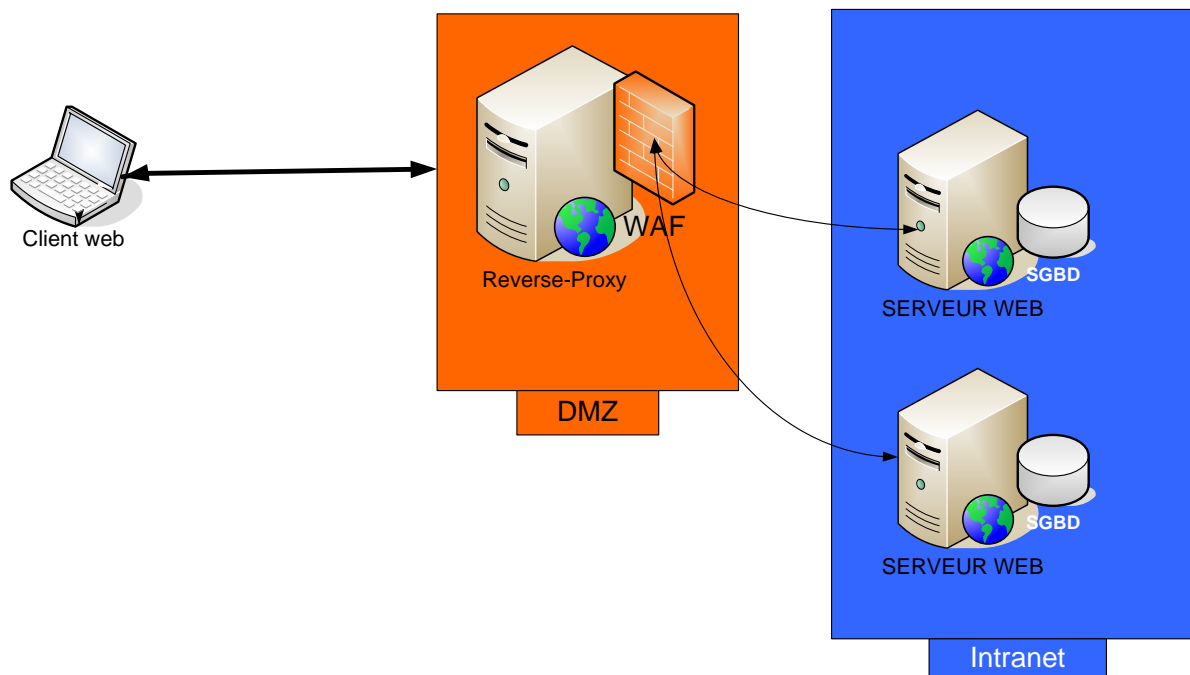


## TP2 : Déploiement d'une Solution de filtrage Web

### 1- L'architecture :

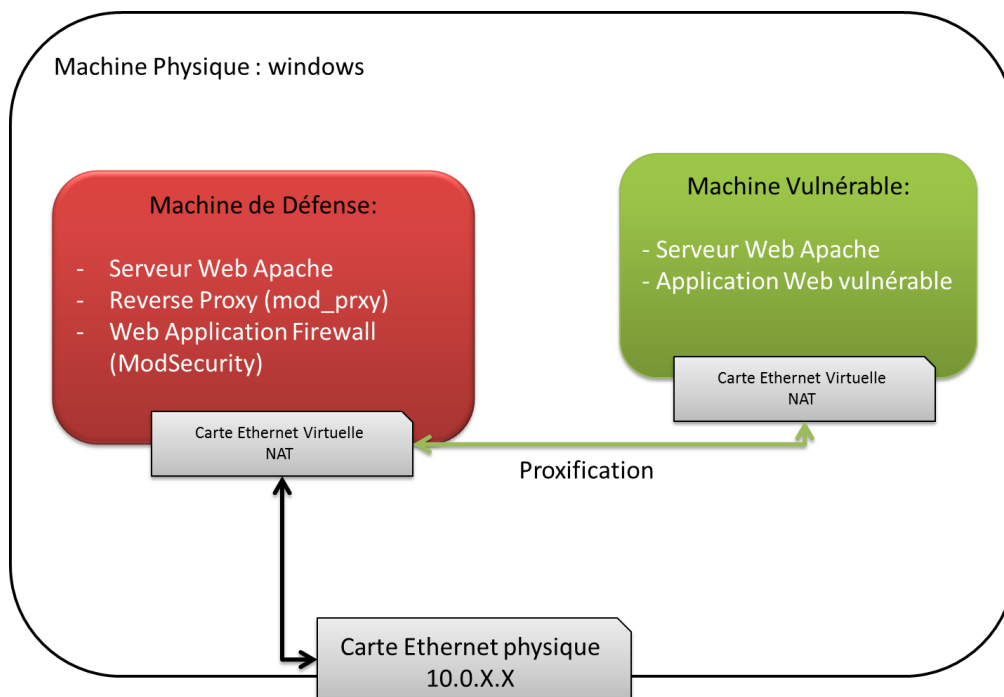
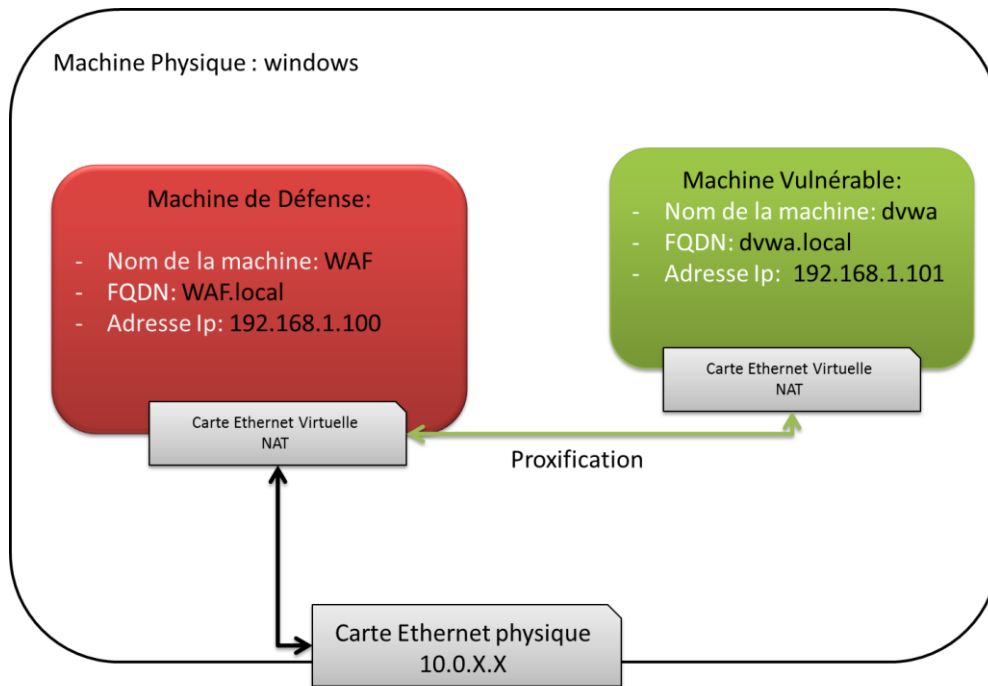
L'objectif de cet atelier est de démontrer deux niveaux de de défense pour sécuriser vos applications web.

- Le premier niveau est une solution architecturale. En effet il est possible de protéger les serveurs web physiques qui hébergent les applications web (Back-ends) en utilisant un serveur web placé frontalement (Front-end) et qui joue le rôle d'un relais inverse Reverse-Proxy dans une DMZ par exemple.
- Le deuxième niveau est une solution de filtrage actif des requêtes et éventuellement des réponses http. Le filtre en question est Web Application Firewall WAF qu'on placera dans le Reverse-Proxy pour profiter d'un niveau supplémentaire de sécurité.



Pour émuler ces deux aspects, nous allons utiliser deux machines virtuelles à l'aide de VMware.

- La première machine joue le rôle de défense, elle héberge :
  - Un serveur Web Apache
  - Un reverse Proxy e assuré par le mod\_proxy de Apache
  - Un WAF intégré assuré par le module ModSecurity
- La deuxième machine héberge une application vulnérable (DVWA)

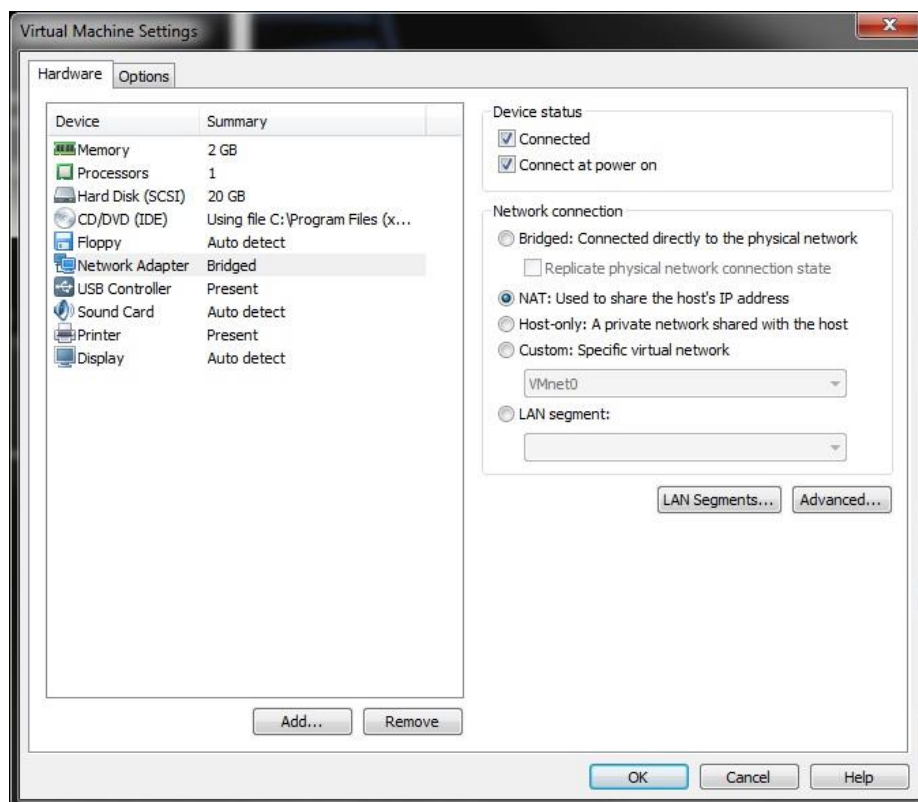
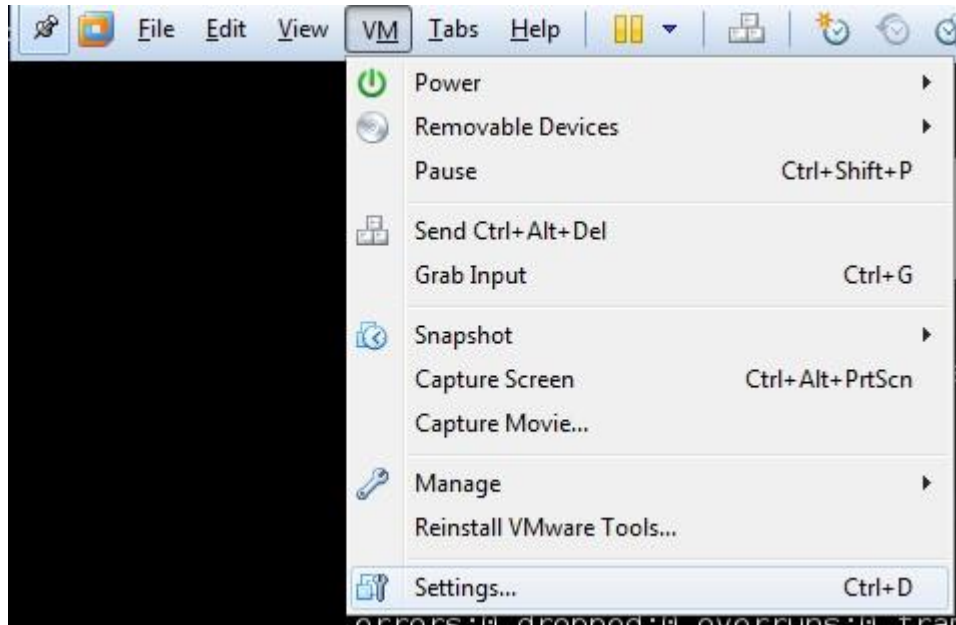


Les adresses IP sont données à titre indicatif, il faut impérativement les changer avec celles des machines qu'on va réellement utiliser dans cet atelier.

## 2- La machine vulnérable:

Pour gagner du temps, la machine vulnérable est fournie. Le mot de passe root de cette machine est : *root*.

Assurez-vous que la machine virtuelle tourne avec une carte Ethernet en mode NAT :



Pour récupérer l'adresse IP réelle de cette machine vulnérable :

```
ifconfig eth0
```

### 3- La machine de défense:

Nous utilisons la machine KALI Linux (disponible dans vos PCs) pour configurer notre machine de défense. Comme le cas précédent, il faut que cette machine aussi soit en mode NAT. Notez l'adresse Ip réelle et procédez à l'installation du reverse proxy et du WAF comme suit:

#### 3.1- Configuration DNS

Le but de cette étape est de faire connaître le nom du serveur web (vulnérable) auprès de notre reverse proxy. Cela lui permet de relayer les requêtes entre le serveur web et le monde externe en se basant sur le FQDN (FullyQualified Domain Name) envoyé dans l'entête Host de la requête.


- a- Changement du nom de la machine KALI : Dans le fichier /etc/hostname le nom devient WAF

```
echo "WAF"> /etc/hostname
```

- b- Association des adresses Ip aux noms des machines dans le fichier /etc/hosts

```
leafpad /etc/hosts
```

```
127.0.0.1      localhost
127.0.1.1      WAF.local WAF
192.168.116.136 dvwa.local dvwa
```



Remplacer avec l'adresse Ip réelle de la dvwa

```
reboot
```

Rebooter la machine WAF pour la prise en compte des nouveaux paramètres

- c- Associer le nom de domaine **dvwa.local** à la machine **WAF** au niveau de la machine physique (**Windows**). Dans le fichier : c:\WINDOWS\system32\drivers\etc\hosts rajouter les deux lignes après la ligne vide dans un terminal DOS avec **les droits d'administrateur** :

```
echo . >> c:\WINDOWS\system32\drivers\etc\hosts
```

```
echo 192.168.116.130 dvwa.local >> c:\WINDOWS\system32\drivers\etc\hosts
```

Remplacer avec l'adresse Ip réelle de WAF

### 3.2- Configuration du Reverse-Proxy

- a- Activation du module mod\_proxy qui donne à notre serveur web Apache la fonctionnalité supplémentaire du relais inverse ou Reverse-Proxy

```
a2enmod proxy_http
```

- b- Création d'un virtual host pour notre machine vulnérable dvwa sur domaine dvwa.local

```
leafpad /etc/apache2/sites-enabled/000-default
```

Ajout des lignes ci-dessous en fin de fichier

```
<VirtualHost *:80>
ServerName dvwa.local
ProxyPreserveHost On
ProxyRequests off
ProxyPass / http://dvwa.local/
ProxyPassReverse / http://dvwa.local/
</VirtualHost>
```

Test du Reverse-Proxy

```
service apache2 restart
```

Dans la machine Windows, lancez le navigateur internet et tapez dans la barre des adresses :

<http://dvwa.local>

### 3.3- Installation du WAF ModSecurity dans le Reverse-Proxy

Nous commençons par télécharger et installer le module ModSecurity pour Apache

```
apt-get install libapache2-modsecurity
```

Attention à la version de KALI. En cas de problème, renouvelez les clés de l'entrepôt des téléchargements de apt-get.

Nous activons le fichier de configuration recommandé par l'éditeur.

```
cp /etc/modsecurity/modsecurity.conf-recommended /etc/modsecurity/modsecurity.conf
```

Pour le fonctionnement du moteur de détection de ModSecurity on a besoin d'une base de règles de sécurité qu'on va télécharger à partir de l'entrepôt de SpiderLabs « projet modsecurity du owasp »

```
cd /etc/modsecurity
```

```
git clone https://github.com/SpiderLabs/owasp-modsecurity-crs.git
```

Nous remarquons qu'il a téléchargé et a créé un répertoire owasp-modsecurity-crs. Nous allons activer aussi le fichier de configuration des règles de sécurité qui contient des directives d'organisation des fichiers de règles de sécurité ( scores d'anomalies, les règles collaboratives..).

```
cp /etc/modsecurity/owasp-modsecurity-crs/crs-setup.conf.example  
/etc/modsecurity/owasp-modsecurity-crs/crs-setup.conf
```

On va pouvoir spécifier au moteur de détection qu'elles sont les règles de sécurité à utiliser :

```
leafpad /etc/apache2/mods-available/mod-security.conf
```

Ajoutez ces deux lignes à l'intérieur du contexte **<IfModule security2\_module>**

```
Include "/etc/modsecurity/owasp-modsecurity-crs/crs-setup.conf"  
Include "/etc/modsecurity/owasp-modsecurity-crs/rules/*.conf"
```

Redémarrez Apache2:

```
service apache2 restart
```

Si on à un message d'erreur, c'est que la version actuelle des CoreRule Set CRS du OWASP est de 2.7 et elle n'est pas complètement compatible avec le moteur d'analyse de modsecurity installé sur la KALI, donc il faut revenir vers la version 2.6. Il existe un script perl qui permet justement de revenir sur cette version.

```
perl /etc/modsecurity/owasp-modsecurity-crs/util/rule-  
management/remove-2.7-actions.pl -t 2.6 -f .
```

## 4- Tester les règles de Sécurité de ModSecurity

Réaliser une attaque injection SQL ou une commande exécution en mode « securityLow »

Remarquez que le serveur Web laisse passer l'attaque, mais si on consulte les logs de ModSecurity on trouve la trace de l'attaque k.

```
tail -f /var/log/apache2/modsec_audit.log
```

Pour activer le mode de Détection et l'interception des attaques il faut placer l'option **SecRuleEngine** sur **On** au lieu de **DetectionOnly** dans le fichier de configuration de ModSecurity

```
leafpad /etc/modsecurity/modsecurity.conf
```

**SecRuleEngine On**