11. HTTPS

25 octobre 2023

Développement web il3

HTTPS

HE-Arc (DGR) 2022

Sécuriser un site web

- Authentification du serveur
  - Assurer que le serveur est celui qu'il prétend être
- Intégrité des données
  - Assurer que les données reçues sont celles qui ont été envoyées
- Confidentialité des données
  - Eviter que des tiers ne puissent voir les données
- Authentification du client (optionnelle)
  - Assurer que le client est celui qu'il prétend être
- Pour un site web, ces services sont fournis par https
  - HTTPS : HTTP sécurisé par SSL/TLS, par défaut sur le port 443

Secure Socket Layer –> Transport Layer Security

- Conçu par Netscape (v2.0 en 1994, v3.0 en 1996)
- Brevet racheté par l'IETF : TLS v1.0 en 1999 (SSL 3.1), v1.3 en 2018
- Couche Application :
  - Entre les couches transport et application
  - Pas besoin de modifier la pile TCP/IP
- Possibilité de sécuriser d'autres protocoles :
  - HTTP, SMTP, SIP, …
- Services offerts :
  - Authentification serveur + intégrité données
  - Confidentialité des données
  - Authentification optionnelle du client
- Certificats (clé publique associée au certificat)

Rôle d'un certificat

- Garantir le lien entre une entité physique et une entité numérique :
  - Intégrité des données
  - Authentification
  - Confidentialité
- Document contenant une identité et une signature numérique
- Utilisations courantes : https, mails
- Délivré par une autorité de certification
- Certificats clients

Autorité de Certification

- Tiers de confiance
  - enregistrée et certifiée par des autorités publiques ou de gouvernance de l'Internet
- Rôle :
  - Vérifier et garantir les informations sur l'entité
  - Emettre, délivrer et révoquer les certificats
  - Leur assigner une période de validité
  - Maintenir la liste des certificats valides/révoqués

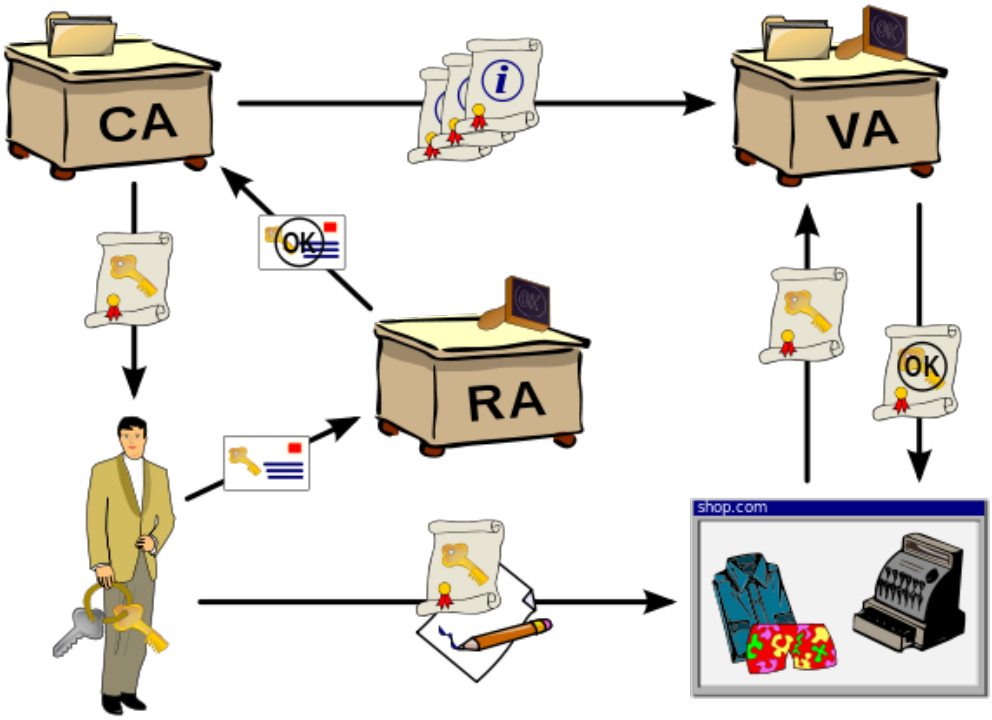
- Certificats auto-signés :
  - usage interne
  - pas de tiers de confiance

Contenu d'un certificat X509

- version de X.509 (v3, depuis 1996)
- numéro de série du certificat
- algorithme de chiffrement utilisé pour signer le certificat
- nom de l'AC émettrice
- informations sur la clé publique
- dates de début et fin de validité du certificat
- clé publique du propriétaire du certificat
- signature de l'émetteur du certificat (thumbprint)
- ...

Composants d'une PKI[1]

CA : Autorité de certification - VA : Autorité de validation - RA : Autorité d'enregistrement



Scénario simplifié de connexion HTTPS

1. Le client demande une page sécurisée
2. Le serveur émet sa clé publique et son certificat
3. Le client vérifie la validité du certificat (et qu'il correspond au site)
4. Le client utilise la clé publique pour chiffrer la clé symétrique (CS) utilisée ensuite
5. Le serveur déchiffre cette CS (avec sa clé privée) et l'utilise pour décoder la requête HTTPS
6. Le serveur répond à la requête en chiffrant avec la CS
7. Le navigateur décode la réponse avec la CS

- En images[2], ou ici[3] ou en slides[4]
- 2-5 en TCP

[1] https://en.wikipedia.org/wiki/Public_key_infrastructure
[2] https://tiptopsecurity.com/how-does-https-work-rsa-encryption-explained/
[3] http://software-engineer-tips-and-tricks.blogspot.ch/2012/08/ssl-in-pictures.html?view=sidebar
[4] https://www.youtube.com/embed/iQsKdtjwtYI?rel=0

Déploiement

- Installer OpenSSL
- (Créer son autorité de certification si autosigné)
- Obtenir le certificat et la clé privée du serveur
- Configurer httpd. Pour Apache :
  - virtual host (port 443), ssl.conf, (ports.conf)
- Création de l'arborescence sécurisée
- Démarrage serveur
- OU BIEN utiliser Let's encrypt[5]
- OU BIEN utiliser un serveur pré-configuré comme Caddy[6]

HTTPS Aujourd'hui

- Il n'y a plus de bonne raison d'utiliser HTTP
- TLS toujours utilisé avec HTTP2 et HTTP3
- HTTP2 et 3 minimisent et accélèrent les échanges
- Certificats gratuits
- Mise en place simplifiée

Ressources

- Security Party 23.10.2009[7]
- SebSauvage[8]
- HTTPS en détails :
  - Diagramme de séquence HTTPS[9]
  - Diagramme de séquence SPDY[10]
  - SSL[11] en détails
- Durée de vie de la Clé Symétrique[12]
- Faux Certificat[13]
- Autorités de certification :

[5] https://letsencrypt.org/
[6] https://caddyserver.com/
[7] https://wiki.alphanet.ch/Ateliers/PresentationSecurityParty
[8] http://www.sebsauvage.net/comprendre/ssl/
[9] https://www.eventhelix.com/networking/SSL.pdf
[10] https://www.eventhelix.com/networking/ssl-tls/https-ssl-tls-session-for-spdy.pdf
[11] https://security.stackexchange.com/questions/20803/how-does-ssl-tls-work/20847#20847
[12] https://security.stackexchange.com/questions/55454/how-long-does-an-https-symmetric-key-last
[13] https://www.win.tue.nl/hashclash/rogue-ca/

  - Let's Encrypt[14]
  - CA Cert[15]
  - SSLforFree[16]

- Différences TLS / SSH : Snailbook[17], StackExchange[18]

Sources

[14] https://letsencrypt.org/
[15] http://www.cacert.org/
[16] https://www.sslforfree.com/
[17] http://www.snailbook.com/faq/ssl.auto.html
[18] http://security.stackexchange.com/questions/1599/what-is-the-difference-between-ssl-vs-ssh-which-is-more-secure