SDN + Blockchain Security Framework: Mathematical Model

1 Key Mathematical Components

We define the following mathematical frameworks:

- Graph Theory for Network Topology and Routing
- Game Theory for Attack Response Decisions
- Blockchain Model (PoA & PoW)
- Queuing Theory for Traffic and Congestion Management
- Smart Contract Trigger Function (Mathematical Definition)

2 Graph Model for Network and Routing

Since SDN handles routing dynamically, we represent the network as a weighted graph:

$$G = (V, E, W) \tag{1}$$

where:

- V is the set of network nodes (switches, routers, firewalls, endpoints).
- E is the set of network links.
- W is the weight function:

  $$w : E \to \mathbb{R}^+ \tag{2}$$

  where the weight represents the cost (latency, congestion, or security risk).

Routing between nodes follows Dijkstras Algorithm:

$$d(u, v) = \min \sum w(e), \quad e \in P \tag{3}$$

where P is the set of all possible paths from node u to node v.

3 Game Theory for Attack Response

We model network attack response as a two-player game between:

- Defender (SDN Controller + Blockchain)
- Attacker (Malicious Node or External Threat)

3.1 Payoff Matrix

Let:

- $A = \{A_1, A_2, A_3, A_4, A_5\}$ be SDN actions:
  - A_1: Remove Edge Device
  - A_2: Change Routing
  - A_3: Block Malicious IP
  - A_4: Revoke Endpoint PoA Certificate
  - A_5: Do Nothing (False Alarm case)
- $X = \{X_1, X_2, X_3\}$ be Attacker's actions:
  - X_1: DDoS Attack
  - X_2: Network Breach
  - X_3: Fake Attack to Evade Detection

The expected utility function for SDN (Defender) is:

$$U_D(A, X) = P_D(A) \cdot R(A) - P_A(X) \cdot C(X) \tag{4}$$

where:

- $P_D(A)$: Probability of correct response by SDN.
- $R(A)$: Reward of mitigating attack successfully.
- $P_A(X)$: Probability of attack occurring.
- $C(X)$: Cost of attack impact.

4 Blockchain Model (PoA & PoW)

We define the blockchain as a state transition system:

$$S_t = H(S_{t-1}, T_t) \tag{5}$$

where:

- S_t is the blockchain state at time t.

- H is a cryptographic hash function.
- T_t is the set of transactions (routing updates, security events).

For PoA-based node authentication, each node N_i must have a signed certificate:

$$\text{Cert}(N_i) = \text{Sign}_{CA}(ID_{N_i}, K_{pub}) \tag{6}$$

where K_{pub} is the nodes public key.

For PoW-based verification, each node computes a verification function:

$$V(P) = \sum_{i=1}^{n} f(P_i) \tag{7}$$

where $f(P_i)$ is a routing validation function ensuring the new path P satisfies latency and security constraints.

5 Queuing Model for Traffic Congestion

We model network congestion using M/M/1 Queues:

$$\rho = \frac{\lambda}{\mu} \tag{8}$$

where:

- λ is the packet arrival rate.
- μ is the packet processing rate.
- ρ is the traffic intensity (if $\rho > 1$, network congestion occurs).

If congestion is detected ($\rho > 0.8$), SDN triggers rerouting via Smart Contracts.

6 Smart Contract Function Definition

A smart contract triggers automated network defense. The trigger function is:

$$SC(A, X) = \begin{cases} 1, & \text{if } P_D(A) \cdot R(A) > P_A(X) \cdot C(X) \\ 0, & \text{otherwise} \end{cases} \tag{9}$$

where $SC(A, X) = 1$ means the smart contract executes the action A.

7 Summary

Component	**Mathematical Representation**
Network Routing	Graph $G = (V, E, W)$, Dijkstra's Algorithm
Attack-Response Decision	Game Theory Payoff Function $U_D(A, X)$
Blockchain State	State Transition $S_t = H(S_{t-1}, T_t)$
Authentication (PoA)	$\mathrm{Cert}(N_i) = \mathrm{Sign}_{CA}(ID_{N_i}, K_{pub})$
PoW Verification	$V(P) = \sum_{i=1}^{n} f(P_i)$
Traffic Congestion	M/M/1 Queue $\rho = \frac{\lambda}{\mu}$
Smart Contract Trigger	$SC(A, X) = 1$ if valid, else 0