59/100

G. Bhanu Prakash
3706294

1)

a) Two minimal domain generalization tuples that ensure 3-anonymity with minimum information loss can be done with respect to tuple (B_1, S_0, Z_1) and tuple (B_1, S_0, Z_2).

4

$B_1 \longrightarrow \{73\text{-}82\}$

$S_0 = \{M, F\} \longrightarrow S_1 = \{*\}$

$Z_1 = \{2018*, 2019*, 2017*\}$, $Z_2 = \{201**\}$

Table for B_1, S_1, Z_1

DoB	Sex	Zip	Salary
73-82	*	2018*	67,000
73-82	*	2018*	66,000
73-82	*	2019*	98,000
73-82	*	2019*	82,000
73-82	*	2017*	67,000
73-82	*	2017*	98,000
73-82	*	2018*	66,000
73-82	*	2019*	82,000
73-82	*	2017*	82,000

Table for B_1, S_0, Z_2

DoB	Sex	Zip	Salary
73-82	M	201**	67,000
73-82	F	201**	66,000
73-82	M	201**	98,000
73-82	M	201**	82,000
73-82	F	201**	67,000
73-82	F	201**	98,000
73-82	M	201**	66,000
73-82	F	201**	82,000
73-82	F	201**	82,000

$l = 2$

$l = 4$

1) b) l-diversity: A dataset is l-diverse if each of its quasi identifiers has atleast l "well represented" values for each sensitive attribute to ensure the values are distinct.

What is the value of l?

-1

→ most frequent values does not appear too often when compared to less frequent values in quasi identifier group.

Anonymizing the given table.

DOB	Sex	Zip	Salary
20/01/73	*	201**	67,000
13/04/82	*	201**	66,000
28/02/73	*	201**	98,000
1/02/73	*	201**	82,000
23/03/82	*	201**	67,000
11/05/73	*	201**	98,000.
11/05/73	*	201**	66,000
7/11/80	*	201**	82,000
7/11/80	*	201**	82,000.

The above table confirms atleast 1-well represented values are present for each sensitive attribute.

2) a) Differential privacy is a method of publicly disclosing information about a dataset by defining the patterns of groups within it while withholding information about individuals. Differential privacy is based on the principal that if the impact of the single arbitrary substitution in the database is small enough, the query result cannot be used to infer much about single entity, and thus provides privacy.

→ Differential privacy can be achieved by adding "noise" to an aggregate query result to prevent joining or linking attack without significantly changing the result.

8

2) b) Given D_1 and D_2 are adjacent databases.

$\mathcal{A}$ is randomized sanitizing algorithm.

$f : D \to R$

$f(D)$ denote the true answer of query

noise is generated from geometric distribution Geom(α)

$$\Pr[\mathcal{A}_f(D_1) = r]$$
$$= \Pr[f(D_1) + \text{Geom}(\alpha) = r]$$
$$= \Pr[\text{Geom}(\alpha) = r - f(D_1)]$$
$$= (\alpha - 1/\alpha + 1)\, \alpha^{-|r - f(D_1)|}$$

this probability value is for geometric distribution.

-2

Similarly for D_2, we have.

$$\Pr[\mathcal{A}_f(D_2) = r] = (\alpha - 1/\alpha + 1)\, \alpha^{-|r - f(D_2)|}$$

Therefore $$\frac{\Pr[\mathcal{A}_f(D_1) = r]}{\Pr[\mathcal{A}_f(D_2) = r]} = \alpha^{-(|r - f(D_1)| - |r - f(D_2)|)}$$
$$= e^{-\frac{\epsilon}{\Delta f}(|r - f(D_1)| - |r - f(D_2)|)} \leq e^{\epsilon}$$

as $$\frac{|f(D_1) - f(D_2)|}{\Delta f} \leq 1$$

Hence $\Pr[\mathcal{A}(D_1) \in S] \leq e^{\epsilon} \cdot \Pr[\mathcal{A}(D_2) \in S]$.

2) c) Differentially private answer from an database query can be computed by

$$f(D) + Lap\left(\frac{\Delta f}{\epsilon}\right)$$

where $f(D)$ is the true answer of the query f on D.

Let's consider a table

Name	Age	Smoke	Random Sugar	Cancer
Alice	42	1	11.1	0
Bob	43	1	9.3	1
Ann	70	1	8.7	1
Frank	45	0	13.6	0
Lucy	23	1	7.8	0

Differential private answer for number of patients who smoke can be computed by the Sensitivity of the Count query is $\Delta f = 1$, $\epsilon = 0.2$

$$Lap\left(\frac{\Delta f}{\epsilon}\right) = Lap\left(\frac{1}{\epsilon}\right)$$
$$= Lap\left(\frac{1}{0.2}\right)$$
$$= Lap(5)$$

Let choose $p = 0.64334$ and corresponding laplace distribution value is

$$x = -5 \times sign(0.64334 - 0.5) \times \ln(1 - 2|0.64334 - 0.5|)$$
$$= 1.689126$$

The number of patients who smoke is $f(D) = 4$.

Generated noise $= 1.689126$.

Adding noise to original value.

i.e., $f(D) + Lap\left(\frac{\Delta f}{\epsilon}\right) = f(D) + Lap(5) = f(D) + x$

$= 4 + 1.689126 = 5.689126$.

(3)

5

Given

$n = 323$.

Public keys $Pk_1 = (e, n) = (29, 323)$

$Pk_2 = (f, n) = (71, 323)$

$c_1 = 241$

$c_2 = 129$.

from the given n two primes are

$(p, q) = (17, 19)$.

$\phi(n) = (p-1)(q-1)$

$\phi(n) = (16)(18) = 288$

To decipher we need to find d

$d \times e = 1 \bmod \phi(n) \Rightarrow d = e^{-1} \bmod (288)$

d can be calculated using Extended Euclidean algorithm.

gcd (288, 29) is.

$r_i = r_{i-2} - q_{i-1} \times r_{i-1}$, $i \geq 2$	$s_i = s_{i-2} - q_{i-1} \times s_{i-1}$	$t_i = t_{i-2} - q_{i-1} \times t_{i-1}$
$r_0 = a = 288$, $r_1 = b = 29$.	$s_0 = 1$, $s_1 = 0$	$t_0 = 0$, $t_1 = 1$
$r_2 = 288 - (9) \times 29 = 27$	$s_2 = 1 - (-9) \times 0 = 1$	$t_2 = 0 - (9) \times 1 = -9$
$r_3 = 29 - (1) \times 27 = 2$	$s_3 = 0 - (1) \times 1 = -1$	$t_3 = 1 - (1) \times (-9) = 10$
$r_4 = 27 - (13) \times 2 = 1$	$s_4 = 1 - (13) \times (-1) = 14$	$t_4 = -9 - (-13) \times 10 = -139$

gcd (288, 29) $= (a \times s_4) + (b \times t_4)$

$= 288 \times 14 + 29 \times (-139) = 1$

$= (288 \times 14) \bmod 288 + 29 \times (-139) \bmod 288 = 1 \bmod 288$

$= 0 + 29 \times (-139) \bmod 288 = 1 \bmod 288$.

$= 29 \times 149 \bmod 288 = 1$

$\Rightarrow \frac{1}{29} \bmod 288 = 149$. $\Rightarrow d_1$

Similarly we can calculate d_2 as

$gcd(288, 71)$

$\Rightarrow \frac{1}{71} \bmod 288 = 215. \Rightarrow d_2.$

$c_1 = 241$

$m_1 = c_1^{d_1} \bmod n$

$= 241^{149} \bmod 288$

$= 90.$

$c_2 = 129$

$m_2 = c_2^{d_2} \bmod n$

$= 129^{215} \bmod 288$

$= 90.$

So, the plain text message $m = 90.$

4)

Given

user wants to outsource a set of l files.

denoted by $F = \{f_1, f_2, \cdots f_l\}$.

Construction of Bloom filter:

→ Construct an Array ~~of Bloom filters~~ of length n.

→ Create a Bloom Filter, denoted by BF for the file set F.

→ Send files $F = \{f_1, f_2 \cdots f_l\}$ to BF.

This solution is already mentioned in the question

Question asks for a new solution.

For integrity checking:

Bloom Filter an Array is Set to 0.

→ Bloom Filter can judge whether $x \in F$ or not.

→ Bloom Filter use k-independent hash functions with value field $\{1, \cdots m\}$. it makes assumption

-4

that hash functions map each other item in random number in the range $\{1, \ldots m\}$.

→ For each element $x \in F$, the bit ~~with~~ k-hash functions are set to 1.

→ So one can check whether all mapped bits are 1 to assume that $y \in F$.

→ In order to detect if an advesary alters or replace a file with different one we can introduce each entry in bloom filter as a small counter rather than a single bit.

The corresponding counters are incremented when an item is inserted and corresponding counters are decreased when an item is deleted.

Thus if an advesary replaces (or) deletes a file in Bloom filter we can identify by ~~delete~~ difference in the counter values.

5) a)

Given public key = (g, n)

$= (10, 187)$

6.5

Private key = $(P, q) = (11, 17)$.

$C = 81$.

To compute the plain-text message.

$\frac{c}{p} = C^{\frac{p-1}{2}} \bmod P$

$\frac{c}{q} = C^{\frac{q-1}{2}} \bmod q.$

} find out whether result is quadratic residue or QR & NQR.

$\left(\frac{c}{p}\right) = C^{\frac{11-1}{2}} \bmod 11$

$= C^5 \bmod 11$

$= 81^5 \bmod 11 = 1$

$\left(\frac{c}{q}\right) = C^{\frac{17-1}{2}} \bmod 17.$

$= 81^8 \bmod 17.$

$= 1.$

-1

as $\left(\frac{c}{p}\right) = \left(\frac{c}{q}\right) = 1$ then $m = 1$. ✗

$m = 0$

5) b)

Given public key = $(g, n) = (10, 187)$

private key = $(P, q) = (11, 17)$

$c_1 = g^1 r^3 \bmod n$

$= 10^1 . 5^3 \bmod 187.$

$= 128.$

$c_2 = g^1 r^4 \bmod n$

$= 10^1 5^4 \bmod 187$

$= 79.$

For decrypting the message. we must check whether $\left(\frac{c}{p}\right) * \left(\frac{c}{q}\right)$ are quadratic residue or Non-quadratic residue.

$$m = \begin{cases} 0 & \text{if value is QR} \\ 1 & \text{if value is NQR.} \end{cases}$$

$$\left(\frac{c_1}{p}\right) = c_1^{\frac{11-1}{2}} \bmod p$$

$$= 128^5 \bmod 11 = -1 \text{ (NQR)}$$

$$\left(\frac{c_1}{q}\right) = c_1^{\frac{17-1}{2}} \bmod q$$

$$= 128^8 \bmod 17 = 1 \text{ (QR)}$$

since $\left(\frac{c_1}{p}\right)$ is NQR $\left(\frac{c_1}{q}\right)$ is QR result is NQR.

So the message $m = 1$.

Similarly.

$$\left(\frac{c_2}{p}\right) = c_2^{\frac{11-1}{2}} \bmod p$$

$$= 79^5 \bmod 11 = 10 = -1$$

$$\left(\frac{c_2}{q}\right) = c_2^{\frac{17-1}{2}} \bmod q$$

$$= 79^8 \bmod 17 = 16 = 1$$

$\left(\frac{c_2}{p}\right)$ is NQR and $\left(\frac{c_2}{q}\right)$ is QR so result is NQR

Therefore $\left(\frac{c_1}{p}\right) * \left(\frac{c_2}{q}\right)$ = message $m = 1$.

and $\left(\frac{c_2}{p}\right) * \left(\frac{c_2}{q}\right)$ = message $m = 1$.

c_1 is not a valid GM ciphertext.

c_2 is a GM ct.

✗ not correct

-2.5

7) a) Need for Anonymous Communication Networks:

→ Anonymity is one of the most important tools available to counter balance the threat of unknown viewers and secure internet privacy.

→ Privacy and anonymity are essential in today's culture.

→ Law enforcement and legal protections are gradually being forced to adapt to this modern existence in virtual world.

Attacks on Anonymous Communication Networks:

→ Communication pattern Attack:

By looking at communication patterns a lot of useful information can be gained.

- In communicating parties when one party is sending other remains silent.

- The longer this pattern is observed, the less likely it is just an uncorrelated random pattern.

→ This attack can be mounted by a passive adversary that can monitor entry and exit mix nodes.

Ex: Law enforcement officers when they have a hunch that two parties are communicating they will mount this attack.

→ Packet Counting Attack:

- These types of attacks exploit the fact that some communication are easy to distinguish from others.

- If a participant sent unusual number of messages, a passive external attacker can spot these messages coming out from mix-networks.

- Packet counting attacks can be combined to get a "message frequency" attack.

→ Intersection Attack:

- ~~An~~ This Attack is based on observation that user's typically communicate with small number of parties.

- An attacker having information about ~~user~~ user's activity at any given time through repeated observation.

Ex: An user may query same websites in different timings. (not random). By performing an observation similar to an intersection on the sets of active users at different timings, one can gain valuable information.

7) b)

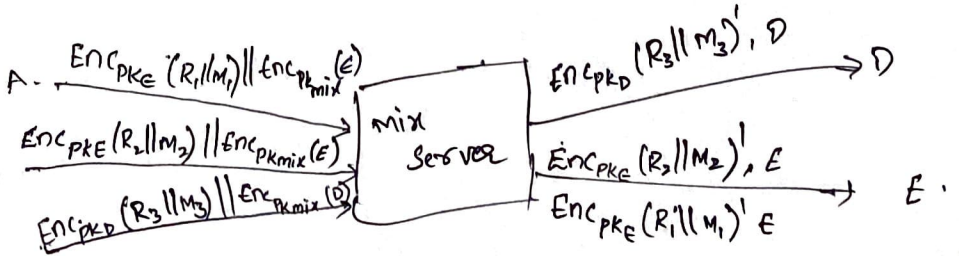
Mixnet:

- Mixnet Server shuffles messages and routes them. and mixnet server has the public keys of each user in the system.

- Each user have access to other user's public keys as well as mixnet server.

- By using Elgamal Re-encryption it is possible to use mix Server's public key to encrypt recievers's information.

- With Reciever's public key, the mix Server needs to re-pack Each incoming packet i.e., $Enc_{PKE}(R_1||M_1)$ to $Enc_{PKE}(R_1||M_1)^*$, so as to make them unlinkable, but the transformed ciphertext can still be recovered.



- Elgamal Re-encryption is applied to achieve this goal

  - let $pk = (q, g, y)$

  - Enc : on an input message m, and public key PK, randomly generates (r, s) and computes.

    $$e = [(c_0, c_1), (c_2, c_3)] = [(g^r, y^r \cdot m), (g^s, y^s)]$$

    out c as ciphertext for m.

  - Dec : For c_1 it computes $m_0 = \frac{c_1}{c_0^x}$ and $m_1 = \frac{c_3}{c_2^x}$.

  - Re-encryption : For a ciphertext c and randomization factor (r', s'), it computes another ciphertext c' as.

    - $c_0' = c_0 c_2^{r'}$, $c_1' = c_1 c_3^{r'}$,
    - $c_2' = c_2^{s'}$ and $c_3' = c_3^{s}$.

8) a) 5

Given health records are denoted by $D = (d_{i,j})_{m \times m}$ stored in the form of two dimensional Array.

→ Alice do not want to reveal index pair (u, v).

→ using pailier encryption to design 1-out-of-n OT.

Alice

Input $\sigma \in \{0, \ldots n-1\}$

$\sigma = (\sigma_0, \sigma_1)$ ← as Alice don't want to share (u, v).

$(PK, SK) \leftarrow KGen(), E_{PK}(\cdot), D_{SK}(\cdot)$ ← Decryption key

$e = (e_1, e_2, \ldots e_m)$

$e_{\sigma_2} = 1, \; e_i = 1 \; \forall i, i \neq \sigma_1$ ($e_i = 0$)

for $i = 1, \ldots m$

$c_i = E(e_i)$

$\xrightarrow{c_1, c_2 \ldots c_m}$

Bob

Input: $D = (d_{i,j})_{m \times m}$

$$\begin{pmatrix} d_{11} & \cdots & d_{1m} \\ d_{21} & \cdots & d_{2m} \\ \vdots & \vdots & \vdots \\ d_{m1} & \cdots & d_{mm} \end{pmatrix}$$

$c_1, c_2 \ldots c_m$

$c_i = \prod_{j=1}^{m} c_j^{d_{ij}} = E(d_{i\sigma_1})$

$i = 1, 2, \ldots m.$

$\xleftarrow{1\text{-out-of-}m \text{ OT}}$ $\{c_1, c_2 \ldots, c_m\}$

$c_{\sigma_0} \leftarrow OT^m_1(\sigma_0, \{c_0 \ldots c_m\})$

Protocol does not catch Alice's malicious activities.

σ_0

$c_{\sigma_0} = E(d_{\sigma_0 \sigma_1})$

$= d_{\sigma_0 \sigma_1}$

-2

ADD COMMITMENT to solve

→ from the above protocol Alice recieved only the information with given input not any other results.

→ Bob does not recieve any index pair (u, v) from Alice.

→ Communication complexity of given protocol is $O(\sqrt{n})$.

Step① → Alice encrypted input and sent $E(e)$ to bob.

4) b) designed protocol is Secure. it follows the steps.

step ①. Alice encrypts the input.

$E(e) = E(e_1), E(e_2) -- E(e_m)$. { Bob does not know index values }

and sends $E(e)$ to Bob.

This is not the security of the protocol.

step ②

Bob computes m encrypted answers.

$c_i = \prod_{j=1}^{m} c_j^{d_{ij}} = E(d_{i\sigma_1})$ where $i = 1, -- m$

Security?

-2

step ③. Alice & Bob jointly run 1-out-of-m OT protocol.

- Alice input σ_0.
- Bob's input $\{c_1, c_2 -- , c_m\}$
- Alice obtains c_{σ_0} as output.

step ④. Alice decrypts c_{σ_0} and gets

$x_{\sigma_0 \sigma_1} = D(c_{\sigma_0})$ { Alice only recieves the required info not any other records }.

5) a) computional complexity of above designed protocol

are :

Communication complexity $O(m) = O(\sqrt{n})$

- Communication cost for $E(e)$ = m cipher Texts.
- Communication cost for 1-out-of-m OT is $O(m)$.

this is communication. Where is computational cost?

-1

4

9) a) Given server holds a database with n elements.

Let us consider a database $x = (x_{i,j})$ ~~with~~

we can construct a protocol using PIR as PIR allows a user to retrive an item from a server in possession of a database without revealing which item is retrieved.

This is not correct

use 1-server (computational PIR using GM.

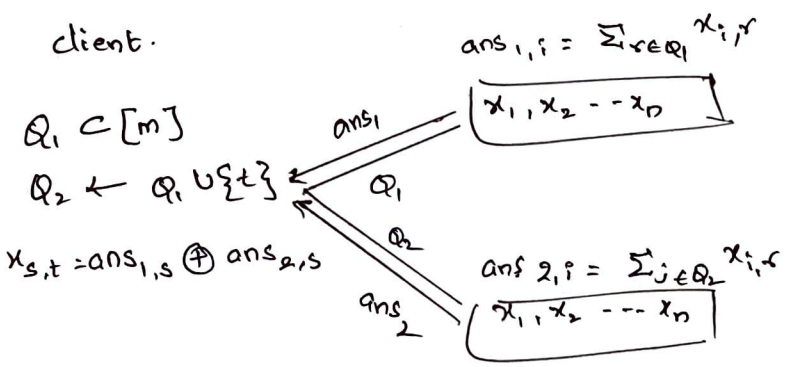
user constructs two queries to retrieve $x_{s,t}$:

- randomly generate a set $Q_1 \subset [m]$ such that $t \notin Q_1$.
- Construct $Q_2 = Q_1 \cup \{t\}$.

→ Servers compute an answer $ans_k = (ans_{k,1}, \cdots ans_{k,m})$

where $ans_{k,i} = \sum_{r \in Q_k} x_{i,r} \bmod 2, \quad k = 1, 2$.

-6



Query size : $|Q_k| = m = O(n^{1/2})$ ≐

9) b) Designed protocol preserves privacy :

The client ~~receives~~ generates a random string r to construct a pair of queries (Q_1, Q_2) as $Q_1 = Query(1, r)$

$Q_2 = Query(2, r)$.

→ Corresponding to index i query are sent to Q_1 to S_1 and Q_2 to S_2.

→ Upon recieving Q_i, each server computes an answer.

$ans_j = Answer(Q_i, x)$ and sends answer to client.

→ For two non-communicating servers they can't learn about i thus preserves the privacy.

→ for an index i returns x_i preserving correctness.

9) c) Security is established against a computationally bounded adversary.

Such that it preserves Computational privacy based on Cryptographic assumptions.

You did not use any crypto. scheme.

10) a) Given

Alice public key as $I = g^r$

g is the generator of Z_p^*

and p is large prime

5.5

Recovering r from I is hard due to hardness of the discrete logarithm problem.

we can use zero knowledge proof for designing one-round protocol.

step ①: Alice sends public key to Bob.

step ②: Bob uses the public key and creates an encrypted message and sends to alice.

step ③: Alice decrypts the key using private key and sends the message back to bob. confirming that Alice knows r

Alice		Bob
$q \leftarrow Z_p^*$. $s = g^q$	$\xrightarrow{s}$	ϕ .
b	$\xleftarrow{b}$	$b \leftarrow \{0,1\}$.
$z = q + b(r)$	$\xrightarrow{z}$	$g^z \overset{?}{=} s I^b$.

This is a 3-round protocol.

Question asks for 1-round protocol.

→ Repeat the process k-times to increase probability.

-2.5

10) b) If Alice doesn't know r she cannot convince Bob.

Step ①: Alice sends a public key. ← What's public my hero?

Step ②: Bob sends an encrypted ~~msg~~ message using public key.

Step ③: Alice is unable to decrypt the key as she does not ~~have~~ & know the value r.

Resulting Alice cannot convince Bob. −1

~~Alice~~ Bob

$q \rightarrow z_p^*$, $s = g^q$ —s→ s

b ←b— $b \leftarrow \{0, 1\}$

$z = q + b$ (?) —z→

Fails as ~~Alice~~ does not know r value and Bob cannot convince as result is wrong. (if Alice sent r by prediction).

10) c) the verifier cannot know the value of r as the verifier knows only s. ~~which is the~~ ↑ Why?

~~and~~ → Verifier cannot know anything ~~from~~ more than $q \in z_*^p$. −1

ii) Secure data aggregation:

Collecting aggregated energy consumption from smart meters in a secure manner i.e., protecting consumers meter's data privacy.

→ Aggregations of power consumption data is done at different levels for:

- Monitoring and predicting power consumption
- allocating and balancing loads and resources.
- administering power generation.

Model for Data Aggregation:

- Smart meters are used to transmit information using radio frequency electromagnetic fields.
- Smart meter Data collector: collects data from smart meter's.
- Aggregation is done in distributed manner, instead of centralizing.

→ The whole model is presented in the form of tree called Aggregation tree. Where:

- ~~where~~ smart meter Data collector is root of the tree.
- each node has Smart meters.

how to define privacy of a data agg. sch?

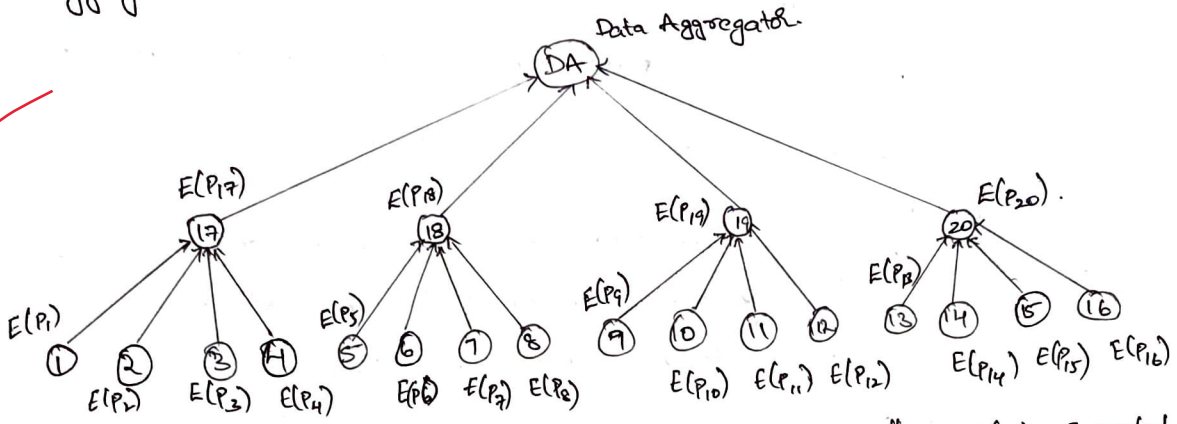
Securing Aggregation tree:

Each smart meter are encrypted by using additive homomorphic encryption (HE), say Pailier cryptosystem, which generates private & public key.

- public key given to all meters.
- Each smart meter has capability to perform ~~homo~~ homomorphic computation.
- Smart meter data management center decrypts the readings using private key ensuring privacy of users.

(2) a) Given a quadtree has 21 nodes where root node is data aggregator.

6



where $E(P_i)$ = Encrypted Smart meter P_i

→ All smart meters are encrypted by using paillier Cryptosystem. and public keys are given to each system.

Node 17 aggregates data from node 1, 2, 3, 4.

→ i.e., each node homomorphically aggregates data reading from it's child nodes.

Node id.	Aggregation
17	$E(O_{17}) = E(P_1 + P_2 + P_3 + P_4 + P_{17})$
18	$E(O_{18}) = E(P_5 + P_6 + P_7 + P_8 + P_{18})$

Similarly Data Aggregator computes. $E(O) = E(O_{17} + O_{18} + O_{19} + O_{20})$. and forwards it to Smart data management system.

→ Smart data management center decrypts $E(O)$ using the private key of the paillier encryption and obtains aggregated reading.

$$O = \sum_{i=1}^{20} P_i$$

Privacy:

→ Encrypted Smart meter readings do not leak any information

→ The Aggregated sum $O = \sum_{i=1}^{t} P_i$ do not leak any info about individual as long as t is sufficiently large.

12) b) from above quadtree homomorphic multiplication of cipher text can be defined as.

for $E(O_{17}) = E(P_1+P_2+P_3+P_4) + P_{17}$) so the cipher text multiplication done 4 times.

$E(O_{18}) = E(P_5+P_6+P_7+P_8) + P_{18}$

$E(O_{19}) = E(P_9+P_{10}+P_{11}+P_{12}) + P_{19}$)

$E(O_{20}) = E(P_{13}+P_{14}+P_{15}+P_{16}) + P_{20}$) multiplications 16 times.

$E(O) = E(O_{17}+O_{18}+O_{19}+O_{20})$ 20 times.

$E(O) = E(\sum_{i=1}^{20} P_i)$ fowards to Smart meter data management center.

So these are around 20 multiplication of cipher text multiplication.

b) ⇒ total # of multiplication ? -3

12) c) Latency:

Latency in aggregation Schema is defined as a time delay between the transmission of data between nodes.

i.e., there might be delay in transmission of data from child to parent nodes which is considered as latency in aggregation Schema.

The latency upon data aggregator can be considered by child nodes of data aggregator as aggregation is performed in distributed manner there nodes might be a delay in data aggregator child nodes. as they process

6) a)

Given Alice has 2 bit message (m_1, m_2)

Bob has 2 bit message (m_3, m_4)

$$y = f(m_1, m_2, m_3, m_4)$$

$$= (m_1 \wedge m_3) \oplus (m_2 \vee m_4)$$

Alice

Input : $\{m_1, m_2\}$

Alice sends m_1, m_2 messages by encrypting

$E(A) = E(m_1, m_2)$

$\xrightarrow[D_{SK}]{E(A)}$

Bob

Input : $\{m_3, m_4\}$.

$E(A)$

Bob decrypts the message. } Does not work

$m_1, m_2 \Rightarrow$ find $m_1 \wedge m_3$ using ? Set intersection protocol.

and performs $m_2 \vee m_4$.

and XOR operation.

Protocol is not correct.