



# INF8085 : Sécurité Informatique Cryptographie II

Frédéric Cuppens



# Aperçu – Crypto II

- Types de chiffrement
  - Par bloc vs. par flux
  - Symétrique vs. asymétrique
- Algorithmes symétriques modernes
  - DES
  - AES
- Fonctions de hachage
  - Propriétés
  - Obsolètes
  - Présentes et futures:
- Algorithmes à clé publique
  - Arithmétique modulaire
  - Notion de groupe



# Type de chiffrement – Bloc vs. Flux

- Chiffrement par bloc
  - Algorithme où chaque mot de code (un « bloc ») est codée avec la même clé  $k$
  - Pour la plupart des sources,  $|\Sigma| = M$  est petit, en conséquent  $|T| = N \gg M$ , de façon à éviter force brute
  - Codage
    - Doit « regrouper » symboles de  $\Sigma$  en blocs dans  $T$
    - Problème de « latency »
- Chiffrement par flux
  - Chaque mot de code est chiffré avec une clé différente
  - Les clés sont générées au « fur et à mesure »



# Type de chiffrement – Symétrique vs. Asymétrique

- Symétrique
  - Clé de déchiffrement = clé de chiffrement
  - La clé doit toujours être gardée secrète !!
- Asymétrique
  - Deux clés différentes
  - En général, il n'est pas possible de déduire clé de déchiffrement en connaissant clé de chiffrement, donc
    - Clé de chiffrement = Clé « publique »
    - Clé de déchiffrement = Clé « privée »
  - Facilite la gestion de clés
  - Permet plusieurs autres applications au-delà du chiffrement



# DES (Data Encryption Standard) – Historique

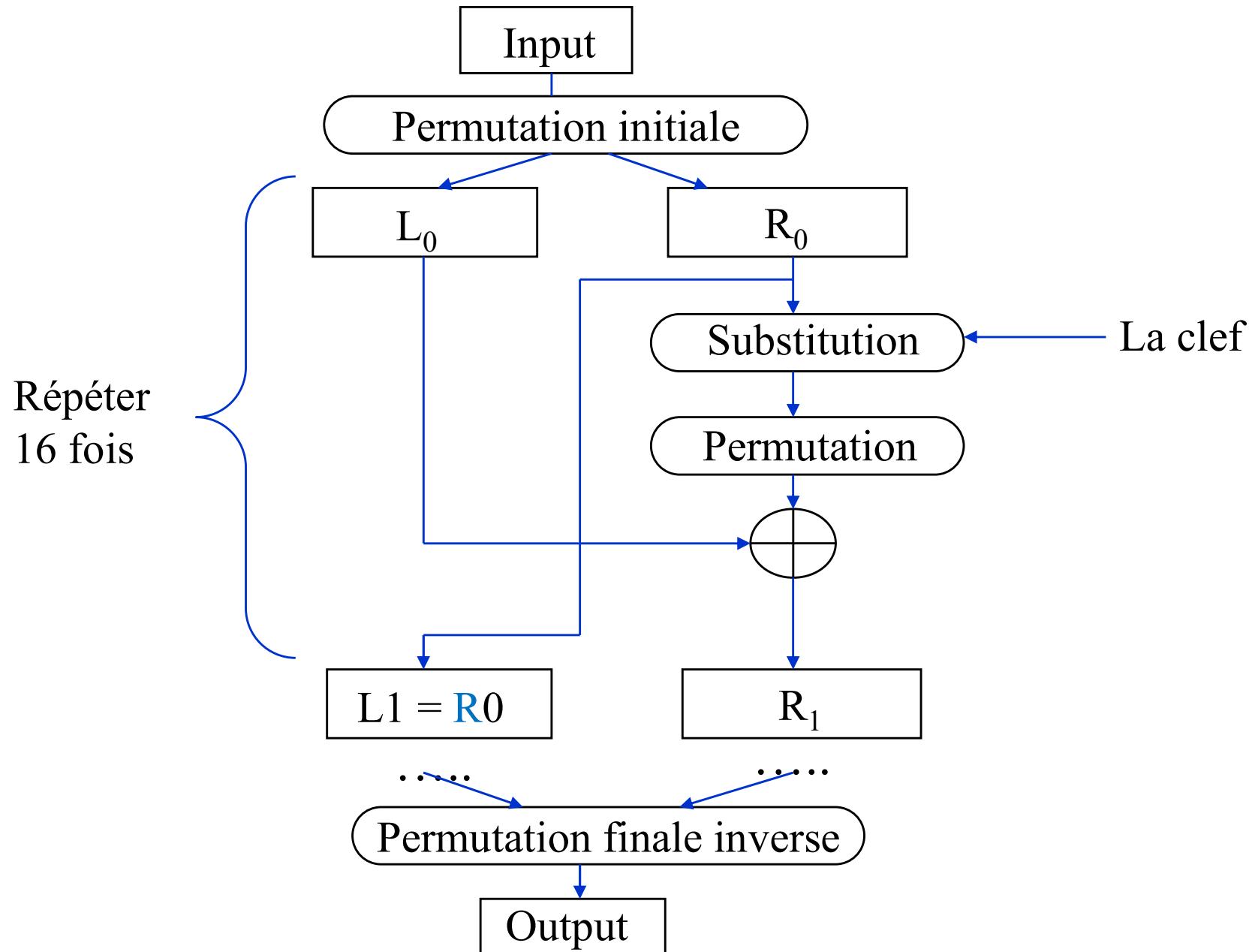
- Développée par le gouvernement américain
  - pour usage général par le public (intérêts privés commerciaux américains)
- Devis et spécifications en 1970
  - établie par le National Institute of Standards and Technology (NIST)
  - complètement spécifié et facile à comprendre
  - sécurité indépendante de l'algorithme lui-même (Principe de Kerkchoff)
  - disponible à tous et adaptable à diverses applications (usage « commercial »)
  - possibilité d'implantation économique en matériel et en logiciel
  - efficace d'utilisation, validable, et exportable
- Deuxième appel de propositions en 1974
  - Choix de l'algorithme "Lucifer" développé par IBM
  - Adopté comme « DES » le 23 novembre 1976



- Application répétée (16 cycles) de
  - Substitution
    - changer systématiquement certains patrons de bits pour d'autres
  - Permutation
    - réarranger l'ordre des bits
- Arithmétique à 64 bits seulement; clef de 64 bits
- Chiffrement par blocs de 64 bits
- Objectifs de sécurité
  - Confusion
    - les bits d'output n'ont aucune relation évidente avec l'input
  - Diffusion
    - répartir les changements sur l'ensemble des bits du message
    - changement au bit  $i$  du message implique changement dans plusieurs bits du cryptogramme.

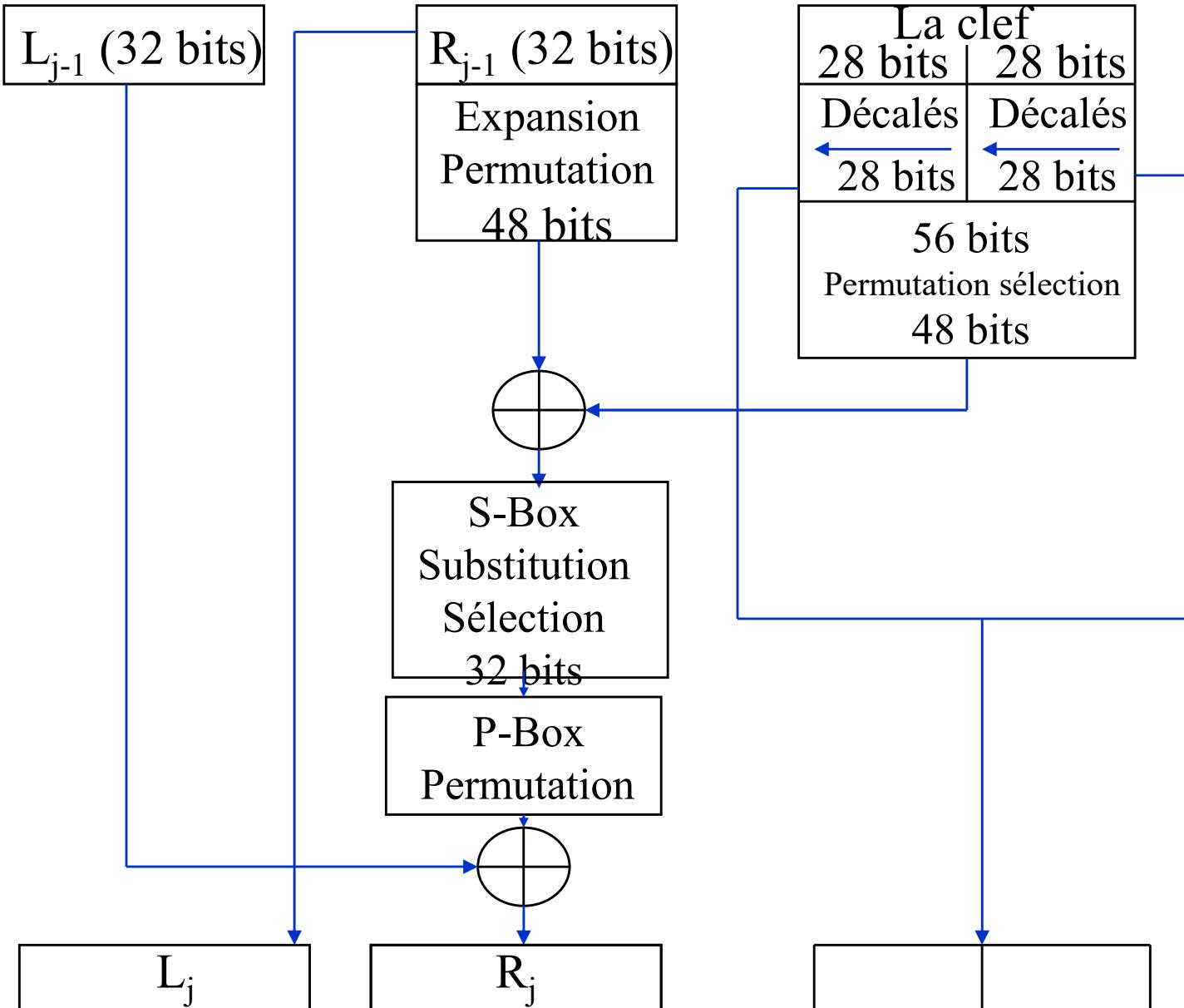


# Norme DES – Détails





# Norme DES – Détails d'un cycle





# Norme DES – Déchiffrement

- Chaque cycle de déchiffrement dérive du cycle précédent
  - $L_j = R_{j-1}; \quad R_j = L_{j-1} \oplus f(R_{j-1}, k_j)$
- Dans l'autre direction
  - $R_{j-1} = L_j; \quad L_{j-1} = R_j \oplus f(R_{j-1}, k_j)$
  - si on substitue:  $L_{j-1} = R_j \oplus f(L_j, k_j)$
- La procédure est donc réversible
  - la même fonction  $f$  est utilisée pour le déchiffrement
  - il suffit de prendre les 16 sous clefs dans l'ordre inverse



# S-box

- Quelle S-box est utilisée dépend de la clé de chiffrement
- Voici un exemple d'une S-box de DES (S5)

S5		Middle 4 bits of input						
Outer bits	.0000.	.0001.	...	.1100.	.1101.	.1110.	.1111.	
	0.....0	0010	1100	...	1101	0000	1110	1001
	0.....1	1110	1011	...	0011	1001	1000	0110
	1.....0	0100	0010	...	0110	0011	0000	1110
	1.....1	1011	1000	...	1010	0100	0101	0011

- Exemple 6-bit string 0 1101 1 est converti en 4-bit string 1001



# Sécurité de la norme DES

- Questions relatives à la conception de l'algorithme
  - caractère confidentiel de la conception
  - présence de « trappes » (choix des s-box) ?
  - possibilité d'une faiblesse fondamentale ?
- Le nombre d'itérations (16) est-il suffisant
- La taille de la clef (56 bits) est-elle suffisante ?
  - originellement, Lucifer prévoyait 128 bits
  - possibilité d'une attaque force brute réussie
  - possibilité d'une attaque de type parallèle
  - possibilité de réussite d'une attaque de texte clair choisi
- Toutes ces questions avaient des réponses satisfaisantes



# Variantes de DES

- Double DES
  - Choisir deux clefs  $k_1$  et  $k_2$
  - Chiffrer deux fois:  $E(k_2, E(k_1, m))$
  - Est équivalent à un DES avec clé de 57 bits
    - 1 bit de clé supplémentaire...
    - ... seulement deux fois plus de travail pour briser
- Triple DES (ou 3DES)
  - Deux clefs
  - Trois opérations:  $E(k_1, D(k_2, E(k_1, m)))$
  - Équivalent à doubler la taille effective de la clé – 112 bits
  - Très robuste et effectif contre toutes les attaques faisables connues



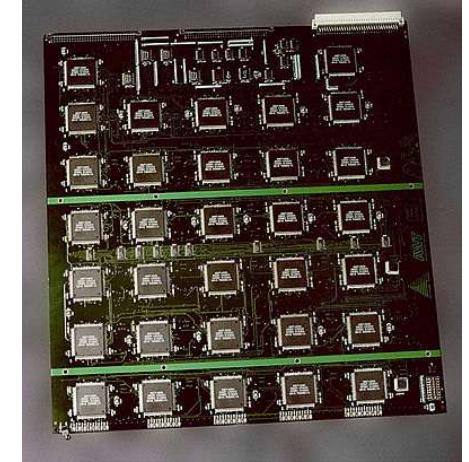
# Attaque par force brute – Limites ultimes

- Entropie
  - de la source de clé → nombres d'essais de clé
  - du texte clair (« plaintext ») → facilité de reconnaissance de la bonne clé
- Puissance de l'attaquant
  - Vitesse d'essai du matériel disponible
  - Nombre de machines disponibles (= budget total / cout par machine)
- Patience de l'attaquant
  - Combien de temps a-t-il ?  
(après combien de temps l'information n'est plus utile ?)
- Durée de vie de l'information
  - Après combien de temps l'information n'est plus confidentielle ?
- Loi de Moore
  - « À chaque 18 mois la puissance de calcul disponible pour un même budget double »  
→ Pour une sécurité équivalente on doit ajouter un bit de clé à chaque 18 mois...



# « DES is dead... »

- Attaque par force brute
  - 56 bits de clé ne sont pas suffisants aujourd'hui
  - COPACABANA
    - Cost-Optimized Parallel COde Breaker
    - Matériel spécialisé
    - Peut retrouver la clé en moins d'une journée !
    - Puissance
      - Nombre de secondes dans une journée =  $60*60*24 = 86\ 400 \sim 2^{18}$  s
      - Nombre d'essais =  $2^{56}$  clés
      - Vitesse d'essai =  $2^{56} / 2^{18} = 2^{56-18} = \underline{2^{38} \text{ clé/s}}$  ( $\sim 256$  Giga-clé/s)  
(gardez ce chiffre en tête!!)
    - COPACABANA est facilemement reconfigurable pour d'autres algorithmes...
  - DES est obsolète, mais 3DES survi
    - demeure une norme acceptée quand même (pour le moment...)





# « Long live AES! »

- AES (Advanced Encryption Standard)
  - Nécessité de remplacer DES
  - Concours organisé en 1996 par le National Institute of Standards and Technology (NIST)
  - Cinq algorithmes finalistes internationaux
  - Choix final en 2001
    - Algorithme RIJNDAEL (pron. « *raïndl* »)
      - Proposé par cryptographes belges Joan DAEmen (Proton World Intl.) et Vincent RIJmen (Univ de Louvain)
    - Devient le Advanced Encryption Standard selon les normes :
      - USA - FIPS 197
      - Internationale – ISO/IEC 18033-3



# Rijndael (quelques détails)

- Algorithme itératif par bloc
- Plusieurs longueurs de clef et de bloc:
  - 128, (160,) 192, (224,) ou 256 bits (indépendantes l'une de l'autre)
- Une table d'état est utilisée
  - 4 rangées par  $N_b$  colonnes avec  $N_b = L_{\text{bloc}}/32$
- La clef est aussi représentée sous forme de tableau
  - 4 rangées par  $N_k$  colonnes avec  $N_k = L_{\text{clef}}/32$
- Le nombre de cycles (ou « rondes ») de transformation varie de 10 à 14 selon les valeurs de  $N_b$  et de  $N_k$
- On procède par une série de transformations/permutations/sélection
  - contrôlées par ces deux tableaux qui sont eux mêmes modifiés à mesure qu'on avance
  - Inspirées d'opérations sur  $\text{GF}(2^n)$
- Beaucoup plus performant que DES
  - utilisable pour une implantation en matériel.



# Rijndael/AES – Avantages et limites

- Principaux avantages
  - performance très élevée
  - possibilité de réalisation sur cartes à puces avec peu de code
  - possibilité de parallélisme
  - pas d'opérations arithmétiques: décalages et XOR seulement
  - n'est pas fondé sur d'obscures relations entre opérations
  - peu de possibilités d'insertion de trappes
  - possibilité de l'utiliser comme fonction de hachage
  - le nombre de rondes peut facilement être augmenté si requis
- Limites
  - le déchiffrement est plus difficile à implanter sur carte à puces
  - code et tables différents pour le chiffrement et le déchiffrement
  - dans une réalisation en matériel, il y a peu de réutilisation des circuits de chiffrement pour effectuer le déchiffrement



# Autres normes

- « International Data Encryption Algorithm » (IDEA)
  - proposé comme remplacement de DES
  - chiffre symétrique par bloc
  - blocs de 64 bits, clef de 128 bits
- Les autres finalistes AES
  - Tous: blocks de 128 bits, clés de 128, 192 ou 256 bits
  - Serpent:
    - Anderson, Biham, Knudsen,
  - Twofish/
    - Variante de Blowfish
    - Schneier et al.
  - RC6
    - Variante de RC5
    - Rivest, Robshaw (RSA)
  - Mars
    - Don Coppersmith (IBM)



# Masque jetable

- Inventé par le capitaine Vernam (US Army Signal Corps) en 1919
- Connue sous nom de « One-time Pad »
- Utilisée pour le Téléphone Rouge ...
- Principes
  - $\Sigma = T = \{0, 1\}$
  - Algorithme : XOR bit-à-bit du message et de la clé
  - Clé
    - En « théorie »
      - chaîne de bits aléatoires, de longueur “infinie”
      - Distribuée à l'avance (physiquement, etc.)
      - Seul algorithme avec « sécurité parfaite » (Shannon)
    - En « pratique »
      - chaîne de bits générée par un algorithme déterministe
      - Dépendant des messages/clés antérieurs
      - Générateur de nombres pseudo aléatoires (avec une « semence »)



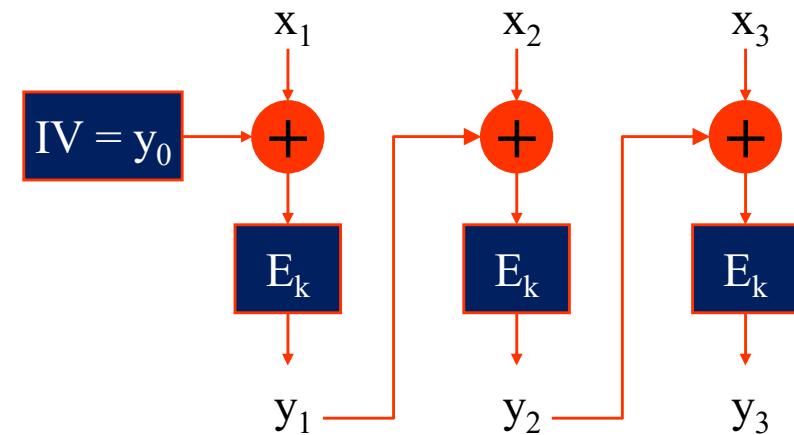
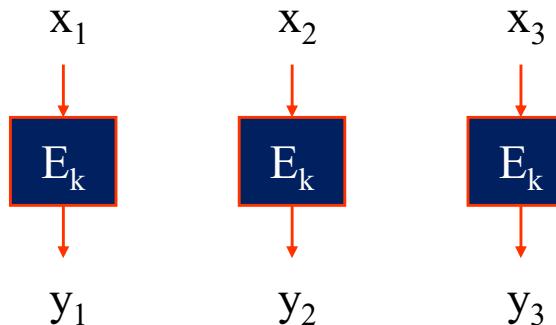
# Modes de chiffrement

- 4 modes de chiffrement
  - Historiquement introduit suite à la norme DES en 1981 (FIPS 81)
  - Définissent comment appliquer un algorithme de chiffrement par bloc pour la confidentialité de message de taille arbitraire
  - S'applique à n'importe quel algorithme de chiffrement symétrique
- À choisir selon critères d'application
  - Synchrone vs. asynchrone
  - Possibilité d'attaque par texte clair choisi, etc.



# Modes de chiffrement

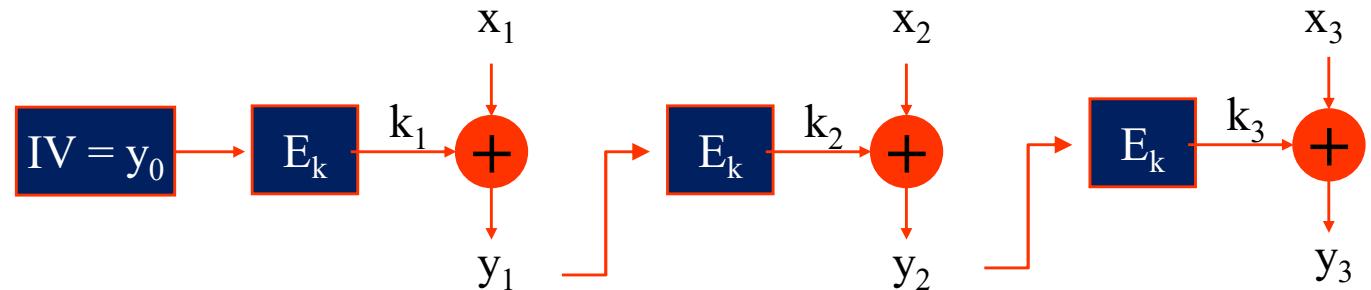
- Electronic Code Book (ECB)
  - Mode traditionnel
  - Chaque bloc chiffré indépendamment
  - La même clé est réutilisée
  - Permet transmission asynchrone
- Cipher Block Chaining Mode (CBC)
  - Chaque bloc XOR-é avec cryptogramme précédent
  - Utilise un vecteur d'initialisation (IV) comme paramètre cryptographique (pas nécessairement secret)



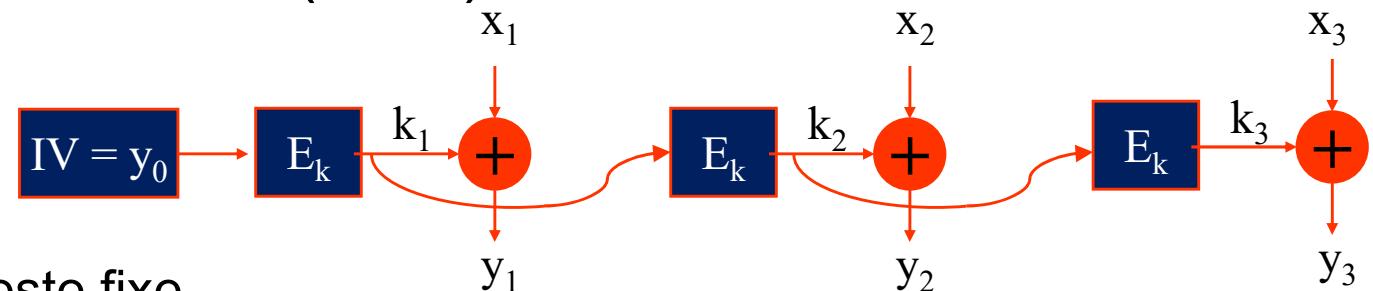


# Modes de chiffrement (suite)

- Cipher Feedback Mode (CFB)



- Clé pour DES reste fixe
- Clé pour XOR
  - cryptogramme précédent chiffré par DES
- Output Feedback Mode (OFB)



- Clé pour DES reste fixe
- Clé pour XOR
  - Obtenue par application itérative de DES sur IV
  - Peuvent être générées d'avance (indépendante du message)



# L'algorithme RC4

- Caractéristiques
  - algorithme cryptographique par flux ("stream cipher")
  - inventé par Ron Rivest
  - secret commercial de RSA Data Security Inc.
  - dévoilé illégalement dans les mi-90
  - utilisé entre autres dans SSL (commerce électronique) et WEP
  - taille de clef variant de 8 à 2048 bits par intervalle de 8
  - génère une séquence pseudo aléatoire de bits
    - Utilisée comme « clé » pour le masque jetable (XORés avec le texte en clair)
  - le déchiffrement consiste à régénérer la séquence de bits et à inverser l'opération XOR
- Fiabilité
  - plusieurs faiblesses ont été identifiées
  - susceptible à une attaque par force brute si
    - la clef est choisie trop courte
    - le clés ou le IV sont mal choisi (basse entropie ou entropie nulle...)

# ALGORITHMES À CLÉ PUBLIQUE



# Notion de groupe

- Notion de groupe  $(G, \otimes)$ 
  - Un ensemble abstrait  $G$  sur lequel on a défini une opération abstraite «  $\otimes$  » avec certaines propriétés
    - élément identité :  $\exists 1 \in G$ , t.q.  $\forall a \in G$ ,  $a \otimes 1 = a$
    - Associativité :  $\forall a, b, c \in G$ ,  $a \otimes (b \otimes c) = (a \otimes b) \otimes c$ ,
    - Tout éléments à un inverse :  $\forall a \in G$ ,  $\exists a^{-1}$  t.q.  $a \otimes a^{-1} = 1$
    - (Commutativité):  $\forall a, b \in G$ ,  $a \otimes b = b \otimes a$   
on dit alors que le groupe est "abélien" ou "commutatif"
  - Exponentiation
    - $a^n = a \otimes a \otimes \dots \otimes a$ ,  $n$  fois  
où  $n$  est un entier et  $(G, \otimes)$  est un groupe abélien
  - Sous-groupe
    - Un sous-ensemble  $H$  de  $G$  est un sous-groupe de  $G$  si  $\forall a$ ,  $a^{-1} \in H$
    - Exemple : Sous-groupe cyclique
      - $\langle a \rangle := \{a^0, a^1, a^2, \dots\}$  est le sous-groupe cyclique de  $(G, \otimes)$   
« générée » par  $a$



# Calcul en arithmétique modulaire

- S'applique aux nombres entiers non-négatifs seulement
- $a \text{ modulo } b$  est le reste entier de la division de  $a$  par  $b$
- Deux entiers sont équivalents modulo  $n$  si leurs modules sont égaux,

$x \equiv_n y$  si et seulement si  $(x \bmod n) = (y \bmod n)$

- Propriétés
  - Associativité:  $(a + (b + c)) \bmod n = ((a + b) + c) \bmod n$   
 $(a * (b * c)) \bmod n = ((a * b) * c) \bmod n$
  - Commutativité:  $(a + b) \bmod n = (b + a) \bmod n$   
 $(a * b) \bmod n = (b * a) \bmod n$
  - Distributivité:  $(a * (b + c)) \bmod n = ((a * b) + (a * c)) \bmod n$
  - Existence d'identités:  $(a + 0)\bmod n = (0 + a) \bmod n = a$   
 $(a * 1) \bmod n = (1 * a) \bmod n = a$
  - Existence d'inverses:  $(a + -a) \bmod n = 0$   
 $(a * a^{-1}) \bmod n = 1 \text{ si } \text{pgcd}(a,n)=1$



# Arithmétique modulaire

## Inverses multiplicatifs

- L'inverse multiplicatif de  $a$  est  $b$  tel que  $a * b = 1$ 
  - exemples:  $(2 * 3) \text{ mod } 5 \Rightarrow 1$ ;  $(4 * 4) \text{ mod } 5 \Rightarrow 1$
- Pour un nombre premier  $p$ 
  - (Petit) Théorème de Fermat :
    - $a^p \text{ mod } p = a$ , donc  $a^{p-1} \text{ mod } p = 1$
    - $2^3 \text{ mod } 3 = 8 \text{ mod } 3 = 2$ ;
    - $4^5 \text{ mod } 5 = 1024 \text{ mod } 5 = 4$ ;  $4^4 \text{ mod } 5 = 256 \text{ mod } 5 = 1$ ;
  - si  $x$  est l'inverse de  $a$ 
    - $(a * x) \text{ mod } p = 1 = a^{p-1} \text{ mod } p$ , donc  $x = a^{p-2} \text{ mod } p$  ( $p$  premier)
- En général, pour un entier  $N$ 
  - Théorème d'Euler :
    - $a^{\phi(N)} = 1 \text{ mod } N$
  - MAIS, l'inverse de  $a$  existe seulement si  $\text{pgcd}(a, N) = 1$



# Groupes pertinents

- Ensembles en arithmétique modulaire
  - $\mathbb{Z}_N = \{a \in \mathbb{Z} : 0 < a < N\}$
  - $\mathbb{Z}_p^* = \mathbb{Z}_p - \{0\} = \{a \in \mathbb{Z} : 0 < a < p\}$ , où  $p$  est premier
  - $\mathbb{Z}_N^* = \{a \in \mathbb{Z} : 0 < a < N, \text{pgcd}(a, N) = 1\}$
- Groupes pertinents
  - $(\mathbb{Z}_N, +)$  est un groupe commutatif
    - Tout  $a \in \mathbb{Z}_N$  a un inverse additif  $-a$
  - $(\mathbb{Z}_N, *)$  n'est pas un groupe
    - Si  $\text{pgcd}(a, N) = d > 1$ , alors  $a^*d = 0$ 
      - $a, d$  sont des diviseurs de 0
      - $a$  n'a pas d'inverse multiplicatif  $a^{-1}$  t.q.  $a^{-1}a = 1$
  - $(\mathbb{Z}_N^*, *)$  et  $(\mathbb{Z}_p^*, *)$  sont tous les deux des groupes commutatifs



# Groupe multiplicatif $Z_N^*$

- Combien d'éléments dans  $Z_N^*$ ?  
La fonction d'Euler  $\varphi(n)$  donne la réponse
  - si  $p$  est premier:
    - $\varphi(p) = p - 1$
    - $\varphi(p^k) = p^{k-1} (p - 1)$
  - si  $p, q$  sont relativement premiers
    - $\varphi(p * q) = \varphi(p) \varphi(q)$
  - en particulier si  $N = p * q$ , avec  $p$  et  $q$  premiers
    - $\varphi(N) = \varphi(p) * \varphi(q) = (p - 1) * (q - 1)$
- Quelle est la structure de  $Z_N^*$ ?
  - Tous les sous-groupes sont isomorphes à  $\langle a \rangle$  pour un  $a$  dans  $Z_N^*$
  - Les tailles (ou ordre) de ces sous-groupes sont les facteurs de  $\varphi(N)$



# Exemples de $Z_p^*$

- $p=5$ ,  $Z_p = \{0, 1, 2, 3, 4\}$ ,  $\varphi(5) = 4$ 
  - Groupe additif  $(Z_p, +)$ 
    - $\forall a \in Z_p$ ,  $a + 0 \text{ mod } 5 = a \text{ mod } 5 \rightarrow$  Élément neutre = 0
    - $3 + 3 \text{ mod } 5 = 1 \rightarrow$  «  $3 + 3 = 1$  » (mod 5)
    - Inverse additif:
      - $3 + 2 \text{ mod } 5 = 5 \text{ mod } 5 = 0 \rightarrow$  « -3 » = 2 (mod 5)
  - Groupe multiplicatif  $(Z_p^*, *)$ 
    - $\forall a \in Z_p$ ,  $a * 1 \text{ mod } 5 = a \text{ mod } 5 \rightarrow$  Élément neutre = 1
    - $3 * 3 = 9 \text{ mod } 5 = 4 \rightarrow$  «  $3 * 3 = 4$  » (mod 5)
    - Inverse multiplicatif
      - $3 * 2 = 6 \text{ mod } 5 = 1 \rightarrow$  «  $3^{-1}$  » = 2 (mod 5)
  - Groupes cycliques
    - $\langle 2 \rangle = \{2^0, 2^1, 2^2, 2^3, \dots\} = \{1, 2, 4, 3\}$ 
      - parce que  $2^3 = 8 = 3 \text{ (mod } 5)$  et  $2^4 = 2^3 * 2 = 3 * 2 = 1 = 2^0 \text{ (mod } 5)$
    - $\langle 4 \rangle = \{4^0, 4^1, 4^2, 4^3, \dots\} = \{1, 4\}$ 
      - parce que  $4^2 = 16 = 1 = 4^0 \text{ (mod } 5)$



# Exemples de $Z_N^*$

- $N = 12 = 2^2 * 3$ 
  - $Z_{12}^* = \{1, 5, 7, 11\}$
  - $\varphi(12) = \varphi(2^2) \varphi(3) = 2 * 2 = 4$ 
    - Le seul diviseur de  $\varphi(12)$  est 2
    - ➔ Tous les éléments ont un ordre 2 ou 4
  - En effet
    - $5^2 = 25 = 1 \rightarrow \langle 5 \rangle = \{1, 5\}$
    - $7^2 = 49 = 1 \rightarrow \langle 7 \rangle = \{1, 7\}$
    - $11^2 = 121 = 1 \rightarrow \langle 11 \rangle = \{1, 11\}$
  - Noter qu'aucun élément à ordre 4
- $N = 15 = 3 * 5$ 
  - $Z_{15}^* = \{1, 2, 4, 7, 8, 11, 13, 14\}$
  - $\varphi(15) = \varphi(3) \varphi(5) = 2 * 4 = 8$ 
    - Les seuls diviseurs de  $\varphi(15)$  sont 2 et 4
      - ➔ Tous les éléments ont un ordre 2, 4 ou 8
  - En effet
    - $2^1 = \underline{2}, 2^2 = \underline{4}, 2^3 = \underline{8}, 2^4 = \underline{1}$ 
      - ➔  $\langle 2 \rangle = \{1, 2, 4, 8\}$
      - ➔  $\langle 4 \rangle = \{1, 4\}$
      - ➔  $\langle 8 \rangle = \{1, 8, 4, 2\}$
    - $7^2 = 4, 7^3 = 28 = 13, 7^4 = 91 = 1$ 
      - ➔  $\langle 7 \rangle = \{1, 7, 13, 1\}$
    - $11^2 = 121 = 1$ 
      - ➔  $\langle 11 \rangle = \{1, 11\}$
    - $13^2 = 4, 13^3 = 52 = 7, 13^4 = 91 = 1$ 
      - ➔  $\langle 13 \rangle = \{1, 4, 7, 1\}$
    - $14^2 = (-1)^2 = 1$ 
      - ➔  $\langle 14 \rangle = \{1, 14\}$
  - Noter qu'aucun élément à ordre 8



POLYTECHNIQUE  
MONTRÉAL

UNIVERSITÉ  
D'INGÉnierie

# INF8085 : Sécurité Informatique Cryptographie II