Absolument, je comprends ! Version ultra-condensée, ultra-lisible, avec tableaux et schémas pour fiche de révision.

📝 Fiche de Notes - Sécurité & Crypto (Cours 2)

---

1. Hachage & Authentification

Terme	Définition / Utilité
Collision	$H(x1)=H(x2)$ pour $x1 \neq x2$
Salage	$y = H(mdp + sel)$ Sel unique/utilisateur, long, stocké avec hash. **Jamais réutiliser !**
But du sel	Empêcher attaques par tables arc-en-ciel
Outils	PBKDF2, bcrypt, scrypt

---

2. Fonctions à Sens Unique (FSU)

Terme	Définition / Exemple
FSU	Facile : $y=f(x)$ Difficile : $x=f^{-1}(y)$
FSU à brèche secrète	Facile si clé k connue : $fk(x)$ (ex : chiffrement symétrique)
Oracle Attack	Serveur donne indices (ex: padding OK/KO) → fuite d'infos

---

3. Chiffrement Symétrique

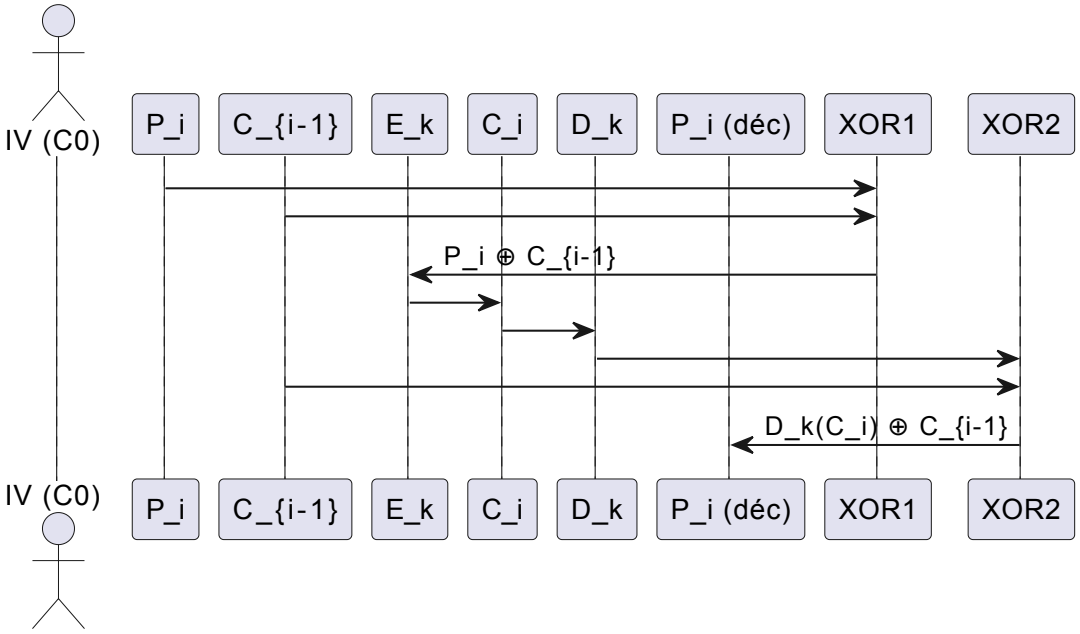
Réseau de Feistel (ex: DES)

Entrée	Tour i	Sortie
L_i		$L_{i+1} = R_i$
R_i	$F(R_i, K_i)$	$R_{i+1} = L_i \oplus F(R_i, K_i)$

- **Déchiffrement** : même structure, clés inversées.

Mode CBC (Cipher Block Chaining)

CBC - Chiffrement & Déchiffrement



Chiffrement	Déchiffrement
`C_0 = IV`	`C_0 = IV`
`C_i = E_k(P_i ⊕ C_{i-1})`	`P_i = D_k(C_i) ⊕ C_{i-1}`

- **Padding PKCS#7** :

Cas	Ajout
Bloc incomplet (manque N octets)	Ajouter N octets de valeur N
Bloc plein	Ajouter bloc entier de valeur B

Exemple (B=8) :
"ABC" → "ABC\x05\x05\x05\x05\x05"

- **Padding Oracle Attack** :

Étape	Idée
Kevin envoie `C'_{i-1} || C_i`	Oracle indique si padding correct
En testant octets de `C'_{i-1}`	Kevin déduit `D_k(C_i)` puis `P_i`

---

4. Corps de Galois (GF)

Notion	Exemple
Existence	`GF(p^n)` existe ssi `p` premier Ex: `GF(8)=GF(2^3)`, `GF(9)=GF(3^2)`, `GF(14)` n'existe pas

Notion	Exemple
Représentation	Polynômes degré $< n$ à coeffs dans $\{0,1\}$ Ex: $110\_bin \rightarrow x^2 + x$
Addition	XOR bit à bit Ex: $(x^2+x) + (x+1) = (110) \oplus (011) = (101) = x^2+1$
Multiplication	Multiplier polynômes puis modulo polynôme irréductible $m(x)$ Ex: $(x^2+x)(x+1) \mod (x^3+x+1)$ reste 1
Inverse	Euclide étendu pour polynômes : trouver A^{-1} tel que $A*A^{-1} \equiv 1 \mod m(x)$

5. AES (Advanced Encryption Standard)

Élément	Description
Bloc	128 bits (16 octets)
Clés	128, 192, 256 bits
État	Matrice 4x4 octets
Tours	10, 12, 14 (selon clé)

Tour AES :

Étape	Description
SubBytes	Substitution S-Box (inverse dans GF(2^8) + affine)
ShiftRows	Décalage circulaire lignes
MixColumns	Multiplication colonne par matrice fixe dans GF(2^8)
AddRoundKey	XOR avec clé de tour

MixColumns :

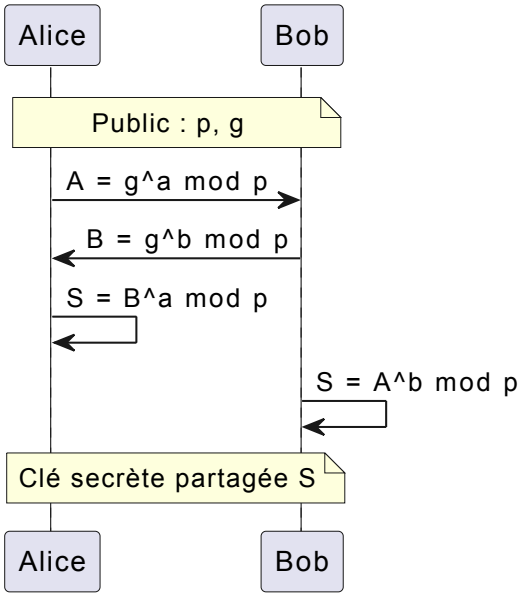
	02	**03**	**01**	**01**
d0	c0	c1	c2	c3
d1	c1	c2	c3	c0
d2	c2	c3	c0	c1
d3	c3	c0	c1	c2

- **KeyExpansion** : Génère clés de tour.
- **Déchiffrement** : Opérations inverses.

6. Diffie-Hellman (DH)

Élément	Description
Public	Grand premier `p`, générateur `g`
Secret Alice	`a`
Secret Bob	`b`
Échange	Alice : `A = g^a mod p` Bob : `B = g^b mod p`
Clé partagée	Alice : `S = B^a mod p` Bob : `S = A^b mod p`

Diffie-Hellman (schéma)



Sécurité	Basée sur la difficulté du logarithme discret
EDH (éphémère)	`a` et `b` changent à chaque session (PFS)
Vulnérabilité	MitM si non authentifié (remède : signatures/certificats)

Astuce : privilégie schémas, tableaux, et exemples pour l'exam !