Sécurité et Réseaux
Licence 3 Informatique

Cours 3: Introduction

Osman SALEM
Maître de conférences - HDR
osman.salem@parisdescartes.fr

MATHÉMATIQUES ET INFORMATIQUE
Sciences
Université de Paris

1

Machines à rotor

- Les machines à rotor ont été les machines les plus répandues, surtout pendant la seconde guerre mondiale
  - Enigma
- Elles implémentaient une méthode de substitution vraiment complexe et variable
- Avec une série de cylindres, chacun définissant une substitution, qui fait tourner et change après chaque lettre chiffrée
- Avec 3 cylindres, 26^3=17576 alphabets

2

Machines à rotor

- Les travaux commencés par les polonais

https://fr.wikipedia.org/wiki/Enigma_(machine)

Dès 1931, le Service français de renseignement (surnommé le « 2e Bureau ») était parvenu à recruter une source (Hans-Thilo Schmidt) au sein même du bureau du chiffre du ministère de la Reichswehr. Il obtient de lui de premières copies de la documentation ; il les proposa à l'Intelligence Service britannique, qui se montra sceptique, et au service polonais, qui fut très intéressé. Une coopération s'instaura, qui allait durer jusqu'en 1939. Les Français continuèrent de fournir de nouveaux renseignements obtenus de la même source, et les Polonais montèrent une équipe qui parvint à reproduire la machine à partir de la documentation de plus en plus précise qui leur parvenait.

- Transmis à l'ambassade de Grande-Bretagne
  - deux jours avant l'invasion par l'Allemagne
- Les informations obtenues donnaient un net avantage dans la poursuite de la guerre
- Le conflit en Europe s'est considérablement écourté grâce à la cryptanalyse du code allemand

3

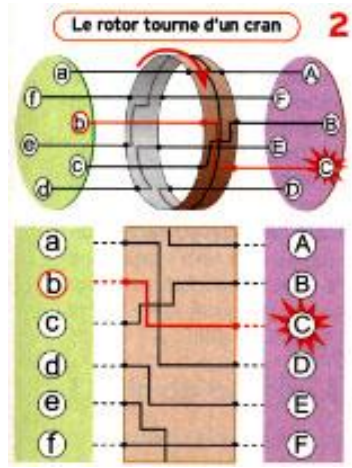
Enigma Machine





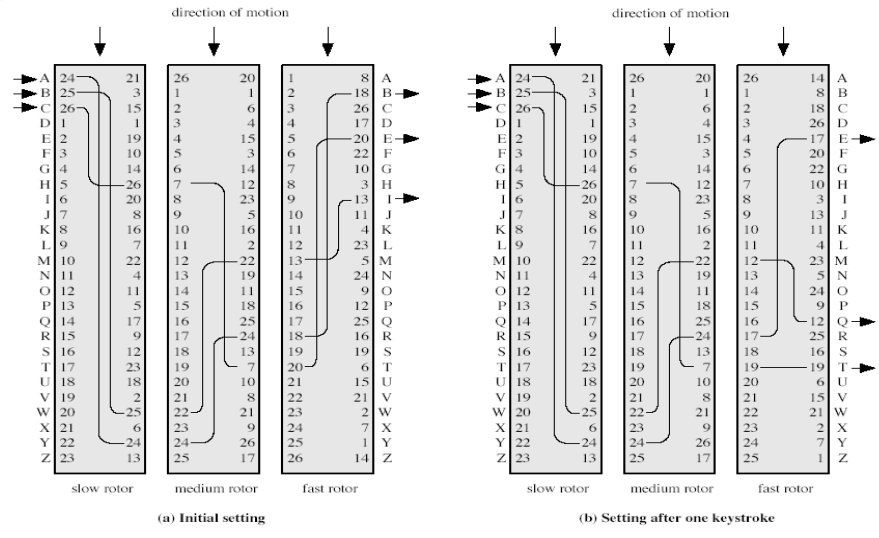
4

Un exemple de machine à rotor



5

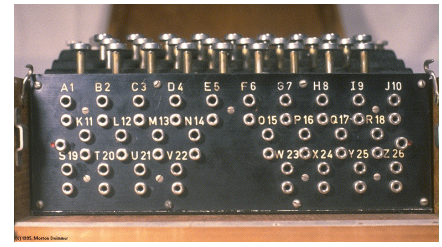
Un exemple de machine à rotor



6

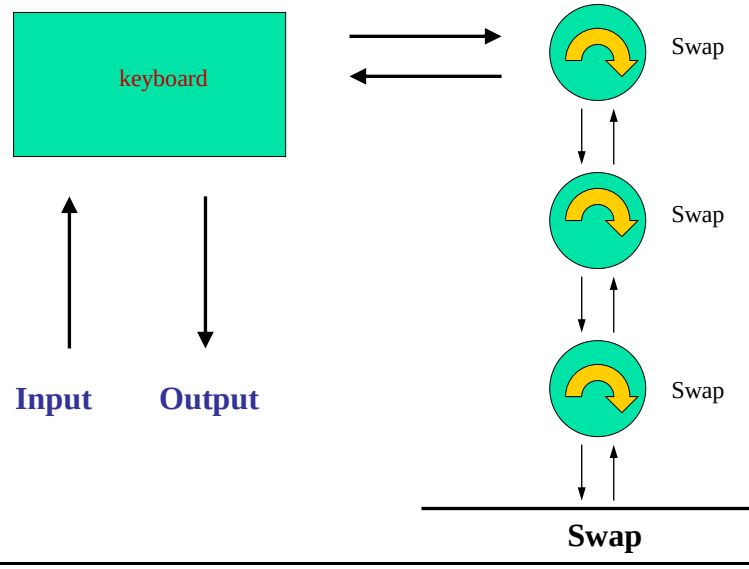
Plug Board

- Changement de lettres avant et après le passage par les cylindres



7

Procédure



8

Stéganographie

- Une alternative au chiffrement ; (du grec steganos, couvert et graphein, écriture)
- Cacher l'existence même des messages
- Utiliser un sous-ensemble de lettres ou de mots dans un message plus long
- Utiliser un pixel précis dans une séquence d'images vidéo…
- Inconvénients
  - Énorme overhead pour cacher peu d'informations

9

Intérêts de la stéganographie

- Communiquer en toute liberté même dans des conditions de censure et de surveillance
- Protéger ses communications privées là où l'utilisation de la cryptographie n'est normalement pas permise ou soulèverait des suspicions
- Contrebalancer toutes les législations ou barrières possibles empêchant l'usage de la cryptographie
- Publier ouvertement (mais à l'insu de tous) des informations qui pourront ensuite être révélées et dont l'antériorité sera incontestable et vérifiable par tous

10

Stéganographie (exemple 1)

- **Alfred de Musset écrit à George Sand :**

Quand je vous jure, hélàs, un éternel hommage
Voulez-vous qu'un instant je change de langage
Que ne puis-je, avec vous, goûter le vrai bonheur
Je vous aime, ô ma belle, et ma plume en délire
Couche sur le papier ce que je n'ose dire
Avec soin, de mes vers, lisez le premier mot
Vous saurez quel remède apporter à mes maux.

- **George Sand a répondu :**

Cette grande faveur que votre ardeur réclame
Nuit peut-être à l'honneur mais répond à ma flamme.

*George Sand est le pseudonyme d' Amandine Aurore Lucile Dupin

11

Stéganographie (exemple 2)

- http://lwh.free.fr/pages/algo/crypto/steganographie.htm

12

Stéganographie (exemple 3)

- Cacher un fichier text dans un autre
  - echo how are you doing > file.txt
  - echo password > file.txt:hidden.txt
  - dir file.txt
  - notepad file.txt:hiddent.txt
- Cacher une image dans un fichier text
  - type image.jpg>file.txt:image.jpg
  - Mspaint file.txt:image.jpg

13

Fonction de hachage : intégrité

- Fonction mathématique qui, à un ensemble de nombres en entrée, fait correspondre un ensemble de nombres de cardinal plus petit en sortie ;
  - La modification d'un élément en entrée engendre une modification de sa fonction de hachage en sortie.

M	h(M)
remarquez la fin de cette ligne,	4b:2c:65:c0:e8:ee:95:5f:eb:05:9f:3c:6d:2f:2f :0f:9a:26:00:b7
remarquez la fin de cette ligne!	9e:e9:22:99:11:7f:41:23:7c:ce:38:3a:d8:05:18 :0c:4a:fc:ab:4c
??	75:cc:4b:e0:a9:7c:76:34:78:58:bf:04:db:3b:90 :2b:45:6a:2b:c0

- Propriétés
  - Deux messages différents ont deux empreintes différentes
  - Connaissant h(M), il est impossible de trouver M

14

Fonction de hachage

- Fonctions de hachage :
  - si *H(x)* est une telle fonction, pour tout y donné, il doit être **quasi-impossible** de trouver un x tel que *H(x)=y* ;
  - si *y=H(x)*, et *x' ≈ x* à un tout petit détail près (ex : changer 1 bit), on doit avoir *y'=H(x')* très ≠ de y
  - Collision : si *H(y)=H(x)* sachant *x ≠ y*
  - Problème: hachage doit avoir un petit nb de collisions
  - Ex de fonctions de hachage: MD5, SHA-1, MD4, RIPEMD-160

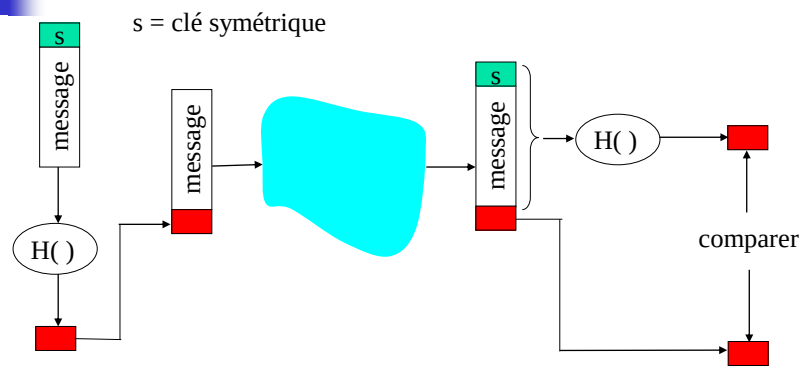
15

Hash Function Algorithms

- **MD5**
  - Digest sur 128-bit
- **SHA-1**
  - Digest sur 160-bit

16

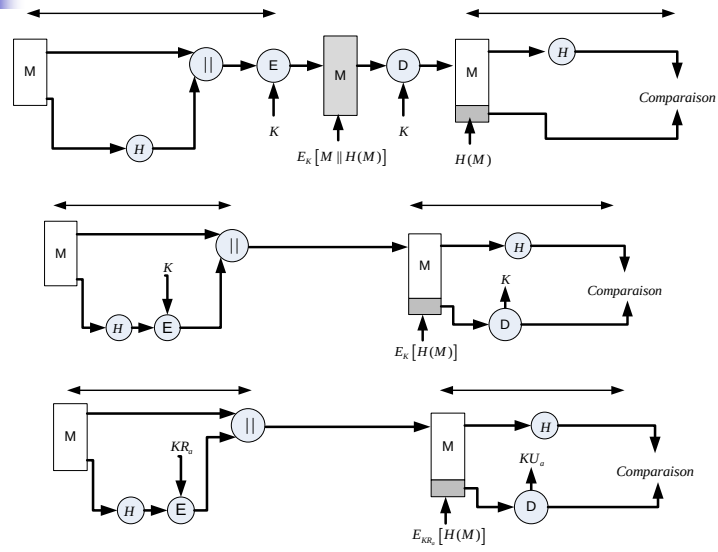
MAC: Message Authentication Code



- Son rôle est d'assurer l'intégrité d'un message ***m***
- *S=H(m,s),* ***m*** est le message et ***s*** est la clé symétrique
- MAC (Message Authentication Code): ne fournit pas la propriété de non-répudiation

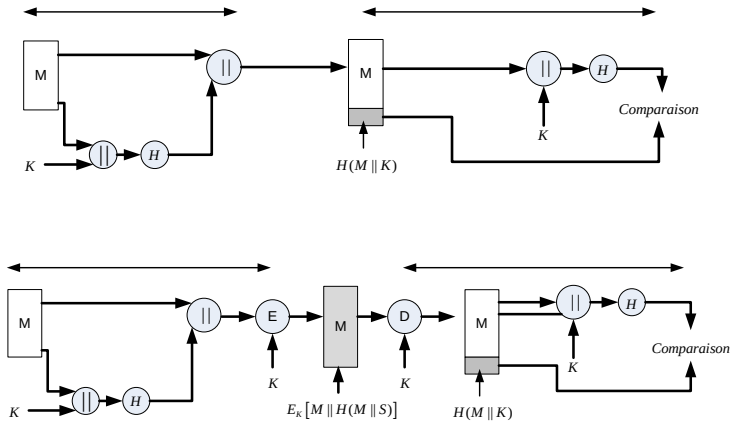
17

Utilisation d'une fonction de hachage (1)



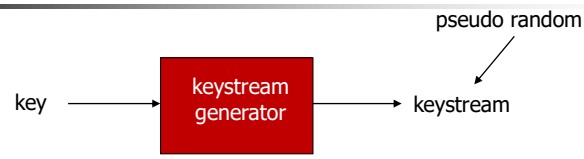
19

Utilisation d'une fonction de hachage (2)



20

Chiffrement en continu



- Combinaison de bit de keystrem avec le bit du message en claire
- m(i) = i^{ième} bit du message
- ks(i) = i^{ième} bit de la clé
- c(i) = i^{ième} bit du message chiffré
- c(i) = ks(i) ⊕ m(i) (⊕ = ou exclusif)
- m(i) = ks(i) ⊕ c(i)

21

Chiffrement par bloc

- Le message est divisé en bloc de k bits (64-bit par bloc).
- Chiffrement du k-bit du bloc en claire texte à k-bit bloc chiffré

Exemple avec k=3:

input	output	input	output
000	110	100	011
001	111	101	010
010	101	110	000
011	100	111	001

Quel est le message chiffré correspondant à 010110001111 ?

22

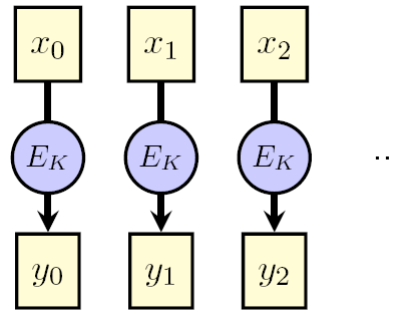
Chiffrement par bloc

- **Les quatre principaux modes de chiffrement**
  - ECB Electronic CodeBook
  - CBC Cipher Block Chaining
  - CFB Cipher FeedBack
  - OFB Output FeedBack

23

ECB : Electronic Code Book

- $C_i = E_k(M_i)$



- Le mode ECB n'assure aucune sécurité : ne pas l'utiliser.

24

Chiffrement par bloc (ECB)

Supposant le codage suivant:

input	output	input	output
A: 000	110	I: 100	011
K: 001	111	E:101	010
N: 010	101	L:110	000
O: 011	100	S:111	001

Quel est le message chiffré correspondant à « le soleil »

L	E	S	O	L	E	I	L
110	101	111	011	110	101	100	110

Texte chiffré:

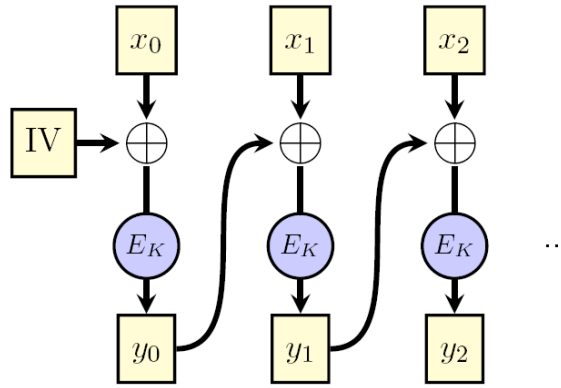
000	010	001	100	000	010	011	000
A	N	K	I	A	N	O	A

Est-ce la fréquence de répétition des lettres est toujours la même avant et après le chiffrement ?

25

CBC : Cipher Block Chaining

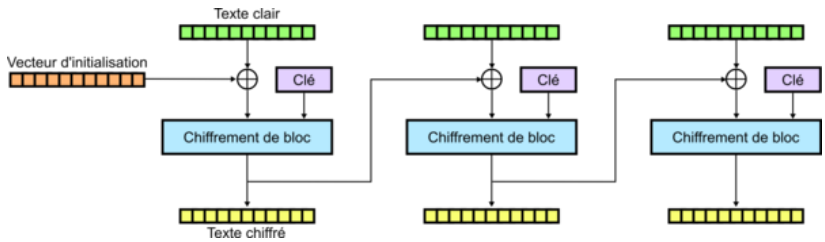
- $C_0 = IV$
  $C_i = E_k(M_i \oplus C_{i-1})$



26

Mode opératoire: Cipher Block Chaining

- Le message est découpé en blocs de taille fixe
- Chaque bloc est chiffré de manière corrélée avec le bloc précédent en utilisant l'opération OU eXlusif (XOR (⊕)) entre le bloc de message i (M_i) et le résultat du chiffrement du bloc de Message M_{i-1}
  - à l'étape i,
    - On calcule: $M_i \oplus C_{i-1}$
    - Puis on chiffre le résultat: $C_i = E(M_i \oplus C_{i-1})$
    - Et on transmet C_i
  - pour l'étape 1 :
    - On introduit une valeur d'initialisation (appelé seed ou initialisation Vector (IV)) pour effectuer le premier XOR.



27

Chiffrement par bloc (CBC)

Supposant le codage suivant:

input	output	input	output
A: 000	110	I: 100	011
K: 001	111	E:101	010
N: 010	101	L:110	000
O: 011	100	S:111	001

Par exemple:
IV: 000

Quel est le message chiffré correspondant à « le soleil »

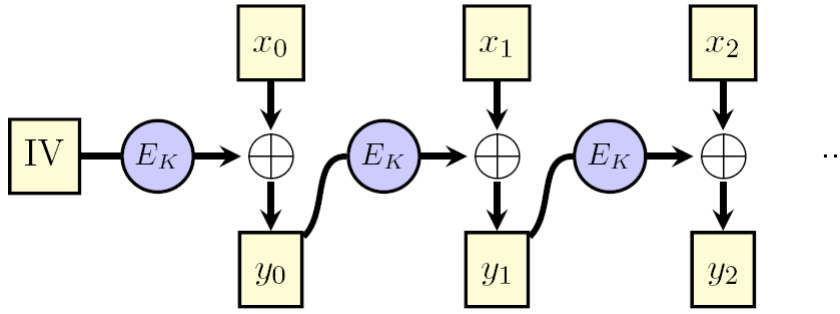
	L	E	S	O	L	E	I	L
	110	101	111	011	110	101	100	110
IV:	000	000	010	010	111	111	101	111
Xor:	110	101	101	001	001	010	001	001
Texte chiffré	000	010	010	111	111	101	111	…

Est-ce la fréquence de répétition des lettres est toujours la même avant et après le chiffrement ?

30

CFB : Cipher FeedBack

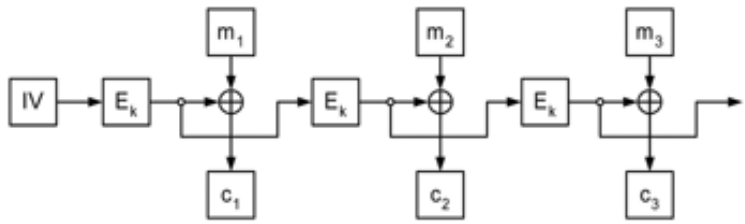
- $C_0 = IV$
  $C_i = M_i \oplus E_k(C_{i-1})$



31

OFB : Output FeedBack

- $Z_0 = IV$
  $Z_i = E_k(Z_{i-1})$
  $C_i = M_i \oplus Z_i$



32

Distribution des clés

- Les algorithmes de chiffrement symétriques nécessitent le partage d'une clé secrète
- Il faut donc assurer le transport sûr de cette clé
- Si la clé est compromise lors de la phase de distribution, toutes les communications le seront !

33

Distribution des clés

- Un utilisateur souhaitant communiquer avec plusieurs autres en assurant de niveaux de confidentialité distincts doit utiliser autant de clés qu'il a d'interlocuteurs

- Pour un groupe de N personnes utilisant un cryptosystème à clés secrètes, il est nécessaire de distribuer un nombre de clés égal à:

$$N * (N-1) / 2$$

34

Schémas de distribution de clés

- Plusieurs variantes pour 2 partenaires
  - A choisit une clé et la transmet physiquement à B (valise diplomatique par exemple)
  - Une tierce partie (de confiance) C choisit un clé et assure la distribution à A et B
  - Si A et B ont déjà partagé une clé auparavant, ils peuvent utiliser l'ancienne clé pour chiffrer une nouvelle
  - Si A et B ont des communications sûres avec une tierce partie C, C peut relayer les clés entre A et B

35

Kerberos

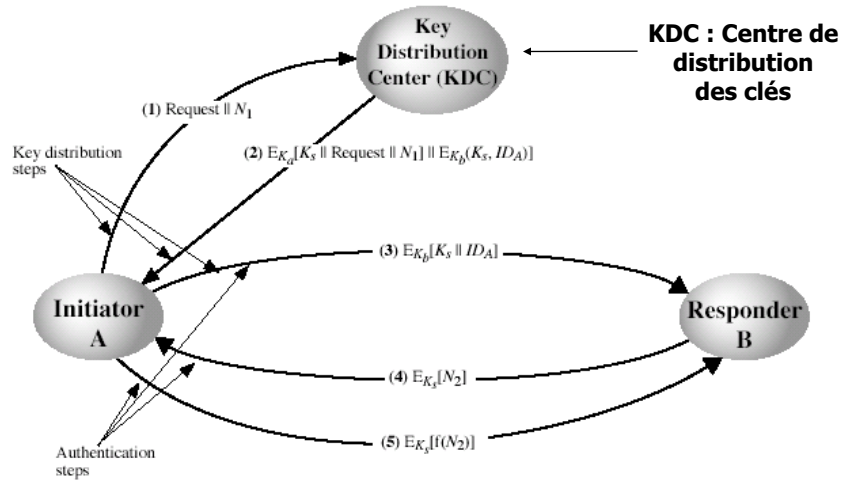
36

Généralités

- Principes
  - Basé sur la notion de « Ticket »
  - Cryptographie à clé secrète (symétrique)
  - Authentification mutuelle
  - Tickets limités dans le temps
  - Mécanismes anti-rejeu (horodatage)
  - Pas de transmission de mot de passe sur le réseau
- Kerberos V5
  - Standards IETF : RFCs 1510 et 1964

37

Scénario de distribution



38

Scénario de distribution

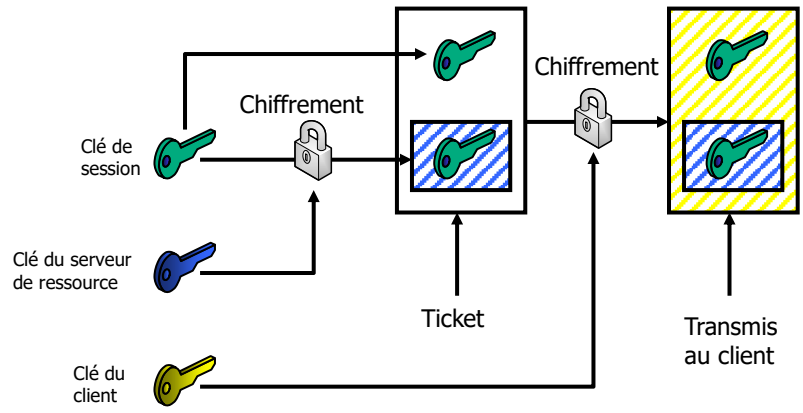
L'utilisateur A (resp. B) dispose d'une clé K_a (resp K_b) avec le KDC

La requête envoyée par A au KDC contient les deux identités ID_a et ID_b

Le nombre aléatoire N_1 (resp. N_2) est destiné à la lutte anti rejeu

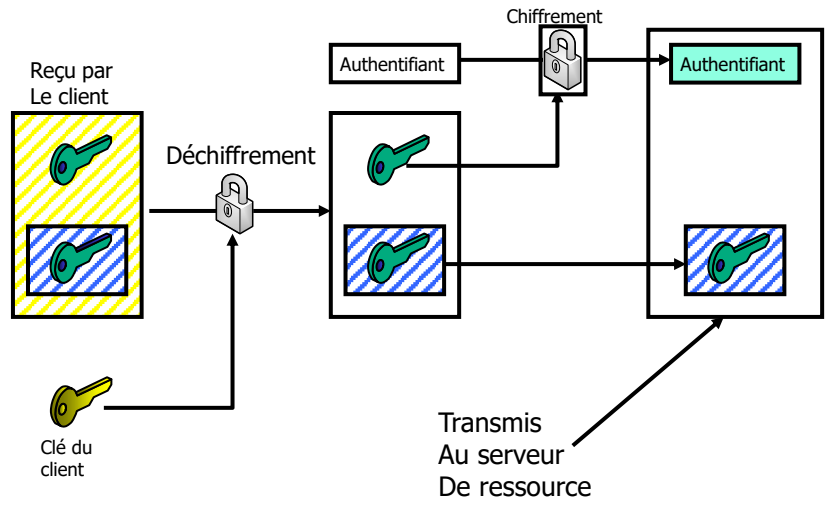
39

Génération d'un ticket



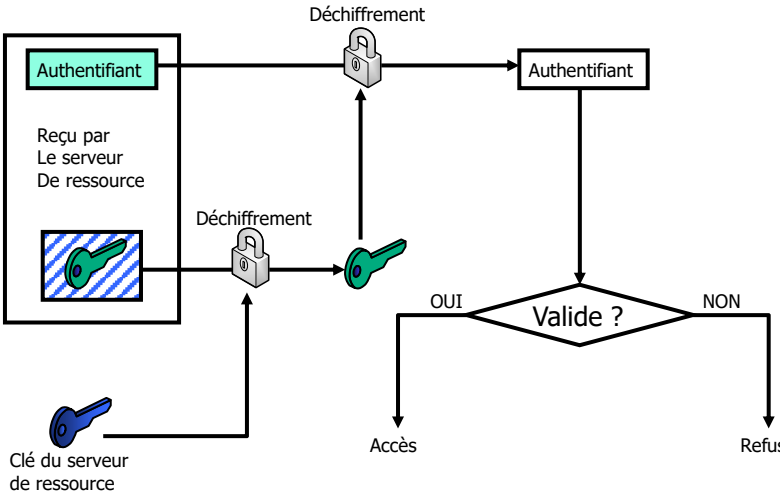
45

Traitement par le client



46

Traitement par le serveur



47

Accès à distance: FTP, telnet et SSH

- telnet 192.168.0.13
  - Login:
  - Password:
  - Escape character is '^]'.

48

Accès à distance :FTP, telnet et SSH

- ftp (serveur)
  - Serveur: apt install vsftpd
    - service vsftpd start
  - nano /etc/vsftpd/vsftpd.conf
    - #write_enable=YES => uncomment this option

49

Accès à distance :FTP, telnet et SSH

SSH : Secure-Shell

- C'est un protocole sécurisée pour une connexion à distance
  - mot de passe n'est pas transmis en clair sur le réseau
  - session est chiffrée.
- SSH remplace *Telnet* et *FTP*
- Le protocole SSH est basé sur une architecture client/serveur
  - Un utilisateur qui veut se connecter exécute la commande «ssh» sur sa machine locale
  - Le client va se connecter à un **serveur** SSH sur une machine distante
  - Un serveur (sshd) est à l'écoute
    - port 22 et protocole TCP
  - La connexion est sécurisée, tout ce qui suit est chiffré

50

Accès à distance :FTP, telnet et SSH

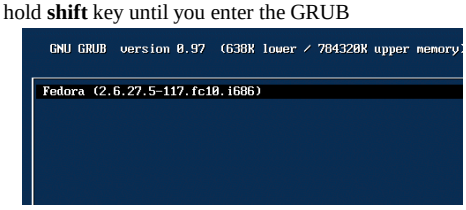
- ssh
  - ssh [*user@*]*hostname*

[user@host]$ ssh login@saphyr.ens.math-info.univ-paris5.fr
password: ********
[user@saphyr]$ exit
logout

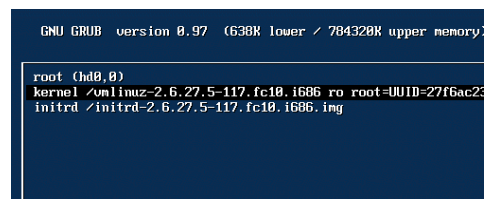
51

RESET root Password

- Boot up
  - press and hold **shift** key until you enter the GRUB



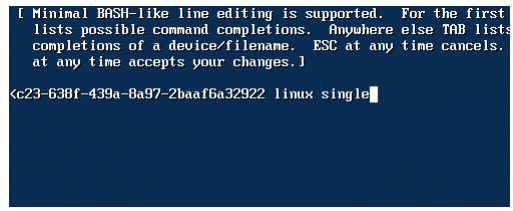
  - Press e (to edit) the kernel line
    - **'kernel /vmlinuz-2.6.27.5-117.fc10.i686 ro root=UUID=27f6ac23-638f-439a-8a97-2baaf6a32922 rhgb quiet'**



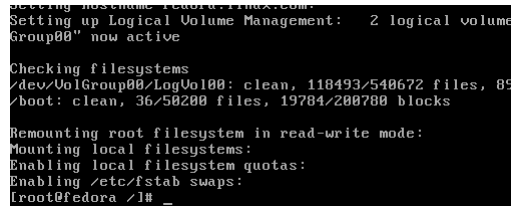
52

RESET root Password

- Add the word ***single*** (runlevel in single user mode) ou 1 ou S



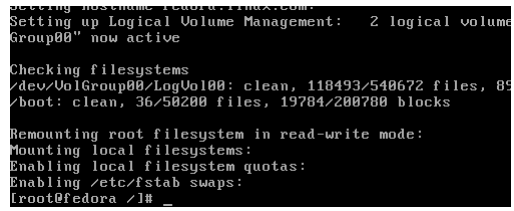
- Press b (to boot) in single mode



53

RESET root Password

- Press b (to boot) in single mode



- **[root@fedora /]# passwd**

Changing password for user root.
New UNIX password:
BAD PASSWORD: it is too simplistic/systematic
Retype new UNIX password:
passwd: all authentication tokens updated successfully.

- **If error authentication token lock busy**
  - mount –o remount, rw /

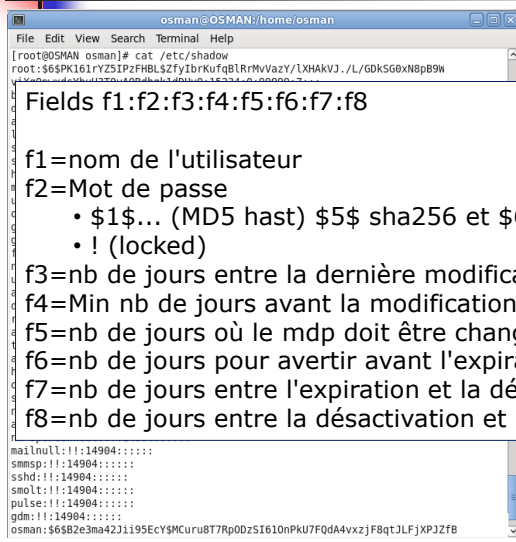
54

RESET root Password

- Debian variations
  - Instead of adding "single" or "1" to your kernel command line, add "**init=/bin/bash rw**"
  - # passwd
  - # reboot

55

/etc/shadow



Fields f1:f2:f3:f4:f5:f6:f7:f8

f1=nom de l'utilisateur
f2=Mot de passe
- 1... (MD5 hast) 5 sha256 et 6 sha512
- ! (locked)

f3=nb de jours entre la dernière modification et 1/1/1970
f4=Min nb de jours avant la modification
f5=nb de jours où le mdp doit être changer
f6=nb de jours pour avertir avant l'expiration
f7=nb de jours entre l'expiration et la désactivation (période de grâce)
f8=nb de jours entre la désactivation et 1/1/70

56

/etc/shadow

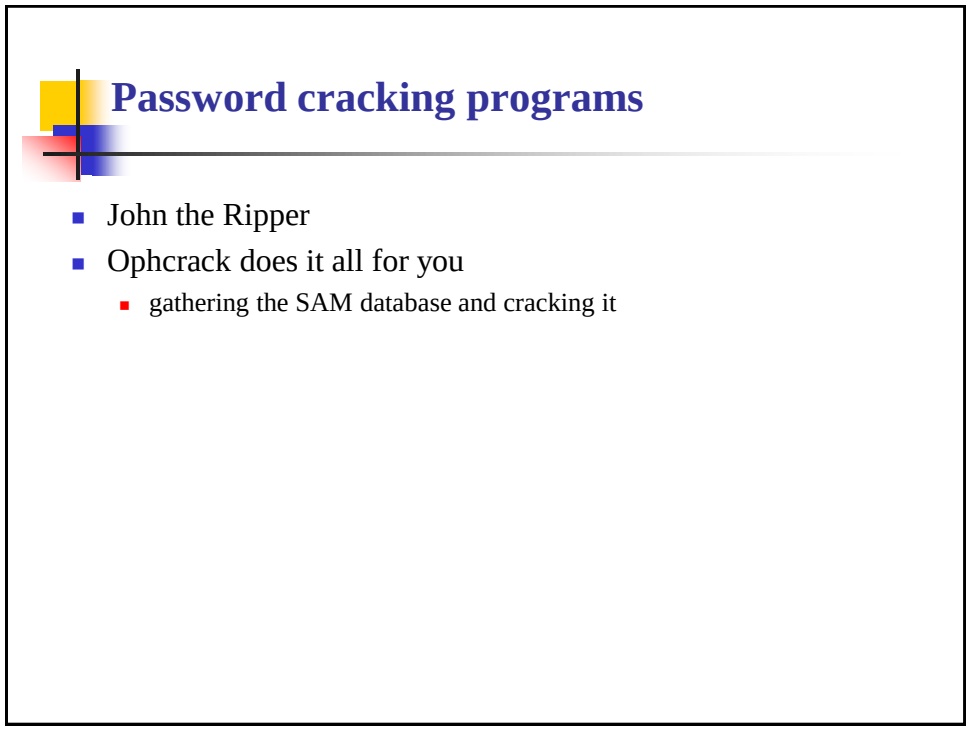
- Password
  - id= 6 (sha512)
    - 1: md5
    - 2a: blowfish
    - 5: sha256
    - 6:sha512
  - Salt= QR3drPrQ
  - Hash= Jlo1PKy ….
- Authentication
  - Users enter his password
  - System gets the salt from /etc/shadow
  - System is hashing the password with the salt
  - System compares the results with hash stored in /etc/shadow

57

Password Cracking

- You need the hashed password file
  - /etc/shadow for UNIX
  - The SAM database in Windows
- Then perform dictionary or brute-force attacks on the file
  - Brute force
  - Dictionary attack
  - Salt prevents Rainbow tables attack

58


Password cracking programs
John the Ripper
Ophcrack does it all for you
gathering the SAM database and cracking it

59