Alice au pays des block-ciphers

Alice, notre apprentie cryptographe favorite, va tenter de créer un block-cipher. Elle a étudié DES and AES, mais les a trouvé trop compliqués à analyser et à améliorer. Dans ses recherches, elle s'est rendu compte que la clé de DES était trop courte et que AES fournissait de bonnes clés. Elle a donc décidé d'utiliser une clé de 128 bits. Ensuite, elle a remarqué que plus le système a de tours, mieux c'est. Elle va donc choisir un nombre de tours supérieur à 10. Le nombre 12 lui semble être un bon choix. Ensuite, elle remarque qu'AES utilise deux de ses opérations favorites: xor et rotations (Si c est un tableau de bits, ROTR(c,n) fait tourner les bits de c de n positions vers la droite, et ROTL(c,n) fait la même chose vers la gauche); Ce sont donc de bonnes opérations pour construire un block cipher. Avec ces éléments en tête, elle a choisi de définir le key-schedule (Alg. 1) et le chiffrement (Alg. 3). Les constantes c_i and the valeurs r_i sont choisies au hasard, et sont fournies avec le cryptosystème (les c_i sont donnés en hexadécimal).

Déchiffrez le challenge (ciphertext, en hexadécimal aussi), et ce sera votre drapeau!!

Algorithm 1 key schedule

$k_0 \leftarrow k$
for $i = 1$ to 11 **do**
    if $i = 1 \mod 2$ **then**
        $k_i \leftarrow \text{ROTR}(k_{i-1}, r_{i-1})$
    else
        $k_i \leftarrow \text{ROTL}(k_{i-1}, r_{i-1})$
    end if
end for
return $[k_0, ..., k_{11}]$

Algorithm 2 Encryption

$s \leftarrow$ plaintext
for $i = 0$ to 10 **do**
    $s \leftarrow s \oplus k_i$
    $s \leftarrow \text{ROTR}(s, r_i)$
    $s \leftarrow s \oplus c_i$
end for
$s \leftarrow s \oplus k_{11}$
return s