COMP3632 A3

SIM, Kyu Doun

20306527

1 Data Privacy

(a)

1. k-anonymity

No. Complementary release attack.

2. Diﬀerential privacy

No. fixing the location. Less than 20m, you make one more, identity, location, gives 20m, less than 20m or more than 20m. Location,

Yes or no, don’t put noise.

Salary: scalar, location? Not a scalar? A vector maybe?

Add noise, make query,

20m,

Less than < 20, exact location of patient?

Yes or No, exaction location

Yes. This 20m is therefore,

Need to know the exaction location.

No.

3. Secure multiparty computation.

You’re the patient, not giving the exact location.

Company and person, Company cannot exact location

4. Private information retrieval: No. The data is private.

(b)

1. k-anonymity: NO

2. Diﬀerential privacy Yes

3. Secure multiparty computation: No

4. Private information retrieval. Yes

PIR: 10 min.

(c)

1. k-anonymity: No

Has only one DB. You don’t know anyways

2. Diﬀerential privacy: No

3. Secure multiparty computation: No

그 사람들끼리 need to talk to each other

4. Private information retrieval: Yes

(d)

1. k-anonymity: No. Eating is a daily habit can be compromised by complementary database. Differential privacy,

2. Diﬀerential privacy: Yes.

3. Secure multiparty computation: No

4. Private information retrieval: Yes

Lecture: iOS, has great interest in privacy, if we search sth on the web, apple can collect what we googled, using noise.

2 Multi-level Security Model

Management ppl: can read all files, no contradiction

Accounting: read any file marked management, cannot read a file marked finance

Acct -> management not covered

Still cannot read,

Single file cannot have 2 levels

Strange case (helps part (d)), it is meant to be difficult

(a)

1. Finn tries to read the file.

Failed operation. Finn can only read all Finance and Basic files

2. Batu tries to write to the file.

Successful operation. Batu is a lower security subject and it can write on higher security level objects. There is no change in the security level of the file.

3. Finn tries to write to the file.

Successful operation. Although we cannot exactly say that the Finance level is higher than Accounting level, for sure they are not in the same level. When Finn writes on the file,

Although Finn and Acc seems to be on the same level of security as it is higher than the Basic security level, they are its own security level. The High-water mark Bell-LaPadula model cannot help Finn to write on the Acc file. <wrong>

Security level changes to finance <wrong>

Successful

Goes to management

4. Ace tries to write to the file.

Successful operation. The security level of the file stays the finance. <wrong>

Success, stays at management

Management

5. Ace tries to read the file.

Fails.

Successful operation. Ace can read all Accounting and Basic files. <wrong>

6. Ma tries to read the file

Successful operation. Ma can read all files. Stays at management.

(b)

Proof: After Finn writes a file, regardless of its original security level, Acc can never read the file.

If Finn was able to successfully write on a file, according to the high-water mark Bell-Lapuadula, the file’s security level must be raised to a level that allowed Finn to write on. Therefore, regardless of any file, being it on the Basic level or at the Finance level even in Accounting level

which are the two security levels that Finn can write on, after Finn writes on the file, it will be placed or remain in the Finance security. No matter what operations come, Ace cannot read the file since the file is under the Finance

Gets higher than the + Finance, acct cannot get read

Any level. If finn writes,

Wherever it starts, if Finn writes on basic, goes to Finance,

(c)

Fail, all success

1. Finn tries to read the file.

Failed operation. Finn can only read all Finance and Basic files.

2. Batu tries to write to the file.

Successful operation. Batu is a lower security subject and it is allowed to write on higher security level objects. There is no change in the security level of the file.

3. Finn tries to write to the file.

Successful operation. In the high water Bell-LaPadula model, lower security subjects are allowed to write on higher security objects. There will be no change in the security level.

4. Ace tries to write to the file.

Successful operation. The security level of the file stays the same.

5. Ace tries to read the file.

Successful operation. Ace can read all Accounting and Basic files.

6. Ma tries to read the file

Failed operation. Ma is a lower security subject compared to the file in the Acct level.

(d)

Hierarchy. Must be higher to lower. If there is a circular, it cannot work.

3 Bitcoin

(a)

1 block per 10 minute = 1 block per 600 seconds

number of bitcoins per second is 6/600 = 0.01.

Total running cost is per second.

So $20/0.01 = $2000 is the minimum price of Bitcoin to cover the running costs.

(b)

1MB = 1048576 Bytes

1MB / 166 bytes = 6316.72289157, round it to 6316 transactions per block.

Then divide it by 600 seconds per blocks = 10.52666… transactions per second.

Therefore 10.53 transactions per second.

(c)

Number of bitcoins per second is 3/600 = 0.005, so 0.005 bitcoin per second is lost.

Monetary loss needed to be covered by transaction fees = $2000∗0.005 = $100 per second.

Therefore the mean transaction fee needs to 100/10.53 = $9.496676 = $9.50

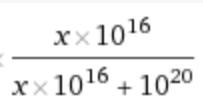
6000 / 6316 = 0.95

(d)

152 devices.

Assume that x machines are needed.

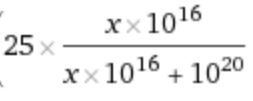
The total share of the bitcoin revenue will be



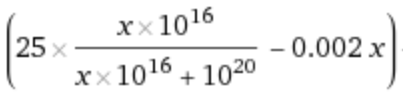
Total share of the bitcoin revenue is /( + ) = 0.00009999.

The number of bitcoins per second is 0.01 and the price is fixed to $2,500. Therefore for each second, the revenue would be 0.01 \* 2500 = $25 per second.

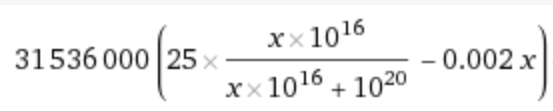
Multiply the (share of revenue) \* (revenue per second) will give $25 \* (ratio above) = 0.00249975002 per second.



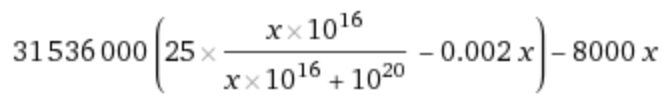
Proﬁt is revenue minus loss. Loss in this case would be the cost of running the miner, which is $ 0.002 per second per machine.



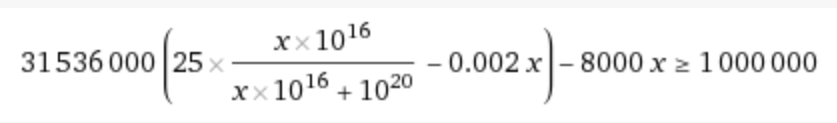
There are about 365 \* (60 \* 60 \* 24) = 31536000 seconds in a year.



Each machine costs $8000.



We are therefore solving the following inequality:



The solution for above is



Therefore the minimum required number of machines are 152 machines.

31536000 \* (25\*(x\*10^16)/(x\*10^16 + 10^20) - 0.002x) - 8000x>= 1000000

