Formation Unix/Linux
Compléments CCPRO

Guillaume Allègre
Guillaume.Allegre@silecs.info

CCPRO

2019

Rappel : cryptographie symétrique et asymétrique *110.3*

Chiffrement symétrique

Une seule clé pour le chiffrage et le déchiffrage

Chiffrement asymétrique

- Principe
  - une clé privée + une clé publique
  - une clé chiffre, l'autre déchiffre
  - secret : chiffrement avec la clé publique du destinataire
  - authentification : chiffrement avec la clé privée de l'expéditeur
  - une infrastructure de distribution des clés publiques (PKI)
- Diversité des usages
  - SSH : connexions distantes sécurisées
  - Cryptographie personnelle : GnuPG (gpg) / PGP, Enigmail...
  - Certificats X.509 (SSL) $\longrightarrow$ HTTPS, IMAPS...

Protocoles SSL et TLS

- Protocoles
  - Protocole réseau cryptographique
  - Situé entre TCP et la couche applicative (HTTP...)
  - SSL : Secure Socket Layer v.2-3 (1995-96 Netscape)
  - TLS : Transport Layer Security

    TLS 1.0 1999 RFC 2246
    TLS 1.1 2006 RFC 4346
    TLS 1.2 2008 RFC 5246

    Report Protocol confidentialité et intégrité
    Handshake Protocol authentification mutuelle par clé publique

Suite cryptographique (*cipher suite*)

Constitution

- Échange de clés — ex. RSA, ECDH
- Authentification — ex. RSA, DSA, ECDSA.
- Chiffrement symétrique (par blocs *Bulk encryption*) — ex. AES, DES
- Code de message d'authentification (MAC)
  - hachage cryptographique — ex. SHA-1, MD5
  - Fonction pseudo-aléatoire (PRF)

Exemple

`ECDHE - RSA - AES128-GCM - SHA256`

Certificats numériques (à clef publique)

- Certificat numérique
  - une identité (nom, adresse, URL...)
  - une clé publique
  - une signature certifiant la correspondance

- Principaux types de certificats

  X.509 (ITU-T) hiérarchique, autorité de certification (CA)

  OpenPGP décentralisé, réseau de confiance (*Web of Trust*)

Certificat numérique en pratique

En pratique : constitution d'un certificat

- Numéro de série : identifiant unique
- Sujet : personne ou organisation identifiée (nom...)
- Clé publique
- Empreinte de clé publique (et algorithme précisé)
- Validité : dates de début et fin du certificat
- Émetteur (*Issuer*), qui a vérifié l'identité (CA...)
- Signature : la signature de l'émetteur du certificat (algo précisé)

...

OpenSSL

- Bibliothèque libre pour les protocoles SSL et TLS
  - Commande `openssl` : utilitaire générique SSL/TLS
  - Bibliothèques `libssl`, `libcrypto` (paquet `libssl`)

- Commande `openssl` : utilitaire générique SSL/TLS
  - création de paramètres des clefs RSA, DH et DSA
  - création de certificats X.509, CSRs et CRLs
  - calcul de condensés de messages
  - chiffrement et le déchiffrement
  - test de clients et serveurs SSL/TLS
  - gestion de courriers S/MIME signés ou chiffrés

OpenSSL - Répertoire `/etc/ssl`

- `openssl.conf`
- `certs/` certificats
  - fournis par la distribution / les navigateurs
  - ajoutés par l'administrateur
- `private/` clés privées

OpenSSL - Fichiers et formats (simplifié)

- `.key` clé privée
- `.csr` *certificate signing request*
- `.crt` certificat (signé)
- `.crl` *certificate revocation list*
- `.pem` format conteneur, encodé en base-64

```
-----BEGIN CERTIFICATE-----
MIIGFDCCA/ygAwIBAgIIU+w77vuySF8wDQYJKoZIhvcNAQEFBQAwUTELMAk
[...]
QOCgFzzr6juwcqajuUpLXhZI9LK8yIySxZ2frHI2vDSANGupi5LAuBft7HZ
jLMi6Et8Vcad+qMUu2WFbm5PEn4KPJ2V
-----END CERTIFICATE-----
```

OpenSSL - exemples

```
-----BEGIN CERTIFICATE-----
MIIGFDCCA/ygAwIBAgIIU+w77vuySF8wDQYJKoZIhvcNAQEFBQAwUTELMAk
[...]
QOCgFzzr6juwcqajuUpLXhZI9LK8yIySxZ2frHI2vDSANGupi5LAuBft7HZ
jLMi6Et8Vcad+qMUu2WFbm5PEn4KPJ2V
-----END CERTIFICATE-----
```

Diagnostic

- `cd /etc/ssl`
- `openssl x509 -in certs/ssl-cert-snakeoil.pem -text -noout`
- `sudo openssl rsa -in private/ssl-cert-snakeoil.key -check -noout`

Utilisation OpenSSL dans Apache

- Séparer `/var/www/html` et `/var/www/ssl`
- Activer HTTP/SSL avec le certificat `snakeoil` (Debian)
  - installer `ssl-cert` si nécessaire
  - `a2enmod ssl`
  - `a2ensite default-ssl`
- Mettre à jour le fichier `ssl.conf` avec notre certificat.

Diagnostic des protocoles texte sur SSL/TLS

- Session `openssl s_client`

```
$ openssl s_client -connect localhost:443
CONNECTED(00000003)
depth=0 /CN=cressida.localnet
[...]
GET /
<html><body><h1>It works!</h1></body></html>
closed
```

- Alternative : `gnutls-cli -p 443 --insecure localhost`

Résolution de noms locale

109.4

Fichier /etc/hosts

- établit une correspondance *nom d'hôte* $\leftrightarrows$ *adresse IP*
- compatible IPv4 et IPv6

Exemple

```
127.0.0.1   localhost localhost.localnet
# Virtualhosts apache
127.0.0.1   www www.localnet

#localnet
192.168.0.1  jupiter  jupiter.localnet
192.168.0.2  saturne  saturne.localnet

# The following lines are desirable for IPv6 capable hosts
::1     ip6-localhost ip6-loopback
fe00::0 ip6-localnet
```

Domaine Name Server (DNS)

- Permet une équivalence entre nom et adresse IP
  - ex. cressida.silecs.info $\leftrightarrows$ 82.67.62.169
  - ex. www.silecs.info $\rightarrow$ silecs.info $\leftrightarrows$ 213.186.33.2 (alias)
  - ex. lear.silecs.info $\rightarrow$ 88.172.133.112 $\rightarrow$ . . . .proxad.net
- Fonctionnement par arborescence de serveurs
  - Dans chaque serveur : cache pour minimiser les requêtes
  - Un *authoritative server* fait autorité pour un domaine

Exemples de TLD

- générique : .com .org .net .name . . .
- pays : .fr .uk .tv .uk .us .eu . . .
- *sponsored : .edu .gov .int .museum .xxx . . .*

Modèle client-serveur

- Côté serveur
  BIND 9 majoritaire (Internet Software Consortium)
  Challengers : PowerDNS, Unbound, MS_DNS

- Côté client
  - Bibliothèque partagée *resolver* dans la *glibc*
  - Configuration via `/etc/resolv.conf`
    - serveurs à interroger (nameserver)
    - domaine de recherche par défaut (search)
  - Configuration des priorités
    - `/etc/hosts` est prioritaire sur DNS par défaut.
    - Pour affiner les priorités : `/etc/nsswitch.conf`

Clients DNS

109.4

- Client léger : `nslookup`
- Clients complets :
  - `dig` (dnsutils)
  - `host` (host)
- DNS et IPv6 ?
  - `host (-t A | -t AAAA | ) www.go6.net`
- Sans oublier...
  `ping` (/etc/hosts puis DNS)

Les enregistrements DNS

109.4

Résolution directe

- `SOA` : début de zone (Start Of Authority)
- `A` : adresse IPv4 (résolution directe)
- `AAAA` : adresse IPv6
- `CNAME` : nom canonique (pour un alias)
- `NS` : serveur de nom primaire (pour une zone)
- `MX` : serveur de mail (Mail eXchanger)
- `HINFO, SPF, ...` (cf RFC 1035)

Résolution inverse

- `PTR` : nom canonique (pour une adresse IPv4)

Exemples

```
host tickets.silecs.info
dig tickets.silecs.info
dig -t CNAME tickets.silecs.info
```

WHOIS - annuaire des adresses et domaines internet

- `whois <objet>` parmi
  - domaine DNS
  - serveur de noms (NS)
  - système autonome (ex. AS12322)
  - adresse IP $\longrightarrow$ AS
  - ... (18 types d'objet)

- Références
- RFC 954, RFC 3912 (cf Bortzmeyer)

Gestion des logs

Les logs

108.2

Tous les événements importants sont consignés dans `/var/log`.

- soit via `syslog` / `rsyslog`
- soit directement par les applications

le service (démon) : syslogd / rsyslog

- collecte les messages de différentes sources
- les analyse (légèrement) et les dispatche

Consultation des logs

- `dmesg` *(noyau : boot + modules)* + `echo 'hello' > /dev/kmsg`
- `last`, `lastlog` *(connexions utilisateurs)*
- `tail (-f)`, `multitail`
- tous les filtres texte : `less`, `grep`...

Évolutions de syslog *108.2*

- `syslog` : un standard BSD, normalisé (RFC 3164)
- Émergence de besoins plus poussés
  - des sources différentes : `syslog`, fichiers ...
  - des backends différents : MySQL, PostgreSQL ...
  - des filtres plus précis : hôtes, calculs, regexps ...
  - des communications sécurisées : fiables, chiffrées
- `syslog-ng` (Balabit, HU)
  - fichier de configuration spécifique
  - définition de modèles : source, destination, log, filtre
- `rsyslog` (Adiscon GmbH, DE)
  - fichier de configuration compatible syslog
  - remplace `syslog` dans Debian depuis Lenny (5.0)
  - architecture modulaire

Composition d'un message

- priorité : 0=debug ... 3=warning ... 5=crit ... 7=emerg
- service (*facility*) (auth mail kern local[0-7] ...)
- texte

Client CLI : `logger`

```
logger -p mail.info -t "essailog[$$]" "Bonjour monde"
```

toutes facilities *sauf kernel*

tester avec `auth` + `emergency` puis `auth` + `debug`

Fichier de configuration syslog

108.2

- sélecteur : <service>.<priorité>
- action : envoi vers
  - fichier, ex. `/var/log/messages`
  - terminal (ou pseudo-term), ex. `/dev/tty8`
  - machine distante (syslog), ex. `@loghost.localdomain`
  - utilisateurs, ex. `root,john` ou tout le monde, `*`
  - pipe **nommé**, ex. `|/var/spool/critMessages`

rsyslog - Travaux pratiques

108.2

Exo

1. Afficher les logs d'authentification sur la console 8.
2. Horodatage de `/var/log/syslog` toutes les 5 minutes.

Exo

1. Passer l'horodatage en format ISO + haute précision
2. Activer la centralisation des logs, en UDP (historique) puis en TCP
3. Ajouter un filtre pour extraire les logs CRON de `auth.log`

Rotation des logs : logrotate *108.2*

- En pratique
  - commande `logrotate` lancée par `cron` (daily)
  - OU forçage manuel `logrotate -f <fichier>`
  - configuration : `/etc/logrotate.conf` et `/etc/logrotate.d/*`
  - état : `/var/lib/logrotate/status`

- Configuration
  - période : `daily, weekly, monthly`
  - OU taille : `size`
  - archivage : `rotate, compress, delaycompress, olddir` ...
  - nommage : `dateext, dateformat` ...
  - scripts : `prerotate, postrotate` et `firstaction, lastaction`

Systemd journal - 1

108.2

- Configuration
  - `/etc/systemd/journal.conf`
  - ex. `Storage = auto | persistent | volatile`
- Stockage des logs
  - `/run/systemd/journal/*` volatil
  - `/var/log/journal/*` pérenne
  - stockage binaire (métadonnées) + texte
- Exercice
  - trouver le démon "journald"
  - trouver ses fichiers, sockets...

Systemd journal - 2

- Consultation
  - commande `journalctl`
  - utilisateur `root` pour les journaux système
- Paramètres
  - reboots : `journalctl -b 0, -b -1 ...`
  - horodatage : `journalctl --since="2015-05-30 12:34:56" --until...`
  - formatage : `journalctl -o short, short-iso, verbose, json...`
  - unité : `journalctl --unit=ssh`
  - processus : `journalctl _PID=12345`

Analyse automatique des logs *108.2*

- `logcheck` (par défaut sous Debian)
  - analyse des logs à intervalles réguliers (1 heure)
  - détection de “traces suspectes”
  - envoi par mail ou vers un fichier, *pipe* ...
  - 3 profils : *paranoid*, *server*, *workstation*
  - 3 niveaux : *system*, *security*, *attack*
- `logwatch` (par défaut sous RedHat)
- pour aller plus loin : IDS (Intrusion Detection Systems)
  OSSEC, Prelude

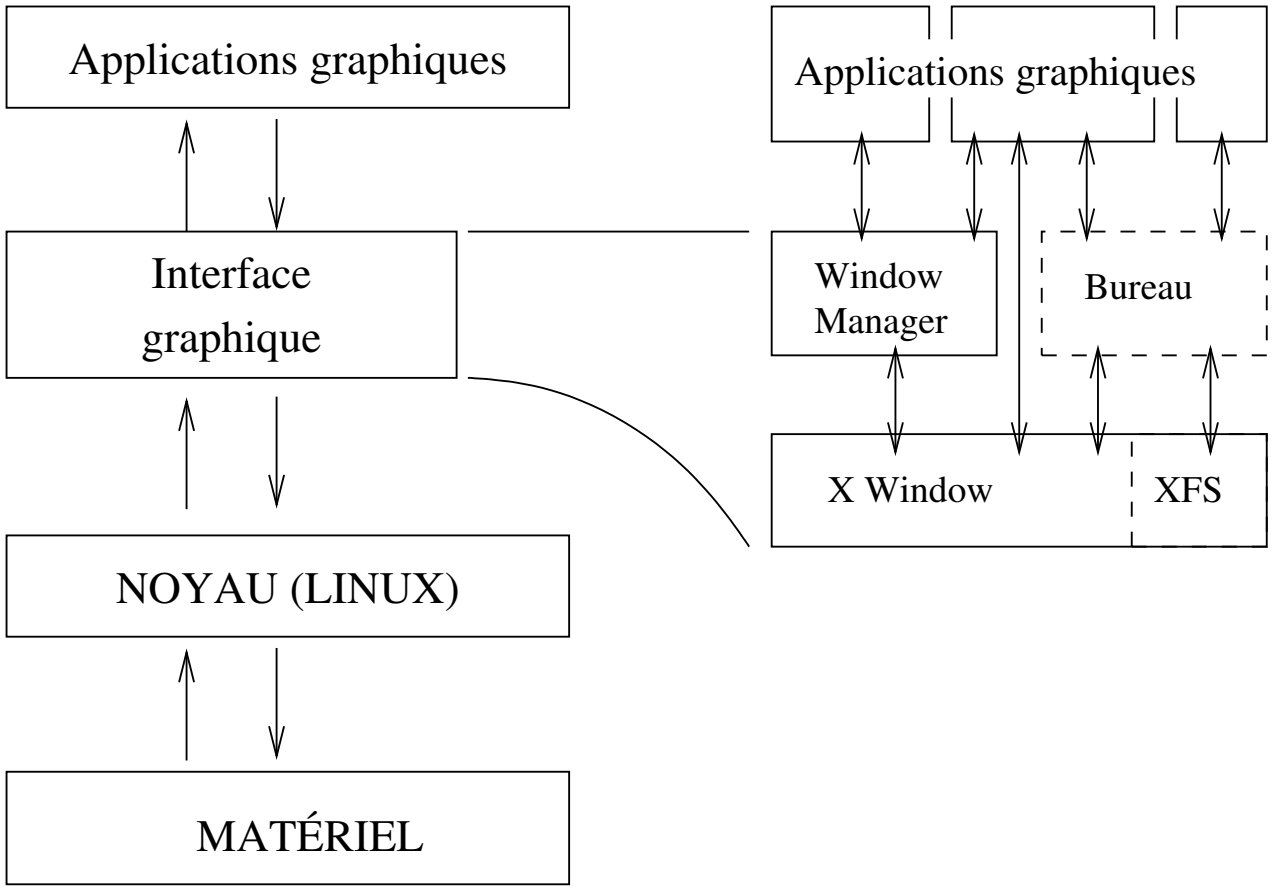
- `multitail`
  - suivi de fichiers multiples
  - agrégation de fichiers successifs
  - filtres de recherche et d'affichage

Pour aller plus loin...

- LIRE (LogReport)
  - synthèses et statistiques
  - analyse cross-fichiers
- LogAnalyzer (Adiscon)
  - interface web (PHP)

X11
(X Window System)

Le système de fenêtrage X-Window (X11)



- Système standard sur tous les Unix (sauf Mac OS X)
- Transparence réseau (presque) totale
- Architecture client-serveur !

X Window : historique

106.1

- Historique
  - juin 1984 : X1, MIT
  - jan. 1985 : X6, première version diffusée (propriétaire)
  - sep. 1985 : X9, couleur, licence MIT
  - sep. 1987 : X11, protocole courant
  - mai 1994 : X11R6
  - déc. 2005 : X11R6.9 + X11R7
  - oct. 2009 : X11R7.5
- Implémentations libres
  - XFree86 : 1992 - 2003 (dissolution de l'équipe) - 2008 ...
  - X.org : fork en 2004 (XFree86 4.4rc2), plus dynamique

X11 en pratique

106.1

- Configuration
  - Fichier `/etc/X11/xorg.conf`
  - Optionnel depuis 1.7.0
  - `X -configure` → `xorg.conf.new`
- Lancement
  - Manuel : `/usr/bin/X` pour tester
  - Via `xdm`... (service) en temps normal
- Logs
  - `/var/log/X.?.log`

X11 : principales composantes

106.1

- Serveur X (`/usr/bin/X`)
- Gestionnaire de session X (X Display Manager)
  ex. xdm, kdm, gdm, slim...
- Bureau graphique (optionnel)
  ex. Gnome, KDE, XFCE...
- Gestionnaire de fenêtres (Window manager)
  ex. metacity, kwm, xfwm4, twm, awesome...
- Console / émulateur de terminal
  ex. xterm, mlterm, xfce4-terminal...

X.org : un système très modulaire

- Diagnostic
  - Répertoire `/usr/lib/xorg/modules`
  - Commande `xdpyinfo`

- Exemples
  - Pilotes de cartes video (`drivers`)
    - Nvidia : `nv, nvidia, nouveau`
    - `intel`
    - `ati`
  - Pilotes de périphériques d'entrée (`input`)
    - standard : `kbd`, `mouse`
    - `synaptics`
    - `wacom`
  - Extensions
    - `libdri` : Direct Rendering Infrastructure...
    - `libglx` : MesaGL / OpenGL pour X...

Concepts et commandes X11 *106.1*

- Évènements X11 (clavier, souris, logiciel)
  `xev` : tester les entrées
- Propriétés et informations
  - Commande `xwininfo`
  - Commande `xprop`
- Ressources X
  - Commande `xrdb (-query -all)`
  - Fichiers `/.Xdefaults` et `/etc/X11/Xresources/*`
- Contrôle des fenêtres
  Commande `xkill`