CRYPT ANALYSIS &CYBERDEFENCE
21CS3041RA, 21CS3041AA, 21CS3041PA

STUDENT ID:

STUDENT NAME:

ACADEMIC YEAR: 2023-24

Table of Contents

A.Y. 2023-24 LAB/SKILL CONTINUOUS EVALUATION

S.No	Date	Experiment Name	Pre-Lab (10M)	In-Lab (25M)			Post-Lab (10M)	Viva Voce (5M)	Total (50M)	Faculty Signature
				Program/ Procedure (5M)	Data and Results (10M)	Analysis & Inference (10M)				
1.		**Introductory Session**	**-NA-**							
2.		Implementation of Caesar Cipher and Vigenère Cipher.								
3.		Implementation of Playfair Cipher substitution technique								
4.		Implementation of Transposition and Columnar Techniques								
5.		Implementation of Simplified - Data Encryption Standard Algorithm								
6.		Implementation of AES Key Generation								
7.		Implementation of Substitute bytes and Shift rows operations in AES.								
8.		Implementation LCG and Blum-Blum Sub generators								
9.		Implementation of RSA Algorithm								
10.		Implementation of Diffie-Hellman Algorithm								
11.		Implementation a Two Simple Hash Functions								

S.No	Date	Experiment Name	Pre-Lab (10M)	In-Lab (25M)			Post-Lab (10M)	Viva Voce (5M)	Total (50M)	Faculty Signature
				Program/ Procedure (5M)	Data and Results (10M)	Analysis & Inference (10M)				
12		Implementation of SHA-512 Algorithm								
13.		Implementation of MD5 Algorithm								
14.		Implementation of One Time Pad and Hill Cipher substitution technique								
15.		Implementation of simplified RC4								
16.		Implementation of Elgamal Cryptosystem Algorithm								
17.		Implement IPsec Site-to-Site								
18.		Detecting different attacks using Wireshark								
19.		Mounting Forensic Images for Scanning and Recovering Files from Forensic Image								
20.		Demonstration on security mechanism incorporated in router								
21		Demonstration of security mechanism incorporated in switches.								

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
SUBJECT CODE: 21CS3041RA
CRYPT ANALYSIS AND CYBER DEFENCE WORKBOOK

1. Implementation of Caesar Cipher and Vigenère Cipher.

Date of the Session: ___/___/___ Time of the Session: _____to_____

Objective

- To understand the concept of Encryption and Decryption.
- To understand the applications of substitution techniques.
- To understand Symmetric Crypto System.

Distribution

The Caesar Cipher is a simple substitution cipher where each letter in the plaintext is shifted a certain number of places down the alphabet. For example, with a shift of 3, 'A' would be encrypted as 'D'. The Vigenère Cipher is an extension of the Caesar Cipher, where instead of using a single shift value, a keyword is used to determine different shift values for each letter. The keyword is repeated until it matches the length of the plaintext.

Pre-Requisites:

Experimental Setup:

- Programming Language: Python 3.9
- Integrated Development Environment (IDE): Jupyter Notebook
- Libraries Used: None

Pre-Lab Task:

1. Define Cryptography and write any two applications of cryptography.

2. What are the different types of Cryptographic Algorithms?

3. Mention the cryptographic algorithm used in Blockchain Technology and Gmail.

4. What is the need for encryption?

5. Write any two substation techniques ?

In-Lab Task:

1. Write a program to implement Caesar Cipher(encryption/decryption) for any given plain text.

(Hint: As a sample input student need to use his/her name as an input for implementing the program. Shift pattern to be taken as 4)

Experiment #		Student ID	
Date		Student Name	

2.Write a program to implement Vigenère Cipher(encryption/decryption) for any given plain text.

(Hint: As a sample input student need to use his/her name as an input for implementing the program. Key to be considered as your choice not exceeding the size 4)

Sol:

Experiment #		Student ID	
Date		Student Name	

Viva questions:

- Explain the concept of a Caesar Cipher and how it works.
- What is the key space of a Caesar Cipher?
- How can the Caesar Cipher be decrypted without knowing the key?
- Describe the Vigenère Cipher and its advantages over the Caesar Cipher.
- How is the Vigenère Cipher key used during encryption and decryption?

Post Lab Task:

1. Write a pseudo code for encryption and decryption using Caesar Cipher and Vigenère Cipher technique.

(For Evaluator's use only)

Comment of the Evaluator (if Any)	Evaluator's Observation
	Marks Secured: _____out of _____ Full Name of the Evaluator: Signature of the Evaluator Date of Evaluation

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
SUBJECT CODE: 21CS3041RA
CRYPT ANALYSIS AND CYBER DEFENCE WORKBOOK

2. Implementation of Playfair Cipher substitution technique

Date of the Session: ___/___/___ Time of the Session: _____to_____

Objective

- To understand the concept of multiple-letter encryption.
- To understand the applications of the technique.

Description:

The Playfair Cipher is a polygraphic substitution cipher that encrypts pairs of letters instead of single letters. It uses a 5x5 matrix (usually called a Playfair Square) containing a keyword to encrypt and decrypt the plaintext. Each letter is replaced by the letter in the same row and column of the pair it forms with the neighboring letter. If the letters are in the same row or column, they are replaced by the next letter in that row or column, forming a rectangle.

Pre requests :

Experimental Setup:

- Programming Language: Python 3.9
- Integrated Development Environment (IDE): Jupyter Notebook
- Libraries Used: None

Pre-Lab Task:

1. Define digram with an example.

2. What is the reason to consider a 5 × 5 matrix in a playfair cipher technique?

3. What to do if letters in plain text reoccur eg: *Hello*?

Experiment #		Student ID	
Date		Student Name	

4. What are the advantages of Playfair cipher?

5. Trace what will be the encrypted message by using Playfair cipher if the message is 'balloon' and the key is "Monarchy".

In-Lab Task:

1) Write a code to implement Playfair Cipher Substitution Technique for the following input:
Sample Input:-

Plain Text: "Student to consider his/her name"

Secret Key: REDHATCLUB

R	E	D	H	A
T	C	L	U	B
F	G	I/J	K	M
N	O	P	Q	S
V	W	X	Y	Z

*(**Note**: Ignore the whitespace and consider the text)*

Viva Questions:

- Briefly explain the Playfair Cipher and its basic principles.
- How are the Playfair Cipher's key matrix and key phrase related?
- What steps are involved in encrypting a message using the Playfair Cipher?
- How is the Playfair Cipher decrypted?
- Discuss any limitations or weaknesses of the Playfair Cipher.

Post-Lab Task:

1) Write a Pseudocode for Playfair Cipher Substitution technique.

Sol)

(For Evaluator's use only)

Comment of the Evaluator (if Any)	Evaluator's Observation
	Marks Secured: _____out of _____ Full Name of the Evaluator: Signature of the Evaluator Date of Evaluation

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
SUBJECT CODE: 21CS3041RA
CRYPTANALYSIS AND CYBER DEFENSE WORKBOOK

3. Transposition and Columnar Techniques

Date of the Session: ___/___/___ Time of the Session: _____to_____

Objective

- To understand the concept of Encryption and Decryption.
- To understand the applications of Transposition techniques
- To understand the applications of Columnartechniques

Description

Transposition Cipher is a type of encryption where the order of letters in the plaintext is rearranged. This can be done by writing the plaintext in a grid and then reading it column by column instead of row by row. Columnar Cipher is a specific type of transposition cipher where the plaintext is written in a grid of a certain number of columns, and then the columns are rearranged according to a keyword or another predetermined pattern.

Pre requests:

Experimental Setup:

Programming Language: Python 3.9

Integrated Development Environment (IDE): Jupyter Notebook

Libraries Used: None

Pre-Lab Task:

1. What is transposition cipher?

2. What are the applications of rail-fence cipher?

3. Brief description of columnar transposition cipher.

4. Columnar transposition cipher is also known as_____.

5. Difference between substation and transposition techniques?

In-Lab Task:

1. Write a program to implement Rail-Fence Cipher (encryption/decryption) for any given plain text.

Sample input: - N=6 Message=S21location56

Sample output: - Si2to1anac5to6l

Viva Questions :

- Describe the concept of transposition and columnar techniques in cryptography.
- How does the transposition technique reorder the plaintext to obtain the ciphertext?
- Explain the key generation process for the columnar technique.
- How is encryption performed using the columnar technique?
- Discuss any potential vulnerabilities or attacks against these techniques.

Post Lab Task:

1. Write a pseudo code for encryption and decryption using Rail Fence Cipher.

(For Evaluator's use only)

Comment of the Evaluator (if Any)	Evaluator's Observation Marks Secured: _____out of _____ Full Name of the Evaluator: Signature of the Evaluator Date of Evaluation

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
SUBJECT CODE: 21CS3041RA
CRYPTANALYSIS AND CYBER DEFENSE WORKBOOK

4. Simplified - Data Encryption Standard Algorithm

Date of the Session: ___/___/___ Time of the Session: _____to_____

Objective

- To understand symmetric key block cipher technique.
- To apply S-DES algorithm for key generation, Encryption and Decryption.

Description:

The Data Encryption Standard (DES) is a symmetric encryption algorithm. The simplified version of DES refers to a variant that simplifies some of the complex operations in the original DES algorithm. It involves the use of a key to encrypt and decrypt data in blocks of fixed size using a series of substitution and permutation operations.

Pre requests:

1. Programming Language: Choose a programming language of your preference to implement the DES algorithm. Some commonly used languages for cryptography include Python, Java, C++, or C#. Ensure that the language supports cryptographic libraries or modules for efficient implementation.

2. Cryptographic Libraries: Depending on the programming language you choose, you might need to use cryptographic libraries or modules that provide functions and methods for encryption, decryption, and other cryptographic operations. Examples include:

- Python: PyCryptodome, cryptography
- Java: Java Cryptography Architecture (JCA) or Bouncy Castle
- C++: Crypto++ or OpenSSL
- C#: .NET Framework's Cryptography API or Bouncy Castle

3. IDE or Text Editor: You'll need an integrated development environment (IDE) or a text editor to write and run your code. Some popular options are:

- Python: PyCharm, Visual Studio Code, or Jupyter Notebook
- Java: Eclipse, IntelliJ IDEA, or Visual Studio Code
- C++: Visual Studio, Code::Blocks, or CLion
- C#: Visual Studio, Visual Studio Code, or JetBrains Rider

4. Test Data: You'll need test data for encrypting and decrypting using the DES algorithm. Prepare a set of sample data that you can use to verify the correctness of your implementation.

5. DES Algorithm Specification: Obtain the DES algorithm specification, which includes the step-by-step instructions and rules for encryption and decryption. This specification will guide your implementation.

Pre-Lab Task:

1. Discuss on the size of Plain text, Cipher text and Key for general DES algorithm.

2. Apply S-DES algorithm and generate key followed by encryption and decryption.

Input:

Plain Text (8 - bits): Use the binary equivalent of the last Number of your register number.

Test case: If the last number is 0, then consider the predecessor non-zero value.

Initial Permutation: 2 6 3 1 4 8 5 7

Expanded Permutation: 4 1 2 3 2 3 4 1

$$S0 = \begin{matrix} 1 & 0 & 3 & 2 \\ 3 & 2 & 1 & 0 \\ 0 & 2 & 1 & 3 \\ 3 & 1 & 3 & 2 \end{matrix} \qquad S1 = \begin{matrix} 0 & 1 & 2 & 3 \\ 2 & 0 & 1 & 3 \\ 3 & 0 & 1 & 0 \\ 2 & 1 & 0 & 3 \end{matrix}$$

P_4: 2 4 3 1

Key (10 – bits) : 1 1 0 1 0 0 0 0 0 1

P_{10} order: *3 5 2 7 4 10 1 9 8 6*

P_8 order: *6 3 7 4 8 5 10 9*

In-Lab Task:

1.Consider the example that was solved in your Prelab, take those as the inputs and implement the algorithm using Java/Python.

Experiment #		Student ID	
Date		Student Name	

Viva Questions

- What is the purpose of the Data Encryption Standard (DES)?
- Explain the basic structure and key components of the DES algorithm.
- How does DES use a Feistel network in its encryption process?
- Describe the key generation process for DES.
- Discuss any known vulnerabilities or weaknesses of the DES algorithm.

Post Lab:

1. Write a Pseudocode to implement key generation using S-DES algorithm.

(For Evaluator's use only)

Comment of the Evaluator (if Any)	Evaluator's Observation
	Marks Secured: _____out of _____ Full Name of the Evaluator: Signature of the Evaluator Date of Evaluation

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
SUBJECT CODE: 21CS3041RA
CRYPT ANALYSIS AND CYBER DEFENCE WORKBOOK

5. Implementation of AES Key Generation.

Date of the Session: ___/___/___ Time of the Session: _____to_____

Objective

- You will understand the importance of generating a strong and unpredictable key for AES encryption.
- You will learn about the key length options for AES and their impact on security.
- AES supports three key lengths: 128-bit, 192-bit, and 256-bit. The longer the key length, the stronger the encryption, but it may also increase computational overhead.
- You will gain an understanding of the need for randomness in key generation.
- You may explore techniques for deriving a key from a passphrase or password.

Description

The Advanced Encryption Standard (AES) is a symmetric encryption algorithm widely used for securing sensitive data. AES key generation involves the process of generating a suitable encryption key for use with the AES algorithm. The key is generated according to specific rules and requirements defined by the AES standard.

Pre requests:

1. OpenSSL: OpenSSL is a widely used open-source cryptographic library that provides various functions, including AES key generation. You can use OpenSSL command-line tools or programming interfaces to generate AES keys.

2. CryptGenRandom (Windows): CryptGenRandom is a Windows API function that generates cryptographically secure random numbers. You can utilize this function in your programming language of choice (such as C/C++, C#, or PowerShell) to generate AES keys.

3. Cryptography libraries: Various programming languages offer cryptography libraries that include functions for AES key generation. For example, Python has libraries like `cryptography` and `pycryptodome` that provide methods for generating AES keys.

4. Hashcat: Hashcat is a popular password recovery and cracking tool that supports various encryption algorithms, including AES. While it is primarily used for password cracking, it can also generate AES keys.

5. Java Cryptography Architecture (JCA): If you are working with Java, you can utilize the Java Cryptography Architecture, which provides APIs for generating AES keys. The `javax.crypto.KeyGenerator` class can be used to generate AES keys of different sizes.

Pre-Lab Task:

1) How many keys are used in AES with respective to key size? And how many rounds are there in AES?

2) What is AES?

3) How is the key generated in AES?

4) What are the steps involved in AES key generation?

5) : What are the characteristics of a good AES key?

In Lab:

Write a program to implement the AES Key generation

Viva Questions:

- Explain the importance of key generation in the AES algorithm.
- How is the key expansion process performed in AES?
- Describe the different key sizes supported by AES and their implications.
- What is the role of the S-box in AES key generation?
- Discuss any security considerations or best practices regarding AES key generation.

Post Lab Task:

- Key Generation Techniques:
- Strength of AES Keys:
- Key Management and Storage:
- Key Generation Performance:

Comment of the Evaluator (if Any)	Evaluator's Observation Marks Secured: _____out of _____ Full Name of the Evaluator: Signature of the Evaluator Date of Evaluation

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
SUBJECT CODE: 21CS3041RA
CRYPT ANALYSIS AND CYBER DEFENCE WORKBOOK

6. Implementation of Substitute bytes and Shift rows operations in AES.

Date of the Session: ___/___/___ **Time of the Session: _____to_____**

Objective

- To use S-Box to perform a byte-by-byte substitution of the block.
- To understand shift row operation using permutation.

Decription:

Substitute Bytes and Shift Rows are two operations used in the AES algorithm as part of the encryption process. Substitute Bytes involves replacing each byte in a block of data with a corresponding byte from a substitution box. Shift Rows involves shifting the rows of the data block cyclically to the left by different offsets.

Pre request:

1. PyCryptodome: PyCryptodome is a powerful library that provides cryptographic functions, including AES, in Python. It offers a high-level interface for implementing AES operations. You can use the `AES` module from PyCryptodome to perform Substitute Bytes and Shift Rows operations.

2. OpenSSL: OpenSSL is a widely used open-source library that implements cryptographic functions, including AES. It provides a command-line tool called `openssl` that allows you to perform cryptographic operations. You can use the `openssl` command-line tool with appropriate options to perform Substitute Bytes and Shift Rows operations.

3. Java Cryptography Architecture (JCA): If you prefer to work with Java, you can utilize the Java Cryptography Architecture (JCA) to implement AES operations. The `javax.crypto` package in Java provides classes and interfaces for cryptographic operations. You can use the `Cipher` class from this package to perform AES encryption/decryption, including the Substitute Bytes and Shift Rows operations.

4. Cryptography.io: Cryptography.io is a Python library that provides a simple and easy-to-use API for various cryptographic operations, including AES. You can use the `cryptography . hazmat. primitives. ciphers` module from this library to perform AES operations. It allows you to implement Substitute Bytes and Shift Rows operations as part of your AES implementation.

Pre-Lab Task:

1) How many keys are used in AES with respective to key size? And how many rounds are there in AES?

2) Why AES is better than DES, Double DES, Triple DES?

3) How many rounds will take place if it 128bit, 192bit, and 256bit respectively?

4) Name the methods present in AES transformation function.

In-Lab Task:

1. Siri is a part of cryptanalysis team in an organization. The team is developing a complete application to decrypt a message using AES algorithm, the team lead has given 'Shift Rows Module' to Siri. To complete this module Siri asked you to write a program which performs shift rows operation in AES algorithm. Write the code by following the Input and Output Format given below:

Input Format: 4X4 matrix will be given as input

Si(0,0)	**Si(0,1)**	**Si(0,2)**	**Si(0,3)**
Si(1,0)	**Si(1,1)**	**Si(1,2)**	**Si(1,3)**
Si(2,0)	**Si(2,1)**	**Si(2,2)**	**Si(2,3)**
Si(3,0)	**Si(3,1)**	**Si(3,2)**	**Si(3,3)**

Output Format: 4x4 matrix after performing shift-row operations

So(0,0)	**So(0,1)**	**So(0,2)**	**So(0,3)**
So(1,1)	**So1,2)**	**So(1,3)**	**So(1,0)**
So(2,2)	**So(2,3)**	**So(2,0)**	**So(2,1)**
So(3,3)	**So(3,0)**	**So(3,1)**	**So(3,2)**

Experiment #		Student ID	
Date		Student Name	

Viva Questions :

- What is the purpose of the Substitute Bytes operation in AES?
- How does the Substitute Bytes operation achieve confusion in the AES encryption process?
- Explain the process of performing the Shift Rows operation in AES.
- How does the Shift Rows operation provide diffusion in the AES encryption process?
- Can you explain the importance of the Substitute Bytes and Shift Rows operations in AES in terms of security?

Post Lab:

Use the above table to perform Substitute Bytes Transformation operation in AES.

i)

fb	43	4d
8f	92	9d
ee	5f	97
6e	22	2a

(For Evaluator's use only)

Comment of the Evaluator (if Any)	Evaluator's Observation
	Marks Secured: _____out of _____ Full Name of the Evaluator: Signature of the Evaluator Date of Evaluation

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
SUBJECT CODE: 21CS3041RA
CRYPT ANALYSIS AND CYBER DEFENCE WORKBOOK

7. Implementation LCG and Blum-Blum Sub generators

Date of the Session: ___/___/___ Time of the Session: _____to_____

Objective

- To understand and implement Pseudo random number generation.
- To generate secure pseudorandom generators using Blum Blum Shub generator.

Description:

LCG (Linear Congruential Generator) is a type of pseudorandom number generator that generates a sequence of numbers based on a linear recurrence relation. Blum-Blum Shub is a cryptographically secure pseudorandom number generator based on the Blum Blum Shub algorithm. It utilizes modular exponentiation and the quadratic residue property

Pre request:

1. Python:

- You can use the built-in `random` module in Python for basic random number generation, including LCG.

- For implementing BBS, you can utilize the `pycryptodome` library, which provides cryptographic functionalities, including a BBS implementation.

2. Java:

- Java provides the `java.util.Random` class for basic random number generation, including LCG.

- For BBS implementation, you can use cryptographic libraries such as Bouncy Castle, which offers cryptographic algorithms and functions.

3. C/C++:

- C and C++ do not have built-in random number generation capabilities, but you can use the `rand()` function from the standard library for LCG.

- For BBS implementation, you can use cryptographic libraries like OpenSSL or Crypto++.

Experiment #		Student ID	
Date		Student Name	

Pre-Lab Task:

1.Why linear congruential pseudo random number generator shall not be used in cryptosystems?

2. How can we find the period of LCG?

3. Explain Blum Blum Shub sub generator in terms of security.

4. In Blum Blum Shub Generator, do we require to hide the prime factors p,q of the Modulus N?

In-Lab Task:

1. Rajesh is working in security domain of a company; his task is to secure the data using different cryptographic algorithm. Help Rajesh in generating an RC4 key using Blum Blum Shub generator.

Viva Questions:

- What is a Linear Congruential Generator (LCG), and how does it work?
- What are the key parameters involved in an LCG?
- Explain the concept of a Blum-Blum Shub generator and its working principle.
- How does the Blum-Blum Shub generator provide better security compared to an LCG?
- What are the potential applications of LCGs and Blum-Blum Shub generators in cryptography or random number generation?

Post Lab Task:

1) Ritish is working on Linear Congruential Generators to generate a series of random numbers for seed $X_0 = 1073$ and parameters a = 35, c = 528 and m = 2547 as given. Write the series of random numbers using linear congruential generators.

(For Evaluator's use only)

Comment of the Evaluator (if Any)	Evaluator's Observation
	Marks Secured: _____out of _____ Full Name of the Evaluator: Signature of the Evaluator Date of Evaluation

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
SUBJECT CODE: 21CS3041RA
CRYPT ANALYSIS AND CYBER DEFENCE WORKBOOK

8. Implementation of RSA Algorithm

Date of the Session: ___/___/___ Time of the Session: _____to_____

Objective

- To implement RSA algorithm.
- To understand encryption as a block.

Description :

The RSA (Rivest-Shamir-Adleman) algorithm is a widely used public-key encryption system. In this experiment, participants will implement the RSA algorithm from scratch. They will understand the mathematical concepts behind RSA, including modular arithmetic, prime number generation, and the Chinese Remainder Theorem. They will code the key generation process, encryption, and decryption algorithms. The participants will also experiment with different key sizes and analyze the trade-offs between security and computational efficiency.

Pre request:

Implementing RSA requires both programming skills and mathematical understanding. Here are some commonly used software tools and programming languages that can be used to complete the implementation of RSA:

1. Programming Languages:

- Python: Python is a popular choice for implementing RSA due to its simplicity and extensive library support. The `cryptography` library in Python provides built-in functions for RSA key generation, encryption, and decryption.
- Java: Java also has libraries, such as `javax.crypto`, that offer cryptographic functions, including RSA.

2. OpenSSL:

- OpenSSL is a widely-used open-source software library that provides cryptographic functions. It includes RSA key generation, encryption, and decryption functions. OpenSSL is available for various programming languages, including C, C++, Python, and Java.

3. Cryptography Libraries:

- There are several cryptographic libraries available that provide RSA implementations. Some commonly used ones include:

- Bouncy Castle: A comprehensive cryptography library for Java and C#, which includes RSA functionality.

- Crypto++: A C++ library that provides a wide range of cryptographic algorithms, including RSA.

- Cryptography.io: A Python library that offers a high-level interface for cryptographic operations, including RSA.

4. Integrated Development Environments (IDEs):

- IDEs can greatly assist in developing and testing RSA implementations. Some popular IDEs for different programming languages include:

- PyCharm: A Python IDE that offers advanced code editing, debugging, and testing features.

- Eclipse: An open-source IDE for Java development, which provides a range of plugins for cryptographic libraries.

- Visual Studio Code: A versatile code editor that supports multiple programming languages, including Python and Java.

Pre-Lab Task:

1.What is Asymmetric key? Explain Public and Private keys in RSA.

2.What is Eulers totient? Generate Eulers totient for the given numbers:

i) P=101, Q=173

3.Encrypt the following message using RSA Algorithm.

Message = CSE

In-Lab Task:

1. Write a program to implement RSA Algorithm for the following input.

Sample Input:

Note: you can use CACD as "03010304"

Experiment #		Student ID	
Date		Student Name	

Viva Questions:

- Describe the key generation process in RSA.
- How does the RSA encryption process work?
- What is the role of the Euler's totient function in RSA?
- Explain the RSA decryption process.
- Discuss the security strengths and limitations of RSA.

Post Lab Task:

1. University decided to make their communications more secure by better encryption technique.So according to trend RSA algorithm is one of the strongest encryption techniques and they finalized it.They want this work done by students, so to find students who are interested in it they gave a challenge to their students to encrypt a message "Hi" with RSA algorithm and submit it so they can find the best now.Its time for you to submit the implementation of encryptionusingRSA algorithm.

(For Evaluator's use only)

Comment of the Evaluator (if Any)	Evaluator's Observation
	Marks Secured: _____out of _____ Full Name of the Evaluator: Signature of the Evaluator Date of Evaluation

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
SUBJECT CODE: 21CS3041RA
CRYPT ANALYSIS AND CYBER DEFENCE WORKBOOK

9. Implementation of Diffie-Hellman Algorithm

Date of the Session: ___/___/___ Time of the Session: _____to_____

Objective

- To understand the key exchange process.
- To understand the purpose of Discrete Logarithms.

Description :

The Diffie-Hellman algorithm is a key exchange protocol that allows two parties to establish a shared secret key over an insecure channel. In this experiment, participants will implement the Diffie-Hellman algorithm. They will understand the mathematical principles behind the algorithm, including modular exponentiation and discrete logarithms. Participants will code the key generation process, the exchange of public keys, and the derivation of the shared secret key. They will verify that the shared key is the same for both parties.

Pre requests :

There are several software libraries and tools available for implementing the Diffie-Hellman algorithm. Here are some popular options:

1. OpenSSL: OpenSSL is a widely used open-source library that provides cryptographic functions, including support for the Diffie-Hellman key exchange. It is available for multiple programming languages, including C/C++ and Python.

2. Cryptography.io: Cryptography.io is a Python library that provides various cryptographic algorithms, including Diffie-Hellman. It offers an easy-to-use API for generating and exchanging Diffie-Hellman keys.

3. Bouncy Castle: Bouncy Castle is a comprehensive cryptography library available for Java and C#. It includes support for Diffie-Hellman key exchange and various other cryptographic algorithms.

4. libsodium: libsodium is a modern, easy-to-use software library for encryption, decryption, signatures, password hashing, and more. It provides a simple API for implementing Diffie-Hellman in several programming languages, including C/C++, Python, and JavaScript.

5. NaCl (Networking and Cryptography library): NaCl is a high-level cryptographic library that aims to be easy to use and resistant to misuse. It includes support for Diffie-Hellman key exchange and various other cryptographic operations. NaCl is available for several programming languages, including C/C++, Python, and JavaScript.

Pre-Lab Task:

1. What is the difference between RSA and Diffie Hellman?

2.What are the main properties of Diffie Hellman?

3.Explain Asymmetric key cryptography in few words and draw a neat diagram on its working.

4.Explain the concept of key exchange in the Diffie-Hellman algorithm.

5.How does the Diffie-Hellman algorithm enable secure communication over an insecure channel?

In-Lab Task:

1. You are trying to encrypt your messages you want to send to your friend because you don't want an outsider to know the confidential information you are sending to your friend so in order to do that use Diffie-Hellman Algorithm to encrypt the messages(choose an appropriate example) and encrypt the messages using Diffie-Hellman Algorithm.

Experiment #		Student ID	
Date		Student Name	

Viva Questions :

- Describe the key exchange process in the Diffie-Hellman algorithm.
- What is the role of the primitive root modulo in Diffie-Hellman?
- How does the Diffie-Hellman algorithm ensure secure key exchange over an insecure channel?
- Explain the process of computing the shared secret key in Diffie-Hellman.
- Discuss the potential vulnerabilities or attacks on the Diffie-Hellman algorithm.

Post Lab Task:

1. Suppose that two parties A and B wish to set up a common secret key (D-H key) between themselves using the Diffie Hellman key exchange technique. They agree on 7 as the modulus and 3 as the primitive root. Party A chooses 2 and party B chooses 5 as their respective secrets. Find the Diffie Hellman key.

(For Evaluator's use only)

Comment of the Evaluator (if Any)	Evaluator's Observation
	Marks Secured: _____out of _____ Full Name of the Evaluator: Signature of the Evaluator Date of Evaluation

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
SUBJECT CODE: 21CS3041RA
CRYPT ANALYSIS AND CYBER DEFENCE WORKBOOK

10. Implementation a two hash functions

Date of the Session: ___/___/___ Time of the Session: _____to_____

Objective

1. Understanding the concept of hashing
2. Apply the Basic hash function implementation
3. Apply the Hash distribution
4. Apply the Collision resolution
5. Apply the Efficiency considerations

Description :

In this experiment, participants will implement two simple hash functions. They will understand the basic principles of hash functions, such as input compression and the avalanche effect. Participants will code the hash functions using a programming language of their choice and test them with various inputs. They will evaluate the collision resistance and distribution properties of the hash functions and discuss their limitations.

Pre request:

There are several software tools and programming languages you can use to implement two hash functions. Here are a few commonly used options:

1. Python: Python is a versatile programming language with a rich set of libraries and modules that make it suitable for hash function implementation. You can use the built-in hashlib module to implement common hash functions like MD5, SHA-1, SHA-256, etc. Additionally, you can also implement custom hash functions using Python's standard library.

2. C/C++: C and C++ are low-level programming languages that provide greater control over memory and hardware. You can implement hash functions from scratch or use existing libraries like OpenSSL to perform hashing operations.

3. Java: Java provides built-in libraries such as java .security. Message Digest that support various hash functions like MD5, SHA-1, SHA-256, etc. You can use these libraries to implement hash functions in your Java application.

4. JavaScript: JavaScript has built-in support for hash functions through the Crypto API. You can use functions like `crypto.createHash` or `crypto.subtle.digest` to implement hash functions in the browser or Node.js environment.

5. Ruby: Ruby has libraries like Digest that provide implementations of various hash functions. You can use them to calculate hashes easily in your Ruby application.

6. Go: Go programming language offers the crypto package that includes hash functions. You can import and use functions like sha256.New() or md5.New() to implement hash functions in Go.

7. Rust: Rust is a systems programming language that prioritizes memory safety and performance. You can use libraries like Rust Crypto's `digest` crate to implement hash functions in Rust.

Pre-Lab Task:

Q: How does the simple addition hash function work?

Q: What are the advantages of the simple addition hash function?

Q: What are the limitations of the simple addition hash function?

Q: How does the simple XOR hash function work?

Q: What are the advantages of the simple XOR hash function?

Q: What are the limitations of the simple XOR hash function?

In-Lab Task:

1. Write a program to implement Simple Addition Hash Function.
2. Write a program to implement Simple XOR Hash Function.

Experiment #		Student ID	
Date		Student Name	

Viva Questions:

- What is a hash function, and what are its primary characteristics?
- Describe the implementation and working principles of the two simple hash functions.
- Discuss the collision resistance property of hash functions.
- How can the quality of a hash function be evaluated?
- Explain any potential weaknesses or limitations of the implemented hash functions.
- Sure! Here are some sample viva questions for the given experiments:

Post Lab Task:

Write the applications of sample Hash Functions

(For Evaluator's use only)

Comment of the Evaluator (if Any)	Evaluator's Observation
	Marks Secured: _____out of _____ Full Name of the Evaluator: Signature of the Evaluator Date of Evaluation

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
SUBJECT CODE: 21CS3041RA
CRYPT ANALYSIS AND CYBER DEFENCE WORKBOOK

11. Implementation a Simple Hash Algorithm(SHA-512)

Date of the Session: ___/___/___ Time of the Session: _____to_____

Objective

- To understand the importance of Hash function for secure data transmission.
- To analyse the difference between SHA and other traditional security algorithms.

Description :

SHA-512 (Secure Hash Algorithm 512-bit) is a widely used cryptographic hash function. In this experiment, participants will implement the SHA-512 algorithm. They will understand the internal workings of the algorithm, including message padding, message expansion, and the compression function. Participants will code the SHA-512 algorithm and verify its correctness by comparing the output with existing implementations. They will also discuss the security properties of SHA-512, such as pre-image resistance and collision resistance.

Pre requests:

To implement a simple hash algorithm like SHA-512, you can use various programming languages and libraries. Here are a few examples of software tools you can use:

1. Python:

- hashlib module: The hashlib module in Python provides various hash algorithms, including SHA-512. You can import the module and use the `sha512()` function to compute the hash.

2. Java:

- Java Cryptography Architecture (JCA): The JCA provides a set of cryptographic APIs in Java. You can use the `MessageDigest` class to compute the SHA-512 hash.

3. C/C++:

- OpenSSL library: OpenSSL is a widely used open-source library that provides cryptographic functions. It includes an implementation of SHA-512.

Pre-Lab Task:

1) What is SHA Algorithm?

2) What is the hash value of SHA-1?

3) Write any 3 differences between SHA-1 and SHA-256?

Q4) What is SHA256 hash function?

Q5) - Can you explain the steps involved in the SHA-512 algorithm?

In-Lab Task:

1. Kiran is doing an internship in Cyber Security. As part of his research, he is assigned a task to demonstrate the working of SHA algorithm in Computer programming language.

Experiment #		Student ID	
Date		Student Name	

Viva Questions:

- What is the purpose of the SHA-512 algorithm?
- How does the SHA-512 algorithm differ from other hash algorithms?
- How is the integrity of data ensured using SHA-512?
- What are some applications of SHA-512 in cryptography and security?
- How does SHA-512 ensure data integrity and message authentication?

Post Lab Task:

1. As Raj groups is using SHA1, SHA256 Hash functions. Help Raj friend Ravi to implement different Hash Functions (SHA-512, MD5)

(For Evaluator's use only)

Comment of the Evaluator (if Any)	Evaluator's Observation
	Marks Secured: _____out of _____ Full Name of the Evaluator: Signature of the Evaluator Date of Evaluation

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
SUBJECT CODE: 21CS3041RA
CRYPT ANALYSIS AND CYBER DEFENCE WORKBOOK

12. Implementation a MD5 Algorithm

Date of the Session: ___/___/___ Time of the Session: _____to_____

Objective

1. Understanding the basics of hash functions
2. Analyze of Hash function properties
3. Understanding the Vulnerabilities and limitations
4. Understanding the Historical context

Description :

MD5 (Message Digest Algorithm 5) is a widely used cryptographic hash function, although it is considered insecure for many applications. In this experiment, participants will implement the MD5 algorithm. They will understand the step-by-step process of the algorithm, including message padding, message segmentation, and the hash computation. Participants will code the MD5 algorithm and test it with different inputs. They will discuss the vulnerabilities of MD5, such as collision attacks and the existence of pre-image and second pre-image attacks.

Pre request:

There are several programming languages and libraries that you can use to implement the MD5 algorithm. Here are a few examples:

1. Python:
   - hashlib library: Python's hashlib module provides a convenient way to calculate the MD5 hash. You can use the `md5()` function to generate the hash.
2. Java:
   - java.security.MessageDigest: The MessageDigest class in Java provides functionality for various hashing algorithms, including MD5.
3. C++:
   - OpenSSL library: OpenSSL provides cryptographic functions, including the MD5 hash algorithm.

These are just a few examples of how you can implement the MD5 algorithm using different programming languages and libraries. Remember to choose a language and library that best fits your needs and project requirements.

Pre-Lab Task:

Q: What is MD5?

Q: How does MD5 work?

Q: What are the advantages of MD5?

Q: What are the limitations of MD5?

Q: How can MD5 be used in practice?

Q: What alternatives to MD5 exist?

In-Lab Task:

1. Write a program to implement MD5 Algorithm.

Viva Questions:

- What is the MD5 algorithm used for?
- What are the key properties of the MD5 algorithm?
- How does the MD5 algorithm generate a hash value?
- Can you explain the concept of collision resistance in the MD5 algorithm?
- What are the limitations or vulnerabilities of the MD5 algorithm?

Post-Lab Task:

- Hash Collision Demonstration:
- Implement a Hash Verification System:
- Analyze Hash Function Performance:

(For Evaluator's use only)

Comment of the Evaluator (if Any)	Evaluator's Observation
	Marks Secured: _____out of _____ Full Name of the Evaluator: Signature of the Evaluator Date of Evaluation

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
SUBJECT CODE: 21CS3041RA
CRYPT ANALYSIS AND CYBER DEFENCE WORKBOOK

13. Implementation of One Time Pad and Hill Cipher substitution technique (A/P)

Date of the Session: ___/___/___ Time of the Session: _____to_____

Objective: The purpose of this endeavor is to evaluate the efficacy of the One Time Pad and Hill Cipher in protecting sensitive data.

Description: The One Time Pad is a symmetric encryption technique that provides perfect secrecy when employed properly. The experiment entails generating a random key of the same length as the plaintext message and XORing it with the plaintext to produce the ciphertext. In order to recover the original message, decryption requires XORing the ciphertext with the same key. The experiment will involve the development of a program or algorithm for performing encryption and decryption with the One Time Pad.

Hill Cipher is a polygraphic substitution algorithm that operates on plaintext letter blocks. The experiment entails generating a matrix key and multiplying it with the plaintext to produce the ciphertext. The process of decryption entails multiplying the ciphertext matrix by the key matrix's inverse. The endeavor will involve the creation of a program or algorithm to implement and decrypt the Hill Cipher.

Pre request:

To implement the One Time Pad and Hill Cipher substitution techniques, you can use various programming languages and cryptographic libraries. Here are a few examples of software/tools that you can use:

1. Python: Python is a popular programming language with extensive cryptographic libraries that can be used for implementing both the One Time Pad and Hill Cipher.

- PyCryptodome: It is a collection of cryptographic algorithms and protocols, including support for One Time Pad and Hill Cipher. You can use the library to perform encryption and decryption operations.
- pycipher: This library provides implementations for various classical ciphers, including the Hill Cipher. It offers a straightforward interface for encrypting and decrypting messages using the Hill Cipher technique.

2. Java: Java also provides cryptographic libraries that can be used to implement the One Time Pad and Hill Cipher.

- Bouncy Castle: It is a widely used cryptographic library in Java that supports a variety of algorithms, including those used for One Time Pad and Hill Cipher. It provides APIs for encryption and decryption operations.

3. MATLAB: MATLAB is a powerful mathematical computing software that can be used for implementing both the One Time Pad and Hill Cipher.

- MATLAB Cryptography Toolbox: This toolbox provides functions for cryptographic operations, including the implementation of classical ciphers like the One Time Pad and Hill Cipher.
- MATLAB built-in functions: MATLAB also offers built-in functions for matrix operations, which can be utilized to implement the Hill Cipher encryption and decryption processes.

4. C/C++: If you prefer working with C/C++, you can use cryptographic libraries to implement the One Time Pad and Hill Cipher.

- OpenSSL: It is a widely used library for secure communications and cryptography. You can utilize OpenSSL's functions for encryption and decryption operations.
- Crypto++: This is a popular C++ cryptographic library that provides a wide range of cryptographic algorithms, including the Hill Cipher. It offers a simple API for implementing encryption and decryption using the Hill Cipher technique.

Pre-Lab Task:

1. Hill Cipher is a block cipher. Justify.

2. If $A=\begin{matrix}4 & 5\\ 2 & 7\end{matrix}$, find |A|.

3. Write the mathematical formula for encryption and decryption in Hill Cipher.

4. If $A = \begin{matrix} 1 & 3 & 1 \\ 3 & 2 & 5 \\ 2 & 2 & 2 \end{matrix}$, find A^{-1}.

5. Can we consider the matrix $\begin{matrix} 6 & 6 \\ 6 & 6 \end{matrix}$ as a key matrix in Hill Cipher. Justify.

In-Lab Task:

Q.1) Write a program to implement Hill Cipher Substitution technique for the following input.

Sample Input:

Plain Text: *Student to consider His/Her name*

Key: *ALBO*

Note: White space in the plaintext can be ignored and the key matrix must be 2×2 matrix.

Filer character to be taken as 'x'.

Sol)

Experiment #		Student ID	
Date		Student Name	

2) Write a program to implement one time padding

Viva Questions :

- What is the concept behind the One Time Pad encryption technique?
- How does the One Time Pad ensure perfect secrecy?
- Explain the working principle of the Hill Cipher substitution technique.
- What are the advantages and limitations of the Hill Cipher substitution technique?
- Can you discuss any real-world applications where these techniques can be used?

Post-Lab Task:

1. Write a Pseudocode to find the inverse of a 3×3 matrix.

(For Evaluator's use only)

Comment of the Evaluator (if Any)	Evaluator's Observation
	Marks Secured: _____out of _____ Full Name of the Evaluator: Signature of the Evaluator Date of Evaluation

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
SUBJECT CODE: 21CS3041RA
CRYPT ANALYSIS AND CYBER DEFENCE WORKBOOK

14. Implementation of simplified RC4 (A/P)

Date of the Session: ___/___/___ Time of the Session: _____to_____

Objective: This experiment's objective is to implement a streamlined version of the RC4 stream cipher algorithm and evaluate its encryption and decryption capabilities.

Description: RC4 is an extensively employed symmetric stream cipher that is renowned for its simplicity and speed. The experiment entails creating a streamlined version of the RC4 algorithm, including the critical scheduling and pseudo-random generation algorithms. Based on a given key, the simplified RC4 implementation should generate a stream of pseudo-random bits. The experiment will concentrate on comprehending the RC4 algorithm's fundamental principles and developing a functional implementation.

The experiment will consist of evaluating the encryption and decryption efficacy of the simplified RC4 implementation using various plaintext and key combinations. Throughput, security, and resistance to known assaults will be measured as part of the evaluation. The results will illuminate the applicability and limitations of simplified RC4 as a cryptographic algorithm.

Pre request:

1. Python: Python is a popular programming language that is often used for cryptographic implementations. You can use libraries like `pycryptodome` or `cryptography` to implement RC4 in Python.
2. C/C++: C and C++ are commonly used programming languages for low-level implementations. You can write your own RC4 algorithm in these languages or use existing libraries like OpenSSL or Crypto++.
3. Java: Java is another widely used programming language that provides good support for cryptographic operations. The `javax.crypto` package in Java includes classes for implementing various encryption algorithms, including RC4.
4. OpenSSL: OpenSSL is a widely used open-source library that provides cryptographic functions and protocols. It includes support for RC4, and you can use it in various programming languages like C/C++, Python, or Ruby.
5. Ruby: Ruby is a dynamic, object-oriented programming language that has libraries like `openssl` that provide cryptographic functionalities. You can use the library to implement RC4 in Ruby.

Pre-Lab Task:

1. What are the pros and cons of RC4?

2. Is RC4 better than AES? Justify your answer.

3. Write short notes on stream generation.

4. . What is RC4 and what is its purpose in cryptography?

5. Who developed the RC4 algorithm and when was it introduced?

In-Lab Task:

1. After the second apocalypse on earth no one survived except 5 who went to space in a spaceship: Clarke , Bellamy , Revan , Murphy and Echo . After 20 years of the apocalypse they surprisingly got an encrypted text message from earth,no one understood the message.Clarke expected that the message is encrypted by RC4 method and asked Revan to decrypt the message. So , write a python program to decrypt the RC4 encrypted text by giving key and text as user input.

Message from earth : " **AEB19F4906E2717ADA765DDC5D21F336C84F98** "

Key: **Can be considered as your choice.**

Experiment #		Student ID	
Date		Student Name	

Viva Questions :

- What is the RC4 algorithm used for?
- Explain the key generation process in the RC4 algorithm.
- How does the RC4 algorithm ensure confidentiality?
- What are some vulnerabilities or weaknesses of the simplified RC4 algorithm?
- Are there any improvements or modifications made to address the vulnerabilities?

Post Lab Task:

1. Write a Pseudo code for the encryption and decryption of RC4.

(For Evaluator's use only)

Comment of the Evaluator (if Any)	Evaluator's Observation
	Marks Secured: _____out of _____ Full Name of the Evaluator: Signature of the Evaluator Date of Evaluation

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
SUBJECT CODE: 21CS3041RA
CRYPT ANALYSIS AND CYBER DEFENCE WORKBOOK

15. Implementation of Elgamal Cryptosystem Algorithm (A/P)

Date of the Session: ___/___/___ Time of the Session: _____to_____

Objective: The objective of this experiment is to implement the Elgamal cryptosystem, a public-key encryption algorithm, and evaluate its security and efficiency in protecting sensitive information.

Description: Elgamal is based on the Diffie-Hellman key exchange protocol and provides secure communication between two participants using public-private key pairs. The experiment entails implementing the Elgamal algorithm's key generation, encryption, and decryption processes. Key generation requires the selection of suitable prime numbers and the calculation of the required parameters. Encryption involves converting plaintext into ciphertext using the recipient's public key, whereas decryption utilizes the recipient's private key to recover the plaintext message.

In addition to evaluating the efficacy of the Elgamal algorithm in terms of computational complexity and encryption/decryption speed, the experiment will also assess its computational complexity and encryption/decryption speed. In addition, the security of the Elgamal cryptosystem will be evaluated in light of probable attacks, including brute-force, chosen-plaintext, and known-plaintext attacks. The findings will shed light on the advantages and disadvantages of the Elgamal algorithm in practical cryptographic applications.

Pre request:

1. Programming Languages:
   - Python: A versatile language with several cryptographic libraries.
   - Java: Provides built-in support for cryptographic operations.
   - C/C++: Offers low-level control and high-performance capabilities.

2. Cryptographic Libraries:
   - Cryptography (Python): A powerful library that provides various cryptographic algorithms, including ElGamal.
   - Bouncy Castle (Java/C#): A comprehensive library offering cryptographic algorithms and protocols.
   - OpenSSL (C/C++): A widely used library that provides a range of cryptographic functions.

3. Mathematical Libraries:

- GMP (GNU Multiple Precision Arithmetic Library): A library for high-precision arithmetic operations, useful for large number computations involved in ElGamal.
- BigInteger (Java): A class in Java's standard library for handling arbitrary precision integers.

4. Integrated Development Environments (IDEs):

- PyCharm (Python): A popular IDE for Python development.
- Eclipse (Java): An IDE widely used for Java development.
- Visual Studio (C/C++): A powerful IDE for C/C++ development.

Remember that implementing cryptographic algorithms is a complex task, and it's crucial to follow best practices and guidelines to ensure security. It is advisable to consult cryptographic experts and references while implementing such algorithms. Additionally, make sure to thoroughly test your implementation and consider using well-established libraries whenever possible to avoid common pitfalls and vulnerabilities.

Pre Lab

Q: What is the ElGamal cryptosystem?

Q: How does the ElGamal cryptosystem work?

Q: What are the advantages of the ElGamal cryptosystem?

Q: What are the limitations of the ElGamal cryptosystem?

Q: How can the security of the ElGamal cryptosystem be enhanced?

Q: In what scenarios is the ElGamal cryptosystem commonly used?

Q What are the mathematical principles behind the Elgamal cryptosystem?

In Lab:

Write a program to implement the Elgamal Cryptosystem Algorithm

Experiment #		Student ID	
Date		Student Name	

Viva Questions :

- What is the Elgamal cryptosystem used for?
- Explain the key generation process in the Elgamal cryptosystem.
- How does the Elgamal algorithm achieve secure communication?
- Can you discuss the mathematical principles behind the Elgamal cryptosystem?
- What are the advantages and limitations of the Elgamal cryptosystem?

Post Lab Task:

Write a short notes for the following

- Algorithm Overview and Analysis:
- Key Exchange Protocol:
- Security Analysis and Enhancements:

(For Evaluator's use only)

Comment of the Evaluator (if Any)	Evaluator's Observation
	Marks Secured: _____out of _____ Full Name of the Evaluator: Signature of the Evaluator Date of Evaluation

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
SUBJECT CODE: 21CS3041RA
CRYPT ANALYSIS AND CYBER DEFENCE WORKBOOK

16. Implement IPsec Site-to-Site (A/P)

Date of the Session: ___/___/___ Time of the Session: _____to_____

Objective :

This experiment aims to implement IPsec (Internet Protocol Security) in a site-to-site VPN (Virtual Private Network) configuration and evaluate its efficacy at securing communication between two network sites.

Description :

IPsec is a protocol suite that provides IP communications with authentication, integrity, and privacy. In a site-to-site VPN configuration, IPsec assures secure communication over an untrusted network, such as the internet, between two distinct networks. Configuring IPsec on network hardware (routers or firewalls) at each site, establishing the requisite security associations, and setting up a secure connection comprise the experiment.

Pre request:

1. VPN Gateway/Firewall: A network device or software that acts as the endpoint for the IPsec tunnel. It handles the encryption and decryption of traffic and manages the VPN connections. Popular VPN gateways/firewalls include Cisco ASA, Juniper SRX, Palo Alto Networks, and Check Point.

2. IPsec VPN Client: If you are connecting to a remote site as a client, you will need an IPsec VPN client software installed on your device. Some popular IPsec VPN clients are Cisco AnyConnect, Palo Alto GlobalProtect, and FortiClient.

3. IPsec Implementation: IPsec is a protocol suite used for securing IP communications. Most modern operating systems have built-in IPsec support. Here are some examples:

- Windows: Windows operating systems have built-in IPsec support through the Windows Firewall with Advanced Security. You can configure IPsec policies and rules using the "Windows Firewall with Advanced Security" console.
- Linux: Linux distributions often use strongSwan or OpenSwan as IPsec implementations. These tools provide the necessary components for configuring and managing IPsec tunnels.
- macOS: macOS has built-in IPsec support through the "Network" settings. You can configure IPsec tunnels by creating VPN configurations in the network settings.

4. Key Management: IPsec requires the use of encryption keys for secure communication. Key management protocols like Internet Key Exchange (IKE) are used to establish and manage the keys. Most IPsec implementations have built-in support for key management.

5. Network Monitoring and Troubleshooting Tools: When implementing IPsec Site-to-Site, it is essential to have network monitoring and troubleshooting tools to diagnose issues and monitor the VPN tunnels. Tools like Wireshark, tcpdump, and various network monitoring platforms can help in capturing and analyzing IPsec traffic.

Pre Lab:

1. What is IPsec, and what is its purpose in a site-to-site VPN?

2. How does IPsec provide secure communication between two sites?

3. What are the different components of IPsec?

4. Explain the two main modes of IPsec operation.

5. What is the difference between transport mode and tunnel mode in IPsec?

6. Describe the process of establishing an IPsec tunnel between two sites.

7. What are the necessary parameters for configuring IPsec site-to-site VPN?

8. Explain the concept of Security Associations (SAs) in IPsec.

9. How do encryption and authentication algorithms play a role in IPsec?

In Lab:

Experiment #		Student ID	
Date		Student Name	

Viva Questions:

- What is IPsec and what is its purpose?
- Explain the concept of a site-to-site VPN using IPsec.
- What are the components required to establish an IPsec site-to-site connection?
- Can you describe the process of securing the communication using IPsec?
- Are there any common challenges or considerations when implementing IPsec site-to-site?

Post-Lab Task:

- Identify Routers properties used in the experiment.
- Identify Routers Cables used in the experiment.

(For Evaluator's use only)

Comment of the Evaluator (if Any)	Evaluator's Observation
	Marks Secured: _____out of _____ Full Name of the Evaluator: Signature of the Evaluator Date of Evaluation

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
SUBJECT CODE: 21CS3041RA
CRYPT ANALYSIS AND CYBER DEFENCE WORKBOOK

17. Detecting different attacks using Wireshark (A/P)

Date of the Session: ___/___/___ Time of the Session: _____to_____

Objective :

The purpose of this experiment is to use Wireshark, a network protocol analyzer, to detect and investigate different forms of network attacks. The experiment seeks to identify and comprehend the signatures and patterns associated with various attacks, such as denial-of-service (DoS) attacks, port scanning, packet sniffing, and network intrusions, by capturing and analyzing network traffic.

Description :

Wireshark must be installed on a monitoring computer in order to capture network packets in this experiment. Various assault scenarios, including known attack vectors and techniques, will be simulated. Wireshark's filtering and analysis features will be used to identify the attack patterns and characteristics from the intercepted packets. By analyzing packet headers, payloads, and protocols, participants will gain insight into the network vulnerabilities exploited by a variety of attacks and the countermeasures that can be implemented to mitigate these threats.

Pre request:

Wireshark is a powerful network analysis tool that can be used to detect and analyze various attacks on a network. While Wireshark itself does not have built-in detection capabilities, it provides a rich set of features for capturing and inspecting network traffic, which can be used to identify potential security threats. Here are some common attacks and techniques you can employ with Wireshark to detect them:

1. Malware Infections:

- Look for unusual traffic patterns, such as a high volume of suspicious outgoing connections or repeated download attempts.
- Analyze HTTP traffic for malicious file downloads or suspicious URLs.
- Identify abnormal DNS requests or responses that may indicate command-and-control (C2) communication.

2. Denial of Service (DoS) Attacks:

- Monitor network traffic for a sudden increase in traffic volume or specific protocols (e.g., ICMP flood).
- Identify a large number of incomplete TCP connections or SYN flood attacks.
- Look for abnormal patterns in network behavior, such as an excessive number of connection resets.

3. Network Scanning and Reconnaissance:

- Analyze traffic for a series of sequential port scans or repeated failed login attempts.
- Look for ARP scanning activities, which involve a high volume of ARP requests and replies.
- Monitor for suspicious ICMP echo requests (ping) targeted at multiple hosts.

4. Man-in-the-Middle (MitM) Attacks:

- Identify ARP spoofing or poisoning attacks by analyzing ARP traffic and detecting inconsistencies.
- Look for SSL/TLS handshake failures or certificate errors that may indicate a MitM attempt.
- Monitor for abnormal MAC or IP address changes associated with network devices.

5. Password Sniffing:

- Inspect network traffic for unencrypted protocols (e.g., FTP, Telnet) that transmit credentials in plain text.
- Look for password-related keywords or patterns in captured packets.
- Analyze DNS traffic for requests to known malicious domains associated with credential theft.

Pre Lab:

1. What is Wireshark and how does it work?

2. How can Wireshark be used to detect different types of attacks?

3. What are some common types of attacks that can be detected using Wireshark?

4. Explain the process of capturing network traffic with Wireshark.

5. What are some key features of Wireshark that aid in detecting attacks?

6. How can you filter and analyze captured packets in Wireshark to identify attacks?

7. Describe the steps involved in detecting a DDoS attack using Wireshark.

8. How can you identify a network intrusion using Wireshark?

In Lab:

Write a procedure to implement Detecting different attacks using Wireshark.

Experiment #		Student ID	
Date		Student Name	

Viva Questions

- How does Wireshark help in detecting network attacks?
- Explain some of the common types of attacks that can be detected using Wireshark.
- What are the steps involved in capturing and analyzing network traffic with Wireshark?
- Can you demonstrate how to identify a specific attack using Wireshark?
- What are the limitations or challenges of using Wireshark for attack detection?

Post-Lab Task:

What are things identified During the analysis, several attacks using Wireshark.

(For Evaluator's use only)

Comment of the Evaluator (if Any)	Evaluator's Observation
	Marks Secured: _____out of _____ Full Name of the Evaluator: Signature of the Evaluator Date of Evaluation

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
SUBJECT CODE: 21CS3041RA
CRYPT ANALYSIS AND CYBER DEFENCE WORKBOOK

18. Mounting Forensic Images for Scanning and Recovering Files from Forensic Image(A/P)

Date of the Session: ___/___/___ Time of the Session: _____to_____

Objective:

This experiment's purpose is to demonstrate the mounting of forensic images and file recovery. Participants will learn how to access, investigate, and extract valuable information from forensic images, which are exact copies of digital storage media, without altering the original data.

Description:

In this investigation, each participant will receive a forensic image file, typically in Advanced Forensic Format (AFF) or Raw format. Participants will learn how to mount these forensic images and construct a virtual representation of the storage media using specialized forensic tools, such as The Sleuth Kit or Access Data FTK Imager. Once mounted, they will investigate the file system structure and employ scanning techniques to look for specific files, deleted content, or concealed data. Participants will also practice recovering files from the forensic image, recognizing the significance of maintaining data integrity and ensuring that their findings are properly documented.

Pre request:

There are several software tools available for mounting forensic images and scanning/recovering files from them. Here are some commonly used ones:

1. FTK Imager: FTK Imager, developed by AccessData, is a popular forensic imaging and analysis tool. It allows you to create forensic images, mount forensic image files, and scan/recover files from them. FTK Imager supports various image formats, including EnCase (E01), Advanced Forensic Format (AFF), and raw (dd).

2. EnCase Forensic: EnCase Forensic, developed by Guidance Software (now OpenText), is a comprehensive forensic investigation tool. It provides features for imaging, evidence collection, and analysis. EnCase Forensic supports the mounting of forensic images and facilitates the scanning and recovery of files.

3. X-Ways Forensics: X-Ways Forensics is a powerful forensic software tool that supports imaging, analysis, and file recovery. It allows you to mount forensic images and examine their contents. X-Ways Forensics offers advanced searching, sorting, and filtering capabilities to aid in the recovery of files.

4. Autopsy: Autopsy, also known as The Sleuth Kit, is an open-source digital forensic platform. It provides a graphical interface for analyzing and recovering data from various image formats. Autopsy supports the mounting of forensic images and includes file system analysis and keyword searching features.

5. ProDiscover Forensic: ProDiscover Forensic is a commercial forensic software tool that offers imaging, analysis, and file recovery capabilities. It allows you to mount forensic images and browse their contents. ProDiscover Forensic provides advanced search options and can recover files based on various criteria.

6. AccessData FTK (Forensic Toolkit): FTK, developed by AccessData, is a comprehensive forensic analysis tool. It includes features for imaging, analysis, and file recovery. FTK supports the mounting of forensic images, enabling you to explore and recover files from the mounted images.

7. Oxygen Forensic Detective: Oxygen Forensic Detective is a forensic analysis tool designed for mobile devices. It supports the mounting of mobile device images, such as physical dumps or logical extractions, allowing you to scan and recover files from these images.

Pre Lab:

1. What is a forensic image, and why is it important in digital forensics?

2. Explain the process of creating a forensic image from a storage device.

3. What are the different formats of forensic images commonly used in digital forensics?

4. What is the purpose of mounting a forensic image?

5. What are the potential risks or challenges involved in mounting a forensic image?

6. Describe the steps involved in mounting a forensic image for scanning and file recovery.

7. How can you determine the file system of a mounted forensic image?

8. What tools or software are commonly used to mount forensic images?

Experiment #		Student ID	
Date		Student Name	

In Lab :

Write a Procedure to Mounting Forensic Images for Scanning and Recovering Files from Forensic Image

Experiment #		Student ID	
Date		Student Name	

Viva Questions:

- What is the purpose of mounting forensic images in digital forensics?
- Explain the process of mounting a forensic image.
- How can scanning and recovering files be performed on a mounted forensic image?
- What are some precautions or best practices to follow when working with forensic images?
- Can you discuss any specific tools or techniques used in mounting forensic images?

Post-Lab Task:

1. what is Forensic Image
2. Properties of Virtual Machine
3. what are the Uses of Mounting Software

(For Evaluator's use only)

Comment of the Evaluator (if Any)	Evaluator's Observation
	Marks Secured: _____out of _____ Full Name of the Evaluator: Signature of the Evaluator Date of Evaluation

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
SUBJECT CODE: 21CS3041RA
CRYPT ANALYSIS AND CYBER DEFENCE WORKBOOK

19. Demonstration on security mechanism incorporated in router

Date of the Session: ___/___/___ Time of the Session: _____to_____

Objective:

The purpose of this demonstration is to demonstrate how a router's security mechanisms protect and secure network traffic. Access control lists (ACLs), firewalls, virtual private network (VPN) support, and intrusion detection and prevention systems (IDPS) are just a few of the features and configurations available in routers that contribute to network security.

Description:

A router with sophisticated security capabilities will be used as a demonstration platform in this experiment. Participants will investigate the graphical user interface (GUI) or command-line interface (CLI) of the router in order to access and configure its security features. Participants will learn how to configure ACLs to control traffic flow, firewall rules to filter and monitor network traffic, VPN support to establish secure connections, and IDPS to detect and prevent potential attacks. Through this hands-on demonstration, participants will comprehend the significance of routers in securing networks and preventing unauthorized access and malicious behavior.

Pre request:

There are several software tools available that can help you demonstrate the security mechanisms incorporated in a router. Here are a few examples:

1. Wireshark: Wireshark is a widely-used network protocol analyzer that allows you to capture and analyze network traffic. It can help you demonstrate how the router handles various network protocols and identify any potential security vulnerabilities.

2. Nessus: Nessus is a vulnerability scanning tool that can scan the router's configuration and identify any known security vulnerabilities. It can help you demonstrate the importance of keeping the router's firmware up to date and configuring it securely.

3. Metasploit: Metasploit is a penetration testing framework that can be used to simulate real-world attacks on the router. By exploiting known vulnerabilities, you can demonstrate the potential consequences of inadequate security measures and the importance of implementing strong defenses.

4. Nmap: Nmap is a network scanning tool that can be used to discover hosts, open ports, and services running on the router. It can help you demonstrate the need for proper firewall configuration and port management to prevent unauthorized access.

5. Router-specific tools: Many router manufacturers provide their own software tools for managing and configuring their routers. These tools often include security features such as firewall settings, access control lists, and VPN configurations. Using the manufacturer's tools can help you demonstrate the specific security mechanisms incorporated in their routers.

Pre Lab:

1. What is the purpose of a security mechanism in a router?

2. Explain the concept of firewall and its role in router security.

3. What are some common types of attacks that routers face, and how does the security mechanism in a router help protect against them?

4. How does Network Address Translation (NAT) contribute to router security?

5. What is port forwarding, and how can it be used securely in a router?

6 Explain the difference between stateful and stateless packet filtering, and how they are implemented in router security.

7. Discuss the importance of firmware updates for router security and how they help mitigate vulnerabilities.

8. What is VPN (Virtual Private Network), and how can it be configured on a router to enhance security?

In Lab :

Write a Procedure to Demonstration on security mechanism incorporated in router.

Viva Questions:

- What are the common security mechanisms incorporated in routers?
- Explain the purpose and functionality of each security mechanism.
- Can you demonstrate the configuration or setup of a specific security mechanism in a router?
- What are some potential risks or vulnerabilities that routers may face despite these security mechanisms?
- How can these security mechanisms be enhanced or customized based on specific network requirements?

Experiment #		Student ID	
Date		Student Name	

Post-Lab Task:

1.Explain various Security Mechanisms Explored in this experiment.

(For Evaluator's use only)

Comment of the Evaluator (if Any)	Evaluator's Observation
	Marks Secured: _____out of _____ Full Name of the Evaluator: Signature of the Evaluator Date of Evaluation

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
SUBJECT CODE: 21CS3041RA
CRYPT ANALYSIS AND CYBER DEFENCE WORKBOOK

20. Demonstration of security mechanism incorporated in switches.

Date of the Session: ___/___/___ Time of the Session: _____to_____

Objective:

The purpose of this experiment is to demonstrate the security mechanisms built into switches, which protect network infrastructure and assure secure device communication. Participants will acquire an understanding of the security features offered by switches, including port security, VLAN segmentation, and MAC filtering.

Description:

A switch with sophisticated security capabilities will be utilized in this experiment to demonstrate its security mechanisms. To configure security settings, participants will access the switch's administration interface via a graphical user interface or command line interface. They will discover how to enable port security, which limits the number and types of devices that can connect to a switch port. VLAN segmentation will be demonstrated in order to segregate network traffic, thereby preventing unauthorized access and enhancing network security. Participants will also investigate MAC address filtering, a technique that permits or disallows devices based on their unique MAC addresses.

Pre request:

1. Wireshark: Wireshark is a widely used network protocol analyzer that allows you to capture and analyze network traffic. It can help you monitor and inspect the security features implemented in switches, such as VLANs, port security, access control lists (ACLs), and more.

2. Nmap: Nmap (Network Mapper) is a powerful network scanning and security auditing tool. It can be used to discover and assess the security of switches by performing port scanning, OS fingerprinting, and vulnerability scanning.

3. Cisco Packet Tracer: Cisco Packet Tracer is a network simulation and visualization tool specifically designed for Cisco network equipment. It allows you to create virtual network topologies and configure switches with various security features. You can then observe the behavior of these mechanisms in action.

4. GNS3: GNS3 (Graphical Network Simulator-3) is another network simulation tool that enables you to build and emulate complex network topologies. It supports virtual switches from different vendors, allowing you to configure and test security features on a variety of switches.

5. OpenVAS: OpenVAS (Open Vulnerability Assessment System) is an open-source vulnerability scanner. It can be used to assess the security of switches by scanning for known vulnerabilities, misconfigurations, and weaknesses in the switch's firmware or software.

6. Nessus: Nessus is a widely used vulnerability scanner that can help you identify security flaws in switches. It provides comprehensive vulnerability assessment, including checks for common misconfigurations, weak passwords, and outdated firmware.

Pre Lab:

1. What is the purpose of security mechanisms in switches?

2. What are some common security threats that switches can mitigate?

3. Explain the concept of VLANs (Virtual Local Area Networks) and how they contribute to switch security.

4. How does port security work in switches? What are the main features and benefits?

5. Describe the process of implementing access control lists (ACLs) on switches and their significance in securing network traffic.

6. What is MAC address filtering, and how does it enhance switch security?

7. Explain the concept of Secure Shell (SSH) and its role in securing switch management access.

8. How does the Spanning Tree Protocol (STP) contribute to switch security?

9. Discuss the importance of firmware updates and patch management in maintaining switch security.

In Lab:

Write a procedure to Demonstration of security mechanism incorporated in switches.

Experiment #		Student ID	
Date		Student Name	

Experiment #		Student ID	
Date		Student Name	

Viva Questions:

1. What are the primary security mechanisms incorporated in switches?

2. Can you explain the concept of VLAN (Virtual Local Area Network) and its role in network security?

3. How does Port Security work in switches and what are its benefits?

4. Can you demonstrate the configuration of Access Control Lists (ACLs) on a switch for enhanced security?

5. Explain the purpose and functionality of Spanning Tree Protocol (STP) in ensuring network security.

6. What is DHCP Snooping and how does it prevent unauthorized DHCP server attacks?

7. Can you demonstrate the setup of Dynamic ARP Inspection (DAI) on a switch to mitigate ARP spoofing attacks?

8. Discuss the concept of Private VLANs (PVLANs) and their significance in network isolation and security.

Post-Lab Task:

1.Explain various Security Mechanisms incorporated in switches.

(For Evaluator's use only)

Comment of the Evaluator (if Any)	Evaluator's Observation
	Marks Secured: _____out of _____ Full Name of the Evaluator: Signature of the Evaluator Date of Evaluation