A Brief Overview of A3S

pot

July 21, 2021

Abstract

A3S is a cipher inspired by AES and base 3. It was developed for the 2021 RaRCTF competition but it may be used again in the future. This document will be a brief overview of A3S and should give you some understanding so reading code is easier. However, it will not cover implementation of finite field arithmetic and such.

Contents

1 Definitions

Trit A unit having one of three values (0, 1, 2).

Tryte 3 trits.

Word 3 trytes.

LE Little-endian

BE Big-endian

RM Row-major order

2 Input and Output

A tryte array is needed but data given is usually in bytes. One way to convert is to and from an integer. The tryte array will be used as a matrix.

$$B_0, B_1 ... \xrightarrow{BE} I \xrightarrow{LE} T_0, T_1 ... \xrightarrow{RM} \begin{bmatrix} T_0 & T_1 & T_2 \\ T_3 & T_4 & T_5 \\ T_6 & T_7 & T_8 \end{bmatrix}$$

This process can be reversed for an output.

3 The cipher

3.1 The algorithm

Input: Plaintext P (Trytes)
Key K (Trytes)
Output: Ciphertext C
$K_{0 \cdots N} \leftarrow Expand(K)$
$C \leftarrow Apply(P, K_0)$
for $i \leftarrow 1$ **to** $N - 1$ **do**
  $C \leftarrow Substitute(C)$
  $C \leftarrow Shift(C)$
  $C \leftarrow Mix(C)$
  $C \leftarrow Apply(C, K_i)$
end
$C \leftarrow Substitute(C)$
$C \leftarrow Shift(C)$
$C \leftarrow Apply(C, K_N)$
return C

3.2 Substitution

Trytes are replaced using a table of values. For example, 1 could be changed to 16 during this step.

3.3 Shift rows

The trytes are rearranged. Different letters will be used to make this more easier to see.

$$\begin{array}{|c|c|c|} \hline A_0 & A_1 & A_2 \\ \hline B_0 & B_1 & B_2 \\ \hline C_0 & C_1 & C_2 \\ \hline \end{array} \longrightarrow \begin{array}{|c|c|c|} \hline A_0 & A_1 & A_2 \\ \hline B_1 & B_2 & B_0 \\ \hline C_2 & C_0 & C_1 \\ \hline \end{array}$$

3.4 Mix columns

Every column in the matrix will be written as a polynomial then multiplied by a constant in a polynomial ring (b).

$$\begin{aligned} f(A_{old}, B_{old}, C_{old}) &= constant * (C_{old} * b^2 + B_{old} * b + A_{old}) \\ &= C_{new} * b^2 + B_{new} * b + A_{new} \end{aligned}$$

The coefficients of the result with respect to a are used to replace the original values. For example, the location of C_{old} will now have the value C_{new}.

3.5 Round keys

The number of keys generated is represented as the following where x is the length of the tryte array. x also needs to be greater than 0.

$$\begin{aligned} f(x) &= \lceil x/3 \rceil + 3 \\ &= N \end{aligned}$$

The + 3 means extra keys are created compared to the original AES for added "security". Moving on, round constants are defined as the powers of a in the finite field.

$$\begin{aligned} f(x) &= a^x \\ &= rcon_x \end{aligned}$$

L will be used to represent the expanded key and K being the original key and M as its length. i will go from 0 to $3N - 1$ (Shamelessly stolen from Wikipedia). *Rot* moves the first tryte to the end and *Sub* applies substitution to all trytes. The *rcon* will only be applied to the first tryte.

$$L_i = \begin{cases} K_i & \text{if } i < M \\ L_{i-M} \oplus Sub(Rot(L_{i-1})) \oplus rcon_{i/M} & \text{if } i \equiv 0 \pmod{M} \text{ and } i \neq 0 \\ L_{i-M} \oplus L_{i-1} & \text{otherwise} \end{cases}$$

Once the key words are generated they are packed in 3s to produce a 3x3 matrix of keys.

$$W = [T_0 \ T_1 \ T_2]$$

$$\begin{bmatrix} W_0 \\ W_1 \\ W_2 \end{bmatrix} \longrightarrow \begin{bmatrix} T_0 & T_1 & T_2 \\ T_3 & T_4 & T_5 \\ T_6 & T_7 & T_8 \end{bmatrix}$$

Applying them to the plaintext is as simple as adding (in GF(3)) to their corresponding element.