الجمهورية الجزائرية الديمقراطية الشعبية
République Algérienne Démocratique et Populaire
وزارة التعليم العالي والبحث العلمي
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

N° Réf :……

Centre Universitaire
Abdelhafid Boussouf Mila

Institut des Sciences et Technologie Département de Mathématiques et Informatique

Mémoire préparé en vue de l'obtention du diplôme de Master

En : Informatique

Spécialité: Sciences et Technologies de l'Information et de la Communication (STIC)

Mobilité et Sécurité sur le réseau [universitaire], mise en place de solutions VPN.

Préparé par :

- ***Belmahboul Oussama.***

Soutenue devant le jury :

Encadrer par	**Guettiche Mourad**	**M.A.A**	**C.U.Abd Elhafid Boussouf.**
Président :	**Boukhchem Nadir**	**M.A.A**	**C.U.Abd Elhafid Boussouf.**
Examinateur :	**Boumsata Meryem**	**M.A.A**	**C.U.Abd Elhafid Boussouf.**
Invité :	**Rezal Farid**	**Assistant de sécurité**	**Ooredoo – Alger .**

Année Universitaire : 2018/2019

"لله الحمد من قبل ومن بعد"

Remerciements

Je tiens à remercier en premier lieu mon créateur Allah, Grand et Miséricordieux, le tout Puissant de m'avoir donné courage et santé pour achever ce travail.

A cet effet, je tiens à exprimer mes vifs remerciements à Monsieur M. GUETTICHE et Monsieur F. REZAl pour leurs encouragements, leurs conseils, leur disponibilité et surtout pour leur patience dans l'encadrement de ce mémoire.

Mes remerciements s'adressent ensuite aux membres du jury. Je remercie vivement Mr. BOUKHCHEM et Mme. BOUMSATA pour l'honneur qu'ils m'ont fait en acceptant d'être membre de jury et pour leurs contributions à l'évaluation et l'enrichissement du présent travail.

Je remercie profondément tous les enseignants qui m'ont encouragés et soutenus pendant mon cursus.

Je remercie du fond du cœur, ma famille qui m'ont soutenu, encouragés et motivés tout au long de ce travail.

Afin de n'oublier personne, mes vifs remerciements s'adressent à tous ceux qui m'ont aidé à la réalisation de ce modeste mémoire.

Dédicace

Je dédie ce mémoire

A mes chers parents ma mère et mon père

Pour leur patience, leur amour, leur soutient et leur

Encouragement

A mes sœurs

A mes frères

A toute ma famille

A tous mes amis

A tous ceux qui ont une relation de proche ou de loin

Avec la réalisation du présent mémoire.

En reconnaissance de tous les sacrifices consentis par tous et

chacun pour me permettre d'atteindre cette étape de ma vie.

Avec toute ma tendresse.

OUSSAMA BELMAHBOUL

Table des matières

Introduction Général **11**

1 Cryptographie et réseau privé virtuel **13**

Introduction . . . 13

I.Cryptographie . . . 13

1.1 Introduction à la cryptographie . . . 13

1.1.1 Définition de cryptographie . . . 14

1.1.2 Vocabulaire de base . . . 14

1.1.3 L'usage de la cryptographie . . . 14

1.1.4 Types de systèmes cryptographique . . . 15

1.2 Algorithmes de chiffrement par blocs . . . 17

1.2.1 D.E.S. - Data Encryption Standard . . . 17

1.2.2 Principe d'algorithme D.E.S . . . 17

1.2.3 A.E.S. - Advanced Encryption Standard . . . 22

1.2.4 Principe d'algorithme A.E.S . . . 22

1.2.5 Avantages et limites de AES . . . 23

1.3 Algorithmes de chiffrement par clef publique . . . 24

1.3.1 RSA : Rivest - Shamir - Adleman . . . 24

1.4 Algorithmes de hachage . . . 26

1.4.1 MD5 . . . 26

1.4.2 SHA-1 . . . 29

1.4.3 SHA-256 . . . 30

1.5 Authentification et intégrité des données . . . 30

1.5.1 HMAC - Message Authentication Code . . . 30

1.5.2 Signatures électronique . . . 31

II. Réseau privé virtuel (VPN) . . . 33

1.6 Présentation réseau privé virtuel (VPN) 34
1.6.1 Définition d'un VPN 34
1.6.2 Principe de fonctionnement 34
1.6.3 Intérêt d'un VPN 34
1.7 Les fonctionnalités d'un réseau privé virtuel (VPN) 35
1.8 Protocoles utilisée pour réaliser une connexion VPN 35
1.8.1 PPTP (Point-to-Point Tunneling Protocol) 36
1.8.2 L2TP (Layer Two Tunneling Protocol) 36
1.8.3 IPSEC (Internet Protocol Security) 36
1.8.4 SSL/TLS 37
Conclusion 37

2 Les outils de réalisation **39**
Introduction 39
2.1 VMware Workstation 14 39
2.1.1 Définition du VMware Workstation 39
2.1.2 Les composants du réseau virtuel 40
2.1.3 Les types de réseau VMware 40
2.2 EVE-ng 42
2.2.1 Qu'est-ce que EVE-ng ? 42
2.2.2 Installation et configuration EVE-ng sur VMware Workstation 43
2.2.3 Mise en place et créer un fichier lab en EVE 43
2.2.4 Interfaces cloud et types de réseaux EVE-ng 45
2.3 Etude des différents pare-feux supportant VPN 46
2.3.1 Aperçu des Pares-feux 46
2.3.2 Solution PFsense 48
2.3.3 Installation de PFsense 48
2.4 Docker 52
2.4.1 Qu'est-ce que Docker ? 52
2.4.2 L'architecture de Docker 52
2.4.3 l'installation de Docker edition community(CE) 54
2.5 Nextcloud 55
2.5.1 Définition Nextcloud 55
2.5.2 Fonctionnalité de Nextcloud 55
2.5.3 Installe nextCloud dans Docker 56
Conclusion 56

3 Réalisation technique **57**

Introduction 57

3.1 Virtualisation 57

3.1.1 Mécanisme de virtualisation 57

3.1.2 Virtualisation Système 58

3.2 L'architecte DMZ(DéMilitarized Zone) 59

3.3 Présentation de l'architecture 60

3.4 Configuration de PFsense 62

3.4.1 Configuration générale 62

3.4.2 Configuration des interfaces 64

3.4.3 Règles du pare-feu 64

3.5 OpenVPN 65

3.5.1 présentation d'OpenVPN 65

3.5.2 Critères d'OpenVPN 65

3.5.3 Fonctionnement d'OpenVPN 66

3.6 Configuration d'OpenVPN sous PFsense 68

3.6.1 Création du tunnel OpenVPN 68

3.6.2 Exemple de configuration d'OpenVPN sous PFSense 73

3.7 Scénarios d'exécution de l'application Nextcloud 80

3.7.1 Scénario d'exécution normal 80

3.7.2 Scénarios d'attaque 82

Conclusion 82

Conclusion Générale **84**

Bibliographie **84**

Table des figures

1.1 Principe d'un algorithme de controle d'intégrité. 15
1.2 cryptographie symétrique (à clé secrète). 16
1.3 Principe du chiffrement à clef publique. 16
1.4 Roundes de DES. 18
1.5 Algorithme de DES. 18
1.6 Représentation matricielle d'un bloc de 16 octets.[1] 23
1.7 Processus de chiffrement de l'AES.[17] 24
1.8 Vue d'ensemble du MD5.[2] 27
1.9 L'algorithme MD5. 28
1.10 Le i-éme tour dans MD5 ($0 \leq i \leq 63$).[2] 28
1.11 Le i-ème tour dans SHA-1 ($0 \leq i \leq 79$).[2] 29
1.12 Calcul du W_t dérivé.[2] 30
1.13 Principe de création des certificats. 33
1.14 Vérification d'identité de certificate. 33

2.1 Le réseau Bridge. 41
2.2 Le réseau NAT. 42
2.3 Le réseau host-only. 42
2.4 Créer un nouveau projet. 43
2.5 Ajouter nouveau fichier lab. 44
2.6 Créer un fichier lab. 44
2.7 Gérer le fichier lab. 44
2.8 L'interface graphique de lab. 45
2.9 Ajouter un équipement réseau. 45
2.10 Creér un machine virtuelle. 49
2.11 Lancer l'installation de pfsense. 49

2.12 Écran de démarrage de FreeBSD. 50
2.13 Boite de dialogue pour la configuration de VLAN. 50
2.14 Validation des noms d'interface. 50
2.15 Menu textuel de PFsense. 51
2.16 Configuration de l'adresse IP LAN. 52
2.17 L'architecture Docker [12]. 53
2.18 Conteneur Vs machines virtuelle. 54

3.1 Hyperviseur de Type 1. 59
3.2 Hyperviseur de Type 2. 59
3.3 Un schéma de DMZ (Démilitarized Zone). 59
3.4 Un schéma général de VPN. 60
3.5 Schéma détaillé de l'architecture adoptée. 61
3.6 Schéma global. 62
3.7 Portail de connexion à PFsense. 62
3.8 Paramètres généraux de PFsense. 63
3.9 Écran d'accueil de PFsense. 63
3.10 Configuration de l'interface DMZ. 64
3.11 Fonctionnement d'OpenVPN. 66
3.12 Protocols SSL/TLS. 69
3.13 Dialogue TLS entre un client et un serveur.[3] 70
3.14 Chiffrement avec le record protocole 72
3.15 Installer le package d'export client OpenVPN. 73
3.16 Créer un certificat autorité. 74
3.17 Créer un certificat autorité. 74
3.18 Certificat autorité(CA). 75
3.19 Certificat de serveur. 75
3.20 Installation wizard de OpenVPN. 76
3.21 Choisissez l'autorité de certification. 76
3.22 Choisir du certificat de serveur. 77
3.23 Configurez les paramètres de base de serveur OpenVPN. 77
3.24 Configurez les paramètres du réseau de tunnels. 78
3.25 Régle autoriser le flux VPN sur l'interface WAN. 78
3.26 Régle autoriser le flux VPN sur l'interface OpenVPN. 79
3.27 Créer comptes utilisateurs OpenVPN. 79
3.28 Téléchargement des packages clients OpenVPN. 80

3.29 Importer les fichers de configuration OpenVPN. 80
3.30 Portail de connexion à VPN. 81
3.31 Portail de connexion à Nextcloud. 81
3.32 Page d'accueil de Nextcloud. 82

Liste des tableaux

1.1 Matrice de réduction de la clé. 19
1.2 Nombre de chiffres tournés ver la gauche dans chaque itération. 20
1.3 Matrice de permutations initiale. 21
1.4 Les informations d'un certificat. 32

2.1 Interfaces cloud et ses correspondantes dans Vmware. 46

3.1 Plan d'adressage. 62

Introduction Générale

L'évolution de la technologie a conduit au développement des réseaux informatiques d'une façon considérable. En effet ils prennent de plus en plus une place stratégique au sein des entreprises, des sociétés et des universités qui les utilisent pour partager des informations, généralement selon le modèle client-serveur dans n'importe quel endroit du monde.

Alors que l'informatique est devenue un outil incontournable de gestion, d'organisation de communication. les données traitées par les systèmes d'informations subirent à des échanges internes et externes et stockées dans différent serveurs ce qui l'exposent aux différent menaces notamment les méthodes d'intrusion qui devenu ces jours sophistiquées.

La sécurité informatique est une problématique importante car les effets sont devenu très lourds. Notamment avec le développement de l'utilisation d'Internet, de nombreuses universités connectent leurs réseaux, ce qui l'expose à une menace ciblé qui peut apporter des effets désastreux sur la fonctionnalité de son système.

Afin de résoudre ce problème, la technologie VPN (Virtual Private Network) est apparait pour contrer ce problème de sécurité et permettre à l'utilisateur de pouvoir accéder au réseau via internet.

L'objectif de notre projet consiste donc à implémenter une solution VPN à travers OpenVPN garantissant l'interconnexion entre l'université, les enseignants et les étudiants et d'offrir un accès distant aux services de partage des fichiers d'une manière sécurisée et fiable.

Hormis, l'introduction générale et la conclusion, ce manuscrit est structuré en trois chapitres :

Le premier est théorique.il est subdivisé en deux parties dont nous donnons sommairement le contenu ci-dessous :

Dans la première partie : elle est dédiée aux généralités sur la cryptographie, ses principaux concepts, les algorithmes de chiffrement AES, DES et ceux de l'authentification et d'intégrité : MD5, SHA-1, SHA-256 et RSA. Puis dans la deuxième partie nous nous intéresserons au réseau

privé virtuel y compris : sa définition, son principe de fonctionnement ainsi que ses différents protocoles utilisés pour réaliser une connexion VPN.

Le deuxième chapitre est destiné à la présentation des outils nécessaires à la réalisation de notre travail y compris : VMware Workstation 14, EVE-ng (Emulated Virtual Environment - Next Generation), Docker et Nextcloud et aussi les étapes d'installation de ces derniers en passant par l'étude des différents pare-feu qui supporte le VPN en choisissant le PFsense comme une solution.

Dans le troisième chapitre, nous avons commencé par la présentation de l'architecture adoptée puis nous avons focalisé sur le mécanisme de partage des document utilisé (Nextcloud). Finalement, nous avons montré le fonctionnement de l'application à travers des scénarios typiques qui montre le comportement de l'application en répondant aux différents attaques possibles.

Chapitre 1

Cryptographie et réseau privé virtuel

Introduction

L'expansion et l'importance grandissante des réseaux informatiques, ont engendré le problème de sécurité des systèmes de communication ; dans la plupart d'organisations informatisées, partager les données directement entre machines est leur souci majeur. Il s'avère indispensable de renforcer les mesures de sécurités, dans le but de maintenir la confidentialité, l'intégrité et le contrôle d'accès au réseau pour réduire les risques d'attaques. Ce renforcement consiste à utiliser les réseaux privés virtuels (VPN), qui sont idéals pour pouvoir exploiter au mieux les capacités du réseau Internet et de relier des différents sites en toute sécurité.

Ce chapitre est consacré tout d'abord pour parler sur les principes de concepts cryptographiques et le principe de fonctionnement des algorithmes de cryptages et de hachage. Ensuite s'arrive la présentation de la technologie réseau privé virtuel (VPN) et les différents protocoles utilisés pour réaliser une connexion VPN.

I. Cryptographie

1.1 Introduction à la cryptographie

Protéger les informations confidentielles ou simplement s'assurer de leur authenticité ou de leur provenance est devenu, avec le développement des réseaux informatiques, un problème crucial. Pour cela, Le cryptographie cherche des méthodes qui assurer la sûreté et la sécurité des conversations.

1.1.1 Définition de cryptographie

La cryptographie est une science mathématique permettant d'assurer la confidentialité en rendant un message inintelligible afin de le transférer à un destinataire de telle sorte que s'il est intercepté il ne puisse être lu.[2]

1.1.2 Vocabulaire de base

La cryptographie comporte plusieurs vocabulaires :

a) **Cryptosystème :** Il est défini comme l'ensemble des clés possibles (espace de clés), des textes clairs et chiffrés possibles associés à un algorithme donné.

b) **Cryptanalyse :** Opposée à la cryptographie, elle a pour but de retrouver le texte clair à partir de textes chiffrés en déterminant les failles des algorithmes utilisés.

c) **Chiffrement :** Le chiffrement consiste à transformer une donnée (texte, message, ...) afin de la rendre incompréhensible par une personne autre que celui qui a créé le message et celui qui en est le destinataire. La fonction permettant de retrouver le texte clair à partir du texte chiffré porte le nom de déchiffrement.

d) **Clef :** Il s'agit du paramètre impliqué et autorisant des opérations de chiffrement et/ou déchiffrement. Dans le cas d'un algorithme symétrique, la clef est identique lors des deux opérations. Dans le cas d'algorithmes asymétriques, elle diffère pour les deux opérations.

e) **Texte chiffré :** Appelé également cryptogramme, le texte chiffré est le résultat de l'application d'un chiffrement à un texte clair.[2]

1.1.3 L'usage de la cryptographie

Les communications échangées entre deux personnes (par exemple Alice et Bob) sont sujettes à un certain nombre de menaces(Attaques passives/actives, Attaque par analyse différentielle, Attaque par séquences forcées..). La cryptographie apporte un certain nombre de fonctionnalités permettant de pallier ces menaces pour :

- **La confidentialité** des informations stockées ou manipulées par le biais des algorithmes de chiffrement. La confidentialité consiste à empêcher l'accès aux informations qui transitent à ceux qui n'en sont pas les destinataires. Ils peuvent lire les messages cryptés transmis sur le canal mais ne peuvent pas les déchiffrer.
- **L'authentification** d'une communication, il faut pouvoir détecter une usurpation d'identité. Par exemple, Alice peut s'identifier en prouvant à Bob qu'elle connait un secret S qu'elle est la seule à pouvoir connaitre.

- **L'intégrité** des informations envoyer, on vérifier que le message n'a pas subi d'altérations lors de son parcours. Elle concerne plutôt une modification volontaire et malicieuse l'information provoquée par un tiers lors du transfert sur le canal. Ces modifications sont en générales masquées par le tiers pour être difficilement détectables. Sur la figure 1.1, par exemple, le contrôle d'intégrité sur un message M se fait grâce à une fonction f telle qu'il doit être très difficile de trouver deux messages M1 et M2 ayant la même image A par f.[1]

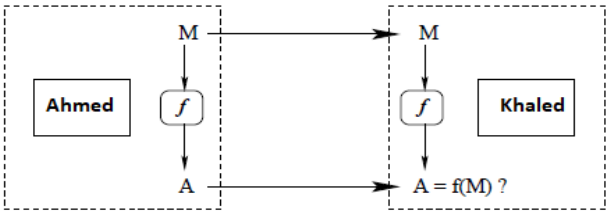


FIGURE 1.1 – Principe d'un algorithme de controle d'intégrité.

1.1.4 Types de systèmes cryptographique

la cryptographie se répartissent en deux grandes catégories : cryptographie symétrique (à clé secrète) et cryptographie asymétrique (à clé publique).

a) **Cryptographie symétrique (à clé secrète)**

La cryptographie symétrique utilise la même clé K = $K_{Emiteur} = K_{Destinataire}$ pour les processus de chiffrement et de déchiffrement. cette clé est le plus souvent appelée "secrète" car toute la sécurité de l'ensemble est directement liée au fait que cette clé n'est connue que par l'expéditeur et le destinataire, (Voir figure 1.2). Il existe deux types de chiffrement à clé symétrique :

- Le chiffrement par blocs : l'opération de chiffrement s'effectue sur des blocs de texte clair, tel le DES et AES.
- Le chiffrement par flots (ou par stream ou de flux) : l'opération de chiffrement s'opère sur chaque élément du texte clair (caractère, bits). On chiffre un bit/caractère à la fois.

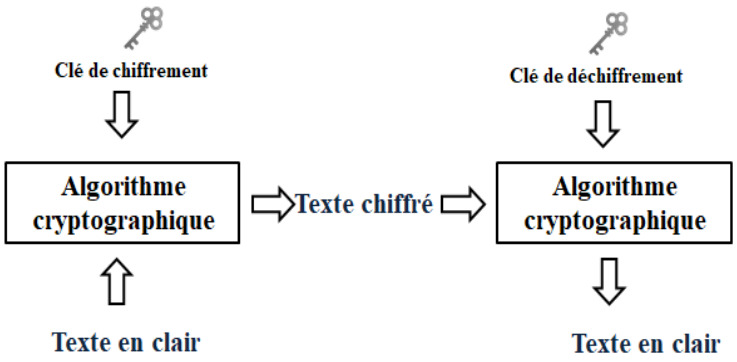


FIGURE 1.2 – cryptographie symétrique (à clé secrète).

b) **Cryptographie asymétrique (à clé publique)**

Dans le cas des systèmes symétriques, la même clef est utilisée pour le chiffrement et le déchiffrement. Le problème essentiel de la cryptographie symétrique est la distribution des clefs.

L'idée de base des cryptosystèmes à clefs publiques a été proposée dans un article fondamental de Diffie et Hellman en 1976. Le principe fondamental est d'utiliser des clefs de chiffrement et déchiffrement différentes, non reconstructibles l'une à partir de l'autre, (Voir figure 1.3) :

– Une clef publique pour le chiffrement.

– Une clef privée (secrète) pour le déchiffrement.

Le gros avantage de ce système est La distribution des clés est grandement facilitée car l'échange de clés secrètes n'est plus nécessaire. Chaque utilisateur conserve sa clé secrète sans jamais la divulguer. Seule la clé publique devra être distribuée.

En revanche les implémentations de tels systèmes (RSA,...) ont un inconvénient majeur : leur lenteur par rapport à leurs homologues à clefs secrètes qui tournent eux jusqu'à près de mille fois plus vite.

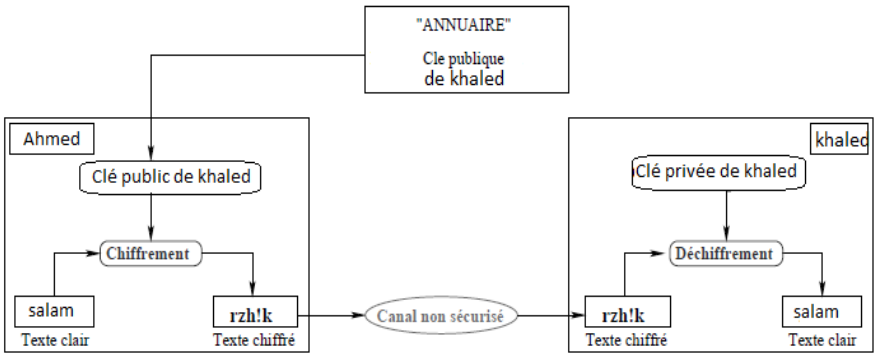


FIGURE 1.3 – Principe du chiffrement à clef publique.

1.2 Algorithmes de chiffrement par blocs

1.2.1 D.E.S. - Data Encryption Standard

Le D.E.S. (Data Encryption Standard, c'est-à-dire Standard de Chiffrement de Données) est un standard mondial depuis la fin des années 1970.

Au début de cette décennie, le développement des communications entre ordinateurs a nécessité la mise en place d'un standard de chiffrement de données pour limiter la prolifération d'algorithmes différents ne pouvant pas communiquer entre eux. Pour résoudre ce problème, L'Agence Nationale de Sécurité américaine (N.S.A.) a lancé des appels d'offres. La société I.B.M. a développé alors un algorithme nommé Lucifer, relativement complexe et sophistiqué. Après quelques années de discussions et de modifications (applications de S-Boxes et réduction à des clés de 56 bits), cet algorithme, devenu alors D.E.S., fut adopté au niveau fédéral le 23 novembre 1976.[2]

Le DES comporte plusieurs avantages qui en ont fait l'algorithme de chiffrement symétrique standard pendant longtemps, jusqu'il y a quelques années. En voici quelques-uns :

- il possède un haut niveau de sécurité.
- il est facile à comprendre.
- il est rendu disponible à tous, par le fait qu'il est public.
- il est adaptable à diverses applications (logicielles et matérielles).
- il est facile à implémenter.

1.2.2 Principe d'algorithme D.E.S

Le D.E.S. est un cryptosystème agissant par blocs. Cela signifie que D.E.S. ne chiffre pas les données à la volée quand les caractères arrivent, mais il découpe le texte clair en blocs de 64 bits qu'il code séparément, puis qu'il concatène. Un bloc de 64 bits du texte clair entre par un coté de l'algorithme et un bloc de 64 bits de texte chiffré sort de l'autre coté, (Voir figure 1.5). L'algorithme est assez simple puisqu'il ne combine en fait que des permutations et des substitutions.

C'est un algorithme de chiffrement à clé secrète. La clé sert donc à la fois à chiffrer et à déchiffrer le message. Cette clé a ici une longueur de 64 bits mais dont seulement 56 bits sont utilisés. On peut donc éventuellement imaginer un programme testant l'intégrité de la clé en exploitant ces bits inutilisés comme bits de contrôle de parité. L'entière sécurité de l'algorithme repose sur les clés puisque l'algorithme est parfaitement connu de tous. La clé de 64 bits est utilisée pour générer 16 autres clés de 48 bits chacune qu'on utilisera lors de

chacune des 16 itérations du D.E.S, (Voir figure 1.4). Ces clés sont les mêmes quel que soit le bloc qu'on code dans un message.

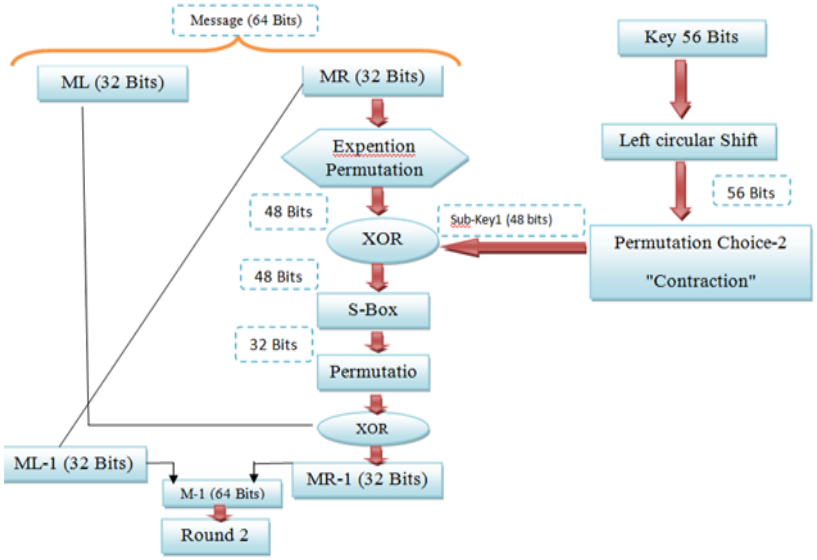


FIGURE 1.4 – Roundes de DES.

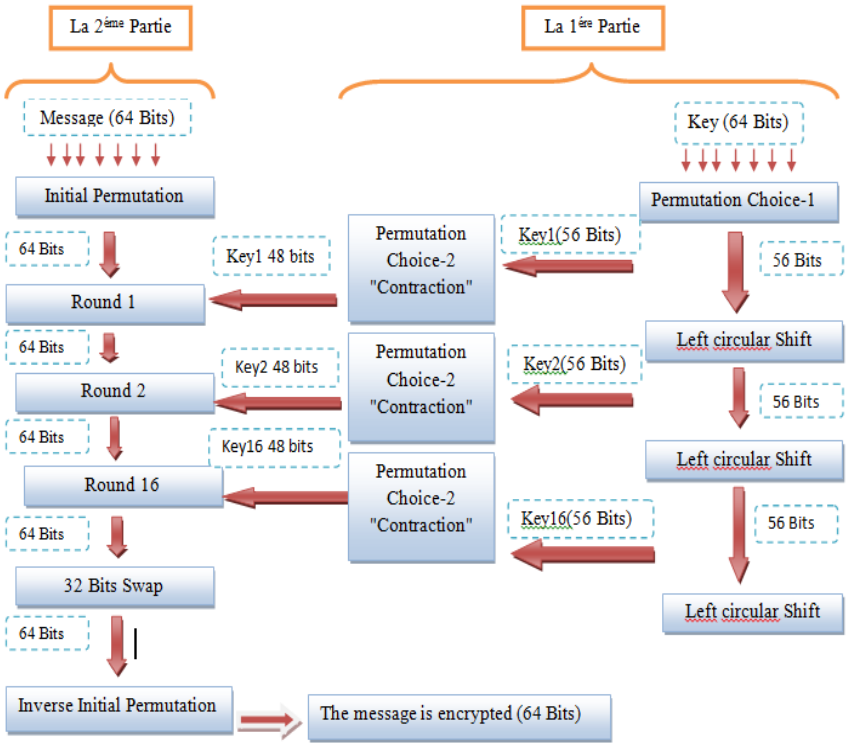


FIGURE 1.5 – Algorithme de DES.

L'algorithme repose principalement sur 3 étapes :

1. Permutation initiale.
2. Calcul médian (16 fois) : application d'un algorithme complexe appliqué en fonction de la clé.
3. Permutation finale.

Nous notons également que le graphique est divisé en deux parties, on va expliquer en détail chaque partie.

- La 1ère partie on applique des règles sur la clé qui sont :

1. **Permutation Choice-1**

   La clé est constituée de 64 bits dont 56 sont utilisés dans l'algorithme.Il changer sa place d'une façon aléatoire en utilisant des numéros variés de 1 à 56 qui sont déterminés par le crypteur et doivent être en désordre, comme il est indiqué sur la matrice suivante :

57	49	41	33	25	17	9
1	58	50	42	34	16	18
10	2	59	51	43	35	27
19	11	3	60	52	44	36
63	55	47	39	31	23	15
7	62	54	46	38	30	22
14	6	61	53	45	37	29
21	13	5	28	20	12	4

TABLE 1.1 – Matrice de réduction de la clé.

   - On commence par le nombre 57 dans la première case dans la matrice, on calcule les bits de clé (K) jusqu'à ce que nous arrivons au Bit de numéro 57.

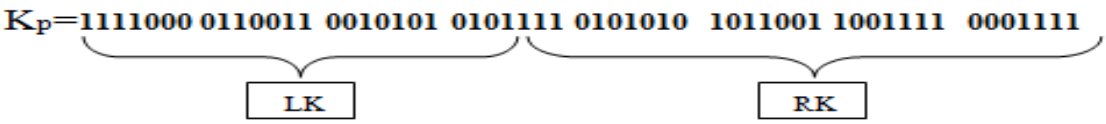
   - On change par la suite le rang de bit 57 à la première place.

   - On fait la même chose pour le nombre 49 dans la deuxième case et ainsi de suite jusqu'à la fin.

   Cette opération va donner une clé (Kp) de 56 Bits comme montre l'exemple suivant : Kp=1111000 0110011 0010101 0101111 0101010 1011001 1001111 0001111

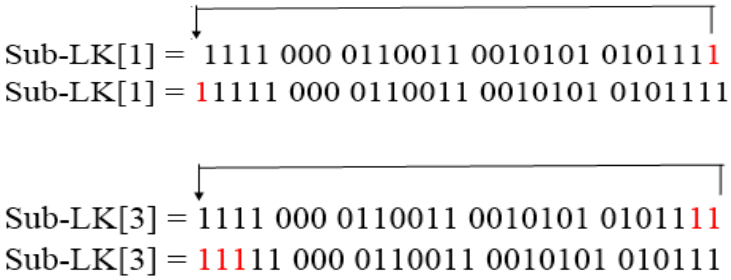
2. **Left circular Shift**

   La clé résultante de l'opération de Permutation Choice-1 sera divisée en deux partie : left Key (LK) et right Key (RK).



On applique sur LK et RK le tableau 1.2, Dans la 1ère, 2éme et 9éme itération on tourne à gauche un seul Bit, par contre dans les autres itérations on tourne 2 Bits.

Exemple :



Il se résulte 16 sous-clés de LK (LK [1]... LK [16]) et 16 sous-clé de RK (RK [1]... RK [16]).

Round Number	Number of left shift
1	1
2	1
3	2
4	2
5	2
6	2
7	2
8	2
9	1
10	2
11	2
12	2
13	2
14	2
15	2
16	2

TABLE 1.2 – Nombre de chiffres tournés ver la gauche dans chaque itération.

3. **Permutation Choice-2 :** Le principe de fonctionnement de cette règle est le même que celui de la permutation choice-1, sauf que dans cette foie la sous clé entrée de taille de 56 Bits se réduise à une sous clé de taille de 48 Bits. La taille de la matrice que l'on applique cette règle est de 6∗8.

- La 2ème partie

les règles appliquées sur le message sont :

1. **Initial Permutation**

- Les 64 bits du bloc d'entrée subissent la permutation de la table 1.3.

59	50	42	34	26	18	10	14
1	58	50	42	34	16	18	56
10	2	59	51	43	35	27	11
19	11	3	60	52	44	36	22
63	55	47	39	31	23	15	43
7	62	54	46	38	30	22	35
14	6	61	53	45	37	29	24
21	13	5	28	20	12	4	12

TABLE 1.3 – Matrice de permutations initiale.

- On commence par le nombre 59 dans la première case dans la matrice, on calcule les bits de message (M) jusqu'à ce que nous arrivons au Bit de numéro 59. le rang du Bit 59 est par la suite changé à la première place.

- La même chose sera faite pour le nombre 49 dans la deuxième case et ainsi de suite jusqu'à le dernier nombre et obtenu un message (Mp) de 64 Bits .

2. **Rounds 1...16**

Les étapes présentées sur le schéma de la figure 1.4 sont appliquées sur le message produit par la règle précédente tel que :

- Le message résultant du l'opération de Initial Permutation est divisé en deux partie ML et MR.

- Expention Permutation est de même principe que celui de la permutation choice-1 et choice-2 , mais elle augmente le nombre de bits par 16 Bits qui devient 48 Bits.

- le message de taille 48 Bits est soumis à l'opération XOR avec Sub-Key1.

- le message de taille 48 Bits est soumis à l'opération XOR avec Sub-Key1.

- dans cette étape le message MR obtenu à l'aide de l'opération XOR est soumis à l'opération S-BOX en réduisant la taille du message à 32 Bits.

- on fait l'opération permutation expliquée précédemment.

- l'opération XOR est ensuite appliquée entre le message produit et ML(32 Bits) ,ce qui permet d'avoir le message MR-1 (32 Bits).

3. **Inverse Initial Permutation**

   - Le message résultant de l'opération"32 Bits Swap" subit une permutation en utilisant la matrice inverse de celle utilisée dans l'opération "initial permutation".

1.2.3 A.E.S. - Advanced Encryption Standard

La progression de la puissance des ordinateurs a causé la mort du DES. Ce dernier n'est plus jamais utilisé lorsque la sécurité demandée est forte (utilisation militaire, documents "secrets", etc.).

En Janvier 1997, la NIST (National Institute of Standards and Technology) lance un appel d'offre international pour remplacer le vieillissant DES : il en résulte 15 propositions.

En octobre 2000 la NIST élit Rijndael comme nouveau standard qu'on nomme aussi AES (Advanced Encryption Standard).

Rijndael, du nom condensé de ses concepteurs Rijmen et Daemen, est un algorithme de chiffrement par blocs à plusieurs tours similaire à DES mais avec une taille de blocs et de clefs supérieures et variables, choisis entre 128, 196 et 256 bits.

AES est un sous-ensemble de Rijndael, il ne travaille qu'avec des blocs de 128 bits. La différence entre AES-128, AES-192 et AES-256 , c'est la longueur de la clé : 128, 192 ou 256 bits.

Des blocs d'octets sont organisés sous forme matricielle, selon un modèle illustré dans la figure 1.6. Cette matrice aura nécessairement 4 lignes et un nombre de colonnes fonction de la taille choisie du bloc. On note Nb le nombre de colonnes dans une matrice pour écrire un bloc d'octets, et Nk ce même nombre pour une clé [1].

1.2.4 Principe d'algorithme A.E.S

Nous décrivons ici le fonctionnement de la version 128 bits de l'AES. Soit un clair de 128 bits M = (m_0, . . . ,m_{15}) et une clef de chiffrement de 128 bits K = (k_0, . . . ,k_{15}) l'AES-128 calcule un chiffré C = (c_0, . . . , c_{15}), par la méthode représentée en Figure 1.7, en débutant par une première opération **XOR** entre M et K puis en en injectant le résultat, S0, dans un cycle de 10 tours. Pour r = 1, . . . , 9, chaque tour d'AES-128 **r** calcule son état de sortie S_r en appliquant successivement quatre transformations :

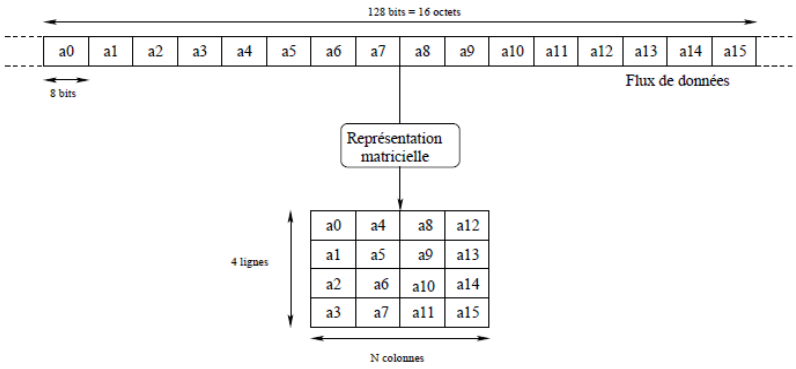


FIGURE 1.6 – Représentation matricielle d'un bloc de 16 octets.[1]

1. **SubBytes :**Est une substitution lors de laquelle chaque octet est remplacé par un autre octet choisi dans une table particulière (une Boîte-S).
2. **ShiftRows :**Est une étape de transposition où chaque élément de la matrice est décale cycliquement vers la gauche.
3. **MixColumns :**Calcule chaque colonne de sortie comme le produit d'une matrice constante et de la colonne d'entrée correspondante
4. **AddRoundKey :**C'est un simple **XOR** des clés. Il s'agit d'additionner des sous-clés aux sous-blocs correspondants.
5. **La diversification de la clef dans AES :** Cette étape, notée **KeyExpansion**, permet de diversifier la clef de chiffrement **K**, la clé sera découpée en sous-clés **ki**. Le nombre de sous-blocs **ki** dépendra bien de bloc du message.

Le chiffré est finalement défini comme étant la sortie du dixième et dernier tour qui a pour différence de ne pas comporter d'opération **MixColumns**. Le processus de chiffrement est résumé :

- S_0 = M XOR K_0 (K_0 = K)
- S_r = MixColumns(ShiftRows(SubBytes(Sr1))) XOR K_r (r = 1, . . . , 9)
- C = ShiftRows(SubBytes(S_9)) XOR $K - 10$

1.2.5 Avantages et limites de AES

Le AES comporte plusieurs avantages. En voici quelques-uns :

- Des performances très élevées (plus performant que le DES).
- il ne comprend pas d'opérations arithmétiques : ce sont uniquement des décalages et des XOR
- le nombre de rondes peut facilement être augmenté si c'est requis.

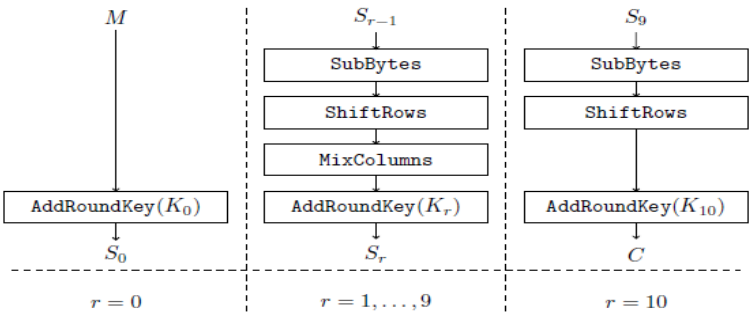


FIGURE 1.7 – Processus de chiffrement de l'AES.[17]

– il ne possède pas de clés faibles.

Il possède aussi quelques inconvénients et limites :

– le déchiffrement est plus difficile à implanter en "Smart Card".

– dans une réalisation matérielle, il y a peu de réutilisation des circuits de chiffrement pour effectuer le déchiffrement.[2]

1.3 Algorithmes de chiffrement par clef publique

1.3.1 RSA : Rivest - Shamir - Adleman

L'algorithme RSA a été décrit en 1977 par Ron Rivest, Adi Shamir et Len Adleman, d'où le sigle RSA. C'est un algorithme asymétrique de cryptographie à clé publique, très utilisé dans le commerce électronique, et plus généralement pour échanger des données confidentielles sur Internet. Il s'agit du système le plus connu et le plus largement répandu.

a) Création des clés

1. Choisir p et q, deux nombres premiers distincts.
2. Calculer leur produit n = p∗q, appelé module de chiffrement.
3. Calculer $\varphi(n) = (p - 1)(q - 1)$ (c'est la valeur de l'indicatrice d'Euler en n).
4. Choisir un entier naturel e premier avec $\varphi(n)$ et strictement inférieur à $\varphi(n)$, appelé exposant de chiffrement.
5. Calculer l'entier naturel d, inverse de e modulo $\varphi(n)$ (c.à.d. $ed \equiv 1(\ mod\ \varphi(n))$), et strictement inférieur à $\varphi(n)$, appelé exposant de déchiffrement.
6. d peut se calculer efficacement par l'algorithme d'Euclide étendu.

- Le couple (n,e) est la clé publique du chiffrement.
- Le couple (n,d) est la clé privée de déchiffrement.

b) Chiffrement du message

Si M est un entier naturel strictement inférieur à n représentant un message, alors le message chiffré sera représenté par :C $\equiv M^e mod n$

L'entier naturel C étant choisi strictement inférieur à n.

c) Déchiffrement du message

Pour déchiffrer C, on utilise d, l'inverse de e modulo (p − 1)(q − 1), et l'on retrouve le message clair M par : $M \equiv C^d mod n$

On a alors : C^d mod n ≡ $(M^e)^d$ mod n ≡ $(M_e)^d$ mod n

Exemple

- Ahmed choisit p = 17 et q = 19
  - On a : n = p ∗ q = 323, φ(n) = (p-1) * (q-1) = 288
- Elle choisit e = 5
  - On détermine, alors, que d =173 (inverse modulaire de e sur $Z_{\varphi(n)}$: 173∗5= 3∗288+1)
  - La clé publique est donc (5, 323) et la clé privée est (173, 323).
- Supposons que Khaled veut envoyer à Ahmed le message « BONJOUR » en se servant de la position des lettres dans l'alphabet pour les transformer en nombres. Cela donne :

B	O	N	J	O	U	R
2	15	14	10	15	21	18

- Après avoir chiffré en remplaçant chaque nombre b par (b^e mod n) on obtient le message que Khaled envoie à Ahmed :

2	15	14	10	15	21	18

- Pour le déchiffrement, Ahmed calcule pour chaque nombre b du message reçu b = (C^d mod n) :

2	15	14	10	15	21	18
B	O	N	J	O	U	R

qui est bien le message initial.

d) Attaques

Il existe deux approches pour attaquer le RSA :

1. Recherche par force brute de la clé (impossible étant donné la taille des données).
2. Attaques mathématiques (basées sur la difficulté de calculer φ(n), la factorisation du module n) :
   - Factoriser n = p∗q et par conséquent trouver φ(n) et puis d.
   - Déterminer φ(n) directement et trouver d.
   - Trouver d directement.

Conseils d'utilisation du RSA Pour garantir une bonne sécurité, il faut respecter certains règles telles que :

- Ne jamais utiliser de valeur n trop petite.
- N'utiliser que des clés fortes (p-1 et q-1 ont un grand facteur premier).
- Ne pas utiliser de n communs à plusieurs clés.

1.4 Algorithmes de hachage

1.4.1 MD5

Message Digest 5 est une fonction de hachage cryptographique conçue par Ronald Rivest, c'est le dernier d'une série (MD2, MD4), cet algorithme est utilisé pour produire une empreinte ou une signature de 128 bits d'un fichier ou d'un message.[2] L'implémentation de l'algorithme se faire par les étapes suivantes :

a) **Préparation du message**

MD5 travaille avec un message de taille variable. Le message est divisé en blocs de 512 bits comme illustré à la figure 1.8 on applique un complétion de manière à avoir un message dont la longueur est un multiple de 512. La complétion se présente comme suit :

- On ajoute un 1 à la fin du message.
- On ajoute une séquence de '0' (le nombre de zéros dépend de la longueur du padding nécessaire).
- On ajoute la longueur réelle du message (64 bits) après les 448 bits. En conséquence, la taille totale de dernier bloc atteint 512 bits.

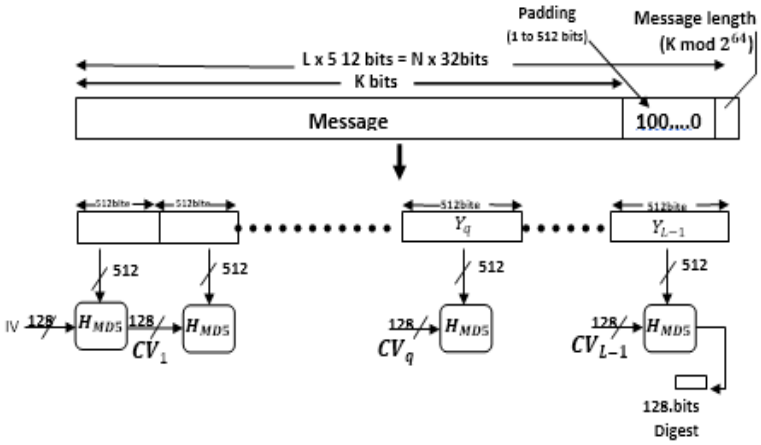


FIGURE 1.8 – Vue d'ensemble du MD5.[2]

b) **Fonction de compression**

Le calcule de l'empreinte s'appuie sur des registres A, B, C, D de longueur de 32 bits. Pour le premier bloc de 512 bits, ces registres sont initialisés avec les valeurs suivants (vecteur d'initialisation) en hexadécimal.

- A = 67452301
- B = EFCDAB89
- C = 98BADCFE
- D = 10325476

Pour les blocs suivants de 512 bits, ces registres sont initialisés à partir des valeurs obtenues pour le bloc précédent de 512 bits. Lorsque l'ensemble des blocs de 512 ont été traités, l'empreinte finale correspond à la concaténation des valeurs des registres A, B, C, D (Voir la figure 1.9).

Les opération sur un bloc de 512 bits se décomposent en 04 étapes, elle même subdivisées de 16 itérations élémentaire basées sur une fonction qui varie selon l'étape. Les quatre fonctions définies sont les suivant :

- Etape 1 : F(B,C,D) = (B $\wedge$ C) $\vee$ ($\neg$B$\wedge$ D)
- Etape 2 : G(B,C,D) = (B $\wedge$ D) $\vee$ ($\neg$D $\wedge$ C)
- Etape 3 : H(B,C,D) = B $\oplus$ C $\oplus$ D
- Etape 4 : I(B,C,D) = C $\oplus$ ($\neg$D $\oplus$ B)

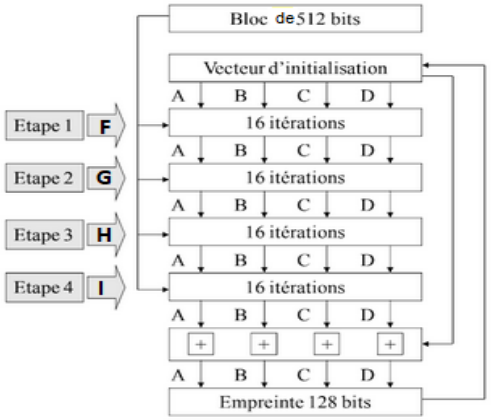


FIGURE 1.9 – L'algorithme MD5.

Chaque itération est décrite a la forme suivant :
(A,B,C,D)⟵(D,((A+g(B,C,D)+X[k]+ T[i])<<< s + B),B,C) où :

- Les lettres a, b, c, d se rapportent aux 4 registres, mais sont utilisés selon des permutations variables.
- g(b, c, d) est une fonction non-linéaire différente dans chaque étape (notée F,G, H et I dans la figure 1.9).
- X[K] = K-éme en bloc de 512 bit.
- T[i] est une valeur déterminer par la formule suivante : T[i]= E(abs(sin(i+1))∗ 232) où **E** désigne la partie entière et (i+1) est exprimer en radian.
- (<<<s) représente un décalage circulaire à gauche de s bits.

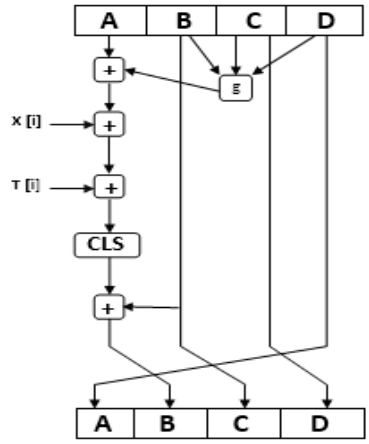


FIGURE 1.10 – Le i-éme tour dans MD5 (0≤ i ≤ 63).[2]

1.4.2 SHA-1

SHA-1 fut publié en 1995 par le NIST sous la forme d'une norme. La fonction de compression de SHA-1 utilise des blocs B de b = 512 bits pour produire une empreinte de n = 160 bits. Comme pour MD5, le bloc B est d'abord découpé en 16 sous-blocs Bi ($0 \leq i \leq 15$) de 32 bits chacun, qui sont ensuite étendus en 80 nouveaux blocs Wi ($0 \leq i \leq 79$).
L'algorithme travaille donc sur un état de 160 bits subdivisés en 5 registres, de 32 bits chacun : A, B, C, D et E (initialisés au début avec des constantes).

Les tours sont divisées en 4 étapes comme l'algorithme MD5, (voir figure 1.9) où la fonction F prendra les valeurs successives suivantes :

1. Étape 1 : F = (B $\bigwedge$ C) $\vee$ ($\neg$B $\bigwedge$ D)
2. Étape 2 : F = B $\oplus$ C $\oplus$ D
3. Étape 3 : F = (B $\bigwedge$ C) $\oplus$ (B $\bigwedge$ D)$\oplus$(C $\bigwedge$ D)
4. Étape 4 : F = B $\oplus$ C $\oplus$ D

Chaque étape comprend 20 tours qui manipulent ainsi les 5 registres :
(A,B,C,D,E)$\longleftarrow$((E+f(B,C,D)+(A << 5)+ W_t + K_t)A,(B << 30),C,D)

Où A,B,C,D,E se rapportent aux 5 registres, t est le numéro de l'itération, f(B, C, D) est une fonction primitive non-linéaire d'un étape, W_t est dérivé du bloc de message(32 bits) et K_t est une valeur additive. Le détail d'un tour SHA-1 est fourni dans la figure 1.11. Le calcul

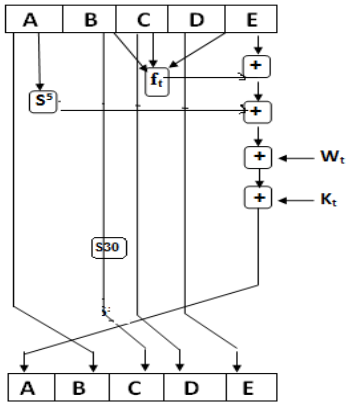


FIGURE 1.11 – Le i-ème tour dans SHA-1 ($0 \leq i \leq 79$).[2]

des W_t, dérivés des blocs de messages est donné à la figure 1.12. Pour ce calcul, on applique la formule suivante :

- $0 \leq 16$: les 16 premiers valeur de bloc.
- $16 \leq t$: $W_t = (W(t-16) \oplus W(t-14) \oplus W(t-8) \oplus W(t-3) <<< 1)$

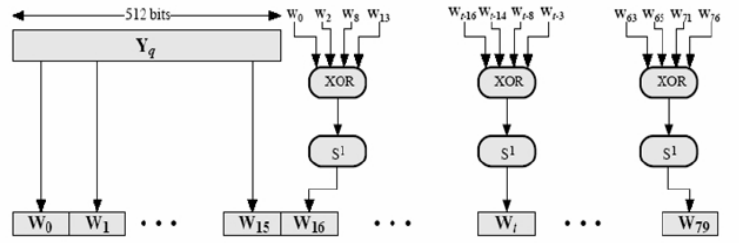


FIGURE 1.12 – Calcul du W_t dérivé.[2]

1.4.3 SHA-256

SHA-256 est une amélioration plus robuste (et supportant une taille d'empreinte plus grande) que SHA-1 publiée en 2000. Elle se base sur un chiffrement à clef secrète par bloc (SHACAL-2) et utilise des blocs B de b = 512 bits pour produire une empreinte de n = 256 bits.

Sécurité de SHA-256 : La taille d'empreinte dans SHA-256 est de 256 bits. La recherche de collisions par force brute (attaque de Yuval) nécessite 2128 calculs d'empreintes. à l'heure actuelle, il n'y a pas d'attaque efficace connue contre SHA-256 qui est donc considérée comme encore sûre.[2]

1.5 Authentification et intégrité des données

1.5.1 HMAC - Message Authentication Code

HMAC (Hashed-Based Message Authentication Code) est une alternative de l'algorithme MAC [1], consistant à protéger l'intégrité d'un message ainsi que l'authenticité de l'expéditeur, par l'utilisation d'un mécanisme de protection de l'intégrité qu'est la fonction de Hash. Le processus d'authentification de l'expéditeur suppose le partage d'une clé secrète partagée.

Soit H une telle fonction, K le secret et M le message à protéger. H travaille sur des blocs de longueur b octets (64 en général) et génère une empreinte de longueur l octets (16 pour MD5, 20 pour SHA et RIPE-MD-160). Il est conseillé d'utiliser un secret de taille au moins égale à l octets. On définit deux chaînes, ipad (inner padding data) et opad (outer padding data), de la façon suivante : ipad = l'octet 0x36 répété b fois, opad = l'octet 0x5C répété b

1. Un code d'authentification de message (MAC, Message Authentication Code) est un code accompagnant des données dans le but d'assurer l'authenticité de ces dernières, en permettant de vérifier qu'elles n'ont subi aucune modification, après une transmission.

fois. Le HMAC se calcule alors suivant la formule suivante : HMAC(K,m) = H ((K $\oplus$ opad) $\oplus$ H ((K $\oplus$ ipad) $\|$ m)).[4]

1.5.2 Signatures électronique

Les algorithmes de chiffrement asymétrique sont basés sur le partage entre les différents utilisateurs d'une clé publique. Toutefois ce mode de partage a une grande lacune : rien ne garantit que la clé est bien celle de l'utilisateur a qui elle est associée. En effet un pirate peut corrompre la clé publique en la remplaçant par sa clé publique. Ainsi, le pirate sera en mesure de déchiffrer tous les messages. Un certificat permet d'associer une clé publique à une entité afin d'en assurer la validité, délivrée par un organisme appelé autorité de certification(CA).

a) **Autorité(s) de certification**

Une autorité de certification (CA) est une entitée de confiance qui se charge de délivrer des certificats. Elle peut être est elle même reconnus/certifiée (elle possède un certificat) par une autorité supérieure, ou bien par elle même(son certificat est auto-signé). Le niveau de confiance d'une CA dépend des mécanismes pratiques qu'elle met en œuvre pour certifier (signer des certificats).[5]

b) **Certificat électronique**

Est une carte d'identité de la clé publique, délivrée par une autorité de certification. Il contient un ensemble des informations d'identification permettant aux clients et les serveurs de prouver l'authenticité de leur identité en ligne. Les certificats sont des petits fichiers divisés en deux parties :

- La partie contenant les informations.
- La partie contenant la signature de l'autorité de certification.

la fonction de signature doit répondre à plusieurs critères :

- Authentique : L'identité du signataire doit pouvoir être retrouvée de manière certaine.
- Infalsifiable : La signature ne peut pas être falsifiée. Quelqu'un ne peut se faire passer pour un autre.
- Inaltérable : Un document signé est inaltérable. Une fois qu'il est signé, on ne peut plus le modifier.
- Non réutilisable : La signature n'est pas réutilisable. Elle fait partie du document signé et ne peut être déplacée sur un autre document.

On distingue différents types de certificats selon le niveau de signature :

- Les certificats auto-signés sont des certificats à usage interne. Signés par un serveur local, ce type permet de garantir la confidentialité des échanges au sein d'une organisation, par exemple pour le besoin d'un intranet. Il est ainsi possible d'effectuer une authentification des utilisateurs grâce à des certificats auto-signés.
- Les certificats signés par un organisme de certification sont nécessaires lorsqu'il s'agit d'assurer la sécurité des échanges avec des utilisateurs anonymes, par exemple dans le cas d'un site web sécurisé accessible au grand public. Le certificateur tiers permet d'assurer à l'utilisateur que le certificat appartient bien à l'organisation à laquelle il est déclaré appartenir.

La structure des certificats est normalisée par le standard X.509, qui contient les informations suivante :

La version de X.509 à laquelle le certificat correspond
Le numéro de série du certificat
L'algorithme de chiffrement utilisé pour signer le certificat
Le nom (DN, pour Distinguished Name) de l'autorité de certification émettrice
La date de début de validité du certificat
La date de fin de validité du certificat
L'objet de l'utilisation de la clé publique
La clé publique du propriétaire du certificat.
La signature de l'émetteur du certificat (thumbprint).

TABLE 1.4 – Les informations d'un certificat.

L'ensemble de ces informations (informations + clé publique du demandeur) est signé par l'autorité de certification, (Voir figure 1.13). La Procédure de signature suivi par les étapes suivante :

- Encrypter les données avec SHA-256 qui produit un hash de 256 bit.
- Il chiffre le résultat du hash avec a clé privée de l'autorité de certification
- Le résultat du chiffrement c'est la signature.

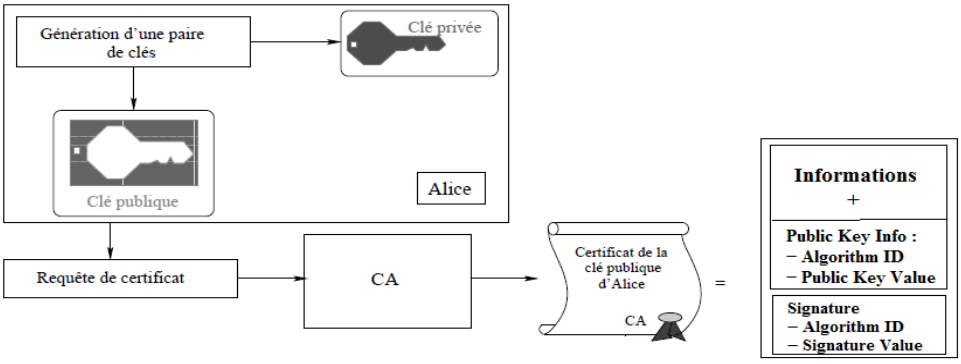


FIGURE 1.13 – Principe de création des certificats.[1]

Lorsqu'un utilisateur désire communiquer avec une autre personne, il lui suffit de se procurer le certificat du destinataire (Voir figure 1.14). Il est donc possible de vérifier la validité du message en appliquant les étapes suivant :

- Il hash votre certificat avec du SHA-256 (défini dans le certificat).
- Il déchiffre la signature du certificat avec la clé publique récupérée dans le certificat.
- Il compare le résultat du hash avec le résultat du déchiffrement de la signature.

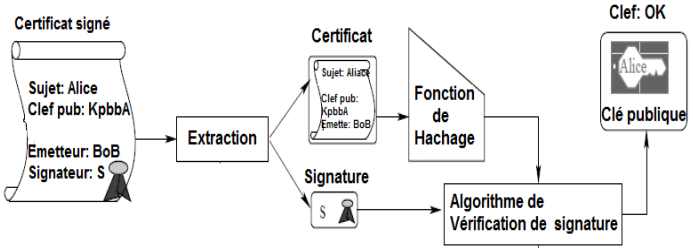


FIGURE 1.14 – Vérification d'identité de certificate.[1]

c) **Usages du certificat** Les certificats servent principalement dans deux types de contextes :

- Le certificat client, stocké sur le poste de travail de l'utilisateur ou embarqué dans un conteneur tel qu'une carte à puce, permet d'identifier un utilisateur . Dans la plupart des scénarios il est transmis au serveur lors d'une connexion, qui affecte des droits en fonction de l'accréditation de l'utilisateur.
- Le certificat serveur, installé sur un serveur web permet d'assurer le lien entre le service et le propriétaire du service. Dans le cas d'un site web, il permet de garantir que l'URL et en particulier le domaine de la page web appartiennent bien à telle ou telle entreprise. Par ailleurs il permet de sécuriser les transactions avec les utilisateurs grâce au protocole SSL.

II. Réseau privé virtuel (VPN)

1.6 Présentation réseau privé virtuel (VPN)

1.6.1 Définition d'un VPN

Un VPN est un tunnel sécurisé permettant la communication entre deux entités y compris au travers de réseaux peu sûrs comme peut l'être le réseau Internet. Les VPN ont pour objectif de contribuer à la sécurisation des échanges de données privées, sensible, sur les réseaux publics.

1.6.2 Principe de fonctionnement

Un réseau VPN repose sur un protocole appelé "protocole de tunneling". Ce protocole permet de faire circuler les informations de façon cryptée d'un bout à l'autre du tunnel. Ainsi, les profs ou les étudiants ont l'impression de se connecter directement sur le réseau de leur université.

Le principe de tunneling consiste à construire un chemin virtuel après avoir identifié l'émetteur et le destinataire. Par la suite, la source chiffre les données et les achemine en empruntant Ce chemin virtuel. Afin d'assurer un accès aisé et peu coûteux aux intranets ou aux extranets d'université, les réseaux privés virtuels d'accès simulent un réseau privé, alors qu'ils utilisent en réalité une infrastructure d'accès partagée, comme Internet.

Les données à transmettre peuvent être prises en charge par un protocole différent d'Ip. Dans Ce cas, le protocole de tunneling encapsule les données en ajoutant une en-tête. Le tunneling est l'ensemble des processus d'encapsulation, de transmission et de désencapsulation.[13]

1.6.3 Intérêt d'un VPN

La mise en place d'un réseau privé virtuel permet de connecter de façon sécurisée des ordinateurs distants au travers d'une liaison non fiable (Internet), comme s'ils étaient sur le même réseau local.

Ce procédé est utilisé par de nombreuses entreprises afin de permettre à leurs utilisateurs de se connecter au réseau d'entreprise hors de leurs lieux de travail. Nous pouvons facilement imaginer un grand nombre d'applications possible :

– Les connexions VPN offrent un accès au réseau local (d'entreprise) à distance et de façon sécurisée pour les travailleurs nomades.

- Les connexions VPN permettent d'administrer efficacement et de manière sécurisé un réseau local à partir d'une machine distante.
- Les connexions VPN permettent aux utilisateurs qui travaillent à domicile ou depuis d'autres sites distants d'accéder à distance à un serveur d'entreprise par l'intermédiaire d'une infrastructure de réseau public, telle qu'Internet.
- Les connexions VPN permettent également aux entreprises de disposer de connexions routées partagées avec d'autres entreprises sur un réseau public, tel qu'Internet, et de continuer à disposer de communications sécurisées, pour relier, par exemple des bureaux éloignés géographiquement.

Les connexions VPN permettent de partager des fichiers et programmes de manière sécurisée entre les machines distante.

1.7 Les fonctionnalités d'un réseau privé virtuel (VPN)

Un système de VPN doit pouvoir mettre en oeuvre les fonctionnalités suivantes :

- **Authentification d'utilisateur :** Seuls les utilisateurs autorisés doivent pouvoir s'identifier sur le réseau virtuel. De plus, un historique des connexions et des actions effectuées sur le réseau doit être conservé.
- **Gestion d'adresses :** Chaque client sur le réseau doit avoir une adresse propre. Cette adresse doit rester confidentielle. Un nouveau client doit pourvoir se connecter facilement au réseau et recevoir une adresse.

- **Cryptage des données :** Lors de leurs transports sur le réseau public les données doivent être protégées par un cryptage efficace.
- **Gestion de clés :**Les clés de cryptage pour le client et le serveur doivent pouvoir être générées et régénérées.

1.8 Protocoles utilisée pour réaliser une connexion VPN

Les VPN s'appuient, comme la plupart des technologies réseaux, sur des protocoles. Plusieurs protocoles sont utilisés dans les technologies de VPN. Certains d'entre eux visent uniquement à établir un tunnel, d'autres y ajoutent la composante sécurité. Les différents protocoles utilisés sont [14] :

1.8.1 PPTP (Point-to-Point Tunneling Protocol)

Est un protocole développé par US Robotics. Il s'agit un protocole d'encapsulation de bout en bout sur IP qui permet la mise en place de VPN au-dessus d'un réseau public.

- Avantages
  - Le protocole est intégré dans la plupart des OS donc son utilisation ne nécessite pas l'installation d'une application spécifique.
  - Le protocole PPTP est très simple à utiliser et à mettre en place.
- Inconvénients
  - Ce protocole VPN peut être facilement bloqué par les fournisseurs d'accès Internet.
  - Le protocole PPTP est mal sécurisé.

1.8.2 L2TP (Layer Two Tunneling Protocol)

Est un protocole standard de tunnelisation très proche de PPTP utilise deux couches d'encapsulation. La première couche utilise un datagramme IP comme PPTP ainsi qu'un L2TP et un en-tête de protocole de datagramme utilisateur(UDP). Il est ensuite encapsulé à nouveau en utilisant un protocole Internet Security (IPsec) Encapsulation Security Payload . Les données est sécurisé par des clés de cryptage à l'aide de Data Encryption Standard.

- Avantages :
  - Le protocole L2TP/IPsec offre une bonne protection (l'authentification,chiffrement, l'intégrité des données.
  - l'utilisation de UDP rend L2TP plus rapide.

- Inconvénients :
  - Il faut beaucoup de temps nécessaire de configuration.
  - Plus lent que OpenVPN.

1.8.3 IPSEC (Internet Protocol Security)

Est un protocole de niveau 3, fournir différents services de sécurité. Fait appel à deux mécanismes de sécurité pour le trafic IP : Le premier, AH, pour Authentification Header vise à assurer l'intégrité et l'authenticité des datagrammes IP. Il ne fournit par contre aucune confidentialité : les données fournies et transmises par Ce "protocole" ne sont pas encodées et Le second, ESP, pour Encapsulating Security Payload peut aussi permettre l'authentification des données mais est principalement utilisé pour le cryptage des informations.

- Avantages :

  - Sécurité totale grâce à la combinaison de certificats numériques et de Pki pour l'authentification ainsi qu'à une série d'options de cryptage, DES et AES notamment.
- Inconvénients :
  - Pas d'authentification des utilisateurs.
  - Lourdeur des opérations.

1.8.4 SSL/TLS

A l'heure actuelle, SSL (Secure Sockets Layer) et TLS (Transport Layer Security) une très bonne solution de tunnelisation. C'est une famille qui prend de plus en plus de place dans les implémentations de tunnels et nombreux sont maintenant les pare-feu proposant des tunnels SSL.

L'origine de ces protocoles remonte à un des navigateurs historiques : Netscape. Les équipes de développement de ce produit ont cherché dès 1994 à mettre en place un canal sécurisé afin de permettre l'échange de données confidentielles (authentification, carte bancaire...) entre le navigateur et le serveur. C'est ainsi qu'ont été successivement développées la version SSLv1 peu diffusée, puis la version SSL v2. Mais cette version souffrant d'un certain nombre de défauts importants en matière de sécurité a été remplacée fin 1995 par la v3.

En 1996 l'IETF (Internet Engineering Task Force) souhaita normaliser un protocole de type SSL. C'est ainsi que fut mis en place un groupe de travail TLS (Transport Layer Security). Après diverses péripéties le groupe décida de publier le résultat de ses travaux sous le nom TLS v1.0 dans le RFC 2246 (janvier 1999). En avril 2006 la version TLS v1.1 a été publiée dans la RFC [2] 4346, puis en août 2008 la v1.2 dans la RFC 5246. Les mécanismes fournis par ces protocoles sont : l'authentification du serveur, l'authentification du client et l'authentification du serveur. le mécanisme d'établissement d'un tunnel SSL entre un client et un serveur présenter dans dernié chapitre.

SSL est un protocole de couche 7 utilise par une application par une application pour établir un canal de communication sécurité avec une autre application.

SSL a deux grandes fonctionnalités : l'authentification du serveur et de client à l'établissement de la connexion et le chiffrement des données durant la connexion.[3]

2. RFC «requests for commentssont» : une série numérotée de documents officiels décrivant les aspects et spécifications techniques de l'Internet

Conclusion

Ce chapitre nous a permis de prendre connaissance des différents concepts et généralités associés à la cryptographie et au réseau privé virtuel(VPN).

Dans le chapitre suivant on s'intéressera à la présentation des outils nécessaires à la configuration de l'environnement de travail et ceux nécessaire à la création d'un réseau VPN.

Chapitre 2

Les outils de réalisation

Introduction

Dans ce chapitre, nous allons présenter les outils utilisés dans notre Lab pour implémenter notre solution, notamment :

- VMware Workstation 14.
- EVE-ng.
- PFsense.

puis, nous présenterons le gestionnaire des conteneurs (Docker) qui est l'outil utilisé pour le téléchargement, la configuration et l'exécution de l'application Nextcloud.

2.1 VMware Workstation 14

Afin de créer une machine virtuelle capable d'exécuter son système d'exploitation et ses propres applications comme un ordinateur réel, on utilise le VMware Workstation 14 comme une solution professionnelle, puissante et complète qui permet de lancer notre machine virtuelle.

2.1.1 Définition du VMware Workstation

VMware Workstation 14 est la version station de travail du logiciel VMware. Il permet la création de plusieurs machines virtuelles, simule le fonctionnement de ces machines comme étaient des ordinateurs réels avec différent systèmes d'exploitation et les fait fonctionner simultanément. Il permet aussi de relier une telle machine à un réseau réele en l'attribiant sa propre adresse IP. La différence avec une machine réelle se trouve dans les perfermances qui dépendde celle de la machine utilisée pour créer cette machine virtualle.[8]

2.1.2 Les composants du réseau virtuel

Un réseau virtuel créé par VMware Workstation 14 se constitue de [7]

a) Commutateur virtuel : Comme un commutateur physique, permet de connecter les composants réseau entre eux. On peut créer jusqu'à Vingt commutateurs virtuels. Il est également possible de relier plusieurs machines virtuelles au même commutateur. Par défaut, certains commutateurs sont réservés aux configurations réseau de base (bridged, nat et host-only) comme suit :
   - vmnet0 pour le réseau bridge.
   - vmnet1 pour le réseau hôte uniquement.
   - vmnet8 pour le réseau NAT.

   Les autres réseaux disponibles seront appelées : vmnet2, vmnet3, ...vmnet19.

b) Le bridge : Installé pendant l'installation de VMware Workstation, il relie l'adaptateur de l'hôte virtuel de la machine virtuelle à l'adaptateur éthernet physique. Lorsque on crie une nouvelle machine virtuelle à l'aide de la mise en réseau bridge, le bridge est automatiquement se configuré.

c) L'adaptateur virtuel hôte : Est un adaptateur éthernet virtuel permet d'assurer la communication entre l'hôte et les machines virtuelles. Il est utilisé dans les configurations host-only et NAT.

d) Le périphérique NAT (Network Address Translation) : Configuré automatiquement durant l'installation du VMware Workstation. Il permet de connecter les machines virtuelles à un réseau externe quand on ne possède qu'une seule adresse IP sur le réseau physique et que cette adresse est utilisé par l'hôte.

e) Le serveur DHCP (Dynamic Host Configuration Protocol) : Fournit des adresses réseau aux machines virtuelles,hors les réseaux bridgées,à un réseau externe, par exemple, les configurations host-only et NAT.

f) L'adaptateur réseau virtuel : Lors de la création d'une machine virtuelle, un adaptateur réseau virtuel est installé sur l'hôte et permet d'utiliser tous les types des réseaux [8].

2.1.3 Les types de réseau VMware

VMware Workstation propose principalement trois types de réseau : le Bridge, le NAT et le Host-Only [7].

a) Le réseau ponté ou «bridged» :
   - La machine virtuelle est directement connectée sur le réseau physique.

- La machine a accès à tous les participants composants du réseau physique.
- La machine virtuelle est accessible aux autres machines sur le réseaux.
- La machine virtuelle peut offrir des services réseau comme tout participant sur le réseau physique.

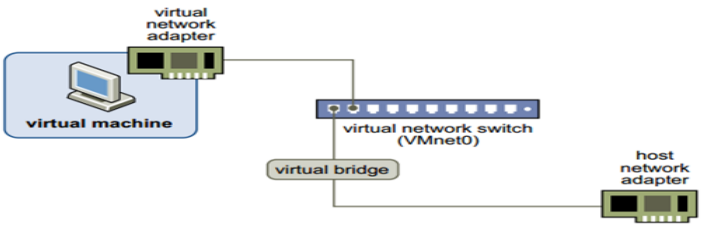


FIGURE 2.1 – Le réseau Bridge.

b) Le réseau NAT :

Le réseau NAT traduisant les adresses des machines virtuelles en adresses de la machine hôte. Lorsqu'une machine virtuelle envoie une demande d'accès à une ressource réseau, cette dernière interprète cette requête comme si elle provenait de la machine hôte. NAT utilise les ressources réseau de l'hôte pour se connecter au réseau externe.

Le périphérique NAT est connecté au commutateur virtuel vmnet8. Les machines virtuelles connectées au réseau NAT utilisent également ce commutateur. Le périphérique NAT attend l'arrivée de paquets des machines virtuelles sur le réseau virtuel vmnet8. Lorsqu'un paquet arrive, le périphérique NAT traduit l'adresse de la machine virtuelle en une adresse de l'hôte avant de transférer le paquet au réseau externe. Lorsque des données arrivent du réseau externe à la machine virtuelle sur le réseau privé, le périphérique NAT reçoit les données, remplace l'adresse réseau par celle de la machine virtuelle et transfère les données à la machine virtuelle sur le réseau virtuel.

Les machines virtuelles fonctionnant sur le réseau avec le périphérique NAT peuvent obtenir leur adresse IP de façon dynamique en envoyant des requêtes DHCP au serveur DHCP VMware, (Voir figure 2.2).[8]

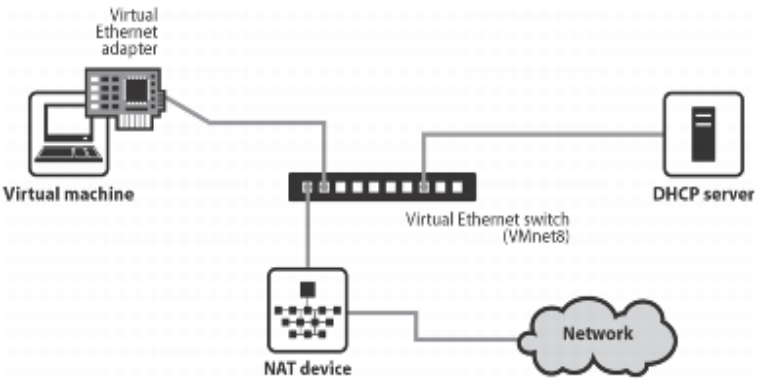


FIGURE 2.2 – Le réseau NAT.

c) Le réseau host-only :

- La configuration «host-only» fournit une connexion réseau entre la machine virtuelle et l'ordinateur hôte, au moyen d'un adaptateur éthernet virtuel.
- Les machines virtuelles et l'hôte sont connectés dans un réseau privé.
- Les machines virtuelles sont isolées et ne peuvent se connecter sur le réseau externe, ni sur internet.
- Les machines virtuelles obtiennent des adresses IP par le serveur DHCP VMware, (Voir figure 2.3).

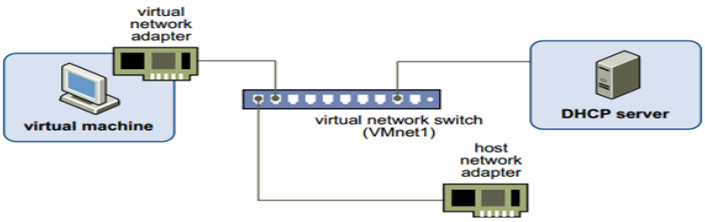


FIGURE 2.3 – Le réseau host-only.

2.2 EVE-ng

Pour construire un réseau virtuel interactif sur une interface graphique et observer son fonctionnement, on utilise EVE-ng.

2.2.1 Qu'est-ce que EVE-ng ?

EVE-ng (Emulated Virtual Environment - Next Generation)) est un émulateur permet de créer des réseaux virtuel, les tester et les connecter aux autres réseaux réels ou virtuels,

ce qui permet de réduire le coût, le temps de l'installation et conduire à la résolution des problèmes par la création des modèles et les teste par différents scénarios de sécurité.

2.2.2 Installation et configuration EVE-ng sur VMware Workstation

On créer une machine virtuelle EVE-ng sous VMware Workstation 14 à l'aide de l'image (.iso) téléchargée à partir du site web officiel (https ://www.eve-ng.net/downloads/eve-ng-2) en passant par les étapes d'installation exprimées dans le document (http ://www.eve-ng.net/images/EVE- COOK-BOOK-1.0.pdf). Pour le fonctionnement optimal, la machine EVE nécessite :

- Procésseur Intel VT-x / EPT ou un autre comme AMD-V.
- VMware Workstation 12.5 ou plus.
- La RAM doit être au minimum = 4Go.

2.2.3 Mise en place et créer un fichier lab en EVE

Après l'installation de la machine EVE, on doit connecter à son interface web. On saisi dans notre navigateur web l'url (http ://ip-machine-eve/) qu'il s'agit d'une page d'authentification permet d'afficher le login/mot de passe qui sont par défaut «admin/eve».

Sur la fenêtre principale de l'interface utilisateur, la machine EVE affiche un gestionnaire de fichiers «file manager». Pour créer un nouveau projet, un nouveau nom du dossier est entré dans la zone de texte en un cliquant sur le bouton**add new foldre**. Dans cet exemple, le dossier crié est nommé VPN.

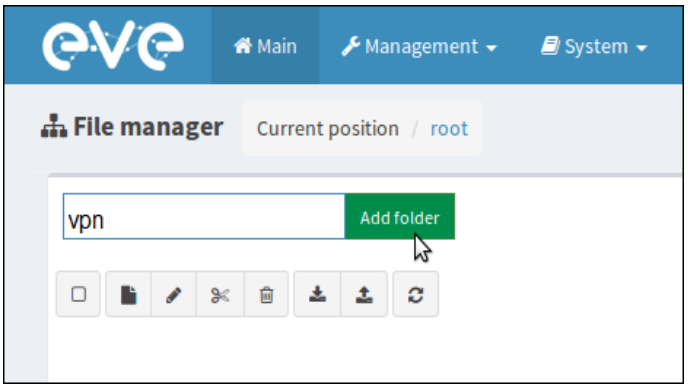


FIGURE 2.4 – Créer un nouveau projet.

- Après l'ouverture du dossier, nous cliquons sur l'icône ajouter un nouveau laboratoire **"add new lab"**.

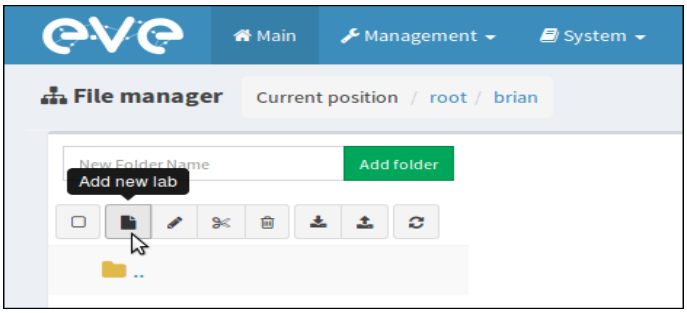


FIGURE 2.5 – Ajouter nouveau fichier lab.

- Une boîte de dialogue s'affiche pour demander des informations sur le laboratoire. Ici, nous devons entrer le nom du laboratoire et la version, des champs facultatifs sont aussi disponibles comme le champ description qui permet de fournir des informations aux utilisateurs ayant le droit de manipuler le fichier crié.

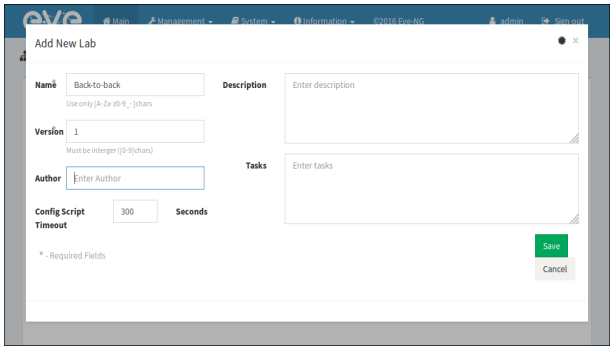


FIGURE 2.6 – Créer un fichier lab.

- Le fichier de laboratoire sera créé après qu'on clique sur le bouton enregistrer. Vous trouverez ci-dessous un ensemble de boutons qui permet à un utilisateur de gérer le laboratoire, l'utilisateur peut ouvrir le fichier lab, modifier ses informations ou supprimer le fichier.

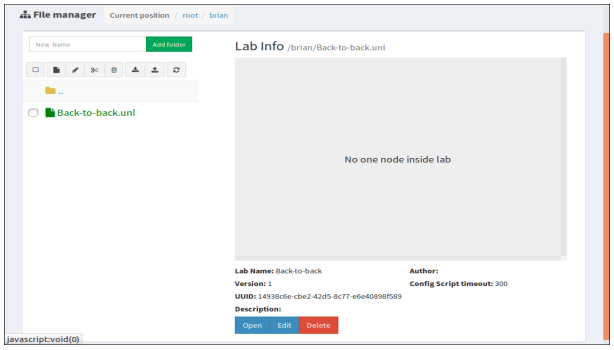


FIGURE 2.7 – Gérer le fichier lab.

- On clique sur le bouton Ouvrir, une interface graphique vide s'affiche. La barre d'outils à gauche est utilisée pour créer et gérer les éléments de la topologie.

FIGURE 2.8 – L'interface graphique de lab.

- Si nous voulons ajouter un équipement, il faut visualiser la liste complète des équipements réseaux supportés, qui est vide,par défaut on va donc ajouter quelques équipements pour qu'ils puissent être disponibles sur la liste
- Pour pouvoir ajouter certains équipements réseaux, comme un hôte Windows, un hôte linux ou des routeurs Cisco, il faut d'abord télécharger ses fichiers images qui seront importés dans la machine EVE en utilisant l'outil WInscap.

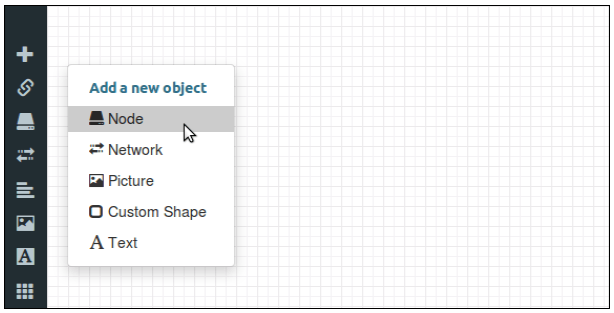


FIGURE 2.9 – Ajouter un équipement réseau.

2.2.4 Interfaces cloud et types de réseaux EVE-ng

Le réseau virtuel créé dans le fichier lab peut communiquer avec un autre réseau ou un périphérique externe par les interfaces cloud selon le mode de connexion qui est bridge ou NAT.

Pnet est le nom de l'interface du pont à l'intérieur de la machine EVE. Le nom «cloud» est utilisé comme alias de pnet dans l'interface graphique dans laquelle on a simule notre topologie. L'interface cloud/pnet est reliée à la carte réseau de la machine EVE. Le tableau suivant montre les interfaces cloud qu'on peut créer dans Eve-ng et celles correspondantes on VMWare Workstation [9].

Nom du Cloud	Nom de l'interface EVE (à l'intérieur)	Type	Interface correspondante dans VMware	Remarque
Cloud0	Pnet0	Bridge	Network Adapter 1	Cloud0 / pnet0 est relié à la port Ethernet EVE principal. On lui attribue une adresse IP de gestion utilisée pour l'accès à l'interface graphique Web. Peut être utilisé dans la gestion des réseaux dans les laboratoires.
Cloud1	Pnet1	Bridge	Network Adapter 2	Cloud1 peut être relié à la deuxième port Ethernet EVE pour établir la connexion à un autre réseau ou périphérique. L'adresse IP n'a pas besoin d'être configurée.
Cloud2-9	Pnet2-9	Bridge	Network Adapter 3-9	Identique à Cloud1

TABLE 2.1 – Interfaces cloud et ses correspondantes dans Vmware.

2.3 Etude des différents pare-feux supportant VPN

2.3.1 Aperçu des Pares-feux

Pour assurer une bonne protection de données de notre réseau local des utilisateurs étrangers et assurer une connexion distante sécurisée, on utilise des pares-feux.

Dans ce qui suit, nous allons présenter quelques pares-feux Open source les plus utilisé :

a) **IPcop :[11]**

Est une distribution linux (Open Source) gratuite, basée sur Linux *From Scratch*[1], et il est téléchargeable sous forme d'un fichier image. Dans le but est de protéger un réseau des menaces Internet, surveiller son fonctionnement et fournir une connexion fiable entre différents réseaux.

Ses principaux Caractéristiques sont :

- Complet (routeur, DHCP, Firewall, proxy, DNS, VPN).
- l'administration facile depuis un navigateur.
- Possibilité d'ajouter des modules (extensible).

1. Linux From Scratch(LFS) : Le projet Linux From Scratch est un livre qui décrit les diverses étapes pour créer un système Linux à partir des sources logiciels et permet de comprendre le fonctionnement de noyau du système d'exploitation Linux.

- IPcop peut gérer jusqu'à 4 réseaux différents, classé par colleurs comme suite :
  - Interface Verte : Correspond au réseau local protégé par IPCOP. Ce réseau a le droit d'accès aux 3 autres réseaux (sauf paramétrage spéciaux : URL Filter qui limite l'accès au Web).
  - Interface Orange(optionnelle) : Ce réseau est une sorte de DMZ (Demilitarized Zone = Zone Démilitarisée) qui permet de relier des serveurs au réseau Internet. Les ordinateurs de ce réseau ne peuvent pas accéder aux ordinateurs des interfaces Bleu et Verte sauf mise en place de règles explicites.
  - Interface Bleu(optionnelle) : C'est une interface spécifique aux réseaux sans fil. Elle permet aux ordinateurs connectés d'accéder au réseau rouge et orange sans accéder au réseau Vert.
  - Interface Rouge : Ce interface correspond au réseau Internet.

b) **ClearOS**[10] :

Anciennement appelé Clark Connect, est une distribution linux, basé sur CentOS et Red Hat Entreprise Linux, conçu pour une utilisation dans les petites et moyennes entreprises comme une passerelle réseau et le serveur de réseau avec une interface d'administration basé sur le Web.

Caractéristiques :

- Pare-feu stateful (iptable), la mise en réseau et de la sécurité.
- Proxy Web, avec le filtrage de contenu et antivirus (Squid, DansGuardian).
- Réseaux virtuels utilisant IPsec, L2TP et OpenVPN.

Le(s)limite(s) :

- ClearOS n'offre pas beaucoup de fonctionnalité dans sa version gratuite.

c) **OPNsense**[11] :

Est une distribution de pare-feu open source basé sur FreeBSD [2]. Débuté en janvier 2015 en tant que branche du pare-feu pfSense. Déploiements typiques sont les pares-feux à états de périmètre, routeurs, points d'accès sans fil, serveurs DHCP et DNS et points de terminaison VPN.

IL peut être configuré et mise a jour par le biais d'une interface Web, et ne nécessite aucune connaissance du système FreeBSD sous-jacent à gérer.

Caractéristiques :

- Complet(DHCP, VLAN, multi-NAT, multi-WAN, DNS dynamique...etc).

2. FreeBSD : Est un système d'exploitation UNIX libre. L'objectif du projet FreeBSD est de fournir un logiciel pour n'importe quelle utilisation, ce avec le moins de restrictions possibles.

- Réseaux virtuels utilisant IPsec, L2TP et OpenVPN.
- Filtrage par source/ de destination adresse IP et le protocole.
- Routage flexible.

Le(s)limite(s) :

- C'est un nouveau pare-feau qui n'est pas encore mature comme les autres.
- Manque de packages par rapport au PFsense.

d) **PFsense [11] :**

Packet Filter Sense est une distribution FreeBSD développée en 2004. L'objectif de départ est d'assurer les fonctions de pare-feu et de routeur, dant l'installation se fait facilement via une distribution dédiée et toutes les configurations peuvent se faire soit en ligne de commande (SSH) ou via l'interface web. PFsense est basé sur PF (packet filter), comme iptables sur GNU/Linux et il est réputé pour sa fiabilité. Distribution PFsense assure une évolution constante grâce à des mises à jour régulières dont l'installation est gérée automatiquement dans une partie au niveau du panneau d'administration. La sauvegarde et la restauration de configuration est disponible à travers une interface web.

2.3.2 Solution PFsense

Bien que nous n'ayons pas mis en pratique toutes ces solutions pour les comparer, l'étude théorique permet de retenir la dernier solustion puisque PFsense présente une plus grande assurance car la communauté des utilisateurs est trés actives, offre des fonctionnalités souvent trouvés firewalls commerciaux coûteuse et répondent à toutes les besoins :

- Solution riche et performante à moindre coût (basé sur des logiciels libres).
- Toute la configuration du système est stockée dans un fichier xml (/cf/conf/config.xml).
- Simplicité de l'activation / désactivation des régles de filtrage.
- il existe plusieurs packages disponibles permettent de mettre en place un réseau virtuel privé : un VPN client.
- Performances sont liées au matériel.

De plus PFsense présente une interface plus conviviale et une page principale en tableau de bord où l'on retrouve toutes les informations essentielles et que l'on peut modifier en fonction des besoins.

2.3.3 Installation de PFsense

Pour une mise en œuvre pratique nous allons décrit pas à pas les différentes étapes d'installation :

a) **Obtention de PFsense :**

La versions de pfSense sont téléchargeables dans la rubrique «download» du site www.pfSense.org.

b) **Pré requis :**

Via l'interface web de EVE en click sur l'onglet +**add on object** puis **Node**, après choisi le template PFsense et on crée une machine virtuelle dans la quelle PFsense sera installé. Pour cela on a besoin de la configuration suivant : interface(LAN, WAN, MGT, DMZ), un CPU et 2048 Go de RAM.

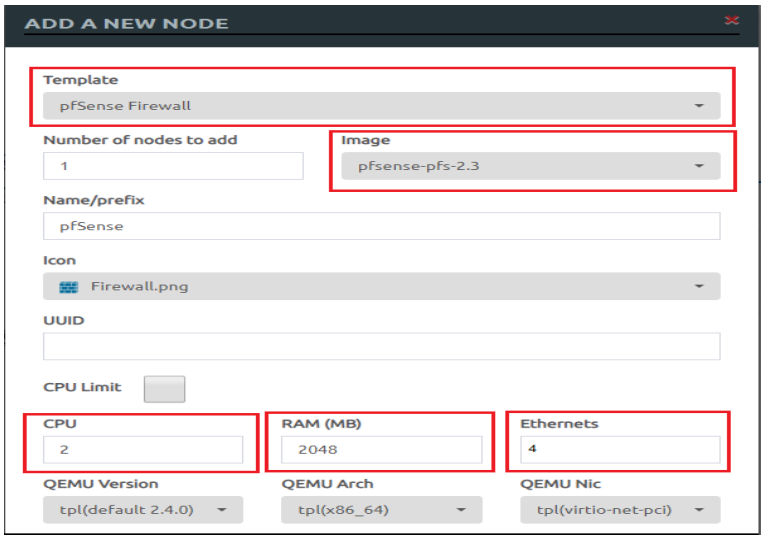


FIGURE 2.10 – Creér un machine virtuelle.

c) **Les étapes d'installation :**

- on click sur l'onglette "Start".

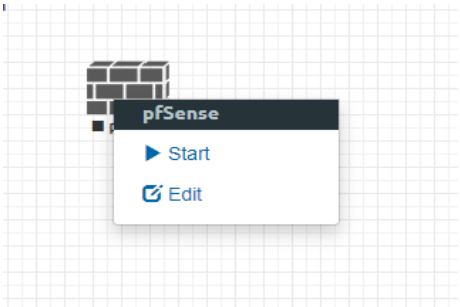


FIGURE 2.11 – Lancer l'installation de pfsense.

- Taper 1 pour choisir le boot PFsense ou laisser démarrer avec l'option par défaut.

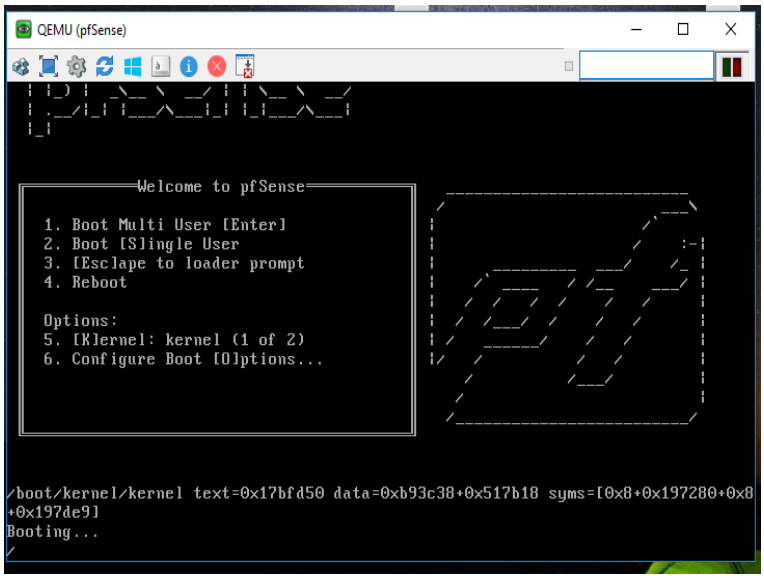


FIGURE 2.12 – Écran de démarrage de FreeBSD.

- Ici, on n'a pas encore configuré le VLAN, la réponse à la question de la figure sera "n".

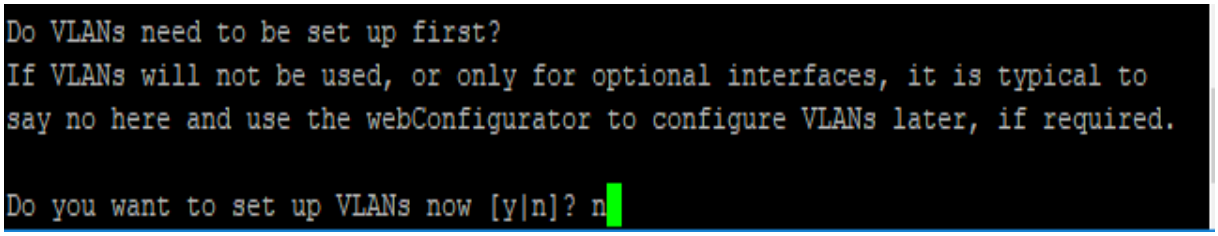


FIGURE 2.13 – Boite de dialogue pour la configuration de VLAN.

- Ensuite on détermine l'interface WAN parmi : vtnet0, vtnet1, vtnet2 et vtnet3. Sachent que,dans notre cas, "vtnet0" est notre interface WAN. Ensuite, on fait la même chose pour la carte réseau LAN, choisissant "vtnet1", et pareillement pour les autres interfaces.
FreeBSD résume l'attribution des cartes réseaux aux différentes interfaces et nous validons avec "y".

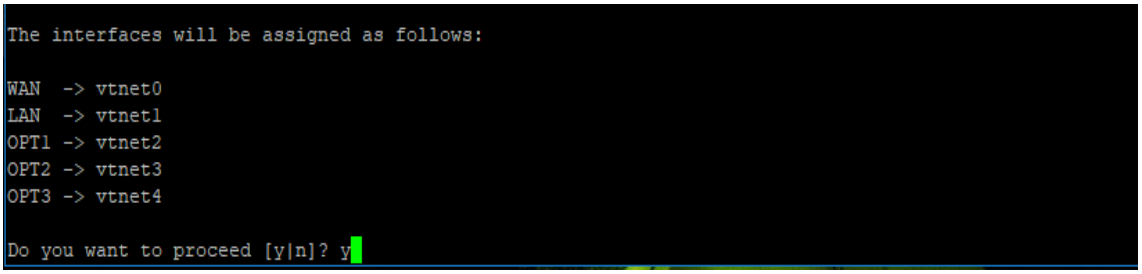


FIGURE 2.14 – Validation des noms d'interface.

- Une fois l'installation terminée, le menu de console PFsense apparaîtra. La plupart des options présents dans ce menu sont également disponibles via l'interface web. Le serveur DHCP fonctionne sur l'interface WAN, donc automatiquement avoir une adresse IP.

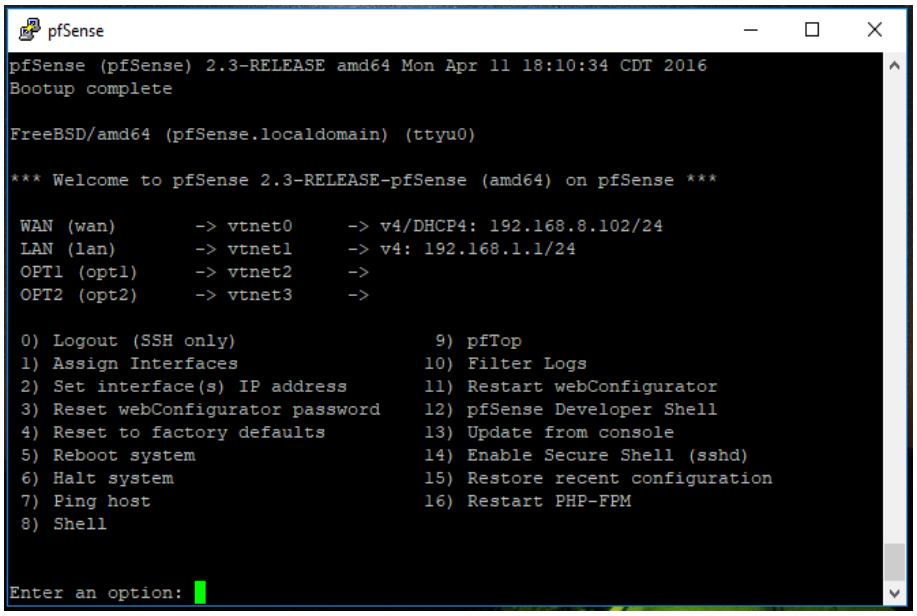


FIGURE 2.15 – Menu textuel de PFsense.

- Nous choisissant l'option 2 du menu afin de configurer une adresse IP pour l'interface LAN, en répondant aux questions de la configurer d'interfaces suivant :

- Entrez le numéro de l'interface que nous souhaitez configurer (en choisie : 2 pour l'interface LAN).
- Entrer la nouvelle adresse IPv4 LAN, par exemple 192.168.120.1
- Entrer le nouveau nombre de bits du sous-réseau LAN IPv4 (1 à 31). Exemple, 24 (équivalent à 255.255.255.0, c'est-à-dire 254 adresses IP).
- Dans la question suivante, une passerelle doit être définie. Ce n'est pas nécessaire pour une interface LAN, Appuyer simplement sur ENTER.
- Entrer la nouvelle adresse IPv6 du réseau local. Appuyer sur <ENTER>
- Vouloir revenir à HTTP en tant que protocole webConfigurator ? y

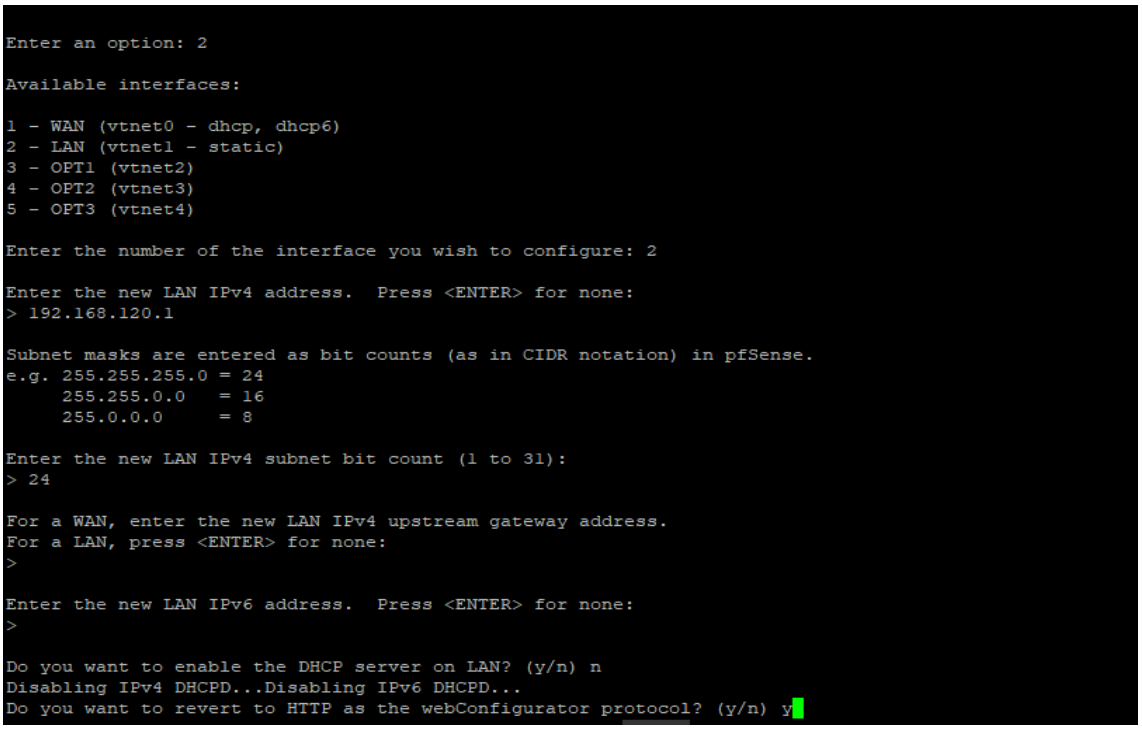


FIGURE 2.16 – Configuration de l'adresse IP LAN.

2.4 Docker

Pour faciliter le développement des applications et résoudre le problème d'exécution causé par des versions dotées des bibliothèques manquantes dans différents environnements, on propose Docker comme solution.

2.4.1 Qu'est-ce que Docker ?

Docker est une plate-forme de virtulisation, conçue pour développer, exécuter et déployer des applications dans un environnement isolé appelé conteneur. Il peut donc avec la propriété d'isolation gérer plusieurs conteneurs simultanément sur un hôte donné et résoudre le problème d'environnement, car quelle que soit la machine que nous utiliserons, le code s'exécutera de la même manière [12].

2.4.2 L'architecture de Docker

L'architecture de base de Docker se compose de 3 parties principales : Host Docker, Registre, Client Docker, (Voir figure 2.17).

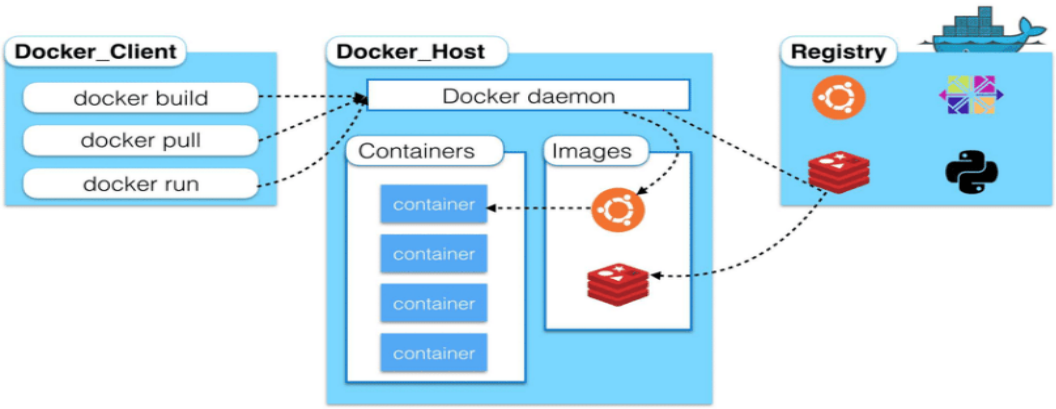


FIGURE 2.17 – L'architecture Docker [12].

- **Client docker :**

  Se présente sous la forme de l'exécutable docker. Il accepte des commandes de l'utilisateur (docker run, docker build, docker push.) et les transmet au daemon Docker.

- **Registre :**

  Docker registre est un service permet de stocker et de télécharger des images. Les registres publics incluent le Docker Hub et Docker Cloud, Les commandes courantes lors l'utilisation ces registres est : docker push, docker pull, docker run.

- **Host Docker :**

  L'hôte Docker fournit un environnement complet pour exécuter des applications. Il comprend le démon Docker, les images et les conteneurs.

  - **Démon Docker :**

    écoute les demandes client Docker et gère les images et les conteneurs.

  - **Image :**

    Est un paquet exécutable qui inclut tout ce qui est nécessaire pour exécuter une application : le code, les outils, les bibliothèques et les fichiers de configuration.

  - **Conteneur :** Est une instance d'exécution d'une image. Il ressemble à une machine virtuelle basé ce le noyau Linux de notre machine physique qui charge une application avec tous les éléments nécessaire pour sa fonctionnement : outils, bibliothèques, fiches ... etc. L'exécution de l'application et ces éléments se fait directement dans le système d'exploitation de serveur comme des processus.

Docker permet d'exécuter l'application dans un environnement isolé (conteneur), contrairement à La virtualisation qui ne permet de créer des machines virtuelle, chacune d'elle se comporte comme un machine intégrant son propre SE sur lequel les applications seront déployées et exécutées, (Voir figure 2.18).

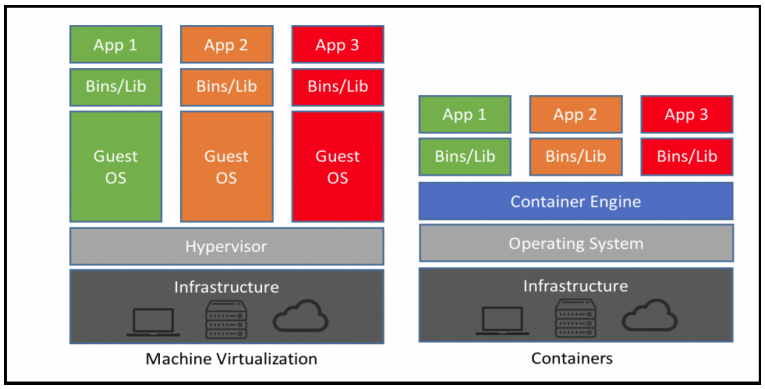


FIGURE 2.18 – Conteneur Vs machines virtuelle.

2.4.3 l'installation de Docker edition community(CE)

pour l'installation docker il faut disposer une connexion internet et avant d'installer docker CE pour la première fois il faut configurer le repository comme suite :

a) Mettre en place le répository docker :

1. Mettre à jour l'index du paquet d'APT [3] :

   `sudo apt-get update`

2. ajouter la clé GPG du site de Docker :

   `curl -fsSL https ://download.docker.com/linux/ubuntu/gpg | sudo apt-key add -`

3. Vérifiez que vous avez bien installé cette clé en comparant avec les 8 derniers caractères de l'empreinte ci-dessous : 9DC8 5822 9FC7 DD38 854A E2D8 8D81 803C 0EBF CD88 Pour cela, taper :

3. Advanced Packaging Tool est un système complet et avancé de gestion de paquets en ligne de commande qui automatise le processus d'installation, désinstallation, mise à jour de logiciels installés sur les systèmes d'exploitation basés sur Unix, tels GNU/Linux.

```
sudo apt-key fingerprint 0EBFCD88
pub rsa4096 2017-02-22 [SCEA]
9DC8 5822 9FC7 DD38 854A E2D8 8D81 803C 0EBF CD88
uid [ unknown] Docker Release (CE deb) <docker@docker.com>
sub rsa4096 2017-02-22 [S]
```

b) Installer docker(CE) :

1. Mettre à jour APT : `sudo apt-get update`
2. Installer la dernière version de Docker CE : `sudo apt-get install docker-ce`
3. Une fois Docker installé, vous pouvez lancer le service : `sudo service docker start`
4. Une fois Docker installé, vous pouvez lancer le service : `docker run hello-world`

2.5 Nextcloud

2.5.1 Définition Nextcloud

Nextcloud est un logiciel de cloud, libre et open source dérivée par Owncloud [4]. C'est un site d'hébergement des fichiers, c'est à dire, permet de placer les données sur notre serveur dédié ou quelque part sur l'Internet de telle sorte quelles soient accessibles depuis à Internet.

Nextcloud est un logiciel collaboratif (facilite le travail en groupe) par le partage des documents, d'agendas ou de contacts en gérant les groupes d'accès.[6]

2.5.2 Fonctionnalité de Nextcloud

Nextcloud offre divers fonctionnalité suivante :

- Synchronisation de fichiers entre différents ordinateurs.
- Partage des fichiers sur internet.
- Partager un fichier avec un utilisateur ou un groupe d'utilisateurs.
- Télécharger un fichier depuis le serveur vers son ordinateur.
- Ajouter un fichier dans votre cloud.

4. Owncloud : Est un logiciel libre, lancé en janvier 2010 offrant une plateforme de services de stockage et partage des fichiers en ligne, peut installé sur n'importe quel serveur et supportant SQLite (base de données par défaut), MariaDB, MySQL ou PostgreSQL.

2.5.3 Installe nextCloud dans Docker

Il existe deux versions (l'image apache et l'image fpm). Dans notre application nous avant choisie l'image apache qui dispose une installation complet de nextcloud contenant un serveur web apache et une base de données intégrer.

Âpre l'installation de docker, on tape les commandes suivant :

- docker run -d -p 8080 :80 nextcloud : Permet de télécharger et installer l'image Nextcloud dans docker.
- docker ps -a : Cette commande permet d'indiquer si le processus d'installation est effectuer correctement en fournissant un ensemble d'information et nous fournit un ensemble d'informations comme l'identificateur de conteneur, le nome de l'image et le protocole utiliser.
- docker start Id-conteneur : Pour lancer l'instance Nextcloud, en fournissant l'id du conteneur, après avoir accéder à Nextcloud via l'url : http ://serveur-web :8080/

Conclusion

Dans ce chapitre, après que nous avant décri les outils nécessaire pour la réalisation de l'architecture, nous avons explicitement fourni les différentes étapes d'installation et de configuration des outils et d'application. Le chapitre suivant va consacrer à la phase de la réalisation technique tels que le paramétrage de réseau et les configurations spéciales de PFsense et la configuration OpenVPN.

Chapitre 3

Réalisation technique

Introduction

Dans ce chapitre, nous allons expliquer tout d'abord le mécanisme et la technique de virtulisation du système et introduire c'est quoi la DMZ. Ensuite, nous présenterons l'architecture réseau utilisée. Après la mise en œuvre de PFsense, nous allons configurer OpenVPN et décrire les déférents scénarios possibles d'exécution de l'application Nextcloud.

3.1 Virtualisation

La virtualisation est l'ensemble des technologies matérielles et/ou logiciels qui permettent de faire fonctionner sur une seule machine plusieurs systèmes et/ou plusieurs applications, séparément les uns des autres, comme s'ils fonctionnaient sur des machines physiques distinctes.[15]

3.1.1 Mécanisme de virtualisation

Le principe de la virtualisation se base sur 3 pivots majeurs qui sont le système hôte, l'hyperviseur et le système invité. La combinaison de l'ensemble de ces éléments permet de créer une virtuelisation qui se base sur le mécanisme suivant :

- Un système d'exploitation principal «système hôte» est installé sur un serveur physique.
- Un logiciel de virtualisation «hyperviseur» est installé sur le système hôte, son rôle est de permettre de créer plusieurs environnement fermés et indépendant qui pourront à leur tour d'héberger plusieurs systèmes invités. Ces environnements sont les «machines virtuelles».
- Un système invité est un système d'exploitation installé dans une machine virtuelle qui fonctionne indépendamment des autres systèmes invités dans d'autres machines virtuelles d'un même serveur physique. Chaque machine virtuelle dispose d'un accès aux ressources du serveur physique (mémoire, espace disque...).[16]

3.1.2 Virtualisation Système

La virtualisation système fait référencée à l'utilisation de logiciel pour permettre au matériel d'exécuter plusieurs instances de différents systèmes d'exploitation en même temps. Ce qui permet d'exécuter différentes applications sur différents systèmes d'exploitation dans le même serveur physique.

Les technologies et les logiciels qui offrent la virtualisation systéme qui permet la création et la gestion de cette tâche sont VM et L'hyperviseur.

a) **Machine virtuelle(VM) :**

Une machine virtuelle est un environnement fourni par un logiciel qui permet de simuler une machine physique au sein d'un système hôte ou directement sur le matériel physique. L'interaction avec une machine virtuelle est la même qu'avec un matériel dédié. Une machine virtuelle (VM) se comporte exactement comme un ordinateur physique. Elle contient son propre matériel virtuel : CPU, mémoire RAM, disque dur et carte d'interface réseau (NIC) basés sur du logiciel.

b) **Hyperviseur (VMM)**

Un hyperviseur est une plateforme de virtualisation permettant à différents systèmes d'exploitation de s'exécuter parallèlement sur une même machine physique tout en partageant les ressources de cette dernier. Le rôle de l'hyperviseur est de permettre aux différents environnements de s'exécuter sur l'ordinateur hôte tout en isolant les environnements les uns des autres de la même façon comme ils s'étaient exécutés sur des machines physiques distinctes.

L'hyperviseur ayant la responsabilité de fournir les ressources matérielles nécessaires aux différents environnements virtuels.

Il existe deux types d'hyperviseurs.

- Hyperviseur type 1 : Un logiciel qui s'exécute directement sur le matériel de l'ordinateur hôte sans dépendance avec le système d'exploitation de celui-ci pour le partage des ressources matérielles. Les systèmes d'exploitation invités restent donc près des ressources physiques de l'hôte, ce qui n'est pas influ les performance accentue les performances (Voir Figure 3.2).Comme exemple d'hyperviseur de type 1 nous avant : Hyper-V, Xen, ESXI...
- Hyperviseur type 2 : Est un logiciel qui s'exécute directement sur le système d'exploitation de l'ordinateur hôte. L'exécution des machines virtuels dépend alors directement du système d'exploitation de l'hôte. Il y a donc deux couches logicielles qui s'insèrent entre les systèmes d'invités et les ressources physiques de l'hôte. Ce

qui peut affecter les performances (Voir Figure 3.2). Comme exemple d'hyperviseur de type deux nous avant : VMware Workstation, QEMU, VirtualBox...

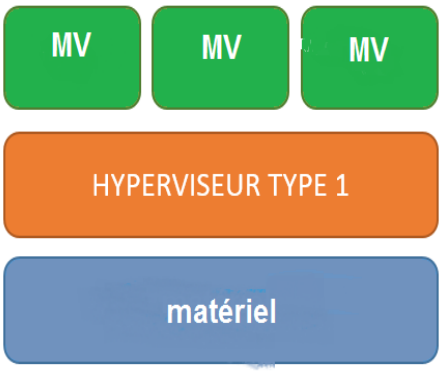


FIGURE 3.1 – Hyperviseur de Type 1.

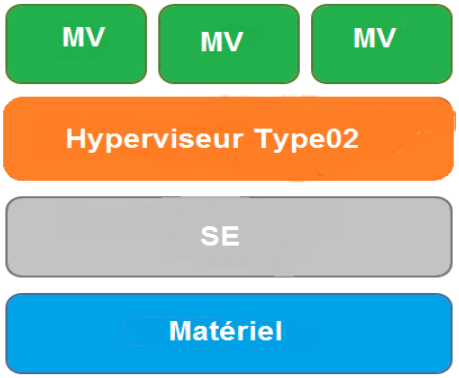


FIGURE 3.2 – Hyperviseur de Type 2.

Il existe plusieurs façons d'organiser le réseau mais, pour que cette organisation soit bien sécurise en doit adoptée une architecture bien structurer comme DMZ.

3.2 L'architecte DMZ(DéMilitarized Zone)

DMZ (Démilitarized Zone) est une zone contient des serveurs dotés d'un certain nombre de services accessibles depuis Internet et du réseau local (LAN).ces services sont réaliser en se basent sur un pare-feu qui contient trois cartes réseaux, la première pour le réseau local, la deuxième pour Internet et la dernière pour la DMZ (Voir figure 3.3).

La DMZ est accessible à la fois par un réseau interne et autre externe, sachant que le réseau interne peut accéder accès au réseau externe.

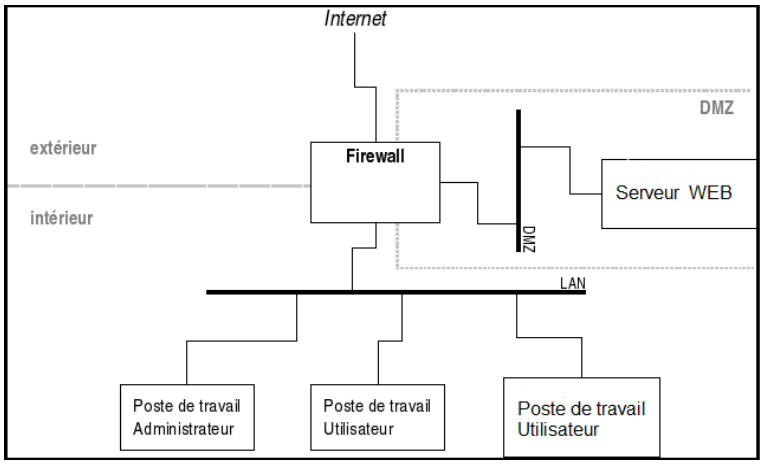


FIGURE 3.3 – Un schéma de DMZ (Démilitarized Zone).

3.3 Présentation de l'architecture

Dans ces dernières années, l'accès distants des étudiants, enseignants et employeurs aux différents documents de l'université devenu une nécessite accrue. Ces accès doivent d'être sûres et fiables.

Le but donc est d'offrir une solution VPN qui permet de fournir un accès sécurisé au réseau local de l'université. Les utilisateurs peuvent utiliser un ordinateur ou un smartphone pour se connecter au réseau.

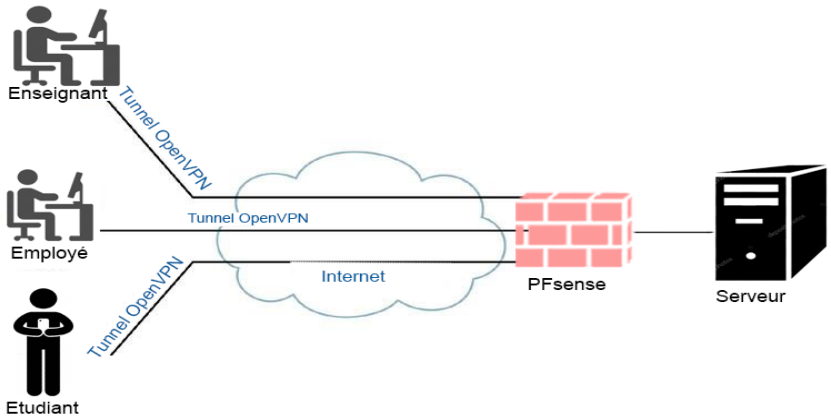


FIGURE 3.4 – Un schéma général de VPN.

D'abord on installe VMware Workstation 14 dans notre système d'exploitation « système hôte » comme un hyperviseur de type 02. Puis nous créons la machine virtuelle EVE dans lequel on installe le système invité EVE-ng à travers laquelle on établir la simulation de notre réseau.

Avant de commencer la simulation, on doit faire une conception détaillée de l'infrastructure de notre réseau qui présente les cartes réseaux attachées entre eux (Voir figure 3.5).

La machine virtuelle EVE-ng comporte trois cartes réseau qui sont :

- la carte réseau01 (Network adapter 1) et la carte réseau03 (Network adapter 3)sont en mode Host Only, donc Il y a un réseau fermé entre la machine virtuelle et la machine hôte via les switch virtual Vmnet1 et Vmnet2.
- la carte réseau02 (Network adapter 2) qui en mode bridge. La machine virtuel EVE est directement connecté sur la carte réseau physique ou la carte réseau wifi de l'hôte par le switch vertuelle vmnet0.

Dans la machine virtuelle EVE-ng on installe PFsense qui contient 04 interfaces (MGT, DMZ, LAN, WAN).Ces interfaces sont des passrelle comme suite :

- MGT : lié avec cloud 2 qui est l'interface pnet2.

Cette dernière est reliée par défaut en mode bridge avec la cartes réseau03 (Network adapter 3), donc on peut accéder au page web de PFsense à travers la machine hôte.
Alors :

cloud 2 –> Network adapter 3 –> Vmnet2 –> VMware Network Adapter Vmnet2

- WAN : lié avec cloud01 est donc l'interface pnet1. Cette dernière est reliée par défaut en mode bridge avec la cartes réseau02 (Network adapter2), donc on peut connecter aux réseau WAN à travers cette interface.
  Alors :
  WAN –> cloud01 –> Network adapter2 –> Vmnet0 –> Carte réseau wifi

- Pour établir la page web de EVE-ng :
  Network adapter1 –> Vmnet1 –> VMware Network Adapter Vmnet1

- DMZ reliée avec une zone contient un serveur web.
- LAN reliée avec une zone local des utilisateurs.

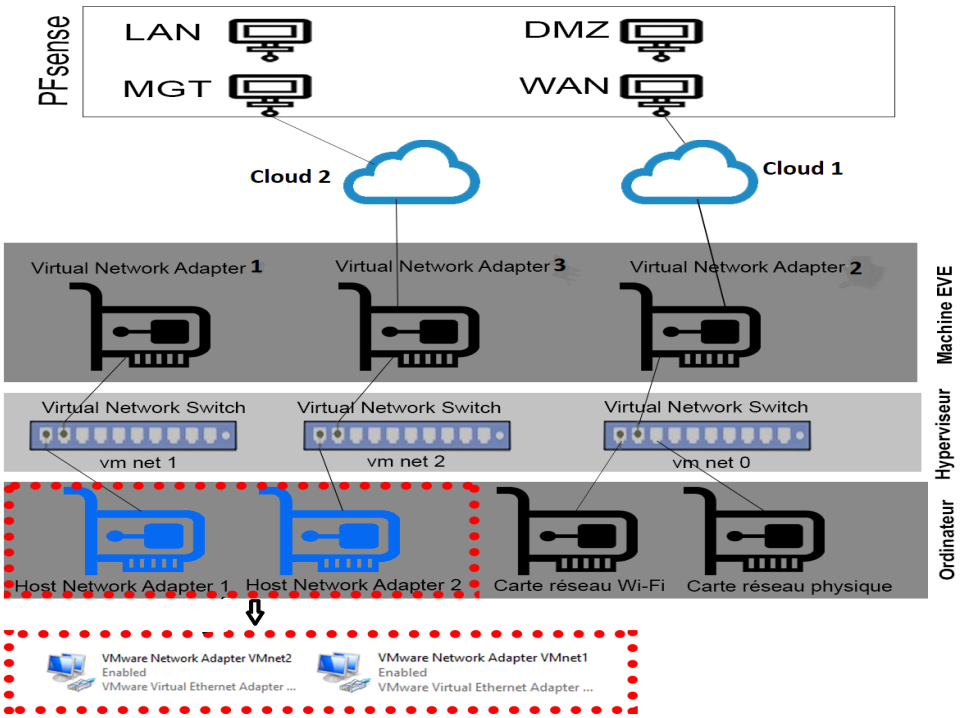


FIGURE 3.5 – Schéma détaillé de l'architecture adoptée.

Dans ce qui suite, nous allons former le plan d'adressage pour notre schéma qui détermine l'adresse IP et donc des équipements laquelle composent le réseau de l'université.

Interfaces			Addresses IP	Masque
e0	WAN	WAN	Fournir par DHCP	255.255.255.0
e1	LAN	LAN	192.168120.1	255.255.255.0
e2	OPT1	DMZ	192.168.110.1	255.255.255.0
e3	OPT2	MGT	192.168.140.10	255.255.255.0

TABLE 3.1 – Plan d'adressage.

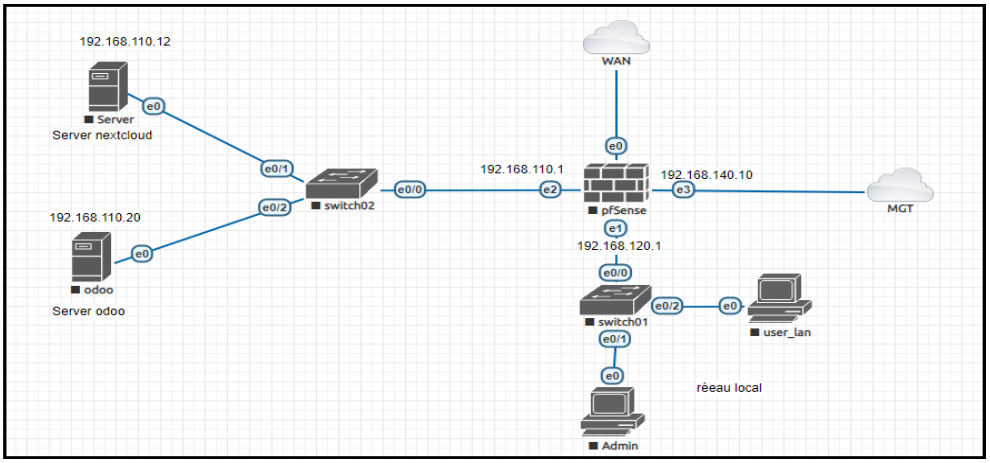


FIGURE 3.6 – Schéma global.

3.4 Configuration de PFsense

3.4.1 Configuration générale

Pour configurer l'interface via le web il faut d'abord connecter à l'interface LAN de PFSense. Commencer par ouvrir un navigateur web et entrer l'adresse IP LAN de la machine (PFsense) dans la barre d'adresse : http ://IP-LAN. Dans notre cas, nous ferons http ://192.168.120.1 pour accéder à l'interface de connexion (Voir figure 3.7) où il est demandé d'entrer un nom d'utilisateur et un mot de passe.

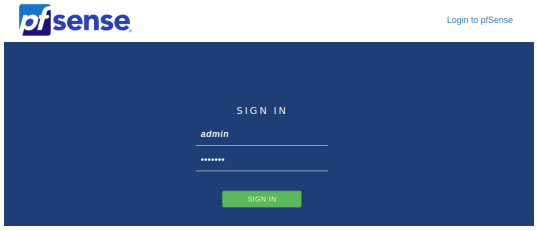


FIGURE 3.7 – Portail de connexion à PFsense.

- Une fois le nom d'utilisateur qui est par défaut "admin" et le mot de passe qui est par défaut"pfsense" sont saisis sur la page de configuration générale, les paramètres permettant d'attribuer un nom d'hôte au pare-feu s'affichent, le domaine et saisir l'adresse du serveur DNS primaire et secondaire(dans notre cas nous n'avons pas créé un serveur DNS, nous avons utilisé celui du fournisseur d'accès Internet),(Voir figure 3.8). Dans les étapes suivantes, nous allons vérifier à nouveau la configuration de l'interface WAN et LAN et attribuer un nouveau mot de passe administrateur.

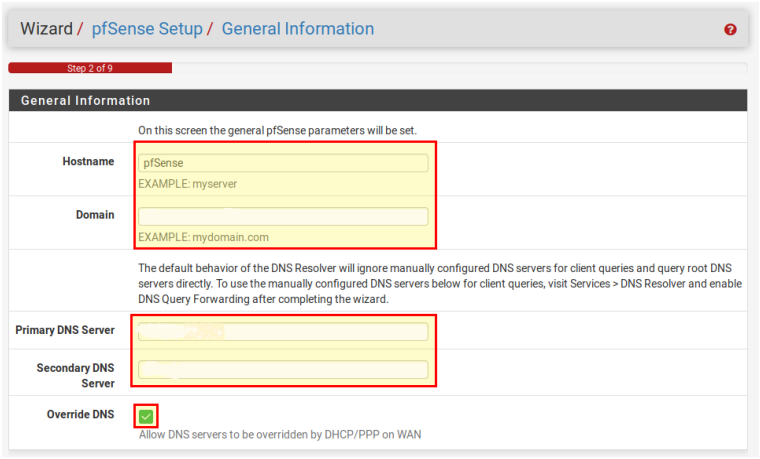


FIGURE 3.8 – Paramètres généraux de PFsense.

- Après avoir terminé la configuration, sur la page d'accueil de PFsense, les informations globales sur le logiciel (version, dernier changement de configuration, version du noyau, utilisation de la mémoire, le nombre d'interfaces connectées etc..)s'affichent, (Voir figure 3.9).

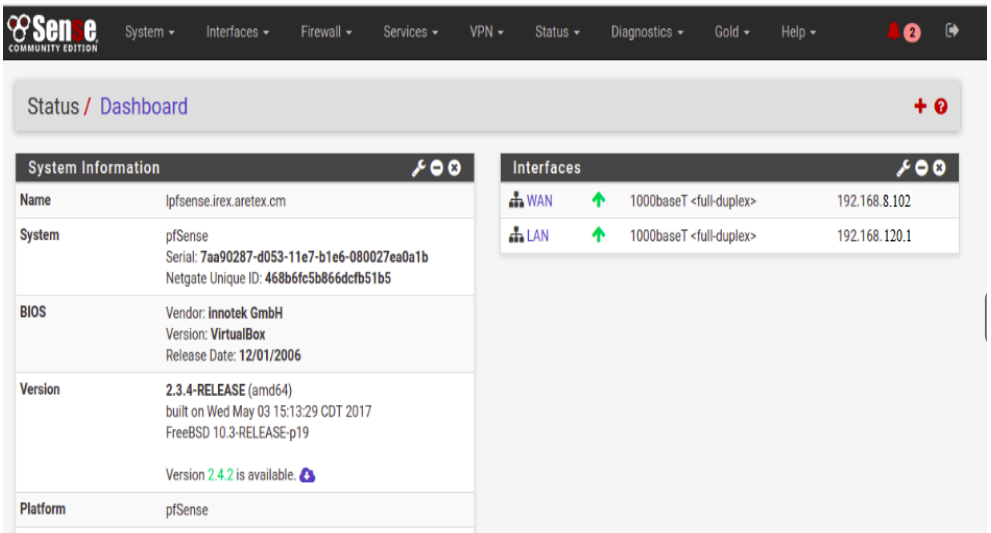


FIGURE 3.9 – Écran d'accueil de PFsense.

3.4.2 Configuration des interfaces

- WAN : configurée à par du guide d'installation de PFsense.
- LAN : Celle-ci est également configurée dans le guide d'installation de PFsense.
- OPT : L'interface (OPT) correspond à notre liaison de secours. Elle porte le nom «Optional» car peuvent être ajoutés à l'aide d'interfaces OPT des réseaux supplémentaires. dans notre cas, nous allons ajouter deux réseaux, DMZ et MGT qui seront reliées à l'interface OPT1 et OPT2 de PFsense. Dans l'onglet Interfaces, sélectionner OPT1 puis l'activer en cochant Enable interface, sélectionner ensuite le type d'adressage statique ou DHCP. Ici nous avons assigné une adresse statique. Ensuite en précisant l'adresse IP (ici 192.168.0.x/24). Laisser les autres paramètres par défaut. Cette étape est représentée par la figure 3.10. Nous faisons Les mêmes étapes sur l'interface MGT.

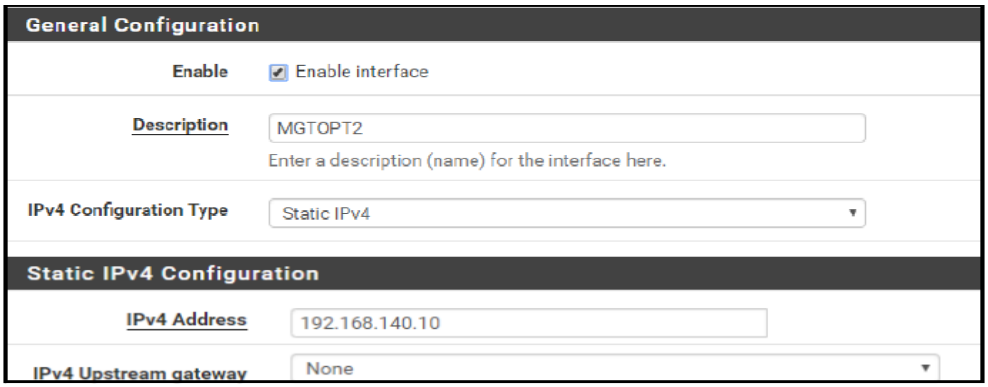


FIGURE 3.10 – Configuration de l'interface DMZ.

3.4.3 Règles du pare-feu

Par défaut toutes les connexions sont bloquées. On applique un ensemble de règles sur les interfaces de PFsense pour permettre ainsi à ces interfaces de communiquer entre elles et avec l'extérieur.

PFsense contient un ensemble des règles prédéfinies permettant de :

- autoriser la connexion (allow).
- bloquer la connexion (deny).
- rejeter la demande de connexion sans avertir l'émetteur (drop).

L'ensemble de ces règles permet de mettre en œuvre un filtrage précis sur les paquets des données transmis par différents réseaux.

Lorsque le paquet de donnée arrive au PFsense, soit par les réseaux : WAN, DMZ, LAN ou vpn. Il analyse les packages et vérifie leur conformité aux règles.

Dans l'onglet **Firewall**, on clique sur la section **Rules** puis on sélectionne une interface sur laquelle on veut définir des règles. La politique de sécurité mise en œuvre sur la topologie est la suivante :

a) WAN :
   - Le trafic du réseau externe (WAN) vers la DMZ : autorisé.
   - Trafic du réseau externe (WAN) vers le réseau interne(LAN) : interdit.

b) LAN :
   - Trafic du réseau interne (LAN) vers la DMZ : autorisé.
   - Trafic du réseau interne (LAN) vers le réseau externe(WAN) : autorisé.

c) DMZ :
   - Trafic de la DMZ vers le réseau interne (LAN) : interdit.
   - Trafic de la DMZ vers le réseau externe (WAN) : refusé.

d) MGT :
   - Le trafic du réseau externe (laptop) vers pare-feu (PFsense) : autorisé.

3.5 OpenVPN

3.5.1 présentation d'OpenVPN

Outil de création de réseau VPN facile à utiliser, robustesse et basé sur SSL/TLS. Il peut être utilisé pour relier d'une manière sécurisée deux réseaux privés ou plus. Il est le seul produit VPN open source qui supporte entièrement l'ICP (infrastructure à Clef Publique), OpenSSL [1] qui implémente des protocoles SSL/TSL pour la session d'authentification et l'échange de clef, l'algorithme HMAC pour authentifier les données encapsulées, et tout ceci au travers d'un unique port UDP. Celui-ci contient un exécutable pour les connexions du client et du serveur et une ou plusieurs méthodes d'authentification choisie.

3.5.2 Critères d'OpenVPN

Il existe de nombreuses solutions VPN qui sont disponibles, néanmoins il y a une solution qui a été favorisée, on parle de la solution OpenVPN. En effet, Le choix d'OpenVPN se fait pour de nombreux raisons, parmi lesquels citant :

– OpenVPN est une solution open source et gratuite.

1. OpenSSL est une boîte à outils de chiffrement comportant deux bibliothèques (libcrypto fournit les algorithmes cryptographiques et libssl implémente le protocole de communication (TLS) ainsi que son prédécesseur Secure Sockets Layer (SSL))

– OpenVPN est facile à configurer et à mettre en place en plus d'offrir un niveau de sécurité équivalent à d'autres solutions plus complexes.
– OpenVPN permet d'authentifier les utilisateurs et pas seulement les machines.

Il existe deux produits OpenVPN :

- OpenVPN Access Server.
- OpenVPN Community software.

Les deux ont les mêmes fonctionnalités et le même niveau de sécurité si ce n'est qu'OpenVPN Access Server se présente sous forme d'interfaces graphiques plus conviviales qu'OpenVPN Community Software qui s'utilise en ligne de commandes. La configuration est donc plus facile avec OpenVPN Access Server, surtout pour les clients, en effet les clients n'auront besoin que d'un mot de passe et l'adresse du serveur pour télécharger l'application cliente qui leur permettra de se connecter au VPN. Alors que dans OpenVPN Community software il faut transporter physiquement les fichiers client (certificat, clé) sur la machine cliente pour lui permettra de se connecter, cependant OpenVPN Access Server ne permis d'avoir que deux utilisateurs connectés simultanément, pour avoir plus d'utilisateurs connectés en simultané il faut payer une licence, c'est le principal inconvénient qui nous a empêcher d'opter pour cette solution qui est de ce fait la solution idéale pour une entreprise qui a les moyens.

Dans notre cas, nous avons choisie PFsense qui favorise OpenVPN Community Software, car ce dernier peut être configurer à partir de l'interface web du PFsense.

3.5.3 Fonctionnement d'OpenVPN

OpenVPN fonctionne en mode client-serveur. Ceci signifie qu'il y a un serveur VPN qui est connecté à internet. Les clients OpenVPN se connectent au serveur VPN. une fois le tunnel établie, le client peut accèdes au réseau interne et DMZ de l'entreprise ou de l'université.

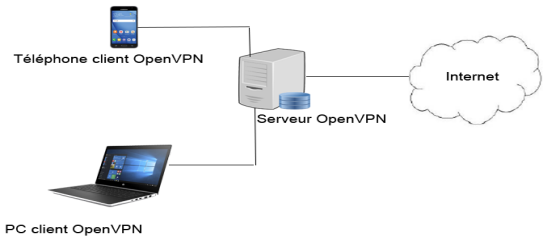


FIGURE 3.11 – Fonctionnement d'OpenVPN.

Avant de connecter par OpenVPN, de nombreuses options doivent être considérées, tels que :

a) **Interfaces virtuelles (TUN/TAP)**

OpenVPN fonctionne avec des interfaces virtuelles TUN/TAP, ces interfaces sont vues comme des interfaces physiques, mais elles renvoient le trafic vers le logiciel OpenVPN plutôt que sur un lien physique.

OpenVPN peut fonctionner avec deux types d'interfaces virtuelles selon le type de tunnel. i) Si on souhaite créer un tunnel de niveau 2 (on se basant sur le modèle OSI), c'est-à-dire un tunnel Ethernet on utilisera une interface virtuelle de type TAP. ii) Si on veut créer un tunnel de niveau 3, c'est-à-dire un tunnel IP, on utilisera une interface de type TUN. .

- En mode routeur (TUN) : nous créons un tunnel de couche 3, nos connexions prendront l'aspect d'un routeur. C'est à dire que OpenVPN devra maintenir une table de routage pour rediriger les clients dans les bonnes directions.
- En mode bridge (TAP) : le client OpenVPN se connecte directement sur le réseau local. Symboliquement, c'est exactement comme brancher le client directement sur le réseau. La configuration dans ce cas nécessite de créer une interface de bridge sur le serveur OpenVPN, et de rattacher l'interface réseau physique du serveur et l'interface créée par OpenVPN à ce bridge.

b) **Choix du protocole**

OpenVPN peut utiliser le protocole UDP ou TCP, ceci pour augmenter les possibilités de connexion aux utilisateurs.

- **OpenVPN avec TCP**

  Signifie « Transmission Control Protocol ». Contrairement à UDP, TCP est orienté connexion, c'est à dire, il assure un circuit virtuel entre les clients et les serveurs Openvpn. Le protocole TCP établit un mécanisme d'acquittement et de retransmission de paquets manquants. c'est à dire, lorsqu'un paquet se perd et ne parvient pas au destinataire, TCP permet de prévenir l'expéditeur et lui réclame de renvoyer les informations non parvenues. Il permet donc de garantir une certaine fiabilité des transmissions. Cependant la majorité des applications utilisent TCP, OpenVPN ne se bases pas sur le port TCP car la création du tunnel TCP sur un autre tunnel TCP (TCP over TCP) permet de retransmettre de paquets à l'intérieur du tunnel et provoque des interruptions fréquentes de la connexion. Ce qui peut conduire à la dégradation des performances d'OpenVPN.

- **OpenVPN avec UDP** UDP est le protocole le plus adéquat avec UDP, qui est généralement plus rapide. UDP signifie « User Datagram Protocol ». C'est le protocole de transport sans confirmation qui permet aux applications d'échanger des datagrammes sans accusé de réception ni remise garantie. Cela réduit la charge et rend UDP plus rapide parce qu'il ne fournit pas de mécanisme de vérification d'erreur comme le fait TCP.

c) **Méthodes d'authentification** Lors d'une négociation HANDSHAKE, il faut s'assurer de l'identité du client, il faut aussi assurer que le serveur auquel on parle est bien celui qu'il prétend être. OpenVPN propose plusieurs méthodes d'authentifications, comme par exemple :

- Remote Access (User Auth) : qu'il s'agit d'une méthode simple utilise un paire nom d'utilisation/mot de passe.
- Remote Access (pre-shared key) : tous les clients doivent avoir une clef pour pouvoir s'authentifier.
- Remote Access (SSL/TLS) : Une méthode d'authentification par certificat où chaque client possède un certificat (et donc une clef privée) validé par une autorité de certification.
- Remote Access (SSL/TLS + User Auth) : Nécessite un certificat et un nom d'utilisateur/mot de passe. Chaque utilisateur dispose d'une configuration client unique qui inclut son certificat et sa clé personnels.

La méthode d'authentification par le couple nom d'utilisation/mot de passe et les certificats est la plus sécurisée et la plus flexible d'administration, elle permet d'assurer l'identité du client et du serveur avant tout échange de données.

3.6 Configuration d'OpenVPN sous PFsense

3.6.1 Création du tunnel OpenVPN

Pour la construction du tunnel, le serveur OpenVPN utilise les protocoles SSL/TLS. Il ya quatre protocoles différents qui interviennent dans les différentes phases d'une communication sous SSL/TLS, ces protocoles peuvent être divisés en 2 couches. La première couche est constituée par des protocoles de négociation (Handshake, Cipher, Alert) et la deuxième couche est le protocole Record. La figure 3.12 illustre ces différentes couches :

- **Protocol Handshake :** Ce protocole permet l'authentification obligatoire du serveur, au client(optionnelle) et négocier sur les algorithmes de chiffrement,les algorithmes de

HMAC et les clés symétriques qui se servir au chiffrement avant que l'application transmette son premier octet.

- **Protocol CCS (ChangeCipherSpec) :** Ce protocole comprend un et un seul message (1 octet) qui porte le même nom que le protocole, il permet d'indiquer au protocole Record les algorithmes de chiffrement négociés.
- **Protocol Alert :** Le message de ce protocole est dans le but de i) signaler une erreur fatale (qui va donc provoquer la fermeture de la session) ii) ou une simple alerte qui laisse le choix à l'autre extrémité de fermer connexion ou non. Dans les deux cas, le message comporte un code spécifiant la raison de l'alerte. La liste des codes utilisés peut indiquer : une notification de fermeture, certificat expiré, accès refuse...etc. Ce message peut également être employé pour signaler la fin normale de la connexion.
- **Protocol Record :** Ce protocole intervient après l'émission du message ChangeCipherSpec. Il permet de garantir la confidentialité et l'intégrité des données.

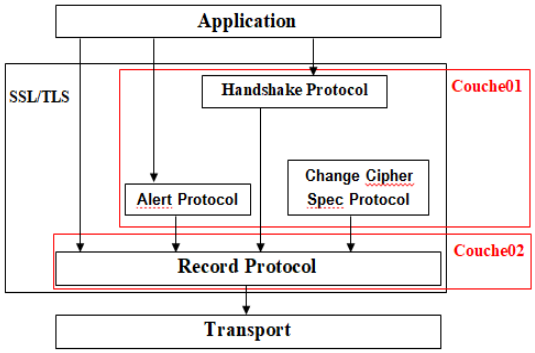


FIGURE 3.12 – Protocols SSL/TLS.

La création du tunnel OpenVPN se fait en deux phases : une phase de négociation des paramètres de sécurité (échangés 1 à 9 dans le schéma ci-après) et une phase de transfert sécurisé de données.

a) **Négociation des paramètres de sécurité**

Nous commençons d'abord par les trois protocoles qui sont mis en œuvre lors de l'établissement d'une session TLS. Ces protocoles permettent au serveur et au client de faire :

- S'authentifier mutuellement.
- Négocier sur :
  - les algorithmes de chiffrement.
  - les algorithmes de HMAC (Message Authentification Code).
  - les clés symétriques qui vont servir au chiffrement avant que l'application ne transmette son premier octet.

Le schéma suivant va nous permettre de résumer les principaux échanges effectuées lors d'un dialogue entre le client et le serveur OpenVPN.

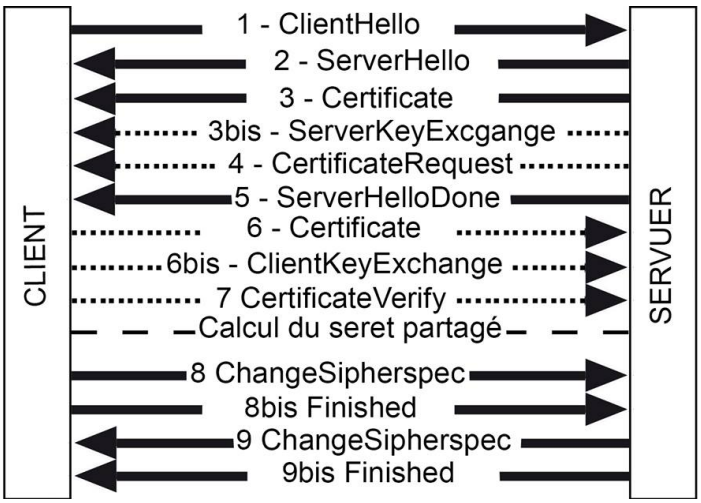


FIGURE 3.13 – Dialogue TLS entre un client et un serveur.[3]

1. Client Hello : Ce message permet à un client de demander l'établissement d'une connexion sécurisée avec un serveur.

   Voici les principaux éléments constitutifs de cet échange :
   - Version de SSL (3.0 pour SSLv3.0, 3.1 pour TLSv1.0, 3.3 pour TLSv1.2).
   - Proposition de chiffrement et de calcule d'intégrité, généralement appelées Cipher-Suite. Ces propositions définissent également l'algorithme de négociation de clés(RSA, Diffie-Hellman).
   - Des données aléatoire pour générer les clés.
   - Identifiant de session qui sera vide pour une session totalement nouvelle ou rempli avec l'identifiant d'une session déjà existante si le client souhaite répondre à celle-ci.

2. Sevrer Hello : Ce message permet au serveur de répondre au Client Hello en précisant les éléments suivants :
   - Version de serveur.
   - Une seule ciphersuite choisie parmi celles proposées par le client.
   - Des données aléatoire pour la génération les clés.
   - Identifiant de session qui peut soit contenir le Même identifiant que celui demandé par le client dans le cas d'une reprise de session ou une valeur nouvelle dans le cas contraire. Si la valeur est vide, cela signifie que le serveur a refusé la reprise de session demandée par le client et il faut donc que celui-ci redemande une nouvelle session.
3. Certificat : Ce type de message permet au serveur d'envoyer un certificat au client. Ce certificat contient les données nécessaires à l'établissement des clés
4. Server Key Exchange : Pour ce qui concerne les paramètres d'échange de clé, (ServerKeyExchange), leur envoi n'a lieu que si les certificats du serveur ne contiennent pas toutes les informations nécessaires pour créer un premastersecret. Dans le cas du Diffie-Hellman, ce message contiendra notamment la clé publique du serveur.
5. Certificat Request :

   Si le serveur souhaite également authentifie le client par un certificat, il va le demander au moyen du message CertificateRequest. Celui-ci contient la liste des types de certificats (RSA, DSS, RSA-DH, DSS-DH) que le serveur accepte et, optionnellement, la liste des autorités de certifications qu'il reconnait. Dans le cas de TLS, la liste des algorithmes de signature reconnus par le serveur est également spécifiée.
6. Server Hello Done :

   Ce message sert à indiquer au client que le serveur a fini d'envoyer les message de cette phase.
7. Client certificat :

   Dans le cas d'une requête CertificateRequest été émise par le serveur, ce message permet au client d'envoyer son certificat.
8. Client Key Exchange :

   Ce message suit le message Clientcertificat si celui-ci existe, sinon c'est le premier message envoyé par le client après avoir reçu ServerHelloDone. Le contenue de ce message se varié selon les algorithmes de clé publique utilisés par le serveur dans le message ServerHello.

9. Certificate Verify :

   Ce message n'est émis que si un certificat client est envoyé du client au serveur par un message ClientCertificate.

10. Client Change Cipher Spec :

    Le client déclare par ce message au serveur que tous les échanges seront dorénavant authentifiés selon les paramétré négociés ou déterminés lors des échanges précédents.

11. Client Finished :

    Dans ce cas le client envoie un message crypté Finished contenant un hachage et un code HMAC couvrant tous les messages Hndshake déjà échangés (donc les messages ChangCipherSpec en sont exclus). Ce message sera décrypté par le serveur. Si cette tentative de décryptage échoue, le serveur rejettera le handshake et termine la connexion.

12. Server Change Cipher Spec :

    Après avoir décodé avec succès le message Finished du client, le serveur déclare par ce message au client que tous les échanges seront dorénavant cryptés authentifiés.

13. Server Finished :

    Puis le serveur envoie un message crypté Finished contentent un hachage et un code HMAC couvrant tous les messages Handshake déjà échangés. Le client procédé également au décodage et à l'authentification de message Finished du serveur. Si tout se passe bien, la phase Handshake est alors terminée. Les échanges de données vont pouvoir alors commencer (Application Phase).

b) **Transfert sécurisé de données**

Le protocole record assure la confidentialité et l'intégrité des message.Le schéma suivant nous montre en détail comment se déroule ce protocole :

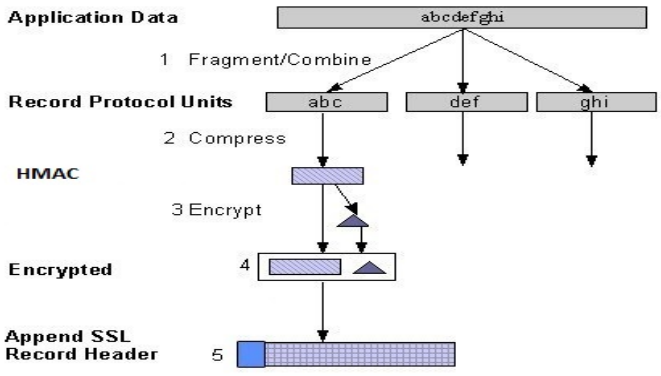


FIGURE 3.14 – Chiffrement avec le record protocole

1. Segmentation : les données sont découpées en blocs.
2. Champ HMAC : les données sont compressées en utilisant la fonction HMAC(Message Authentication Code).
3. Chiffrement : le paquet obtenu est chiffré par le clé récupérée lors de la négociation.
4. Un en-tête de 5 octets est ajouté. Le champ "Type= Application Data" de cet en-tête définit le type du protocole de niveau supérieur au Record Protocol.

À la réception des paquets, le destinataire effectue les opérations suivantes :

1. Vérification de l'en-tête SSL/TLS.
2. Déchiffrage du paquet.
3. Vérification du champ HMAC (en appliquant la même fonction que celle décrite ci-dessus, aux données déchiffrées puis en comparant le résultat au HMAC reçu).
4. déchiffrée des données.
5. Ré-assemblage des blocs de données.

3.6.2 Exemple de configuration d'OpenVPN sous PFSense

a) **Installation du package OpenVPN Client Export Utility**

- Afin de faciliter l'intégration sur les postes clients, nous allons installer Le paquetage «OpenVPN Client Export Utility». Cet outil permet l'export directement à partir de PFsense d'un client préconfiguré OpenVPN pour Windows ou d'un fichier de configuration pour android (Voire figure 3.15).

FIGURE 3.15 – Installer le package d'export client OpenVPN.

1. Se rendre dans «System» –> «Package Manager».
2. Choisir «Available Packages» pour installer des packages.
3. Choisir «openvpn-client-export».

b) **Création de l'infrastructure des certificat requise**

- La première étape à faire sur PFSense est de créer une autorité de certification Comme suite (Voir figure 3.16)) :

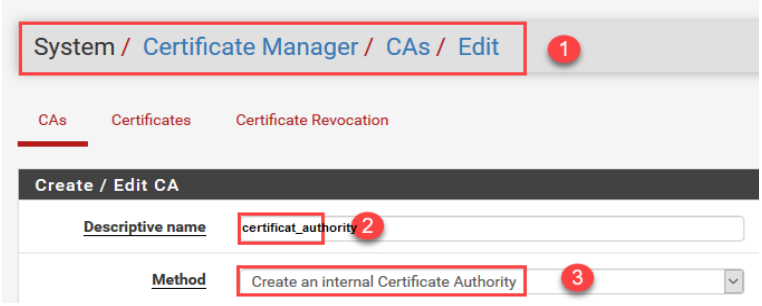


FIGURE 3.16 – Créer un certificat autorité.

1. Se rendre dans la partie «System» -> «Certificate Manager» -> «CAs».
2. Choisir un nom explicite pour la CA.
3. Choisir «Create an Internal Ceritificate Authority» pour créer une CA.

- Remplir le formulaire que PFSense utilise pour créer le certificate.

FIGURE 3.17 – Créer un certificat autorité.

1. La longueur de la clé doit être au minimum de 2048.
2. L'algorithme de hashage doit appartenir au minimum SHA2, nous optons pour SHA256.
3. La validité du certificat (CA) : Un CA possède généralement une durée de validité longue. Notre choix est donc de la garder 10 ans.

4. Si vous avez une résolution DNS, choisissez le nom du serveur comme « Common Name » même si cela n'influ pas sur la sécurité des certificats : nous laissons « internal-ca » qui est la valeur par défaut.
5. Remplir les différents champs qui concerne le sujet d'autorité du certificat. Par défaut tous les certificats émis par cette CA auront ces valeurs : AL pour le pays Algeria.
6. Beinen pour la région.
7. Mila pour la ville.
8. «Company Name» pour le nom de la société.

- Le CA est maintenant créée. Vous pouvez la visualiser dans la page d'accueil des CA (Voir figure 3.18).

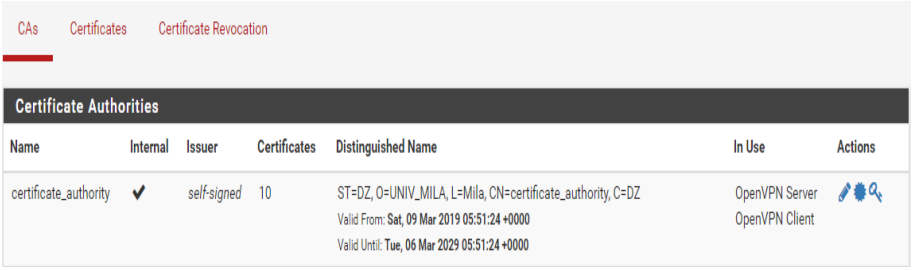


FIGURE 3.18 – Certificat autorité(CA).

- L'étape suivante consiste à créer le certificat du serveur OpenVPN, que les clients utiliseront pour vérifier l'identité du serveur lors de la connexion. Nous utilise ons le certificat généré précédemment qui est la CA pour le certificat de serveur.
- Nous devons ensuite remplir le formulaire que nous utilisons pour créer le certificat de serveur avec l'identification du type de certificat "Server Certificate ou Client Certificate" . par l'accès à l'onglet Certificats en cliquant sur "+Ajouter/Signer" (Voir figure 3.19).

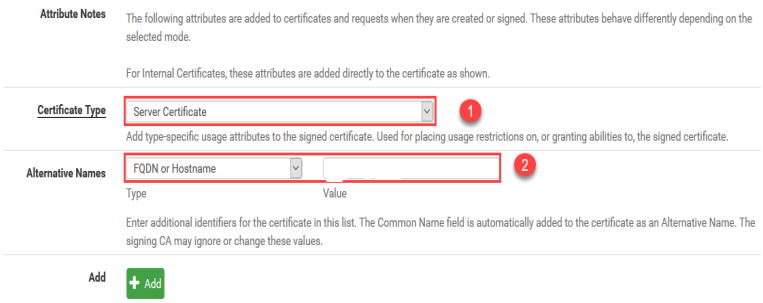


FIGURE 3.19 – Certificat de serveur.

c) Configuration d'OpenVPN côté "serveur"

- Pour démarrer la configuration, ouvrez le menu VPN de l'interface Web et sélectionnez OpenVPN, puis cliquez sur l'onglet Wizard.
- Dans la première étape de la configuration, nous devrons choisir le type de serveur d'authentification «Local User Access» pour gérer les utilisateurs VPN à l'aide du gestionnaire d'utilisateurs local PFsense (Voir figure 3.20).

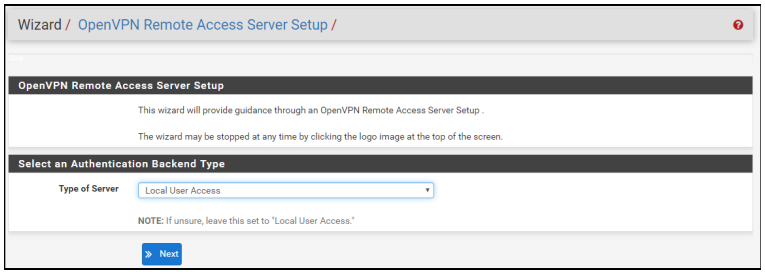


FIGURE 3.20 – Installation wizard de OpenVPN.

- L'étape suivante de la configuration consiste à créer une autorité de certification pour l'émission de certificats. S'il existe déjà une autorité de certification configurée dans PFSense, vous pouvez choisir de l'utiliser pour OpenVPN au lieu d'en créer une nouvelle. Nous choisissons l'autorité de certification « certificate-authority ».
- Si on crée une nouvelle autorité de certification, vous devrez remplir tous les champs de l'assistant pour pouvoir continuer. La longueur de clé par défaut est 2048 bits(suffisant), mais on peut utiliser une clé plus longue. Les tailles de clé plus importantes sont plus sûres mais elles nécessiteront plus de performance.

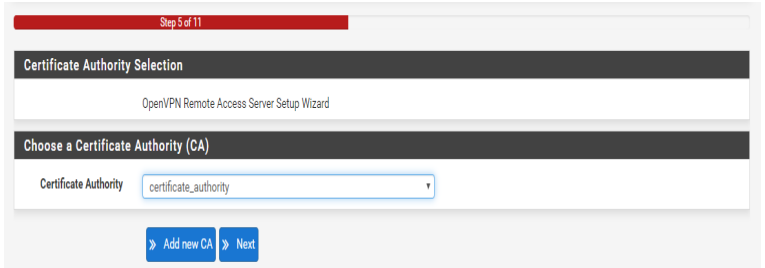


FIGURE 3.21 – Choisissez l'autorité de certification.

- Après que nous avants sélectionné l'autorité de certificat, nous choisissons le certificat du serveur que nous avons déjà créé auparavant.

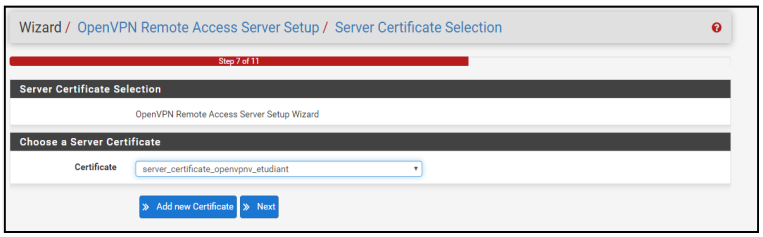


FIGURE 3.22 – Choisir du certificat de serveur.

- Ensuite, on remplit le formulaire de configuration du serveur, qui comprend quatre sections : Informations générales sur le serveur OpenVPN, paramètres cryptographiques, paramètres du tunnel et paramètres du client.
- Commençons par configurer les informations générales du serveur OpenVPN. Nous laisserons le rest tout par défaut en donnant une description de notre OpenVPN.

FIGURE 3.23 – Configurez les paramètres de base de serveur OpenVPN.

Sous "Cryptographic Settings", laissez tout comme valeur par défaut, mais modifiez l'algorithme Auth Digest en SHA256.
- Dans les Paramètres de tunnel, nous faisons entrer la plage d'adresses IP pour le réseau de tunnels (il s'agit de la plage d'adresses IP que OpenVPN utilise pour effectue l'adressage IP aux clients VPN). Vous devez également cocher la case intitulée passerelle de redirection pour assurer que tous les clients utilisent uniquement le VPN pour tout leur trafic. Puis nous saisissons la plage d'adresses IP du réseau local(il s'agit de notre réseau dmz), puis on défini le nombre maximal de connexions simultanées.

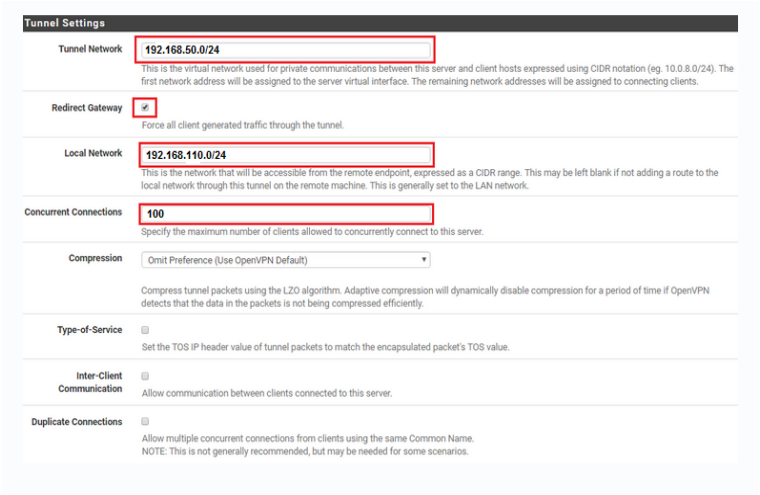


FIGURE 3.24 – Configurez les paramètres du réseau de tunnels.

d) **Définition des réglés d'OpenVPN**

- Lors de l'installation de l'assistant OpenVPN, nous étions en mesure de générer automatiquement les règles de pare-feu nécessaires dans PFsense afin d'autoriser les connexions au serveur OpenVPN. Dans la configurations de base, vous devez activer ces deux options. Si vous n'utilisez pas des règles automatiques, nous devons créer manuellement des règles pour permettre aux clients de se connecter au VPN.
- Dans notre cas nous choisissons de générer les règles manuellement, pour cela on désélectionne les champs qui permet de créer automatiquement les règles de pare-feu et la règle OpenVPN, puis on clique sur "Next". Finalement, on clique sur Terminer pour appliquer tous les paramètres à PFsense.
- Dans l'interface sur laquelle le serveur OpenVPN est en écoute, nous créant une règle qui autorise le trafic à atteindre l'adresse IP et le port du serveur OpenVPN, (Voir figure 3.25). Dans notre exemple, nous travaillons sur l'interface WAN et l'adresse IP de notre pfSense sur notre WAN est 192.168.8.102.

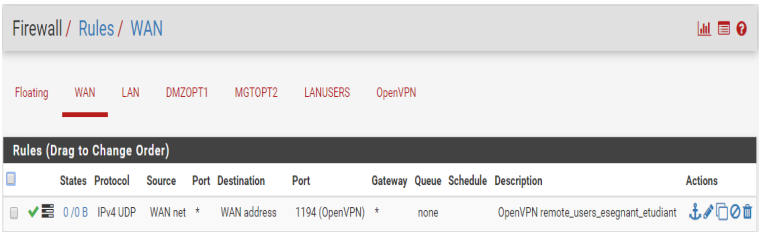


FIGURE 3.25 – Régle autoriser le flux VPN sur l'interface WAN.

- Après que la configuration côté serveur est terminée, il nous reste simplement qu à filtrer nos flux transitant à travers notre nouvelle interface OpenVPN. Les règles qu'on doit ajouter sont les suivantes :

– Trafic de client OpenVPN vers réseau externe (WAN) : autorisé.

– Trafic de client OpenVPN vers la DMZ : autorisé, (Voir figure 3.26).

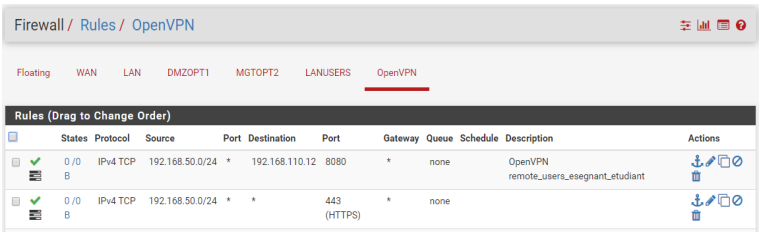


FIGURE 3.26 – Régle autoriser le flux VPN sur l'interface OpenVPN.

e) **Ajouter les utilisateurs OpenVPN**

- Pour ajouter un nouveau utilisateur OpenVPN, ouvrez le menu **System** et cliquez sur **User Manager**.

- Entrez un nom d'utilisateur, un mot de passe et cliquez sur la case à cocher **Certificat** pour générer un certificat utilisateur. Assurez-vous de définir un nom dans le champ nom descriptif, puis cliquez sur le bouton Enregistrer pour terminer le processus d'ajout de l'utilisateur, (Voir figure 3.27).

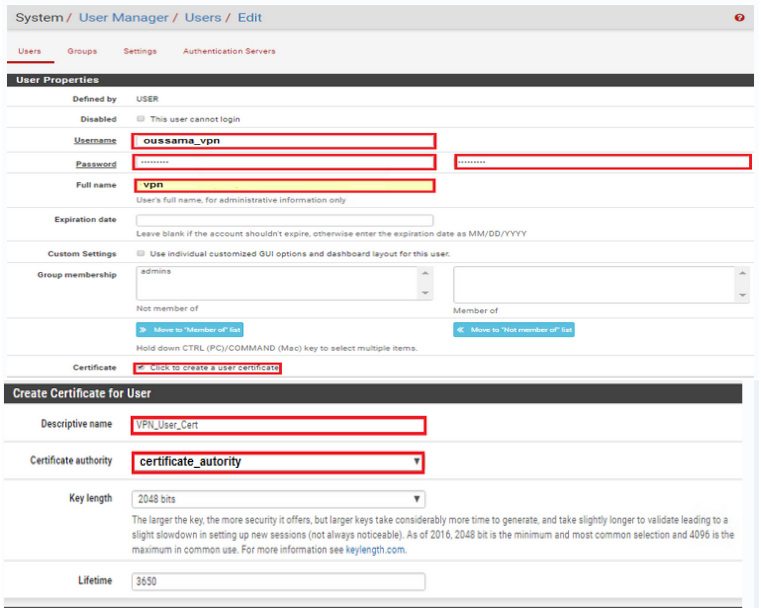


FIGURE 3.27 – Créer comptes utilisateurs OpenVPN.

f) **Export du client OpenVPN**

PFsnse permet, grâce au package installé « OpenVPN Client Export Utility », d'exporter un exécutable pour l'installation du client sous Windows ou autre systèmes (Voir figure 3.28). Si tout est configuré correctement, différentes options de téléchargement vous seront, on click sur :

VPN -> OpenVPN -> onglet'Client Export'.

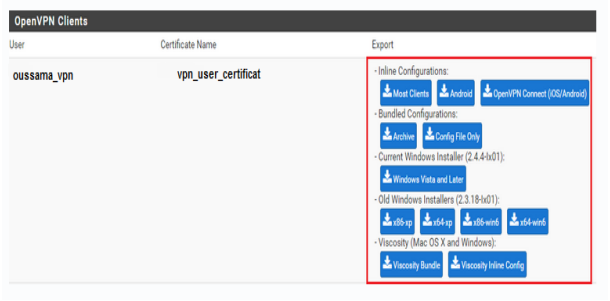


FIGURE 3.28 – Téléchargement des packages clients OpenVPN.

3.7 Scénarios d'exécution de l'application Nextcloud

3.7.1 Scénario d'exécution normal

Dans ce scénario, nous déroulons les actions réalisé par l'enseignant. L'enseignant choisit l'emplacement dans laquelle il importe ses ficher de configuration de l'application OpenVpn installée sur son smartphone, il saisit sont login et clique sur le bouton **ajouter**, (Voir figure 3.29).

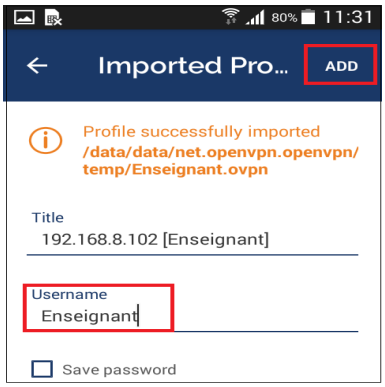


FIGURE 3.29 – Importer les fichers de configuration OpenVPN.

- Si l'enseignant veut connecter avec l'application Nextcloud, il doit cliquer sur bouton **Connecter** où une boite s'affiche sur laquelle il s'authentifier est demandé d'entrer son mot de passe afin d'établir la connexion avec le serveur OpenVPN, (Voir figure 3.30).

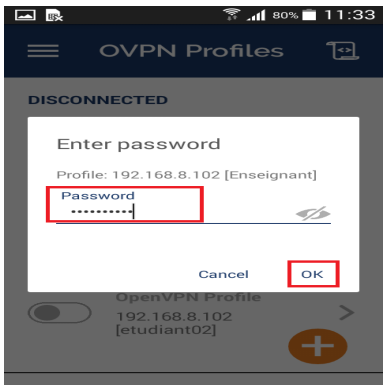


FIGURE 3.30 – Portail de connexion à VPN.

- Ensuite, l'enseignant saisi l'URL de l'application dans la barre d'adresse d'un navigateur web quelconque : http ://192.168.110.12 :8080/ pour accéder à l'interface Nextcloud où il est demandé d'entrer un nom d'utilisateur et un mot de passe, (Voir figure 3.31).

FIGURE 3.31 – Portail de connexion à Nextcloud.

- Maintenant, la page d'accueil de son compte apparue, c'est à lui de faire les différents fonctionnalités(Télécharger un fichier depuis le serveur vers son smartphone, Ajouter un fichier, partage le cour avec un étudiant ou un groupe des étudiants), (Voir figure 3.32).

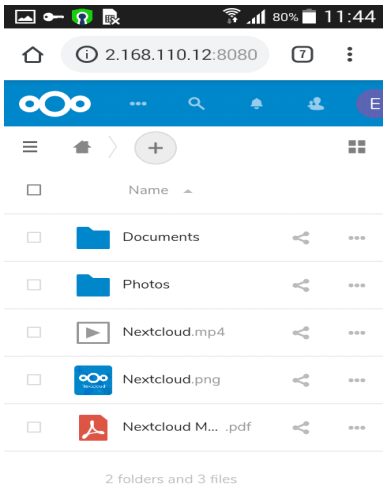


FIGURE 3.32 – Page d'accueil de Nextcloud.

3.7.2 Scénarios d'attaque

Les données en ligne peuvent soumettre à de nombreux scénarios d'attaques, parmi eux citant :

- **Voler des données :**sur internet rien n'est confidentielle, afin d'empêcher d'espionner les indiscrètes et les communications entre les postes distant, le VPN utilise une clé secrète créé lors du processus de négociation d'établissement de la connexion entre l'enseignant et le serveur OpenVPN pour chiffrer les données avant de les transmettre sur internet.

- **Modification des données par autres tiers :**les messages transmis par de nombreux points avant qu'ils nous parvient, il est donc tout à fait possible de les capturer et les modifier en chemin. Imaginez un fichier contient les résultats d'une recherche envoyés à l'université, une modification du bénéficiaire en cours de route serait fâcheuse. Afin de contrer cette menace, chaque portion de message qui passe par le tunnel OpenVPN est munie d'un mécanisme de détection des modifications. En cas d'altération de contenu du message sera écarté.

- **Voler l'identité :**sur internet, les identités d'une personne peut être usurpé facilement par quelqu'un d'autre. Dans notre exemple il convient d'être certain que seules les personnes autorisées auront accès aux données de l'université. C'est pourquoi les VPN utilisent des systèmes de contrôle d'identité. Le système que nous avons choisi se base sur les mots de passes et les certificats.

Conclusion

Dans ce chapitre, nous avons pu décrit la procédure de configuration concernant le VPN par le protocole OpenVPN sous le pare-feu PFsense.

L'objectif était d'inter-connecter le client et le serveur sur le quel est installée l'application Nextcloud via un tunnel sécurisé, nous avons atteint notre objectif comme nous avons pu le constater grâce aux captures ci-haut.

Conclusion Générale

Mon projet consistait à la mise en œuvre d'un VPN d'accès pour le centre universitaire Abdelhafid Boussoufe -Mila-, ayant comme objectif principal d'offrir un moyen sûr et sécurisé qui permettra au enseignants et étudiants d'accéder au réseau interne de l'université par l'Internet, depuis l'extérieur du réseau (leur domicile, place publiques) afin de poursuivre leurs activités sans pour autant risquer de compromettre la sécurité du réseau interne.

Au travers de ce travail, la mise en œuvre du VPN s'est basée sur la solution SSL OpenVPN qui présente l'avantage d'offrir une facilité de configuration et offre une authentification en deux niveaux, au niveau de la machine avec des certificats SSL et au niveau des utilisateurs avec un login et un mot de passe.

Pour conclure, ce contexte ma permit de réaliser un travail dans des conditions similaires à celle d'un vrai poste dans l'administration réseau et faisant appel aux différentes connaissances et un maximum de recherches. En guise de perspectives, nous envisageons de développer une application de partage des fichiers au lieu de Nextcloud dans la plateforme docker et implémenter le VPN dans une université ou au sein d'une entreprise publique ayant plusieurs sites.

Bibliographie

[1] Dumas.J ; Roch.J ; Tannier.É ; Varrette.S, 2007, Livre.théorie des codes(Compression, cryptage, correction), Dunod, Paris.France, 354p.

[2] Dumont.R, 2010, Cryptographie et Sécurité informatique, Thé.Doc, Université de Liège, 261p.

[3] Jean-paul.A, 2013, Les VPN(Foncionnement, mise en oeuvre et maintenance des réseaux privés virtuqls), Editions ENI, France, pp335-360.

[4] Introduction à la cryptologie, cour disponible sur :(http ://www.labouret.net/crypto/), , Consulté en 25 Mai 2019.

[5] Introduction au certificats, cour disponible sur :(https ://web.maths.unsw.edu.au/ lafaye/CCM/crypto/certificat.htm), Consulté en 25 Mai 2019.

[6] Nextcloud User Manual, cour disponible sur : (https ://docs.nextcloud.com/server/12/Nextcloud-User-Manual.pdf). Consulté en 26 Avril 2019.

[7] Site officiel de vmware, disponible sur :(https ://www.vmware.com/support/ws5/doc/ws-net.html). Consulté en 12 Mai 2019.

[8] Disponible sur :(http ://www.formations-virtualisation.fr/definition-de-vmware/)

[9] EVE-NG Professional Cookbook, cour disponible sur :(http ://www.eve-ng.net/images/EVE-COOK-BOOK-1.0.pdf). Consulté en 20 Mars Avril 2019.

[10] Documentation ClearOS, disponible sur :(https ://www.clearos.com/resources/documentation/clea guide6). Consulté en 27 mai 2019.

[11] Meilleurs pare-feu Linux gratuits de 2019, disponible sur :(https ://www.techradar.com/news/best-free-linux-firewall). Consulté en 27 mai 2019.

[12] Site officiel de Docker, disponible sur :(https ://docs.docker.com/engine/docker-overview/). Consulté en 12 Avril 2019.

[13] Principe de fonctionnement de VPN, cour disponible sur :(http ://electro-media.blogspot.com). Consulté en 5 juin 2019.

[14] Principe de fonctionnement de VPN, cour disponible sur :(https ://googleweblight.com/i ?u=https ://www.frameip.com/vpn/hl=fr-DZ). Consulté en 5 juin 2019

[15] Disponible sur :(http ://www.marchepublic.fr/Terminologie/Entrees/virtualisation.htm)

[16] Disponible sur :(http ://www.qsp-systems.com/la-virtualisation-mode-demploi/)

[17] Antoine.W, 2015, étude de la sécurité d'algorithmes de cryptographie embarquée vis-à-vis des attaques par analyse de la consommation de courant, Thé.Doc, Université de Limoges, pp8-9.

Résumé

Dans ce mémoire, nous nous somme basés sur des solutions de sécurité standardisées pour mettre en place une architecture sécurisée de partage des documents universitaires (examen, cours...). Ce qui permet l'échange des documents entre les différents acteurs de l'université notamment les étudiants et les enseignants d'une manière fiable et sécurisé.

L'architecture proposée se base sur l'OpenVPN qui est considéré comme un standard pour le VPN. L'OpenVPN se base sur SSL/TLS, il est très répondu et fournit des services de sécurité comme la confidentialité, l'intégrité, l'authentification.

L'architecture proposé est fondu sur trois élément principales, i) La plateforme EVE-ng pour simuler notre topologie, ii)le pare-feu PFsense pour la configuration de OpenVPN et iii) docker pour la mise en œuvre de partage des fichiers Nextcloud.

Mots.clés : VPN, OpenVPN, SSL/TLS, Certificat, PFsense, Docker, EVE-ng.

Abstract

In this thesis, we have based ourselves on standardized security solutions to set up a secure architecture for sharing university documents (exam, courses...). This allows the exchange of documents between the different actors of the university, in particular students and teachers, in a reliable and secure way.

The proposed architecture is based on OpenVPN which is considered as a standard for VPN. OpenVPN is based on SSL/TLS, it is highly responsive and provides security services such as privacy, integrity, and authentication.

The proposed architecture is based on three main elements, i)The EVE-ng platform to simulate our topology, ii)the PFsense firewall for OpenVPN configuration and iii) docker for Nextcloud file sharing implementation.

Keywords : VPN, OpenVPN, SSL/TLS, Certificat, PFsense, Docker, EVE-ng.