

SSL/TLS

Réalisé par: *Hamza AIT LAMAALAM*
Nadir ALMELLOUKI
Mohammed ABIDOU
Saad ADDAR
Omar ANOUAR





CONTENU

- 01 Introduction**
- 02 Fonctionnement et Certificats SSL/TLS**
- 03 Types de Certificats et Utilisation de TLS**
- 04 Vulnérabilités et Attaques liées à SSL/TLS**
- 05 Configuration et Bonnes Pratiques**
- 06 Conclusion**

INTRODUCTION



QU'EST-CE QUE SSL ? (SECURE SOCKETS LAYER)

SSL, ou Secure Sockets Layer, est un protocole cryptographique. Il assure une communication sécurisée sur un réseau. Son fonctionnement de base inclut le chiffrement des données. Il permet aussi l'authentification du serveur. Les principaux objectifs de SSL sont la confidentialité, l'intégrité et l'authentification des données échangées.

Confidentialité

Assurer que seules les parties autorisées peuvent lire les données.

Intégrité

Garantir que les données ne sont pas modifiées en transit.

Authentification

Vérifier l'identité du serveur avec lequel vous communiquez.

Importance de Protocole SSL

T

a. Protection des Données Sensibles

- Chiffrement des données : SSL/TLS crypte les informations échangées entre un client (navigateur) et un serveur (site web), empêchant les interceptions par des pirates.
- Sécurité des transactions : Essentiel pour les paiements en ligne (e-commerce, banques), les connexions sécurisées (emails, VPN), etc.

Importance de Protocole SSL



b. Authentification des Sites Web

- Certificats SSL : Vérifient l'identité du site, évitant les attaques de type phishing (sites frauduleux).
- Confiance utilisateur : Les sites avec HTTPS (et le cadenas vert) inspirent confiance aux visiteurs.

c. Conformité aux Normes

- Exigences légales (RGPD, PCI DSS) imposent le chiffrement des données pour protéger la vie privée.
- Amélioration du référencement (SEO) : Google favorise les sites HTTPS dans ses résultats.

Évolution des Protocoles SSL

T

SSL 1.0 (1994) - Le Prototype Inachevé

- **Contexte** : Développé par Netscape pour sécuriser les communications web naissantes
- **Caractéristiques techniques** :
 - Chiffrement basé sur RC4 avec clés 40 bits
 - Mécanisme rudimentaire d'échange de clés
- **Problèmes majeurs** :
 - Vulnérabilité aux attaques par brute-force
 - Absence de vérification d'intégrité des messages
- **Sort** : Jamais publié officiellement en raison de failles fondamentales

Évolution des Protocoles SSL

T

SSL 2.0 (1995) - La Première Implémentation

- **Innovations :**
 - Introduction des certificats X.509
 - Support des algorithmes MD5 et SHA-1
- **Faiblesses critiques :**
 - Vulnérable aux attaques "Man-in-the-Middle"
 - Pas de protection contre les attaques par repli (downgrade attacks)
 - Longueur fixe des clés (40 bits) facilement cassable
- **Dépréciation :**
 - Interdit par RFC 6176 en 2011
 - Support abandonné par tous les navigateurs majeurs

Évolution des Protocoles SSL

T

SSL 3.0 (1996) - Le Dernier de la Lignée

- **Améliorations notables :**
 - Introduction du chiffrement CBC (Cipher Block Chaining)
 - Mécanisme de re-négociation plus sécurisé
 - Support des clés jusqu'à 256 bits
- **Échec final :**
 - Découverte de la vulnérabilité POODLE (2014)
 - Faiblesse dans l'implémentation du padding CBC
 - Déprécié par RFC 7568 en 2015

Évolution des Protocoles SSL

Analyse Comparative des Versions SSL

Version	Année	Force de Chiffrement	Algorithmes	Principales Failles	Statut Actuel
SSL 1.0	1994	40 bits	RC4	Conception défectueuse	Jamais publié
SSL 2.0	1995	40-128 bits	MD5, RC4	MITM, Downgrade	Interdit depuis 2011
SSL 3.0	1996	jusqu'à 256 bits	SHA-1, CBC	POODLE	Déprécié depuis 2015

QU'EST-CE QUE TLS? (TRANSPORT LAYER SECURITY)

TLS (Transport Layer Security) est un protocole cryptographique. Il sécurise les communications entre des applications sur un réseau. Son fonctionnement repose sur trois mécanismes clés : le chiffrement des données, la vérification de leur intégrité et l'authentification des parties communicantes. Les objectifs principaux de TLS sont :

Confidentialité

Protection des données contre les interceptions.

Intégrité

Garantie que les données ne sont pas altérées.

Authentification

Vérification de l'identité du serveur (et optionnellement du client)

Importance de Protocole TLS



Les protocoles TLS (Transport Layer Security) jouent un rôle essentiel dans la sécurité des communications numériques. Leur adoption est cruciale pour :

- ✓ Protéger les données sensibles (mots de passe, informations bancaires données personnelles)
- ✓ Garantir la confidentialité en chiffrant les échanges pour éviter les interceptions
- ✓ Authentifier les serveurs (et parfois les clients) pour lutter contre le phishing
- ✓ Respecter les réglementations (RGPD, PCI DSS, HIPAA) imposant le chiffrement des données
- ✓ Améliorer la confiance des utilisateurs (via l'affichage du cadenas  dans les navigateurs)

Évolution vers TLS

T

TLS 1.0 (1999 - RFC 2246) - La Succession de SSL

- **Innovations clés :**
 - Remplacement de l'algorithme MD5 par HMAC
 - Implémentation plus sécurisée du chiffrement CBC
 - Support amélioré des suites cryptographiques
- **Problèmes persistants :**
 - Vulnérabilité aux attaques BEAST (2011)
 - Faiblesses dans l'implémentation initiale de CBC
 - Déprécié par RFC 7568 en 2015
- **Statut actuel :** Déprécié en 2021 (RFC 8996)

Évolution vers TLS



TLS 1.1 (2006 - RFC 4346) - Premières Corrections Majeures

Améliorations notables :

- Protection contre les attaques par padding CBC
- Support explicite des vecteurs d'initialisation (IV)
- Meilleure gestion des erreurs cryptographiques
- **Limitations :**
 - Maintenance des algorithmes considérés comme faibles (RC4)
 - Complexité dans la re-négociation
- **Statut actuel :** Déprécié en 2021 (RFC 8996)

Évolution vers TLS

TLS 1.2 (2008 - RFC 5246) - La Maturité

Avancées majeures :

- Support des suites cryptographiques modernes(AES-GCM, SHA-256)
- Flexibilité accrue dans le choix des algorithmes
- Meilleure résistance aux attaques par timing

• Adoption :

- Devenu standard de facto pendant plus d'une décennie
- Support universel par tous les navigateurs modernes

• Statut actuel : Toujours largement utilisé mais en voie de remplacement

Évolution vers TLS

T

TLS 1.3 (2018 - RFC 8446) - La Révolution

- **Innovations radicales :**
 - Suppression des algorithmes obsolètes (RC4, DES, MD5)
 - Handshake simplifié (1-RTT et 0-RTT modes)
 - Forward secrecy obligatoire
 - Séparation claire des négociations cryptographiques
- **Adoption :**
 - Réduction de la latence
 - Meilleure résistance aux attaques par downgrade
 - Suppression des fonctionnalités dangereuses

Évolution vers TLS

T

Comparatif des Versions TLS

Version	Année	Chiffrements Modernes	Forward Secrecy	Handshake	Sécurité	Statut
TLS 1.0	1999	Limité	Optionnel	Complexé	Faible	Déprécié
TLS 1.1	2006	Partiel	Optionnel	Complexé	Moyenne	Déprécié
TLS 1.2	2008	Complet	Optionnel	Standard	Bonne	Actif (en déclin)
TLS 1.3	2018	Exclusif	Obligatoire	Optimisé	Excellente	Recommandé

Pourquoi SSL est Obsolète ?

1- Vulnérabilités Critiques Inhérentes

a) Failles Cryptographiques Majeures

- Algorithmes faibles : Utilisation de MD5 (cassé depuis 2004) et RC4 (brisé en 2015)
- Longueur de clé insuffisante : 40 bits (SSL 2.0) et 56 bits (DES) sont trivialement cassables
- Attaques POODLE (SSL 3.0) : Exploitation des vulnérabilités du padding CBC

b) Attaques Spécifiques

- BEAST (TLS 1.0 mais lié à l'héritage SSL)
- CRIME (Compression Ratio Info-leak Made Easy)
- DROWN (Attaque cross-protocole exploitant SSL 2.0)

Pourquoi SSL est Obsolète ?

2- Dépréciation Officielle

a) Standards de Sécurité

- PCI DSS : Interdit SSL depuis 2018 (v3.2)
- NIST : Recommande la désactivation depuis 2014 (SP 800-52 Rev. 2)
- RFC 7568 (2015) : "Dépréciation explicite de SSL 3.0"

b) Support Navigateurs

- Chrome/Firefox : Désactivé par défaut depuis 2014
- Edge/Safari : Suppression complète dans les versions récentes

Pourquoi SSL est Obsolète ?

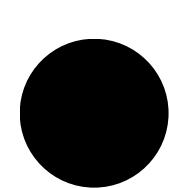
3- Comparaison Technique SSL vs TLS 1.3

Caractéristique	SSL 3.0 (1996)	TLS 1.3 (2018)
Algorithmes	MD5, RC4, DES	AES-GCM, ChaCha20
Longueur clé	Jusqu'à 256 bits	Minimum 256 bits
Forward Secrecy	Absent	Obligatoire
Handshake	2-RTT (lent)	1-RTT/0-RTT (rapide)
Vulnérabilités connues	12+ (critiques)	0 (à ce jour)

FONCTIONNEMENT ET CERTIFICATS SSL/TLS

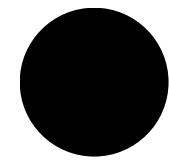


ÉTABLISSEMENT D'UNE CONNEXION SÉCURISÉE (HANDSHAKE TLS)



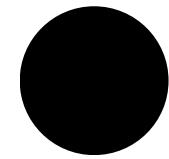
ClientHello

Le client envoie les versions TLS supportées, les suites de chiffrement disponibles et un nombre aléatoire



ServerHello

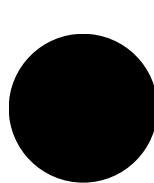
Le serveur répond avec la version TLS et la suite de chiffrement choisies, son propre nombre aléatoire et un identifiant de session



Certificat du serveur

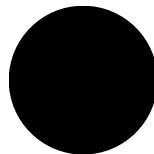
Le serveur envoie son certificat numérique contenant sa clé publique

ÉTABLISSEMENT D'UNE CONNEXION SÉCURISÉE (HANDSHAKE TLS)



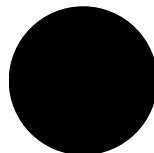
Échange de clés

Le client génère un "secret prémaître" et l'envoie au serveur (chiffré avec la clé publique du serveur)



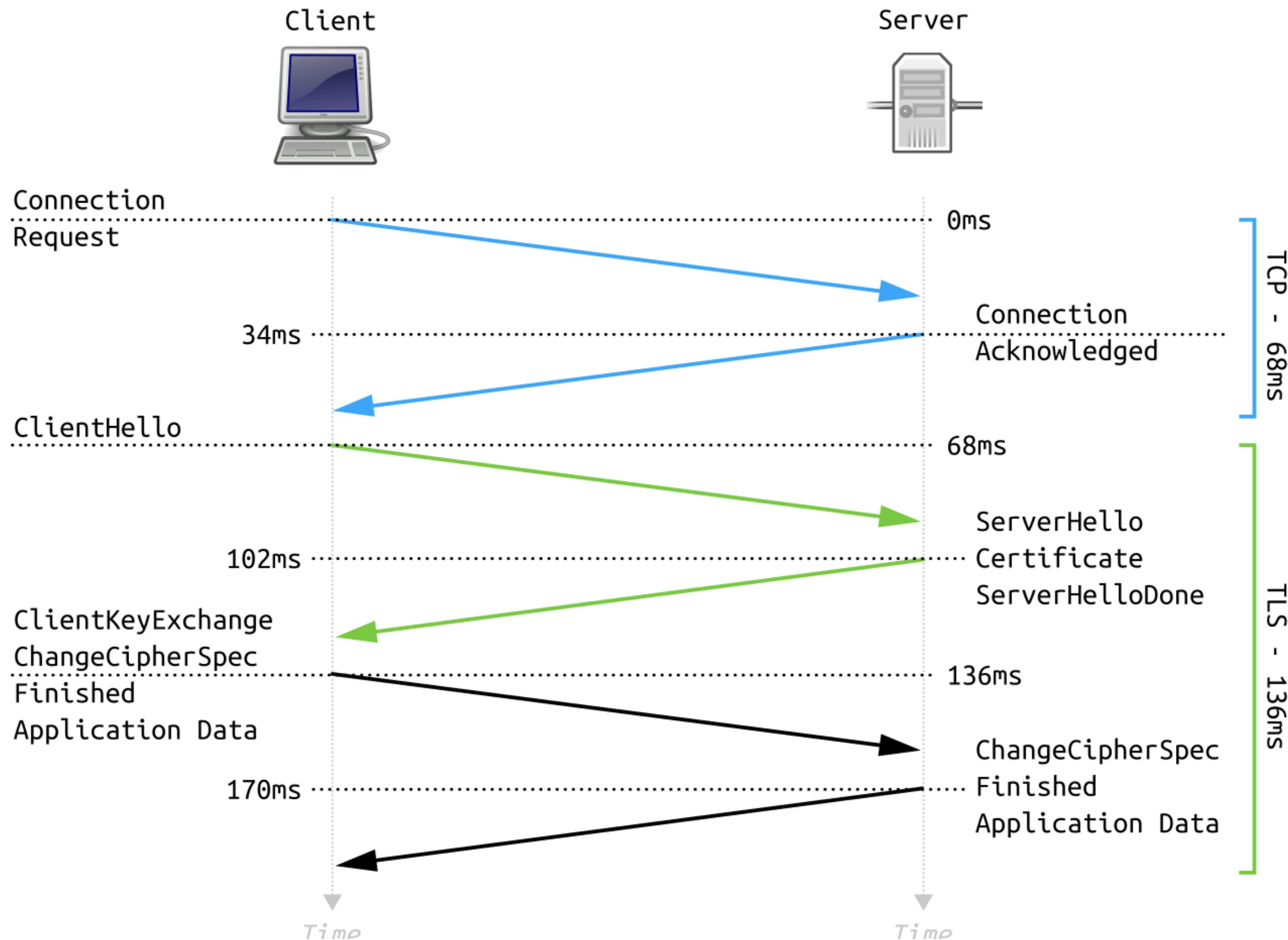
Génération de la clé de session

Les deux parties calculent indépendamment la même clé de session symétrique



Confirmation

Échange de messages "Finished" chiffrés avec la nouvelle clé de session

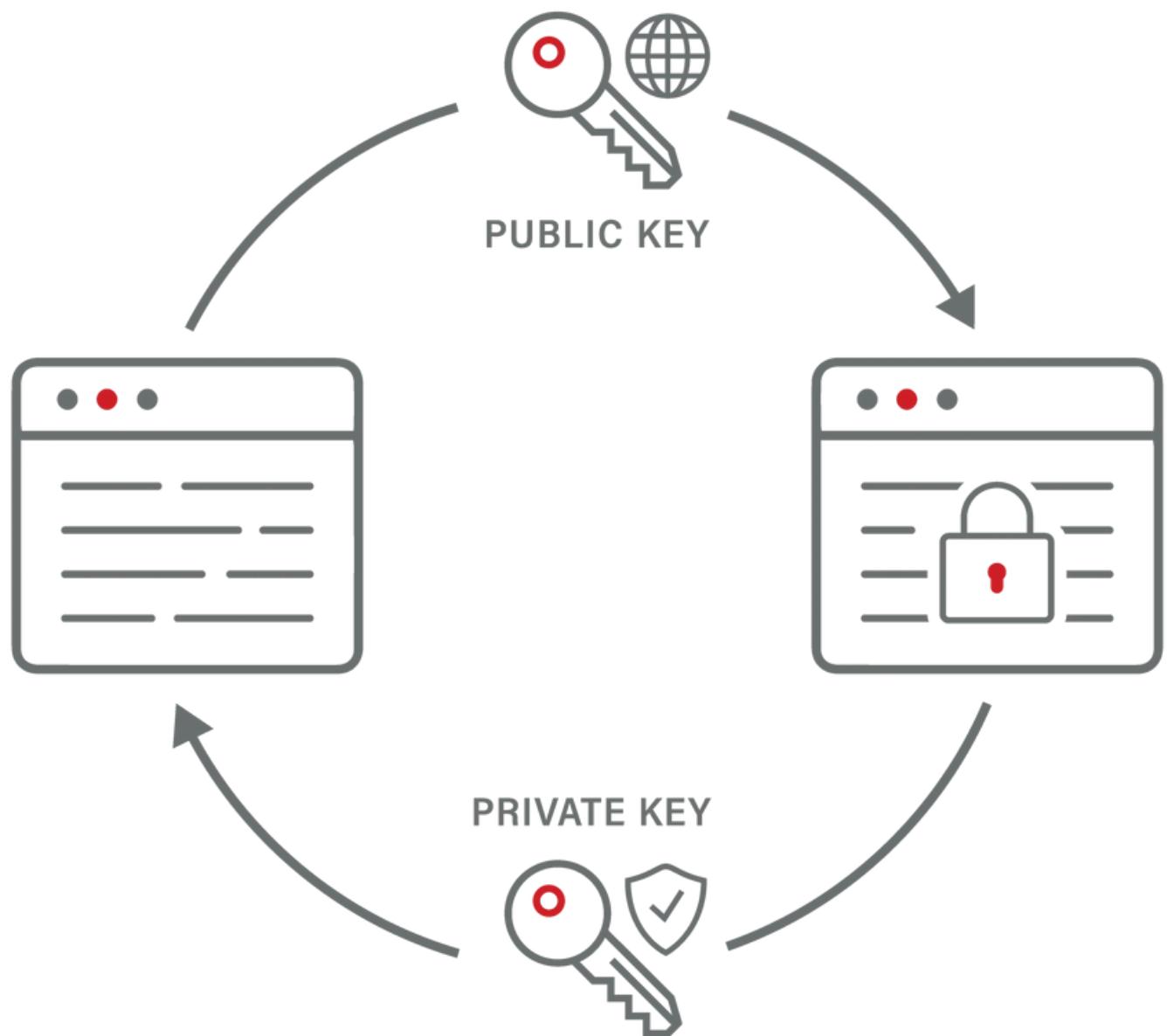


CRYPTOGRAPHIE DANS SSL/TLS

T

Cryptographie asymétrique (à clé publique)

- Utilise une paire de clés : publique (chiffrement) et privée (déchiffrement)
- Algorithmes : RSA, ECC (Elliptic Curve Cryptography), Diffie-Hellman
- Utilisée principalement pendant le handshake pour l'échange sécurisé de clés

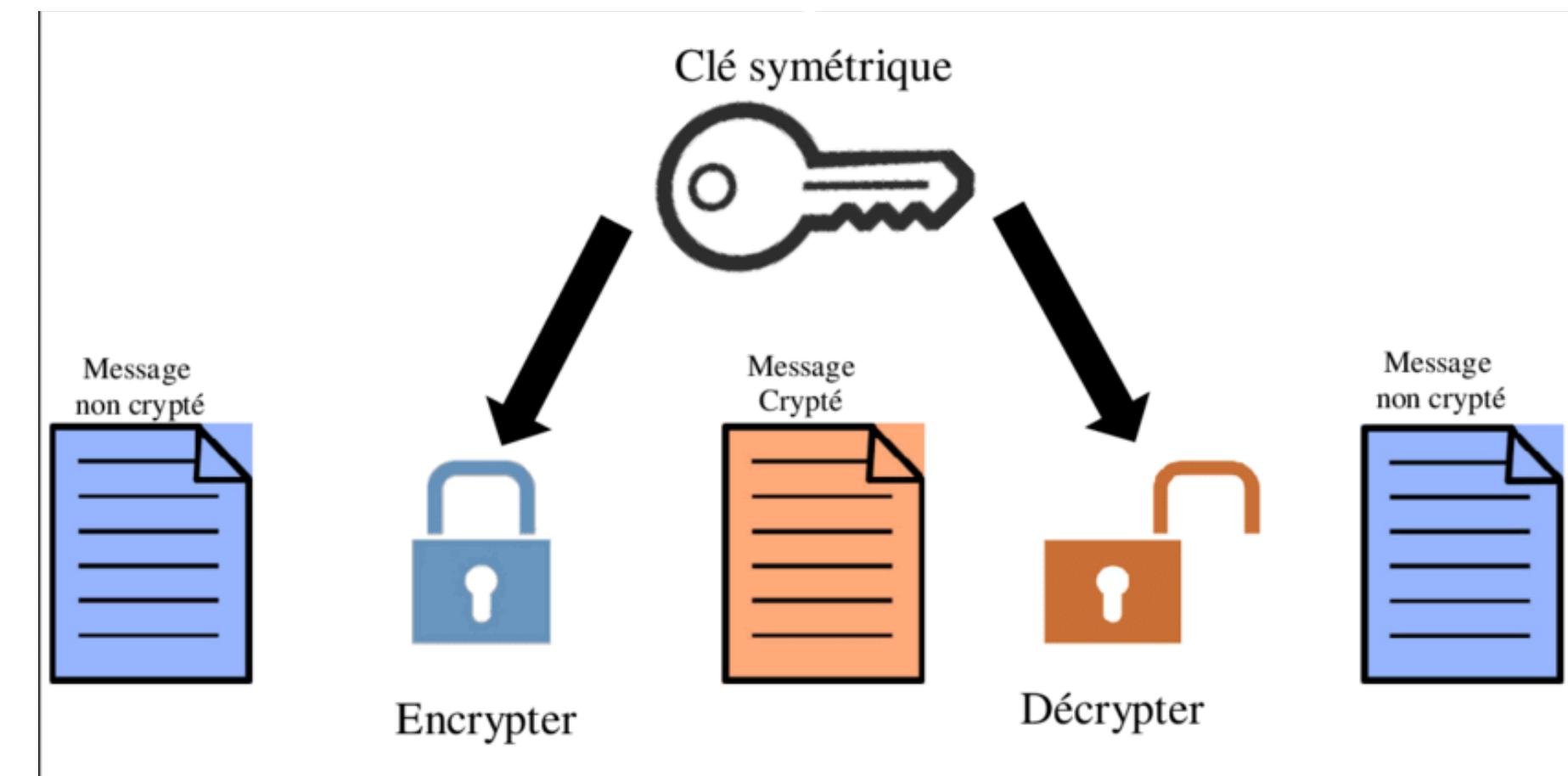


CRYPTOGRAPHIE DANS SSL/TLS

T

Cryptographie symétrique

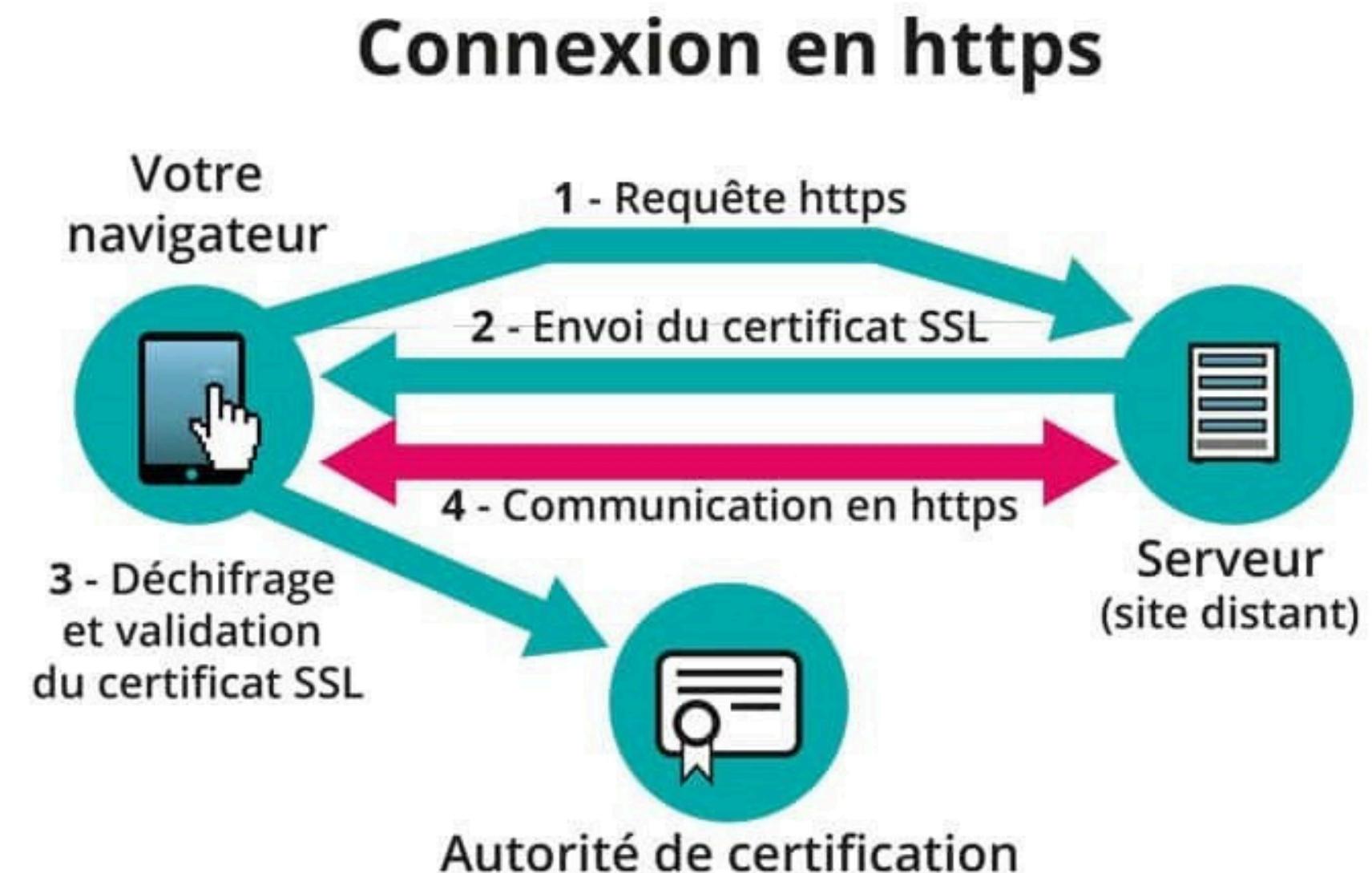
- Utilise une seule clé partagée pour le chiffrement et le déchiffrement
- Algorithmes : AES (Advanced Encryption Standard), ChaCha20
- Plus rapide que la cryptographie asymétrique
- Utilisée pour chiffrer les données échangées après le handshake



LES CERTIFICATS SSL/TLS

Définition et rôle

- Document numérique qui authentifie l'identité d'un site web
- Contient la clé publique du serveur et des informations d'identification
- Établit un lien sécurisé entre un serveur web et un navigateur



LES CERTIFICATS SSL/TLS

T

Autorités de Certification (CA)

- Organismes de confiance qui émettent et vérifient les certificats
- Valident l'identité du demandeur avant d'émettre un certificat
- Signent numériquement les certificats avec leur propre clé privée



CONFIGURER UN SERVEUR WEB AVEC SSL/TLS

1. Mettre à jour le système

```
nadir@nadir:~$ sudo apt update && sudo apt upgrade -y
```

2. Installer Apache

```
nadir@nadir:~$ sudo apt install apache2 -y  
[sudo] Mot de passe de nadir :
```

3. Autoriser Apache dans le pare-feu

```
nadir@nadir:~$ sudo ufw allow 'Apache Full'
```

```
nadir@nadir:~$ sudo ufw enable  
Le pare-feu est actif et lancé au démarrage du système
```

CONFIGURER UN SERVEUR WEB AVEC SSL/TLS

4. Configurer un VirtualHost

```
nadir@nadir:~$ sudo nano /etc/apache2/sites-available/nadir.com.conf
```

```
GNU nano 7.2          /etc/apache2/sites-available/nadir.com.conf *
<VirtualHost *:80>
    ServerName nadir.com
    ServerAlias www.nadir.com
    DocumentRoot /var/www/nadir.com

    ErrorLog ${APACHE_LOG_DIR}/error.log
    CustomLog ${APACHE_LOG_DIR}/access.log combined
</VirtualHost>
```

```
nadir@nadir:~$ sudo a2ensite nadir.com.conf
```

```
nadir@nadir:~$ sudo systemctl reload apache2
```

CONFIGURER UN SERVEUR WEB AVEC SSL/TLS

T

5. Créer un dossier pour les certificats

```
nadir@nadir:~$ sudo mkdir -p /etc/ssl/private
```

6. Générer le certificat et la clé privée

```
nadir@nadir:~$ sudo openssl req -x509 -nodes -days 365 \ -newkey rsa:2048 \
-keyout /etc/ssl/private/selfsigned.key \ -out /etc/ssl/certs/selfsigned.crt
```

CONFIGURER UN SERVEUR WEB AVEC SSL/TLS

CONFIGURATION APACHE AVEC SSL AUTO-SIGNÉ

7. Activer le module SSL

```
nadir@nadir:~$ sudo a2enmod ssl
```

8. Créer un VirtualHost HTTPS

```
nadir@nadir:~$ sudo nano /etc/apache2/sites-available/nadir.com-ssl.conf
```

```
GNU nano 7.2 /etc/apache2/sites-available/nadir.com-ssl.conf *
<VirtualHost *:443>
    ServerName nadir.com
    DocumentRoot /var/www/nadir.com

    SSLEngine on
    SSLCertificateFile /etc/ssl/certs/selfsigned.crt
    SSLCertificateKeyFile /etc/ssl/private/selfsigned.key

    <Directory /var/www/nadir.com>
        AllowOverride All
    <Directory>
</VirtualHost>
```

CONFIGURER UN SERVEUR WEB AVEC SSL/TLS

9. Activer la config SSL

```
nadir@nadir:~$ sudo a2ensite nadir.cm-ssl.conf
```

```
nadir@nadir:~$ sudo systemctl reload apache2
```

TYPES DE CERTIFICATS ET UTILISATION DE TLS



QU'EST-CE QU'UN CERTIFICAT SSL/TLS ?

Définition

Fichier numérique liant une clé cryptographique à un domaine/organisation.

Rôle

- Authentifie l'identité du serveur.
- Chiffre les données échangées.

Analogie

"Passeport numérique" pour les sites web.



LES CERTIFICATS SSL/TLS

Tableau de Comparaison des certificats (DV, OV, EV)

Type	Validation	Niveau de Confiance	Cas d'Usage
 DV	Vérification du domaine (DNS/email)	Faible ( <td>Blogs, sites personnels</td>	Blogs, sites personnels
 OV	Vérification de l'entreprise (registre commerce)	Moyenne ( <td>Sites d'entreprise, intranets</td>	Sites d'entreprise, intranets
 EV	Audit juridique approfondi (KYC)	Élevée ( <td>Banques, plateformes de paiement</td>	Banques, plateformes de paiement

Points Clés :

- DV : Émis en quelques minutes, idéal pour les petits projets.
- OV : Affiche le nom de l'entreprise dans le certificat (augmente la confiance).
- EV : Affiche la barre verte dans les anciens navigateurs (indice visuel fort).

CERTIFICATS WILDCARD & MULTI-DOMAINES (SAN)

Certificats Wildcard :

- Un certificat SSL Wildcard permet de sécuriser un nombre illimité de sous-domaines – à un niveau spécifique – pour un seul domaine avec un unique certificat.
- Protège *.domaine.com
- Idéal pour :
 - SaaS.
 - Hébergement mutualisé.
- Économique (1 cert = ∞ sous-domaines)



Certificats Multi-Domaines (SAN) :

- Un certificat SSL Multi-Domaines permet de sécuriser plusieurs noms de domaines complets (FQDN) avec un seul certificat.
- Sécurise plusieurs noms de domaine
- Idéal pour :
 - Entreprises multi-marques
 - Plateformes multi-sites
- Flexible (ex : domaine.com, domaine.net)

Exemples:



TLS POUR HTTPS ET SITES WEB SÉCURISÉS

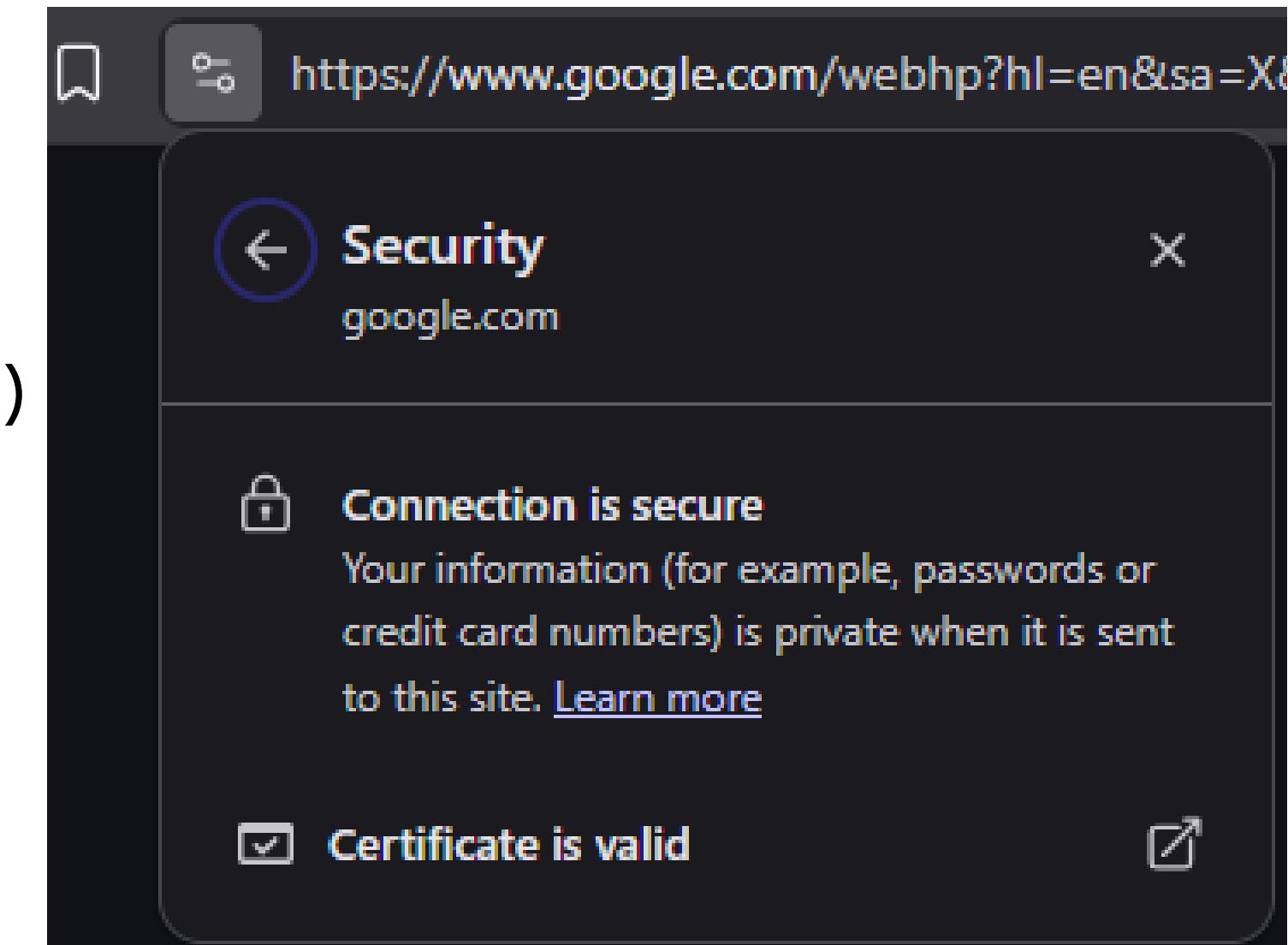


1. Fonctions clés :

- Chiffrement :
 - Données sensibles (logins, paiements, formulaires)
 - Algorithmes modernes : AES-256-GCM / ChaCha20-Poly1305
- Authentification :
 - Preuve d'identité du serveur (via certificats)

2. Indicateurs visuels :

- Navigateurs :
 - Cadenas vert (valide) / Rouge (expiré/risque)
 - "https://" en noir (standard) ou vert (EV)



TLS POUR LES E-MAILS (SMTP/IMAP/POP3)

1. Protocoles & Ports Sécurisés

Protocole	Port Standard	Port Sécurisé (TLS)	Usage
SMTP	25	587 (STARTTLS)	Envoi d'emails
IMAP	143	993 (IMAPS)	Synchronisation
POP3	110	995 (POP3S)	Téléchargement

2 Bonnes Pratiques

- Forcer TLS : désactiver les ports non chiffrés (25/143/110)
- MTA-STS : Prévenir les attaques "downgrade" (ex : STARTTLS stripping)
- Renouvellement : certificats valides < 1 an (Let's Encrypt)

3 Exemple Concret

- Gmail rejette les e-mails non TLS depuis 2020.
- Office 365 impose TLS 1.2+ pour les transferts.

TLS DANS LES VPN & CLOUD

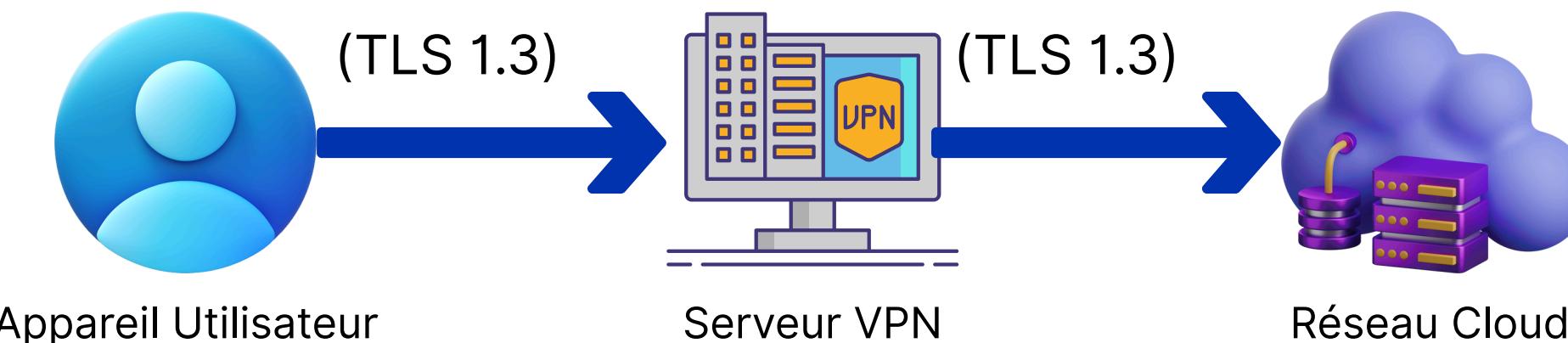
1. TLS pour les VPN

- OpenVPN :
 - Utilise TLS pour l'échange de clés
 - Ports standards : 1194 (UDP) / 443 (TCP)
- WireGuard :
 - Alternative moderne (cryptographie plus légère)
 - Tunnel chiffré via Noise Protocol (inspiré de TLS)

2. TLS dans le Cloud

- Protection des Données :
 - Chiffrement des API (AWS API Gateway, Azure Functions)
 - Transferts sécurisés (S3 → EC2, etc.)
- Authentification Mutuelle (mTLS) :
 - Obligatoire pour les architectures Zero Trust
 - Exemple : Accès aux bases de données cloud

4. Diagramme VPN :



3. Chiffres Clés :

- 92% des entreprises cloud utilisent TLS 1.2+ (2024)
- Réduction de 67 % des fuites de données avec mTLS

VULNÉRABILITÉS ET ATTAQUES LIÉES À SSL/TLS



POURQUOI S'INTÉRESSER AUX VULNÉRABILITÉS ?

Protocole très utilisé

SSL/TLS protège la majorité des communications sur Internet : sites web, emails, paiements, etc.



Cible fréquente

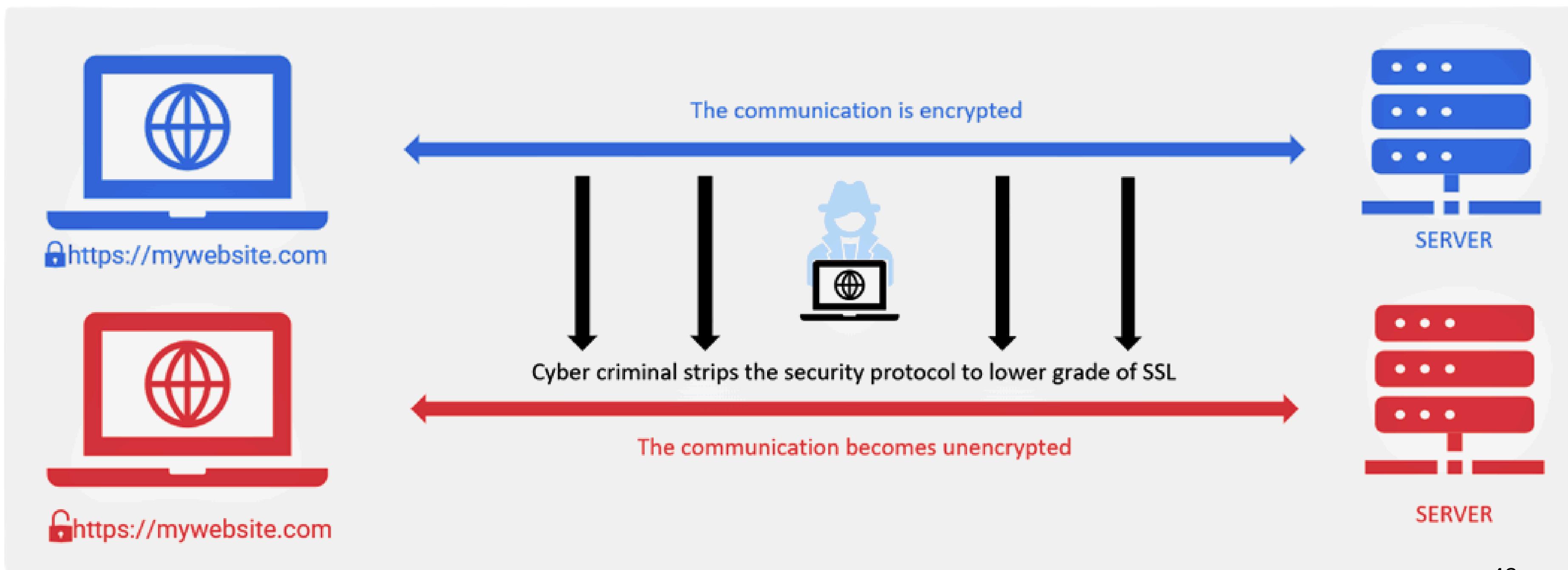
Les attaquants cherchent les failles des systèmes les plus répandus.

Protocole Versions obsolètes = dangerutilisé

Les anciennes versions (SSL 3.0, TLS 1.0/1.1) contiennent des failles graves.

ATTAQUE MAN-IN-THE-MIDDLE (MITM)

L'attaquant se place entre le client et le serveur, et intercepte les messages échangés.

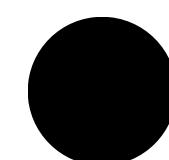


POODLE ATTACK (SSL 3.0)

- Poodle = Padding Oracle On Downgraded Legacy Encryption
- Cible : le chiffrement CBC dans SSL 3.0
- Résultat : le pirate peut récupérer des données sensibles en clair (cookies, sessions...)

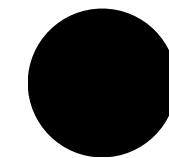


AUTRES ATTAQUES CONNUES



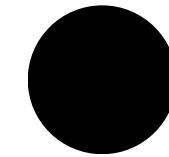
BEAST

Injection CBC dans TLS 1.0



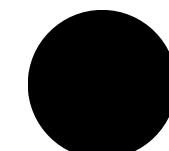
Heartbleed

Fuite de mémoire dans OpenSSL



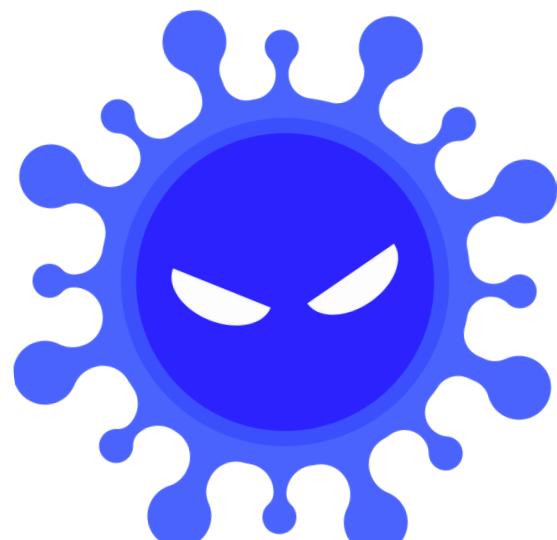
DROWN

Déchiffrement via anciens serveurs SSL



CRIME/BREACH

Exploite la compression pour voler des infos



POURQUOI DÉSACTIVER SSL ET TLS 1.0/1.1 ?

Nouvelles versions = Sécurisation optimale

TLS 1.2 et TLS 1.3

- Handshake plus rapide (surtout TLS 1.3)
- Chiffrement moderne (AES-GCM, ChaCha20)
- Compatibilité avec les normes modernes de cybersécurité (ex. : PCI-DSS, RGPD)
- Support complet des mécanismes modernes : PFS, ALPN, SNI

Anciennes versions = Risques majeurs

SSL 3.0, TLS 1.0, TLS 1.1

- Contiennent des failles critiques :
 - POODLE, BEAST, DROWN
- Incompatibles avec les standards modernes de sécurité
- Absence de mécanismes avancés comme Perfect Forward Secrecy (PFS)
- Ciblées par les attaques les plus courantes

OUTILS POUR TESTER LA SÉCURITÉ TLS

T

Outils



Qualys SSL Labs

Analyse complète d'un site, donne une note (A à F)



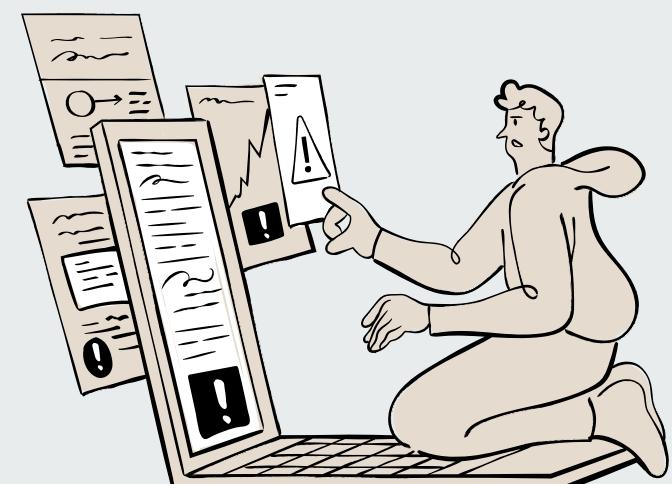
Outil en ligne de commande pour tester les versions et certificats



Capture et analyse des paquets TLS

Objectifs

- Déetecter les versions obsolètes (SSL 3.0, TLS 1.0/1.1)
- Vérifier la validité des certificats
- S'assurer que les suites de chiffrement sont à jour
- Repérer les faiblesses dans la configuration du serveur



CONCLUSION

- Évolution de **SSL** vers **TLS** : SSL, obsolète et vulnérable, a été remplacé par TLS pour corriger des failles majeures comme Man-in-the-Middle, POODLE et Heartbleed. Les versions récentes de TLS (1.2 et 1.3) offrent une sécurité renforcée.
- Fonctionnement des Certificats et Applications de **TLS** : TLS repose sur des certificats valides, qui assurent la confiance grâce aux autorités de certification (CA). Il est utilisé dans divers contextes, comme HTTPS pour les sites web, les emails sécurisés, et les VPN.
- Vulnérabilités et Attaques liées à **SSL/TLS** : Les failles comme BEAST, Heartbleed, et DROWN montrent la nécessité de désactiver SSL et les anciennes versions de TLS. Tester la sécurité avec des outils comme SSL Labs et OpenSSL est essentiel pour garantir une protection optimale.



MERCI
POUR VOTRE
ATTENTION

