Sécurité des Systèmes Avancée

Houssem MAHMOUDI houssem.mahmoudi@ensicaen.fr

AU : 2024-2025

L'École des INGÉNIEURS Scientifiques

CHAPITRE 3 : SÉCURITÉ DES DONNÉES ET COMMUNICATIONS

Introduction

I. INTRODUCTION

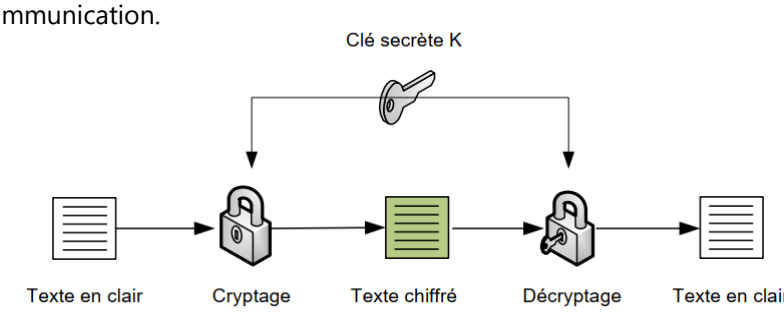
- Plusieurs approches pour sécuriser les échanges d'informations dans un réseau TCP/IP.
- Utilisation des protocoles de communication dans leurs versions sécurisées : HTTPS, FTPS, SSH,…
- Chiffrer la communication entre le client et serveur par l'utilisation de SSL/TLS, IPSec

Chiffrement

II. CHIFFREMENT

1. Chiffrement symétrique

- Le chiffrement symétrique, ou cryptage à clé secrète, consiste à utiliser la même clé pour le chiffrement et le déchiffrement.
- Cette clé secrète ne doit être connue que par les deux intervenants en communication.



II. CHIFFREMENT

1. Chiffrement symétrique

Avantages	Inconvénients
• Rapide et chiffre grande quantité des données	• Problème d'échange de clés.
• Utilise peu de ressources système.	• La non-répudiation, l'intégrité et la confidentialité n'est pas assurée.

II. CHIFFREMENT

1. Chiffrement symétrique

- **Exemple des algorithmes :**
  - **DES** : (**D**igital **E**ncryption **S**tandard) : Algorithme de chiffrement qui utilise des blocs de 64 bits et une clé de 56 bits. Il est considéré comme vulnérable aux attaques par force brute.
  - **3DES** : Version renforcée de DES qui applique l'algorithme DES trois fois de suite avec deux ou trois clés différentes.
    - 3DES à trois clés (K1 ≠ K2 ≠ K3, sécurité maximale, clé de 168 bits).
    - 3DES à deux clés (K1 = K3, clé de 112 bits)
    - 3DES à une clé (K1 = K2 = K3, clé de 56 bits)

II. CHIFFREMENT

1. Chiffrement symétrique

- **Exemple des algorithmes :**
  - **IDEA** : (**I**nternational **D**ata **E**ncryption **A**lgorithm) : IDEA est un algorithme de chiffrement par bloc de taille 64 bits et des clés de 128 bits. Il a été utilisé dans des applications comme PGP (Pretty Good Privacy).
  - **Blowfish** : Algorithme de chiffrement par bloc de taille 64 bits et utilise des clés de longueur variable allant jusqu'à 448 bits

II. CHIFFREMENT

1. Chiffrement symétrique

- **Exemple des algorithmes :**
  - **AES** (**A**dvanced **E**ncryption **S**tandard) : standard de cryptographie approuvé par NIST largement utilisé. Il fonctionne avec des blocs de 128 bits et utilise des clés de taille (128 bits, 192 bits, 256 bits). Il est Considéré comme efficace et résistant aux attaques.

II. CHIFFREMENT

1. Chiffrement symétrique - Démo

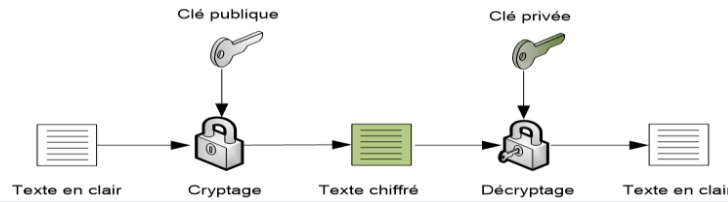
1. Créer une clé de chiffrement symétrique avec l'algorithme AES.
2. Afficher le contenu de cette clé.
3. Utiliser cette clé pour **chiffrer** un fichier texte.
4. Utiliser cette clé pour **déchiffrer** le fichier chiffré.

OpenSSL

II. CHIFFREMENT

2. Chiffrement asymétrique

- Appelée aussi cryptographie à clé publique, cette technique utilise deux clés :
  - **Une clé publique** : qui sert à chiffrer le message et qui peut être diffusée sans créer de risques.
  - **Une clé privée** : accessible seulement par son propriétaire et qui sert à déchiffrer le message.
- Seul le détenteur de la clé privée est en mesure de lire le message



II. CHIFFREMENT

2. Chiffrement asymétrique

Avantages	Inconvénients
• Sûre	• Nécessite une puissance de calcul et de mémoire.
• Fiable	• Lenteur (complexité mathématique).
• Durable	• Chiffre petite quantité des données.
	• Vérifier le détenteur du clé publique

II. CHIFFREMENT

2. Chiffrement asymétrique

- **Exemple des algorithmes :**
  - **RSA** : (**R**ivest-**S**hamir-**A**dleman) très utilisé dans le e-commerce et pour échanger des données confidentielles sur Internet. Clé varies entre 1024 et 4096 bits.
  - **Diffie-Hellman :** Un protocole d'échange de clés qui permet à deux parties de générer une clé secrète partagée sur un canal non sécurisé.
    - Échange de clés sécurisés (exemple: TLS/SSL).
    - Deux variantes : DH classique et ECDH (Elliptic Curve Diffie-Hellman)

II. CHIFFREMENT

2. Chiffrement asymétrique

- **Exemple des algorithmes :**
  - **El-Gamal** : inventé par Taher Elgamal, un algorithme de chiffrement asymétrique basé sur le problème du logarithme discret. Il est utilisé pour le chiffrement et la signature numérique.
  - **ECC :** (Elliptic Curve Cryptography), Utilise les mathématiques des courbes elliptiques pour offrir une sécurité équivalente à RSA mais avec des clés beaucoup plus petite.

II. CHIFFREMENT

2. Chiffrement asymétrique - Démo

1. Utiliser l'algorithme RSA pour générer une clé privée « **private.key** » de taille 2048 bits.
2. Afficher la clé privée.
3. En se basant sur la clé privée, créer la clé publique « **public.key** ».
4. Afficher la clé publique et la comparer avec la clé privée.
5. Chiffrer le fichier **message.txt** avec votre clé publique **message_crypt.txt**
6. Déchiffrer le fichier **message_crypt.txt** avec la clé privée.

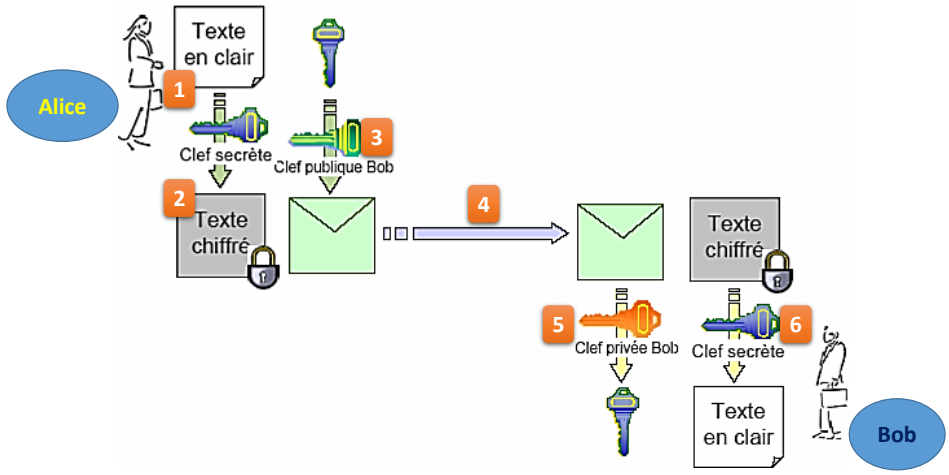
II. CHIFFREMENT

3. Chiffrement hybride

- Utilise à la fois la technique de chiffrement symétrique et celle du chiffrement asymétrique.
- **Principe :**
  - Chiffrement symétrique du message avec une clé secrète (clé de session).
  - Ensuite chiffrement de la clé secrète par la clé publique de destinataire.
  - Message chiffré et clé chiffré seront tous les deux envoyés au destinataire.
  - Le destinataire déchiffre la clé secrète par sa clé privée.
  - Ensuite il déchiffre le message par la clé obtenu

II. CHIFFREMENT

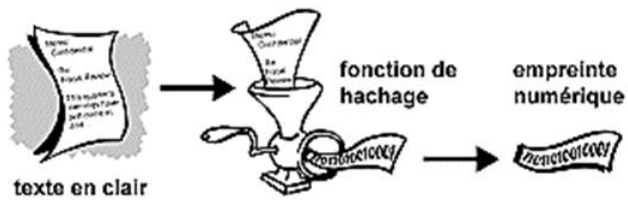
3. Chiffrement hybride



II. CHIFFREMENT

4. Fonction de hachage

- Une fonction de hachage est une fonction à sens unique permettant d'obtenir un hach ou empreinte digitale d'un message (image, texte, fichier,…).
- Il est impossible de retrouver le message original à partir de l'empreinte.
- La fonction de hachage est utilisée pour garantir **l'intégrité** des données.



II. CHIFFREMENT

4. Fonction de hachage

- **Exemple des algorithmes :**
  - **MD5** : (Message Digest 5) permet de créer une empreinte digitale de taille 128 bits.
  - **SHA** : (Secure Hash Algorithm) permet de créer des empreintes d'une longueur de 128 bits.
  - **SHA-1** : Version améliorée de SHA datant en 1994 et produisant une empreinte de 160 bits.
  - **SHA-2** : Successeur de sha-1, il comprend quatre types de hash : sha-224, sha-256, sha-384 et sha-512. Il fonctionne sur le même principe que sha-1 mais plus résistant aux attaques.

II. CHIFFREMENT

4. Fonction de hachage

- **Vérification de l'intégrité du message :**
  - Pour garantir l'intégrité du message, on peut envoyer un message accompagné par son empreinte.
  - Le destinataire peut vérifier l'intégrité du message en calculant à nouveau le hach du message et le comparé par ce qui à reçu.
  - Si les deux empreintes sont égales, alors l'intégrité du message est garantie

II. CHIFFREMENT

4. Fonction de hachage - Démo

1. Calculer l'empreinte du fichier info.txt pour les fonctions de hachage suivantes et déterminer la longueur de chaque empreinte :
   a. MD5
   b. SHA1
   c. SHA256
   d. SHA384
   e. SHA512

OpenSSL

II. CHIFFREMENT

5. Signature numérique

- La signature numérique d'un document a pour objectifs de prouver **l'identité** de l'auteur du message et **d'empêcher** qu'on modifie le message.
- Se base sur la technique de chiffrement et la technique de hachage.
- La signature numérique est **authentique** et ne peut ni être falsifiée ni imitée. Elle ne peut pas être reniée ni réutilisée pour signer un autre document.
- **DSA** (**D**igital **S**ignature **A**lgorithm) : Un standard pour les signatures numériques, utilisé pour les certificat (SSL/TLS).
- **ECDSA** : **E**lliptic **C**urve **D**igital **S**ignature **A**lgorithm) : Une version plus efficace basée sur les courbes elliptiques.

II. CHIFFREMENT

5. Signature numérique – Principe

A l'émission du message M

- Appliquer une fonction de hachage **H** au message clair M pour avoir une empreinte « **e** » tel que : e=H(M).
- Chiffrer l'empreinte e, par la clé privé de l'expéditeur $\mathbf{K_{pr}}$ pour obtenir une signature **S**, tel que S= K_{pr}(e).
- Envoyer le message et la signature au destinataire, c'est-à-dire [M + S].

II. CHIFFREMENT

5. Signature numérique – Principe

A la réception du message M signé :

- Déchiffrer la signature S par la clé publique K_{pu} de l'expéditeur ; $K_{pu}(S)$ = **e**
- Appliquer la même fonction de hachage H au message d'origine M pour avoir un condensé **e'**=H(M).
- Comparer e et e' ; S'ils sont identiques, alors le message envoyé est valide sinon il a été intercepté et altéré.

II. CHIFFREMENT

6. Certificat numérique

- Appelé aussi certificat électronique, est un document électronique qui **authentifie** l'identité d'un individu, d'une organisation ou d'un dispositif.
- Le certificat est crée par une Autorité de Certification (CA : Certificate Authority)
- Un certificat permet de garantir :
  - La sécurité des transactions numériques,
  - L'authenticité,
  - L'intégrité des communications.

II. CHIFFREMENT

6. Certificat numérique

- Un certificat numérique contient une série d'informations :
  - Nom du certificat et son utilisation,
  - Informations identifiant le propriétaire,
  - Clé publique,
  - Date d'expiration du certificat.
  - Nom de l'organisme de certification.

- Le CA utilise sa **clé privée** pour **signer les certificats** et assure ainsi une sécurité supplémentaire.
- Ainsi le CA permet de valider l'origine de la clé publique.

Protocoles sécurisés

III. Protocoles sécurisés

1. Protocoles SSL/TLS

- Les protocoles **SSL** (Secure Socket Layer) et **TLS** (Transport Layer Security) permettent de sécuriser les transactions sur Internet.
- SSL/TLS sont Utilisés en commerce en ligne et le paiement électronique pour chiffrer le numéro de la carte bancaire qui sera déchiffrer sur le serveur du marchand.
- Les deux protocoles **SSL 2.0** et **SSL 3.0** sont vulnérables.
- Actuellement **TLS 1.2** et **TLS 1.3** sont les deux versions recommandées dans une communication sécurisée mieux que SSL.

III. Protocoles sécurisés

1. Protocoles SSL/TLS

- SSL et TLS permettent d'aboutir aux objectifs suivants :
  - **Authentification** : Le client doit pouvoir s'assurer de l'identité du serveur. Cela est réalisé par l'emploi de certificat électronique.
  - **Confidentialité** : Toutes les données qui transitent entre le client et le serveur sont chiffrées par l'émetteur, et déchiffrées par le destinataire
  - **Identification et intégrité** : Le client et le serveur doivent pouvoir s'assurer que les messages transmis ne sont altérés. Ces fonctionnalités sont assurées par la signature.

III. PROTOCOLES SÉCURISÉS

Version TLS - Démo

1. Afficher le certificat et la version de protocole TLS utilisée par les sites suivants :
    a. Facebook.com
    b. Gmail.google.com
    c. Ensicaen.fr

III. PROTOCOLES SÉCURISÉS

2. Protocole HTTPS

1. Le protocole HTTPS (HyperText Transfer Protocol Secure) est une version sécurisée du protocole HTTP, utilisé pour la communication sur le Web.
2. HTTPS assure la confidentialité, l'intégrité des données et l'authenticité entre un client (navigateur) et un serveur Web
3. Le protocole HTTPS se base sur l'utilisation des protocoles SSL/TLS.

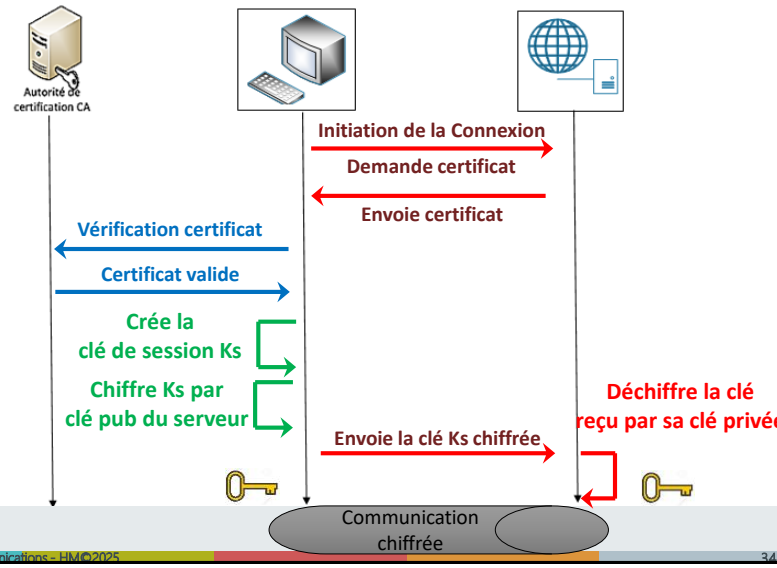
III. Protocoles sécurisés

2. Protocoles HTTPS – Principe de fonctionnement

1. **Initiation de la Connexion** : Lorsque l'utilisateur entre une adresse URL commençant par « **https://** », le navigateur envoie une demande de connexion et au certificat du serveur.
2. **Vérification du certificat** : Le client vérifie la validité du certificat du serveur (authenticité) auprès du CA.
3. **Echange de clé de session** : Si le certificat est valide, le client crée une clé secrète aléatoire (clé de session), la chiffre avec la clé publique du serveur puis envoie le résultat au serveur.
4. **Communication sécurisée** : Le serveur déchiffre la clé de session avec sa clé privée. La clé secrète est connue, toutes les données échangées vont être cryptées

III. Protocoles sécurisés

2. Protocoles HTTPS – Principe de fonctionnement



III. Protocoles sécurisés

3. Protocole SSH

- SSH (Secure Shell) permet d'établir une connexion sécurisée entre deux systèmes informatiques.
- Le protocole SSH est utilisé pour réaliser des tâches d'administration à distance des serveurs, routeurs ou les Firewalls.
- SSH fonctionne selon le modèle client-serveur et utilise le chiffrement pour sécuriser la connexion, il permet de garantir la confidentialité, l'intégrité et authenticité.
- SSH a remplacé les anciens protocoles comme Telnet et FTP qui transmettent les données en clairs sur le réseau.

III. Protocoles sécurisés

3. Protocole SSH – Avantages

- **Sécurité des communications** : SSH offre une confidentialité totale des communications en utilisant des algorithmes de chiffrement puissants.
- **Intégrité des données** : SSH utilise des algorithmes de hachage pour garantir que les données échangées dans le réseau n'ont pas été altérées.
- **Authentification** : SSH supporte plusieurs méthodes d'authentification, comme mot de passe, clé publique/privée et 2FA.
- **Interopérabilité** : SSH est compatible avec plusieurs systèmes d'exploitation, ce qui le rend accessible à un grand nombre utilisateurs.

III. PROTOCOLES SÉCURISÉS

4. FTPS

- **FTPS** (File Transfer Protocol Secure) est une version sécurisée du protocole FTP standard, qui ajoute une couche de chiffrement pour protéger les transferts de fichiers. Voici les principales caractéristiques du FTPS.
- Le protocole FTPS offre :
  - Chiffrement des données et des commandes,
  - Authentification du serveur via certificats,
  - Protection contre l'interception des identifiants et des données transférées.

III. PROTOCOLES SÉCURISÉS

5. SMTPS

- **SMTPS** (**S**imple **M**ail **T**ransfer **P**rotocol **S**ecure) est une version sécurisée du protocole SMTP standard pour l'envoi d'emails.
- SMTPS utilise le protocole SSL/TLS pour chiffrer les communications entre le client de messagerie et le serveur SMTP.
- Le protocole SMTPS offre :
  - Chiffrement des données et des commandes
  - Authentification sécurisée du client
  - Protection contre l'interception des identifiants et du contenu des emails

VPN

IV. VPN

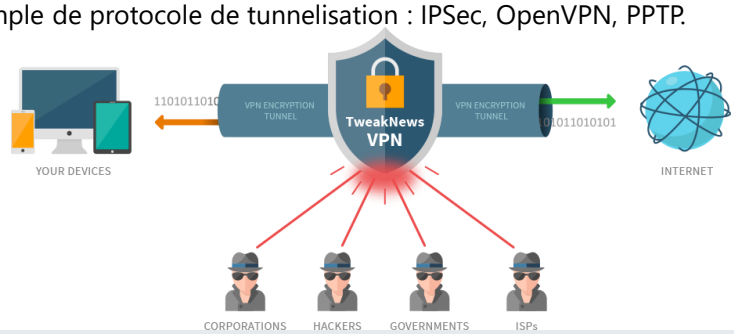
1. Définition

- **VPN** : **V**irtual **P**rivate **N**etwork (pour réseau privé virtuel), est une technique qui permet d'établir un canal chiffré (**tunnel**) entre deux nœuds quelconques de l'Internet. Le VPN permet d'offrir :
  - **Confidentialité** : VPN réalise le chiffrement des données.
  - **Anonymat** : VPN masquant l'adresse IP du client et permet de naviguer de manière plus anonyme.
  - **Sécurité** : Le chiffrement protège vos informations sensibles, notamment sur les réseaux Wi-Fi publics.
  - **Contournement des restrictions géographiques** : VPN permet d'accéder à des contenus bloqués dans votre région.

IV. VPN

2. Fonctionnement

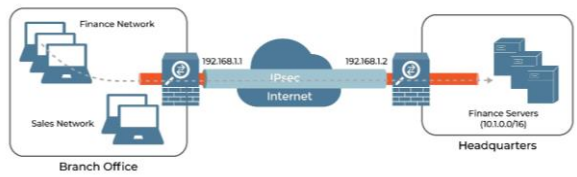
- Le VPN repose sur un **protocole de tunnelisation** (Tunneling).
- Un protocole de tunnelisation permet d'encapsuler et transporter les données d'une manière sécurisée.
- Exemple de protocole de tunnelisation : IPSec, OpenVPN, PPTP.



IV. VPN

3. Types de VPN

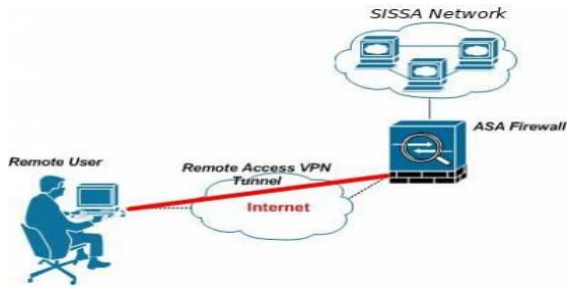
VPN de site à site (Site-to-Site VPN) : Permet de créer un tunnel sécurisé entre deux réseaux locaux distincts via Internet, comme des bureaux distants ou des succursales d'une entreprise.



IV. VPN

3. Types de VPN

VPN d'accès distant (Remote-Access VPN) : Ce type de VPN permet à des utilisateurs individuels d'accéder, d'une façon sécurisée, à un réseau privé via le réseau Internet.



IPSec

V. IPSec

1. Définition

- Le protocole **IPSec** (Internet Protocol Security) désigne un ensemble des mécanismes destinés à protéger le trafic au niveau de la couche réseau du modèle OSI.
- IPSec permet d'offrir :
  - L'intégrité des données,
  - L'authentification de l'origine des données,
  - La confidentialité des données : protection contre le Sniffing
  - Protection contre l'analyse du trafic et IP Spoofing.

V. IPSec

2. Modes du protocole IPSec

a. Mode transport :

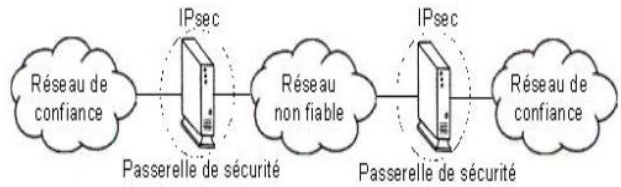
- On chiffre et/ ou authentifie la partie data (payload) d'un paquet IP excepté les champs variables de l'en-tête (TTL,….)
- Les machines source et destination sont les 2 extrémités de la connexion sécurisée.

V. IPSec

2. Modes du protocole IPSec

b. Mode tunnel :

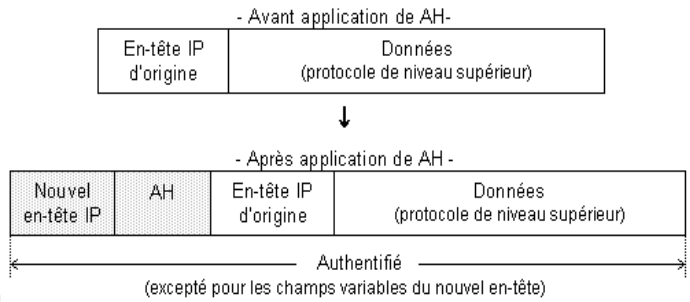
- Le paquet complet (en-tête IP + donnée) est encapsulé dans un nouveau paquet, avec un nouvel en-tête IP, qui sera chiffrer et transmise.
- les extrémités de la connexion sécurisée sont des passerelles qui permettent de masquer les adresses IP source et destination d'origine, assurant une protection contre l'analyse de trafic.



V. IPSec

3. Architecture du protocole IPSec

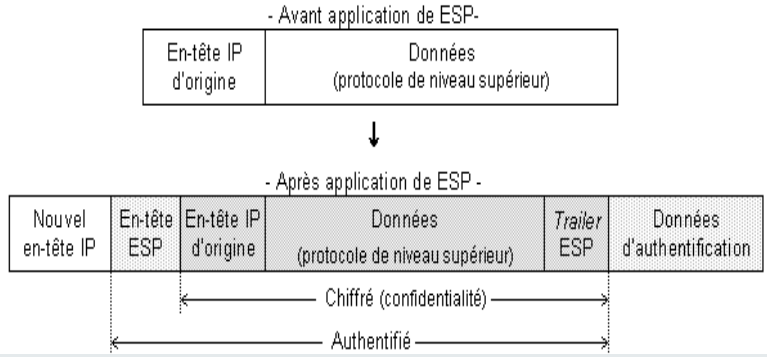
- le protocole IPsec fait appel à des mécanismes de sécurité sur le trafic IP qui sont :
- **Protocole AH** (Authentification Header) : Permet d'assurer l'intégrité et l'authentification des paquets IP <u>sans chiffrement des données.</u>



V. IPSEC

3. Architecture du protocole IPSec

- **Protocole ESP** (**E**ncapsulating **S**ecurity **P**ayload) : Ce protocole a pour but d'assurer la confidentialité des données et générer un nouveau paquet, les données et l'entête originale sont chiffrées.



V. IPSEC

3. Architecture du protocole IPSec

- **Protocole IKE** : (Internet Key Exchange) est un protocole fondamental dans l'établissement d'une communication sécurisée, il se base sur l'algorithme d'échange de clés comme **Diffie-Hellman**.
- Son rôle principal est de négocier, établir et gérer les clés cryptographiques utilisées pour sécuriser les échanges de données.
- Le protocole IPSec s'appuie sur IKE pour négocier les clés et établir des associations de sécurité (SA).

V. IPSec

3. Architecture du protocole IPSec

	Protocole AH	Protocole ESP
Mode Transport	Intégrité et authenticité des données.	Chiffrement, Intégrité et Authenticité des données. Entête IP non chiffrée.
Mode Tunnel	Authentification et intégrité du paquet complet (nouvel Entête-IP).	Chiffre le paquet IP original (en-tête + données). Nouvel en-tête IP externe reste en clair pour le routage.

V. IPSec

4. Capture du trafic IPSec

No.	Time	Source	Destination	Protocol	Length	Info
9	24.166099	172.16.0.1	172.16.0.2	ESP	124	ESP (SPI=0x8379a96e)
10	24.171513	172.16.0.2	172.16.0.1	ESP	124	ESP (SPI=0x1382e3df)
11	24.189728	172.16.0.1	172.16.0.2	ESP	108	ESP (SPI=0x8379a96e)
12	24.190504	172.16.0.1	172.16.0.2	ESP	444	ESP (SPI=0x8379a96e)
13	24.196837	172.16.0.2	172.16.0.1	ESP	108	ESP (SPI=0x1382e3df)
14	24.215748	172.16.0.2	172.16.0.1	ESP	636	ESP (SPI=0x1382e3df)
15	24.225200	172.16.0.2	172.16.0.1	ESP	1500	ESP (SPI=0x1382e3df)
16	24.233218	172.16.0.2	172.16.0.1	ESP	1500	ESP (SPI=0x1382e3df)
17	24.233310	172.16.0.1	172.16.0.2	ESP	124	ESP (SPI=0x8379a96e)
18	24.233351	172.16.0.1	172.16.0.2	ESP	124	ESP (SPI=0x8379a96e)
19	24.233597	172.16.0.2	172.16.0.1	ESP	236	ESP (SPI=0x1382e3df)
20	24.249161	172.16.0.1	172.16.0.2	ESP	124	ESP (SPI=0x8379a96e)
21	24.249247	172.16.0.1	172.16.0.2	ESP	108	ESP (SPI=0x8379a96e)
22	29.214551	172.16.0.2	172.16.0.1	ESP	108	ESP (SPI=0x1382e3df)

```
▷ Frame 25: 108 bytes on wire (864 bits), 108 bytes captured (864 bits) on interface -, id 0
▷ Cisco HDLC
▷ Internet Protocol Version 4, Src: 172.16.0.2, Dst: 172.16.0.1
◢ Encapsulating Security Payload
    ESP SPI: 0x1382e3df (327345119)
    ESP Sequence: 11
```

OpenVPN

VI. OpenVPN

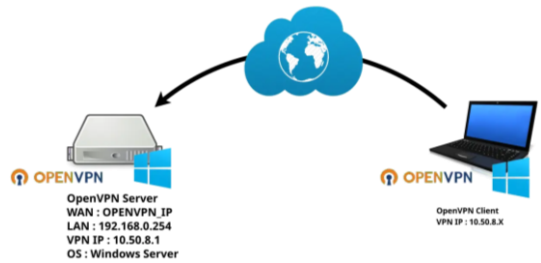
1. Définition

- OpenVPN est une solution VPN open-source qui utilise les protocoles SSL/TLS pour créer des tunnels sécurisés sur Internet.
- OpenVPN permet d'offrir :
  - Sécurité : Prise en charge de l'authentification par certificats et clés.
  - Compatibilité : Disponible sur tous les OS,
  - Flexibilité : Fonctionne en mode TCP (fiabilité) ou UDP (rapidité).
  - Faible coût : Gratuit et open source.

VI. OpenVPN

2. Fonctionnement d'OpenVPN

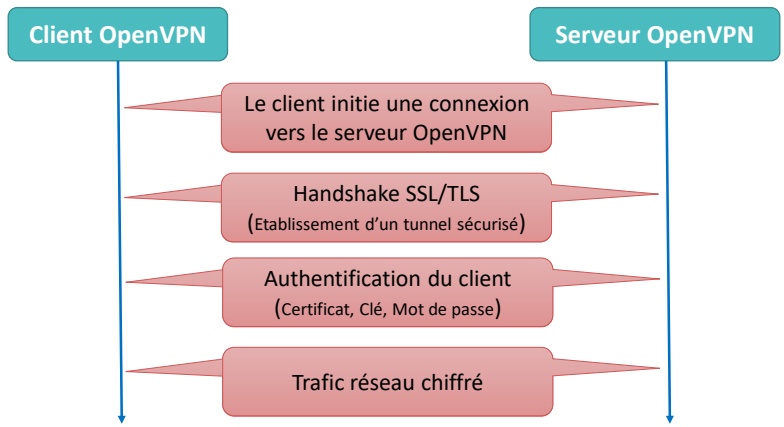
- OpenVPN fonctionne sur un modèle client-serveur :
  - **Serveur OpenVPN** : Gère les connexions VPN et l'authentification des clients.
  - **Client OpenVPN** : Se connecte au serveur pour accéder au réseau distant.



VI. OpenVPN

2. Fonctionnement d'OpenVPN

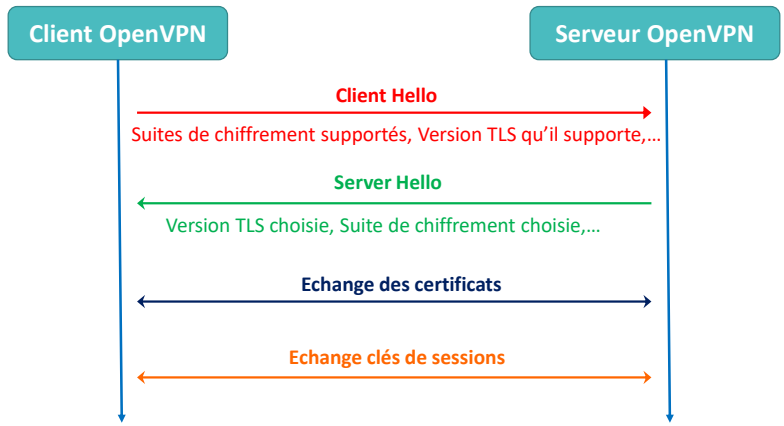
- Le processus de connexion avec le serveur OpenVPN est le suivant :



VI. OpenVPN

2. Fonctionnement d'OpenVPN

- **Handskake SSL/TLS**



VI. OpenVPN

3. Caractéristiques d'OpenVPN

- OpenVPN supporte plusieurs méthodes d'authentification :
  - Par clé pré-partagée (Pre-Shared Key PSK).
  - Par certificats SSL/TLS.
  - Par login/mot de passe (serveur d'authentification : Radius, LDAP).
- OpenVPN utilise OpenSSL pour réaliser le chiffrement (AES-256, ChaCha20)
- OpenVPN utilise les ports :
  - UDP 1194 : Rapide et recommandé (par défaut).
  - TCP 443 : Peut être utilisé pour contourner le pare-feu.

Evaluation des connaissances

V. Evaluation des connaissances

Question n° 1

- Vous devez mettre en place un système de cryptographie dans votre organisation. Vous devez pouvoir envoyer de grandes quantités de données, rapidement, sur le réseau. Le système ne sera utilisé que par un très petit groupe d'utilisateurs et l'échange de clés ne pose pas de problème. Lequel des éléments suivants devez-vous prendre en compte ?
  - A. Chiffrement asymétrique
  - B. Chiffrement hybride
  - C. Chiffrement symétrique
  - D. Chiffrement par bit

V. Evaluation des connaissances

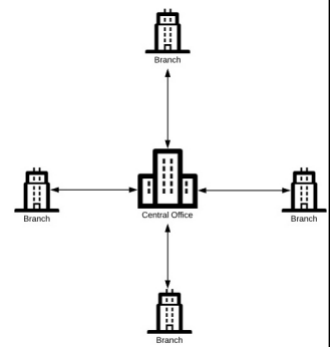
Question n° 2

- Si une personne crypte un message avec sa propre clé privée, qu'est-ce que cela garantit ?
  - A. Confidentialité
  - B. Authenticité du message
  - C. Intégrité
  - D. Disponibilité

V. Evaluation des connaissances

Question n° 3

- Tom prévoit le déploiement d'un nouveau VPN, illustré dans le diagramme de haut niveau présenté ici. Quel type de VPN Tim déploie-t-il ?
  - A. TLS VPN
  - B. Remote access VPN
  - C. Site-to-site VPN
  - D. IPsec VPN



V. Evaluation des connaissances

Question n° 4

- Lors d'un audit de sécurité de l'environnement Web de son organisation, Robert découvre que son serveur Web prend en charge SSL v2.0. Quelle action doit-il recommander en fonction de ces informations ?

  A. L'organisation devrait remplacer SSL par TLS.

  B. L'organisation doit désactiver SSL v2.0 et prendre en charge uniquement SSL v3.0 ou supérieur.

  C. L'organisation devrait remplacer SSL with SSH.

  D. Aucune action n'est nécessaire

V. Evaluation des connaissances

Question n° 5

- Laquelle des affirmations suivantes concernant les protocoles IPsec est correcte ?

  A. AH prend en charge l'authentification, l'intégrité et la confidentialité. ESP prend en charge la confidentialité et l'authentification

  B. AH prend en charge l'authentification, l'intégrité et la confidentialité. ESP prend en charge la confidentialité et l'intégrité.

  C. AH prend en charge l'authentification et l'intégrité. ESP prend en charge la confidentialité, l'authentification et l'intégrité

  D. AH prend en charge l'authentification et la confidentialité. ESP prend en charge l'intégrité et l'authentification

V. Evaluation des connaissances

Question n° 6

Les visiteurs du site Web de l'organisation de Michel voient le message d'erreur suivant. Quelle est la manière la plus simple pour Michel de résoudre ce problème ?

A. Exiger l'utilisation de TLS
B. Renouveler le certificat
C. Remplacer le certificat
D. Bloquer les chiffrements non sécurisés

V. Evaluation des connaissances

Question n° 7

Quelle est la principale vulnérabilité de l'implémentation des certificats HTTPS lorsque l'autorité de certification (CA) n'est pas correctement vérifiée ?

A. L'attaque par injection SQL
B. L'attaque par interception des cookies de session
C. L'usurpation de certificat, permettant un attaquant d'intercepter et de décrypter les données
D. La fuite d'informations par le biais du cache du navigateur

V. Evaluation des connaissances

Question n° 8

Dans quels deux modes IPSec peut-il fonctionner ?

A. Tunneling et Storing
B. Transport et Storing
C. Tunneling et Transport
D. At-Rest et At-Ease

V. Evaluation des connaissances

Question n° 9

Par quel moyen la clé de session est envoyée au destinataire ?

A. Envoyer la clé de session chiffrée par la clé privée de la destinataire.
B. Envoyer la clé de session chiffrée par la clé publique de l'émetteur.
C. Envoyer l'empreinte de la clé de session, le destinataire s'en charge d'en déduire.
D. Envoyer la clé de session chiffrée par la clé publique de la destinataire.

V. Evaluation des connaissances

Question n° 10

Quelle est la principale différence entre un VPN basé sur TLS et un VPN classique utilisant IPSec ?

A. Les VPN TLS n'utilisent pas d'encapsulation des paquets IP

B. Les VPN TLS nécessitent toujours un client logiciel dédié

C. Les VPN IPSec fonctionnent uniquement sur IPv4 tandis que TLS peut être utilisé sur IPv6

D. Les VPN TLS chiffrent les données applicatives, tandis qu'IPSec (en mode tunnel) chiffre les en-têtes IP, les données de la couche transport et les couches supérieures.