Introduction à la sécurité

Sécurité des systèmes

Daniel De Almeida Braga

whoami

- Daniel De Almeida Braga
- Ancien ingénieur analyste chez Amossys
- Enseignant-chercheur à l'ISTIC/IRISA
- Thématique cybersécurité

Mes enseignements

- Cryptographie
- Sécurité système
- Programmation C
- ...

Ma recherche (équipe CAPSULE)

- Sécurité des implémentations cryptographiques
- Sécurité des microarchitectures

Références conseillées

- Cours en ligne de Gildas Avoine
- Rapports de l'ENISA (par exemple, celui de l'état des menaces)
- MOOC de l'ANSSI
- Guide d'hygiène numérique de l'ANSSI
- LiveOverflow (Youtube)
- pwn.college

Plan du cours

1. Sécurité des systèmes
2. Panorama des menaces
3. Mécanismes de protection
4. L'authentification
5. Le contrôle d'accès

Sécurité des systèmes

Qu'est-ce que la sécurité d'un système ?

- Ensemble des mesures techniques et non techniques de protection

Qu'est-ce que la sécurité d'un système ?

- Ensemble des mesures techniques et non techniques de protection
- permettant à un système d'information de résister

Qu'est-ce que la sécurité d'un système ?

- Ensemble des mesures techniques et non techniques de protection
- permettant à un système d'information de résister
- à des événements susceptibles de compromettre :

Qu'est-ce que la sécurité d'un système ?

- Ensemble des mesures techniques et non techniques de protection
- permettant à un système d'information de résister
- à des événements susceptibles de compromettre :
  - la disponibilité

Qu'est-ce que la sécurité d'un système ?

- Ensemble des mesures techniques et non techniques de protection
- permettant à un système d'information de résister
- à des événements susceptibles de compromettre :
  - la disponibilité
  - l'intégrité

Qu'est-ce que la sécurité d'un système ?

- Ensemble des mesures techniques et non techniques de protection
- permettant à un système d'information de résister
- à des événements susceptibles de compromettre :
  - la disponibilité
  - l'intégrité
  - ou la confidentialité

Qu'est-ce que la sécurité d'un système ?

- Ensemble des mesures techniques et non techniques de protection
- permettant à un système d'information de résister
- à des événements susceptibles de compromettre :
  - la disponibilité
  - l'intégrité
  - ou la confidentialité
- des données stockées, traitées ou transmises

Qu'est-ce que la sécurité d'un système ?

- Ensemble des mesures techniques et non techniques de protection
- permettant à un système d'information de résister
- à des événements susceptibles de compromettre :
  - la disponibilité
  - l'intégrité
  - ou la confidentialité
- des données stockées, traitées ou transmises
- et des services connexes que ces systèmes offrent ou qu'ils rendent accessibles.

Quelques exemples

Scénario 1 : M.Dupond partage son ordinateur avec le reste de sa famille, mais ne veut pas que tout le monde puisse fouiller dans ses dossiers.

Quelques exemples

Scénario 1 : M.Dupond partage son ordinateur avec le reste de sa famille, mais ne veut pas que tout le monde puisse fouiller dans ses dossiers.

Scénario 2 : La PME BzhSecurity traite les données personelles de ses clients. Ces données ne doivent pas être altérées, et seules les personnes du service client peuvent y accéder.

Quelques exemples

Scénario 1 : M.Dupond partage son ordinateur avec le reste de sa famille, mais ne veut pas que tout le monde puisse fouiller dans ses dossiers.

Scénario 2 : La PME BzhSecurity traite les données personelles de ses clients. Ces données ne doivent pas être altérées, et seules les personnes du service client peuvent y accéder.

Scénario 3 : Le CHU de Rennes traitent les données personnelles de nombreux patients. L'accessibilité à ces informations sensibles est question de vie ou de mort.

Quelques exemples

Scénario 1 : M.Dupond partage son ordinateur avec le reste de sa famille, mais ne veut pas que tout le monde puisse fouiller dans ses dossiers.

Scénario 2 : La PME BzhSecurity traite les données personelles de ses clients. Ces données ne doivent pas être altérées, et seules les personnes du service client peuvent y accéder.

Scénario 3 : Le CHU de Rennes traitent les données personnelles de nombreux patients. L'accessibilité à ces informations sensibles est question de vie ou de mort.

Scénario 4 : L'entreprise Tholes a de nombreux contrat avec le ministère des armées. Elle travaille sur de nombreux projets classifiés, et toutes les données doivent être traitées avec la plus grande prudence.

Assurer la sécurité d'un système - Les vecteurs d'attaques

Les moyens qu'utilise l'attaquant pour arriver à ses fins :

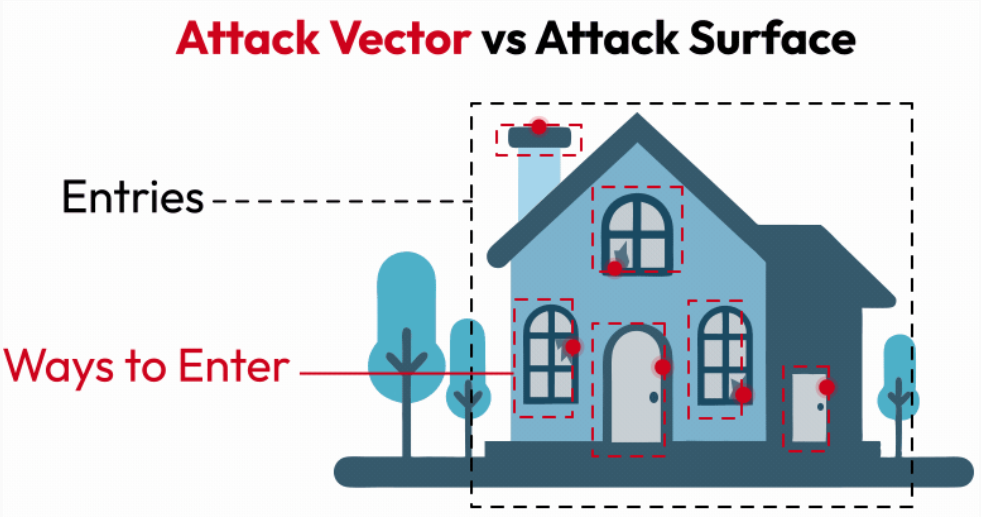
- Exploiter une vulnérabilité
- Tirer partie de la confiance de quelqu'un
- S'infiltrer dans l'entreprise
- Utiliser des clés USB frauduleuses

Assurer la sécurité d'un système - La surface d'attaque

Surface d'attaque: La surface d'attaque d'un système est l'ensemble des points en bords de système [...] par lesquels un attaquant peut tenter de s'introduire, de causer un effet, ou d'extraire des données de, ce système [...].

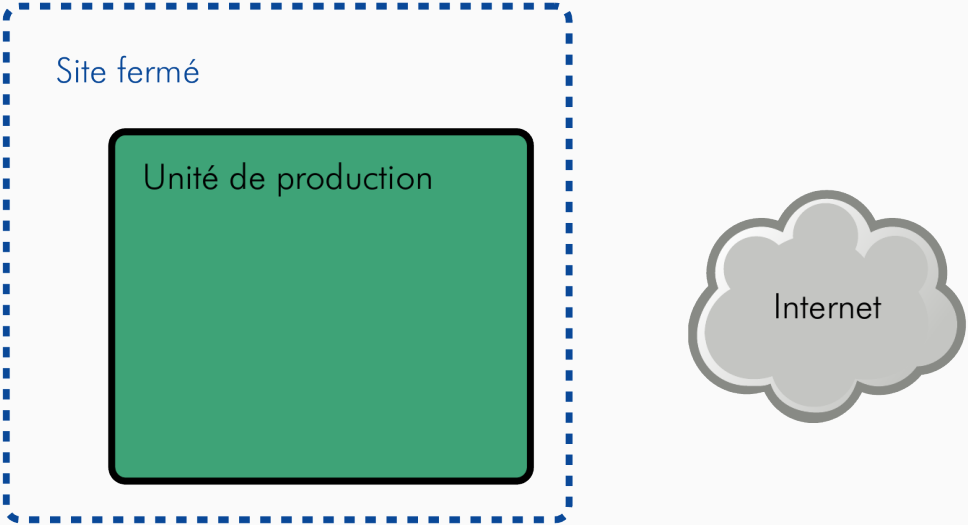
Traduit de NIST SP 800-160 Vol.2

Assurer la sécurité d'un système - La surface d'attaque

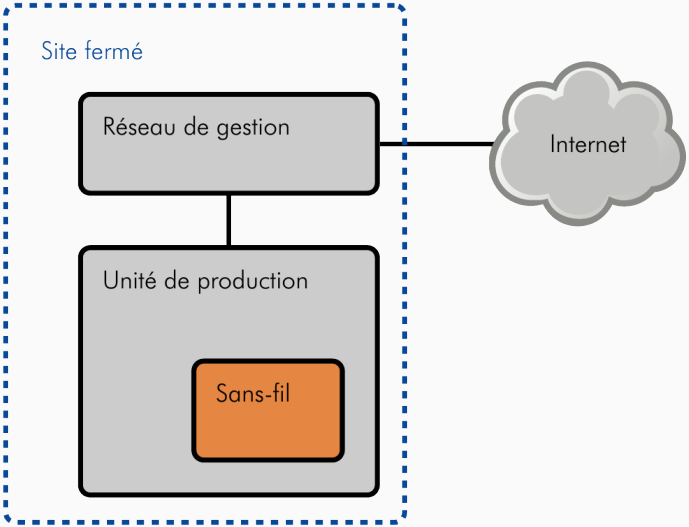


Source : cyber.gouv.fr

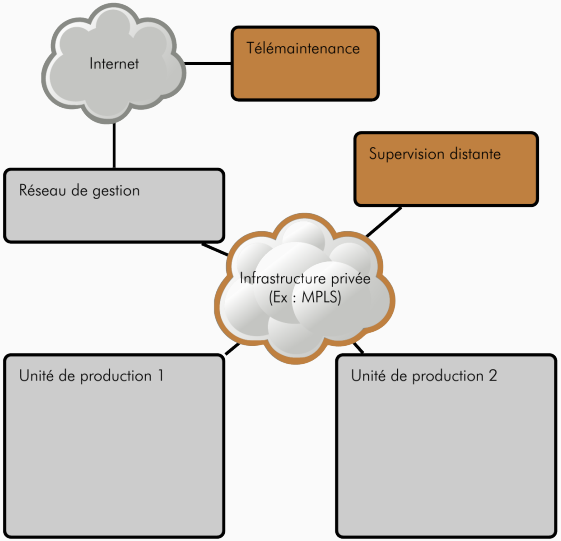
Assurer la sécurité d'un système - La surface d'attaque



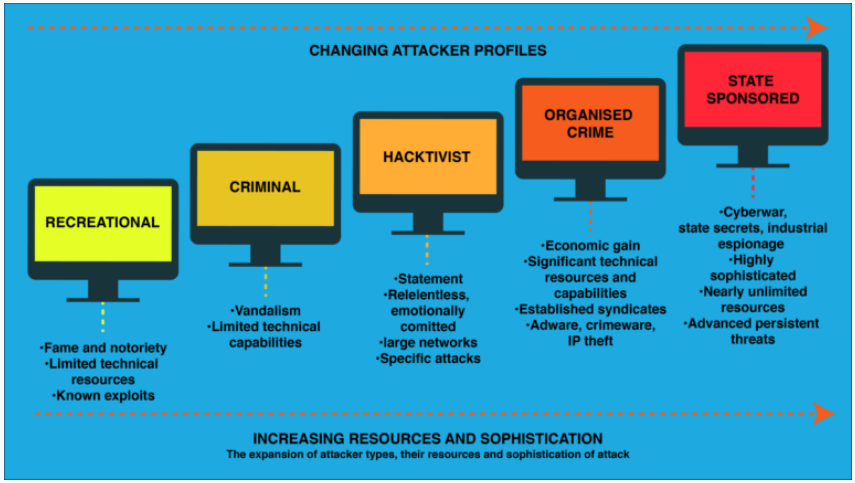
Assurer la sécurité d'un système - La surface d'attaque



Assurer la sécurité d'un système - La surface d'attaque



Assurer la sécurité d'un système - Le profil d'attaquant



Source : futurelearn.com

Panorama des menaces

Cold Cases - Captain Crunch (1971)

Résumé : John Draper utilise un sifflet pour exploiter le réseau téléphonique.

- **Contexte :** Phreaking dans les années 1970, accès gratuit aux appels longue distance.
- **Technique :** Le sifflet émettait une tonalité de 2600 Hz, identique aux signaux de contrôle des réseaux téléphoniques.
- **Impact :** De nombreux hackers influencés par cette découverte.
- **Conséquences :** Renforcement des protections télécoms et début de la culture hacker moderne.

Leçon : La sécurité ne doit pas négliger les attaques basées sur des astuces simples.

Cold Cases - Kevin Mitnick (1980-1995)

Résumé : Hacker célèbre utilisant l'ingénierie sociale pour obtenir un accès non autorisé.

- **Contexte :** Piratage de grandes entreprises technologiques, vol de logiciels propriétaires.
- **Techniques :** Manipulation d'employés, exploitation des failles télécom et informatiques.
- **Impact :** Intrusions dans les systèmes de Motorola, Sun Microsystems, etc.
- **Conséquences :** Condamnation à cinq ans de prison, durcissement des lois sur la cybercriminalité.

Leçon : L'humain est souvent le maillon faible des systèmes de sécurité.

Cold Case - Le Ver Morris (1988)

Résumé : Premier ver informatique à grande échelle qui a paralysé 10% d'Internet.

- **Contexte :** Créé par un étudiant, Robert Tappan Morris, comme un exercice.
- **Vecteur d'attaque :** Exploitait des failles UNIX.
- **Propagation :** Une machine infectée transmettait automatquement le ver.
- **Impact :** Environ 6 000 machines affectées.
- **Conséquences :** Création du CERT, première condamnation judiciaire pour crime informatique.

Leçon : Importance de la gestion des vulnérabilités et des réponses rapides aux incidents.

Cold Case - ILOVEYOU (2000)

Résumé : Un ver par e-mail utilisant l'ingénierie sociale pour se propager rapidement.

- **Contexte :** Créé par un étudiant philippin, exploitant la curiosité des utilisateurs.
- **Vecteur d'attaque :** Pièce jointe infectée en VBScript (.vbs) exécutée par l'utilisateur.
- **Propagation :** Envoi automatique du ver à tous les contacts Outlook de la victime.
- **Impact :** 50 millions d'ordinateurs infectés, entreprises et gouvernements paralysés.
- **Conséquences :** Perte estimée à 8,7 milliards de dollars, renforcement des filtres e-mails.

Leçon : L'ingénierie sociale est un vecteur d'attaque majeur difficile à contrer.

Cold Case - Stuxnet (2010)

Résumé : Première cyberattaque destructrice ciblant une infrastructure industrielle.

- **Contexte :** Malware développé par des États (probablement USA et Israël) pour saboter le programme nucléaire iranien.
- **Vecteur d'attaque :** Se propageait via des clés USB et exploitait quatre vulnérabilités Windows 0-day.
- **Effets :** Altérait la vitesse des centrifugeuses utilisées pour enrichir l'uranium, entraînant leur détérioration.
- **Impact :** 20% des centrifugeuses détruites, retardant le programme nucléaire iranien de plusieurs années.
- **Conséquences :** Révélation du potentiel des cyberattaques étatiques, impactant la cybersécurité industrielle.

Leçon : La cybersécurité des infrastructures critiques est essentielle.

Cold Case - WannaCry (2017)

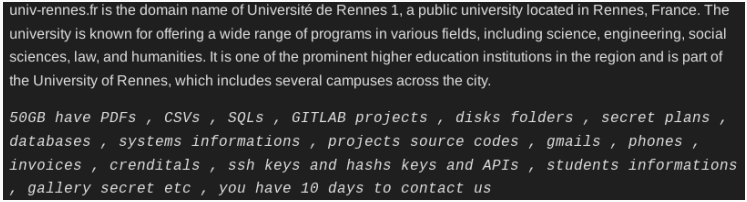
Résumé : Rançongiciel utilisant une faille NSA pour se propager automatiquement.

- **Contexte :** Exploit EternalBlue révélé par un groupe de hackers, affectant Windows.
- **Vecteur d'attaque :** Infection sans interaction utilisateur grâce à un ver.
- **Impact :** 200 000 systèmes touchés dans 150 pays, (hôpitaux et entreprises compris).
- **Conséquences :** Plus de 4 milliards de dollars de pertes économiques, prise de conscience massive sur l'importance des mises à jour.

Leçon : L'application des mises à jour de sécurité est essentielle pour éviter les attaques massives.

Hot Case - Université de Rennes (2025)

Résumé : Groupe Funksec déclare avoir volé des données après s'être introduit sur le réseau de l'université



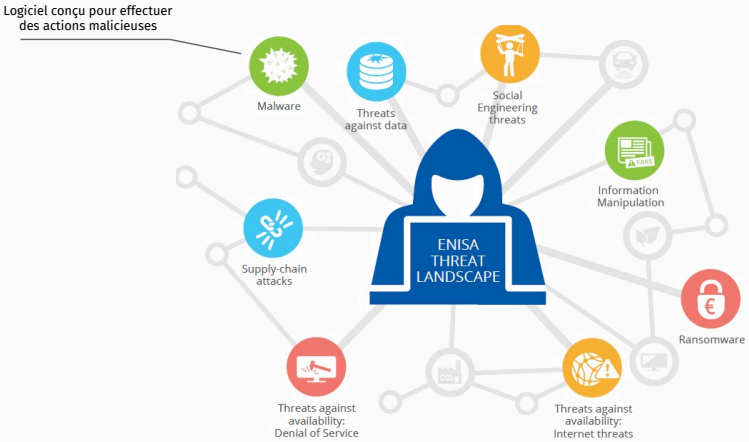
Changer vos mot de passe séasme !

Aperçu des menaces actuelles

- Rapport annuel sur l'état de la cybersécurité dans l'UE
- Analyse des tendances émergentes et des menaces principales
- Importance pour la prise de décisions en matière de cybersécurité

Nombreux vecteurs d'attaque

Nombreux vecteurs d'attaque



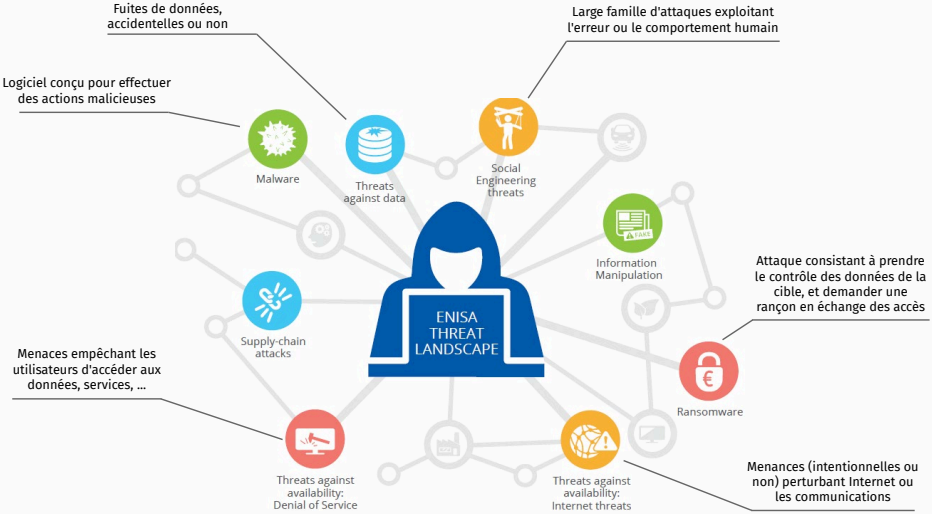
Nombreux vecteurs d'attaque

Nombreux vecteurs d'attaque

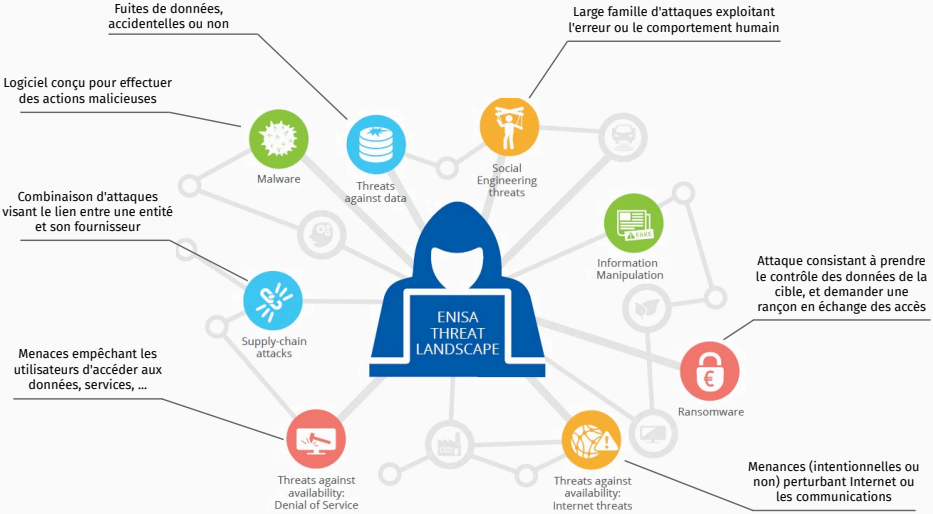
Nombreux vecteurs d'attaque

Nombreux vecteurs d'attaque

Nombreux vecteurs d'attaque



Nombreux vecteurs d'attaque



Nombreux vecteurs d'attaque

Exemple concret : Phishing

Phishing: Technique utilisée par des fraudeurs pour obtenir des renseignements personnels dans le but de perpétrer une usurpation d'identité.

Exemple concret : Phishing

> **Phishing:** Technique utilisée par des fraudeurs pour obtenir des renseignements personnels dans le but de perpétrer une usurpation d'identité.

- E-mail
- Smishing (SMS)
- Vishing (voice phishing)

En fonction de la cible

- Spear phishing
- Whaling
- Fraude au président

Phishing moderne

- Utilisation de l'IA pour créer des e-mails de phishing plus convaincants
- Deepfakes utilisés pour le clonage vocal
- AI-driven data mining pour cibler les victimes

Phishing moderne

- Utilisation de l'IA pour créer des e-mails de phishing plus convaincants
- Deepfakes utilisés pour le clonage vocal
- AI-driven data mining pour cibler les victimes

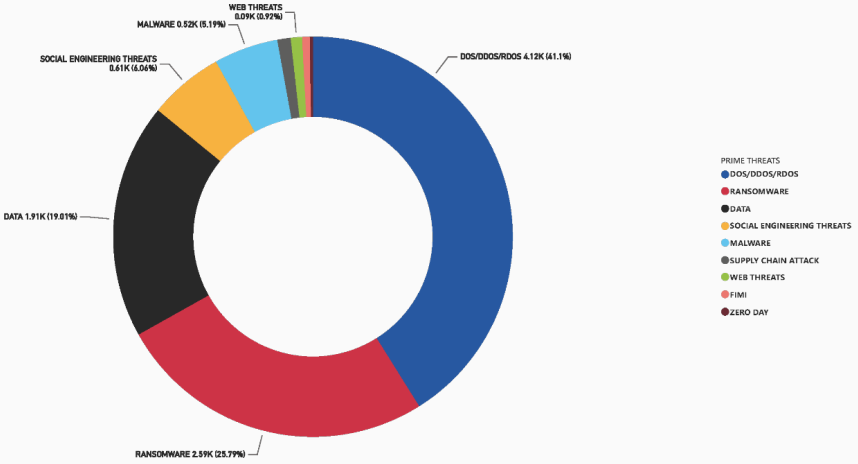
Finance worker pays out $25 million after video call with deepfake "chief financial officer"

Source : `cnn.com`

Phishing-as-a-Service (PhaaS)

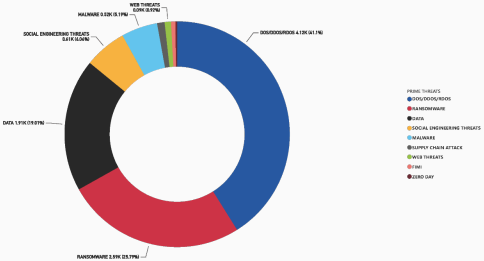
- Phishing devient un service avec des offres à partir de 15$ par jour
- Permet à des individus avec peu de connaissances en cybersécurité de lancer des attaques
- Contribue à la prolifération des attaques de phishing

Proportions des incidents - Juillet 2023 à Juillet 2024

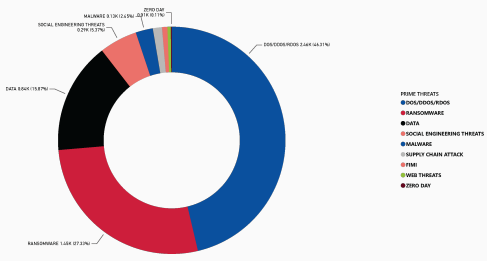


Proportions des incidents - Juillet 2023 à Juillet 2024

Incidents mondiaux



Incident européens



Mécanismes de protection

Défense en profondeur

- La sécurité d'un système se réduit à celle de son maillon le plus faible

Défense en profondeur

- La sécurité d'un système se réduit à celle de son maillon le plus faible
- Éviter de faire reposer tout la sécurité sur un seul mécanisme

Défense en profondeur

- La sécurité d'un système se réduit à celle de son maillon le plus faible
- Éviter de faire reposer tout la sécurité sur un seul mécanisme
- Défense en profondeur : On multiplie les protections de sécurité, pour se prémunir d'éventuelle défaillance.

Défense en profondeur

- La sécurité d'un système se réduit à celle de son maillon le plus faible
- Éviter de faire reposer tout la sécurité sur un seul mécanisme
- Défense en profondeur : On multiplie les protections de sécurité, pour se prémunir d'éventuelle défaillance.
- Par exemple :

Défense en profondeur

- La sécurité d'un système se réduit à celle de son maillon le plus faible
- Éviter de faire reposer tout la sécurité sur un seul mécanisme
- Défense en profondeur : On multiplie les protections de sécurité, pour se prémunir d'éventuelle défaillance.
- Par exemple :
  - Les utilisateurs s'authentifient

Défense en profondeur

- La sécurité d'un système se réduit à celle de son maillon le plus faible
- Éviter de faire reposer tout la sécurité sur un seul mécanisme
- Défense en profondeur : On multiplie les protections de sécurité, pour se prémunir d'éventuelle défaillance.
- Par exemple :
  - Les utilisateurs s'authentifient
  - Un contrôle d'accès empêche un utilisateur lambda d'accéder à toutes les ressources.

Défense en profondeur

- La sécurité d'un système se réduit à celle de son maillon le plus faible
- Éviter de faire reposer tout la sécurité sur un seul mécanisme
- Défense en profondeur : On multiplie les protections de sécurité, pour se prémunir d'éventuelle défaillance.
- Par exemple :
  - Les utilisateurs s'authentifient
  - Un contrôle d'accès empêche un utilisateur lambda d'accéder à toutes les ressources.
  - l'antivirus va détecter les menaces en amont.

Défense en profondeur

- La sécurité d'un système se réduit à celle de son maillon le plus faible
- Éviter de faire reposer tout la sécurité sur un seul mécanisme
- Défense en profondeur : On multiplie les protections de sécurité, pour se prémunir d'éventuelle défaillance.
- Par exemple :
  - Les utilisateurs s'authentifient
  - Un contrôle d'accès empêche un utilisateur lambda d'accéder à toutes les ressources.
  - l'antivirus va détecter les menaces en amont.
  - les applications exposées sont mieux isolées

Le principe du moindre privilège

"le principe de moindre privilège est un principe qui stipule, selon l'ANSSI, qu'une tâche ne doit bénéficier que de privilèges strictement nécessaires à l'exécution du code menant à bien ses fonctionnalités. En d'autres termes, une tâche ne devrait avoir la possibilité de mener à bien que les actions dont l'utilité fonctionnelle est avérée."

Le principe du moindre privilège

"le principe de moindre privilège est un principe qui stipule, selon l'ANSSI, qu'une tâche ne doit bénéficier que de privilèges strictement nécessaires à l'exécution du code menant à bien ses fonctionnalités. En d'autres termes, une tâche ne devrait avoir la possibilité de mener à bien que les actions dont l'utilité fonctionnelle est avérée."

- Un utilisateurs ne doit accéder qu'à ses données
- Un programme ne doit pouvoir accéder qu'au minimum de ressources possibles
- On évite le *tout ou rien*, pour privilégier l'attribution des capacités strictement nécessaires

De la sécurité à tous les niveaux

- Sécurité des données
- Sécurité du système
- Sécurité des réseaux
- Sécurité du matériel

De la sécurité à tous les niveaux

- **Sécurité des données**
- Sécurité du système
- Sécurité des réseaux
- Sécurité du matériel

- Se protégéer des fuites de données, ransomwares, ...
- Stocker uniquement le nécessaire
- Faire des sauvegardes régulières
- Chiffrer ses données au repos
- ...

De la sécurité à tous les niveaux

- Sécurité des données
- **Sécurité du système**
- Sécurité des réseaux
- Sécurité du matériel

- Durcicement des systèmes
- Application de politiques strictes
- Mises à jour de sécurité régulières
- Confinement d'application
- Antivirus
- ...

De la sécurité à tous les niveaux

- Sécurité des données
- Sécurité du système
- **Sécurité des réseaux**
- Sécurité du matériel

- Protection des données en transit
- Utilisation de protocoles sécurisées
- Éventuellement utilisation de VPN
- Cloisonnement réseau
- Règles de parfeu adaptées

De la sécurité à tous les niveaux

- Sécurité des données
- Sécurité du système
- Sécurité des réseaux
- **Sécurité du matériel**

- Contrôle des accès aux matériels
- Équipements souvent peu sécurisés (IoT)

Qui sait ce qu'est un OS?

Sécurité du système - Une vue haut niveaux

- Comment authentifier un utilisateur ?

Sécurité du système - Une vue haut niveaux

- Comment authentifier un utilisateur ?
- Comment s'assurer que n'importe qui n'a pas accès à n'importe quoi ?

Sécurité du système - Une vue haut niveaux

- Comment authentifier un utilisateur ?
- Comment s'assurer que n'importe qui n'a pas accès à n'importe quoi ?
- Comment s'assurer que les programmes ne puisse pas executer tout ce qu'ils veulent ?

Sécurité du système - Une vue haut niveaux

- Comment authentifier un utilisateur ?
- Comment s'assurer que n'importe qui n'a pas accès à n'importe quoi ?
- Comment s'assurer que les programmes ne puisse pas executer tout ce qu'ils veulent ?
- Comment isoler les processus les uns des autres ?

Sécurité du système - Une vue haut niveaux

- Comment authentifier un utilisateur ?
- Comment s'assurer que n'importe qui n'a pas accès à n'importe quoi ?
- Comment s'assurer que les programmes ne puisse pas executer tout ce qu'ils veulent ?
- Comment isoler les processus les uns des autres ?
- Comment protéger les données ?

L’authentification

Qu'est-ce que l'authentification ?

- Prouver l'identité d'une entité (utilisateur, machine, logiciel, ...)
- Lier une identité à une entité
- Peut se faire localement ou à distance
- 2 étapes:
  - Inscription : définit la liaison de l'identité dans le système
  - Connexion : vérifie et lie l'identité de l'entité

Identification $\neq$ Authentication $\neq$ Autorisation

Le concept

- L'authentificateur (*ex.* serveur, site web) vous demande de prouver que vous êtes bien celui que vous prétendez être, en vous basant sur une ou plusieurs preuves appelées facteurs.
  - L'authentification mutuelle est également possible
- Différents types de facteurs
  - Ce que vous savez (mot de passe),
  - Ce que vous possédez (téléphone, carte à puce, jeton),
  - Ce que vous êtes (données biométriques),
  - ...

Authentification Multi-Facteurs

Habituellement appelé MFA (Multi-Factor Authentication)

Idée : Enchaîner plusieurs facteurs pour renforcer la sécurité.

Authentification Forte

- Au moins 2 facteurs (2FA)...
- ...qui ne doivent pas être facilement contournés ou compromis

Ce que vous savez - Authentification par mot de passe

Exemples

- Noms d'utilisateur et mots de passe pour des comptes en ligne.
- Codes PIN pour les transactions aux distributeurs automatiques.

Avantages

- Simple et facile à utiliser.
- Coût de mise en œuvre faible.
- Largement compris par les utilisateurs.

Inconvénients

- Vulnérable aux attaques de phishing, par force brute et par dictionnaire.
- Les utilisateurs choisissent souvent des mots de passe faibles.
- Les mots de passe peuvent être oubliés ou volés.

Authentification par mot de passe - Pièges et recommandations

Pièges

- Reposer uniquement sur les mots de passe sans mesures de sécurité supplémentaires.
- Ne pas appliquer de politiques de mots de passe robustes.
- Mal stocker les mots de passe.

Recommandations

- Utiliser plusieurs facteurs d'authentifiaction.
- Appliquer des politiques de mots de passe robustes, incluant des exigences de complexité.
- Stocker les mots de passe en utilisant une bonne fonction de hachage et du sel.
- Prendre en compte votre modèle d'attaque (en ligne vs hors ligne)

Authentification par mot de passe - Stockage des mots de passe

- Un stockage inapproprié conduit à des fuites
- Les méthodes de stockage courantes incluent :

Authentification par mot de passe - Stockage des mots de passe

- Un stockage inapproprié conduit à des fuites
- Les méthodes de stockage courantes incluent :
  - Clair : **mauvais !**

Authentification par mot de passe - Stockage des mots de passe

- Un stockage inapproprié conduit à des fuites
- Les méthodes de stockage courantes incluent :
  - Clair : **mauvais !**
  - Chiffré : peu courant et sujet aux erreurs

Authentification par mot de passe - Stockage des mots de passe

- Un stockage inapproprié conduit à des fuites
- Les méthodes de stockage courantes incluent :
  - Clair : **mauvais !**
  - Chiffré : peu courant et sujet aux erreurs
  - Haché : un peu mieux, mais toujours insécurisé

Authentification par mot de passe - Stockage des mots de passe

- Un stockage inapproprié conduit à des fuites
- Les méthodes de stockage courantes incluent :
  - Clair : **mauvais !**
  - Chiffré : peu courant et sujet aux erreurs
  - Haché : un peu mieux, mais toujours insécurisé
  - Haché et salé : mieux, mais attention aux paramètres

Authentification par mot de passe - Stockage des mots de passe

- Un stockage inapproprié conduit à des fuites
- Les méthodes de stockage courantes incluent :
  - Clair : **mauvais !**
  - Chiffré : peu courant et sujet aux erreurs
  - Haché : un peu mieux, mais toujours insécurisé
  - Haché et salé : mieux, mais attention aux paramètres
  - Utiliser une fonction de dérivation de clé (KDF) : mieux vaut utiliser une fonction résistante à l'attaque par mémoire

Authentification par mot de passe - Fonctions de hachage

Hash functions		KDF		Memory hard KDF	
MD5	164.1 GH/s	PBKDF2-HMAC-MD5 (1000 iterations)	46 170 kH/s	bcrypt	184 kH/s
SHA1	50 638.7 MH/s	PBKDF2-HMAC-SHA1 (1000 iterations)	19 124.4 kH/s	scrypt (16384 iterations)	7 126 H/s
SHA2-256	51 975.5 MH/s	PBKDF2-HMAC-SHA2-256 (1000 iterations)	8 865.7 kH/s	argon2id	TBD
SHA2-512	7483.4 MH/s	PBKDF2-HMAC-SHA2-512 (1000 iterations)	3 120.9 kH/s		
SHA3-256	5058.7 MH/s				

Table 1: Benchmark des fonctions de hachage avec hashcat sur un GPU NVIDIA GeForce RTX 4090 [2]

[1] https://gist.github.com/Chick3nman/32e662a5bb63bc4f51b847bb422222fd

[2] https://gist.github.com/Chick3nman/32e662a5bb63bc4f51b847bb422222fd

Authentification par mot de passe - Attaques courantes

- Attaques par force brute : peuvent être améliorées par des méthodes probabilistes

Authentification par mot de passe - Attaques courantes

- Attaques par force brute : peuvent être améliorées par des méthodes probabilistes
- Attaques par dictionnaire : utilisent un ensemble de candidats comme base
  - Peut modifier chaque candidat pour tester davantage de possibilités
  - Les dictionnaires sont généralement construits à partir de bases de données fuitées

Authentification par mot de passe - Attaques courantes

- Attaques par force brute : peuvent être améliorées par des méthodes probabilistes
- Attaques par dictionnaire : utilisent un ensemble de candidats comme base
  - Peut modifier chaque candidat pour tester davantage de possibilités
  - Les dictionnaires sont généralement construits à partir de bases de données fuitées
- Compromis Temps-Mémoire (TMTO)
  - Stocker tous les hachages requiert trop de mémoire
  - Le calcul à la volée est trop long
  - Rainbow Tables : pré-calculent beaucoup, mais ne stockent pas toutes les informations (certaines peuvent être récupérées par calcul)
  - Les tables peuvent être construites une fois et réutilisées

Ce que vous possédez - Authentification par *token*

Exemples

- Token matériels (par exemple, RSA SecurID, YubiKey, Google Titan, clés FIDO2...)
- Token logiciels (par exemple, Google Authenticator)
- Cartes à puce

Avantages

- Une sécurité supérieure à celle des mots de passe seuls.
- Réduit le risque d'attaques de phishing.

Inconvénients

- Peuvent être perdus, volés ou endommagés.
- Coût supplémentaire pour les jetons matériels.
- Nécessite que l'utilisateur porte un dispositif supplémentaire.

Authentification par jeton - Pièges et recommandations

Pièges

- Distribution et gestion des jetons non sécurisées correctement.
- Se reposer uniquement sur les jetons sans mesures de sécurité supplémentaires.
- Mécanismes de sauvegarde/révocation inadéquats pour les jetons perdus ou volés.

Recommandations

- Mettre en place des procédures robustes pour l'émission et la gestion des jetons.
- Combiner l'authentification par jeton avec d'autres méthodes (*ex.*, mots de passe) pour l'AMF.
- Fournir aux utilisateurs un moyen de récupérer en toute sécurité en cas de perte ou de vol de jetons.

Ce que vous êtes - Authentification biométrique

Exemples

- Reconnaissance d'empreintes digitales
- Reconnaissance faciale
- Scan de la rétine
- Reconnaissance vocale

Avantages

- Difficile à falsifier ou à voler.
- Pratique pour les utilisateurs (pas besoin de mémoriser des mots de passe ou de porter des tokens).
- Peut une forte sécurité.

Inconvénients

- Problèmes de confidentialité (les données biométriques sont sensibles).
- Risque de faux positifs/négatifs.
- Coût d'implémentation élevé.

Authentification biométrique - Pièges et recommandations

Pièges

- Stocker des données biométriques sans chiffrement approprié.
- Ne pas prendre en compte le consentement de l'utilisateur et les réglementations sur la vie privée.
- Dépendance excessive à un seul trait biométrique.

Recommandations

- Chiffrer les données biométriques et les stocker de manière sécurisée.
- Veiller à la conformité avec les lois sur la vie privée et obtenir le consentement de l'utilisateur.
- Utiliser plusieurs traits biométriques pour améliorer la précision et la sécurité.

Le contrôle d'accès

Introduction au contrôle d'accès

- Le contrôle d'accès détermine qui peut accéder à quelles ressources et quelles opérations ils peuvent effectuer.
- Composants clés :
  - **Sujets** : Entités demandant l'accès (par exemple, utilisateurs, processus)

Introduction au contrôle d'accès

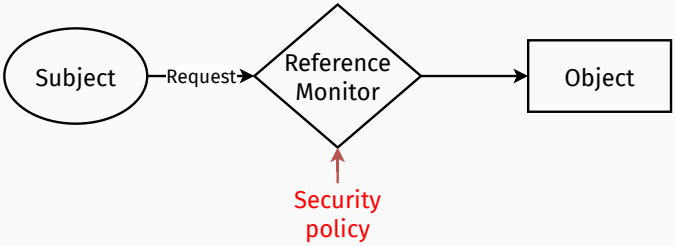
- Le contrôle d'accès détermine qui peut accéder à quelles ressources et quelles opérations ils peuvent effectuer.
- Composants clés :
  - **Sujets** : Entités demandant l'accès (par exemple, utilisateurs, processus)
  - **Objets** : Ressources auxquelles on accède (par exemple, fichiers, bases de données, processus)

Introduction au contrôle d'accès

- Le contrôle d'accès détermine qui peut accéder à quelles ressources et quelles opérations ils peuvent effectuer.
- Composants clés :
  - **Sujets** : Entités demandant l'accès (par exemple, utilisateurs, processus)
  - **Objets** : Ressources auxquelles on accède (par exemple, fichiers, bases de données, processus)
  - **Moniteur de référence** : Le mécanisme d'application qui traite les requêtes

Introduction au contrôle d'accès

- Le contrôle d'accès détermine qui peut accéder à quelles ressources et quelles opérations ils peuvent effectuer.
- Composants clés :
  - **Sujets** : Entités demandant l'accès (par exemple, utilisateurs, processus)
  - **Objets** : Ressources auxquelles on accède (par exemple, fichiers, bases de données, processus)
  - **Moniteur de référence** : Le mécanisme d'application qui traite les requêtes



Access Control Matrix[1]

Objets / Sujets	Fichier A	Fichier B	Process A	...
Alice	r1,r2,r3	r1,r2	r1	...
Bob	r1	r1,r2	-	...
Charlie	-	r1,r3	r2	...
Process 1	r2	r1,r3	r2	...

Permet différentes approches :

[1] *Lampson, 1971*

Access Control Matrix[1]

Objets / Sujets	**Fichier A**	Fichier B	Process A	...
Alice	**r1,r2,r3**	r1,r2	r1	...
Bob	**r1**	r1,r2	-	...
Charlie	**-**	r1,r3	r2	...
Process 1	**r2**	r1,r3	r2	...

Permet différentes approches :

- Liste de contrôle d'accès : qui peut accéder à un objet donné

[1] *Lampson, 1971*

Access Control Matrix[1]

Sujets \ Objets	Fichier A	Fichier B	Process A	...
Alice	**r1,r2,r3**	**r1,r2**	**r1**	**...**
Bob	r1	r1,r2	-	...
Charlie	-	r1,r3	r2	...
Process 1	r2	r1,r3	r2	...

Permet différentes approches :

- Liste de contrôle d'accès : qui peut accéder à un objet donné
- Capacité : définir des paires (objet, requête) autorisées pour un sujet donné

[1] *Lampson, 1971*

Liste de contrôle d'accès

Plus connu sous le nom de *Access Control List (ACL)*

- Une liste attachée à chaque objet spécifiant quels sujets peuvent y accéder et leurs droits.
- **Caractéristiques**:
  - Centrée sur l'objet.
  - Simplifie la gestion des accès pour des ressources spécifiques.
  - La révocation peut être compliquée.

Vue d'ensemble des modèles de contrôle d'accès

- **Contrôle d'accès discrétionnaire (DAC)** : Permissions contrôlées par l'utilisateur.
- **Contrôle d'accès obligatoire (MAC)** : Niveaux de sécurité imposés par le système.
- **Contrôle d'accès basé sur les rôles (RBAC)** : Permissions basées sur les rôles organisationnels.
- **Autres modèles**:
  - **Contrôle d'accès basé sur les attributs (ABAC)** : Basé sur des attributs.
  - **Contrôle d'accès basé sur les capacités** : Basé sur les capacités associées aux sujets.

Contrôle d'accès discrétionnaire (DAC)

- Les droits d'accès sont déterminés par le propriétaire de la ressource.
- **Caractéristiques**:
  - Flexible, facile à mettre en œuvre
  - Susceptible aux erreurs de configuration par les utilisateurs
  - Peut être difficile d'imposer une politique commune sur des systèmes complexes

Contrôle d'accès discrétionnaire (DAC)

- Les droits d'accès sont déterminés par le propriétaire de la ressource.
- **Caractéristiques**:
  - Flexible, facile à mettre en œuvre
  - Susceptible aux erreurs de configuration par les utilisateurs
  - Peut être difficile d'imposer une politique commune sur des systèmes complexes
- **Usage courant** : Systèmes de fichiers comme UNIX et Windows.
  - Les ACL sont stockées avec les objets...
  - ...donc les systèmes de fichiers doivent le supporter

Contrôle d'accès discrétionnaire (DAC)

- Les droits d'accès sont déterminés par le propriétaire de la ressource.
- **Caractéristiques**:
  - Flexible, facile à mettre en œuvre
  - Susceptible aux erreurs de configuration par les utilisateurs
  - Peut être difficile d'imposer une politique commune sur des systèmes complexes
- **Usage courant** : Systèmes de fichiers comme UNIX et Windows.
  - Les ACL sont stockées avec les objets...
  - ...donc les systèmes de fichiers doivent le supporter

Exemple : Alice créé un fichier et décide qui peut le lire, le modifier et l'exécuter.

Contrôle d'accès basé sur les rôles (RBAC)

- Les droits d'accès sont attribués en fonction des rôles au sein d'une organisation.
- **Caractéristiques**:
  - Simplifie la gestion dans les grandes organisations.
  - Les rôles sont alignés avec les fonctions professionnelles.

Contrôle d'accès basé sur les rôles (RBAC)

- Les droits d'accès sont attribués en fonction des rôles au sein d'une organisation.
- **Caractéristiques**:
  - Simplifie la gestion dans les grandes organisations.
  - Les rôles sont alignés avec les fonctions professionnelles.
- **Cas d'utilisation** : Courant dans les environnements d'entreprise où les rôles sont clairement définis.

Contrôle d'accès basé sur les rôles (RBAC)

- Les droits d'accès sont attribués en fonction des rôles au sein d'une organisation.
- **Caractéristiques**:
  - Simplifie la gestion dans les grandes organisations.
  - Les rôles sont alignés avec les fonctions professionnelles.
- **Cas d'utilisation** : Courant dans les environnements d'entreprise où les rôles sont clairement définis.

> **Exemple :** Un rôle de « Manager » peut approuver des budgets, un rôle d'« Employé » peut soumettre des rapports.

Contrôle d'accès obligatoire (MAC) - Concept

- Gestion des droits par une authorité centrale, et non par le propriétaire de la ressource. Ils sont définis pour les sujets et les objets.
- **Caractéristiques**:
  - Plus sécurisé, moins flexible.
  - Les sujets et les objets possèdent des étiquettes.
- **Éléments** : Chaque sujet et objet a à la fois un niveau de sécurité et une catégorie

Contrôle d'accès obligatoire (MAC) - Concept

- Gestion des droits par une authorité centrale, et non par le propriétaire de la ressource. Ils sont définis pour les sujets et les objets.
- **Caractéristiques**:
  - Plus sécurisé, moins flexible.
  - Les sujets et les objets possèdent des étiquettes.
- **Éléments** : Chaque sujet et objet a à la fois un niveau de sécurité et une catégorie
- **Cas d'utilisation** : Courant dans les contextes gouvernementaux/militaires pour les habilitations (Top Secret, Secret, Confidentiel, Public).

Contrôle d'accès obligatoire (MAC) - Concept

- Gestion des droits par une authorité centrale, et non par le propriétaire de la ressource. Ils sont définis pour les sujets et les objets.
- **Caractéristiques**:
  - Plus sécurisé, moins flexible.
  - Les sujets et les objets possèdent des étiquettes.
- **Éléments** : Chaque sujet et objet a à la fois un niveau de sécurité et une catégorie
- **Cas d'utilisation** : Courant dans les contextes gouvernementaux/militaires pour les habilitations (Top Secret, Secret, Confidentiel, Public).

> **Exemple :** Seuls les sujets disposant d'un certain niveau d'habilitation peuvent accéder aux données Top Secret.

Contrôle d'accès en pratique

- **Linux**:
  - Utilisation du DAC via les permissions UNIX.
  - Implémentation du MAC via SELinux pour appliquer des politiques qui restreignent les actions des utilisateurs et des processus au-delà des permissions UNIX standard.
- **Windows**:
  - Utilisation du DAC via les permissions NTFS.
  - MAC et niveaux d'intégrité via Windows Integrity Control.
- **Défis**:
  - Équilibrer la sécurité et l'utilisabilité.
  - Gérer les permissions dans des environnements vastes et dynamiques.

Permission de fichiers UNIX - Mécansimes de bases

- Permissions assignées à l'**u**tilisateur propriétaire, au **g**group propriétaire, et aux autres (**o**thers)
- Elles sont représentées avec une combinaison de lettres (r, w, x), ou de chiffre (en octal).
- **Read (r=4)**
  - Fichier : Lire le contenu du fichier
  - Dossier : Lister les fichiers du répertoire
- **Write (w=2)**
  - Fichier : Modifier ou supprimer le contenu du fichier
  - Dossier : Créer au supprimer des fichiers
- **Execute (x=1)**
  - Fichier : Exécuter un fichier
  - Dossier : Ouvrir des fichier, et rentrer dans le dossier

Informations sur les fichiers

```
$ ls -l file.txt
-rwxr-xr-- 1 Bob Bob 3215 Sept 01 10:21 file.txt
```

Informations sur les fichiers

```
$ ls -l file.txt
-rwxr-xr-- 1 Bob Bob 3215 Sept 01 10:21 file.txt
```

Types de fichiers

- `-` fichier
- `d` dossier
- `b` périphérique bloc
- `s` socket
- `l` lien symbolique
- `p` FIFO pipe
- `c` fichier I/O

Permissions de fichiers

- 3 blocs, 3 permissions par bloc
- User, Group, Other

Informations sur les fichiers

```
$ ls -l file.txt
-rwxr-xr-- 1 Bob Bob 3215 Sept 01 10:21 file.txt
```

- Seul le propriétaire et root peuvent changer les permissions
- Seul root peut changer le propriétaire

Représentation des permissions

- Permissions représentées avec une combinaison de lettres (r, w, x), ou de chiffre (en octal).
- Utilisez `chmod` pour modifier les permissions de base :

  `chmod 754 filename`
  - 7 (`rwx`) pour le propriétaire.
  - 5 (`r-x`) pour le groupe.
  - 4 (`r--`) pour les autres.

rwx	r-x	r--
111	101	100
7	5	4