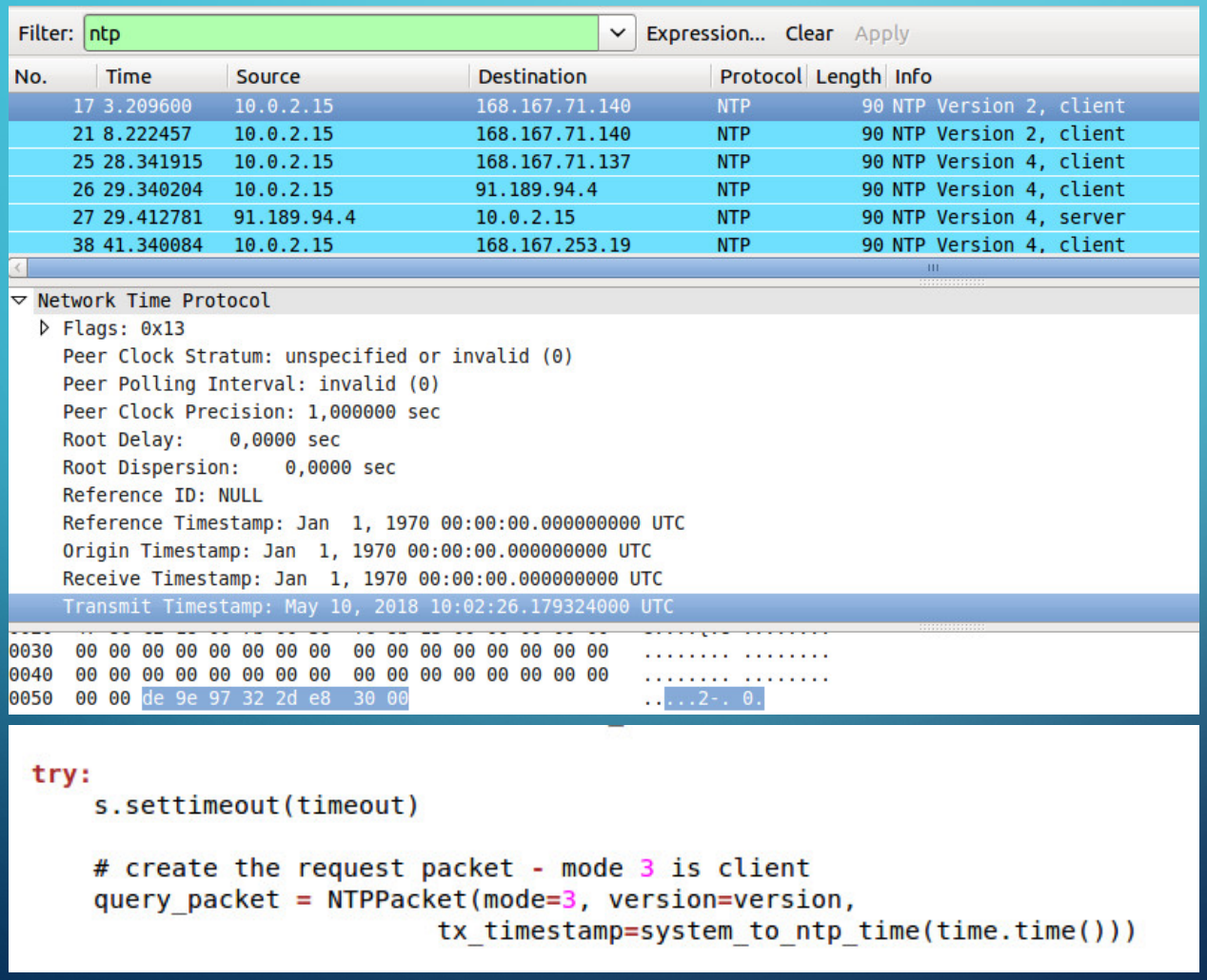
Présenté par:
Christine BADJI
Ndèye Penda FALL
Fatimata KONE

1. PRINCIPE DE NTP



Le client envoie une requête avec un horodatage (T1) et le serveur répond avec 3 horodatages que sont celui du client (T1), celui reçu par le serveur (T2) et la réponse d'horodatage envoyé par le serveur

2. PROBLEMATIQUE




```
try:
    s.settimeout(timeout)

    # create the request packet - mode 3 is client
    query_packet = NTPPacket(mode=3, version=version,
                             tx_timestamp=system_to_ntp_time(time.time()))
```

- A partir de la réponse du serveur, on peut connaitre l'horodatage du client et ainsi le traquer.

3. SOLUTION PROPOSÉE

- **Randomisation de l'horodatage à transmettre**

RFC 5905

> 4.2. Transmit Timestamp Randomization
>
> While this memo calls for most fields in client packets to be set to zero, the transmit timestamp is randomized. This decision is motivated by security as well as privacy.

4. TRAVAIL EFFECTUÉ

- Python: https://github.com/Tipoca/ntplib/

```
root@NTP4AFRICA:~/ntplib# ls
CHANGELOG  COPYING.LESSER  ntplib.py.mv  ntplib.pyc  README.md  test_ntplib.p
COPYING    MANIFEST        ntplib.py     PKG-INFO    setup.py
```

ntplib.py

```
 import time
+import random


 class NTPException(Exception):
@@ -270,10 +271,10 @@ class NTPClient:

         try:
             s.settimeout(timeout)
-
+            aleat = random.random()+ time.time()
             # create the request packet - mode 3 is client
             query_packet = NTPPacket(mode=3, version=version,
-                                     tx_timestamp=system_to_ntp_time(time.time()))
+                                     tx_timestamp=system_to_ntp_time(aleat))

             # send the request
             s.sendto(query_packet.to_data(), sockaddr)
diff --git a/test_ntplib.py b/test_ntplib.py
index 3e559af..5943d0c 100755
```

5. RÉSULTAT

```
root@NTP4AFRICA:~/ntplib# python test_ntplib.py
...
----------------------------------------------------------------------
Ran 3 tests in 8.405s

OK
root@NTP4AFRICA:~/ntplib#
```

No.	Time	Source	Destination	Protocol	Length	Info
4095	34.192606	196.200.214.51	168.167.168.34	NTP	90	NTP Version 2, client
4096	34.192700	196.200.214.51	168.167.168.34	NTP	90	NTP Version 2, client
4097	34.192744	196.200.214.51	168.167.168.34	NTP	90	NTP Version 2, client
4098	34.192784	196.200.214.51	168.167.168.34	NTP	90	NTP Version 2, client
4099	34.192992	196.200.214.51	168.167.168.34	NTP	90	NTP Version 2, client
4100	34.193037	196.200.214.51	168.167.168.34	NTP	90	NTP Version 2, client
4101	34.193076	196.200.214.51	168.167.168.34	NTP	90	NTP Version 2, client
4102	34.193117	196.200.214.51	168.167.168.34	NTP	90	NTP Version 2, client
4137	34.447033	168.167.168.34	196.200.214.51	NTP	90	NTP Version 2, server
4138	34.447034	168.167.168.34	196.200.214.51	NTP	90	NTP Version 2, server
4139	34.447035	168.167.168.34	196.200.214.51	NTP	90	NTP Version 2, server

```
Network Time Protocol (NTP Version 2, client)
    Flags: 0x13, Leap Indicator: no warning, Version number: NTP Version 2, Mode: client
    Peer Clock Stratum: unspecified or invalid (0)
    Peer Polling Interval: invalid (0)
    Peer Clock Precision: 1.000000 sec
    Root Delay: 0 seconds
    Root Dispersion: 0 seconds
    Reference ID: NULL
    Reference Timestamp: Jan  1, 1970 00:00:00.000000000 UTC
    Origin Timestamp: Jan  1, 1970 00:00:00.000000000 UTC
    Receive Timestamp: Jan  1, 1970 00:00:00.000000000 UTC
    Transmit Timestamp: May 10, 2018 13:42:32.177305221 UTC
```

L'horodatage ainsi modifié, il devient donc difficile de traquer le client à partir du transmit timestamp

MERCI POUR VOTRE ATTENTION