Centurion UNIVERSITY
Shaping Lives... Empowering Communities...

School: .... Campus: ....

Academic Year: .... Subject Name: .... Subject Code: ....

Semester: .... Program: .... Branch: .... Specialization: ....

Date: ....

Applied and Action Learning

(Learning by Doing and Discovery)

Name of the Experement : Security First – Understanding Blockchain Attacks

Objective/Aim:

To study the common vulnerabilities and attack vectors in blockchain technology, understand how these attacks are performed, and identify preventive methods to improve blockchain security and network reliability.

Apparatus/Software Used:

- **Remix IDE** – for writing and testing Solidity smart contracts
- **Ganache** – for setting up a local blockchain environment
- **MetaMask** – to interact with decentralized applications
- **Ethereum or Bitcoin Test Network** – for testing attack simulations
- **Blockchain Explorer** – to inspect and analyze transaction behavior

Theory/Concept:

Although blockchain provides a strong security foundation through cryptography and decentralization, it still faces certain weaknesses. Attackers may exploit flaws in smart contracts, consensus protocols, or communication networks to gain control or manipulate data.

A blockchain's safety primarily depends on three aspects:

- **Consensus Security:** Keeps all network nodes synchronized and ensures only valid transactions are confirmed.
- **Cryptographic Integrity:** Ensures data confidentiality and prevents alteration or forgery.
- **Network Protection:** Maintains communication between nodes and prevents external interference.

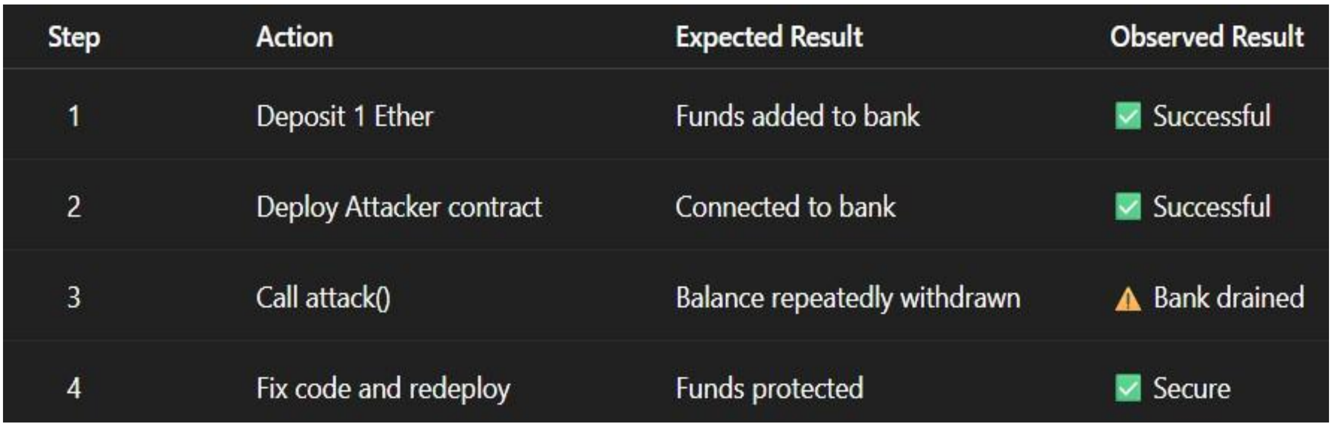
Major Types of Blockchain Attacks:

1. **51% Attack** – When a single entity gains majority control over mining or staking power.
2. **Sybil Attack** – Multiple fake identities are used to manipulate network operations.
3. **Double-Spending Attack** – The same cryptocurrency is spent more than once.
4. **Eclipse Attack** – Isolating a node from the network to feed it false data.
5. **Smart Contract Exploits** – Bugs or logic errors in contract code allow unauthorized fund access.
6. **Routing Attack** – Targeting communication channels between blockchain nodes.

Procedure:

- **Develop a Weak Smart Contract:**
  Create a Solidity contract with an unprotected withdraw function that transfers funds before updating balances.
- **Deploy the Contract:**
  Use Remix IDE to compile and deploy the vulnerable contract on Ganache. Connect MetaMask for transaction execution.
- **Design an Attack Contract:**
  Write another contract that calls the vulnerable function repeatedly, exploiting the timing flaw before the balance is updated.
- **Execute the Exploit:**
  Invoke the attack function and observe the repeated withdrawals from the victim contract.
- **Monitor and Analyze:**
  Use the blockchain explorer or Remix transaction logs to trace the attack pattern and check balance changes.
- . **Patch the Vulnerability:**
  Modify the contract using safe patterns like **Checks-Effects-Interactions** or **ReentrancyGuard**, then redeploy it to confirm the issue is fixed.

Observation Table:



Step	Action	Expected Result	Observed Result
1	Deposit 1 Ether	Funds added to bank	✅ Successful
2	Deploy Attacker contract	Connected to bank	✅ Successful
3	Call attack()	Balance repeatedly withdrawn	⚠ Bank drained
4	Fix code and redeploy	Funds protected	✅ Secure

ASSESSMENT

Rubrics	Full Mark	Marks Obtained	Remarks
Concept	10		
Planning and Execution/ Practical Simulation/ Programming	10		
Interpretation Result and	10		
Record of Applied and Action Learning	10		
Viva	10		
Total	**50**		

Signature of the Student:

Name :

Regn. No.

Signature of the Faculty: