Indian Institute of Technology Delhi

COL 759 - Cryptography

Assignment 3

Security against Active Attacks and Introduction to Public Key Cryptography

Garv Nagori, Naman Agarwal
2021CS10549, 2021CS10555

Contents

§1 Question 1

CPA secure encryption scheme with very weak Ciphertext Integrity, must have a way of checking for $\perp$ or actual message. So there must be some way to include a way to sign messages.
We can use a CPA secure encryption and use a UFCMA-Secure MAC but the problem with that is key space is $\{0,1\}^n$ and the PRF's key space is also $\{0,1\}^n$, so we may need to use the same key twice (or expand the key but that requires a PRG).
So, we make use of the fact, that the scheme only requires very weak ciphertext integrity and we do not actually require a UFCMA secure MAC.
The Encryption Scheme ε is defined as:

$$\begin{aligned} Enc(k,m) &:= r, F(k,r) \oplus m, F(k, F(k,r) \oplus m) \\ Dec(k, ct = (ct_1, ct_2, ct_3)) &:= m \text{ if } F(k, ct_2) = ct_3 \text{ else } \perp \\ m &= F(k, ct_1) \oplus ct_2 \end{aligned}$$

This scheme is CPA secure and also has very weak ciphertext integrity.

CPA Security

We need to show that if an adversary can break CPA security of this scheme, then we can use it to create a reduction that breaks the PRF security of F.
We create a Hybrid World 0 where the pseudorandom function F has been replaced by a truly random function f. If the probability difference of guessing the bit correctly in the CPA security game in both worlds is non-negligible then we can break security of F.
The reduction works as follows:

CPA Security

Challenger $\mathcal{C}$		**Reduction $\mathcal{B}$**		**Adversary $\mathcal{A}$**
$b_c \leftarrow\$ \{0,1\}$		$b_a \leftarrow\$ \{0,1\}$		
$k \leftarrow\$ \{0,1\}^n$				
if $b_c = 0$ then $y = F(k,x)$	$\xleftarrow{x}$	$r \leftarrow\$ \{0,1\}^n$	$\xleftarrow{(m_{i0}\ ,\ m_{i1})}$	
else $y \leftarrow\$ \{0,1\}^n$	$\xrightarrow{y}$	Query for r and $y(r) \oplus m_{ib_a}$	$\xrightarrow{ct}$	
	$\xleftarrow{b''}$	$b'' = 0$ if $b' = b_a$ else $b'' = 1$	$\xleftarrow{b'}$	

The advantage in this game is given by the difference in winning probability between World 0 and Hybrid World 0, hence it is non-negligible.

Now we have to show that the winning probability in Hybrid World 0 is negligible. We do this by creating another Hybrid World 1 where the randomness is chosen without replacement. The difference in probability is negligible because randomness is chosen from $\{0,1\}^n$, which is exponential in the security parameter.
In Hybrid World 1, each time the f chooses output to a new value until $f(r) \oplus m$ becomes equal for two, which has a negligible probability of happening. Hence, the probability of winning in this game is negligible.
Hence, the probability of winning in the CPA security game of this encryption algorithm is negligible and hence, it is CPA secure.

Very Weak Ciphertext Integrity

An Adversary $\mathcal{A}$ has to compute values of both $F(k,r)$ and $F(k,F(k,r)\oplus m)$. The adversary chooses r and m but has no way of knowing the key. The adversary can choose value of any 1 out of r and m to make at least one of the values match. // We can create a Hybrid World 0, where F is replaced by a truly random function f. We can create a reduction that breaks the security of F.

Very Weak CT Integrity

Challenger $\mathcal{C}$		**Reduction** $\mathcal{B}$		**Adversary** $\mathcal{A}$
$b \leftarrow\$ \{0,1\}$				
$k \leftarrow\$ \{0,1\}^n$				
if $b=0$ then $y=F(k,x)$	$\xleftarrow{x}$	Query for r	$\xleftarrow{(r\ ,\ ct_1\ ,\ ct_2)}$	
else $y \leftarrow\$ \{0,1\}^n$	$\xrightarrow{y}$	Query for ct_1		
	$\xleftarrow{b'}$	if $y(ct_1)=ct_2$ send 0 else 1		

The advantage of winning in this game is the difference in probability of winning between World 0 and Hybird World 0, which is non-negligible. Hence, the difference is negligible.

The probability of winning in Hybrid World 0 is $\leq \frac{1}{2^n}$. This is because the adversary has to compute values of $f(r)$ and $f(f(r)\oplus m)$ such that they both match. Since value of f for any input is random, only one of them can match perfectly and the other has a probability of $\frac{1}{2^n}$ of being correct. Basically, the adversary has to guess the output to f for at least 1 input, and has a $\frac{1}{2^n}$ chance of guessing it correctly.
Hence, this encryption scheme has Very Weak Ciphertext Integrity.

§2 Question 2

We define Encryption Scheme for two parties as follows:

$$\text{Enc-two}(k_i, k_j, m) = Enc(k_i, Enc(k_j, m)) \text{ WLG } i < j$$
$$\text{Dec-two}(k_i, k_j, ct) = Dec(k_j, Dec(k_i, ct)) \text{ WLG } i < j$$

To prove security for the game described, we can model it as worlds 0 and 1.
In World 0, we send Enc-two$(k_i, k_j, m^0_{i,j})$ and in World 1, we send Enc-two$(k_i, k_j, m^1_{i,j})$.
We create two Hybrid Worlds. In each, we change which message to send for one pair i, j.
In Hybrid World 0, we send encryptions of $m^0_{1,2}, m^1_{2,3}$ and $m^0_{3,4}$.
In Hybrid World 1, we send encryptions of $m^1_{1,2}, m^1_{2,3}$ and $m^0_{3,4}$.

Claim 2.1 — If an adversary $\mathcal{A}$ can distinguish between World 0 and Hybrid World 0, there exists a reduction $\mathcal{B}$ that can break the CPA security of Enc.

Proof. The reduction $\mathcal{B}$ works as follows:

World 0 and Hybrid World 0

Challenger $\mathcal{C}$		**Reduction $\mathcal{B}$**		**Adversary $\mathcal{A}$**
$b \leftarrow\$ \{0,1\}$				
$k_3 \leftarrow\$ \{0,1\}^n$		$k_2 \leftarrow\$ \{0,1\}^n$	$\xleftarrow{(k_1\ ,\ k_4)}$	
	$\xleftarrow{(m_0\ ,\ m_1)}$	Send 2 queries (m^0_{23}, m^1_{23})	$\xleftarrow[(m^1_{12}\ ,\ m^1_{23}\ ,\ m^1_{34})]{(m^0_{12}\ ,\ m^0_{23}\ ,\ m^0_{34})}$	Challenge Phase
	$\xrightarrow{Enc(k_3\ ,\ m_b)}$	and $(Enc(k_4, m^0_{34}), Enc(k_4, m^0_{34}))$	$\xrightarrow{(ct_1\ ,\ ct_2\ ,\ ct_3)}$	
		if $(i,j) = (2,3)$ Query for m	$\xleftarrow{\text{Enc: } (i\ ,\ j)\ ,\ m}$	Encryption Queries
		if $(i,j) = (3,4)$ Query for $Enc(k_4, m)$	$\xrightarrow{ct}$	
	$\xleftarrow{b'}$		$\xleftarrow{b'}$	

The reduction can compute Encryption of all using k_1, k_2 and k_4. Wherever, "query" is written, reduction can just send the same message twice so that the challenger just computes the encryption of that message.
To compute encryption of m^0_{12}, the reduction can do it directly since it has both the keys. To compute encryption of m^0_{34}, the reduction computes $Enc(k_4, m^0_{34}$ since it has k_4 and then it sends both messages as this to challenger who then encrypts it using k_3 and return it.
If challenger $\mathcal{C}$ chooses bit 0 then it encrypts m^0_{23} using k_3 and then we encrypt it using k_2. Else if challenger chooses bit 1, we do the same thing but for m^1_{23}. Hence, if bit is 0, adversary gets World 0 and if bit is 1 adversary gets Hybrid World 0.
In the encryption phase, if adversary queries for $(i, j) = (1, 2)$, the reduction can just encrypt directly since it has both keys k_1, k_2. If adversary queries for $(i, j) = (2, 3)$ we query for m. The challenger encrypts it using k_3 and then we encrypt that using k_2. If the adversary queries for $(i, j) = (3, 4)$ we encrypt m using k_4 and then query for it so that the challenger can encrypt using k_3.
Thus, if the adversary can distinguish between these 2 worlds with non-negligible probability then the reduction also wins the CPA security game with non-negligible probability. This proves that these

worlds are indistinguishable.

□

Claim 2.2 — If an adversary $\mathcal{A}$ can distinguish between Hybrid World 0 and Hybrid World 1, there exists a reduction $\mathcal{B}$ that can break the CPA security of Enc.

Hybrid World 0 and Hybrid World 1

Challenger $\mathcal{C}$		**Reduction $\mathcal{B}$**		**Adversary $\mathcal{A}$**
$b \leftarrow\$ \{0,1\}$				
$k_2 \leftarrow\$ \{0,1\}^n$		$k_3 \leftarrow\$ \{0,1\}^n$	$\xleftarrow{(k_1\ ,\ k_4)}$	
	$\xleftarrow{(m_0\ ,\ m_1)}$	Send 2 queries (m_{12}^0, m_{12}^1)	$\xleftarrow[(m_{12}^1\ ,\ m_{23}^1\ ,\ m_{34}^1)]{(m_{12}^0\ ,\ m_{23}^0\ ,\ m_{34}^0)}$	Challenge Phase
	$\xrightarrow{Enc(k_3\ ,\ m_b)}$	and $(Enc(k_3, m_{23}^1), Enc(k_3, m_{23}^1))$	$\xrightarrow{(ct_1\ ,\ ct_2\ ,\ ct_3)}$	
		if $(i,j) = (1,2)$ Query for m	$\xleftarrow{\text{Enc: } (i\ ,\ j)\ ,\ m}$	Encryption Queries
		if $(i,j) = (2,3)$ Query for $Enc(k_3, m)$	$\xrightarrow{ct}$	
	$\xleftarrow{b'}$		$\xleftarrow{b'}$	

The reduction can compute Encryption of all using k_1, k_3 and k_4. Wherever, "query" is written, reduction can just send the same message twice so that the challenger just computes the encryption of that message.
To compute encryption of m_{34}^0, the reduction can do it directly since it has both the keys. To compute encryption of m_{23}^1, the reduction computes $Enc(k_3, m_{23}^1$ since it has k_3 and then it sends both messages as this to challenger who then encrypts it using k_2 and return it.
If challenger $\mathcal{C}$ chooses bit 0 then it encrypts m_{12}^0 using k_2 and then we encrypt it using k_1. Else if challenger chooses bit 1, we do the same thing but for m_{12}^1. Hence, if bit is 0, adversary gets Hybrid World 0 and if bit is 1 adversary gets Hybrid World 1.
In the encryption phase, if adversary queries for $(i,j) = (3,4)$, the reduction can just encrypt directly since it has both keys k_3, k_4. If adversary queries for $(i,j) = (1,2)$ we query for m. The challenger encrypts it using k_2 and then we encrypt that using k_1. If the adversary queries for $(i,j) = (2,3)$ we encrypt m using k_3 and then query for it so that the challenger can encrypt using k_2.
Thus, if the adversary can distinguish between these 2 worlds with non-negligible probability then the reduction also wins the CPA security game with non-negligible probability. This proves that these worlds are indistinguishable.

□

Claim 2.3 — If an adversary $\mathcal{A}$ can distinguish between Hybrid World 1 and World 1, there exists a reduction $\mathcal{B}$ that can break the CPA security of Enc.

Hybrid World 1 and World 1

Challenger $\mathcal{C}$		**Reduction** $\mathcal{B}$		**Adversary** $\mathcal{A}$
$b \leftarrow\$ \{0,1\}$				
$k_3 \leftarrow\$ \{0,1\}^n$		$k_2 \leftarrow\$ \{0,1\}^n$	$\xleftarrow{(k_1\ ,\ k_4)}$	
	$\xleftarrow{(m_0\ ,\ m_1)}$	Send 2 queries $(m_{23}^1, m_{23}^1))$	$\xleftarrow[(m_{12}^1\ ,\ m_{23}^1\ ,\ m_{34}^1)]{(m_{12}^0\ ,\ m_{23}^0\ ,\ m_{34}^0)}$	Challenge Phase
	$\xrightarrow{Enc(k_3\ ,\ m_b)}$	and $(Enc(k_4, m_{34}^0), Enc(k_4, m_{34}^1)$	$\xrightarrow{(ct_1\ ,\ ct_2\ ,\ ct_3)}$	
		if $(i,j) = (2,3)$ Query for m	$\xleftarrow{\text{Enc: } (i\ ,\ j)\ ,\ m}$	Encryption Queries
		if $(i,j) = (3,4)$ Query for $Enc(k_4, m)$	$\xrightarrow{ct}$	
	$\xleftarrow{b'}$		$\xleftarrow{b'}$	

The reduction can compute Encryption of all using k_1, k_2 and k_4. Wherever, "query" is written, reduction can just send the same message twice so that the challenger just computes the encryption of that message.
To compute encryption of m_{12}^1, the reduction can do it directly since it has both the keys. To compute encryption of m_{23}^1, the reduction sends both messages as m_{23}^1 to challenger who encrypts it using k_3 and returns it. Then we encrypt it using k_2 and send it to adversary.
Then we encrypt m_{34}^0 and m_{34}^1 using k_4 and send these as a query to challenger.
If challenger $\mathcal{C}$ chooses bit 0 then it encrypts $Enc(k_4, m_{34}^0$ using k_3 . Else if challenger chooses bit 1, it encrypts $Enc(k_4, m_{34}^1$ using k_3 . Hence, if bit is 0, adversary gets Hybrid World 1 and if bit is 1 adversary gets World 1.
In the encryption phase, if adversary queries for $(i,j) = (1,2)$, the reduction can just encrypt directly since it has both keys k_1, k_2. If adversary queries for $(i,j) = (2,3)$ we query for m. The challenger encrypts it using k_3 and then we encrypt that using k_2. If the adversary queries for $(i,j) = (3,4)$ we encrypt m using k_4 and then query for it so that the challenger can encrypt using k_3.
Thus, if the adversary can distinguish between these 2 worlds with non-negligible probability then the reduction also wins the CPA security game with non-negligible probability. This proves that these worlds are indistinguishable.

□

Thus, we have shown that all worlds are indistinguishable to the next world. Since the number of Hybrid Worlds is finite, World 0 and World 1 is indistinguishable. That is to say, an adversary cannot distinguish between World 0 and World 1 with non-negligible probability. This proves the security of our encryption scheme.

§3 Question 3

§3.1 Part a)

We have to make a One-query secure MAC scheme for unbounded adversaries.

$$\begin{aligned}\text{Key Space } \mathcal{K} &= \{0,1\}^{2n^2} \\ \text{Sign}((k_1,k_2,k_3,\ldots,k_{2n}),m) &= (k_1 \& m) \oplus k_2 \,||\, (k_3 \& m) \oplus k_4 \,||\, \ldots \,||\, (k_{2i-1} \& m) \oplus k_{2i} \,||\, \ldots \,||\, (k_{2n-1} \& m) \oplus k_{2n} \\ \text{Verify}(k,m,\sigma) &= (\text{Sign}(k,m) == \sigma)\end{aligned}$$

This signing scheme uses only logical operations and the running time of the signing algorithm is $\mathcal{O}(n)$.

Claim 3.1 — Upper bound on winning probability of any adversary is $\frac{1}{2^n}$.

We will prove this by induction of the number of flipped bits between the message queried and the message whose forgery is being sent.

Proof. **Base Case**: On flipping one bit of the message Pr[A wins] = $1/2^n$
Let the query message be m and the message in forgery be m'.
Suppose the i^{th} bit of m is flipped to get m'.
Note that the sign has n components of a similar signing logic :- $(k_i \$ m) \oplus k_{i+1}$.
One flipping the i^{th} bit in m only the i^{th} bit of this component may be affected. We have to show that this bit might be 0 or 1 with equal probabilities.

No.	k_i	m	k_{i+1}	$(k_i \& m) \oplus k_{i+1}$
1	0	0	0	0
2	0	0	1	1
3	0	1	0	0
4	0	1	1	1
5	1	0	0	0
6	1	0	1	1
7	1	1	0	1
8	1	1	1	0

Table 1: Possible Cases for i^{th} bit of $(k_i \& m) \oplus k_{i+1}$

We can deduce from this table that even if adversary has a message signature pair and he flips one bit of message the probability of the corresponding signature having 0 or 1 is 1/2.
For example, if query message and signature received had i^{th} bit 0 and 1. 2 cases of the table are satisfied the 2^{nd} and 6^{th} case.
On flipping the bit in m we know have corresponding cases 4 and 8 which are both equally likely.
Now there are n such equivalent components in the signature and hence the probability of guessing becomes $1/2^n$.
Induction Hypothesis: If there are r flipped bits between the 2 messages then Pr[A wins] = $1/2^{nr}$.
Induction Step: Suppose there are r+1 flipped bits between the 2 messages. As we showed in the base case that on flipping one bit the probability the adversary can guess the new message is $1/2^n$ and since each bit is independent of each other we can say that

$$\begin{aligned}\text{Pr[A wins in r+1 flipped bits]} &= \text{Pr[A wins in r flipped bits]} \cdot \text{Pr[A wins in 1 flipped bit]} \\ &= 1/2^{nr} \cdot 1/2^n \\ &= 1/2^{n(r+1)}\end{aligned}$$

Hence proved. The upper bound on adversary winning is $1/2^n$.

□

§3.2 Part b)

We can show that if there exists a ppt adversary A that can break the MAC security of given scheme then there exists a ppt adversary which either breaks PRF security or one-query MAC security.
We will prove this using hybrid worlds and showing the corresponding reductions.

World 0

Challenger $\mathcal{C}$		**Adversary** $\mathcal{A}$
$k \leftarrow\$ \mathcal{K}_1 \times \mathcal{K}_2$		
	$\xleftarrow{m}$	Polynomially
$r \leftarrow\$ \mathcal{X}$		
$\sigma_1 = Sign_1(k_1, m)$		
$\sigma = (r, F(k_2, r) \oplus \sigma_1)$	$\xrightarrow{\sigma}$	many queries
	$\xleftarrow{\text{Forgery: } (m'\sigma')}$	

Adversary wins if $Verify(k, m', \sigma') = 1$

Hybrid World 0

Challenger $\mathcal{C}$		**Adversary** $\mathcal{A}$
$k \leftarrow\$ \mathcal{K}_1 \times \mathcal{K}_2$		
	$\xleftarrow{m}$	Polynomially
$r \leftarrow\$ \mathcal{X}$		
$\sigma_1 = Sign_1(k_1, m)$		
$\sigma = (r, f(r) \oplus \sigma_1)$	$\xrightarrow{\sigma}$	many queries
	$\xleftarrow{\text{Forgery: } (m'\sigma')}$	

Adversary wins if $Verify(k, m', \sigma') = 1$

We can say that World 0 and Hybrid World 0 are negligibly close by PRF security.

Difference in World 0 and Hybrid World 0

Challenger $\mathcal{C}$		**Reduction** $\mathcal{B}_1$		**Adversary** $\mathcal{A}$
$k_2 \leftarrow\$ \mathcal{K}_2$		$k_1 \leftarrow\$ \mathcal{K}_1$		
$b \leftarrow\$ \{0,1\}$				
if $b = 0$ then	$\xleftarrow{r}$	$r \leftarrow\$ \mathcal{X}$	$\xleftarrow{m}$	
$y = F(k_2, x)$		$\sigma_1 = Sign_1(k1, m)$		
else $y \leftarrow\$ \{0,1\}^n$	$\xrightarrow{y}$	$\sigma = (r, y \oplus \sigma_1)$	$\xrightarrow{\sigma}$	
	$\xleftarrow{r'}$	$\sigma'' = \sigma' \oplus Sign_1(k_1, m')$	Forgery: m' , $(r'\sigma')$ $\xleftarrow{}$	
	$\xrightarrow{y'}$	if $y' = \sigma''$ then $b' = 0$		
	$\xleftarrow{b'}$	else $b' = 1$		

In this case, the adversary receives **World 0** if $b = 0$, else he receives **Hybrid World 0**.
Therefore $\mathsf{Adv}_{\mathcal{B}} = \Pr[\text{A wins in W0}] - \Pr[\text{A wins in Hybrid World 0}]$.
Hence the worlds are negligibly close.

Hybrid World 1

Challenger $\mathcal{C}$		**Adversary** $\mathcal{A}$
$k \leftarrow\$ \mathcal{K}_1 \times \mathcal{K}_2$		
	$\xleftarrow{m}$	Polynomially
$r \leftarrow\$ \mathcal{X}$ without replacement		
$\sigma_1 = Sign_1(k_1, m)$		
$\sigma = (r, f(r) \oplus \sigma_1)$	$\xrightarrow{\sigma}$	many queries
	Forgery: $(m'\sigma')$ $\xleftarrow{}$	
	Adversary wins if $Verify(k, m', \sigma') = 1$	

We can say that Hybrid World 0 and Hybrid World 1 are negligibly close due to super polynomial size of X.

Hybrid World 2

Challenger $\mathcal{C}$		**Adversary** $\mathcal{A}$
$k_1, k_2 \leftarrow\$ \mathcal{K}_1 \times \mathcal{K}_2$		
	$\xleftarrow{m}$	Polynomially
$r \leftarrow\$ \mathcal{X}$ without replacement $\sigma_1 = Sign_1(k_1, m)$ $y \leftarrow\$ \mathcal{T}$		
$\sigma = (r, y \oplus \sigma_1)$	$\xrightarrow{\sigma}$	many queries
	Forgery: $(m'\sigma')$	
	Adversary wins if $Verify(k, m', \sigma') = 1$	

Hybrid World 1 is equivalent to Hybrid World 2 as randomness is sampled without replacement.

Hybrid World 3

Challenger $\mathcal{C}$		**Adversary** $\mathcal{A}$
$k_1, k_2 \leftarrow\$ \mathcal{K}_1 \times \mathcal{K}_2$		
	$\xleftarrow{m}$	Polynomially
$r \leftarrow\$ \mathcal{X}$ without replacement $y \leftarrow\$ \mathcal{T}$		
$\sigma = (r, y)$	$\xrightarrow{\sigma}$	many queries
	$\xleftarrow{m}$	
$r \leftarrow\$ \mathcal{X}$ without replacement $\sigma_1 = Sign_1(k_1, m)$		
$\sigma = (r, y \oplus \sigma_1)$	$\xrightarrow{\sigma}$	
	$\xleftarrow{m}$	Polynomially
$r \leftarrow\$ \mathcal{X}$ without replacement $y \leftarrow\$ \mathcal{T}$		
$\sigma = (r, y)$	$\xrightarrow{\sigma}$	many queries
	Forgery: $(m'\sigma')$	
	Adversary wins if $Verify(k, m', \sigma') = 1$	

We know that on XORing a random string to any string makes the outcome look random. So if we output a random string instead of $y \oplus \sigma_1$ for all queries after a randomly chosen query the 2 worlds will still be negligibly close!
We now claim that if there exists a ppt adversary $\mathcal{A}$ which wins above game then we can create a reduction $\mathcal{B}_2$ to break security of $\mathcal{I}_1$.

Breaking One-query MAC Security

Challenger $\mathcal{C}$		**Reduction** $\mathcal{B}_2$		**Adversary** $\mathcal{A}$
$k_1 \leftarrow\$ K$				
$\sigma_1 = Sign_1(k_1, m)$	$\xleftarrow{m'}$	$r' \leftarrow\$ \mathcal{X}$	$\xleftarrow{m}$	
		$y' \leftarrow\$ \mathcal{T}$		
	$\xrightarrow{\sigma_1}$	$\sigma = (r', y' \oplus \sigma_1)$	$\xrightarrow{\sigma}$	
		$r \leftarrow\$ \mathcal{X}$	$\xleftarrow{m}$	
		$y \leftarrow\$ \mathcal{T}$	$\xrightarrow{(\text{ r, y })}$	
	$\xleftarrow{\text{Forgery: } m'' \text{ , } y' \oplus \sigma''}$		$\xleftarrow{\text{Forgery: } m'' \text{ , } (r''\sigma'')}$	

We claim that r" chosen by the adversary must be equal to one of the randomness r sent to it earlier for it to break the encryption with non negligible probability.
This is because each y corresponding to r is calculated on the fly and the adversary has $1/|\mathcal{T}|$ probability of guessing it.
So now if r'' given by adversary corresponds to one of the r_i given to it then
$\Pr[\mathcal{B}_2 \text{ wins}] = 1/\mathcal{Q} * \Pr[(\mathcal{A}wins)]$ which is non negligible.
Here Q represents the number of queries made by the adversary which is polynomial.

§4 Question 4

§4.1 Part a)

We can prove the contrapositive, that is, if an adversary $\mathcal{A}$ can break CipherText Integrity, then we can use it to create a reduction $\mathcal{B}$ that breaks either CCA security or Security against Plaintext Integrity attacks.

The reduction works on the fact that when an adversary sends a ciphertext which breaks the Ciphertext Integrity then there are 2 possible cases.

1. Ciphertext sent is of some previous message but the ciphertext is different. In this case, we break CCA security.
2. Ciphertext sent is of some new message. In this case, we break Plaintext Integrity.

Breaking CCA Security

Challenger $\mathcal{C}$		**Reduction** $\mathcal{B}$		**Adversary** $\mathcal{A}$
$b \leftarrow\$ \{0,1\}$				
$k \leftarrow\$ K$				
$y_i = Enc(k, m_{ib})$	$\xleftarrow{(m_i\ ,\ r)}$	$r \leftarrow\$ \mathcal{M}$	$\xleftarrow{\text{Enc } m_i}$	
	$\xrightarrow{y_i}$		$\xrightarrow{y_i}$	Polynomially many
$m' = Dec(k, ct)$	$\xleftarrow{\text{Dec } ct}$		$\xleftarrow{ct}$	
	$\xrightarrow{m'}$	if $m' = m_i$ then $b' = 0$		
	$\xleftarrow{b'}$	else $b' = 1$		

This reduction works on the fact that if Challenger $\mathcal{C}$ chooses bit 0 then Adversary $\mathcal{A}$ can break Ciphertext Integrity but if Challenger $\mathcal{C}$ chooses bit 1, then Adversary $\mathcal{A}$ essentially gets ciphertext of some random message so he cannot create a new ciphertext of some message queries earlier.

Breaking Plaintext Integrity

Challenger $\mathcal{C}$		**Reduction** $\mathcal{B}$		**Adversary** $\mathcal{A}$
$k \leftarrow\$ K$				
$y_i = Enc(k, m_i)$	$\xleftarrow{m_i}$		$\xleftarrow{m_i}$	
	$\xrightarrow{y_i}$		$\xrightarrow{y_i}$	
	$\xleftarrow{ct'}$		$\xleftarrow{ct'}$	

This reduction works on the fact that the ciphertet sent is of some new message and so the adversary can break the plaintext integrity of the encryption scheme.

§4.2 Part b)

To satisfy CCA security we use the 'Encrypt-then-MAC' scheme with a CPA secure encryption scheme. We then create a plaintext integrity attack on it.
The encryption scheme is given by:

$$\begin{aligned} Enc((k_1,k_2),m) &:= r \leftarrow\$ \mathcal{M} \\ &:= ct = r, F(k_1,r) \oplus m, F(k_2, F(k_1,r) \oplus m \oplus F(k_2,r)) \\ Dec((k_1,k_2),(ct_1,ct_2,ct_3)) &:= m = F(k_1,ct_1) \oplus ct_2 \\ &:= m \text{ if } F(k_2,ct_2) = ct_3 \text{ else } 0^n \end{aligned}$$

The encryption scheme basically uses a CPA secure encryption scheme $r, F(k,r) \oplus m$ and uses a UFCMA secure MAC scheme - CBC-MAC with a different key for F. We use two different keys to ensure no related key attacks.
This scheme does not satisfy 'Authenticated Encryption' but is 'CCA secure'. This scheme would have been an 'Authenticated Encryption' scheme if we returned $\perp$ if the sign did not verify since the encryption scheme is CPA and the MAC is UFCMA secure.
We can prove the CCA security of our scheme by porving that if we break CCA of our scheme, we can break the CCA security of the scheme with output as $\perp$ if sign doesn't match.
The reduction works as follows:

CCA Security

Challenger $\mathcal{C}$		**Reduction $\mathcal{B}$**		**Adversary $\mathcal{A}$**
$b \leftarrow\$ \{0,1\}$				
$k \leftarrow\$ K$				
$ct_i = Enc(k, m_{ib})$	$\xleftarrow{(m_{i0}\ ,\ m_{i1})}$		$\xleftarrow{(m_{i0}\ ,\ m_{i1})}$	
	$\xrightarrow{ct_i}$		$\xrightarrow{ct_i}$	
	$\xleftarrow{\text{Dec } ct'_j}$		$\xleftarrow{\text{Dec } ct'_j}$	
$m'_j = Dec(k, ct'_j)$	$\xrightarrow{m'_j}$	$m''_j = 0^n$ if $m'_j = \perp$ else m'_j	$\xrightarrow{m''_j}$	

Hence, the given scheme is CCA secure.
The Plaintext Integrity Attack works because the Adversary can send any ciphertext and it will never return a $\perp$ because if the sign doesn't verify, the decryption algorithm doesn't return a $\perp$ but 0^n.
This attack hinges on the fact that a CCA secure scheme is not necessarily an Authenticated Encryption.

Plaintext Integrity Attack

Challenger $\mathcal{C}$		**Adversary $\mathcal{A}$**
	$\xleftarrow{ct}$	$ct \leftarrow\$ \{0,1\}^{3n}$

§5 Question 5

§5.1 Part a)

We can use the Extended Euclidean Algorithm to find the modular inverse of any number in $\mathbb{Z}_p$
We apply the algorithm on a and p. Since p is a prime $gcd(a, p) = 1$ and so, we get integers x and y such that $ax + py = 1$.
Therefore $ax = 1 \mod p$. We can change x to $x + np$ such that $x + np \in \mathbb{Z}_p$. Now,

$$a(x + np) = ax + anp = ax = 1 \mod p$$

Therefore, the inverse of a is $x + np$.
To show that the inverse is unique, we use Proof by Contradiction.
Assume there exist two inverse for a, b and b' such that $b \neq b'$. Thus,

$$\begin{aligned} & ab = 1 \mod p \text{ and } ab' = 1 \mod p \\ \implies & ab = ab' \mod p \implies a(b - b') = 0 \mod p \end{aligned}$$

Therefore, $p|a(b - b')$. Since p is a prime number either $p|a$ or $p|b - b'$. But since $a \in \mathbb{Z}_p*$, the first is incorrect and since $b, b' \in \mathbb{Z}_p$, $b - b' \in \mathbb{Z}_p \setminus \{0\}$. Therefore both statements are false and there is a contradiction.
The contradiction arises because of our assumption that there exist two inverses. Hence, the inverse must be unique.

§5.2 Part b)

Since, n has to be a product of two primes, both the factors of $h(y)$ must have a solution in $y \in \mathbb{Z}_p$ such that they are prime.
We can construct a polynomial $h(y) = (y + 1)(y + 2) = y^2 + 3y + 2$ for modulus 6.
$h(y) = 0 \mod 6$ implies $(y + 1)(y + 2) = 0 \mod 6$. This polynomial of degree 2 has 3 solutions for $y \in \mathbb{Z}_p$. The solutions are $y = 1, 4, 5$.

§5.3 Part c)

By Fermat's Little Theorem, we can say that since a is coprime to p, $a^{p-1} = 1 \mod p$. Since $ord(a)$ is smallest such that $a^{ord(a)} = 1 \mod p$.
Suppose $ord(a)$ doesn't divide $p - 1$. Therefore we can write $p - 1$ in terms of $ord(a)$ by the division algorithm.

$$p - 1 = n \cdot ord(a) + r \text{ such that } 1 \leq r < ord(a)$$

Hence,

$$\begin{aligned} a^{p-1} = a^{n \cdot ord(a) + r} = (a^{ord(a)})^n \cdot a^r = 1 \mod p \\ \implies a^r = 1 \mod p \because a^{ord(a)} = 1 \mod p \end{aligned}$$

This implies $a^r = 1 \mod p$. But since $r < ord(a)$, and $ord(a)$ is the smallest satisfying $a^t = 1 \mod p$, this is a contradiction.
This arises due to our assumption that $ord(a)$ doesn't divide $p - 1$. Hence, we prove by contradiction that $ord(a)$ divides $p - 1$.

§5.4 Part d)

This variant is a special case of DDH where $b = a^2$. So, if an adversary breaks the DDH problem, we can create a reduction that uses it to break the powersDDH problem.

powersDDH

Challenger $\mathcal{C}$		**Reduction** $\mathcal{B}$		**Adversary** $\mathcal{A}$
$bit \leftarrow\$ \{0,1\}$				
if $bit = 0$ then $b = a^2, c = a^3$	$\xrightarrow{(g\ ,\ g^a\ ,\ g^b\ ,\ g^c)}$		$\xrightarrow{(g\ ,\ g^a\ ,\ g^b\ ,\ g^c)}$	
	$\xleftarrow{bit'}$		$\xleftarrow{bit'}$	

This works because we send $b = a^2$ and so $ab = a^3$ to the adversary for DDH.

§5.5 Part e)

We can define a series of Hybrid Worlds and in each world we change the rows such that with each hybrid world change, one less row satisfies the property of a Rank 1 matrix.
We define Hybrid World i such that the last $t - i$ rows satisfy the property that i^{th} row is $\lambda_i v \forall i$. Hence, World 0 is equivalent to Hybrid World 0 and World 1 is equivalent to Hybrid World t.
Now, we show that if an adversary can distinguish between Hybrid Worlds i and $i + 1$, then we can use it to create a reduction that breaks DDH Problem.
Th reduction is given (g, g^a, g^b, g^c) where c is either equal to ab or chosen at random from the group elements. Then the reduction needs to construct a matrix M such that if $c = ab$, we get Hybrid World i, else we get Hybrid World $i + 1$. Let's call this algorithm $ConstructMatrix$. We show what this algorithm is after the reduction.

Rank 1 Matrix DDH - Hybrids i and $i + 1$

Challenger $\mathcal{C}$		**Reduction** $\mathcal{B}$		**Adversary** $\mathcal{A}$
$bit \leftarrow\$ \{0,1\}$				
if $bit = 0$ then $c = ab$	$\xrightarrow{(g\ ,\ g^a\ ,\ g^b\ ,\ g^c)}$	$g^M = ConstructMatrix[i]$	$\xrightarrow{(g\ ,\ g^M}$	
	$\xleftarrow{bit'}$		$\xleftarrow{bit'}$	

The $ConstructMatrix[i]$ algorithm works as follows:

1. Choose $v_1 \dots v_t$ as t values from $\mathbb{Z}_p^*$ uniformly at random. We create v as a vector with its elements from $v_1 a, v_2 a, \dots v_t a \mod q$.
2. For the $n - i - 1$ rows, we choose λ_i and we can compute $g^{M[i][j]} = g^{\lambda_i a v_j} = (g^a)^{\lambda_i v_j}$, since we know the value of g, g^a, λ_i and v_j. Hence, these rows satisfy the Rank-1 property.
3. For the $n - i$ row which changes between the Hybrid Worlds, we choose $\lambda = b$. Hence, each element of this row is $g^{\lambda v_j a} = g^{v_j ab} = (g^{ab})^{v_j}$. So, we can calculate since we know g^{ab} from DDH challenger.
4. If the DDH challenger sends g^{ab}, it is Hybrid World i since that row follows the property, but if the challenger sends g^c for some random c, then it is Hybrid World $i + 1$ with probability very close to 1.

Since the reduction can distinguish between any two Hybrid Worlds, the advantage between the two Hybrid Worlds is negligible. Since there are t Hybrid Worlds, which is polynomial in n, the advantage between World 0 and World 1 is negligible. Thus, if DDH is hard on G, this Rank 1 Matrix Problem is also hard for G.
Hence, proved.

§6 Acknowledgements

We have used the style file from here[1] to typeset and the style file from here[2] for cryptographic games and protocols to produce this document.

[1] https://github.com/vEnhance/dotfiles/blob/main/texmf/tex/latex/evan/evan.sty
[2] https://github.com/arnomi/cryptocode