© MAR 2026 | IRE Journals | Volume 9 Issue 9 | ISSN: 2456-8880
DOI: https://doi.org/10.64388/IREV9I9-1714875

Forensic-Assisted Structural Self-Reconfiguring Firewall

OMKAR SAHEBRAO KEDARI
PVG Science and Commerce College, affiliated to Savitribai Phule Pune University (SPPU), Pune, India.

Abstract- Traditional firewall architectures rely on static rule-based enforcement and fixed network segmentation, resulting in structural predictability and repeated bypass risks. This paper presents a Forensic-Assisted Structural Self-Reconfiguring Firewall architecture that enables real-time structural mutation governed by risk thresholds rather than conventional rule updates. The proposed framework integrates an AI-driven forensic reconstruction engine, predictive cyber twin simulation, and a blockchain-based structural intelligence ledger for validated containment retrieval. A structural decision engine dynamically modifies segmentation topology, trust boundaries, and routing paths during live threat conditions. To preserve operational continuity, a Transitional Flow Identity (TFI) mechanism ensures transactional dual-configuration switching with zero packet loss and uninterrupted session maintenance. The architecture introduces a self-learning, structurally adaptive firewall model designed to eliminate architectural predictability while maintaining system stability and continuous containment intelligence evolution.

Keywords- Structural Firewall Reconfiguration, AI-Powered Forensic Analysis, AI Cyber Twins, Adaptive Cyber Defense, Blockchain Security Ledger, Moving Target Defense, Zero-Trust Architecture.

I. INTRODUCTION

Modern firewall systems have evolved into next-generation and AI-driven platforms capable of deep packet inspection and behavioral threat detection. However, most existing solutions remain rule-centric and structurally static, leaving network architectures predictable under evolving threats. Even adaptive firewalls primarily modify policies rather than dynamically transforming containment topology during active attacks.

This research proposes a Forensic-Assisted Structural Self-Reconfiguring Firewall that enables real-time topology mutation guided by forensic intelligence and quantified risk thresholds. By integrating predictive simulation, real-time attack reconstruction, and session-preserving reconfiguration, the proposed framework enhances resilience and eliminates architectural predictability in modern network defense systems.

Problem Statement

Modern cyber threats increasingly exploit predictable network segmentation, static trust relationships, reusable authentication contexts, and internally permissive communication models. Traditional firewall architectures rely primarily on rule-based filtering and static policy enforcement. During live multi-stage attack scenarios, these systems are unable to dynamically restructure containment topology. As a result, once attackers perform reconnaissance and identify structural weaknesses, they can reuse discovered communication paths and trust boundaries to bypass defenses repeatedly.

Existing adaptive security solutions lack:

- Real-time structural mutation capability
- Forensic-driven architectural decision authority
- Session-preserving live reconfiguration
- Immediate containment for previously unseen attack patterns
- Structural unpredictability against repeated bypass attempts

Proposed Solution

This research proposes a Forensic-Assisted Structural Self-Reconfiguring Firewall Architecture capable of dynamically modifying containment topology during live attacks. The system integrates an AI-powered forensic engine that reconstructs attack progression in real time and guides structural firewall mutation under governed thresholds. The firewall performs controlled reconfiguration across segmentation, trust boundaries, routing paths, inspection depth, identity enforcement, encryption policies, and security posture layers.

- AI-Powered Cyber Twins for predictive adversarial simulation
- AI-Driven Forensic Reconstruction Engine for real-time attack analysis
- Blockchain-Based Structural Intelligence Ledger for validated containment storage
- Plan A (Ledger-Based Instant Structural Prevention)
- Plan B (Temporary Adaptive Containment for unknown attacks)
- Transactional Dual-Configuration Switching
- Transitional Flow Identity (TFI) Mechanism for zero packet loss
- Auto-Learning Structural Memory for continuous improvement

The firewall evolves from a static enforcement device into a self-learning, structurally adaptive containment system.

Method Used
The architecture operates through three intelligence loops:

1. Predictive Loop – AI Cyber Twins simulate bypass techniques and generate containment models.
2. Reactive Loop – During live attack, AI forensic engine reconstructs attack chain and triggers structural mutation.
3. Evolutionary Loop – Post-incident learning updates structural intelligence ledger and baseline behavior.

Key Results
The proposed system:

- Prevents lateral movement during APT campaigns
- Dynamically reassigns trust boundaries
- Mutates segmentation topology in real time
- Applies temporary containment for unknown attacks
- Preserves active client-server sessions
- Ensures attackers never encounter identical firewall structure twice
- Continuously improves containment intelligence.

II. LITERATURE REVIEW

2.1 Zero Trust Architecture
Strength: Identity-centric enforcement.
Limitation: Does not mutate segmentation topology.

2.2 Moving Target Defense (MTD)
Strength: Randomization of attack surface.
Limitation: Not forensic-driven; limited structural authority.

2.3 AI-Based Adaptive Firewalls
Strength: Improved anomaly detection.
Limitation: Rule updates instead of topology mutation.

2.4 Blockchain-Based Security Frameworks
Strength: Tamper-proof logging.
Limitation: Rarely used for structural containment retrieval.

2.5 Research Gap Identified
No existing framework integrates:

- Real-time forensic reconstruction
- Blockchain-backed containment templates
- Structural topology mutation
- Session-preserving reconfiguration
- Predictive cyber twin simulation

Your system fills this gap.

III. PROBLEM GAP

3.1 Limitations of Existing Firewalls
Current systems:

- Maintain fixed segmentation topology
- Preserve implicit internal trust
- Allow east-west traffic by default
- Reorder policies but not architecture
- Do not mutate containment graph

3.2 Scientific Gap
There is limited research on:

- AI forensic engine acting as structural decision authority
- Blockchain-stored structural intelligence for rapid prevention
- Live topology mutation with zero downtime
- Temporary adaptive containment followed by optimized restructuring

- Structural mutation prevents repeat reconnaissance.

IV. SYSTEM OVERVIEW

The proposed architecture consists of:

1. AI Cyber Twin Layer
   - Red Team AI (Attack Simulation)
   - Blue Team AI (Defense Simulation)
2. Blockchain Structural Intelligence Ledger
   - Stores validated containment templates
   - Stores temporary containment strategies
   - Stores optimized structural models
3. Live Monitoring and Detection Engine
4. AI-Powered Forensic Reconstruction Engine
5. Structural Decision Engine
   - Plan A: Ledger-Based Retrieval
   - Plan B: Temporary Adaptive Containment
6. Transactional Firewall Reconfiguration Engine
7. Auto-Learning Structural Memory.

V. PRE-ATTACK PHASE: AI CYBER TWIN LEARNING

Red Team AI (In-build)

- Simulates attacks
- Tests bypass techniques
- Explores trust boundary weaknesses
- Evaluates lateral movement paths

Blue Team AI

- Designs containment strategy
- Tests segmentation mutation
- Validates routing modifications
- Generates structural defense template

Validated containment model stored in blockchain ledger.
This ensures rapid lookup during real attack.

VI. LIVE APT ATTACK SCENARIO

Phase 1 – Initial Compromise
Phishing infection inside the user subnet.
Phase 2 – Reconnaissance
Attacker scans internal services, identifies database zone.
Phase 3 – Lateral Movement
Credential replay from User Zone to File Server to Database Zone.
Phase 4 – Data Exfiltration Attempt
Outbound encrypted channel initiated.

VII. LIVE ATTACK HANDLING MECHANISM ATTACK (APT)

An Advanced Persistent Threat (APT) is a sophisticated, long-term cyberattack in which an attacker gains unauthorized access to a network and remains undetected for an extended period. Instead of causing immediate damage, the attacker moves slowly and strategically—performing internal reconnaissance, escalating privileges, and maintaining persistence—while targeting sensitive data or critical systems. The primary objective is typically data theft, espionage, or continuous monitoring, rather than rapid disruption.

- Gain initial foothold
- Perform internal reconnaissance
- Identify trust relationships
- Exploit segmentation weaknesses
- Move laterally using valid credentials
- Establish persistence
- Exfiltrate sensitive data

Traditional firewalls enforce policies but do not modify containment structure during attack progression. Even advanced rule updates fail to eliminate structural predictability.
This research introduces a firewall capable of:

- Learning before attack
- Adapting during attack
- Evolving after attack
- Preserving operational continuity
- Preventing structural predictability

Step 1 – AI Forensic Trigger
The forensic engine:

- Reconstructs attack graph
- Maps compromised nodes
- Identifies exploited trust boundaries
- Predicts propagation path

- Calculates structural risk score

If threshold exceeded → structural reconfiguration initiated.

Step 2 – Plan A: Blockchain Match
System searches ledger.
If similar APT pattern exists:

- Retrieve validated structural template
- Apply immediately
- Perform structural mutation
- Preserve active sessions

Step 3 – Plan B: Temporary Adaptive Containment
If no exact match found:

- Apply closest containment strategy
- Tighten segmentation
- Escalate inspection
- Restrict cross-zone communication
- Activate quarantine

Temporary containment prevents propagation while deep analysis continues.

Step 4 – Optimized Structural Reconfiguration
After full forensic analysis, firewall performs structural mutation across:

7.1 Evaluation Metrics Definition

- Detection Time: Time from anomaly detection to forensic trigger.
- Reconfiguration Latency: Time required to apply structural mutation.
- Session Preservation Rate: % of active sessions maintained during mutation.
- Lateral Movement Reduction: % decrease in successful east-west traversal.
- Packet Loss Rate: % packets dropped during transition.
- CPU Overhead: Additional processing load introduced.

VIII. TYPES OF STRUCTURAL RECONFIGURATION PERFORMED

The firewall dynamically modifies:

1. Network segmentation topology
2. Trust boundary relationships
3. Dynamic access control enforcement
4. Traffic routing paths
5. Inspection depth and DPI activation
6. Quarantine and isolation zones
7. Service exposure minimization
8. Identity and authentication enforcement
9. Encryption and TLS inspection policy
10. Policy graph ordering
11. Resource allocation scaling
12. Deception activation
13. Zero-trust tightening
14. Cross-zone communication freeze
15. Behavioral baseline recalibration
16. Rollback and structural restoration.

Governance Properties
All modifications are:

- Threshold-triggered
- Risk-governed
- Transactionally applied
- Reversible
- Session-preserving

IX. TRANSITIONAL FLOW IDENTITY (TFI) MECHANISM

During live reconfiguration:
Incoming packets are assigned:

- Flow Identifier
- Policy Version Binding
- Reconfiguration Epoch Marker

This ensures:

- Existing sessions follow original policy context
- New sessions follow updated structural context
- No packet loss
- No routing confusion
- No TCP reset

The firewall operates under a dual-configuration model, where:

- Old configuration serves active sessions
- New configuration activates atomically

X. PREVENTION OF REPEATED BYPASS

After reconfiguration:

- Trust graph changes
- Segmentation boundaries mutate

- Routing topology adjusts
- Authentication enforcement tightens
- Deception nodes activate

If attacker retries:
They encounter a different structural firewall state.
This eliminates structural predictability.

Feature	Traditional FW	Zero Trust	MTD	proposed d System
Structural Mutation	No	No	Partial	Yes
Forensic Authority	No	No	No	Yes
Blockchain Containment	No	No	No	Yes
Session Preservation	Limited	Limited	No	Yes
Adaptive Learning	No	No	No	Yes

XI. SYSTEM ARCHITECTURE DIAGRAM

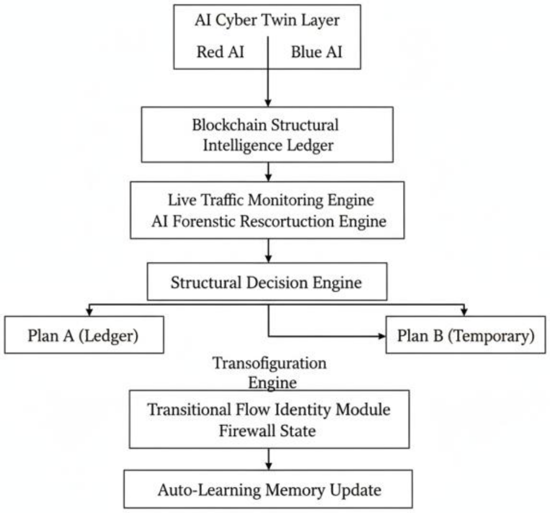
AI Cyber Twin Layer
Red AI
Blue AI
Blockchain Structural Intelligence Ledger
Live Traffic Monitoring Engine
AI Forenstic Rescortuction Engine
Structural Decision Engine
Plan A (Ledger)
Plan B (Temporary)
Transofiguration Engine
Transitional Flow Identity Module
Firewall State
Auto-Learning Memory Update

Figure 1 illustrates the structural interaction between intelligence, decision, and mutation layers.

1. Traffic Monitoring Layer
   - All inbound and outbound traffic first enters the live monitoring engine.
   - Behavioral deviations and anomaly indicators are detected in real time.
   - Suspicious traffic is forwarded to the forensic engine.

2. AI Forensic Reconstruction Engine
   - Reconstructs the attack progression graph.
   - Identifies compromised nodes and exploited trust paths.
   - Predicts potential lateral movement.
   - Calculates structural risk score.

3. Structural Decision Engine
   - Evaluates whether risk threshold is exceeded.
   - Determines whether Plan A (ledger-based) or Plan B (adaptive containment) should be executed.
   - Ensures mutation is governance-controlled.

4. Blockchain Structural Intelligence Ledger
   - Stores validated containment templates.
   - Provides rapid retrieval of previously optimized structures.
   - Enables learning-based structural response.

5. Transactional Reconfiguration Engine
   - Applies segmentation mutation.
   - Modifies routing paths and trust boundaries.
   - Adjusts inspection depth and enforcement policies.
   - Executes changes transactionally.

6. Transitional Flow Identity (TFI) Module
   - Assigns flow identifiers and policy bindings.
   - Maintains dual configuration state.
   - Ensures zero packet loss and no session interruption.

7. Intelligence Loop Integration
   - Predictive Loop (Cyber Twins)
   - Reactive Loop (Forensic Engine)
   - Evolutionary Loop (Ledger Update)

The diagram represents a closed-loop adaptive containment architecture.

X. RECONFIGURATION FLOWCHART

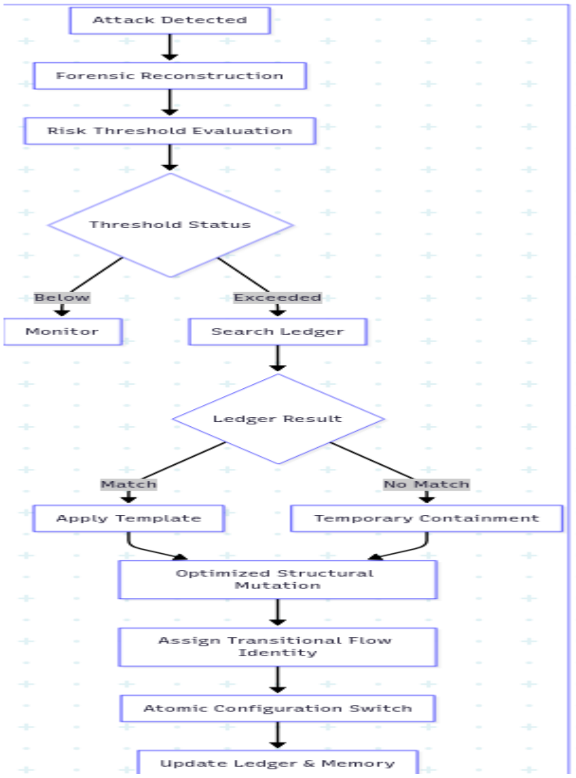

Figure 2 describes the real-time structural decision workflow during an attack.

1. Continuous Monitoring
   - Traffic anomalies are continuously observed.
   - If no anomaly is detected, the system continues normal operation.

2. Forensic Risk Evaluation
   - The attack graph is reconstructed.
   - Structural risk score is computed.
   - Risk threshold comparison performed.

3. Decision Node
   - If risk ≤ threshold → monitoring continues.
   - If risk > threshold → containment triggered.

4. Plan A: Ledger-Based Containment
   - System queries blockchain ledger.
   - If a match found, a validated structural template is retrieved.
   - Immediate topology mutation applied.

5. Plan B: Temporary Adaptive Containment
   - If no template matches, temporary containment is applied.
   - Cross-zone traffic restricted.
   - Quarantine and inspection intensified.

6. Transactional Structural Mutation
   - Structural changes executed atomically.
   - Session continuity maintained via TFI.

7. Post-Incident Learning
   - Optimized structure stored in ledger.
   - Baseline recalibrated.
   - System intelligence updated.

The flowchart demonstrates a threshold-governed, intelligence-driven, session-preserving containment process.

1. Flow Direction (End-to-End Traffic Movement)

Data Flow Direction

- Client traffic enters through the Data Plane Interface.
- Packets pass through the Monitoring Layer.
- Normal traffic continues toward the server.
- Suspicious traffic is mirrored to the forensic engine.
- During reconfiguration, packets are routed through transitional mapping without interruption.

Intelligence Flow Direction

- Monitoring → Forensic Engine → Decision Engine.
- Decision Engine → Ledger Query → Reconfiguration Engine.
- Reconfiguration Engine → Policy Update → Enforcement Layer.
- Post-event intelligence → Ledger Update.

This establishes a bidirectional architecture:

- Forward data flow (traffic movement)
- Feedback intelligence flow (adaptive learning)

2. Module Interaction Explanation

Monitoring ↔ Forensic Engine

- Monitoring detects anomaly patterns.
- Forensic module reconstructs attack graph.

- Risk score is generated and returned.

Forensic Engine ↔ Decision Engine

- Forensic output includes:
  - Compromised nodes
  - Propagation probability
  - Asset sensitivity
  - Decision engine evaluates containment necessity.

Decision Engine ↔ Blockchain Ledger

- Ledger is queried for:
  - Similar attack fingerprints
  - Validated structural templates
- If match is found → Plan A executed.
- If no match → Plan B activated.

Decision Engine ↔ Reconfiguration Engine

- Mutation instructions transmitted.
- Structural policy graph modified.
- Routing and segmentation updated.

Reconfiguration Engine ↔ TFI Module

- Flow IDs assigned.
- Session continuity preserved.
- Dual configuration state maintained during transition.

3. Decision Logic Explanation

The diagram implements threshold-governed containment logic:

1. Detecting anomalies.
2. Compute structural risk score.
3. Compare against a predefined threshold.
4. If below threshold → monitor only.
5. If above threshold:
   - Query ledger.
   - Execute optimal mutation strategy.
6. Store new intelligence.

Decision logic is:

- Deterministic in trigger condition.
- Adaptive in containment strategy.
- Evolutionary in learning behavior.

4 Control Path vs Data Path (Critical Academic Requirement)

◆ Data Path (Traffic Plane)

Handles:

- Packet forwarding
- DPI inspection
- Routing
- Session handling
- Encryption enforcement

This path processes live traffic.

Control Path (Intelligence Plane)
Handles:

- Risk computation
- Attack graph analysis
- Ledger retrieval
- Mutation planning
- Policy generation

This path does NOT forward packets. It governs structural transformation.

Separation Benefit

- Prevents decision delay in data forwarding.
- Enables atomic mutation without dropping sessions.
- Ensures deterministic containment.

Your architecture follows a control-plane governed data-plane mutation model.

5. Connecting Visual Blocks to Operational Behavior

Visual Block	Operational Meaning
Monitoring Layer	Real-time anomaly sensing
Forensic Engine	Attack reconstruction & risk modeling
Decision Engine	Containment strategy selection

Blockchain Ledger	Historical structural intelligence
Reconfiguration Engine	Topology mutation executor
TFI Module	Session preservation controller
Enforcement Layer	Updated policy implementation

Each visual component represents a functional containment stage in the adaptive firewall lifecycle.

6. Structural Transition Behavior During Live Attack

When reconfiguration is triggered:

- The new structural graph is prepared in a shadow state.
- Transitional Flow Identity maintains session mapping.
- Atomic switch applied.
- Old structure safely deactivated.
- No packet loss.
- No session break.
- Structural intelligence updated.

This confirms:

- Real-time mutation.
- Zero-disruption containment.
- Learning-based evolution.

XII. CORE SCIENTIFIC CONTRIBUTION

Firewall reconfiguration is structural, not rule-based
AI forensic engine acts as architectural decision authority
Blockchain stores validated containment intelligence
System adapts only when risk threshold exceeded
Temporary containment ensures zero-delay response
Structural mutation prevents repeated bypass
Session-preserving transactional switching
Every incident enhances structural intelligence.

11.1. Conclusion

This research transforms the firewall from a static enforcement device into a forensic-assisted adaptive containment architecture.

By integrating AI Cyber Twins, real-time forensic reconstruction, blockchain-based intelligence storage, and governed structural mutation, the firewall:

- Learns predictively
- Adapts reactively
- Evolves continuously
- Prevents structural predictability
- Maintains operational stability

The proposed system introduces a new paradigm in adaptive cyber defense against Advanced Persistent Threats.

REFERENCES

[1] S. Rose, O. Borchert, S. Mitchell, and S. Connelly, *"Zero Trust Architecture,"* NIST Special Publication 800-207, National Institute of Standards and Technology (NIST), 2020.

[2] S. Jajodia, A. K. Ghosh, V. Swarup, C. Wang, and X. S. Wang, *Moving Target Defense: Creating Asymmetric Uncertainty for Cyber Threats,* New York, NY, USA: Springer, 2011.

[3] Y. Zhuang, S. Zhang, and A. Dehghantanha, "A Survey on Moving Target Defense Techniques and Applications," *IEEE Communications Surveys & Tutorials,* vol. 22, no. 1, pp. 1–26, 2020.

[4] Y. Xin, L. Kong, Z. Liu, Y. Chen, Y. Li, H. Zhu, M. Gao, and H. Hou, "Machine Learning and Deep Learning Methods for Cybersecurity," *IEEE Access,* vol. 6, pp. 35365–35381, 2018.

[5] R. Sommer and V. Paxson, "Outside the Closed World: On Using Machine Learning for Network Intrusion Detection," in *Proc. IEEE Symposium on Security and Privacy,* 2010, pp. 305–316.

[6] N. Moustafa and J. Slay, "UNSW-NB15: A Comprehensive Data Set for Network Intrusion Detection Systems," in *Proc. Military Communications and Information Systems Conference (MilCIS),* 2015.

[7] W. Wang, M. Zhu, X. Zeng, X. Ye, and Y. Sheng, “Malware Traffic Classification Using Convolutional Neural Network for Representation Learning,” in *Proc. IEEE International Conference on Communications (ICC),* 2017.

[8] K. Christidis and M. Devetsikiotis, “Blockchains and Smart Contracts for the Internet of Things,” *IEEE Access,* vol. 4, pp. 2292–2303, 2016.

[9] M. Conoscenti, A. Vetrò, and J. C. De Martin, “Blockchain for the Internet of Things: A Systematic Literature Review,” in *Proc. IEEE/ACS International Conference on Computer Systems and Applications (AICCSA),* 2016.

[10] J. Sherry et al., “Making Middleboxes Someone Else’s Problem: Network Processing as a Cloud Service,” in *Proc. ACM SIGCOMM,* 2012.