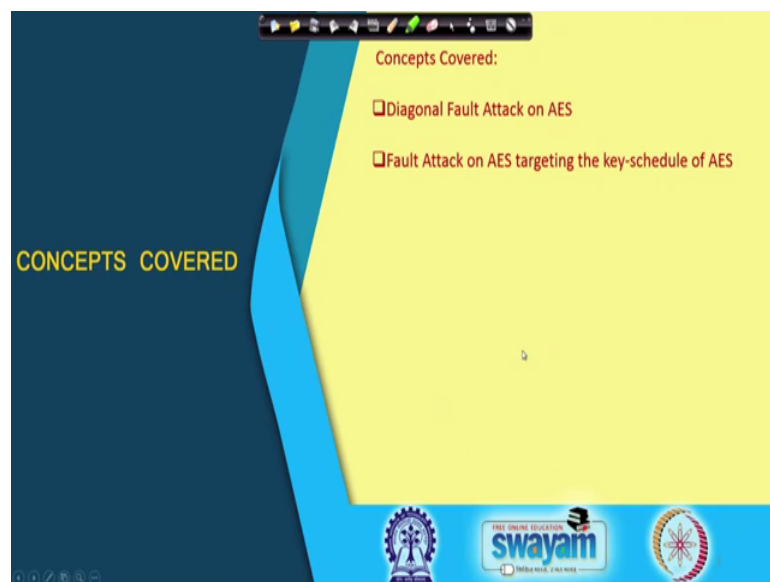


Hardware Security
Prof. Debdeep Mukhopadhyay
Department of Computer Science and Engineering
Indian Institute of Technology, Kharagpur

Lecture - 48
Multi – Byte and Key Scheduling Based Fault Analysis of AES

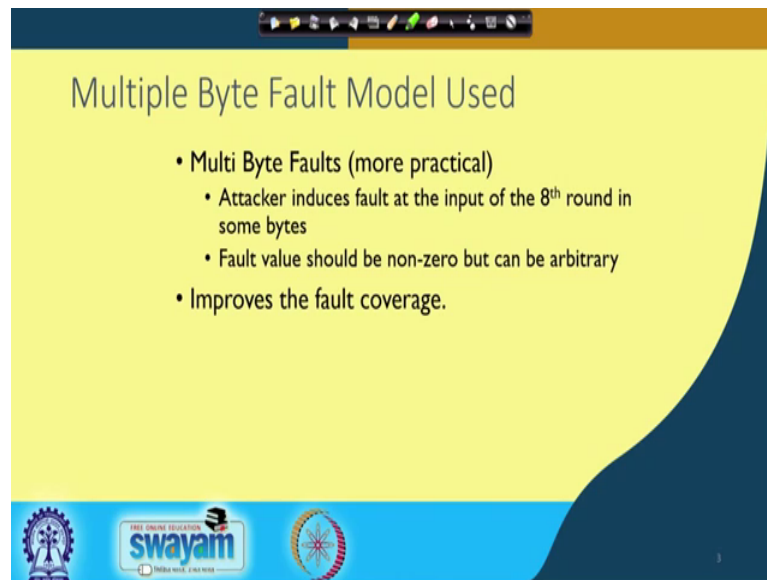
So, welcome to this class on Hardware Security, so today we shall be continuing on our discussions on Fault attacks in particular we shall be talking about multi byte fault attacks and also like fault attacks as targeted on the key scheduling. Like previously we have seen like fault attacks on AES where the data path was targeted. So, here we will see how to exploit a fault analysis where the target is the key schedule of AES.

(Refer Slide Time: 00:39)



So, therefore, we shall be talking about 2 types of fault attacks, the first one is called as diagonal fault attacks and the second one is where we target the key schedule of AES.

(Refer Slide Time: 00:47)



The slide features a yellow background with a dark blue curved shape on the right side. At the top, there is a navigation bar with various icons. The title 'Multiple Byte Fault Model Used' is centered at the top. Below the title, there are three bullet points. At the bottom, there are three logos: a gear-like logo on the left, the 'swayam' logo in the center, and a circular logo on the right.

Multiple Byte Fault Model Used

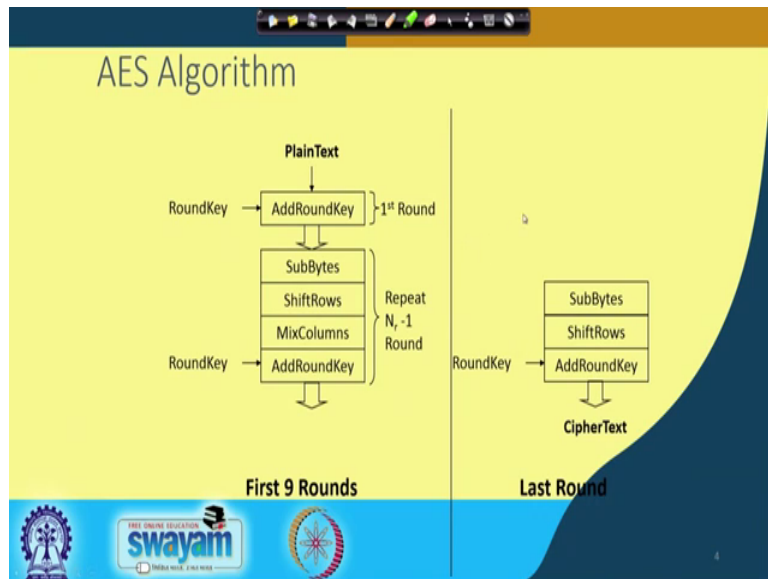
- Multi Byte Faults (more practical)
 - Attacker induces fault at the input of the 8th round in some bytes
 - Fault value should be non-zero but can be arbitrary
- Improves the fault coverage.

Logos at the bottom: A gear logo, the 'swayam' logo (with 'FREE ONLINE EDUCATION' above and 'INDIA WISE, LEAD WISE' below), and a circular logo with a red and white pattern.

To start with we have basically been considering what are called as single byte fault attacks. So, we have been considering the fault essentially the whether fault is restricted to a single bite of the state matrix of AES. But here we will be considering the fact suppose you know like when you are trying to induce the fault the fault propagates to more than 1 bytes and it basically kind of touches pretty a pretty much upon like 2 bytes are 3 bytes or more than one bytes in general.

So of course, the fault value should be a nonzero value but it can be any arbitrary value ok. So, what is the advantage of doing a fault analysis based on that, so it basically improves a fault coverage because as we know that when we are trying to induce a fault and if you can develop a fault attacks strategy which kind of which can encompass or which can exploit all possible types of faults right, essentially then we have got more and more coverage. So of course, as we will see that we will we are not able to get 100 percent fault coverage, but nonetheless we will try to extend the fault coverage of our existing fault analysis.

(Refer Slide Time: 01:48)



So, in this pursue right let us again take a quick recap on the AES algorithm. So, you have got since we are talking without loss of genetic with respect to AES 128, so we know that there are 9 rounds of the AES algorithm which we has got these blocks like AddRoundKey, the SubBytes, the ShiftRows, MixColumns and AddRoundKey and then there is the final round which essentially has everything but not the MixColumn. So, there is a MixColumn which is kind of removed from the first; from the final round.

(Refer Slide Time: 02:15)

Diagonal of AES State Matrix

Diagonal: A diagonal is a set of four bytes of the state matrix, where diagonal i is defined as follows:

$$D_i = \{b_{j, (j+i) \bmod 4} \ ; \ 0 \leq j < 4\}$$

According to the above definition and with reference to the state matrix of AES (refer figure) we obtain the following four diagonals.

b_{00}	b_{01}	b_{02}	b_{03}
b_{10}	b_{11}	b_{12}	b_{13}
b_{20}	b_{21}	b_{22}	b_{23}
b_{30}	b_{31}	b_{32}	b_{33}

D_0
 D_1
 D_2
 D_3

$\left\{ \begin{aligned} D_0 &= (b_{00}, b_{11}, b_{22}, b_{33}) \\ D_1 &= (b_{01}, b_{12}, b_{23}, b_{30}) \\ D_2 &= (b_{02}, b_{13}, b_{20}, b_{31}) \\ D_3 &= (b_{03}, b_{10}, b_{21}, b_{32}) \end{aligned} \right.$

So, now we will be defining the fault model in this case. So, it is essentially what is called as a diagnosis of the AES state matrix. So, what we will try to do here is considered the AES state matrix and then define sudden diagonals in this context ok. So

therefore, imagine that suppose the essentially these bytes for example, this is the state matrix of AES and consider these diagonal like $b_{0,0}$, $b_{1,1}$, $b_{2,2}$ and $b_{3,3}$ so these forms my diagonal D_0 .

On the other end consider these bytes like $b_{0,1}$, $b_{1,2}$, $b_{2,3}$ and along with it consider $b_{3,0}$. So, these together forms what is called as a diagonal D_1 like what is considered these diagonal like this and this ok. So, these 2 makes what is called as a diagonal D_2 and likewise consider this byte it is $b_{0,3}$ and $b_{1,0}$ $b_{2,1}$ and $b_{3,2}$, so these forms my diagonal D_3 . So, these are the 4 diagonals as stated over here D_0 , D_1 , D_2 and D_3 .

So now, we will basically try to you know like define a fault attack on AES based on these definitions of diagonals. So, we have till now we have been talking about the fault model, where the fault model was a random bite but now we will be trying to extend that fault model into what are called as diagonal fault models using these definitions or 4 diagonals in the AES state matrix. So now, let us see how this works so therefore I mean we would like to kind of we can of course you know like in typically, because of the regular structure of AES we can define the diagonal D_i in this compact form, where D_i is basically given by the bytes $b_{j,y}$ and $b_{j,y+i}$, where i are the module 4 in the index and y y j from 0 to 4.

So, you can really see that if i plug in say i equal to 0 then, that means like I am considering the bytes like b say $b_{0,j}$ plus i that will be that would be like $b_{0,0}$ ok. If i plug in the value of j equal to 1 here ok, so that means like I am considering $b_{1,0}$ and likewise $b_{2,1}$ and $b_{3,2}$. Likewise I can define for D_1 D_2 and D_3 ok, so this is the general and compact form of representing the diagonal.

(Refer Slide Time: 05:08)

Fault Models

Model - 0 Model - 1 Model - 2 Model - 3

- M0: One Diagonal affected.
- M1: Two Diagonals affected.
- M2: Three Diagonals affected.
- M3: Four Diagonals affected.

So therefore, I mean with this definition let us define the 4 fault models ok. So, the first fault model is what we call as where 1 diagonal is affected. So, you see that in this particular diagonal or in this particular model when we are talking about the M 0 essentially as one of the one of the fault models then only one diagonal is affected ok. That means, like remember the 4 diagonal so it may be that the fault is as you can see here is restricted to the diagonal D 0.

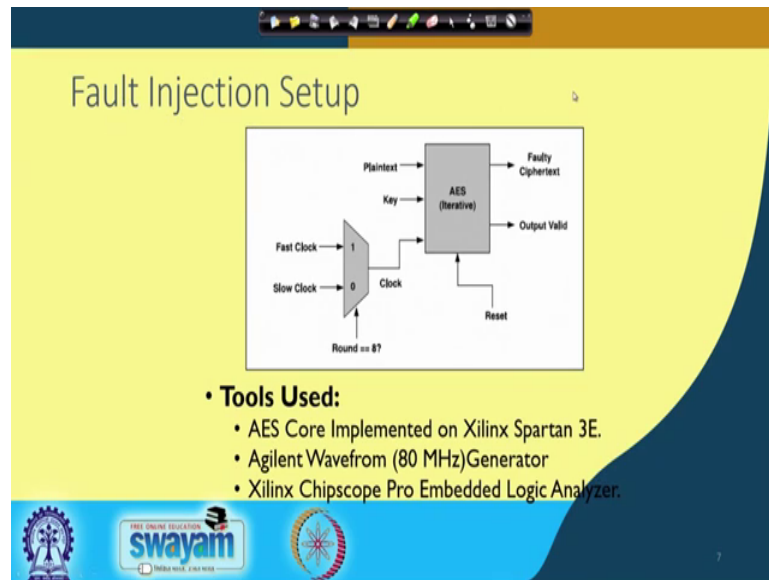
So, likewise we could have scenario where the fault propagates to 2 diagonals ok. So, there are 2 diagonals is there is essentially what we have, so you can see that here diagonal D 0 is effected and also we have got this diagonal which is effected ok, remember that these 3 bytes are this bite are in the same diagonal.

So, likewise your model M 3 or M 2 where there are 3 diagonals which are corrupted and in model M 3 pretty much in if or the 4 diagonals are corrupted ok. So, you can easily see that model M 3 is like universal because all possible faults in AES state matrix would imply model 3, because I am saying that all of your diagonals can also be corrupted. Also at the same time you can see that these models subsume each other, which means like M 0 right essentially it is kind of a sub part of M 1 because if I am saying that the fault is in 1 diagonal, that means like is within the fault model pertaining to model M 1 ok.

Because in model M 1 I am allowing 2 diagonals to be corrupted and it may happen that out of this to diagonals only 1 diagonal is corrupted. So, we can see that M 0 in the

subset of M 1 which is a subset of M 2 which is subset of M 3, so therefore like we have got this 4 fault models and now right we will try to see that whether we can exploit these possible fault models.

(Refer Slide Time: 06:52)



So, again this is our fault injection set up again it is kind of similar to what we have done previously. So, we have got the AES iterative hardware implemented and we are again kind of tinkering with the clock line ok, so we can again try to induce the fault in other mechanisms.

But consider that this is a simple and cost effective way of experimenting on the fault injection, where we basically try to change the clock frequency suddenly say when the clock when the when the 8th round starts or initiates, then we switch from our slow clock that means clock at which AES is operating correctly to suddenly a fast clock and this sudden switch right creates internal faults which are observed as the required faults.

So, what we do is have these faults are basically the observed by an embedded logic analyzer, so this is implemented using a Xilinx Chipscope Pro Embedded Logic Analyzer to inspect like how the internal false behave. So, what injecting the you know like the sudden block frequency we have used an arbitrary function generator and essentially right this is pretty much the implementation setup that we have.

(Refer Slide Time: 07:57)

Equivalence of Faults according to MO $[D_0 \setminus D_1 \setminus D_2 \setminus D_3]$

Faults induced in Diagonal D_0 at the input of 8th round are all equivalent.

swayam

So now, right with this setup we are now set to kind of understand how the fault propagates. So, in particular observe right that we are basically considering faults we are the fault is according to the fault model M_0 in this case ok. So, what does it mean? That means, like only one diagonal is corrupted so we know that there are 4 diagonals, so the fault could be either in D_0 or it could be either in D_1 or it could be in either in D_2 or it could be either in D_3 .

So, it could be any one but not 2 diagonals that is whole idea and if you consider here for example if you consider this state matrix you will see that only one byte in diagonal D_0 is corrupted. If you consider this particular you know like you will see the 2 bytes are corrupted likewise you see that our fault propagates 2 3 bytes and finally 4 bytes, but all of them ensure that the fault is restricted to one diagonal and in fact is a diagonal D_0 .

So now, let us see right and we have already kind of seen that how to do a fault attack when the fault is of this type, but the interesting thing that we see is now is that even these kind of faults are also similarly exploitable. So, remember that the fault location is that the input of the 8th round and because of that right if you observe the fault propagation you will see that this is what will happen is after the byte sub. So, here there is 1 byte so one byte gets corrupted here, here there are 2 bytes, so therefore the fault kind of spreads to 2 bytes and makes you know like objective transformation here the

fault is in 3 bytes. So, therefore we get a napping in 3 bytes, here there are 4 bytes and therefore we get a napping and 4 bytes.

And then this is followed by the shift row and in the shift row we know that in the first row there is nothing which happens, but in the second row third row and the second third and the final one in all of them what happens is the first column right is kind of getting disturbed. So, you see that the disturbance in all of them is restricted to the first diagonal ok. So, the first column on the input of the diagonal or 1 diagonal at the input of the 8th round is basically transformed into 1 column right at the end of the 8th round this is a property of the AES transformation.

Now because of these right when the MixColumn comes in right at this point then we know that all the because of the MixColumn property right the MixColumns has a specific property and because of that property we know right that at this point what we obtain is that we basically see that we get a disturbance right which is of this type.

And now what happens is that after byte sub we have got the fault which propagates which kind of is like F_1 F_2 and F_3 F_4 right and therefore what happens right after shift row that means after the 9th round shift row these bytes gets kind of distributed into 4 columns of the AES state matrix. And that implies that after MixColumns ok; after the MixColumns operation right what happens is that this F_1 gets transformed into these equations that is $2 F_1$ F_1 F_1 and $3 F_1$ and likewise they have got F_4 F_4 $3 F_4$ and $2 F_4$ and so on for F_3 and also for F_2 ok. So, this is because of the mix columns which take place at this points.

But the interesting (Refer Time: 11:16) here is that for all these faults that we have that means, the faults as long as the fault is restricted to the diagonal D_0 , the equations here remains and invariant it does not change, which means all these faults fall into the same equivalence class and they are equivalent. So, the faults induced in diagonal these are the input of 8th round are therefore all equivalent, so this is an important and interesting observation that we observed here.

So, as you can understand the now we can essentially kind of you know like and reduce the possible key size in a similar way, you will basically work behind as we are done previously and we will be trying to exploit these equations as we have done previously. And remember right previously what we have done is that for this case for a single byte

fault we have seen that a 2 power of 128 AES key because of one such fault gets reduced to 2 power of 32 possible values.

So, this reduction is possible even now, even if the fault propagates to more than 1 bytes but is restricted to 1 diagonal. In fact, you do not know which of the diagonals get or you may not know like which of the diagonals get corrupted and you can make a guess right; you can make a guess that there are 4 diagonals and if you paid at the fault you know like propagates according to the model M 0 you can miss any guests which of the diagonals has got corrupted. And if you guess that then that would imply that you basically have about either of these 4 situations.

(Refer Slide Time: 12:48)

Inter-relationships depending on the Diagonals affected

Diagonal Affected	Corresponding Column Affected	Resulting Byte Interrelation
D0	Column 0	$\begin{matrix} F1 & F4 & F3 & F2 \\ F1 & F4 & F3 & F2 \\ F1 & F4 & F3 & F2 \\ F1 & F4 & F3 & F2 \end{matrix}$
D1	Column 1	$\begin{matrix} F2 & F1 & F4 & F3 \\ F2 & F1 & F4 & F3 \\ F2 & F1 & F4 & F3 \\ F2 & F1 & F4 & F3 \end{matrix}$
D2	Column 2	$\begin{matrix} F3 & F2 & F1 & F4 \\ F3 & F2 & F1 & F4 \\ F3 & F2 & F1 & F4 \\ F3 & F2 & F1 & F4 \end{matrix}$
D3	Column 3	$\begin{matrix} F4 & F3 & F2 & F1 \\ F4 & F3 & F2 & F1 \\ F4 & F3 & F2 & F1 \\ F4 & F3 & F2 & F1 \end{matrix}$

The fault has either happened because of D 0 or D 1 or D 2 or D 3 and depending upon that right you have got 4 sets of equations ok. So therefore, right you have to basically kind of guess either of these things and you have to basically kind of solve for any of these equations and that actually means that you can actually you know like reduce the AES possible key size to you know like.

(Refer Slide Time: 13:08)

Equations if Diagonal D_0 is affected

$$CT = \begin{pmatrix} x_1 & x_2 & x_3 & x_4 \\ x_5 & x_6 & x_7 & x_8 \\ x_9 & x_{10} & x_{11} & x_{12} \\ x_{13} & x_{14} & x_{15} & x_{16} \end{pmatrix} \quad CT' = \begin{pmatrix} x'_1 & x'_2 & x'_3 & x'_4 \\ x'_5 & x'_6 & x'_7 & x'_8 \\ x'_9 & x'_{10} & x'_{11} & x'_{12} \\ x'_{13} & x'_{14} & x'_{15} & x'_{16} \end{pmatrix} \quad K_{10} = \begin{pmatrix} k_1 & k_2 & k_3 & k_4 \\ k_5 & k_6 & k_7 & k_8 \\ k_9 & k_{10} & k_{11} & k_{12} \\ k_{13} & k_{14} & k_{15} & k_{16} \end{pmatrix}$$

$$ISB(x_1 + k_1) + ISB(x'_1 + k_1) = 2[ISB(x_8 + k_8) + ISB(x'_8 + k_8)]$$

$$ISB(x_8 + k_8) + ISB(x'_8 + k_8) = ISB(x_{11} + k_{11}) + ISB(x'_{11} + k_{11})$$

$$ISB(x_{14} + k_{14}) + ISB(x'_{14} + k_{14}) = 3[ISB(x_8 + k_8) + ISB(x'_8 + k_8)]$$

- There are in total 4 such systems of equations for a diagonal D_0 .
- Each system of equation gives 2^8 keys on an average.
- AES key size gets reduced to 2^{32} .
- If the attacker does not know which diagonal is affected, then key size is $4 \cdot 2^{32} = 2^{34}$.

So, in this case right you can again you know like take the cipher text as he already shown that suppose you know like the you know the diagonal D_0 is affected and this is your cipher text and this is a faulty cipher text and this is your corresponding 10th round key. So, what you do is that you basically kind of rewards back and target the quartets and kind of you know try to solve each of these quartets using or solving a similar kind of equation as we have previous received ok.

So, if we remember right previously we basically exploited the fact that these differentiators are like toys this is the same as this one and this is a toys of this one. So, we kind of form similar set of equations and these exactly like this is toys this one is equal to this one and this is toys this one and we basically solve them.

So therefore, we know that they are important 4 such system of equations for the diagonal D_0 and in general like you would expect them AES key size would get reduced to around 2 power of 32 values. And right if you know that there as you know like 4 possible diagonals which can get corrupted then that would mean that even if we allow that I do not know which is the exact diagonal, then that would mean that the key size would be for multiplied by 2 power of 32 which is essentially 2 power of 34. So, now you know like you may wonder that what happens right if the fault kind of propagates to more than 1 diagonal.

(Refer Slide Time: 14:29)

Fault Injected across 2 Diagonals (Fault Model M_1)

8th Round

9th Round

Per column there are 2 bytes at any disturbance.

Invariant for any fault injected within diagonal D0 and D1

$a_0 = 2F_1 + 3F_6$	}	$a_1 + a_3 = a_0$
$a_1 = F_1 + 2F_6$		$2a_1 + 3a_3 = 7a_2$
$a_2 = F_1 + F_6$		
$a_3 = 3F_1 + F_6$		

So this is the scenario that will basically cover next. So remember that in the previous case right if you want to kind of do the attack, then the attack essentially has got you know like because you are reducing the key size 2 power of 32 values. So, in this case also right you will see that there is an information theoretic reduction of entropy, for example light we are considering now faults which are propagating according to 2 diagonals, so that means, right either say you know like.

So, in this case I have shown for diagonal D 0 and said diagonal D 1, so you can see that there are 2 diagonals which are corrupted. So, that could happen I do have scenarios like this where maybe you know like even 8 bytes out of 16 bytes are corrupted by faults ok. So, for each of them right therefore if there are 2 diagonals which are corrupted or the input of or 2 diagnose which are corrupted at the input of the 8th round that implies that there are 2 columns which are effective at this point.

Again because of the mix columns right because of the after shift rows you have a situation like this where you can see that the 2 diagonals here are the 2 columns here are kind of spread across in the state matrix ok. In the first rows I there is no shift here, in the second row there is a surplus left shift, so therefore this F 6 comes here as F 2 and F 2 comes here are the as here, so likewise in F 3 and F 7 right they all get propagated in this fashion.

So, what is you know like property here? So, property here is that per column right per column there are there are at max 2 disturbances right there are max to disturbances or 2 bytes there at max 2 byte disturbances. So therefore now if I apply the mix columns I get something like this I get $2 F 1$ plus $3 F 6 F 1$ plus $2 F 6 F 1$ plus $F 6$ and $3 F 1$ plus $F 6$, so that implies that in the pre like unlike the previous case and actually somewhat similar to the previous case.

So, that you see that there is a similarity and there were dissimilarity ok. So, what is the similarity in the similarity right is this that in the previous case the column had 8 bit of entropy, because there was only 1 byte or 1 independent variable if you get that right, then the remaining parts the remaining column gets fixed.

In this right you have got 2 bytes which are independent, for example $F 1 F 6$ are independent the moment you get a $F 1$ and $F 6$ this particular column gets fixed ok. So that means, there is 16 bits of entropy and therefore right you will see that you what you can do is you can now set variables like in the previous case like I said a 0 a 1 a 2 a 3 to all these 3 values are all these 4 values like $2 F 1$ plus $3 F 6 F 1$ plus $2 F 6$ and $F 1$ plus $F 6 3 1$ plus $F 6$ these are all differential equations and you can actually eliminate you know like $F 1$ and $F 6$, because there are 4 equations now and there are 2 variables right.

So therefore you can eliminate them and you form 2 equations and as you can see right this equation is in terms of a 1 a 3 and on the right you have got a 0 and a 2. So, therefore now what you can basically do is that you can take these equations and using these equations you can try to solve the; you know you can try to solve the AES key ok. Again as you observe that because of this reduction in entropy of this column you can use the possible entropy of the key ok. So, how so what does that lead to?

(Refer Slide Time: 18:15)

Equations if Diagonals D_0 and D_1 are affected

$$a_0 = ISB(x_1 + k_1) + ISB(x'_1 + k_1)$$
$$a_1 = ISB(x_8 + k_8) + ISB(x'_8 + k_8)$$
$$a_2 = ISB(x_{11} + k_{11}) + ISB(x'_{11} + k_{11})$$
$$a_3 = ISB(x_{14} + k_{14}) + ISB(x'_{14} + k_{14})$$

- The equation reduces the space of the 4 key bytes of AES to 2^{16}
- Two faulty ciphertexts reduce it to a unique value on an average (experimental result).

swayam

So you can observe here that in similar or similar analysis you can write equations as if this forms like a 0 a 1 a 2 a 3 these are you know like again the inverse of byte when you are again going back. So, you basically essentially you know like come to this column so that is a 0 a 1 a 2 and a 3 and that is written over here ok.

As previously right therefore, you can now relate this a 0 a 1 a 2 and a 3 by these 2 equations and that would imply that you are what you know like what you are basically doing is you are reducing AES key size to now to power of 6. So, this is there are 4 key bytes involved here and because of this entropy reduction that gets you know like converted into or reduced to 2 power of 16 values, because you have 60 bits of entropy. So therefore, I basically reduce it to 2 power of 16 values but then there are 4 key quartets, so that means that total key size now diminishes to 2 power of 64.

So, if you compare this with the previous case like when you have got only 1 diagonal affected and you know like this diagonal then the total key size is reduced to 2 power of 32. But here when you go to diagonals corrupted then the key size reduces to 2 power of 64. So, again you know like if you get to such faulty cipher text then you can again intersect them and you can pretty much reduce it to a unique value.

So, so the idea is that you know like but at the same time you have to keep in mind that you are do you know like you have to basically absorb like 2 power of 64 values and then taken can possible intersection which may not be very practical ok. On the other

hand when the fault is kind of restricted to be a one single diagram then the attack is more practical because you are essentially able to open it with 2 power of 32 values quite easy. But on the other hand right if this also shows that you know like theoretically there is a leakage and there is something that is not very comfortable from the designer point of view.

(Refer Slide Time: 20:10)

Fault Injected across 3 Diagonals (Fault Model M_2)

After Mix Column			
$2F_1 + 2F_6 + F_{11}$	$F_1 + 2F_6 + F_{11}$	$F_1 + F_6 + 2F_{11}$	$3F_1 + F_6 + F_{11}$
$F_1 + 2F_6 + 2F_{11}$	$F_1 + F_6 + F_{11}$	$F_1 + F_6 + 2F_{11}$	$F_1 + 2F_6 + F_{11}$
$F_1 + F_6 + 2F_{11}$	$F_1 + F_6 + F_{11}$	$F_1 + F_6 + 2F_{11}$	$F_1 + 2F_6 + F_{11}$
$2F_1 + F_6 + F_{11}$	$F_1 + 2F_6 + F_{11}$	$F_1 + F_6 + 2F_{11}$	$3F_1 + F_6 + F_{11}$

$$\begin{cases} a_0 = 2F_1 + 3F_6 + F_{11} \\ a_1 = F_1 + 2F_6 + 3F_{11} \\ a_2 = F_1 + F_6 + 2F_{11} \\ a_3 = 3F_1 + F_6 + F_{11} \end{cases} \rightarrow \begin{cases} 11a_0 + 13a_1 = 9a_2 + 14a_3 \end{cases}$$

$3 \times 8 = 24 \text{ bits}$

$\frac{1}{8} \times 2^{24} = 2^{17}$

So, likewise what happens when the fault propagates to 3 diagonal, so here is a fault model M 2 and as you can see right now we are considering scenarios where there are like 3 diagonals which are getting corrupted and it can therefore you know like lead to a scenario like this where you can see that 12 bytes out of 16 bytes are corrupted. Again if there are 3 diagonals which are corrupted then that means, that here you have got like you know I hear the you have got all the 3 the 3 columns which are corrupted and here it means like that every column if you see right because of the shift row there are at was 3 bytes which are corrupted ok.

So, like previously right here at most 3 bytes are corrupted ok, so now if you apply mix columns so again you will get some relationships like as you can see among these 4 bytes and as you can observe now there are 3 independent bytes and that would imply that now you have got you know like 3 individual bytes which are of you know like of consequence and therefore like the entropy is now you know like 3 into 8 that is essentially 24 bits per key quartets ok.

So, if you absorb for example this column here in this column has gotten entropy of 24 bits and that you can easily see that if I write each of these values like as a 0 a 1 a 2 and a 3 and then I can solve right. I basically get only one single equation and essentially write this equation getting satisfied there is a probability of 1 by 2 power of 8 and another way of seeing this is that the probability of a random selection of a 0 a 1 a 2 and a 3 satisfying this equation is 1 by 2 to the power of 8 and I know that totally there are 2 to the power of 32 values. So, I get 2 power of 24 so this is another way of looking at it. So now like I mean so therefore right what is the implication of these on the key size.

(Refer Slide Time: 22:09)

Equations if D_0 , D_1 and D_2 are affected

$$a_0 = ISB(x_1 + k_1) + ISB(x'_1 + k_1)$$

$$a_1 = ISB(x_8 + k_8) + ISB(x'_8 + k_8)$$

$$a_2 = ISB(x_{11} + k_{11}) + ISB(x'_{11} + k_{11})$$

$$a_3 = ISB(x_{14} + k_{14}) + ISB(x'_{14} + k_{14})$$

- The equation reduces the space of the 4 key bytes of AES to 2^{24}
- Four faulty ciphertexts reduce it to a unique value on an average (experimental result).

So, again you can form similar equations as previously similar differential equations by writing each of these column values as a 0 a 1 a 2 and a 3 and then kind of solving them as we are seeing right. We know that is a 0 a 1 a 2 a 3 are kind of dependent. So, what I do is basically I guess this key values given k_1 k_8 k_{11} and k_{14} is guessed here and then we try to see that whether they satisfy this equation.

As you can see right that out of 2 power of 32 values in this case right because of the relationship there will be 2 power of 24 values will satisfy this equation ok. So, what you can probably do is that you can take 2 power of 24 values and right, you can actually you know like try multiple attempts ok. So, what you can do is that you can kind of intersect them and you can reduce the possibility size.

So, for example here what I can do is that I can take 440 cipher text and I can you know like try to kind of intersect them and apparently in that case right I get a unique value for these key quartered. But at the same time like with a single fault injection right it would imply that I have got 2 power of 24 whole to the power of 4 that is 2 power of 96 as my key size which is kind of quite big. But at the same time it is less than 2 part of 120 so it is a theoretical it is an attack, but at the same time that you may not be very practical with a single fault injection ok.

So, on the other hand what you can do is that you can use each of the quadrants by trying multiple attempts right and recover each of the key quadrants individually.

(Refer Slide Time: 23:36)

Experimental Results

Clock Frequency (MHz)	No Fault	Model 0 (M0)	Model 1 (M1)	Model 2 (M2)	Model 3 (M3)	37.9	38.0	38.1	38.2	38.3	38.4	38.5	38.6	38.7	38.8	38.9	39.0	39.1	39.2	39.3	39.4	39.5	39.6	39.7	39.8	39.9	40.0	
36.0	512	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
36.1	512	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
36.2	512	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
36.3	510	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
36.4	511	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
36.5	508	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
36.6	504	8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
36.7	507	5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
36.8	490	22	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
36.9	488	28	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
37.0	419	79	14	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
37.1	448	60	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
37.2	434	64	15	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
37.3	408	94	15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
37.4	408	99	5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
37.5	248	228	38	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
37.6	214	205	84	9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
37.7	128	208	122	57	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
37.8	76	180	133	123	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
37.9	20	122	145	225	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
38.0	138	191	129	34	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
38.1	27	116	185	185	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
38.2	40	127	198	147	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

ATTACK REGION

So, let us see what happens or what is the impact of this finding in on an experimental basis, so again you know like if you remember this is my you know like basic setup and imagine that the AES operates at 36 megahertz ok. So, what we do is that we tried 512 fault injections by using the clock generator, why 512 because that is a limit of the embedded logic analyzer ok. So, we then take it statistical analysis of how the faults distribute across the models M 0 M 1 M 2 and M 3.

So, what we observed that initially when the clock is operating and it is correct frequency range like 36 megahertz or maybe slightly around that there are no fault, there are no faults and that is why we get M 0 M 1 M 2 M 3 all of them having 0 values, that

means there are no faults which are going into these pockets. At the same time there are 512 cases where there are no faults that mean all of them are resulting in no faults.

When I am trying to increase the frequency for example I make it a 36.3 megahertz I start to get the first signs of faults, as you can see the number of no faults is now reduced to 510 and I get to force which follow they all belong to the model M 0. That means, I get faults were 2 where the fault is restricted to 1 diagonal it may be into bytes, but the fault is kind of is remaining is not going across diagonals.

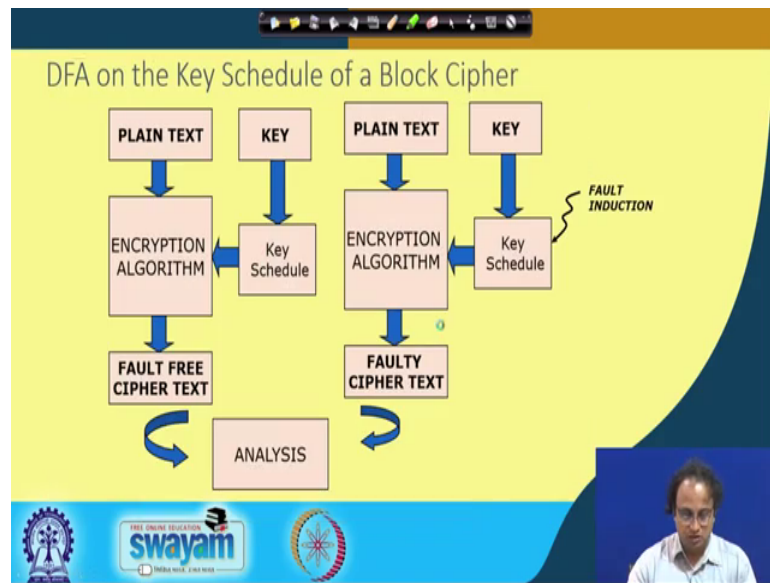
On the other hand where I start increasing the clock frequency what I observed is that I get faults also in the M 1 category, which means like there are more than 1 diagnosis which are now getting corrupted. And likewise when I start to increase the clock frequency farther I have situations where the fault propagates according to model M 2 and also when you know like I increasing farther that I get all the 4 diagonals which are corrupted, that means a faults are now according to the model M 3.

So, as you have seen right or as a finding or as a result of the of this discussion that we had now you can actually exploit till this point, where you can see that the faults are not according to the model M 3. As long as a faulty restricted the model M 2 that means you know like 3 diagonals out of 4 diagonals are corrupted (Refer Time: 25:38) with the you know like with the more faulty injections we can still try to kind of reduce the AES key size. And in fact we can do that pretty much with one single fault even when the fault is restricted to only 1 diagonal.

That means, when I am you know like still operating at a region where the M 0 faults are dominating. So therefore, and the interesting thing is that this particular characterization can be done per device and it seems like that even if you kind of shut down and again we start the machine right, you again get a similar kind of characteristic ok. So, you can basically study a platform and based upon that you can try to find out what will be your ideal clock frequency you know like selection. So, where you get faults where maybe the faults are in this case right and according to model M 0 with more probability.

But at the same time right even if the fault propagates to more than 1 bytes but as long as it kind of is restricted into one single diagonal you can still apply these techniques.

(Refer Slide Time: 26:35)



So, we basically I mean at this point we are kind of done where we are seeing how to kind of target the fault you know like the fault or the fault targets the AES data path. What we will see next is that what happens you know like where the faults target the key schedule of a block cipher ok. So therefore, we have been targeting only the data path now, but we will now investigate on what happens when the fault is kind of touching on the key schedule of AES ok. But that we will take up in the next class.

Thank you.