Chiffrement, certificats

Denis Pugnère – CNRS / IN2P3 / IP2I

ANF « IoT perfectionnement »

11 au 15 septembre 2023, Centre Jean-Bosco, Lyon

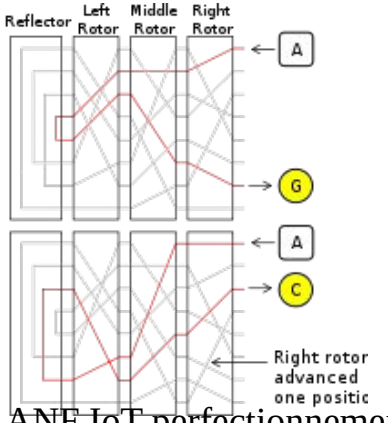
Plan

- Objectif, histoire
- Vocabulaire
- Les nombres aléatoires
- Fonctions de hachage
- Le chiffrement à clé secrète (symétrique)
- Le chiffrement à clé publique (asymétrique)
- La négociation de clés

Introduction

- Les objectifs
  - Conserver un **secret**
  - Assurer du secret des communications entre **plusieurs parties**
- N'est pas nouveau
  - Sparte: les premiers codes militaires
  - Le **Scytale**: le premier dispositif matériel de chiffrement (par transposition)
    le message est inscrit sur une lanière en cuir enroulée sur un bâton,
    le destinataire doit posséder un bâton de même diamètre pour lire le message en clair
  - Le premier carré magique (« **carré de Polybius** », chiffrement par substitution), 150 avant JC
- La machine **Enigma**
  - La plus célébre des machines crypto
  - Une réalisation allemande
  - Des rotors, chiffrement par substitution

	1	2	3	4	5
1	A	B	C	D	E
2	F	G	H	I/J	K
3	L	M	N	O	P
4	Q	R	S	T	U
5	V	W	X	Y	Z



Vocabulaire

- **Cryptographie** : La création et l’utilisation d’écritures secrètes
- **Cryptanalyse** : Le déchiffrement des écritures secrètes
- **Stéganographie** : Les écritures dissimulées
- **Chiffrement** : Passer d’un texte clair à un texte chiffré
- **Déchiffrement** : Passer du texte chiffré au texte clair
- **Décryptage** : Passer du texte chiffré au texte clair sans en connaître la clé ou méthode
- **Cryptage** : ?
- **Texte clair** : Le message original, dans sa forme non dissimulée
- **Clé** : La partie permettant de passer du texte clair au texte chiffré, ou réciproquement
- **Substitution** : Les méthodes de chiffrement remplaçant un signe par un autre
- **Permutation** : Les méthodes de chiffrement modifiant l’ordre des signes
- **Codes** : Les listes arbitraires associant un « code » à un ensemble de signe : Chiffrer -> Coder, Déchiffrer -> Décoder

Les nombres aléatoires

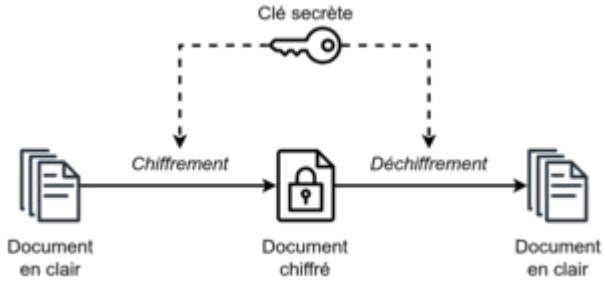
- La sécurité de base en cryptographie
  - Ne pas pouvoir deviner ce qu’une tierce partie a choisi comme base
  - => **Utilisation des nombres aléatoires**
- Les différentes méthodes de génération
  - Le « véritable » aléatoire : dispositif matériel
  - Le pseudo aléatoire déterministe : algorithmes à « source »
  - Le pseudo aléatoire à sources externes
- Le terme « **Entropie** »
  - L’entropie physique est la mesure du « désordre » d’un système
  - L’entropie informatique est appelée ainsi par association
  - X bits d’entropie pour générer Y bits aléatoires
- La solidité des clés
  - Un bon générateur de nombres aléatoire génère des clés solides
  - Un mauvais générateur aléatoire génère des clés faciles à trouver

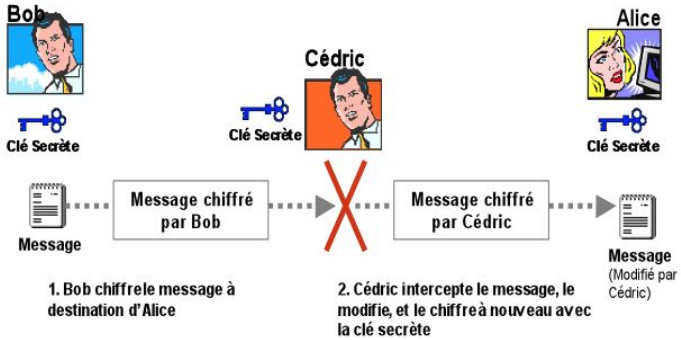
Fonctions de hachage

- But :
  - **Transformer** de manière déterministe, **une suite de bits** de longueur quelconque, **en un condensat**, encore appelé **empreinte** ou **haché**, de taille fixée.
  - **Réduire** une valeur de grande taille en une valeur plus petite
  - La **comparaison de la valeur de hachage** est **plus pratique** que la comparaison de la valeur d'origine
- Caractéristiques
  - non inversible :
    - pas possible étant donné un condensat de trouver un message dont l'image par la fonction de hachage est égale à ce condensat
    - pas possible de trouver deux messages distincts ayant même condensat (fonction de hachage doit être **sans collision**)
- On l'appelle aussi « **empreinte** », à ne pas confondre avec « signature »
- Depuis les attaques sur les principales familles de fonctions de hachage en août 2004, évolution des recommandations :
  - Collisions sur MD5 et SHA-1 => obsolètes, ne plus les utiliser
  - SHA-2 (SHA-224, SHA-256, SHA-384, SHA-512) reste utilisable, SHA-3...
- Applications :
  - **Signature numérique** afin de réduire le message de longueur quelconque à un simple condensat de taille fixée et petite.
  - **Le contrôle d'intégrité = vérification si un document a été modifié** (le changement d'une partie du document change son empreinte),
  - codes d'**authentification de message** (par exemple HMAC).
  - Déduplication : détecter des fichiers identiques sur un serveur
  - Hachage de mots de passe : **stockage de mots de passe hachés.et salés** : salt = random ; h = hash(salt,pw)

Le chiffrement à clés secrètes (symétrique)

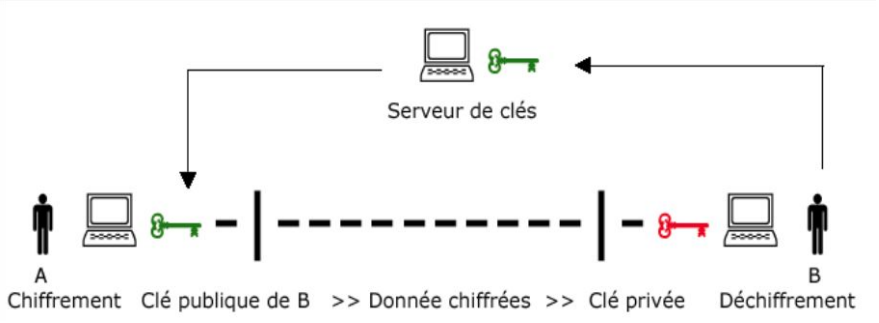
- Les principes de base du **chiffrement à clé secrète**
  - La même clé est utilisée pour chiffrer et déchiffrer
  - La sécurité des données réside dans la solidité de la clé (et sa non divulgation)
  - On l'appelle aussi « **chiffrement à clé symétrique** »
- Les deux modes de chiffrement
  - Le chiffrement **par flot**
  - Le chiffrement **par bloc**
- Liste d'algorithmes symétriques communs : RC4 + RC5 (obsolètes), DES (obsolète), Triple-DES, AES, Blowfish, Serpent, Twofish
- Se référer aux guides de l'ANSSI pour **choisir la taille de la clé** (contre attaques **exhaustives** ou par **dictionnaire**)
- Les **avantages** du chiffrement à clé secrète
  - **Rapidité**
  - Implémentation sur du matériel (primitives AES sur les processeurs…)
- Les **inconvénients** du chiffrement à clé secrète
  - Chiffrement **non authentifié**, Identité de l'expéditeur ?
  - Pas d'intégrité du message => rajouter un mécanisme d'authentification de message (MAC)





Chiffrement par clés publiques (asymétrique)

- Deux grandes familles mathématiques
  - Les facteurs premiers (RSA) : N = P x Q
  - Les courbes elliptiques (Curve25519) et logarithmes « discrets »
- Aussi appelé chiffrement à **clés asymétriques**
  - Une clé « maître » est choisie, on en déduit **une paire** de clés
  - L'une des **clés (publique)** permet de **chiffrer**
  - L'autre **clé (privée)** permet de **déchiffrer**
  - Il n'existe pas de fonction « simple » permettant de déterminer une clé à partir de l'autre
  - Il n'existe pas de fonction permettant de retrouver la clé maître à partir d'une clé
- La grande difficulté de la cryptographie a clef publique
  - Les performances
  - Les tailles des clefs (RSA > 2048 bits)
  - La diffusion de la clé publique



Chiffrement par clés publiques (asymétrique)

- Principe de base du chiffrement à **clé publique**
  - Il est possible de divulguer une partie des clés
  - Il n'est pas possible de chiffrer ou de déchiffrer uniquement à l'aide de la partie publique
- Les clés publiques
  - **doivent** généralement **être distribuées de façon authentique**
    - Via un canal de communication de confiance
    - Via une **Infrastructures de Gestion de Clés (IGC)** / Public Key Infrastructure (PKI)
- **Vérifier l'authenticité** des parties : un serveur (SSH, https…), un utilisateur (certificat X509, bi-clés ssh RSA ou autre)
  - SSH :
    - Du serveur via l'empreinte présentée lors de la connexion (et de la comparaison par rapport à l'empreinte précédente stockée dans ***.ssh/known_hosts*** *du client*)
    - De l'utilisateur par la transmission d'un challenge chiffré par la clé publique de l'utilisateur (stockée dans ***.ssh/authrorized_keys*** du serveur) et déchiffré par la clé privée de l'utilisateur (stocké dans **.ssh/id_rsa** par exemple)
  - HTTPS : via la confiance que l'on place dans le certificat présenté par le serveur

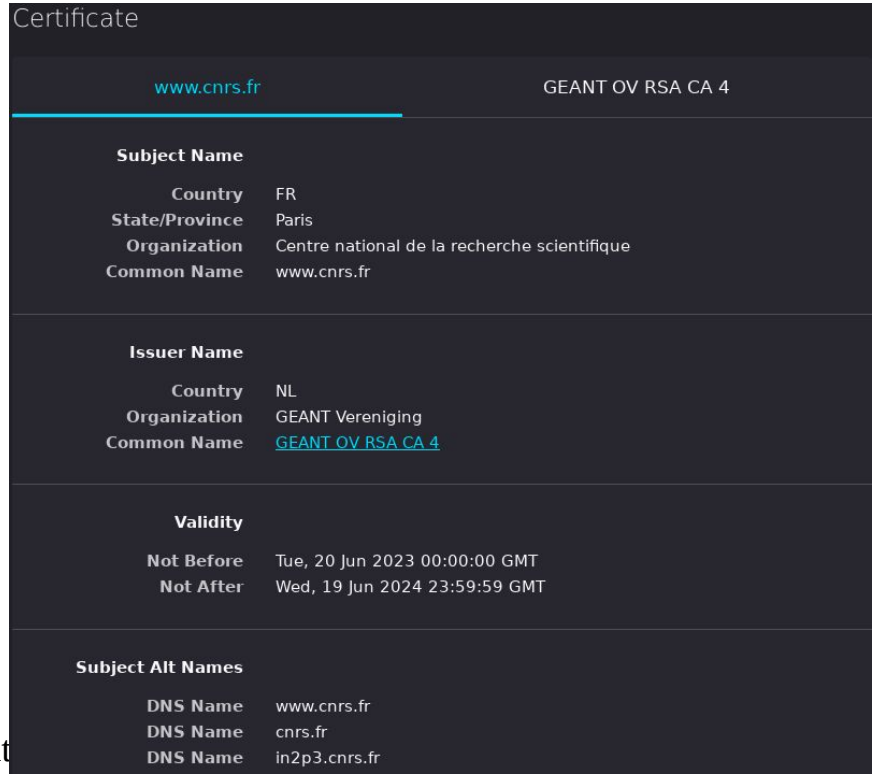
Les combinaisons de clés asymétriques pour des applications

- **Signer un message**
  - Chiffrer un message avec sa clé privée
  - N’importe qui (connaissant la clé publique), peut déchiffrer le message
  - Comme seul l’émetteur connaît la clé privée
    => certitude que seul l’émetteur a pu générer le message => message authentique
    => opposable à l’émetteur (non-répudiation)
  - Pas possible avec un mécanisme symétrique
- **Chiffrement asymétrique + hachage = La signature électronique**
  - Un hachage chiffré par une clef privée
  - Si le déchiffrement par la clef publique donne le bon hachage, les données sont donc les bonnes
- **La certification de la clef publique**
  - La signature ne prouve que **quelqu’un dispose de la clef privée**, mais **pas l’identité de cette personne**
  - L’importance de disposer d’un mécanisme de distribution des clés (IGC) ou **d’authentification de clef publique de confiance**

Les certificats X509

- **Le certificat** atteste l'**association d'<u>une clef</u> avec une <u>entité</u>, <u>vérifiée</u>** par une **<u>organisation</u>**
- Réside sur l'algorithme RSA
  - Un certificat X509 associe la clef (publique) avec un « nom »
  - Le nom est un nom X500 … ou un nom « alternatif » (nom de machine, adresse électronique), exemple ici Certificat de ***www.cnrs.fr*** valable aussi pour ***cnrs.fr*** , ***in2p3.cnrs.fr*** …
- Pour générer certificat SSL il est nécessaire de générer une CSR (Certificate Signing Request (Demande de Signature de Certificat) :
  - CSR : texte chiffré qui précise de manière unique qui vous êtes et quel nom de domaine
  - Clé privée (qui va prouver que vous avez le certificat)
- Le certificat dit que vous êtes « Untel » ici ***www.cnrs.fr***
- Un certificat doit être signé par une autorité de certification (ici par « GEANT Vereniging » pour prouver qu'il n'a pas été altéré

  Lors de la demande du certificat par le CNRS, **le signataire** (GEANT Vereniging), **<u>après contrôle</u>**, utilise sa clef privée pour **signer** le certificat ***www.cnrs.fr***
- Une fois signé par l'autorité de certification (GEANT Vereniging), des attributs sont rajoutés (numéro de série, DN, durée de validité...)
- Infrastructure de gestion de clefs « PKI » : Autorité de certifications enregistrées (ou pas) dans les navigateurs
- Vérification



Clés de session

- Le **chiffrement à clés publiques plus lent** (facteur 100 à 1000) **que le chiffrement à clés secrètes** (symétrique) :
- Introduction du concept de « **clé de session** »
  - Déterminer une clé « **jetable** » basé sur une clé secrète (symétrique)
  - Le principe est de calculer une clé sans qu’une tierce partie ne puisse la trouver, cette **clé est calculée par les 2 parties, dans un canal chiffré (par des clés asymétriques)**
  - **Utilisée qu'une seule fois**, pendant un **laps de temps donné**, pour le chiffrement et le déchiffrement de données entre deux parties
  - les futures conversations entre les deux parties seraient cryptées avec des clés de session différentes.Une clé de session est comme un mot de passe que quelqu'un réinitialise à chaque fois qu'il se connecte.

Guides, référentiels

- https://www.ssi.gouv.fr/guide/mecanismes-cryptographiques/
  - Guide de sélection d'algorithmes cryptographiques
  - Règles et recommandations concernant le choix et le dimensionnement de mécanismes cryptographiques