

**Information Security - 5 - Secure Systems Engineering**  
**Professor Chester Rebeiro**  
**Indian Institute of Technology, Madras**  
**Power Analysis Attacks**

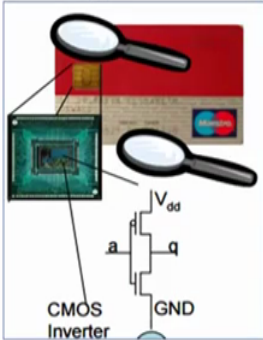
Hello and welcome to this video lecture in the course for secure systems engineering. In this video lecture we will talk about something known as power analysis attacks. So the essential idea over here is that when a device is actually computing something on based on some secret information, what the attacker would do is that the attacker would tap the power lines of that device and monitor the power consumed by that device, using this power consumption, the attacker then would be able to identify the secret information that is being processed by that system.

So we will start of this video lecture with why this is such a problem and we would see a few techniques about how such a problem can be handled and finally we will look at some counter measures for power analysis attacks.

(Refer Slide Time: 1:12)

---

## CMOS Technology



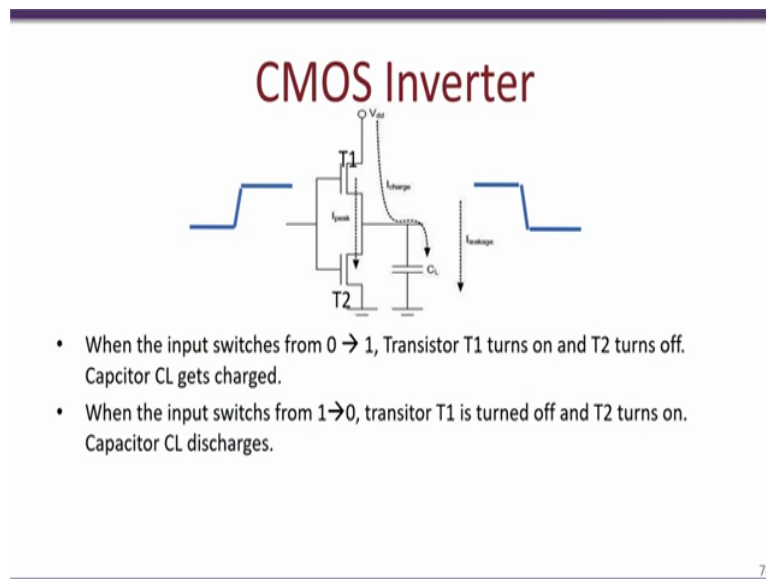
- Almost every digital device is built using CMOS technology.
- CMOS – complimentary metal oxide semiconductor

77

---

Just start out with some basic fundamentals about CMOS technology. Now every digital device that is being used today works on CMOS technology, atleast CMOS is a complementary metal oxide semiconductor technology and I would not say every but I think most of the electronic devices work with the CMOS technology, CMOS would actually be used to actually build a fundamental gates. Like for instance this gate over here uses a PMOS and an NMOS transistor coupled together to form an inverter.

(Refer Slide Time: 1:52)



So we will see little more detail about what a CMOS inverter looks like. So let us look a little more in detail about a CMOS inverter. So this is the CMOS inverter, it comprises of two transistors T1 and T2 and a load capacitor C. So when the input is at 0, the transistor T1 is ON and transistor T2 is OFF and as a result the capacitor gets charged through this  $V_{dd}$  thus we have the output which is 1.

On the other hand when we have an input which is set to 1, the transistor T2 turns ON, while transistor T1 would turn OFF, as a result this capacitor would then discharge through this transistor T2 leading to a output which is 0. So as we see from this particular figure when there is a transition in input from 0 to 1, the output changes from 1 to 0 due to the charging and discharging of the capacitor.


(Refer Slide Time: 3:04)

## Power Consumption of a CMOS Inverter

- Power is consumed when CL charges or discharges (i.e. there is a transition in the output from 0 → 1 or 1→0)
- Using an oscilloscope we can measure the power to determine when the inverter output changes state

Output of inverter

Power consumption



79

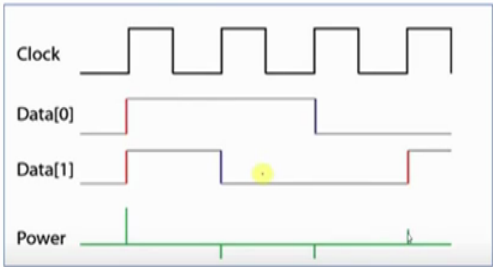
Now if we actually look at the power consumed by the CMOS inverter, it would look something like this. So power is consumed every time the capacitor CL either charges or discharges. In other words every time there is a transition in the output from 0 to 1 or 1 to 0, there is some power consumed by the device. So for example over here we will look at the output of the inverter and what we see is that during this transition from 0 to 1 in this particular case there is some power that gets consumed.

Now if we are able to tap into the power lines of the device we would be able to actually monitor the power consumed by the device using an oscilloscope, so the oscilloscope will give us the dynamic power consumption of the device.

(Refer Slide Time: 4:00)

## Synchronous Digital Circuits

- Most electronic equipment use a clock as reference
- All state transitions are done with respect to this clock
  - Power consumption is therefore at clock edges



80

Now most electronic circuits are actually synchronous digital circuits, these circuits have a clock as input and most of the activity goes on when the clock transitions. So for example over here we have a clock which is sent to a particular device and we have actually showing two inputs data 0 and data 1. So what we see is that every time the clock transitions, it is likely that the data 0 and data 1 may transition.

So in this particular case we are considering the positive edge of the clock and we see that only on this positive edge it is likely that the data actually transitions. So therefore when we actually look at the power consumed by this device we would see that the power would be a function of the type of transitions that happen. So for example over here we have data 0 and data 1 and both of them transition from 0 to 1 in this particular clock pulse and as you have seen in the previous slides during this transition there is a power that is consumed to charge and discharge the capacitor and therefore the total power consumed by the device is the sum of both these transitions.

So we see a high peak in power consumed over here due to the charging of the capacitor in each of these cases. Now in this particular clock pulse when there is a transition from 0 to 1 in this particular clock pulse what we see is that there is no change in data 0 but data 1 transitions from 1 to 0 and as a result if we connect this to the transistors what we have seen in the previous slide, the capacitor would be discharged and the power consumed would be actually negative because of the discharging of the capacitor.

Similarly in this particular transition we have data 0 which is moving from 0 to 1 and therefore the discharging of the capacitor shows up in the power consumption. Now over here again we have data 1 moving from 0 to 1 and the power is used to actually charge the capacitor. So what we notice over here is that first the charging and the discharging of the capacitors which are present in the various gates present within the device shows up in the power consumed.

Secondly what we also see is that the amount of power consumed depends on the number of gates that amount of power consumed depends on the number of gates that is actually making the transition. For instance over here we have two gates making the 0 to 1 transition and therefore the power consumed is quite high because we have both the capacitors getting charged, while in this particular clock pulse we have just one of these lines going from 0 to 1 and therefore the power consumed is comparatively lesser.

(Refer Slide Time: 7:08)

---

## Essence of Power Analysis

- We don't know what is happening inside the device, but we know the power consumption
- Can we deduce secret information from the power consumption

81

---

Now the essence of power consumption attacks is the following. An attacker assuming that he has a device in his position would not be able to look at individual transitions of every signal that is present within the device and therefore for him all of these inter media transitions which we have seen in the previous slide is completely black (7:29), what the attacker would have is possibly a source for the clock he would definitely be able to monitor the power consumed by the device, this is possibly because every device requires an external battery or power source for it to function.

So what the attacker could do is that he could actually connect an oscilloscope and tap out the power consumed by the device and therefore as the device is executing the attacker would be able to monitor the amount of power consumed by that device. The clock is an optional feature, it may or may not be possible for an attacker to always monitor the clock source for a device, every device would possibly have an external crystal oscillator which generates a clock and once this clock enters into the device there may be some operations on that particular clock source which multiply or divide that particular clock frequencies.

So thus the clock source may not always be visible to an attacker, but definitely every device since it would require an external power source therefore an attacker would be able to monitor dynamically the power consumed by that device. The essential idea about a power analysis attack is for the attacker to monitor the power consumed by the device and then be able to predict some internal secrets which are present in the device, needless to say what is required is that the device is actually operating on that particular secret.

For example a very popular application of power analysis attacks is on cryptographic ciphers and the assumption is that we have the attacker has in his position a device which is doing cryptographic operations, the key which is kept secret is stored within the device. Now every time an encryption or a decryption gets triggered, this key is read from the internal storage and use to actually do the corresponding encryption or the decryption.

The objective of the attacker is to monitor the power consumed by this device during the encryption or decryption process and then try to identify what the secret key is stored within the device is, so most of these power analysis attacks make the assumption that the attacker can actually manipulate or monitor the input messages that get encrypted or the output of that device for example the encrypted messages are visible to the attacker.

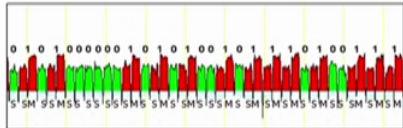
Thus, the major assumptions for a power analysis attack are the following, first the attacker has the device that he wants to actually attack so the attacker can actually power ON this device, he can monitor the various inputs or the outputs from that device and actually he is able to tap into the power lines and during the process of encryption and decryption the attacker should be able to monitor the amount of power consumed by that device using an oscilloscope.

(Refer Slide Time: 11:00)

---

## The Types of Power Analysis

- SPA : Simple Power Analysis



- DPA : Differential Power Analysis  
Requires more strategy and statistics to glean secret information
- Template based attacks

---

82

So there are various types of power analysis attacks ranging from very simple or the simple power analysis, two differential power analysis which is far more difficult and also far more difficult to protect and finally the template based attacks which is the most powerful form of power analysis attacks. Now in the simple power analysis it may not be always applicable to

every type of cryptographic cipher or every application that is running on digital device, but essentially makes use of certain attributes of the particular program.

(Refer Slide Time: 11:38)

---

## Simple Power Analysis

**Algorithm** : SQUARE-AND-MULTIPLY( $x, c, n$ )

```
z ← 1
for i ← ℓ - 1 downto 0
do {
  z ← z2 mod n
  if ci = 1
  then z ← (z × x) mod n
}
return (z)
```

0 1 0 1 0 0 0 0 0 1 0 1 0 1 0 0 1 0 1 1 1 0 1 0 0 1 1 1 1 1

---

83

So let us look at simple power analysis, so let us say we have a device which is performing the following operation, the operation is called a square and multiply, it is a very common operation that is used for cryptographic ciphers like the RSA. So what happens over here is that this particular algorithm which we assume is implemented in a device takes three parameters  $x$ ,  $c$  and  $n$  typically the  $c$  is going to be secret and it is what the attacker wants to obtain.

So the algorithm what it does is that it first initializes  $Z$  to a value of 1 and then it has a loop  $i$  ranging from  $l$  minus 1 downto 0, where  $l$  is the length of  $C$  that is the number of bits present in  $C$ ,  $l$  minus 1 corresponds to the most significant bit of  $C$ , while  $C_0$  corresponds to the least significant bit or the LSB. So in each iteration of this loop we see that there are two operations that are performed, first is a squaring on  $Z$  where we have  $Z$  squared modular  $n$  that is performed and if  $C_i$  is equal to 1 that is if the  $i$ th bit in  $C$  in this particular iteration is set to 1 then we also have a multiplication and this goes on for every bit present in  $C$ .

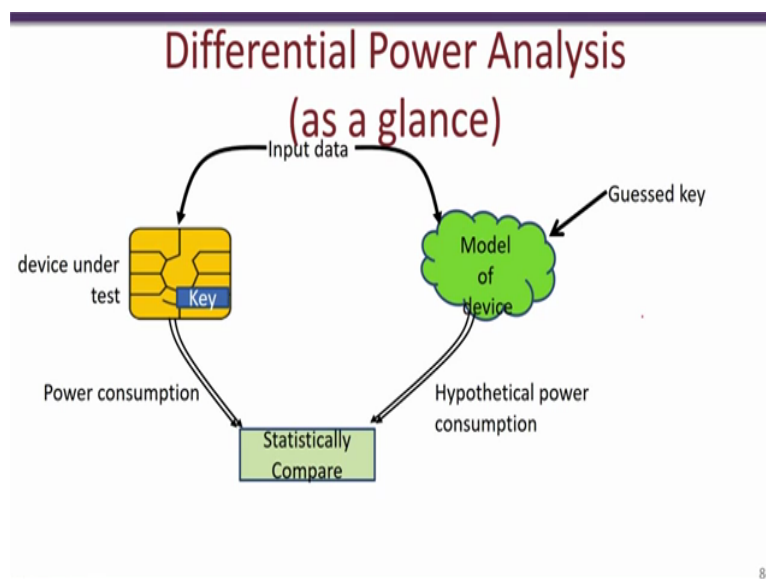
Now consider that we have our device which is actually implementing this particular algorithm is in the hands of the attacker, the attacker let us assume has knowledge of  $x$  and  $n$  although in this particular case it is not very important, but what is important is that the attacker is able to power ON the device, he is able to force this algorithm to execute and he is able to monitor the power consumed by that particular device.

So what we see is that depending on the value of  $c_i$  that is the  $i$ th bit in  $c$  the power would actually vary, if for instance the value of  $c_i$  equal to 0, then only this particular square operation is performed. On the other hand if we have a value of  $c_i$  to be equal to 1, then the square operation is followed by the multiplication operation, this difference between a value of  $c_i$  whether the value of  $c_i$  is 0 or 1 shows up in the power consumed by that device.

So this is something of what the power actually looks like and what we clearly see from here is that certain regions have a distinctively different power profile compared to other regions. So if attacker actually monitors this power consumed over time like this particular figure shown here, then attacker would simply be able to look at this power profile and be able to deduce what the secret values of  $c_i$  are. So in this way the attacker could simply be able to obtain every bit of the secret  $c$ . So for example over here he identifies that there is only a square operation that is performed and therefore the value of  $c$  should be 0.

On the other hand he sees that this region is a characteristic when 1 is obtained and therefore he would be able to deduce that this bit is 1 and so on, so in this way just by monitoring the power consumed the attacker would be able to read out the secret key through the power consumed by the device. Now this is obviously a very simple attack and over the years people have developed much more stronger ways to implement exactly this algorithm and thus simple power attacks like the one we discussed are not very applicable in modern day devices. The reason being is that most modern day cryptographic devices would not be using such algorithms where the leakage through the power is quite obvious.

(Refer Slide Time: 16:00)





## Hypothetical Power Consumption

- CMOS circuits follow the Hamming weight and Hamming distance power models

- Hamming Distance Model

- Consider transitions of register R

(1011) → (1101) → (1001) → (0010) → (0011)

#toggles

2      1      3      1

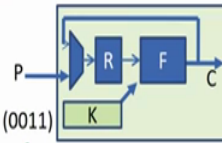
- Hamming Weight Model

(1011) → (1101) → (1001) → (0010) → (0011)

#toggles

3      2      1      3

The Hamming weight model will work, when R is precharged to either 0 or 1



So a much more powerful attack is known as a differential power analysis attack, so the main concept over here unlike the simple power analysis attack is to collect a large number of power traces and then perform some kind of statistical analysis on these power traces from which we deduce the key. So the basic idea of differential power analysis attacks or DPA as it is commonly known as is that we have a particular device over here and the assumption is the attacker has this device and this device has a key which is present inside, so this secret key is what the attacker is interested to actually retrieve.

So what the attacker does is firstly he builds a model of the device, the assumption here is that the attacker known exactly what operations are going on within this device and the only thing that the attacker does not know is the secret key, the whole attack therefore is to be able to retrieve the secret key of this device. Now the attack goes as follows, first the attacker creates a model of this particular device and then also guesses the key and then he generates various input data and feeds the same input data to the device which is under test as well as to the model of that device.

So to the device under test once he feeds the input data and forces that device to start computing on that input data, he measures the power consumed through an oscilloscope, side by side he uses the device model to obtain what is known as a hypothetical power consumption of that device. So note that this power consumption is obtained from an oscilloscope and it is the real power consumption when the actual secret key is being computed upon, so this power consumption is obtained by measuring instruments such as a digital oscilloscope, while on the other hand the power obtained from the model of the device

is obtained just by some calculations or just by some mathematical analysis. So note that this power model is with a guessed key, this is with a key that the attacker actually guesses.

So now what he does, he has these two power consumption the actual power consumption with the real key and the hypothetical power consumption with the guessed key. So he performs a statistical analysis between these two and then compares the results. So if the guessed key is indeed correct this statistical comparison would be able to identify if the guessed key is indeed correct or the key is wrong.

So we will look at more in details about how this differential power analysis attack actually works. So we will take a very small circuit to explain this particular scenario, the circuit that we will actually look at looks something like this way. So what we see over here is that we have a register  $R$  and some function  $F$ , now the output of this particular function is fed back to a multiplexer and then gets latched into this register, the input to this particular circuit is  $P$ .

To understand how differential power analysis attacks actually work, we will take a very small circuit as shown over here. So this small circuit it takes as an input  $P$  and it takes a secret  $K$  and then operates on this secret in an iterative manner. For example, in the first clock pulse the input  $P$  is taken and it gets latched into this register, in the next clock pulse this value of  $P$  is fed into  $F$ ,  $F$  you could think of as any kind of nonlinear function, so what  $F$  does is that it operates on the value of  $P$  and also the secret key  $K$ .

Now the output of  $F$  is fed back through this multiplexer and gets latched in this register. On subsequent clock cycles this register is then fed to  $F$  and there is an operation on  $F$  based on this register contents and the key and the results are fed back and stored to the register. Thus, what we see that this circuit would operate on in an iterative manner, the first iteration would be based on  $P$ , while all subsequent iterations are based on the result of the previous iteration. So the output of  $F$  is  $C$ , we assume that the attacker does not know the value of  $C$  and he is trying to obtain the value of  $K$ .

So as we discussed the first step in a differential power analysis attack is to create a model of the device. So there are two common ways by which this hypothetical power consumption model can be actually created and this is known as the hamming distance model and the hamming weight model respectively, so in the hamming weight model the model essentially looks at this register  $R$  and counts the number of 1s that are present in this register  $R$ .

So for example let us say that the initial value of present in the register R is 1011, in the next transition that is in the next clock pulse we assume that the next value of R is 1101. In this case what we do is we simply count the number of 1s that are present, in this case there are three 1s present and we say that the hypothetical power consumed by the model is 3 in this particular clock pulse.

In the next clock pulse let us if we assume that R has a value of 1001 then the hypothetical power consumed is 2 and so on, so this is the hamming weight model. So note that this is a hypothetical power consumed so the attacker is not actually finding out what the contents of the register is but he is just actually predicting what the contents of the register may be. The other model that is quite often followed in these kind of attacks is known as the hamming distance model.

So here we actually look at the number of bits that toggle from one clock pulse to another. So again we are looking at this register R and between two consecutive clock pulses we look at the number of bits that actually change. So for example over here let us say that the initial value of R is 1011 and in the next clock pulse the register changes to the value of 1101. Thus, we see that there are two bits that get toggled, essentially this bit 0 has changed to 1 and this bit which is 1 has changed to 0 and we compute the hamming distance to be 2.

Similarly if we consider these two consecutive clock pulses and let us say that the register R has changed to a value of 1001, we note that here there is only one bit that has been changed and therefore the hamming distance is 1. So in this way you see as this particular device is operating on in this iterative manner, we get a sequence of hypothetical powers that are consumed, this hypothetical power in the hamming distance model would be 2 1 3 1 and so on, in the hamming weight model this hypothetical power would be 3 2 1 3 and so on.

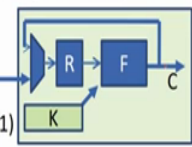
(Refer Slide Time: 23:52)

## Hypothetical Power Consumption

- CMOS circuits follow the Hamming weight and Hamming distance power models
- Hamming Distance Model**
  - Consider transitions of register R

(1011) → (1101) → (1001) → (0010) → (0011)

#toggles            2        1        3        1



- Hamming Weight Model**

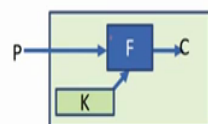
(1011) → (1101) → (1001) → (0010) → (0011)

#toggles            3        2        1        3

The Hamming weight model will work, when R is precharged to either 0 or 1

85

## A Small Example



Device

P	K	C
0000	1010	1010
0001	1010	1011
0010	1010	1000
0011	1010	1001
0100	1010	1110
0101	1010	1111
..	..	..

Mallory has control of this device.

- She can monitor its power consumption
- She can feed inputs P
- She even knows what operations goes on inside.

**The things she doesn't know is K and C**

Her aim is to obtain the secret key K

86

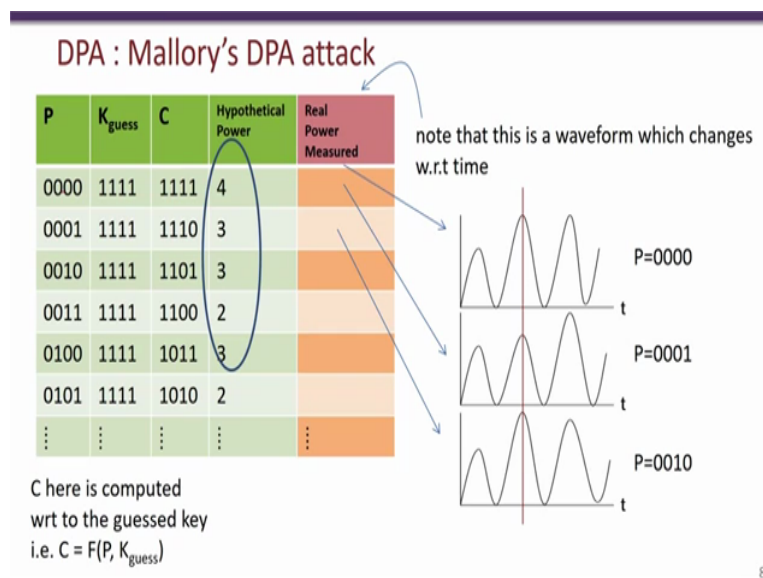
So now let us look at the differential power analysis attack. So what we are interested in is the first iteration that occurs, note that we had mentioned that in the first iteration we have P which is sent as an input which gets stored in the register and then this gets operated on by this function F. So note that we also mentioned that this function F takes as input the key and provides some intermediate output which the attacker is not able to see.

So to simplify this entire figure we just showed the input F and the secret key and the F operation and the corresponding intermediate value C. So what the attacker could do is that he could create a guess of the secret key. So in this case we are assuming that P and K are of 4 bits and also C is a 4 bits and the attacker has actually guessed that the key value is 1010, so now what we are also assuming is that the attacker is able to choose or select plain text and

sent to this device and therefore he knows the value of plain text, side by side as this F function is being operated upon the attacker is able to monitor the power consumed by the device.

So an important assumption that we make over here is that the attacker knows the functionality of F but does not know K nor C, but since has guessed the value of K he can also compute C based on that guess, so this would be essentially guessed value of C. Since he knows the plane text P and he has guessed C, he can also compute the corresponding C value.

(Refer Slide Time: 25:44)

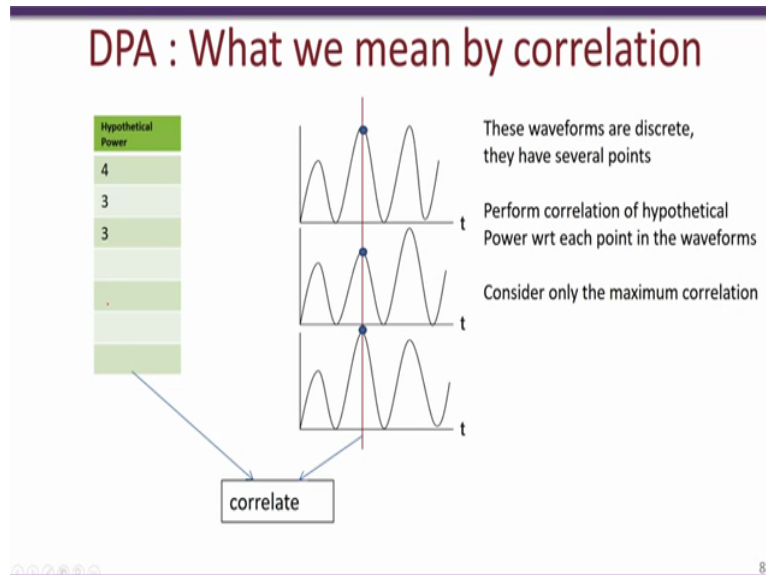


So the attack goes like this, the attacker chooses a particular value of P and sends it to the device and while the device is computing on this P and K value, the attacker has monitored the power consumed. So corresponding to the P value he gets a power traced which looks something like this. Now side by side he also makes a key guess, in this case it is 1111 and then based on the value of P and the key guess he computes what the intermediate value C would be and in the hamming weight model he then creates the hypothetical power consumed by the device.

So we take that the hypothetical power is the number of 1s present in C, so since we are following the hamming weight model. So we see here that there are four 1s so the hypothetical power in this case is 4. Now he repeats this for several more iterations, in each iteration he selects a P value feeds it to the device, gets a corresponding actual power consumed by the device and then for that guessed key which in this case is 1111 computes C

and obtains the hypothetical power consumed based on the hamming weight in this particular case.

(Refer Slide Time: 27:02)



The next step is compute a correlation between the actual power consumed that is this part and the hypothetical power, so note that corresponding to each row in this that is corresponding to each input P he would get one hypothetical power and one real power consumed. So note that this is over time, while this is just a constant value. So what the attacker does if that for each point in this real power consumption he correlates this with a hypothetical power. For example he would compute the Pearson's correlation coefficient and therefore he would get a coefficient score, does what he would obtain is a sequence of correlation values corresponding to each point in this waveform.

So this particular red line over here shows one particular correlation that has been done between these three points and this hypothetical power consumption, among the sequence of correlation values that he obtains he only considers that which has the maximum correlation, so he considers only that particular point for example say this point which has the maximum correlation. So note that this hypothetical power consumed was based on a key guess.

(Refer Slide Time: 28:16)

### DPA : Mallory's DPA attack

P	K <sub>guess</sub>	C	Hypothetical Power	Real Power Measured
0000	1111	1111	4	
0001	1111	1110	3	
0010	1111	1101	3	
0011	1111	1100	2	
0100	1111	1011	3	
0101	1111	1010	2	
⋮	⋮	⋮	⋮	⋮

note that this is a waveform which changes w.r.t time

C here is computed wrt to the guessed key  
i.e.  $C = F(P, K_{\text{guess}})$

87

In this case if you look back we had guessed the key as 1111 of course the attacker does not know what the actual key is so he iterates to every possible value of K.

(Refer Slide Time: 28:28)

### DPA : A small example

P	K <sub>guess</sub>	C	Hypothetical Power	Real Power Measured
0000	0000	0000	1101	1101
0001	0000	0000	1101	1101
0010	0001	0001	1101	1100
0011	0010	0010	1101	1111
0100	0011	0011	1101	1110
0101	0100	0100	1101	1001
⋮	0101	0101	1101	1000
⋮	⋮	⋮	⋮	⋮

correlate

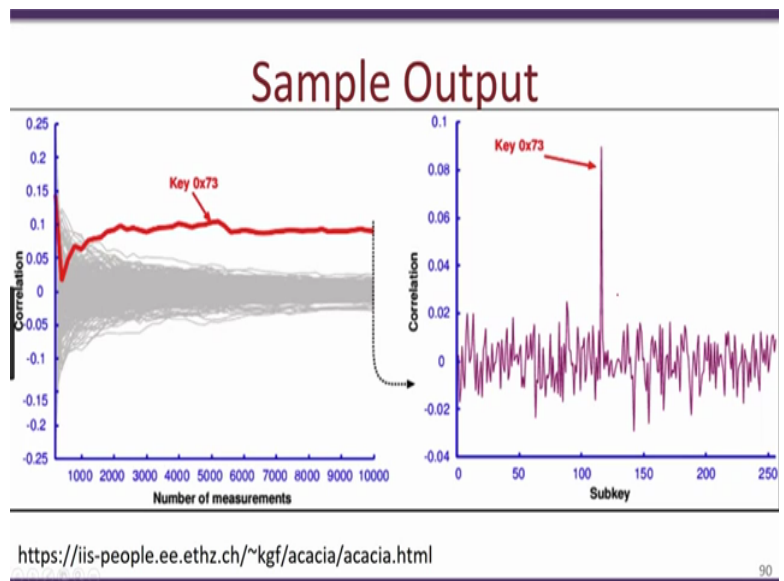
Find maximum correlation

$\rho_{15}$     $\rho_{14}$     $\rho_{13}$     $\rho_{12}$     $\rho_{11}$     $\rho_{10}$

89

So he would get something like this for every possible value of K he would be able to create a table like this corresponding to the same P value, a guessed K value, the real power consumed and the hypothetical power consumed. Since we are considering a 4 bit key, so there would be 16 such tables which would be obtained, for each of these key guesses he would compute the correlation coefficient and therefore he would be able to get 16 different correlation coefficients, he would then chose the one correlation coefficient which is the maximum.

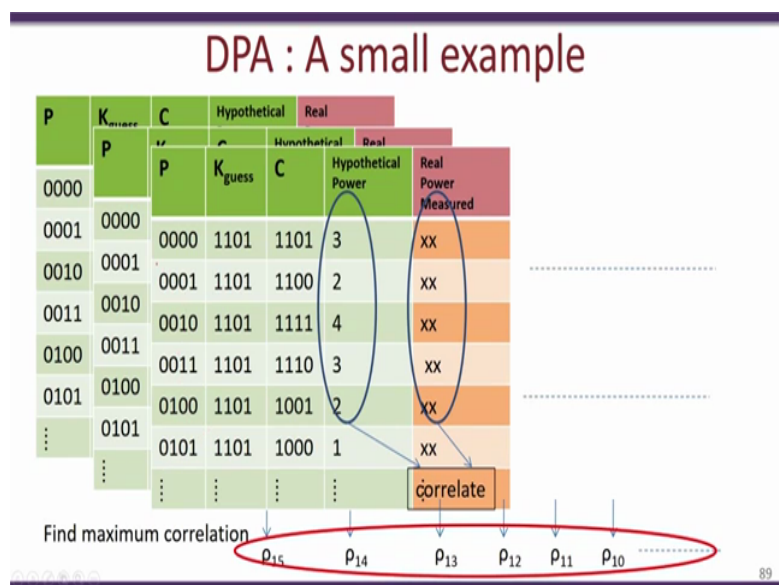
(Refer Slide Time: 28:58)



This graph for example which is obtained from this particular website shows how the maximum correlation coefficient varies with different values of key. So in this particular example they had used the AES block cipher as the device which is being attacked and each part of the AES key is just of 1 byte and therefore has 256 different possibilities, so the Y axis on the other hand has the correlation which is computed for each key guess.

So what we see is that for wrong key guesses we have a correlation which is quite small which is less than 0.02, on the other hand if the guess is correct which in this case is the key is 0x73 we get a high peak in the correlation value, this high peak would thus permit the attacker to identify the correct key.

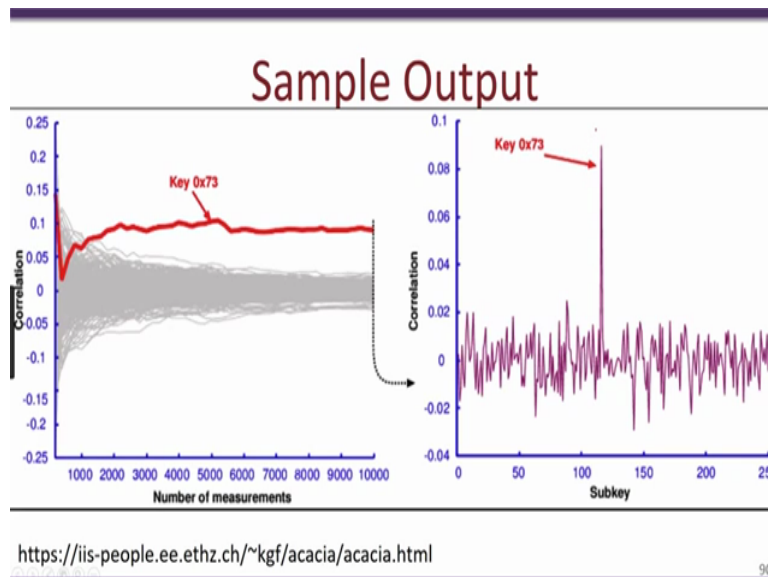
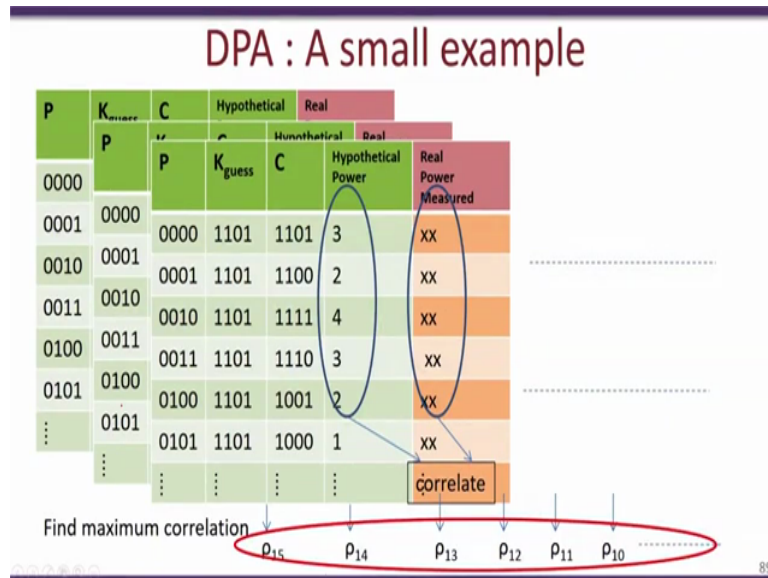
(Refer Slide Time: 30:00)





Now what actually is affected by this particular attack is the number of iterations that are done for each case. Note that we had taken this table and each line in this or each row in this particular table corresponds to one power measurement that is made in the device and one point in the hypothetical power that was computed, so the larger number of such rows present, the more accurately the key can be recovered correctly.

(Refer Slide Time: 30:25)



So this particular graph shows the number of measurements required relating to the previous table, the number of measurements implies the number of rows in the tables and what we see is that as the number of measurements keeps increasing and goes towards 10000 in this case the accuracy of identifying the secret key is increased. So for example if we have very less let us say around 100 or so measurements, we have a correlation which is less than 0.05, but as

we increase the number of measurements we have a correlation which is quite high of 0.1. So speaking in another words what we actually see is that as the number of power measurements increases this peak becomes more and more prominent.

(Refer Slide Time: 31:14)

---

## Statistical Comparison

- **Correlation :**  
Provides a value between -1 and +1. A value closer to the signifies linear dependence between the hypothetical power and the real power consumption

$$\rho_{X,Y} = \frac{E[(X - \mu_X)(Y - \mu_Y)]}{\sigma_X \sigma_Y}$$

- **Mutual Information**  
Quantifies mutual dependence between hypothetical power and real power consumption

$$I(X; Y) = \sum_{y \in Y} \sum_{x \in X} p(x, y) \log \left( \frac{p(x, y)}{p(x)p(y)} \right)$$

---

91

So what we considered in this example was that the attacker use the Pearson's correlation coefficient between a hypothetical power consumed and the actual power consumed in order to identify the correct secret key. So there are various other statistical techniques which have been evaluated, one of the most common things is known as the mutual information which essentially quantifies the mutual dependence between the hypothetical power consumed and the real power consumed.

So this is given by this particular equation, so we will not go more into details about this. So what we had looked in this previous example was that the attacker uses a Pearson's correlation coefficient or something equivalent in order to compute the correlation between a hypothetical power consumption based on a guess key and the actual power consumption measured from the device, a high value of correlation implies that the attacker has guessed the key correctly.

So there are various other statistical tolls that can be used other than a correlation, quite often this mutual information is used which essentially quantifies the dependence between the hypothetical power and the real power consumption.

(Refer Slide Time: 32:32)

## Statistical Comparison

- **Bayes Analysis**  
What is the probability of a hypothesis given a specific leakage  
$$\Pr[\text{Hypothesis} \mid \text{Leakage}]$$
- **Difference of Means**  
next...

92

Other techniques used are the Bayes analysis which computes the probability of the hypothesis given the leakage. One of the most earliest forms of statistical comparison is known as the difference of means. So we will look more in detail about how this difference of means actually works.

(Refer Slide Time: 32:53)

## Difference of Means

- Guess a key :  $k_{\text{guess}}$
- Compute  $C_{\text{guess}} = F(P, K_{\text{guess}})$
- Find the  $k_{\text{guess}}$  such that  $|AVG(B0) - AVG(B1)|$  is maximum

93

What we look at again is this particular computation, where there is a P which is taken as input and computed upon with this function F and there is also a key which is kept secret. Now what the attacker does is that he guesses the key and based on that guessed key and the known value of P he computes C, so C is of course the key guess and therefore based on the

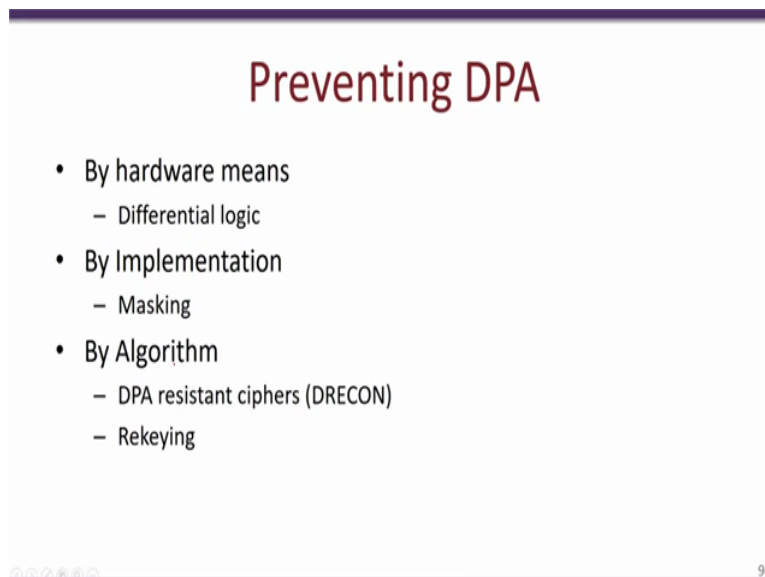
guess key he can obtain this C guess which in this case is 1111, 1110 and 1101, note that this C guess keeping K constant only depends on the value of t.

So what he would next do is that he would consider one single bit in this C guess, so for example let us say that he is considering the least significant bit of this C guess and then what he would do is that depending on the value of this least significant bit of C guess he would distribute these actual power consumed into two buckets, the first bucket corresponds to the C guess being 0, while the next bucket corresponds to the C guess equal 1.

So for example over here the least significant bit is 1 and therefore he would move this power measurement into the bucket 1. In this particular case we have the least significant bit to be 0 in C guess and therefore he moves this to the bucket 0. So in a same way in this case he has the least significant bit of C guess to be 1 and therefore this should be moved to bucket 1. So eventually after doing this over large number of different power measurements he computes the average of bucket 0 and the average of bucket 1, so this is known as the difference of means for that corresponding key guess.

So what is observed is that if the key guess happens to be correct then this particular difference the differences between the average of these two buckets would be maximized and this maximum difference of means can be than used to identify what the secret key is.

(Refer Slide Time: 35:02)



**Preventing DPA**

- By hardware means
  - Differential logic
- By Implementation
  - Masking
- By Algorithm
  - DPA resistant ciphers (DRECON)
  - Rekeying

94

So differential power attacks have been known for almost two decades now and there have been several counter measures that have been developed over these years. So these counter measures have been applied at three different levels they can be applied at the hardware level,

at the implementation level, or at the algorithm level. At the hardware level what people have suggested is that the standard CMOS logic be replaced with other logic which are resistant to these power analysis attacks. So these techniques for example the WDDL gates would be built in such a way so that the power consumed is independent of the computations that are performed by the corresponding gates.

At the implementation level techniques such as masking and threshold implementations have been used where randomization has been incorporated into the design so that where the randomness limits the amount of information that the attacker can gain from the power consumed by the device.

A third method is at the algorithmic level, where researchers have actually been able to build algorithms in particular cipher algorithms as well as protocols which can prevent power analysis attacks, these algorithms inherently are built so that the leakage would not give enough of information to an attacker to glean secret information like the secret key. Two of the popular algorithms in this case is known as DRECON which stands for DPA resistance by construction and the rekeying techniques, thank you.