TP4 Filtrage TCP et Supervision

Partie 1

Objectifs : réglementer les accès à une machine ou à un réseau par la mise en place d'un firewall sur une machine ou sur un routeur qui devient ***routeur filtrant***.

Cas 1 : sécurisation des accès à un PC.

Architecture : au moins deux PC sur le même réseau.

Dans ce premier cas, après avoir capturé par un logiciel de métrologie comme Sniffer et observé des paquets de données circulant sur le réseau, mettre en place des « Règles » sur le FW.

Le FW fonctionne au niveau Transport et ses règles s'appuient sur les protocoles(souvent ICMP, TCP et UDP, etc), la source, la destination des paquets et les ports utilisés. La règle a pour but de spécifier l'**action** à effectuer sur le paquet selon sa source, sa destination, le protocole, le port : bloquer, autoriser, détruire, logger, etc.

Ex : Bloquer tous les paquets TCP/UDP de 172.16.0.2 à destination de 192.168.1.2

Cas 2 : Filtrage sur un routeur (sous Linux) par l'outil ***iptables*** (Netfilter)

Architecture : deux réseaux interconnectés par un routeur faisant aussi office de Firewall.

Travail à faire

1-Mettre en place l'architecture de réseau logique donnée en annexe 1.

Cette architecture présente une interconnexion de deux réseaux TCP/IP.

Le poste ***merlin*** sous linux est à la fois routeur et filtre de paquets. Mettre en œuvre le **routage** IP entre les deux réseaux.

2-Installer les services réseau proposés

3-Mettre en œuvre le **filtrage** TCP/UDP en utilisant

-l'outil GUFW, interface de configuration de **Netfilter** (intégré au noyau)

-l'utilitaire ***iptables*** en ligne de commande ou fichier de configuration à exécuter

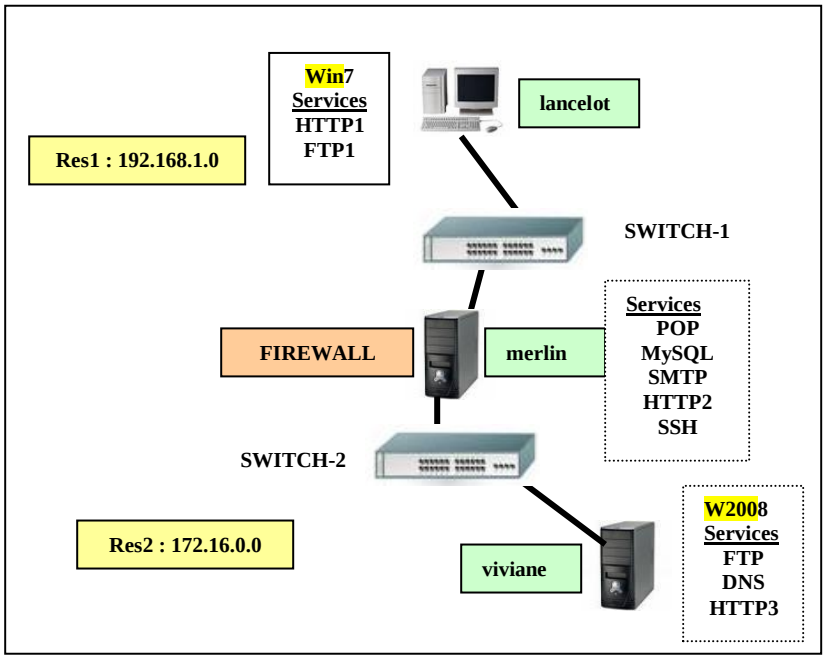
-l'outil ***webmin*** pour spécifier et activer les règles du FW.

4-Tester le respect des règles dans plusieurs cas. Exemples dans le tableau en annexe 1.

5-Créer un fichier **script** des règles à lancer automatiquement au démarrage.

Mots-clés : Protocole de transport, Paquet, filtrage, firewall, règle de filtrage, Netfilter, *iptables*, shorewall, fwbuilder

Annexe 1 Architecture du réseau logique



Annexe

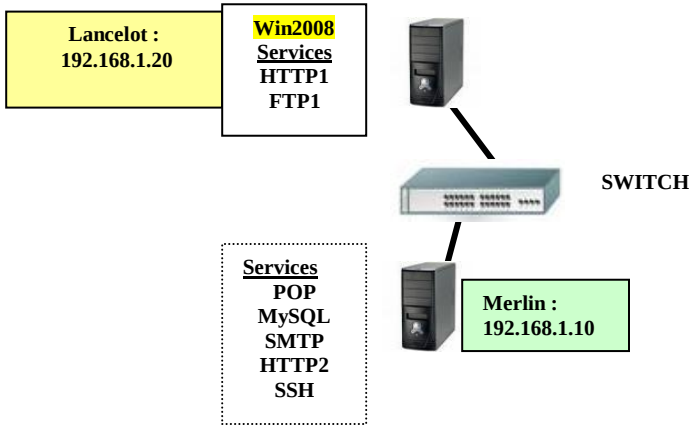
Exemples de règles de filtrage à paramétrer et tester

Destination	Source	Protocole	Port	ACTION
merlin	lancelot	FTP		REFUSER
merlin	lancelot	Ping		ACCEPTER
merlin			8080	REFUSER
merlin		HTTP2		ACCEPTER
merlin	lancelot	FTP1		REFUSER
www.google.fr	merlin			LOGGER
*	*	*	*	REFUSER

Partie 2 (page suivante)

Partie 2 Monitoring de composants et services avec **Nagios 3.5** ou **ZABBIX 3.x**

Dans cette partie, je vous propose un travail de découverte et mise en œuvre de la supervision des services et composants par le logiciel l'un des logiciels nagios ou zabbix.



Travail à faire

-Pour zabbix :

1)-Réaliser l'architecture ci-dessus comprenant :
- -un serveur windows (2008 server par ex .)
- -un serveur linux (ubuntu, xubuntu ou debian)

2)-Installer les services réseau proposés.
3)-Installer zabbix-server (avec les paquets ou les sources).
4)-Installer les agents zabbix
- Sur windows
- Sur linux.

5)-Configurer/Créer les hôtes à monitorer lancelot windows et merlin sous linux.
6)-Créer les services des hôtes précédents.
7)-Visualiser les états des différents services.
8)-Créer deux incidents (arrêts volontaire des services) et revisualiser les Graphes des hôtes et services.

NB : Vous proposerez toutes les captures qui vous semblent illustrer des situations intéressantes.

-Pour Nagios 3.5 :

Faire la même chose, changer les icones et essayer la gestion d'**alertes par mail et par SMS**.