CSA E0 235: Cryptography September 18, 2019

Scribe For Lecture 12

Instructor: Arpita Patra *Submitted by: Virti Savla*

1 Recap

In the previous lecture, we learned about Security definitions of MAC- cma, strong cma.We saw its construction from PRF. We also learned Domain Extension: How to find a tag for long messages. We saw definition of One-time Information-theoretic MAC and Construction pairwise independent functions (PUFs).

2 Authenticated Encryption : Hybrid of SKE and MAC

Secret Key Encryption provides privacy on the other hand MAC provides integrity and authentication. Thus, they provide complementary goals. SKE that provides privacy of message does not necessarily provide integrity of message.For example, CPA security provides standard amount of security but does not provide authentication and message integrity as it is easy to come up with modified valid ciphertext not seen before and its also easy to manipulate the known ciphertext. MAC on the other hand does not provide privacy as its easy to distinguish tag of two different messages. Thus, Authenticated Encryption is hybrid of both SKE and MAC. It provides privacy and message integrity and authentication.

3 Authenticated Encryption

It is possible to obtain secrecy in the SKE setting using encryption and its also possible to ensure integrity using Message Authentication Codes(MAC). It is human nature to always ask for more and better, thus one might naturally want to achieve both goals simultaneously. In a nut shell Authenticated Encryption can be given as follows :

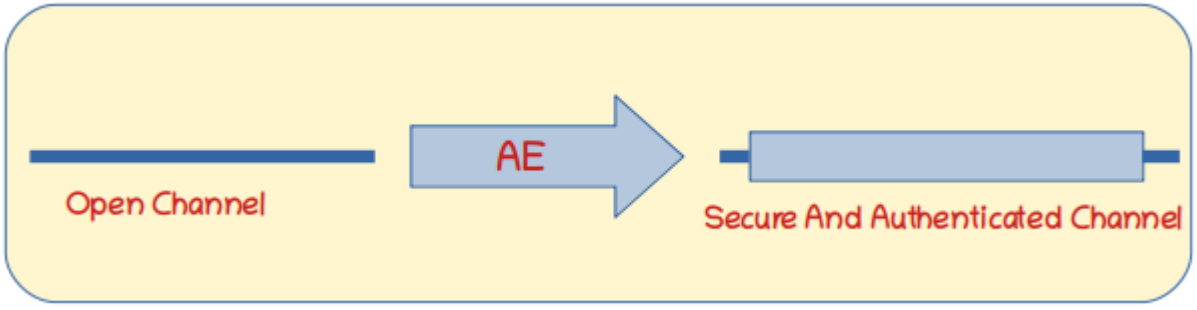

Figure 1

Our goal is to realize an "ideally secure" communication channel that provides both message secrecy and integrity. Authenticated Encryption provides both privacy and message integrity.

Definition 1 Authenticated Encryption can be defined as follows:
π= (Gen, Enc, Dec) is an authenticated encryption if
π = (Gen, Enc, Dec) is cpa-secure AND
π = (Gen, Enc, Dec) has ciphertext integrity (hard to come up with a ciphertext that has valid decryption even after sufficient training) ♢

4 Ciphertext Integrity Experiment

Consider the game in *Figure 2* for Ciphertext integrity experiment.The adversary can query a polynomial number of messages and obtain the corresponding ciphertexts Q = { c_1 ,..., c_t } from the Encryption Oracle encrypted with key k,where k has been generated by verifier using Gen().This is known as the training phase.The adversary sends a message m to verifier, who in turn generates the corresponding ciphertext with same key k.This is known as challenge phase. Finally the adversay has to come up with new ciphertext(not same as any of the ciphertext received from the challenger during training phase) that decrypts to a valid message. If the ciphertext that adversary has output decrypts to a valid message the game output is 1 i.e. adversary WINS else the game output is 0 i.e. adversary LOSES.

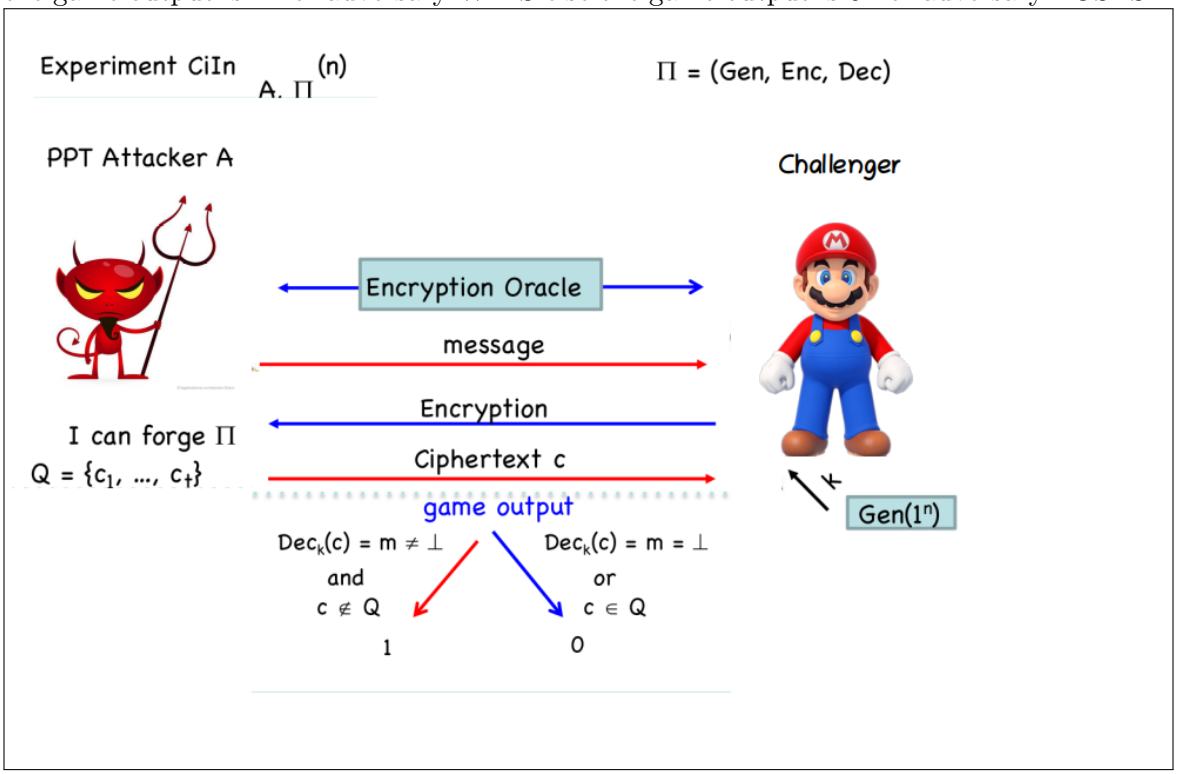

Figure 2

Thus, π has ciphertext integrity if for every PPT adversary the probability of an advesary coming up with a valid cipher text is less than a negligible function of n i.e.

$$\Pr\left[CiIn_{A,\pi}(\mathrm{n}) \right] \leq \mathrm{neg}$$

5 Construction Of Authenticated Encryption

We have to combine cpa-secure SKE and scma-secure MAC to construct a scheme which follows Authenticated Encryption.CPA-Secure SKE and SCMA-Secure MAC can be combined in various ways. We try various combinations to see which combination is AE secure

5.1 Attempt I (Encrypt And Authenticate)

Let π_E = (Enc, Dec) be a cpa-secure SKE and π_M = (Mac, Vrfy) be a scma-secure MAC.The Gen() algorithm in both π_E and π_M selects a random key from the respective domains. Consider the encryption and decryption scheme in *Figure 2*

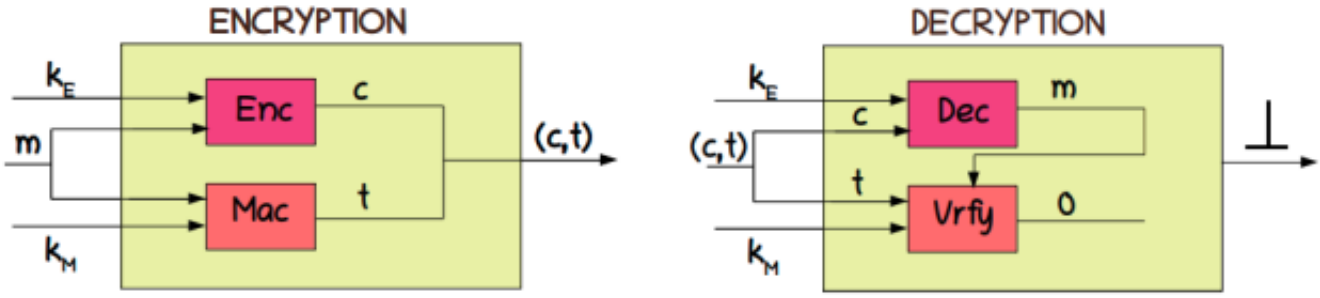

Figure 3

The keys k_E and k_M are independent keys for π_E and π_M.This approach is used in SSH. But this scheme does not follow the norms of Authenticated Encryption as it does not provide message privacy. Since the encrypted message and tag together form the ciphertext, tag in ciphertext can leak some information about the message.Eg. a MAC may always output the first two bits of m as the first two bits of MAC tag.

5.2 Attempt II (Authenticate Then Encrypt)

Let π_E = (Enc, Dec) be a cpa-secure SKE and π_M = (Mac, Vrfy) be a scma-secure MAC.The Gen() algorithm in both π_E and π_M selects a random key from the respective domains. Consider the encryption and decryption scheme in *Figure 4*

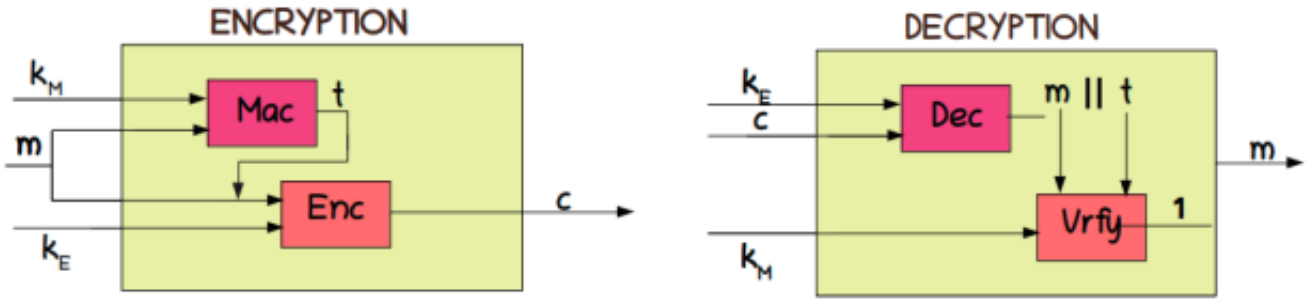

Figure 4

The keys k_E and k_M are independent keys for π_E and π_M.This approach is used in SSL.But this scheme does not follow the norms of Authenticated Encryption as it does not provide an authenticated cipher. There exists an instantiation of π_E and which is cpa-secure and which when combined with any MAC using the above approach does not lead to an authenticated cipher.CBC-mode of encryption along with MAC using above approach does not give an Authenticated Encryption. In general this approach is not recommended.

5.3 Attempt III (Encrypt Then Authenticate)

Let π_E = (Enc, Dec) be a cpa-secure SKE and π_M = (Mac, Vrfy) be a scma-secure MAC.The Gen() algorithm in both π_E and π_M selects a random key from the respective domains. Consider the encryption and decryption scheme in *Figure 5*

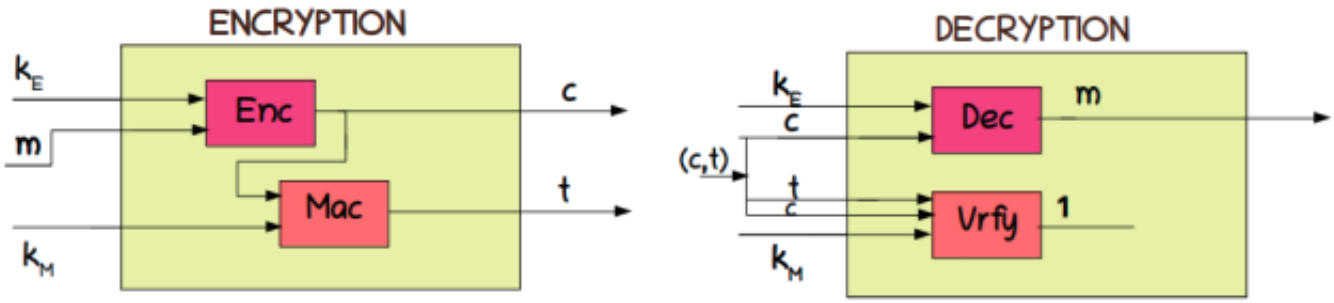

Figure 5

This approach used in IPSec.Fortunately this approach always lead to an Authenticated Encryption, irrespective of how π_E and π_M are instantiated.

5.3.1 Proof - Encrypt Then Authenticate construction leads to Authenticated Encryption

To prove that the scheme proposed is AE we have to prove two statements
(1)CPA-security

(2)Ciphertext Integrity
Consider the following construction

$$\pi' = (\text{Gen}', \text{Enc}', \text{Dec}')\text{: authenticated encryption}$$

• **Key Generation Algorithm (Gen'())** - This algorithm generates two secret keys,such that $k_E \in_R \{0,1\}^n$ and $k_M \in_R \{0,1\}^n$
• **Encryption Algorithm (Enc'())** - This algorithm takes the inputs message m and the two keys k_E and k_M. We get ciphertext and tag as output computed in following manner

$$\text{c} \leftarrow Enc_{kE}(\text{m}) \text{ and } \text{t} \leftarrow MAC_{kM}(\text{c})$$

• **Decryption Algorithm (Dec'())** - This algorithm takes as input the tuple of the ciphertext and the tag and first verifies whether the ciphertext is indeed the same ciphertext which was sent, that is, it corresponds to the tag which has been sent along, and only once it is verified will the ciphertext be decrypted and the original message is retrieved.

π_E = (Enc,Dec) be a cpa-secure SKE and π_M = (Mac, Vrfy) be a scma-secure MAC

Lemma 1 *If π_E is cpa-secure then π is cpa-secure.*

Proof : The contrapositive is : If π is not cpa-secure then π_E is not cpa-secure. To prove this we will use a successfull adversary A_π for scheme π.Hence, in the game based strategy we can say that there exists a polytime adversary A_π ,who can win the game with a non-negligible probability and break the scheme π. The reduction based proof is given below in *Figure 6*

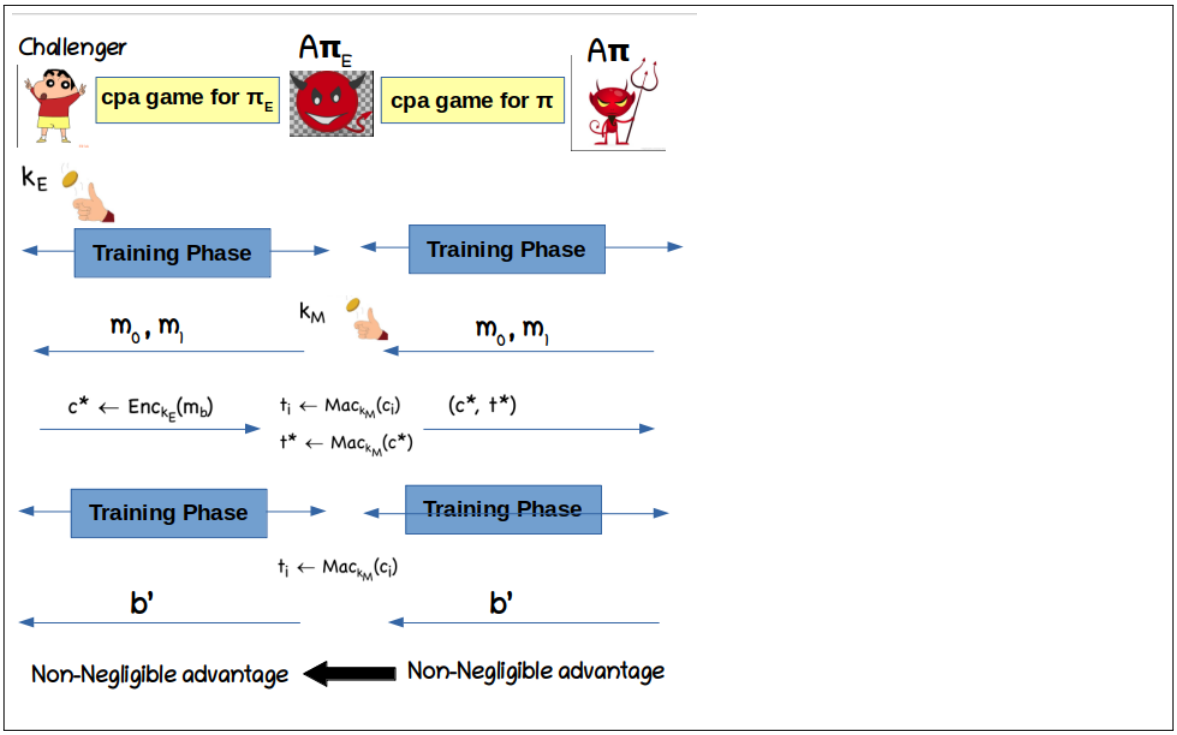

Figure 6

Here, $A_{\pi E}$ uses the power of A_π to break the CPA security of scheme. $A_{\pi E}$ simulates the scheme and plays the CPA game with the verifier. From the reduction based game above it is clear that if A_π has a non-negligible advantage then so does $A_{\pi E}$, i.e. scheme is not CPA-secure. Hence, proved that If π_E is cpa-secure then π is cpa-secure. ■

Lemma 2 *If π_E is scma-secure then π has ciphertext integrity.*

Proof : The contrapositive is : If π is not ciphertext integrity-secure then π_E is not scma-secure. To prove this we will use a successfull adversary A_π for scheme π.Hence, in the game based strategy we can say that there exists a polytime adversary A_π ,who can win the game with a non-negligible probability and break the scheme π. The reduction based proof is given below in *Figure 7*

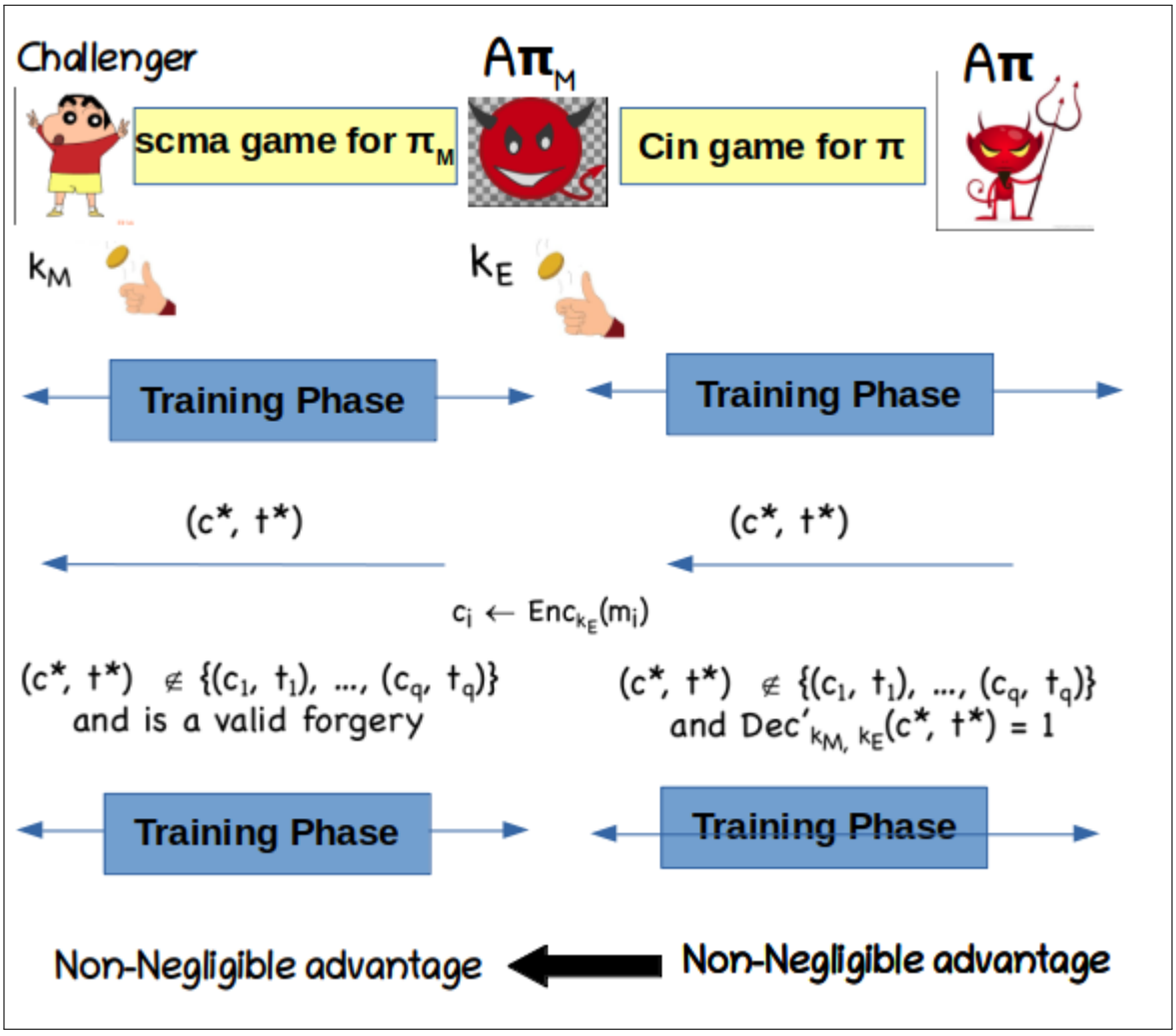

Figure 7

Step 1 - Training Phase
Similar to our Ciphertext-Integrity game the adversary A_π will send some messages to $A_{\pi\mathrm{M}}$. $A_{\pi\mathrm{M}}$ will generate a key k_E and encrypt the message with the key as follows $Enc_{k\mathrm{E}}(m_i)$ and forward it to verifier(challenger). Verifier in return generates a valid tag t_i on ciphertext c_i and sends the (c_i , t_i) pair to $A_{\pi\mathrm{M}}$ who simply forwards it to A_π.
Step 2 - Challenge Phase
In challange phase A_π sends a new valid (c^* , t^*) pair not seen before (as we assumed A_π can break ciphertext integrity) to $A_{\pi\mathrm{M}}$ who simply forwards it to the verifier.

$A_{\pi\mathrm{M}}$ can directly forward this to verifier because (1) If c^* send by A_π is new then its valid in scma game as it has not been seen before. (2) If c^* send by A_π is same as one of training ciphertext then during the training phase reduction would have already queried on c^*, since c^* is already repeated the t^* component send by A_π must be new.

From the reduction based game above it is clear that if A_π has a non-negligible advantage then so does $A_{\pi\mathrm{M}}$, i.e. scheme is not scma-secure.

5.3.2 Need for Independent Keys k_M and k_E

The keys k_M and k_E must be chosen independently. Consider the following schemes.
π_E: To encrypt m $\in \{0, 1\}^{n/2}$, select a random r $\in \{0, 1\}^{n/2}$
π_M: To authenticate c $\hat{\mathrm{I}}\in \{0, 1\}^{n}$, output tag t = F_k^{-1} (c)
Now assume, $k_E = k_M$ = k
$Enc_k^{'}$(m) = $Mac_k(Enc_k(\mathrm{m}))$ =$F_k^{-1}(F_k$ (m | | r)) = m | | r
Thus, same encryption and MAC keys leads to insecure Encrypt-then-authenticate approach. This means that Encrypt-then-authenticate approach is not insecure provided the encryption and MAC keys are independent

5.4 Looking Back and Forward

The *Figure 8* gives the outline of the things we have seen so far

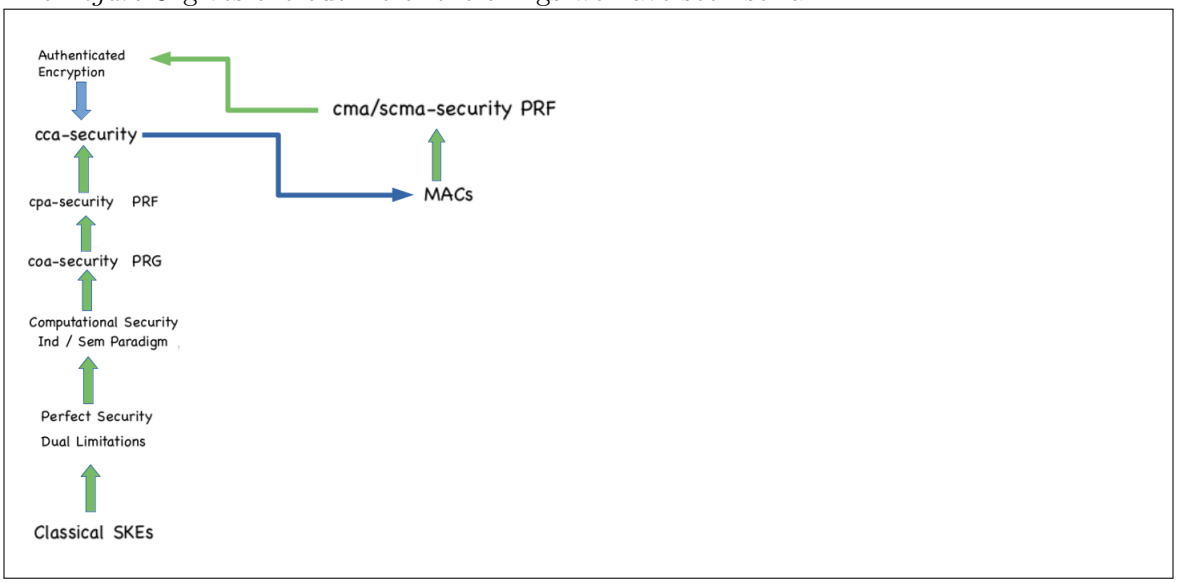

Figure 8

5.4.1 Minicrypt

Minicrypt consist of all the primitives that we can build believing that OWF exists. All the cryptographic primitive we know rely on OWF.

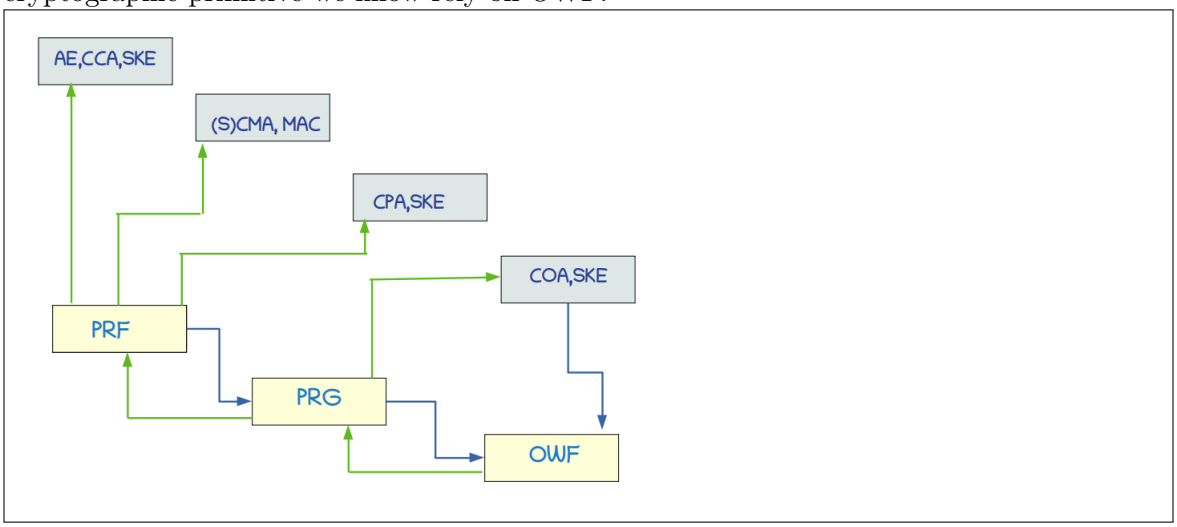

Figure 9

5.4.2 Roadmap

Figure 10 gives the roadmap from OWF to PRF

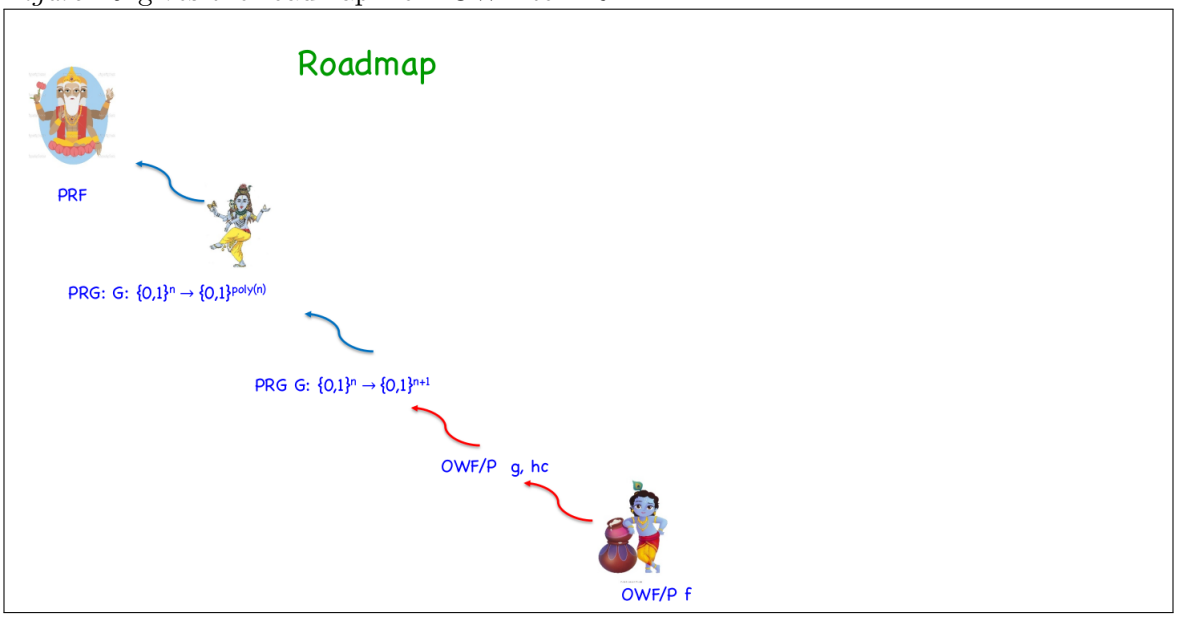

Figure 10

5.5 One Way Function

Functions that are easy to compute but **difficult** to invert (almost-always) are One Way Functions.AE,CPA,COA,SCMA,SKE are all part of minicrypt.

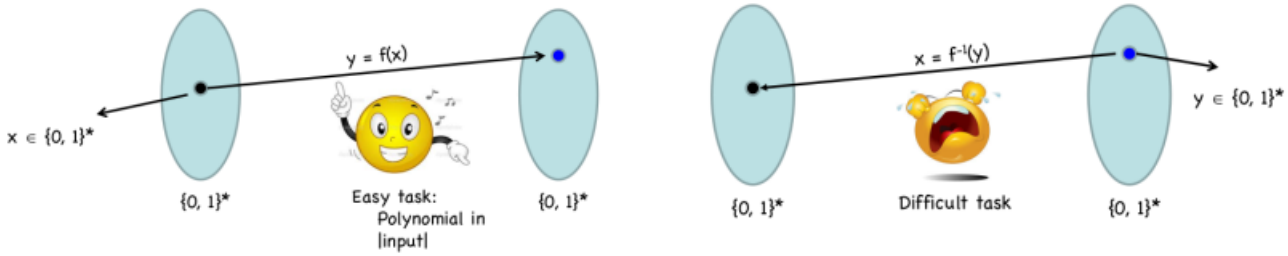

Figure 11

5.5.1 The Inverting Experiment

Consider the game in *Figure 12* for Inverting Experiment. The verifier computes y=f(x) and sends it to the adversary. The adversary outputs $x' = f^{-1}(y)$ (pre-image of y). If $f(x')=y$ then game output is 1 and adversary wins the game else if $f(x')\neq y$ then game output is 0 and adversary looses.A need not have to find the original x to win the game its sufficient to find one pre-image

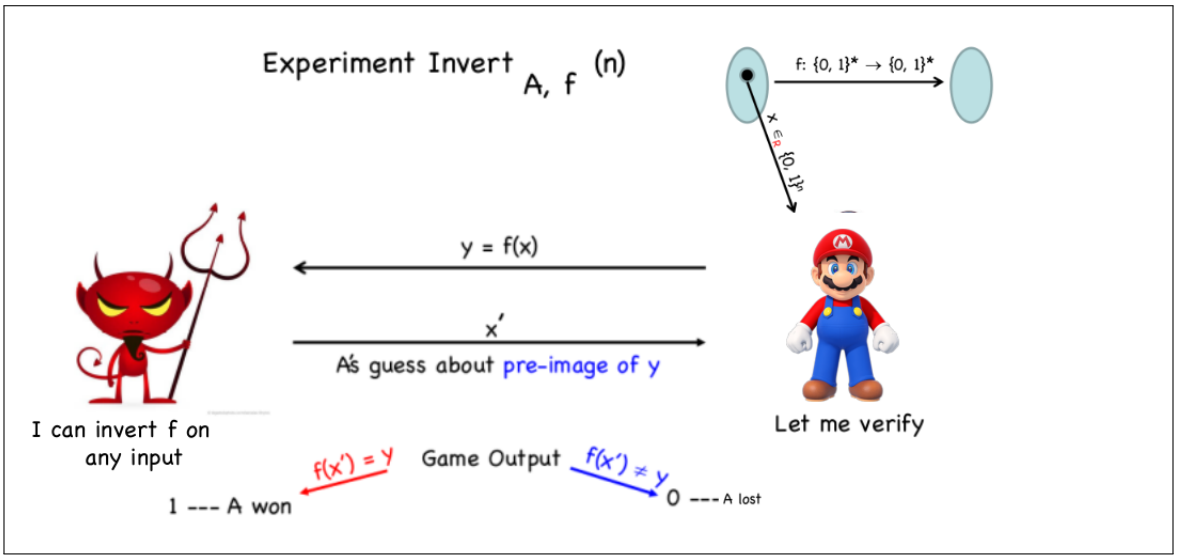

Figure 12

Thus, function f is a One Time Function if the following two conditions hold :
(1) Easy to compute: for every x ∈ {0, 1}*, f(x) can be computed in poly(|x|) times.
(2) Hard to Invert: For every PPT algorithm A, there is a negligible function negl() :

$$\Pr\left[Invert_{A,f}(n)=1 \right] \leq \text{neg} \approx \Pr[\ A(f(x),1^n) \in f^{-1}(f(x))] \leq \text{negl}(n) \quad x \leftarrow \{0,1\}^n$$

If the adversary has unbounded computational power, in this case OWF does not exist.
• Any function is invertible in principle given enough time/computational power.
•The assumption of existence of OWF is about computational hardness.

5.5.2 Functions that are not one-way (non-OWFs)

For a function to be non-OWF, there should exist an Adversary A, p(n) such that:

$\Pr[\ A(f(x),1^n) \in f^{-1}(f(x))] \geq 1/p(n)$ for infinitely many n's
$x \leftarrow \{0,1\}^n$

Following are the example of the functions that are **NOT One Way**

Example 1 :
$\Pr[\ A(f(x),1^n) \in f^{-1}(f(x))] > 1/n^{10}$ when n is even
. $\leq \text{neg}(n)$ when n is odd
As it is easy to compute inverse when n is even.

Example 2 :
f(x, y) = x.y, where x, y ∈ N
$\Pr[\ A(f(x,y),1^n) \in f^{-1}(f(x,y))] \geq 3/4$
$x \leftarrow \{0,1\}^n$ xy: even → ((2, xy/2) is a pre-image)

Example 3 :
$f(x) = x_1\ x_2\ .....x_{n-1}$, where $x \in \{0,1\}^n$

$Pr[\ A(f(x),1^n) \in f^{-1}(f(x))] = 1/2$
$x \leftarrow \{0, 1\}^n$