coupefeu à états

- termes équivalents: coupefeu dynamique, à états, par suivi de connexion, « Statefull Packet Inspection»
- enrichit le filtrage des paquets par la mémorisation de l'état des sessions,d'échanges de données en fonction des paquets déjà vus
- analyse s'appuyant sur l'historique des sessions
- session
  - naturel avec tcp
  - la connaissance des couches réseau, transport, voire application permet d'en gérer avec udp et icmp

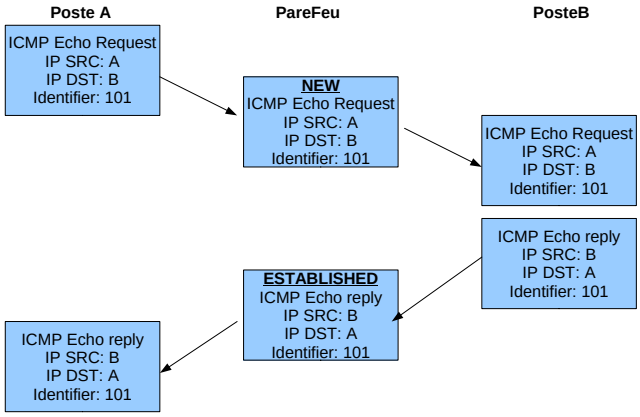
parefeu à état: état d'une session

- avec le parefeu NetFilter (Linux 2.4+), un paquet faisant partie d'une session peut être l'un des 4 états suivants :
  - New: ne correspond à aucune entrée de la table des états. Création d'une nouvelle entrée
  - Established: le paquet fait partie d'une connexion existante (entrée existante dans la table des états)
  - Related: le paquet fait partie d'une nouvelle connexion faisant partie d'une session existante.
  - Invalid: paquet dont l'état n'a pu être déterminé
- il y a des états internes plus détaillés accessibles par « cat /proc/net/ip_conntrack »

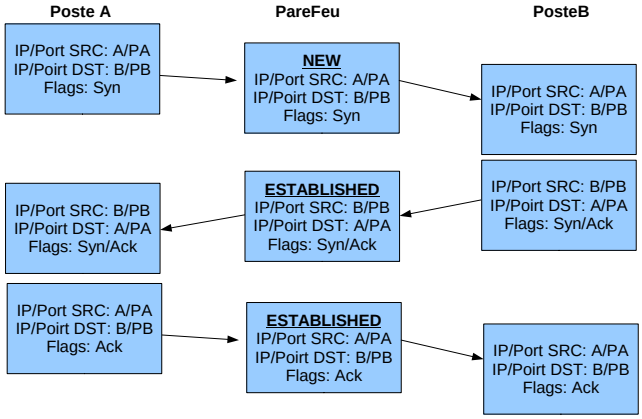
pare feu à état: états d'une session

- Attention: c'est l'étude de l'historique des paquets qui permet de déterminer l'état, pas les FLAGS TCP
  - les états fournissent « seulement » des critères supplémentaires pour le filtrage:
- l'utilisation dépend du logiciel firewall:
  - NETFILTER (linux 2.4+):
    - autoriser les paquets TCP SYN sortant
    - autoriser les paquets TCP et ICMP entrants dont l'état est RELATED ou ESTABLISED
    - interdire les paquets TCP NEW sans flag SYN
  - IPFilter (FreeBSD, Solaris 10, ...), pf (OpenBSD, FreeBSD, ...):
    - autoriser les paquets TCP SYN sortant et tous les paquets suivants de la session seront automatiquement acceptés

exemple de sessions: icmp echo



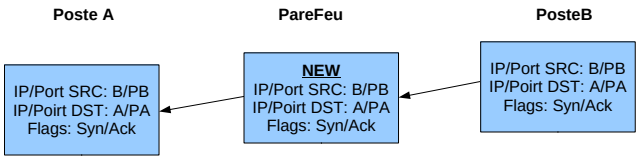
exemple de sessions: tcp



exemple de session: tcp

- règles associées pour autoriser un accès sortant au WeB
  - autoriser les paquets TCP sortant NEW vers le port http ou https
  - autoriser les paquets TCP entrant ESTABLISHED
  - autoriser les paquets icmp RELATED entrant
  - refuser le reste
- ràf: animation pour illustrer une connexion sortante et une connexion entrante venant du port 80 d'une machine inconnue

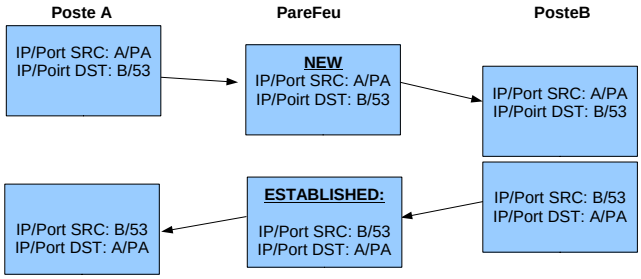
exemple de sessions: tcp particularité de netfilter



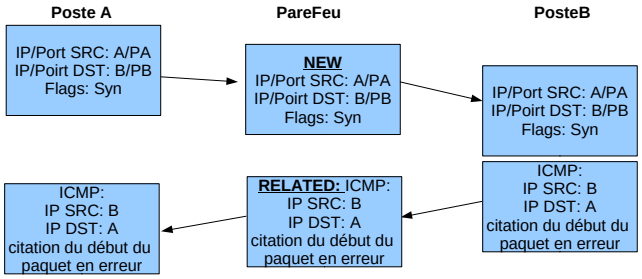
le premier paquet vu sera considéré comme NEW même s'il est incorrect comme premier paquet du point de vue tcp.
Dans l'exemple, ce premier paquet est un segment d'acquittement (alors qu'un premier paquet devrait être un SYN)
cet exemple illustre deux points :

- avec NetFilter, les états sont un critères utilisable supplémentaire qu'il faut croiser avec les autres critères pour en faire ce que l'on veut
- Application : l'une des règles usuelles utilisées avec netfilter consiste à filtrer les paquets TCP NEW sans flag SYN.

exemple de session: udp (dns)



exemple de session: tcp/icmp(host unreachable)



Suivi de fenêtre TCP

- Problème :
  - filtrer les paquets incorrects
    - pour éviter la fuite d'information
    - pour éviter certaines attaques liées à la façon dont ces paquets incorrects vont être gérés par la machine cible
  - un segment peut être incorrect si ses No de séquence sont incohérent par rapport à ceux de la connexion en cours
  - ràf: un dessin qui illustre la chose (fait au tableau)

Suivi de fenêtre TCP

- le coupe feu doit prendre des décisions à partir des informations qu'il a :
  - ce qui passe par le FW est un sous ensemble de ce qui est émis par A (pertes ou retards possibles entre A et FW)
  - ce qui arrive en B est un sous ensemble de ce qui est passe par le FW (pertes ou retards possibles entre FW et B)
- ne pas en tenir compte, c'est refuser des paquets réémis suite à des pertes

Exemples concrets (1)

S-D	contenu	No pas-sage FW
B-A	win 2048 ack 1	1
A-B	1:1000	2
B-A	win 1048 ack 1001	3
A-B	1001:2000	4
B-A	win 2048 ack 2001	5
A-B	2001-3000	6
B-A	win 2048 ack 3001	7

1-cas standard sans perte

S-D	contenu	No pas-sage FW	
B-A	win 2048 ack 1	1	
A-B	1:1000	2	
B-A	win 1048 ack 1001	3	
A-B	1001:2000	4	
B-A	**win 2048 ack 2001**	**5**	**perdu**
A-B	1001:2000	6	
B-A	win 2048 ack 2001	7	
A-B	2001-3000	8	
B-A	win 2048 ack 3001	7	

2-cas d'un paquet perdu en FW et A

3-cas d'un paquet retardé

S-D	contenu	No pas-sage FW
B-A	win 2048 ack 1	1
A-B	1:1000	2
A-B	1001-2000	3
B-A	win 2048 ack 2001	4
A-B	2001-3000	5
B-A	win 2048 ack 3001	6
B-A	**win 1048 ack 1001**	**7**

developper au tableau les cas 1 et 2 pour rappeler le mécanisme de fenêtre tcp

bornes sup des numeros de seq/ack (IPFilter 4)

- A envoie un paquet à B contenant l'intervalle de données [s, s+n[
- borne supérieurs des données envoyées par A :
  - notation: B-A/C: paquet de B vers A vu en C

dernier octet envoyé ≤ octet max que A peut envoyer
équivaut à :
$s+n \leq \text{octet max} + 1 \leq max_{B-A/A}(ack+win)$
cas particulier : fenêtre nulle (tampon de B plein).
A envoie des paquets (x) pour tester la fenêtre de B.
(x) : en général de 1 octet
$s+n \leq max_{B-A/A}(ack+max(1,win))$
$\boldsymbol{s+n \leq max_{B-A/FW}(ack+max(1,win))}$

bornes inf des numeros de seq/ack (IPFilter 4)

$max(ack) \leq s(1)$
nous savons que :
$s+n \leq max_{B-A/A}(ack+max(1,win))$
$s+n \leq max_{B-A/A}(ack)+max_{B-A/A}(max(1,win))$
$s+n-max_{B-A/A}(1,win) \leq max_{B-A/A}(ack)$
en prenant ≤ max du tout par rapport au paquet envoyés par A :
$max_{A-B/A}(s+n)-max_{B-A/A}(1,win) \leq max_{B-A/A}(ack)$
soit en appliquant (1) :

$\boldsymbol{max_{A-B/A}(s+n)-max_{B-A/A}(1,win) \leq s}$

car
$max_{A-B/FW}(s+n) \leq max_{A-B/A}(s+n)$
et $max_{B-A/A}(1,win) \leq max_{B-A/FW}(1,win)$

ràf:

- donner un exemple où les bornes sont atteintes pour montrer qu'elles sont optimales
- fait au tableau

Exemples concrets (1)

S-D	contenu	No pas-sage FW
B-A	win 2048 ack 1	1
A-B	1:1000	2
B-A	win 1048 ack 1001	3
A-B	1001:2000	4
B-A	win 2048 ack 2001	5
A-B	2001-3000	6
B-A	win 2048 ack 3001	7

1-cas standard sans perte

S-D	contenu	No pas-sage FW	
B-A	win 2048 ack 1	1	
A-B	1:1000	2	
B-A	win 1048 ack 1001	3	
A-B	1001:2000	4	
B-A	**win 2048 ack 2001**	**5**	**perdu**
A-B	1001:2000	6	
B-A	win 2048 ack 2001	7	
A-B	2001-3000	8	
B-A	win 2048 ack 3001	7	

2-cas d'un paquet perdu en FW et A

3-cas d'un paquet retardé

S-D	contenu	No pas-sage FW
B-A	win 2048 ack 1	1
A-B	1:1000	2
A-B	1001-2000	3
B-A	win 2048 ack 2001	4
A-B	2001-3000	5
B-A	win 2048 ack 3001	6
B-A	**win 1048 ack 1001**	**7**

developper au tableau les cas 1 et 2 pour rappeler le mécanisme de fenêtre tcp

Exemples concrets

	A->B	A->B	B->A	B->A		**B-/FW**	A-/FW	B-/FW	
S-D	seq	octet fin	win	ack	No pas-sage FW	**max (ack + max (win, 1))**	max(s + n)	max(max(win,1))	**borne inf s**
B-A			2048	1	1	**2049**		2048	
A-B	1	1000			2	**2049**	1001	2048	**-1047**
A-B	1001	2000			3	**2049**	2001	2048	**-47**
B-A			2048	2001	4	**4049**	2001	2048	**-47**
A-B	2001	3000			5	**4049**	3001	2048	**953**
B-A			2048	3001	6	**5049**	3001	2048	**953**

ràf: faire un exemple normal mais long

Exemples concrets

S-D	contenu	No pas-sage FW	max (ack + max (win, 1))	max(s + n)	max(max(win,1))	borne inf s
B-A	win 2048 ack 1	1	-	-	-	
A-B	1:1000	2	2049	-	2048	-
B-A	win 1048 ack 1001	3	2049	1001	2048	-1047
A-B	1001:2000	4	2049	1001	2048	-1047
B-A	**win 2048 ack 2001**	**5**	2049	2001	2048	-47
A-B	1001:2000	6	4049	2001	2048	-47
B-A	win 2048 ack 2001	7	4049	2001	2048	-47
A-B	2001-3000	8	4049	2001	2048	-47
B-A	win 2048 ack 3001	7	4049	3001	2048	953

paquet No 8: si A se permet d'envoyer jusqu'à l'octet 3000, c'est qu'il a reçu un ack le lui permettant (rappel: la fenêtre de B est <= 2048).
Logiquement, la borne inf en tient compte.

Exemples concrets

	A->B	A->B	B->A	B->A		**B-/FW**	A-/FW	B-/FW	
S-D	seq	octet fin	win	ack	No pas-sage FW	**max (ack + max (win, 1))**	max(s + n)	max(ma x(win,1))	**borne inf s**
B-A			2048	1	1	**2049**		2048	
A-B	1	1000			2	**2049**	1001	2048	**-1047**
A-B	1001	2000			3	**2049**	2001	2048	**-47**
B-A			2048	2001	4	**4049**	2001	2048	**-47**
A-B	2001	3000			5	**4049**	3001	2048	**953**
B-A			2048	3001	6	**5049**	3001	2048	**953**

Limitation des pares-feux

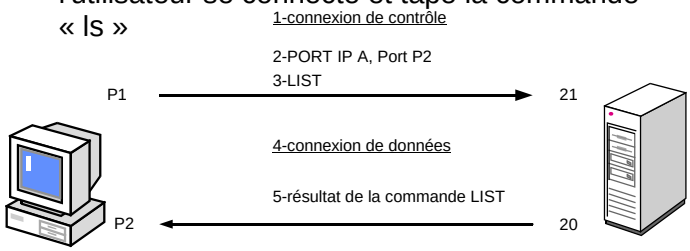
- but d'un pare feu:
  - protéger des machines internes
  - interdire les sorties/entrées d'information (plus dur)
- pare feu sans état:
  - soit on ouvre trop peu, soit on ouvre trop (ex.: connexion WeB qui ouvre tout en entrée depuis un port 80 distant)
- gestion de la fragmentation en particulier et de la normalisation de paquets en général:
  - attaque: fragmenter pour diminuer les possibilités d'identification de charge malicieuse
  - attaque: mécanisme de recouvrement de fragment

limitation des pares feux à état

- qualité du suivi de session: icmp, fenêtre tcp, ...
- analyse du niveau application souvent nécesaire (ftp, H323, ...)=> module d'analyse spécifique au protocole (ALG de la RFC 2663 ou 2993)
- insuffisant si l'information ne transite pas dans la connexion (exemple irc, sip, skype, ...)
- des applications utilisent les ports http/https
  - => vérifier que ce qui y passe est http/https
- des applications s'encapsulent dans http ou https.

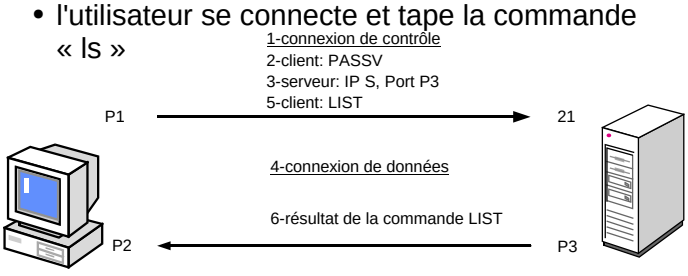
ftp : mode actif

- l'utilisateur se connecte et tape la commande « ls »



ftp : mode passif

- l'utilisateur se connecte et tape la commande « ls »



limitation des pares-feux à etat: irc