COMP3632 A3

SIM, Kyu Doun

20306527

1 Data Privacy

(a)

1. k-anonymity

2. Diﬀerential privacy

3. Secure multiparty computation

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

(b)

1. k-anonymity

2. Diﬀerential privacy

3. Secure multiparty computation

4. Private information retrieval

(c)

1. k-anonymity

2. Diﬀerential privacy

3. Secure multiparty computation

4. Private information retrieval: No. Data is private

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.

Failed operation. 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.

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

Successful operation. Ma can read all files.

(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 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

(c)

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)

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.

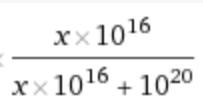
Therefore the mean transaction fee needs to 100/10.53 = $9.496676 = $9.50

(d)

152 devices.

Assume that x machines are needed.

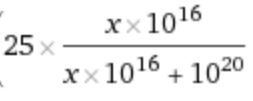
The total share of the bitcoin revenue will be



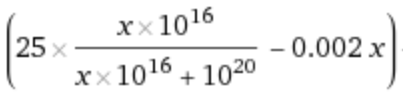
Total share of the bitcoin revenue is /( + ) = 0.00009999.

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.

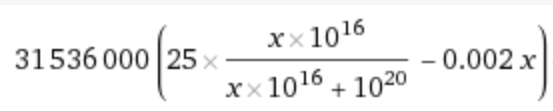
Multiply the (share of revenue) \* (revenue per second) will give $25 \* (ratio above) = 0.00249975002 per second.



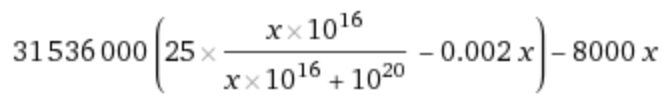
Proﬁt is revenue minus loss. Loss in this case would be the cost of running the miner, which is $ 0.002 per second per machine.



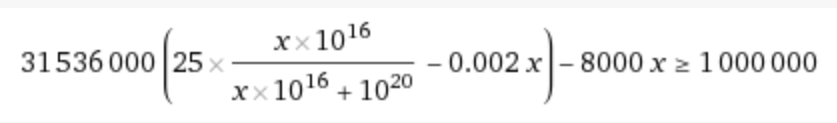
There are about 365 \* (60 \* 60 \* 24) = 31536000 seconds in a year.



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

