Synopsis submitted in partial fulfillment of requirements for the degree of

Masters of Technology (Hons.)
In
Computer Science and Engineering

By

Nitin Bansal (03CS3015)

Under the guidance of

Prof. Dipanwita Roychowdhury

" YOGA KARMASU KAUSALAM "

Department of Computer Science and Engineering
Indian Institute of Technology
Kharagpur

A New Key Schedule Proposal

1. Introduction

The Advanced Encryption Standard (AES) is the most significant standard of the block ciphers, so its security is of paramount importance. However, the key schedule of AES has a clear weakness that directly assists the execution of most effective attacks. To combat these weaknesses, we propose a different approach to the AES Key Schedule design. We demonstrate that it avoids the weakness of the existing key schedule.

The analysis of weak key schedules has led to the guidelines for robust key schedule design that borrows from well known and accepted design principles for block algorithms in the broader sense. Our design follows these key schedule guidelines.

The goal of a strong key schedules is to overcome any perceived weakness which may be used in attacking the block cipher system .Designers already ensure Shannon's property of confusion and diffusion properties in their cipher algorithms, so similar properties could be achieved for key schedules algorithms.

Biham showed that in some simple cases, simple key schedules exhibit relationships between keys that may be exploited. Also Knudsen listed four necessary but not sufficient properties for secure Fiestel ciphers .Two of these *,no simple relation and all keys are equal good* ,are achievable with strong key schedules .The generic properties of a strong key schedule that are readily measurable are:

1): Function should be infeasible (or at least hard) to invert
2): Minimal mutual information (between all sub key bits and master key bits)

Property 1 ensures that given any round sub key it should be infeasible to get back the other round sub keys or master key just by inverting the functions used to get it.

Property 2 aims to eliminate bit leakage between sub keys and master keys, weakness that assists cryptanalysis by reducing the complexity of some attack scenarios on block ciphers. As some of the attacks make use of the relations between key bytes and would have a higher complexity if these relations did not exist.

Leakages of information from subkey i to subkey i-1 or subkey i+1 is directly prevented by Property 2. Using master keys directly in sub keys leads to the worst case of bit leakage; however this can be easily avoided.

2. Objective

In this work, we analyze the AES key schedule; discuss its security properties and weaknesses that assist the execution of effective attacks. We then propose and analyze a more efficient key schedule making use of features and properties provided by linear and non-linear Cellular Automata (CA). CA has been shown to be capable of generating complex and random patterns out of simple rules. Therefore, it has been used to provide randomness and nonlinearity to the key schedule proposal.

3. A New AES key schedule proposal

The aim of this section is to define a suitable key schedule which satisfies the desired properties outlined for key schedule.

Cellular Automaton (CA) has been shown to be capable of generating complex and random patterns out of simple rules. Moreover, they can be implemented efficiently in hardware .So it seems logical to include these in our key schedule design.

3.1. 128 bit key schedule Proposal

Proposal 1

```
// First we create Rconstant for every round of AES
   For r = 0 to 10 {
        For j = 0 to 15 {
             $rconstant_j = r* 16 + j$
        }
   }
```

- Rconstant is of 128 bits and equal to $rconstant_0 | rconstant_1 |$….$rconstant_{15}$
  This Rconstant is different for every round and while generating round subkey we will use round constant corresponding to that round.
  (Here $rconstant_j$ is of 8bits and | represents concatenation)

- The inclusion of round dependent round constant (Rconstant) eliminates the symmetry, or similarity, between the ways in which round keys are generated in different rounds. It not only isolates each resulting subkey from others, but also breaks up possible weak keys, for example, if all the master keys were identical.

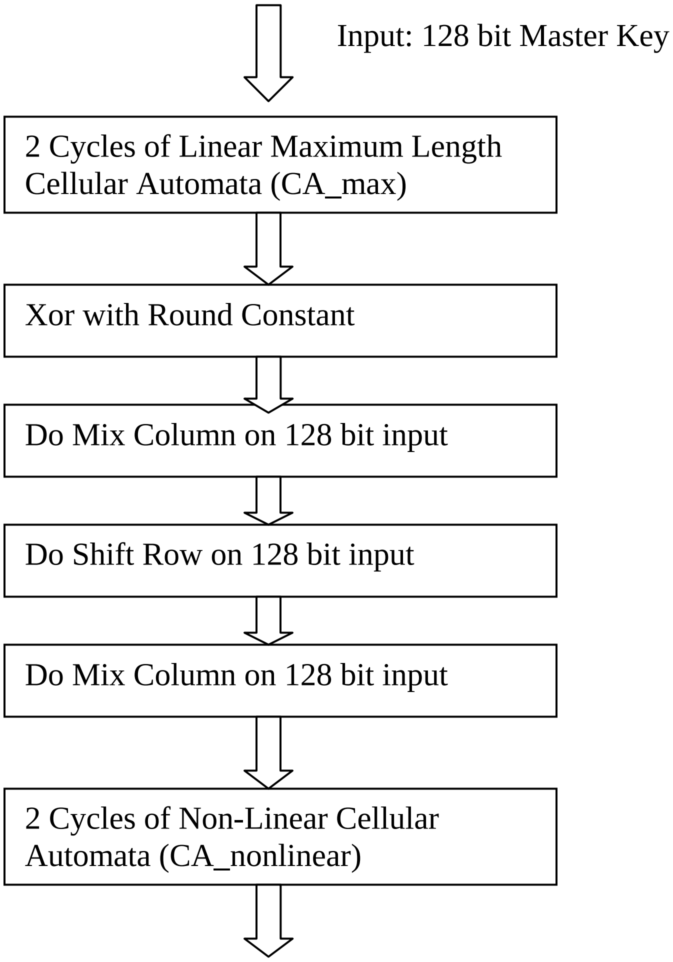

Input: 128 bit Master Key
2 Cycles of Linear Maximum Length Cellular Automata (CA_max)
Xor with Round Constant
Do Mix Column on 128 bit input
Do Shift Row on 128 bit input
Do Mix Column on 128 bit input
2 Cycles of Non-Linear Cellular Automata (CA_nonlinear)

Fig 3.1: Key schedule proposal for 128 bit keys

- Final round key bits are selected on the basis of Master key

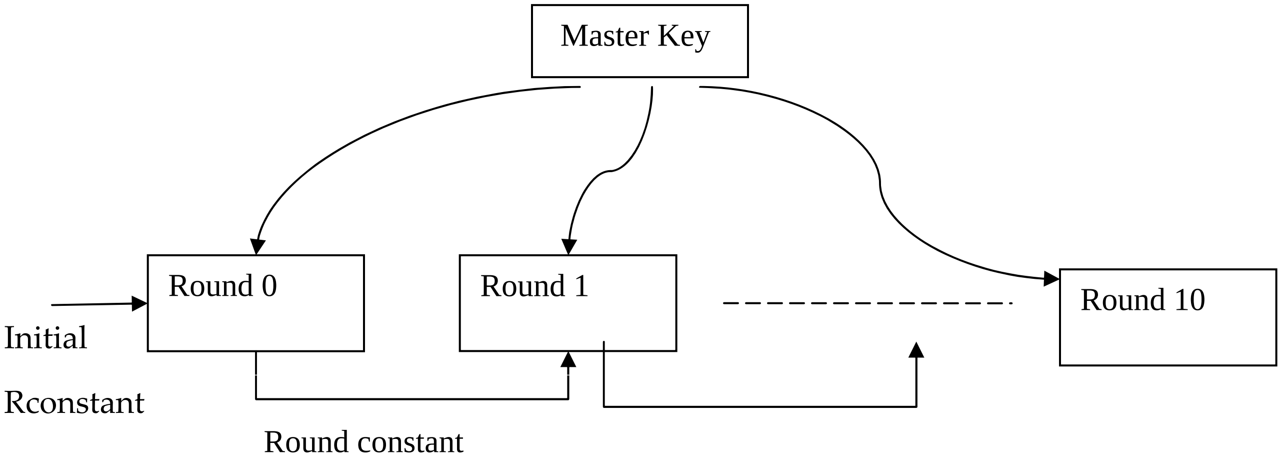
```
If Master key[i] = 1 then
    Subkey [i] =CA_nonlinear_output [cycle1] [i]
Else
    Subkey [i] = CA_nonlinear_output [cycle2] [i]
```

- CA_max uses 2 clock cycles of maximum length 128 bit CA where rules used are 90 and 150. We know that maximum length CA is random in nature. So this is used to provide randomness to the round sub keys.

- Mix column and shift row are the same operation used in AES round function and they are used to provide the required diffusion.

- CA_nonlinear is a 128 bit periodic nonlinear CA using rule 30. This is used to provide nonlinearity to the key schedule algorithm.

Proposal 2



In this proposal , round key of previous round acts as a round constant for the current round .However, the algorithm to generate round keys remain the same as one used in earlier proposal.

3.2. Security Analysis

Various tests were done to measure properties like diffusion and consecutive round key bit difference for both the proposals. Bit variance test was done to measure the uniformity of round key bits. So, based on those tests we found that proposal 2 satisfies most of the properties and hence we can conclude the following:

- As each round sub key is generated independently in the proposal, and, consecutive sub keys differ in half of bits there is no bit leakage. Also the master key is not directly used as sub key in the proposal.

- **One way ness** is achieved by using master key bits in the selection of sub key bits from the outputs of nonlinear periodic CA having rule 30. So even if cryptanalyst knows any particular subkey he will be not able to know from which CA output this particular bit was selected as he doesn't have master key with him. So every bit has 2 choices and which gives **2^128** cases to be considered and hence hard to reverse without the knowledge of master key. Thus this proposal overcomes the weakness of AES key schedule which can be inverted.

- Tests were formed to calculate the number of bit changes in the output with a single bit change in the input. We found that with a single bit change in input**, on an average half of the output bit changes which is a good measure of the Shannon's diffusion property.** Moreover, complete diffusion was achieved when we used 2 clocks of the Cellular automata used and 2 rounds of Mix columns, thus clarifying the decision for choosing 2 CA clock cycles and 2 mix columns in the proposed key schedule This is particularly useful in thwarting related key attacks, as altering even one bit in the master key changes approximately half the bits in each subkey.

- A generic attack solicits some round subkey bits by forceful means. In contrast to the current AES key schedule, even if an entire 128 round subkey is known, as proven, it is hard to retrieve the master. It is not possible to obtain subkey bits from one round using material purely from another.

Anonymous Authentication in VANETs

(This work has been done jointly with Umang Jain (03CS3005))

1. Introduction

Vehicular networks are likely to become the most relevant form of mobile ad hoc networks. Thus, security of these networks becomes paramount. Manufacturers are about to make a quantum step in terms of vehicular IT, by letting vehicles communicate with each other and with roadside infrastructure; in this way, vehicles will dramatically increase their awareness of their environment, thereby increasing safety and optimizing traffic. There are many aspects to vehicular communication in terms of implementation and inherent challenges. One of these challenges is security; very little has been devoted so far to the security of vehicular networks. Yet, security is crucial. For example, it is essential to make sure that life-critical information cannot be inserted or modified by an attacker; likewise, the system should be able to help establish the liability of drivers; but at the same time, it should protect as far as possible the privacy of the drivers and passengers.

2. Objective

In this work we have addressed the problem of entity authentication in VANET's. For every received message the recipient should be able to verify whether the information sent has been sent by an authentic sender or not. On the other hand it is also paramount that the identity of the sender is not revealed through his signature. Though if the need arises some trusted party or a law enforcement agency should be able to establish liability through the signature.

2.1 Infrastructure and Network Assumptions

The communicating nodes in VANETs are either vehicles or base stations (Road Side Units). The communications are assumed to be like those in DSRC (Dedicated Short Range Communication). All the vehicles broadcast. They can send to and receive messages from vehicles within 1000 meters distance. Messages are sent every 100ms. Messages can be event driven or periodic information about a vehicles position, speed etc.

3. State of the Art

More attention is being paid to the security of VANET's these days. Having said that, there is still no authentication algorithm that has been developed keeping VANET and its needs in minds. There have been many suggestions about security protocols for VANET's all of which seem to use already established and secure algorithms like RSA and ones based on these. But the major bottleneck in VANET is that vehicles are not expected to have huge computational or storage resources. Therefore, schemes like RSA or those based on Elliptic Curve Cryptography do not make for very suitable candidates for authentication in VANET. In this work we explore the possibility of using a group signature scheme for authentication in VANET. In the next section we propose a novel group signature schemes based on the Chinese Remainder Theorem.

4. Group signature scheme based on the Chinese Remainder Theorem

Let there are k group members.
Public Information (known to all members and manager)
N_G - A relatively prime number

Group Manager has following information:
N_o - Private relatively prime number (known only to manager) used to reveal identity of the message sender

Group Members:

Each member M_i is given the following information by the group manager.

N_i - A relatively prime number known only to M_i

a_i - A random number ($< N_i$) known only to M_i used for sign verification

$Pr_i = \prod (N_j)$ where j!=i

N_G - A prime number known to all members.

Each N_i is relatively prime to each other.

$CRTK_i$ which is created as follows:

$CRTK_i \bmod N_0 \equiv ID_i$

$CRTK_i \bmod N_1 \equiv a_1$

$CRTK_i \bmod N_2 \equiv a_2$

....

$CRTK_i \bmod N_k \equiv a_k$

$CRTK_i = < ID_i, a_1, a_2, a_3, a_4 \ldots a_k >$ (k Tuple) as in CRT

The modulus is taken with respect to N_0 and all other N_j.

All this information is available with a particular member.

Note: *$CRTK_i \bmod N_i \equiv a_i$ is not used in creation of $CRTK_i$*

4.1 Signature Generation

To send the message the member creates a signature Y in the following manner.

$Y \bmod Pr_i \equiv CRTK_i$

$Y \bmod N_G \equiv$ Hash (Message)

$Y = < CRTK_i$, Hash (Message) $>$

4.2 Signature Verification

To verify the signature a member M_j does the following -:

$X = Y \bmod N_j$

If ($X == a_j$) the signature is verified.

It is important to note that the verifier does not need to and cannot extract $CRTK_i$ of the sender, to verify the authenticity of the sender.

4.3 Identity Extraction

Only Manager will be able to reveal the identity of message sender by doing following operation:

$ID_i = Y \bmod N_o$

This ID_i then can be mapped to the actual identity of the sender.

Note: *$N_0, N_1, N_2, \ldots\ldots N_k, N_G$ they are all relatively prime to each other.*

4.4 Correctness

In order to verify the receiver does the following check -:

If($Y \bmod N_i == a_i$)

We have to prove that in case of an authorized sender, this check does stand to be true.

$CRTK_i = (\sum a_j * ((Pr_i/N_j)* (((Pr_i/N_j)^{-1} \bmod N_j)))) \bmod Pr_i$ -------------- (1)

Where j varies from 0 to k and j!=i

$Y = (CRTK_i (N_G * (N_G^{-1} \bmod Pr_i))+Hash<Message>(Pr_i*(Pr_i^{-1} \bmod N_G))) \bmod Pr_i*N_G$ -(2)

Let Z be a number such that

$Z \bmod N_0 \equiv ID_i$

$Z \bmod N_1 \equiv a_1$

$Z \bmod N_2 \equiv a_2$

....

$Z \bmod N_k \equiv a_k$

$Z \bmod N_G \equiv Hash<Message>$

Let $N_{k+1} = N_G$, $a_0 = ID_i$, $a_{k+1} =$ Hash<Message>

Let $P = Pr_i* N_G$

Therefore Z can be written as-:

$Z = (\sum a_j * ((P/N_j)* (((P/N_j)^{-1} \bmod N_j)))) \bmod P$ where j varies from 0 to k+1 and j!=i

$Z = (\sum a_j * ((P/N_j)* (((P/N_j)^{-1} \bmod N_j)))) \bmod P + a_{k+1} * (Pr_i * (Pr_i^{-1} \bmod N_G))\bmod P$,

j varies from 0 to k and j!=i

Let Z = Z1 + Z2, where Z1 and Z2 are the two terms in the above equation

$Z1 = (\sum a_j * ((Pr_i*N_G/N_j)* (((Pr_i*N_G/N_j)^{-1} \bmod N_j)))) \bmod P$

Since Pr_i is a multiple of N_j,

$N_G^{-1} \bmod N_j = N_G^{-1} \bmod Pr_i$

$Z1 = (((\sum a_j * ((Pr_i/N_j)* (((Pr_i*/N_j)^{-1} \bmod N_j)))) *(N_G*(N_G^{-1} \bmod Pr_i)))) \bmod P$ – (3)

Now we have from equation 1

$\sum a_j * ((Pr_i/N_j)* (((Pr_i*/N_j)^{-1} \bmod N_j)) = qPr_i + CRTK_i$ for some integer q ----- (4)

From (4) and (5)

$Z1 = ((qPr_i + CRTK_i)* N_G*(N_G^{-1} \bmod Pr_i))\bmod P$

$= (((qPr_i*N_G + CRTK_i * N_G) \bmod P * (N_G^{-1} \bmod Pr_i) \bmod P) \bmod P$

$= (((qP + CRTK_i * N_G) \bmod P * (N_G^{-1} \bmod Pr_i) \bmod P) \bmod P$

$= (((CRTK_i * N_G) \bmod P * (N_G^{-1} \bmod Pr_i) \bmod P) \bmod P$

$= ((CRTK_i * N_G* (N_G^{-1} \bmod Pr_i)) \bmod P$

$Z1 = ((CRTK_i * N_G* (N_G^{-1} \bmod Pr_i)) \bmod P$ --------- (5)

$Z = Z1 + a_{k+1} * (Pr_i * (Pr_i^{-1} \bmod N_G)) \bmod P$

$= ((CRTK_i * N_G* (N_G^{-1} \bmod Pr_i)) \bmod P + Hash<Message> * (Pr_i * (Pr_i^{-1} \bmod N_G)) \bmod P$

$Z=((CRTK_i * N_G* (N_G^{-1} \bmod Pr_i)+Hash<Message> * (Pr_i * (Pr_i^{-1} \bmod N_G)))\bmod Pr_i* N_G$

= Y from (2)

Therefore Y = Z

Hence,

$Y \bmod N_j = Z \bmod N_j = a_j$

5. Application to VANET

We assume the VANET to be divided into several groups with one trusted party (certifying authority) acting as a Group Manager for each group. The Group Manager thus is not a vehicle but some sort of a government agency. Each vehicle or member of a group shall be given public and private information by the Group Manager at the onset. Each of them can sign and verify messages as mentioned above.

5.1 Communication Overhead

Let each N_i's have size b bits and there are k members. CRTK's and Pr's will have size of the order of b*k bits. This poses a problem for large groups as CRTK and Pr will lead unacceptable size requirements. The overhead will be the size of Y.
For b = 80 bits and k = 10000
Overhead is of the order b*k bits = 100*8 kilo bites = **100 Kilo bytes**

5.2 Storage Overhead

We need to store following things for this proposal
CRTK, Pr , N_i and corresponding a_i and N_G
Order of Storage overhead =100 kilo bytes (CRTK) + 100 kilo bytes (Pr) + 160 bits (N_i's and corresponding a_i's) + 80 bits (N_G) = **200 kilo bytes.**

6. Conclusion

Evidently the overhead and storage requirements of the scheme are huge at the moment, even to the scale of being impractical. Intuitively the message signing and verification seems to be faster than those in RSA which involve exponentiation. But the huge size of the signature generated might well nullify this intuition as well. Further in this work we show that we are right in saying that the scheme presently is impractical both time-complexity and storage complexity wise. After that we modify the scheme to drastically reduce the amount of storage and overhead and we show that while signature generation time is comparable to that in RSA, the verification time is less which is significant as in VANET vehicles are likely to verify more often that they sign.