Théorie des codes - TP 5
ZZ3 F5 - Réseaux et Sécurité Informatique
Cryptographie symmétrique : Chiffrement de type Rijndael

Rendu : au plus tard le 13 novembre, par mail à charles.olivier-anclin@uca.fr.

Définition du chiffrement mini-Rijndael

Le chiffrement AES, est un membre d'une famille plus vaste de fonctions de chiffrement appelée Rijndael. Au sein de cette famille, il est possible d'utiliser des états rectangulaires, contrairement à l'AES qui utilise des états carrés.

Nous considérons un chiffrement de type Rijndael, en utilisant des paramètres différents de ceux de l'AES et en mettant en place des fonctions adaptées à ces paramètres.

- l'état est un rectangle de 3 lignes sur 4 colonnes ;
- les éléments du rectangle sont des éléments du corps $\mathbb{F}_{2^{16}}$ défini par $P_{16}(X) = X^{16} + X^5 + X^3 + X^2 + 1$;
- la taille de bloc et de la clef est de 192 bits.

On représente comme d'habitude en binaire les éléments du corps $\mathbb{F}_{2^{16}}$ avec la convention que les poids faibles correspondent aux petits degrés, par exemple l'élément $X^{13} + X^{10} + X^9 + X^8 + X^7 + X^3 + X + 1$ s'écrit $\{0010011110001011\}_2$, ou encore $\{\texttt{278b}\}_{16}$ en hexadécimal.

Pour convertir un bloc de texte clair en un état rectangulaire (et inversement), on numérote les cases de l'état de la façon suivante :

a_0	a_1	a_2	a_3
a_4	a_5	a_6	a_7
a_8	a_9	a_{10}	a_{11}

correspond au bloc $a_0||a_1||a_2||a_3||a_4||a_5||a_6||a_7||a_8||a_9||a_{10}||a_{11}$ (concaténé des représentations binaires). La clef maîtresse et les sous-clefs de tours faisant chacune 192 bits s'interprètent de la même façon comme un état de 3 lignes sur 4 colonnes d'éléments de $\mathbb{F}_{2^{16}}$.

On définit les opérations suivantes sur l'état :

SubBytes Remplacer chaque élément de l'état par son inverse dans $\mathbb{F}_{2^{16}}$ (l'élément nul est invariant). Contrairement à AES, on ne fait pas d'opération linéaire sur les bits après l'inversion.

ShiftRow La deuxième ligne est décalée cycliquement d'une position vers la gauche et la troisième ligne est décalée cycliquement de deux positions vers la gauche.

MixColumns Multiplier chacune des colonnes vue comme un polynôme (degré faible à fort de haut en bas) par $G = Y + \{\texttt{3}\}$ et réduit modulo $C = Y^3 + \{\texttt{1}\}$. Exemple pour la première colonne le calcul sera $(a_0 + a_4 Y + a_8 Y^2) \cdot G \mod C$.

AddRoundKey Ajouter à chaque élément de l'état l'élément de position correspondante dans la sous-clef du tour.

On définit le chiffrement d'un bloc clair M par l'algorithme suivant :

Chiffrement $E_K(M)$

1. $A \leftarrow M$
2. $A \leftarrow AddRoundKey(A, K_0)$
3. Pour i de 1 à 4 Faire
   (a) $A \leftarrow SubBytes(A)$
   (b) $A \leftarrow ShiftRow(A)$
   (c) $A \leftarrow MixColumns(A)$
   (d) $A \leftarrow AddRoundKey(A, K_i)$
4. $A \leftarrow SubBytes(A)$
5. $A \leftarrow ShiftRow(A)$
6. $A \leftarrow AddRoundKey(A, K_5)$
7. Return A

Extension de clef Notre mini-Rijndael a besoin des plusieurs sous-clefs de tour (de K_0 à K_5) qu'il faut générer à partir de la clef maîtresse K.

On décrit l'algorithme de génération des sous-clefs à l'aide de la fonction Mix :

$$\begin{aligned} b_0||b_1||\dots||b_{11} &= Mix(a_0||a_1||\dots||a_{11}, u) \\ b_i &= SubByte((a_{(i+1) \bmod 12} + X^i) \bmod P_{16}) + u \\ K_0 &= Mix(K, \{\texttt{6582}\}) \\ K_1 &= Mix(K_0, \{\texttt{a536}\}) \\ K_2 &= Mix(K_1, \{\texttt{b14}\}) \\ K_3 &= Mix(K_2, \{\texttt{3885}\}) \\ K_4 &= Mix(K_3, \{\texttt{41b6}\}) \\ K_5 &= Mix(K_4, \{\texttt{5f28}\}) \end{aligned}$$

1 Arithmétique dans $\mathbb{F}_{2^{16}}$

1.1 Une première implémentation

1. Écrire l'addition de deux éléments de $\mathbb{F}_{2^{16}}$. Vous pouvez voir a et $b \in \mathbb{F}_{2^{16}}$ comme des polynomes à coefficients binaires (*i.e.,* dans $\mathbb{F}_2$) de degré au plus 16.
2. Écrire la multiplication par X (le polynome de $\mathbb{F}_{2^{16}}$) d'un élément quelconque de $\mathbb{F}_{2^{16}}$ (on pensera évidemment à réduire par P_{16}).
3. En déduire comment multiplier deux éléments de $\mathbb{F}_{2^{16}}$, et écrire la fonction correspondante.
4. Comment utiliser la multiplication dans $\mathbb{F}_{2^{16}}$ pour calculer l'inverse de tout élément non nul de $\mathbb{F}_{2^{16}}$? Indication : l'élément $g = X$ est un générateur du groupe multiplicatif de $\mathbb{F}_{2^{16}}$. (Cette question ne requière pas l'écriture de code.)

1.2 Multiplication tabulée

Pour améliorer le coût des multiplications et divisions obtenues précédemment, on souhaite utiliser une représentation logarithmique. Pour cela, on associe à chaque élément x du groupe multiplicatif $(\mathbb{F}_{2^{16}})^*$ son logarithme discret k par un générateur g : tel que $x = g^k$. Par convention, on associera l'indice $2^{16} - 1$ à l'élément 0.

1. Quel générateur peut-on utiliser avec le polynômes primitifs P_{16} ? Indication : la réponse est donné plus haut.
2. Construire la table de correspondance qui pour un générateur G associe à un indice i, le polynome $P = G^i$, *i.e.*, $i \hookrightarrow P \equiv G^i \in \mathbb{F}_{2^{16}}$.
3. Construire la table de correspondance inverse.
4. Construire la table des logarithmes de Zech : pour un indice i la table renvoie l'indice j tel que $G^j \equiv 1 + G^i \in \mathbb{F}_{2^{16}}$ (cela correspond à la fonction $i \hookrightarrow j$ tel que $G^j \equiv 1 + G^i \in \mathbb{F}_{2^{16}}$).

2 Implémentation du mini-Rijndael

1. Implémenter la fonction **SubBytes** à l'aide d'une table pré-calculée. Quelle est la taille de cette table ?
2. Implémenter la fonction **ShiftRow**.
3. Comme dans AES, l'opération **MixColumns** est linéaire. Expliciter sa représentation matricielle (dans les bases canoniques de $\mathbb{F}_{2^{16}}[X]$).
4. Implémenter l'opération **MixColumns**.
5. On pourrait utiliser des tables pour accélérer l'opération **MixColumns**. Expliquer comment, et donner la taille des tables correspondantes.
6. Écrire la fonction de chiffrement E, et donner le temps de chiffrement nécessaire pour un message allant de 1 bloc à 1 million de blocs, sous la forme d'une courbe.

3 Déchiffrement

1. Écrire l'inverse de **ShiftRow**.
2. Écrire l'inverse de **MixColumns**.

   Indication : on donne

   $$(\{\texttt{5b65}\}Y^2 + \{\texttt{edaf}\}Y + \{\texttt{36dc}\})G = \{\texttt{1}\} \bmod C.$$

3. A-t-on besoin d'écrire l'inverse de **SubBytes** ?
4. Écrire la fonction de déchiffrement D, et donner le temps de déchiffrement nécessaire pour un message allant de 1 bloc à 1 million de blocs, sous la forme d'une courbe.

4 Mode CFB

On rappelle le mode de chiffrement CFB :

$$\begin{aligned} c_i &= E(c_{i-1}) \oplus m_i \\ c_0 &= \text{IV}. \end{aligned}$$

1. Donner les équations de déchiffrement correspondantes.
2. Quelle particularité voyez vous dans le mode CFB dans l'utilisation des fonctions E et D ?
3. Déchiffrer le fichier `secret.enc` chiffré en mode CFB (le premier bloc du fichier correspond à l'IV) avec la clef {`6edc2fbe41b6e8ca1a10b7105509e71140f58f3b79a7d1b6`}.

Tests

Quelques vecteurs de test pour vous aider à débugger :

$$\{\texttt{a3a6}\} \times \{\texttt{3df5}\} = \{\texttt{61a6}\}$$
$$\{\texttt{a3a6}\}^{-1} = \{\texttt{4103}\}$$
$$\{\texttt{3fd4}\} \times \{\texttt{5162}\} = \{\texttt{3ede}\}$$
$$\{\texttt{3fd4}\}^{-1} = \{\texttt{c4e2}\}$$
$$\{\texttt{3e0d}\} \times \{\texttt{cc22}\} = \{\texttt{cb52}\}$$
$$\{\texttt{3e0d}\}^{-1} = \{\texttt{114d}\}$$

{8c45}	{c3c5}	{906f}	{4601}	$\xrightarrow{MixColumns}$	{a8f4}	{cfaf}	{cf4a}	{3f58}
{cb2b}	{f025}	{aa8a}	{fba2}		{d115}	{d387}	{6fdc}	{4aca}
{3c16}	{8bcd}	{7fd6}	{f55b}		{8f11}	{6c5f}	{2af0}	{e462}

{4e23}	{fd5a}	{25ce}	{2c8}	$\xrightarrow{E}$	{3a2b}	{2807}	{ee3a}	{2c8e}
{d0d8}	{2c9c}	{b4de}	{d3ed}		{60}	{4389}	{db92}	{a064}
{df51}	{7b1}	{fef6}	{e2ac}		{e4fa}	{340c}	{cce9}	{1f3b}

{8c07}	{101e}	{f24f}	{e20d}	$\xrightarrow{D}$	{b6e3}	{60bd}	{3519}	{a1c9}
{391c}	{d98e}	{b0b7}	{ed7f}		{6184}	{f9c4}	{8c05}	{ffaa}
{f74c}	{7a37}	{78f}	{a1f6}		{45c9}	{a97}	{1d02}	{15eb}

Les vecteurs de test pour E et D sont données pour la clef maîtresse :

$$\{\texttt{6d3945e301b915a4a811d173d1fb66c25de3b33dbd8d2b02}\}.$$

Pour finir, voici un vecteur de test pour Mix :
$Mix(\{\texttt{d437e27cb55113d3ab9f7670c7b5401615eaa1fd40d2709e}\}, \{\texttt{4242}\}) =$
$\{\texttt{b8b22cd4928c3f1a2408d122939f67bcd9dc826cc1a2df6b}\})$