




COSC2536/2537 Security in Computing and Information Technology

Assignment 2

	<p>Assessment Type: Individual assignment; no group work. Submit online via Canvas→Assignments→Assignment 2.</p> <p>Marks awarded for meeting requirements as closely as possible. Clarifications/updates may be made via announcements/relevant discussion forums.</p>
	<p>Due date: Week 10, Friday the 2nd October 2020 11:59pm</p> <p>Deadlines will not be advanced, but they may be extended. Please check Canvas→Syllabus or via Canvas→Assignments→Assignment 2 for the most up to date information.</p> <p>As this is a major assignment in which you demonstrate your understanding, a university standard late penalty of 10% per each working day applies for up to 5 working days late, unless special consideration has been granted.</p>
	<p>Weighting: 45 marks (Contributes 45% of the total Grade)</p>

1. Overview

The objective of Assignment 2 is evaluating your knowledge on the topics covered mainly in Lecture 5 to 10. Topics include Privacy-preserving computations based on RSA, ElGamal and Paillier Cryptosystems; Digital Signature, Blockchain and Cryptocurrency, Digital Authentication & Security Protocols, and Digital Authorization and Intrusion Detection. However, topics covered in Lecture 1 to 10 are required as prerequisite. Assignment-2 will focus on developing your abilities in application of knowledge, critical analysis and decision making. Assignment 2 contains several problems related to the topics mentioned above. You are required to prepare your answers and upload them as a single PDF or Word document in CANVAS. Some of the questions require video demonstrations and code submission in the CANVAS. For a video demonstration, you are required to upload your video in the Youtube (private link) or any other platform and provide the link in the PDF document. The corresponding code should be uploaded as ZIP file on the CANVAS.

In this assignment, there are 6 (six) questions in total. Question 1 is about designing **Privacy Preserving System**. The question has two options. You need to answer any 1 out of two options. In the first option, you are required to design a **privacy-preserving revenue model** using the homomorphic property of the Exponential ElGamal cryptosystems. In the second option, you are expected to apply your understanding of privacy preserving computation in the context of electronic voting (E-Voting) based on the homomorphic property of Paillier Cryptosystems.

Question 2 is about the application of **Digital Signature Schemes**. In this question, you are expected to demonstrate your understanding of the RSA digital signature scheme and its security limitation. You are required to show step-by-step processes of how RSA digital signature can be forged. Marks will be deducted if you fail to show the detail computation correctly, skip the computation steps, or do not provide explanations.

Question 3 is about showing your understanding on **secure data hiding**. In this question, there are 2 (two) options: Q3(a) and Q3(b). You need to choose any 1 out of the two options. In question Q3(a), you are required to

implement an image steganography method based on **LSB Image steganography algorithm**. In question Q3(b), you are required to discuss embedding and extraction methods using **Text steganography algorithm on HTML source**. You should describe detail steps of the steganography with necessary screenshots.

Question 4 is related to the implementation of a secure file sharing system based on the concept of public-key cryptosystem-based encryption-decryption and digital signature. You should use **OpenSSL** and **IPFS commands** to show step by step processes to perform the secure file sharing tasks. you are expected to provide screenshots of the outcomes for commands. Marks will be deducted if you fail to show the commands correctly, skip any command, or do not provide screenshots.

Question 5 is on **report writing on Blockchain or implementation of a blockchain-based system**. **Only for this question, you can submit the solution individually or in a group. In the case of a group submission, the maximum group members can be 3 (three), and you must mention the names of group members in the solution of this question.** In this question, there are 2 (two) options: Q5(a) and Q5(b). You need to choose any 1 out of the two options. The first option Q5(a) is on report writing and the option, Q5(b) is about the implementation of a blockchain-based system. If you select Q5(a), you are expected to demonstrate your understanding of the Blockchain and write a well-organized report on a given topic. We are looking for interesting and innovative system design in the report. The report should be appended in the same document where you write the answers for other questions. Further instructions are given in the question. If you select Q5(b), you are expected to implement a blockchain system for a given scenario. You should describe detail steps of implementation with necessary code segments. Additionally, you need to provide a video demonstration and submit the code on the CANVAS. For a video demonstration, you are required to upload your video in the Youtube or any other platform and provide the link in the PDF document. The corresponding code should be uploaded as a ZIP file on the CANVAS.

Question 6 is related to **analyzing the security of authentication protocols**. Your answer should contain necessary explanation. Marks will be deducted if you fail to provide the explanation correctly for all of the protocols.

Develop this assignment in an iterative fashion (as opposed to completing it in one sitting). You should be able to start preparing your answers immediately after Lecture-5 (in Week-5). At the end of each week starting from Week-5 to Week-10, you should be able to solve at least one question.

If there are questions, you must ask via the relevant Canvas discussion forums in a general manner.

Overall, you must follow the special instructions below:

- You must fulfil the requirements in the questions.
- For the questions that require implementation, you must implement the functionalities stated in the questions. Any change in a user interface is acceptable if the functionality is there.
- In your solution, you must show all of the steps with necessary code segments and screenshots for each question.
- Upload your solution as a single PDF or Word document in CANVAS. Also, upload codes as a single ZIP file in the CANVAS.
- Do not put the PDF withing the ZIP file.

2. Assessment Criteria

This assessment will determine your ability to:

- Follow requirements provided in this document and in the lessons.
- Independently solve a problem by using cryptography and cryptanalysis concepts taught over the last six weeks from fifth to tenth weeks of the course.

- Meeting deadlines.

3. Learning Outcomes

This assessment is relevant to the following Learning Outcomes:

1. CLO 1: explain the functioning of security services in computing environments and the security issues in networked applications.
2. CLO 2: discuss various types of data integrity and confidentiality mechanisms including public key cryptography.
3. CLO 3: describe basic system security mechanisms and protocols, such as those used in operating systems, file systems and computer networks.
4. CLO 4: analyse the overarching importance of IT security in areas such as networking, databases, operating systems, and web systems.
5. CLO 5: apply privacy principles in basic practical settings in IT environments.
6. CLO 6: analyse and evaluate the security of computing and IT systems on a practical level and privacy related issues in computing.

4. Assessment details

Please ensure that you have read **Section 1 to 3** of this document before going further. Assessment details (i.e. question Q1 to Q6) are provided in the [next page](#).

Q1. Privacy Preserving Secure Models (Marks: 8)

You need to answer any 1 of the following questions:

(a). Privacy Preserving Revenue Model

Nowadays, many business organizations prefer outsourcing their business data (e.g. sales data) to cloud platforms. A cloud platform provides data storage and computation as services that are cost-effective for business organizations. For example, a business organization with multiple branches uses cloud platform for storing collected sales data from different branches and computing the sales revenue remotely. However, outsourcing sensitive sales data to the cloud introduces privacy risk for the business organization. Giving an example, the cloud service provider can collect sensitive sales data of business organization and reveal them to the competitor business organizations. In order to protect sensitive sales data from misuse, a privacy-preserving computation technique can be used. In a privacy-preserving computation technique, sales data can be encrypted at a branch before sending it to the cloud. The cloud can perform revenue calculations on encrypted sales data. The business owner can collect encrypted revenue from the cloud and decrypt to get the plaintext sales revenue.

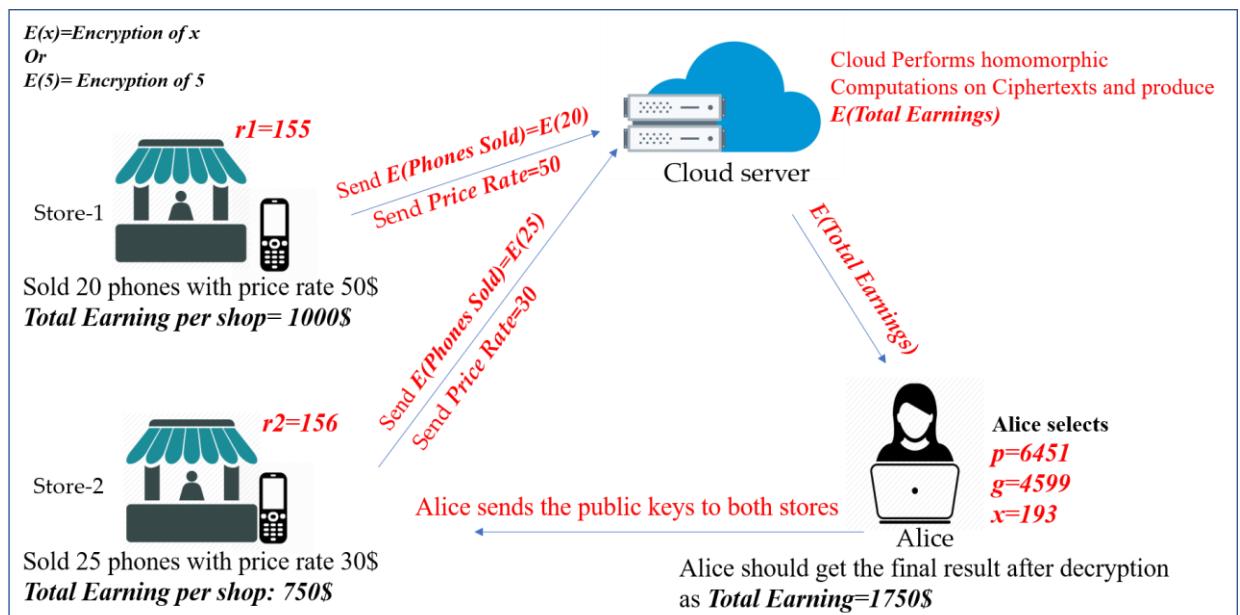


Figure-1.1: Privacy-preserving revenue model

The homomorphic property of the **Exponential ElGamal Cryptosystem** allows multiplication of encrypted numbers. Therefore, the Exponential ElGamal Cryptosystem can be used in developing a privacy-preserving revenue calculation application.

- In this task, you need to design a privacy-preserving revenue calculation application as shown in **Figure-1.1** using the **Exponential ElGamal Cryptosystem**. Suppose that Alice owns two different shops where she sells mobile phones of a specific brand. With the help of a cloud server, Alice wants to know how much she earned by selling the mobile phones in both shops remotely in a privacy-preserving manner. An example of sales information is shown in **Table-1**:

Your designed privacy-preserving revenue calculation application must fulfil the following requirements:

- The number of shops should be **two** and named as **Shop-1** and **Shop-2**.
- Alice generates public and private keys using the parameters: a prime number $p = 6451$, a generator g

= **4599**, and a private key $x = 193$.

- Each shop is considered as sender and **encrypts** the number of mobile phones sold.
- Each shop sends the price rate of mobile phones to cloud as **plaintext**.
- The cloud server receives encrypted numbers of mobile phones sold and the unit prices from different shops and computes the total revenue. Finally, the cloud server sends the encrypted revenue to the owner, Alice. Only Alice should be able to decrypt Total earning.

Table-1: Sales information of Shop-1 and Shop-2

Shops	Shop-1	Shop-2
<i>Phones sold</i>	20	25
<i>Price rate</i>	50	30
<i>Total Earning per shop</i>	1000	750
<i>Total Earning</i>	1750	

Show detail step-by-step computations of the **key generation**, **encryption**, **homomorphic computations** and **decryption** processes for the given details shown in **Table-1**.

[**Note:** Refer to the lecture-5 and tutorial-5.]

[\[If you are interested to implement a broader version of this system as the Capstone/Honours project, please contact the Lecturer\]](#)

(b). Privacy Preserving Online Voting System

Recently, several controversies have been observed in the voting around the world. Even electronic voting can be manipulated¹. In an electronic voting system, the voting authority cannot be trusted completely as it can be biased. Using privacy preserving online voting system removes controversy in voting system. In this privacy preserving online voting system, voters encrypt their votes in the voting booth before sending them to the voting authority. A voting server records each encrypted vote and determines the voting result on behalf of the voting booth as the voting booth does not have enough computation power. The encrypted result is sent to the voting authority who determines the winner based on encrypted votes. The homomorphic property of the **Paillier Cryptosystem** allows addition of encrypted numbers. Therefore, the Paillier Cryptosystem can be used in developing a privacy-preserving online voting application.

In this task, you need to design a privacy preserving online voting system as shown in **Figure-1.2** using the **Paillier cryptosystem**. Suppose that a group of students want to elect their club president.

Your designed privacy-preserving voting application must fulfil the following requirements:

- Votes must be **encrypted** from **Voting Booth** using **Paillier Cryptosystem** before sending them to the **Voting Server**.
- A vote is an integer number which should be equivalent to a **12-bit binary string**.
- The number of candidates should be three. For example, **ALICE**, **BOB**, and **EVE**.
- The number of voters can be **maximum 16**.
- Assume that **four voters** will vote for **ALICE**, **three voters** will vote for **BOB**, and **three voters** will vote for **EVE**. after counting the votes, the **Voting Authority (VA)** should find **four votes for Alice**, **three votes for BOB**, and **three votes for EVE**.
- The **Voting Authority** chooses $p=107$, $q=61$ and select $g=7019$.

- The private numbers chosen by **10 voters** and their votes are as follows:

Voter No.	Voter's Private Number, r	Vote for	Voting message, m
1	71	ALICE	00010000000 = 256
2	72	ALICE	00010000000 = 256
3	73	ALICE	00010000000 = 256
4	74	ALICE	00010000000 = 256
5	75	BOB	00000010000 = 16
6	76	BOB	00000010000 = 16
7	77	BOB	00000010000 = 16
8	78	EVE	00000000001 = 1
9	79	EVE	00000000001 = 1
10	80	EVE	00000000001 = 1

- The **Voting Authority** sets up required **public and private keys** and makes the public-keys to all **voting booth** before the vote starts.

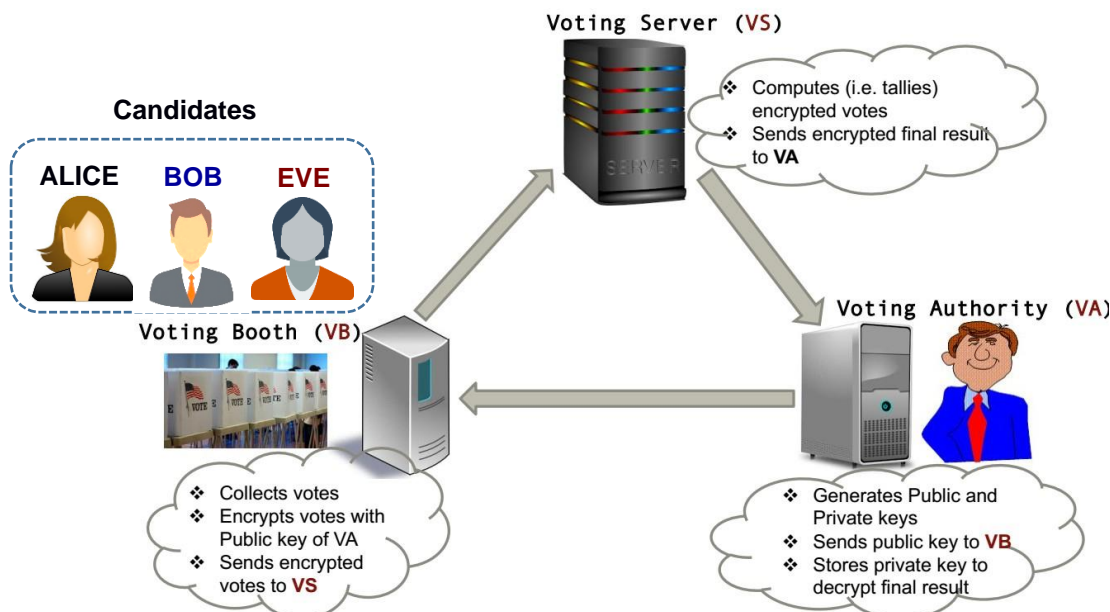


Figure-1.2: Privacy-preserving voting system

Show detail step-by-step computations of the **key generation**, **encryption**, **homomorphic computations**, and **decryption** processes for votes of **10 voters**.

[**Note:** Refer to the lecture-5 and tutorial-5.]

[1. <https://www.dw.com/en/democracy-in-danger-elections-are-easy-to-manipulate/a-45858161>]

[If you are interested to implement a broader version of this system as the Capstone/Honours project, please contact the Lecturer]

Q2. Forging Digital Signature (Marks: 4)

The working procedure of the digital signature is illustrated in **Figure-2.1**.

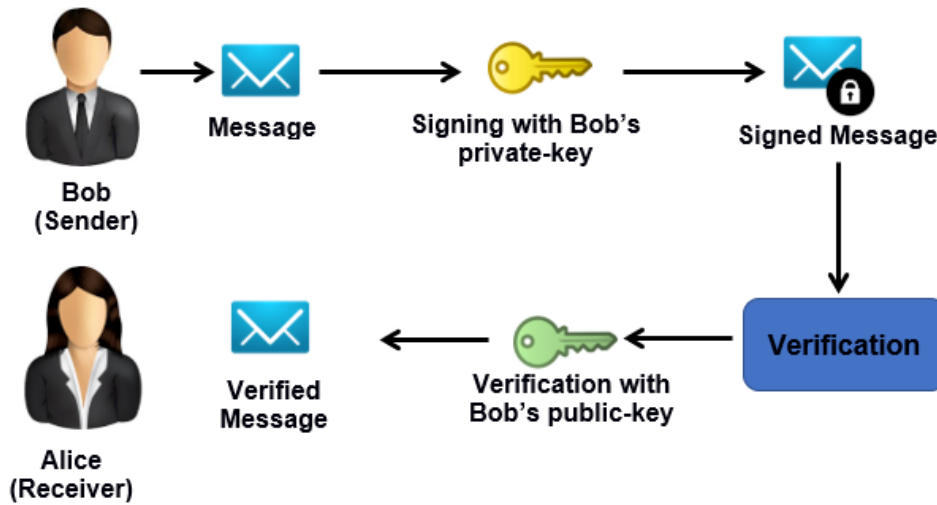


Figure-2.1: Overview of the Digital Signature

Recently, Charlie has repaired Alice's laptop. Alice creates a message for Bob as follows:

Hi Bob,
Please pay \$100 to Charlie.
Thanks,
Alice

Figure-2.2 (a): Message signed and sent by Alice to Bob

Alice creates the hash of the above message and signs the hash with her **RSA private key** and sends the message and signed hash value to Bob via email. Charlie is a smart tech person who compromises Alice's network and captures the email that is sent to Bob. Charlie modifies the content of the message, creates a hash of the message and generates a new signed hash value using Alice's signature. In other words, Charlie modifies the content of the message, performs a cryptanalysis to identify Alice's private key, and signs the hash of the message with the signature of Alice. Charlie sends the modified email to Bob. The **modified message** looks like the following:

Hi Bob,
Please pay \$200 to Charlie.
Thanks,
Alice

Figure-2.2 (b): Forged Message (modified message) created by Charlie and sent to Bob

Bob opens Alice's message in his email and finds the message in **Figure-2.2(b)**. Bob verifies Alice's digital signature using Alice's **RSA public key** and finds that the signature has been created by Alice. Hence, Bob trusts the message and pays \$200 to Charlie.

There are some assumptions as follows:

- Charlie knows that Alice used **RSA based digital signature** to sign the message.
- Charlie collects Alice's **RSA public key** (i.e., modulus n and public exponent e) and finds the equivalent decimal numbers as follows:

$n = 3585650403147635920968822644931356870369$

$e = 887$

- Charlie knows that Alice uses **MD5-Hash algorithm** generate the hash of the message before signing it.

In this task, you need to show how Charlie forges Alice's signature and change the payable amount. To do this:

- Show how to determine p and q from n and compute the private-key d using prime factorization method. Use online prime factorization calculators to find p and q .
- Show how **Charlie** would *hash* the forged or modified message as shown in **Figure-2.2(b)** and *sign* the hash using Alice's private-key. Also, show how **Bob** would verify the signature using Alice's public-key.
- To prevent this signature forgery will ElGamal Signature approach offer a better and robust solution? Justify why or why not.

[**Note:** Refer to the lecture-6 and tutorial-6.]

Q3. Hiding Secret Message in Images (Marks: 3)

You need to answer any 1 of the following question:

(a) Image Data Hiding

The XYZ Gallery preserves digital copies of different famous art works. Assume that Alice works as a painting expert for XYZ Gallery. Alice is sent two image files (see **Figure-3**) from two different sources that are the digital copies of the famous *Mona Lisa*, created by the Italian artist Leonardo da Vinci. As a painting expert, Alice's job is to identify the digital copy of the original *Mona Lisa* from the received image files. After the verification, Alice wants to put a hidden message "**Real Mona Lisa**" inside the **Figure-3(a)** and another message "**Fake Mona Lisa**" within the **Figure-3(b)**.



(a) mona_lisa_real.png



(b) mona_lisa_fake.png

Figure-3: Image files for LSB Image Steganography: (a) real image, (b) fake image

Assume that you are Alice's friend and know secure data hiding technique very well. So, she asks you to develop a program that will hide a secret message in an *image file* using secure data hiding technique.

Using **LSB Image steganography** technique, you need to develop the program that will take an image file as input, hide the corresponding secret message within the *image file*, and generate a **stego key (SK)**. The

program should also be able to extract the secret message from an image if the appropriate stego key (SK) is provided.

Use any programming language (ex: JAVA, Python, etc.) to perform this task. Upon completion of the implementation, you are expected to:

- I. Describe the implementation details and user instructions.
- II. Upload your code in the CANVAS.

[**Note:** Refer to the lecture-8 and tutorial-8.]

[\[If you are interested to implement a broader version of this system as the Capstone project, please contact the Lecturer\]](#)

(b) Text Data Hiding using HTML File

Alice writes articles and publishes on a popular website. Bad people copy texts of articles from the website and use them somewhere else without permission. To ensure the copyright of all articles, the website uses one of the text data hiding techniques for HTML source files. The website hides a secret code of each author in the HTML source of article page. It is assumed that the source code of a HTML file contains many color codes. The color code of a HTML source file is typically a 6 (six) character hexadecimal string. An example of the color code is given in **RED** color below:

` This is a funny text.
`

The website uses text data hiding methods to hide the bits of a secret code in the color codes of HTML source. Say, Alice has a secret code **101010**. The recent article page that Alice has authored has the following HTML source:

```
<body>
<font color=#000000> During World War II, invisible inks offered a <br/>
<font color=#000101> common form of invisible writing. With the <br/>
<font color=#000111> invisible ink, a seemingly innocent letter could <br/>
<font color=#00000d> contain a very different message written between <br/>
<font color=#000d0d> the lines. Therefore, the document text can conceal a <br/>
<font color=#000b1b> hidden message through the use of null ciphers <br/>
<font color=#0d0c01> (unencrypted message), which perfectly camouflage <br/>
<font color=#01010e> the real message in an ordinary letter. Open-coded <br/>
<font color=#000101> messages in which are plain text passages, but they <br/>
<font color=#000000> are shown in only ordinary occurrence. The suspect <br/>
<font color=#011001> communication can be detected by mail filters while <br/>
<font color=#000000> "innocent" messages are allowed to flow through. <br/>
<font color=#000011> There is an example on one of the most significant <br/>
<font color=#000000> null cipher messages sent by a Nazi spy: <br/>
<font color=#000000> "Apparently neutral's protest is thoroughly <br/>
<font color=#001101> discounted and ignored. Isman hard hit. Blockade <br/>
<font color=#000000> issue affects pretext for embargo on by-products, <br/>
<font color=#000111> ejecting suets and vegetable oils". <br/>
<font color=#000000> By extracting the second letter from each word, this <br/>
<font color=#0d0c01> hidden message can be decoded as: <br/>
<font color=#000000> "Pershing sails from NY June 1". <br/>
</body>
```

You are required to discuss the followings:

- i. Show how would the website hide bits of Alice's secret code (**101010**) within color codes of the above HTML source file that would cause minimum distortion. Provide the screenshot of the **Cover HTML page** after hiding the secret code within the HTML file.
- ii. What is the **stego-key (SK)** in this case?
- iii. Also, show how a verifier would retrieve the secret code from the HTML source using the **stego-key (SK)**.

Q4. Implementation of a Secure File Sharing System for Exam Papers (Marks: 6)

Implement a **Secure File Sharing System** for **RMIT University Computer Science Discipline** that will ensure sharing sensitive files among authorized users in a secure way.

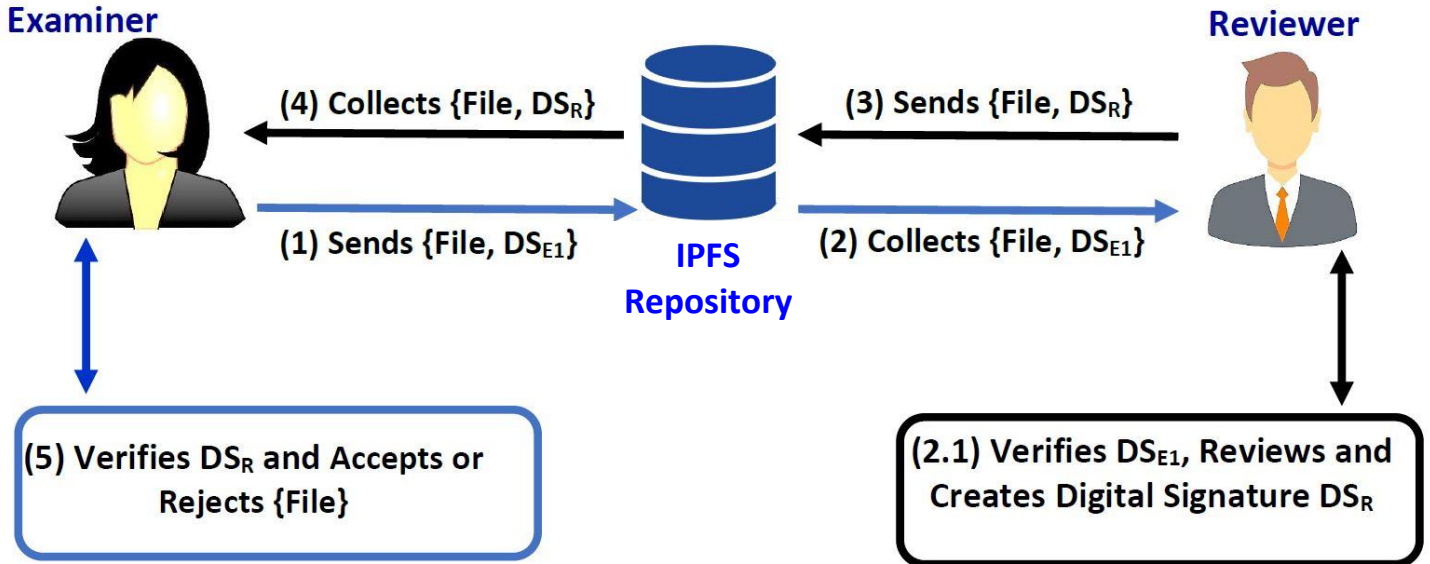


Figure-4: Overview of the file sharing system

The implemented Secure File Sharing System should fulfil the following requirements:

- The Secure File Sharing System uses **public-key cryptography systems** for secure communication.
- The Secure File Sharing System should have two participants: **examiner** and **reviewer**. Each participant creates their own public and private key pair before any communication and publish the public-key for other users. Assume that files are stored in the IPFS-based repository and communication between IPFS repository and any other participant is secure. However, the sender of a message that is sent to the repository needs to be verified.
- An examiner creates a file (say, an exam paper), signs the file using his/her private-key to create a digital signature for the file. Let us name this digital signature as **DS_{E1}**. Next, the examiner stores the file and **DS_{E1}** to the IPFS repository for a reviewer's approval.
- A reviewer downloads the file from the IPFS repository and verifies the signature of the examiner. If the signature is verified, the reviewer reviews the file for approval. For the sake of simplicity, we assume that the reviewer approves it. Next, the reviewer signs the file with his/her private-key and creates a digital signature for the file. Let's name this digital signature as **DS_R**. Finally, the reviewer sends the approved file, along with the digital signature (**DS_R**) of the reviewer, to the IPFS repository.
- After receiving the notification from the reviewer, the examiner collects the file and the corresponding digital signature of the reviewer (i.e., **DS_R**) for verifying the digital signature. If the verification is successful, the examiner uses the verified file for exam.

An overview of the system is illustrated in **Figure-4**.

You are required to show above steps using **OpenSSL** with proper commands and corresponding screenshots. Consider IPFS as the data repository and use IPFS "**add**" command to send file to the IPFS repository and "**get**" command to collect file from the IPFS repository.

[**Note:** Refer to the lecture-3, 4, and 6 and tutorial-3, 4, and 6.]

[If you are interested to implement a broader version of this system as the Capstone/Honours project, please contact the Lecturer]

Q5. Application of Blockchain Technology - Report Writing or Implementation (Marks: 20)

You need to answer **any 1** of the following question in a group of **maximum 3 (three)** people. However, it is absolutely fine if you want to do it individually.

(a) Report Writing on Application of Blockchain Technology

Write a report on **how the blockchain technology can be used to manage pandemic such as COVID19**. Please consider one or more of the followings (but not limited to) in your report:

- i. Explain how arrivals of passengers at the airport could be managed better with blockchain technology in a trustworthy and verifiable manner.
- ii. Explain how a comprehensive contact tracing can be realized where all state governments share data with each other.
- iii. Explain how blockchain can be effective for people in hotel quarantine. This should include people coming back to Australia from overseas trips, as well as workers serving residents in quarantine hotels. Also, consider people who are in quarantine but refusing covid-19 tests.
- iv. Explain how blockchain technology mitigate risks of doctors and nurses serving covid-19 patients at various hospitals.
- v. Explain how blockchain technology can track large gathering of people (e.g. 50 people being served at restaurants, 20 people attending religious services etc.) better to mitigate covid-19 risks and reduce community transmissions.

For readability of the report, and to make it self-contained, you may consider the following in your report:

- i. Briefly explain your understanding on the Blockchain technology and its usefulness in different applications.
- ii. Explain with necessary diagrams how your system can be designed using blockchain technology.
- iii. Explain how the **integrity** and **traceability** of data can be achieved using blockchain in your specified system.
- iv. Explain the advantages and disadvantages of using blockchain technology in your specified system.
- v. Briefly discuss how existing security and privacy preserving approaches can be adopted for managing sensitive data on the blockchain.

The report should be developed in a well-structured manner. You must provide necessary diagrams based on your own thoughts as well as collected from different sources. You must provide necessary references (at least 20) using **APA referencing style** including both research and online articles. Articles can be searched in:

- Google,
- Google Scholar (<https://scholar.google.com/>),
- IEEEXplore (<https://ieeexplore.ieee.org/Xplore/home.jsp>), etc.

Texts should be presented in IEEE Double-Column format with maximum 6 pages. For your convenience, a MS Word template for the report is provided in the Assignment-2 Home Page on CANVAS.

[Note: Refer to the lecture-7 and tutorial-7.]

(b) Implementation of a Blockchain-based Student Data Management System

Implement a **blockchain-based Student Data Management System** that will ensure integrity and traceability of student academic data.

The implemented blockchain-based Student Data Management System should fulfil the following requirements:

- The student data management system should have two types of users: **admin**, and **employer**
- Admins are responsible for adding or updating student records, while employer can see and verify student records. Say, student records include **Student Number, Student Name, Date of Birth (DOB), Degree Name, Graduation Year, and CGPA**.
- Student data should be sent to the data server. The data server cluster (set of P2P computers) is a collection of **three servers** and maintains the same copy of the student data. Each data server maintains a Merkle tree of hash values of data.
- Employers can verify if all of the data servers have the same copy of a particular student data. If all of the data servers have the same copy, the employer accepts the data. Otherwise, the data is rejected.

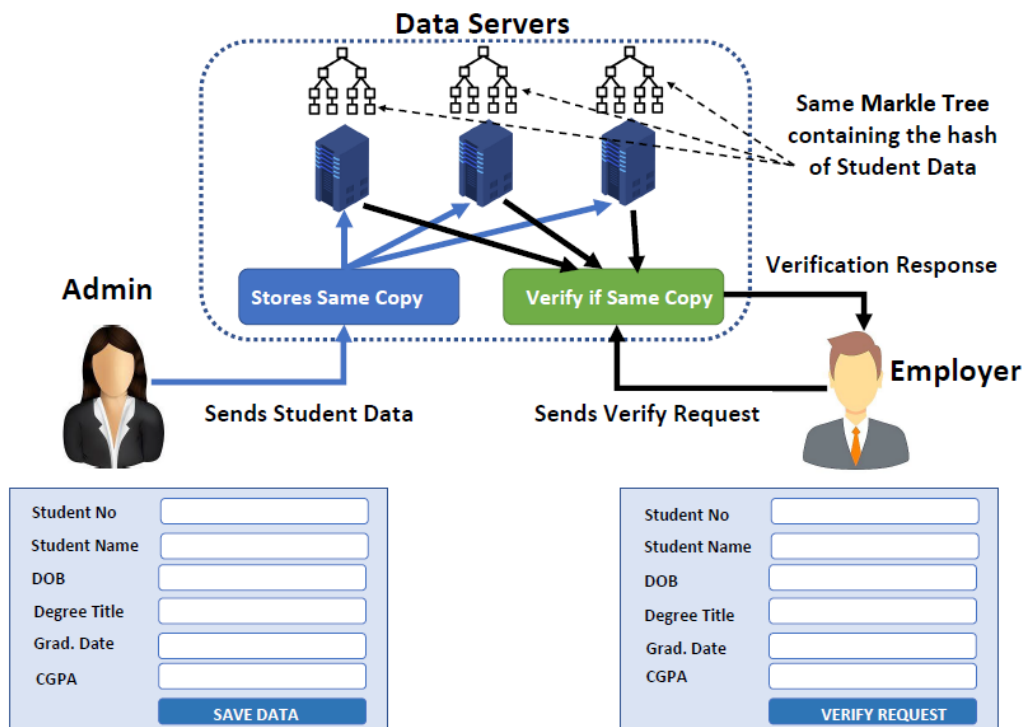


Figure-5: Overview of the file sharing system

An overview of the system is shown in Figure-5.

You are allowed to use any programming language or scripting language such as Java, PHP, Python, JavaScript, etc. A good **graphical user interface (GUI)** is expected. However, you can also provide user friendly **command-line user interface**. Upon completion of the implementation, you are expected to:

- I. Create a report containing the implementation details and user instructions.
- II. Upload your code and report in the CANVAS.

[**Note:** Refer to the lecture-8 and tutorial-8.]

[If you are interested to implement a broader version of this system as the Capstone/Honours project, please contact the Lecturer]

Q6. Designing a Secure Authentication Protocol for a One-to-One Secure Messaging Platform (Marks: 4)

WhatsApp, the Facebook-owned messaging platform, has been compromised by security issues recently and conversations of multiple celebrities, including world's richest man Zeff Bezos, have been leaked ([see the link at the end of this question](#)). Therefore, many celebrities are now looking for a one-to-one secure messaging platform.

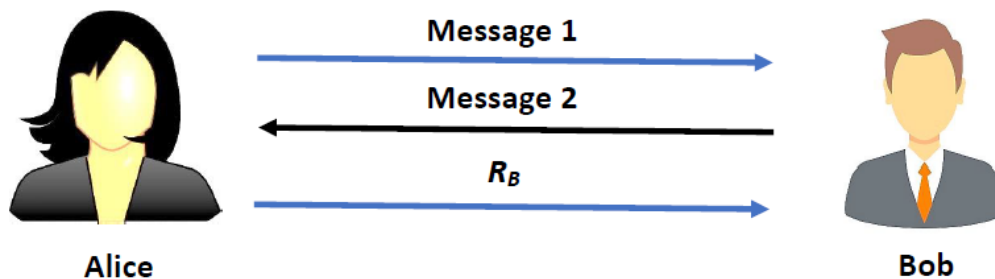


Figure-6: Overview of the secure mutual authentication and key establishment protocol

Assume that David is a software engineer in XYZ IT Company. He has been assigned a task to design a *secure mutual authentication and key establishment protocol* for a new messaging software. In the software, two users (ex: Alice and Bob) needs to exchange messages to achieve mutual authentication and establish a secure session key (K) before the start of the conversation as shown in **Figure-6**. According to the given scenario, Alice and Bob should exchange three messages to achieve mutual authentication and establish the secure session key (K). Assume that Alice is the *initiator* of the communication. Alice sends "**Message 1**" to Bob and Bob replies with "**Message 2**".

David is thinking several protocols and analyzing their security strength. The prospective security protocols are as follows:

- a) In protocol-1, **Message 1:** $E(\text{"Alice"}, K, R_A, K_{AB})$, **Message 2:** $R_A, E(R_B, K_{AB})$
- b) In protocol-2, **Message 1:** $\text{"Alice"}, E(K, R_A, K_{AB})$, **Message 2:** $R_A, E(R_B, K)$
- c) In protocol-3, **Message 1:** $\text{"Alice"}, E(K, R_A, K_{AB})$, **Message 2:** $R_A, E(R_B, K_{AB})$
- d) In protocol-4, **Message 1:** $\text{"Alice"}, R_A$, **Message 2:** $E(K, R_A, R_B, K_{AB})$

In this task, you need to critically analyze the above protocols and clearly explain which protocol or protocols would be secured and why. Notations are summarized below:

K	: Session key
R_A	: Nonce generated by Alice
R_B	: Nonce generated by Bob
K_{AB}	: Shared secret key between Alice and Bob
$E(\text{"Message"}, K_{AB})$: Symmetric Encryption Function that encrypts "Message" using K_{AB}

[**Note:** Refer to the lecture-9 and tutorial-9.]

Reference:

<https://www.forbes.com/sites/zakdoffman/2020/01/25/whatsapp-users-beware-this-stupidly-simple-new-hack-puts-you-at-riskheres-what-you-do/#3b1541a11d76>

5. Academic integrity and plagiarism (standard warning)

Academic integrity is about honest presentation of your academic work. It means acknowledging the work of others while developing your own insights, knowledge and ideas. You should take extreme care that you have:

- Acknowledged words, data, diagrams, models, frameworks and/or ideas of others you have quoted (i.e. directly copied), summarized, paraphrased, discussed or mentioned in your assessment through the appropriate referencing methods,
- Provided a reference list of the publication details so your reader can locate the source if necessary. This includes material taken from Internet sites.

If you do not acknowledge the sources of your material, you may be accused of plagiarism because you have passed off the work and ideas of another person without appropriate referencing, as if they were your own.

RMIT University treats plagiarism as a very serious offence constituting misconduct. Plagiarism covers a variety of inappropriate behaviors, including:

- Failure to properly document a source
- Copyright material from the internet or databases
- Collusion between students

For further information on our policies and procedures, please refer to the [University website](#).

6. Assessment declaration

When you submit work electronically, you agree to the [assessment declaration](#).

Rubric/assessment criteria for marking

All of the computations must be correct and only provided values must be used. Instructions must be followed.

Criteria The characteristic or outcome that is being judged.						Total
Question 1 Privacy Preserving Secure Models	Step-by-step processes are shown with detail computations. All of the computations shown are correct.	Step-by-step processes are shown with detail computations. Most of the computations are correct with few errors.	Step-by-step processes are shown with detail computations. Most of the computations are incorrect with few correct computations.	Step-by-step processes are not shown with detail computations. All of the calculations are wrong.	Not answered.	8 Marks
	8 Marks	6 Marks	4 Marks	2 Marks	0 Marks	
Question 2 Forging Digital Signature	Step-by-step processes are shown with required explanation. All of the computations are shown correctly in detail. Effectiveness of ElGamal Signature is well justified.	Step-by-step processes are shown with required explanation. Not all of the computations are shown correctly in detail. Effectiveness of ElGamal Signature is NOT well justified.	Steps that are shown partially correct and explanations are not up to the mark. Or, Steps are not shown correctly. Effectiveness of ElGamal Signature is NOT well justified or justification is NOT provided.	Steps that are shown are not correct. Or, The answer is incomplete.	Not answered.	4 Marks
	4 Marks	3 Marks	2 Marks	1 Marks	0 Marks	
Question 3 Hiding Secret Message	Steps of Data Hiding and extraction are described as per the requirements. Stego-Key is provided. Or, Steps of Implementation is described properly. Code works fine. Code is provided in the CANVAS.	Steps of Data Hiding and extraction are described as per the requirements. Stego-Key is not provided. Steps of Implementation is described properly. Code works fine. Code is not provided in the CANVAS.	Steps of either Data Hiding or extraction are described as per the requirements. Stego-Key is provided. Steps of Implementation is described properly. Code is provided in the CANVAS. But, code does not work.	Steps of either Data Hiding or extraction are described as per the requirements. Stego-Key is not provided. Steps of Implementation is not described properly. Code is provided in the CANVAS. But, code does not work. Or, code is not provided.	Not answered	3 Marks
	3 Marks	2 Marks	1.5 Marks	1 Marks	0 Marks	

Question 4	OpenSSL and IPFS Commands							6 Marks
Implementation of a Secure File Sharing System for Exam Papers	Answer is correct	Answer is correct but not structured	Answer is partially correct	Only few commands are correct	Answer is not correct			
	All of the commands are correctly and sequentially presented with appropriate screenshots	All of the commands are correct. But commands are not sequentially presented.	Some of the commands are correct. Commands are not sequentially presented. However, appropriate screenshots are provided for the correct commands.	Sequence of the commands are not followed Or some of the commands are missing Or screenshots are insufficient/missing	Or	Not answered		
	4 Marks.	3 Marks	2 Marks	1 Marks	0 Marks			
Question 5(a)	Report Writing							
Report Writing on Application of Blockchain Technology	The report is prepared fulfilling all of the requirements	The report is prepared fulfilling all of the requirements. However, could have been better.	The report is prepared fulfilling all of the requirements. However, the content is not enough to express the main theme of the given topic.	The report is NOT prepared fulfilling all of the requirements. The key topics are not well connected. Presentation is poor	The report addresses only few of the requirements. The key topics are missing or not connected. Presentation is poor.	None of the requirements are addressed correctly. The key concept is missing.	Not answered	20 Marks
	20 Marks	16 Marks	12 Marks	8 Marks	6 Marks	4 Marks	0 Marks	
Question 5(b)	Implementation							
Implementation of a Blockchain-based Student Data Management System	Implementation is described with proper screenshots.	Implementation is described with proper screenshots.	Implementation is described with proper screenshots. However, functionalities or user interface could have been better. Description is not provided with proper screenshots.	The implementation does not fulfil all the requirements. Description is not provided with proper screenshots.	The implementation does not fulfil all the requirements. Description is not provided with proper screenshots.	Only a few requirements are fulfilled. Descriptions and implementation is not adequate.	Not answered	
	Video demonstration is provided.	Video demonstration is provided but not up to the mark.	Video demonstration is provided but not up to the mark. Code is provided in the CANVAS.	Video demonstration is provided but not up to the mark.	Video demonstration is NOT provided.	None of the video demonstration and code is provided.		
	Code is provided in the CANVAS.	Code is provided in the CANVAS.		Code is provided in the CANVAS.	Code is provided			

	20 Marks	16 Marks	12 Marks	8 Marks	6 Marks	4 Marks	0 Marks	
Question 5 Designing a Secure Authentication Protocol for a One-to-One Secure Messaging Platform	Analysis on all of the scenarios in the given authentication protocols is presented clearly. 4 Marks	Analysis on 3 scenarios in the given authentication protocols is presented clearly. 3 Marks	Analysis on 2 scenarios in the given authentication protocol is presented clearly 2 Marks			Analysis on only 1 scenario in the given authentication protocol is presented clearly 1 Marks	Not answered 0 Marks	4 Marks