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 Experiment: Security First – Understanding Blockchain Attacks

*Theory :

Blockchain is a decentralized ledger system, but it can be targeted by various attacks that exploit its consensus and communication mechanisms. Understanding these attack vectors helps strengthen blockchain architecture.

Common Blockchain Attacks:

- 51% Attack – When a single entity controls more than half of the network's mining or validation power.
- Double Spending – Spending the same digital token twice due to transaction race conditions.
- Sybil Attack – Creating multiple fake identities to disrupt consensus or influence governance.
- Replay Attack – Reusing valid transactions across different chains after forks.
- Smart Contract Exploits – Vulnerabilities in code such as reentrancy or integer overflow.
- Eclipse Attack – Isolating a node from the main network to manipulate its view of the blockchain.

*Coding Phase: Pseudo Code / Flow Chart / Algorithm

1. Start
2. Select a blockchain attack simulation environment (e.g., SimBlock, Crypto51.app, Remix IDE)
3. Choose an attack type to simulate
4. Configure simulation parameters (nodes, hash power, block rate, etc.)
5. Execute attack and monitor network or contract behavior
6. Record results (block reorganization, transaction failures, or exploits)
7. Identify countermeasures and suggest mitigation strategies
8. End

*Software used:

SimBlock	Blockchain network simulator for consensus and propagation attack testing.
Remix IDE	Smart contract deployment and exploit simulation.
Crypto51.app	Visualization of 51% attack costs and network vulnerability.
Etherscan / Testnets	Verification of real blockchain transaction behaviors.

**** As applicable according to the experiment.***

Two sheets per experiment (10-20) to be used.

* Testing Phase: Compilation of Code (error detection)

NO ERROR

* Implementation Phase: Final Output (no error)

1. In SimBlock, configured a network with 100 nodes, attacker possessing 51% hash power.
2. Simulated block mining and observed the attacker overtaking the main chain.
3. In Remix IDE, deployed a vulnerable Solidity smart contract to test reentrancy exploits.
4. Executed malicious contract interactions that drained funds before the balance reset.
5. Observed attack outcomes and validated results against secure code patterns.

Attack Type	Parameters	Result	Impact
51% Attack	Attacker with >50% hash power	Chain reorganization	Consensus compromised
Double Spending	Two conflicting transactions	One invalidated	Funds spent twice
Sybil Attack	100 fake nodes	Consensus delay	Network instability
Reentrancy Attack	Unprotected withdraw() function	Funds drained	Contract emptied
Eclipse Attack	Node isolation	Partial fork	Transaction visibility lost

* Observation :

- Majority control in PoW systems can lead to block manipulation.
- Smart contract vulnerabilities stem from coding oversights rather than blockchain flaws.
- Network decentralization reduces exposure to Sybil and Eclipse attacks.
- Attack simulations help visualize how consensus models react to adversarial conditions.

**** As applicable according to the experiment.***
Two sheets per experiment (10-20) to be used

ASSESSMENT

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

Signature of the Student :

Name :

Signature of the Faculty :

Regn. No. :

**** As applicable according to the experiment.***
Two sheets per experiment (10-20) to be used