L'outil Nmap-Stateful

Olivier Courtay

Thomson R&D / IRISA

Plan

- Introduction
- Nmap-Stateful: Principe et Fonctionnement
- Méthodes et Résultats
- Prospective
- Conclusion

Nmap

- Outil réseau classique
- Scanner de ports avancé
- Détection de machines
- Détection des services

- **Détection des systèmes d'exploitation (OS)**
  - Caractéristique TCP/IP
  - Base de signatures (> 700)

Nmap – Détection d'OS

Port ouvert

ISN (séquence TCP)

IP ID (ouvert)

TCP SYN + options

Port fermé

IP ID (fermé)

TCP SYN, TCP ACK

UDP (réponse ICMP)

TCP ACK

TCP S/F/P/U

TCP NULL

TCP Xmas (F/P/U)

Paquets non-standards

Limites de Nmap

- En environnement filtré
  - UDP passe rarement
  - Pas de port fermé
  - Contrôle des flags par les firewalls stateful

→*Peu de tests fonctionnent dans ces conditions*

- Tests non configurables
- Seulement deux états TCP utilisés

Nmap-Stateful

Principe et Fonctionnement

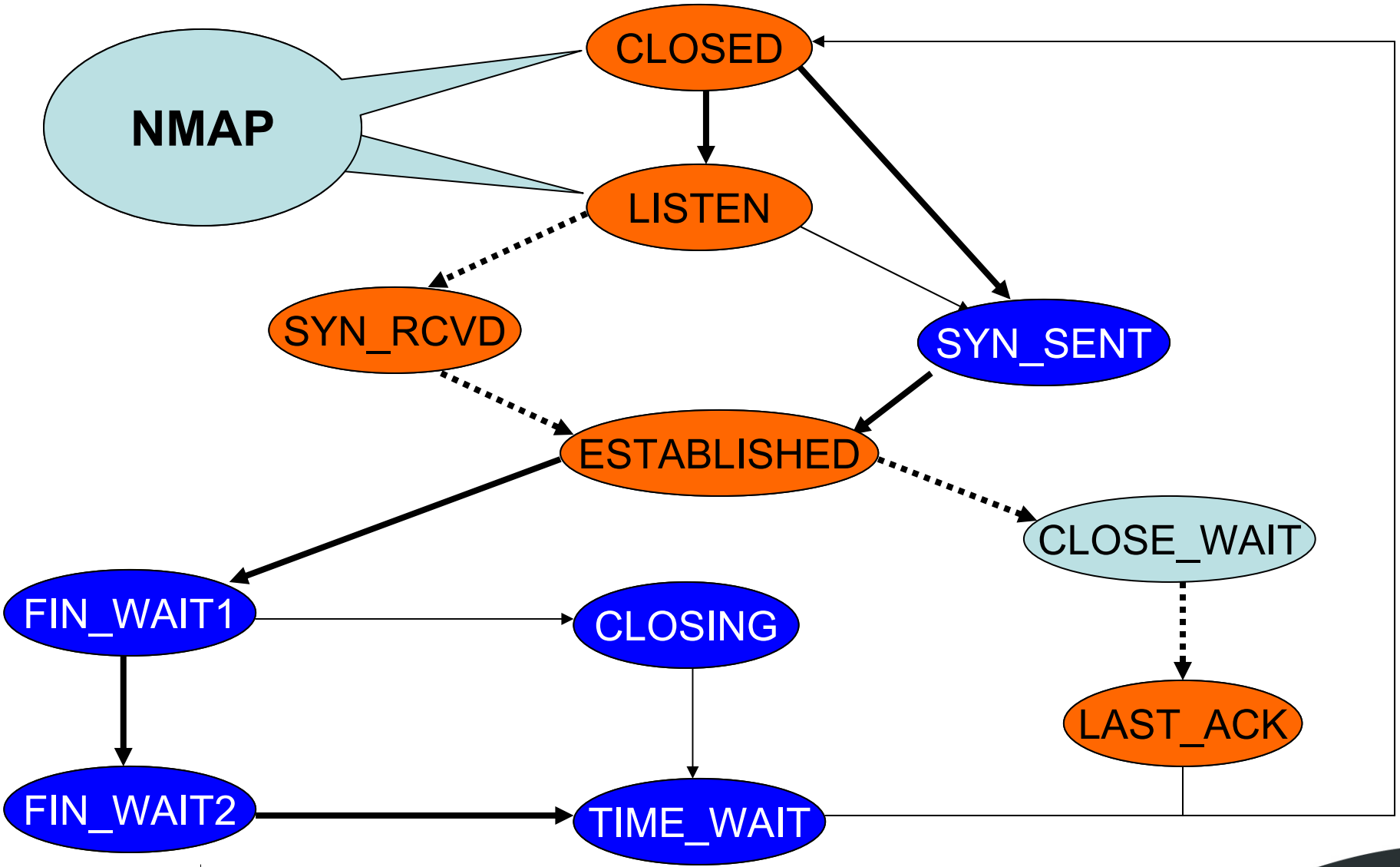
Principe

- Extension du code de Nmap

- Teste plusieurs états TCP (Stateful)
  1) La machine testée est amenée dans l'état voulu
  2) Le test est ensuite lancé
  3) La réponse est analysée

- Tests configurables par l'utilisateur

Le diagramme d'état TCP



Trace d'exécution

```
#nmap-stateful --otf test-estab-SYN -p 22 192.168.1.1
SYN sent for test ESTAB_SYN from port 8557 to port 22
...
SYN_SENT  seq:34 sp:8557 -> dp:22 ack:00 SYN_RECV flags:S
ESTABLISH ack:35 dp:8557 <- sp:22 seq:78 SYN_RECV flags:SA
ESTABLISH seq:35 sp:8557 -> dp:22 ack:79 ESTABLISH flags:A
Launch test: ESTAB_SYN
UNKONWN   seq:35 sp:8557 -> dp:22 ack:79 UNKONWN flags:S
UNKONWN   ack:42 dp:8557 <- sp:22 seq:00 UNKONWN flags:RA
...
Fingerprint:
ESTAB_SYN(Resp=Y%DF=Y%W=0%ACK=O%Flags=AR%Ops=)
```

Implémentation

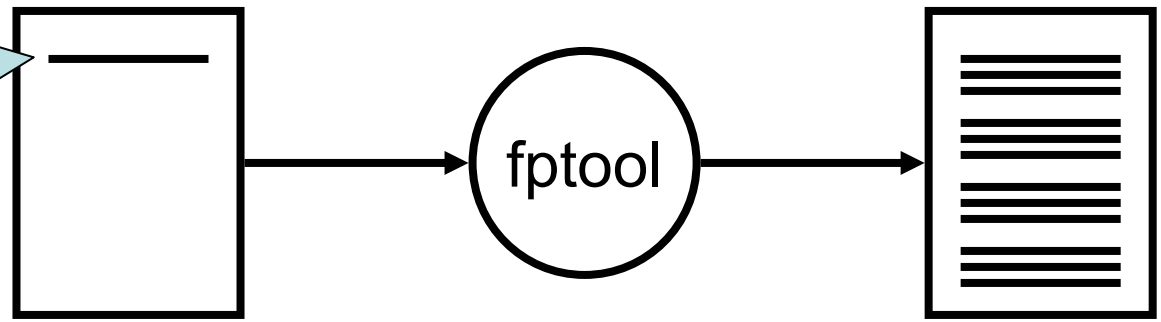
- Licence GPL
- Mini pile TCP/IP
- Firewall
  - Inhibe la réaction de la machine testeur
- Linux supporté
  - Utilisation de Iptables
- Aide à la création de tests

Méthodes et Résultats

Détection d'OS

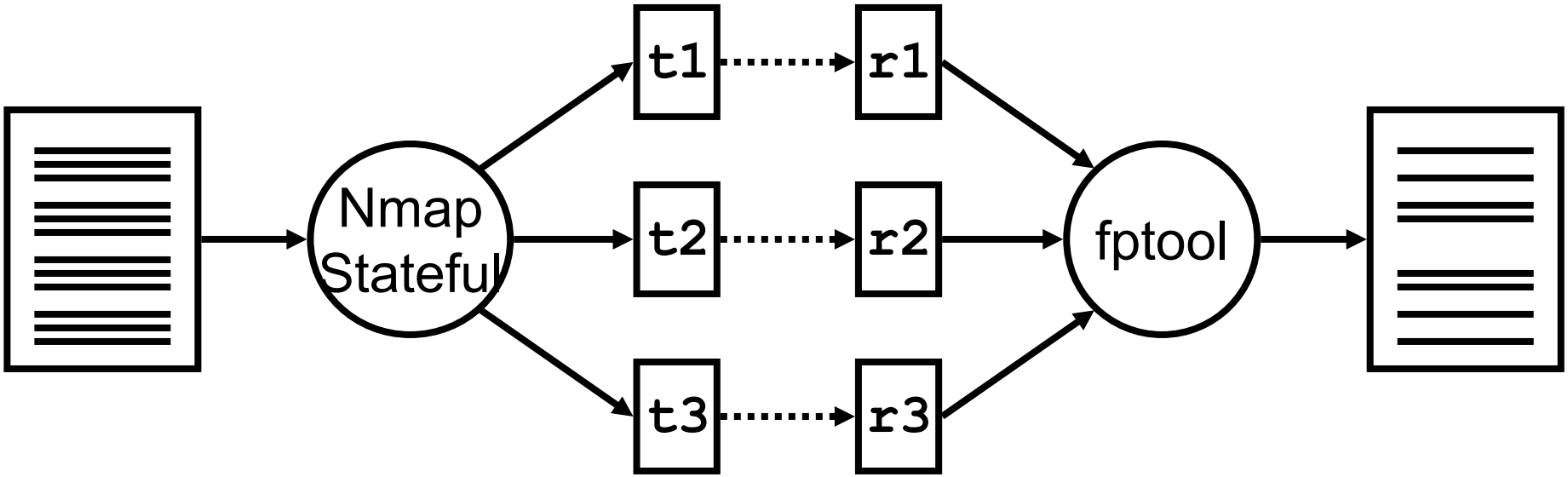
Génération des tests

```
NAME=template
   ESTABLISHED
   TH_SYN
   TH_ACK
   TH_FIN
   TH_PUSH
   DATA=foobar
   DATALEN=7
   SEQ=1
END
```




```
#fingerprinttool -g template -o test
```

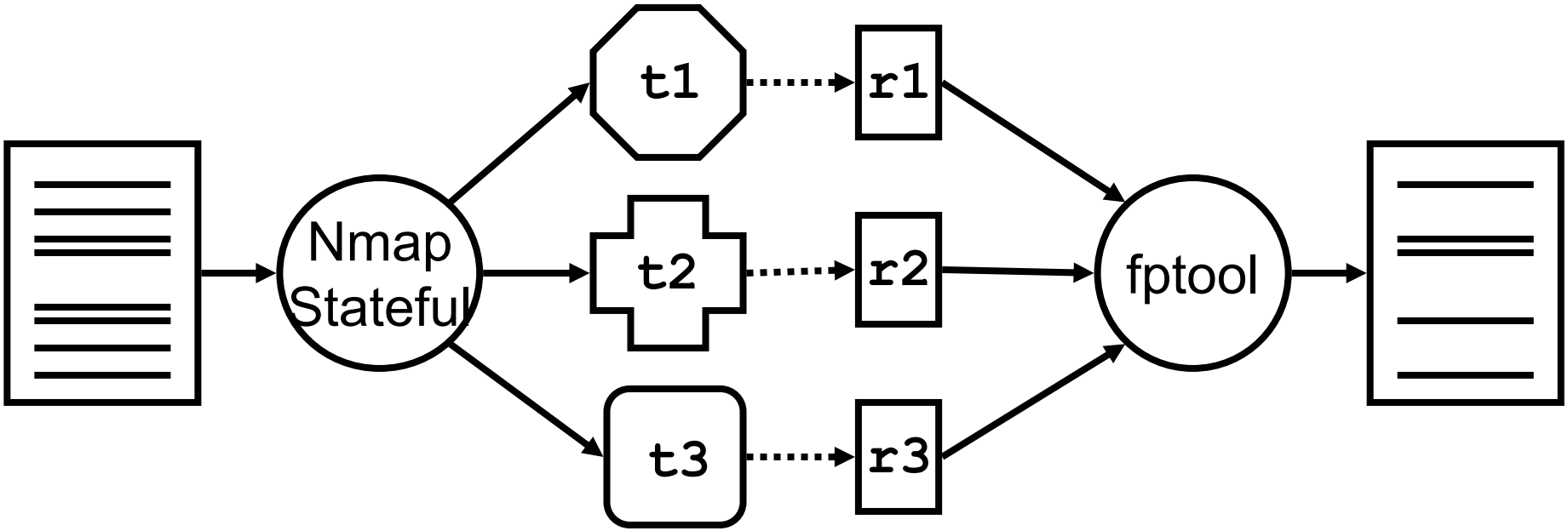
Sélection des tests stables




```
#nmap-stateful -p 22 --otf test --orf r1  t1
#nmap-stateful -p 22 --otf test --orf r2  t2
#nmap-stateful -p 22 --otf test --orf r3  t3

#fingerprinttool -s -t test -o stable r1 r2 r3
```

Sélection des tests pertinents




```
#nmap-stateful -p 22 --otf stable --orf r1  t1
#nmap-stateful -p 22 --otf stable --orf r2  t2
#nmap-stateful -p 22 --otf stable --orf r3  t3

#fingerprinttool -s -t stable -o good r1 r2 r3
```

Validation de l'outil

- Test sur un Linux 2.4

```
#nmap-stateful --otf good --off signatures -p 80  x.x.x.x
Interesting ports on x.x.x.x:
PORT         STATE        SERVICE
80/tcp       open         http
OS details: Linux 2.4
```

- Autre test sur un Linux 2.4

```
#nmap-stateful --otf good --off signatures -p 80  y.y.y.y
Interesting ports on y.y.y.y:
PORT         STATE        SERVICE
80/tcp       open         http
Aggressive OS guesses: Linux 2.6 (97%), Linux 2.4 (96%)
```

Prospective

Actions des Firewalls (1/3)

- Ils bloquent certains paquets
- Test sur un Solaris 9 protégé

```
#nmap-stateful --otf good --off signatures -p 80  x.x.x.x
Interesting ports on x.x.x.x:
PORT         STATE        SERVICE
80/tcp       open         http
No OS match
```

➔ Échec !

Actions des Firewalls (2/3)

- Méthode pour en tirer avantage
  - Construire des tests non-sensibles aux Firewalls
  - Découvrir l'OS de la machine protégée
  - Construire des tests **sensibles** aux Firewalls
  - Caractériser le Firewall qui protége la machine

Actions des Firewalls (3/3)

- Test sur un Solaris 9 protégé par un Firewall

```
ESTABLI_AP_SEQ-2(Resp=Y%DF=Y%W=832C%ACK=0%Flags=A%Ops=)
ESTABLI_AP_SEQ-1(Resp=Y%DF=Y%W=832C%ACK=0%Flags=A%Ops=)
ESTABLI_AP_SEQ0(Resp=Y%DF=Y%W=832C%ACK=0%Flags=A%Ops=)
ESTABLI_AP_SEQ1(Resp=Y%DF=Y%W=832C%ACK=0%Flags=A%Ops=)
ESTABLI_AP_SEQ2(Resp=Y%DF=Y%W=832C%ACK=0%Flags=A%Ops=)
```

- Test sur un Solaris 9 protégé par un autre type de Firewall

```
ESTABLI_AP_SEQ-2(Resp=N)
ESTABLI_AP_SEQ-1(Resp=N)
ESTABLI_AP_SEQ0(Resp=Y%DF=Y%W=8325%ACK=0%Flags=A%Ops=)
ESTABLI_AP_SEQ1(Resp=N)
ESTABLI_AP_SEQ2(Resp=N)
```

Améliorations (1/2)

- Mini-pile TCP/IP
- Portabilité
  - Unix → Libdnet (Dug Song)
  - Windows → PktFilter (HSC)
- Expressivité des tests

Améliorations (2/2)

- Approche combinatoire
  - Plus systématique
  - Sur un panel complet de configurations (OS / FW)

→ Fournir un jeu de tests
  - Robuste
  - Complet

Conclusion

- Les premiers résultats sont encourageants

- Problème de traitement de données
- L'aide de la communauté est utile
  - Proposer / Tester de nouveaux jeux de tests
  - Proposer des améliorations de l'outil

Des Questions ?

http://home.gna.org/nmapstateful