

# La Carte à Microprocesseur

## Un système embarqué en plein essor

**Tegawendé F. Bissyandé**

tegawende.bissyande@fasolabs.org

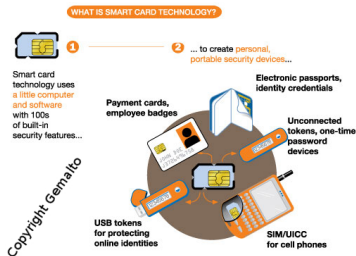
*Cours préparé pour*

L'Université Ouaga I Pr. Joseph Ki-Zerbo(UFR SEA)

24 Février 2015

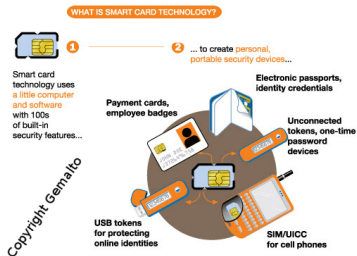
## Qu'avons-nous appris jusqu'à présent? :

- Caractéristiques physiques & électriques d'une carte à puce
- Dialogue avec une carte à puce
- Convention de nommage des fichiers de données (MF, DF, EF)
- Organisations des données dans les fichiers (linéaire fixe/variable, cyclique, transparent)
- Commande de gestion des fichiers (SELECT, READ, WRITE, ...)



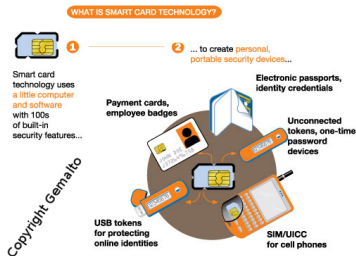
## Qu'avons-nous appris jusqu'à présent? :

- Caractéristiques physiques & électriques d'une carte à puce
- Dialogue avec une carte à puce
- Convention de nommage des fichiers de données (MF, DF, EF)
- Organisations des données dans les fichiers (linéaire fixe/variable, cyclique, transparent)
- Commande de gestion des fichiers (SELECT, READ, WRITE, ...)



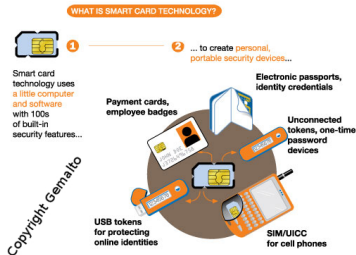
## Qu'avons-nous appris jusqu'à présent? :

- Caractéristiques physiques & électriques d'une carte à puce
- Dialogue avec une carte à puce
- Convention de nommage des fichiers de données (MF, DF, EF)
  - Organisations des données dans les fichiers (linéaire fixe/variable, cyclique, transparent)
  - Commande de gestion des fichiers (SELECT, READ, WRITE, ...)



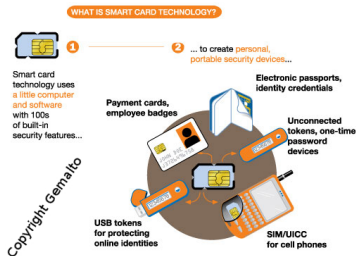
## Qu'avons-nous appris jusqu'à présent? :

- Caractéristiques physiques & électriques d'une carte à puce
- Dialogue avec une carte à puce
- Convention de nommage des fichiers de données (MF, DF, EF)
- Organisations des données dans les fichiers (linéaire fixe/variable, cyclique, transparent)
- Commande de gestion des fichiers (SELECT, READ, WRITE, ...)



## Qu'avons-nous appris jusqu'à présent? :

- Caractéristiques physiques & électriques d'une carte à puce
- Dialogue avec une carte à puce
- Convention de nommage des fichiers de données (MF, DF, EF)
- Organisations des données dans les fichiers (linéaire fixe/variable, cyclique, transparent)
- Commande de gestion des fichiers (SELECT, READ, WRITE, ...)



## Serious Online Security

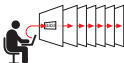
How **smart cards** provide solutions to software-based security problems

### PROBLEMS WITH SOFTWARE-BASED SECURITY APPLICATIONS

Single sign-on, password vaults and other software-based security applications consolidate multiple passwords, but leave them vulnerable.

1

If passwords are the only protection used, then anyone obtaining the password can gain access to the system.



2

A master password is vulnerable to...



And a master password is static: once stolen, it can be used anytime.

3

Insiders and IT contractors can compromise master password files.



### SOLUTIONS PROVIDED BY SMART CARDS



A smart card is a small computer that is separate from the desktop, and is used as part of the authentication process.

1

You must physically have the smart card or token for network or information access.



2

A smart card is invulnerable to keyboard logging because security key calculations are done by the computer inside it; only encrypted information passes through the computer.



3

Since insiders cannot log in without the token, and there is no master password, smart cards are virtually impossible to attack.

Smart card technology is already at the heart of security systems for billions of dollars of individual transactions worldwide in mobilecom, banking, cable and satellite TV, and the U.S. Department of Defense.



## Copyright Gemalto

Marketing de base :)

## A quels niveaux ?

- physique  
*(besoin de matériel d'accès aux données)*
- du modèle de système de fichier  
*(condition d'accès)*
- de la carte  
*(identification mutuelle entre la carte et l'application qui l'utilise)*
- au niveau des fichiers  
*(Mot de passe ou code PIN pour accéder à un fichier)*



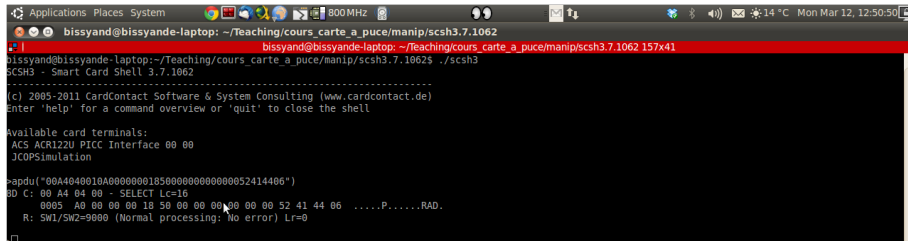
## Garder en mémoire...

- Sécurité de la carte à puce : ensembles de moyens très efficaces
- Application mal implémentée  $\Rightarrow$  carte défaillante



## Commande SELECT

- Sélectionner un fichier ou un répertoire
- Précède généralement les commandes de lecture/écriture
- Fichier/répertoire sélectionné  $\Rightarrow$  Fichier/répertoire courant
- Code de l'Instruction : A4



```
bissyand@bissyande-laptop: ~/Teaching/cours_carte_a_puce/manip/scsh3.7.1062
bissyand@bissyande-laptop: ~/Teaching/cours_carte_a_puce/manip/scsh3.7.1062 157x41
bissyand@bissyande-laptop:~/Teaching/cours_carte_a_puce/manip/scsh3.7.1062$ ./scsh3
SCSH3 - Smart Card Shell 3.7.1062
-----
(c) 2005-2011 CardContact Software & System Consulting (www.cardcontact.de)
Enter 'help' for a command overview or 'quit' to close the shell

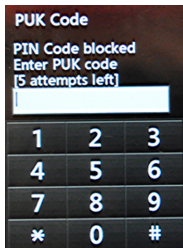
Available card terminals:
ACS ACR122U PICC Interface 00 00
JCOPSimulation

> apdu("00A4040010A000000018500000000000052414406")
BD C: 00 A4 04 00 - SELECT Lc=16
      0005 A0 00 00 00 18 50 00 00 00 00 00 00 52 41 44 06 .....P.....RAD.
      R: SW1/SW2=9000 (Normal processing: No error) Lr=0
```

- READ/WRITE/UPDATE/ERASE BINARY  $\rightarrow$  fichier à structure transparente
- READ/WRITE/UPDATE/APPEND RECORD  $\rightarrow$  fichier à structure linéaire fixe, variable ou cyclique

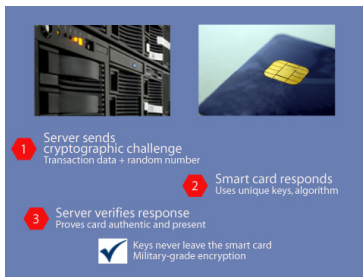
## Commande VERIFY

- Comparaison, au niveau de la carte, d'un "mot de passe" avec des infos de référence contenues dans la carte
- Certaines commandes d'accès à des fichiers ne sont possibles que si VERIFY a été un succès.
- Après retrait de la carte, ou un reset, le caractère réussi de VERIFY est annulé  $\Rightarrow$  pas d'échange de carte possible...
- Possibilité d'inscrire dans la carte le nombre de tentatives ratées pour la bloquer (3 mauvais codes PIN, vous savez!!)
- Code de l'Instruction : 20



## Commande INTERNAL AUTHENTICATE

- Première vraie fonction sécuritaire de la carte à puce
- Utilisation de fonctions cryptographiques
- Authentification de la carte vis-à-vis de l'application pilotant le lecteur
- Pas de transfert de mot de passe en clair entre la carte et le lecteur
  - 1 l'application de lecteur génère un nombre aléatoire et l'envoie à la carte (facultativement : + algo + clé)
  - 2 la carte crypte le nombre aléatoire (facultativement : avec l'algo et la clé reçus)
  - 3 l'application du lecteur réalise la meme opération de calcul
  - 4 la carte renvoie au lecteur le résultat de son calcul. Après comparaison, le lecteur détermine l'authenticité de la carte
- Code de l'Instruction : 88



# Commandes relatives à la sécurité (3)

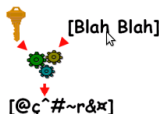
## Commande GET CHALLENGE

- Contexte particulier, car utilisé avec une autre commande : EXTERNAL AUTHENTICATE
- Demande la génération, par la carte, d'un défi (*challenge*): nombre aléatoire en fait
- challenge valable au moins pour la commande suivante
- Code de l'Instruction : 84

## Commande EXTERNAL AUTHENTICATE

- La réciproque de INTERNAL AUTHENTICATE pour boucler l'authentification mutuelle
- Authentifier l'application pilotant le lecteur vis-à-vis de la carte
- Après authentification, la carte peut laisser l'application accéder à des infos "sensibles"
- Recours à GET CHALLENGE car la carte ne peut pas générer spontanément un nombre aléatoire
- Code de l'Instruction : 82

N.B: l'algo et la clé connus seulement de l'application et de la carte



# Commandes relatives à la sécurité (3)

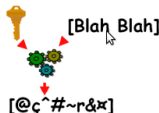
## Commande GET CHALLENGE

- Contexte particulier, car utilisé avec une autre commande : EXTERNAL AUTHENTICATE
- Demande la génération, par la carte, d'un défi (*challenge*): nombre aléatoire en fait
- challenge valable au moins pour la commande suivante
- Code de l'Instruction : 84

## Commande EXTERNAL AUTHENTICATE

- La réciproque de INTERNAL AUTHENTICATE pour boucler l'authentification mutuelle
- Authentifier l'application pilotant le lecteur vis-à-vis de la carte
- Après authentification, la carte peut laisser l'application accéder à des infos "sensibles"
- Recours à GET CHALLENGE car la carte ne peut pas générer spontanément un nombre aléatoire
- Code de l'Instruction : 82

N.B: l'algo et la clé connus seulement de l'application et de la carte



## Commande ENVELOPE

- Envoi d'une commande dans une commande
- Un APDU de commande ou un morceau est mis dans la partie données d'une autre APDU
- Possibilité de crypter entièrement l'APDU mis en données
- Peut être utilisé pour se prémunir contre l'espionnage des communications lecteurs - cartes
- Code de l'Instruction : C2



### Bref...

- La sécurité de la carte à puce est gérée à plusieurs niveaux
- Les commandes relatives à la sécurité sont nombreuses et couvrent plusieurs cas
- 1 vrai atout de la carte à puce: son moteur cryptographique intégré

## Commande ENVELOPE

- Envoi d'une commande dans une commande
- Un APDU de commande ou un morceau est mis dans la partie données d'une autre APDU
- Possibilité de crypter entièrement l'APDU mis en données
- Peut être utilisé pour se prémunir contre l'espionnage des communications lecteurs - cartes
- Code de l'Instruction : C2



## Bref...

- La sécurité de la carte à puce est gérée à plusieurs niveaux
- Les commandes relatives à la sécurité sont nombreuses et couvrent plusieurs cas
- 1 vrai atout de la carte à puce: son moteur cryptographique intégré

## Vocabulaire:

- **Texte en clair** : Données compréhensibles qu'un expéditeur veut envoyer à un destinataire
- Texte Chiffré / Cryptogramme : Données incompréhensibles obtenu à partir du texte en claire
- Chiffrement : Processus transformant le texte en clair en texte chiffré
- Déchiffrement/Décryptage : Processus inverse du chiffrement
- Cryptographie : science consistant à garder les messages secrets
- Cryptographe : Individu pratiquant de la cryptographie
- Cryptanalyste : Individu spécialisé dans le décryptage des messages chiffrés  
⇒ ennemi du cryptographe
- Cryptologie : Branche mathématique s'intéressant au cryptage
- Chiffrement continu / par blocs : selon que les données à crypter sont découpées en blocs ou pas

## Equations

- 1  $C = E(M) \leftarrow$  chiffrement
- 2  $D(C) = M \leftarrow$  déchiffrement
- 3  $D(E(M)) = M \leftarrow$  Equivalence



## Vocabulaire:

- **Texte en clair** : Données compréhensibles qu'un expéditeur veut envoyer à un destinataire
- **Texte Chiffré / Cryptogramme** : Données incompréhensibles obtenu à partir du texte en claire
- **Chiffrement** : Processus transformant le texte en clair en texte chiffré
- **Déchiffrement/Décryptage** : Processus inverse du chiffrement
- **Cryptographie** : science consistant à garder les messages secrets
- **Cryptographe** : Individu pratiquant de la cryptographie
- **Cryptanalyste** : Individu spécialisé dans le décryptage des messages chiffrés  
⇒ ennemi du cryptographe
- **Cryptologie** : Branche mathématique s'intéressant au cryptage
- **Chiffrement continu / par blocs** : selon que les données à crypter sont découpées en blocs ou pas

## Equations

- 1  $C = E(M) \leftarrow$  chiffrement
- 2  $D(C) = M \leftarrow$  déchiffrement
- 3  $D(E(M)) = M \leftarrow$  Equivalence

## Vocabulaire:

- **Texte en clair** : Données compréhensibles qu'un expéditeur veut envoyer à un destinataire
- **Texte Chiffré / Cryptogramme** : Données incompréhensibles obtenu à partir du texte en claire
- **Chiffrement** : Processus transformant le texte en clair en texte chiffré
- **Déchiffrement/Décryptage** : Processus inverse du chiffrement
- **Cryptographie** : science consistant à garder les messages secrets
- **Cryptographe** : Individu pratiquant de la cryptographie
- **Cryptanalyste** : Individu spécialisé dans le décryptage des messages chiffrés  
⇒ ennemi du cryptographe
- **Cryptologie** : Branche mathématique s'intéressant au cryptage
- **Chiffrement continu / par blocs** : selon que les données à crypter sont découpées en blocs ou pas

## Equations

- 1  $C = E(M) \leftarrow$  chiffrement
- 2  $D(C) = M \leftarrow$  déchiffrement
- 3  $D(E(M)) = M \leftarrow$  Equivalence

## Vocabulaire:

- **Texte en clair** : Données compréhensibles qu'un expéditeur veut envoyer à un destinataire
- **Texte Chiffré / Cryptogramme** : Données incompréhensibles obtenu à partir du texte en claire
- **Chiffrement** : Processus transformant le texte en clair en texte chiffré
- **Déchiffrement/Décryptage** : Processus inverse du chiffrement
- **Cryptographie** : science consistant à garder les messages secrets
- **Cryptographe** : Individu pratiquant de la cryptographie
- **Cryptanalyste** : Individu spécialisé dans le décryptage des messages chiffrés  
⇒ ennemi du cryptographe
- **Cryptologie** : Branche mathématique s'intéressant au cryptage
- **Chiffrement continu / par blocs** : selon que les données à crypter sont découpées en blocs ou pas

## Equations

- 1  $C = E(M) \leftarrow$  chiffrement
- 2  $D(C) = M \leftarrow$  déchiffrement
- 3  $D(E(M)) = M \leftarrow$  Equivalence

## Vocabulaire:

- **Texte en clair** : Données compréhensibles qu'un expéditeur veut envoyer à un destinataire
- **Texte Chiffré / Cryptogramme** : Données incompréhensibles obtenu à partir du texte en claire
- **Chiffrement** : Processus transformant le texte en clair en texte chiffré
- **Déchiffrement/Décryptage** : Processus inverse du chiffrement
- **Cryptographie** : science consistant à garder les messages secrets
- **Cryptographe** : Individu pratiquant de la cryptographie
- **Cryptanalyste** : Individu spécialisé dans le décryptage des messages chiffrés  
⇒ ennemi du cryptographe
- **Cryptologie** : Branche mathématique s'intéressant au cryptage
- **Chiffrement continu / par blocs** : selon que les données à crypter sont découpées en blocs ou pas

## Equations

- 1  $C = E(M) \leftarrow$  chiffrement
- 2  $D(C) = M \leftarrow$  déchiffrement
- 3  $D(E(M)) = M \leftarrow$  Equivalence

## Vocabulaire:

- **Texte en clair** : Données compréhensibles qu'un expéditeur veut envoyer à un destinataire
- **Texte Chiffré / Cryptogramme** : Données incompréhensibles obtenu à partir du texte en claire
- **Chiffrement** : Processus transformant le texte en clair en texte chiffré
- **Déchiffrement/Décryptage** : Processus inverse du chiffrement
- **Cryptographie** : science consistant à garder les messages secrets
- **Cryptographe** : Individu pratiquant de la cryptographie
- **Cryptanalyste** : Individu spécialisé dans le décryptage des messages chiffrés  
⇒ ennemi du cryptographe
- **Cryptologie** : Branche mathématique s'intéressant au cryptage
- **Chiffrement continu / par blocs** : selon que les données à crypter sont découpées en blocs ou pas

## Equations

- 1  $C = E(M) \leftarrow$  chiffrement
- 2  $D(C) = M \leftarrow$  déchiffrement
- 3  $D(E(M)) = M \leftarrow$  Equivalence

## Vocabulaire:

- **Texte en clair** : Données compréhensibles qu'un expéditeur veut envoyer à un destinataire
- **Texte Chiffré / Cryptogramme** : Données incompréhensibles obtenu à partir du texte en claire
- **Chiffrement** : Processus transformant le texte en clair en texte chiffré
- **Déchiffrement/Décryptage** : Processus inverse du chiffrement
- **Cryptographie** : science consistant à garder les messages secrets
- **Cryptographe** : Individu pratiquant de la cryptographie
- **Cryptanalyste** : Individu spécialisé dans le décryptage des messages chiffrés  
⇒ ennemi du cryptographe
- **Cryptologie** : Branche mathématique s'intéressant au cryptage
- **Chiffrement continu / par blocs** : selon que les données à crypter sont découpées en blocs ou pas

## Equations

- 1  $C = E(M) \leftarrow$  chiffrement
- 2  $D(C) = M \leftarrow$  déchiffrement
- 3  $D(E(M)) = M \leftarrow$  Equivalence

## Vocabulaire:

- **Texte en clair** : Données compréhensibles qu'un expéditeur veut envoyer à un destinataire
- **Texte Chiffré / Cryptogramme** : Données incompréhensibles obtenu à partir du texte en claire
- **Chiffrement** : Processus transformant le texte en clair en texte chiffré
- **Déchiffrement/Décryptage** : Processus inverse du chiffrement
- **Cryptographie** : science consistant à garder les messages secrets
- **Cryptographe** : Individu pratiquant de la cryptographie
- **Cryptanalyste** : Individu spécialisé dans le décryptage des messages chiffrés  
⇒ ennemi du cryptographe
- **Cryptologie** : Branche mathématique s'intéressant au cryptage
- Chiffrement continu / par blocs : selon que les données à crypter sont découpées en blocs ou pas

## Equations

- 1  $C = E(M) \leftarrow$  chiffrement
- 2  $D(C) = M \leftarrow$  déchiffrement
- 3  $D(E(M)) = M \leftarrow$  Equivalence

## Vocabulaire:

- **Texte en clair** : Données compréhensibles qu'un expéditeur veut envoyer à un destinataire
- **Texte Chiffré / Cryptogramme** : Données incompréhensibles obtenu à partir du texte en claire
- **Chiffrement** : Processus transformant le texte en clair en texte chiffré
- **Déchiffrement/Décryptage** : Processus inverse du chiffrement
- **Cryptographie** : science consistant à garder les messages secrets
- **Cryptographe** : Individu pratiquant de la cryptographie
- **Cryptanalyste** : Individu spécialisé dans le décryptage des messages chiffrés  
⇒ ennemi du cryptographe
- **Cryptologie** : Branche mathématique s'intéressant au cryptage
- **Chiffrement continu / par blocs** : selon que les données à crypter sont découpées en blocs ou pas

## Equations

- 1  $C = E(M) \leftarrow$  chiffrement
- 2  $D(C) = M \leftarrow$  déchiffrement
- 3  $D(E(M)) = M \leftarrow$  Equivalence



## Vocabulaire:

- **Texte en clair** : Données compréhensibles qu'un expéditeur veut envoyer à un destinataire
- **Texte Chiffré / Cryptogramme** : Données incompréhensibles obtenu à partir du texte en claire
- **Chiffrement** : Processus transformant le texte en clair en texte chiffré
- **Déchiffrement/Décryptage** : Processus inverse du chiffrement
- **Cryptographie** : science consistant à garder les messages secrets
- **Cryptographe** : Individu pratiquant de la cryptographie
- **Cryptanalyste** : Individu spécialisé dans le décryptage des messages chiffrés  
⇒ ennemi du cryptographe
- **Cryptologie** : Branche mathématique s'intéressant au cryptage
- **Chiffrement continu / par blocs** : selon que les données à crypter sont découpées en blocs ou pas

## Equations

- 1  $C = E(M) \leftarrow$  chiffrement
- 2  $D(C) = M \leftarrow$  déchiffrement
- 3  $D(E(M)) = M \leftarrow$  Equivalence

## Types d'algos:

- algo = fonction mathématique utilisée pour le chiffrement/déchiffrement
- Deux types d'algo : algo secret<sup>a</sup> et algo public

<sup>a</sup>encore appelé algo restreint, n'est quasiment plus le cas aujourd'hui

## Algo restreint

- Réalisation de systèmes sûrs
- Moindre "fuite" au niveau algo détruit toute la sécurité
- Pas de standardisation possible  $\Rightarrow$  pas de compatibilité/interopérabilité

## Algo public

- Sont connus de tous
- Recours à une information "secrete" constituée par une ou plusieurs clés
- En cas de fuite au niveau de la clé, tout le système n'est pas à casser

## Types d'algos:

- algo = fonction mathématique utilisée pour le chiffrement/déchiffrement
- Deux types d'algo : algo secret<sup>a</sup> et algo public

---

<sup>a</sup>encore appelé algo restreint, n'est quasiment plus le cas aujourd'hui

## Algo restreint

- Réalisation de systèmes sûrs
- Moindre "fuite" au niveau algo détruit toute la sécurité
- Pas de standardisation possible  $\Rightarrow$  pas de compatibilité/interopérabilité

## Algo public

- Sont connus de tous
- Recours à une information "secrete" constituée par une ou plusieurs clés
- En cas de fuite au niveau de la clé, tout le système n'est pas à casser

## Types d'algos:

- algo = fonction mathématique utilisée pour le chiffrement/déchiffrement
- Deux types d'algo : algo secret<sup>a</sup> et algo public

---

<sup>a</sup>encore appelé algo restreint, n'est quasiment plus le cas aujourd'hui

## Algo restreint

- Réalisation de systèmes sûrs
- Moindre "fuite" au niveau algo détruit toute la sécurité
- Pas de standardisation possible  $\Rightarrow$  pas de compatibilité/interopérabilité

## Algo public

- Sont connus de tous
- Recours à une information "secrete" constituée par une ou plusieurs clés
- En cas de fuite au niveau de la clé, tout le système n'est pas à casser

## Types d'algos:

- algo = fonction mathématique utilisée pour le chiffrement/déchiffrement
- Deux types d'algo : algo secret<sup>a</sup> et algo public

---

<sup>a</sup>encore appelé algo restreint, n'est quasiment plus le cas aujourd'hui

## Algo restreint

- Réalisation de systèmes sûrs
- Moindre "fuite" au niveau algo détruit toute la sécurité
- Pas de standardisation possible  $\Rightarrow$  pas de compatibilité/interopérabilité

## Algo public

- Sont connus de tous
- Recours à une information "secrete" constituée par une ou plusieurs clés
- En cas de fuite au niveau de la clé, tout le système n'est pas à casser

## Types d'algos:

- algo = fonction mathématique utilisée pour le chiffrement/déchiffrement
- Deux types d'algo : algo secret<sup>a</sup> et algo public

<sup>a</sup>encore appelé algo restreint, n'est quasiment plus le cas aujourd'hui

## Algo restreint

- Réalisation de systèmes sûrs
- Moindre "fuite" au niveau algo détruit toute la sécurité
- Pas de standardisation possible  $\Rightarrow$  pas de compatibilité/interopérabilité

## Algo public

- Sont connus de tous
- Recours à une information "secrete" constituée par une ou plusieurs clés
- En cas de fuite au niveau de la clé, tout le système n'est pas à casser

## Types d'algos:

- algo = fonction mathématique utilisée pour le chiffrement/déchiffrement
- Deux types d'algo : algo secret<sup>a</sup> et algo public

<sup>a</sup>encore appelé algo restreint, n'est quasiment plus le cas aujourd'hui

## Algo restreint

- Réalisation de systèmes sûrs
- Moindre "fuite" au niveau algo détruit toute la sécurité
- Pas de standardisation possible  $\Rightarrow$  pas de compatibilité/interopérabilité

## Algo public

- Sont connus de tous
- Recours à une information "secrete" constituée par une ou plusieurs clés
- En cas de fuite au niveau de la clé, tout le système n'est pas à casser

## Types d'algos:

- algo = fonction mathématique utilisée pour le chiffrement/déchiffrement
- Deux types d'algo : algo secret<sup>a</sup> et algo public

<sup>a</sup>encore appelé algo restreint, n'est quasiment plus le cas aujourd'hui

## Algo restreint

- Réalisation de systèmes sûrs
- Moindre "fuite" au niveau algo détruit toute la sécurité
- Pas de standardisation possible  $\Rightarrow$  pas de compatibilité/interopérabilité

## Algo public

- Sont connus de tous
- Recours à une information "secrete" constituée par une ou plusieurs clés
- En cas de fuite au niveau de la clé, tout le système n'est pas à casser



## Types d'algos:

- algo = fonction mathématique utilisée pour le chiffrement/déchiffrement
- Deux types d'algo : algo secret<sup>a</sup> et algo public

---

<sup>a</sup>encore appelé algo restreint, n'est quasiment plus le cas aujourd'hui

## Algo restreint

- Réalisation de systèmes sûrs
- Moindre "fuite" au niveau algo détruit toute la sécurité
- Pas de standardisation possible  $\Rightarrow$  pas de compatibilité/interopérabilité

## Algo public

- Sont connus de tous
- Recours à une information "secrete" constituée par une ou plusieurs clés
- En cas de fuite au niveau de la clé, tout le système n'est pas à casser