MSW MANAGEMENT -Multidisciplinary, Scientific Work and Management Journal
ISSN: 1053-7899
Vol. 36 Issue 1, 2026, Pages: 2647-2655

Reinforcement Learning–Driven Moving Target Defense for Ransomware Mitigation in SDN-Enabled Enterprise Networks

Rajneesh Present SS[1]
School of Computer Science Engineering and Artificial Engineering (SCAI)
VIT Bhopal University, Bhopal, Madhya Pradesh
rajneesh.present2021@vitbhopal.ac.in

Dr. Adarsh Patel[2]
School of Computer Science Engineering and Artificial Engineering (SCAI)
VIT Bhopal University, Bhopal, Madhya Pradesh
adarsh.patel@vitbhopal.ac.in

Dr. Shahana Gajala Queshi[3]
School of Computer Science Engineering and Artificial Engineering (SCAI)
VIT Bhopal University, Bhopal, Madhya Pradesh
shahana.quershi@vitbhopal.ac.in

Dr. Rizwan Ur Rahman[4]
School of Computer Science Engineering and Artificial Engineering (SCAI)
VIT Bhopal University, Bhopal, Madhya Pradesh
rizwan.ur@vitbhopal.ac.in

Abstract:
Ransomware has evolved into a highly organized and technically sophisticated cyber threat capable of causing large-scale disruption to enterprise infrastructures. The increasing adoption of Software-Defined Networking (SDN) has introduced architectural efficiencies through centralized control and programmability; however, these same characteristics have expanded the attack surface available to adversaries. Conventional security mechanisms predominantly rely on static configurations and reactive detection models, which are insufficient against ransomware campaigns that employ adaptive reconnaissance, lateral movement, and evasion techniques [1]. This research presents the design and experimental evaluation of a Reinforcement Learning (RL)–based Moving Target defence (MTD) framework tailored for SDN-enabled enterprise environments. The framework models network defence as a sequential decision-making problem and dynamically mutates network attributes, routing paths, and exposure surfaces to invalidate attacker knowledge. A controlled experimental testbed is used to simulate realistic enterprise traffic and ransomware attack scenarios. The experimental analysis indicates measurable improvements in early-stage attack disruption, response latency, and containment efficiency when compared to traditional rule-based and static defence systems [2]. The results demonstrate that integrating RL-driven intelligence with proactive defence strategies significantly enhances the resilience of programmable enterprise networks against ransomware threats.

Keywords: Reinforcement Learning, Moving Target defence, Ransomware, Software-Defined Networking, Adaptive Network Security.

INTRODUCTION

Ransomware has transitioned from opportunistic malware into a strategic cyber weapon employed by organized criminal groups and advanced persistent threat actors. Modern ransomware campaigns typically involve multi-stage operational workflows that begin with initial access, followed by internal reconnaissance, privilege escalation, lateral movement, data exfiltration, and eventual encryption of critical enterprise assets [3]. The financial and operational consequences of such attacks have escalated significantly, with enterprises facing prolonged downtime, loss of sensitive information, and reputational damage.

The rapid digital transformation of enterprise infrastructures has led to the adoption of highly programmable and centralized networking paradigms. Software-Defined Networking represents a paradigm shift that decouples the network control logic from data forwarding components, enabling fine-grained traffic management and rapid policy enforcement. While these characteristics improve operational agility, they simultaneously create systemic security challenges. A single compromised control point can influence large portions of the network, thereby enabling the rapid propagation of ransomware across segmented infrastructure [4].

Traditional cybersecurity strategies within enterprise networks have primarily relied on perimeter-based defence models. Firewalls, intrusion detection systems, and signature-based malware scanners have historically served as the primary lines of defence. However, these mechanisms are inherently reactive and depend on predefined rules or previously observed attack signatures. Ransomware operators increasingly employ polymorphic payloads, fileless execution, and encryption-based command-and-control channels, which significantly reduce the effectiveness of static detection techniques [5].

Moving Target defence has emerged as a proactive cybersecurity paradigm designed to reduce system predictability and invalidate adversarial reconnaissance. By continuously altering system configurations, network attributes, and exposed services, MTD mechanisms seek to increase attacker uncertainty and operational complexity. In their conventional form, MTD techniques rely on static schedules or randomization intervals, such as periodic IP address shuffling or port remapping. While such approaches introduce diversity, they remain susceptible to pattern learning by persistent attackers [6].

Reinforcement Learning provides a mathematical framework for autonomous decision-making under uncertainty. By interacting with the environment and optimizing long-term reward signals, RL agents can learn adaptive strategies that respond effectively to changing system conditions. Within cybersecurity, RL has been applied to intrusion response, adaptive access control, and network traffic optimization. The integration of RL with MTD offers the possibility of creating intelligent defence mechanisms that proactively adapt to adversarial behaviour rather than relying on fixed randomization patterns [7].

The primary goal of this research is to design an intelligent security framework that combines the adaptive capabilities of Reinforcement Learning with the proactive surface randomization principles of Moving Target defence within an SDN-based enterprise environment. Unlike existing approaches that treat detection and response as discrete processes, the proposed framework tightly integrates continuous monitoring, decision-making, and dynamic network reconfiguration. This integration enables near real-time defence adaptation based on observed behavioural indicators of ransomware activity [8].

This study contributes to the field of network security by presenting a comprehensive architecture for RL-driven Moving Target defence, implementing the architecture within a programmable SDN environment, and experimentally evaluating its effectiveness against realistic ransomware scenarios. The experimental results highlight the potential of learning-enabled proactive defence systems to significantly improve the security posture of modern enterprise networks.

MSW MANAGEMENT -Multidisciplinary, Scientific Work and Management Journal
ISSN: 1053-7899
Vol. 36 Issue 1, 2026, Pages: 2647-2655

RELATED WORK

Early research in ransomware defence has predominantly focused on signature-based malware detection and static behavioural heuristics. Signature-based techniques attempt to identify malicious binaries through cryptographic hashes or known byte patterns. These approaches, while effective against previously catalogued malware samples, exhibit minimal resilience against evolving ransomware variants and zero-day payloads [9]. Behaviour-based systems attempt to identify malicious intent by monitoring file system activity, process injection, and abnormal encryption behaviour. Although these methods improve detection capabilities, they often suffer from high false-positive rates in complex enterprise environments.

Research into SDN security has introduced centralized firewalling, flow-level anomaly detection, and controller-integrated policy enforcement mechanisms. These systems benefit from the global visibility provided by SDN controllers, enabling consistent policy application across the network [10]. However, the majority of these frameworks rely on static rule sets and predefined thresholds, limiting their effectiveness against adaptive threats such as ransomware.

Research on Moving Target defence has gained significant attention as a proactive alternative to traditional reactive security mechanisms. Early implementations of MTD focused primarily on operating system level randomization, such as address space layout randomization and instruction set randomization, to mitigate exploitation attempts against memory corruption vulnerabilities [11]. Network-level MTD approaches later introduced techniques such as IP address hopping, port shuffling, and dynamic routing mutation. These techniques demonstrated measurable success in disrupting automated scanning tools and exploit frameworks by continuously invalidating attacker reconnaissance data [12].

Despite these advances, existing MTD implementations have largely relied on fixed or time-based randomization strategies. Such deterministic behaviour can be observed and modelled by persistent adversaries over time. Attackers with sufficient dwell time can learn the reconfiguration patterns and adapt their strategies, reducing the long-term effectiveness of static MTD deployments. Furthermore, traditional MTD does not incorporate intelligence regarding threat context, resulting in blind randomization that may introduce unnecessary overhead without proportional security benefit [13].

Machine learning techniques have been increasingly explored to enhance cyber defence adaptability. Supervised learning methods have been applied to malware classification, intrusion detection, and anomaly-based monitoring in enterprise environments. However, supervised approaches require labelled datasets and struggle with previously unseen threats and evolving attack methodologies [14]. Reinforcement Learning has emerged as a promising alternative due to its ability to learn optimal behaviour through interaction with the environment. Existing studies have demonstrated the potential of RL in adaptive routing security, dynamic firewall rule management, and autonomous intrusion response systems [15].

In the context of SDN, RL has been applied to optimize traffic engineering, congestion control, and attack mitigation. Prior work has shown that RL-enabled controllers can dynamically install flow rules to mitigate distributed denial-of-service attacks and traffic flooding scenarios [16]. However, the majority of these studies focus on availability attacks rather than ransomware-specific behaviours such as lateral movement, file system targeting, and stealthy command-and-control communication.

A critical gap in the current literature lies in the lack of integrated frameworks that combine RL-driven intelligence with Moving Target defence specifically tailored to ransomware mitigation in SDN-based enterprise networks. Most existing solutions address isolated aspects of the problem, either emphasizing detection or focusing on generic network hardening. Few studies have explored proactive, learning-enabled surface randomization that continuously adapts in direct response to ransomware behaviour patterns [17]. The present research addresses this gap by proposing a tightly integrated RL-MTD framework designed explicitly for ransomware defence within programmable network infrastructures.

SYSTEM ARCHITECTURE

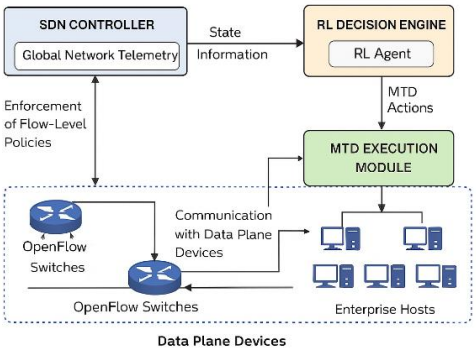

Fig. 1. Reinforcement Learning–Based Moving Target defence Architecture in an SDN Environment

The proposed framework is designed as a modular architecture that integrates Reinforcement Learning–based decision-making with SDN-enabled network control and Moving Target defence execution mechanisms. The architecture is structured to enable real-time monitoring, adaptive policy generation, and rapid enforcement of network reconfiguration actions without disrupting legitimate enterprise operations.

Fig. 1 illustrates the overall system architecture, showing the interaction between the SDN controller, the RL decision engine, and the MTD execution module. The SDN controller acts as the central coordination entity responsible for collecting global network telemetry, enforcing flow-level policies, and managing communication with data plane devices. The RL agent operates as an intelligent control module that continuously analyses network state information and determines optimal defence actions.

Enterprise Sdn Infrastructure Model

The enterprise network model is constructed using a multi-tier SDN architecture consisting of access, aggregation, and core layers. End hosts, application servers, and storage systems are connected through OpenFlow-enabled switches that forward traffic according to policies installed by the SDN controller. Logical segmentation is implemented using virtual local area networks and software-defined overlays to isolate critical assets and limit unnecessary east–west communication [18].

This design allows the defence framework to dynamically adjust segmentation boundaries in response to observed ransomware activity. For example, high-risk network segments can be isolated in real time, and access to sensitive services can be selectively restricted without manual administrative intervention.

MSW MANAGEMENT -Multidisciplinary, Scientific Work and Management Journal
ISSN: 1053-7899
Vol. 36 Issue 1, 2026, Pages: 2647-2655

Threat Model And Assumptions

The threat model assumes that adversaries gain initial access through common enterprise attack vectors such as phishing emails, malicious attachments, compromised credentials, or exploitation of unpatched vulnerabilities. Once inside the network, attackers are assumed to possess the capability to perform network reconnaissance, lateral movement, privilege escalation, and encrypted communication with external command-and-control infrastructure [19].

The framework assumes that attackers do not possess real-time visibility into defence reconfiguration decisions. While attackers may collect historical information about network structure, continuous MTD-driven mutations invalidate stale reconnaissance data. The model further assumes that the SDN controller and RL components are protected by baseline hardening mechanisms, including secure boot and access control.

Design Principles of the Framework

The framework is guided by three primary design principles: adaptivity, resilience, and performance preservation. Adaptivity ensures that defensive behaviour evolves in response to observed threat dynamics. Resilience emphasizes fault tolerance and the ability to maintain defensive capability under partial system compromise. Performance preservation ensures that legitimate traffic experiences minimal disruption during security-driven network mutations [20].

To achieve these goals, the architecture separates observation logic, decision logic, and execution logic into independent modules. This decoupling allows the RL agent to evolve its decision-making capabilities without interfering with the underlying network control mechanisms.

Reinforcement Learning Decision Engine

The RL decision engine is implemented as a policy-driven agent that continuously interacts with the SDN environment. The agent observes network state variables derived from flow statistics, anomaly detection modules, and system telemetry. Based on these observations, the agent selects actions that correspond to specific MTD operations.

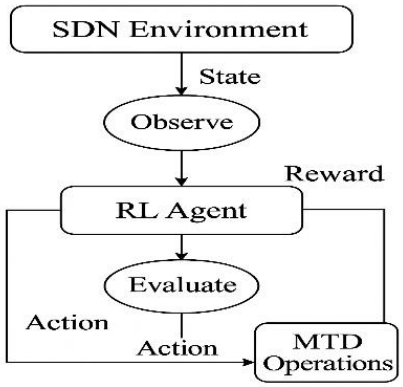

Fig. 2. RL Decision Flow

The decision-making process follows an observe–evaluate–act cycle, as conceptually depicted in Fig. 2. The agent evaluates the expected long-term impact of each potential defence action before issuing reconfiguration commands to the SDN controller. Experience replay and delayed target updates are incorporated into the learning process to stabilize policy convergence under dynamic network conditions [21].

Moving Target Defence Execution Module

The MTD execution module translates high-level defence actions produced by the RL agent into concrete network mutations. These mutations include dynamic reassignment of IP addresses, randomized port exposure, adaptive path diversification, and on-demand service relocation. All mutations are performed in a controlled and reversible manner to ensure system stability [22].

The module maintains synchronization with the SDN controller to ensure that reconfiguration actions do not conflict with existing traffic engineering policies. This coordination enables seamless integration of proactive defence behaviour with routine network management operations.

METHODOLOGY

The suggested defence system is developed in a systematic methodological manner that ransomware defence is a discrete decision-making dilemma. Reinforcement Learning integration with Moving Target Defence demands that the network environment, adversarial behaviour and system feedback systems be formalised accurately, as only in this way can stable and effective learning in dynamic and potentially hostile environments be achieved. This part describes the methodological underpinnings of the formal construction of the RL-based MTD framework, such as the underlying Markov model, state description, action design, reward formulation and training processes.

Formulation of the defence problem

The network defence problem is modelled as a Markov Decision Process (MDP), and every decision epoch is associated with a discrete time-step during which the RL agent monitors the prevailing network circumstances and chooses a suitable mitigation strategy. The formal definition of the MDP is presented by the (S, A, P, R, g) where S is the set of all possible network states, A is the action space, P(s'|s,a) is the transition probability between the current state s and s' when taking action a, R(s,a) is the instantaneous reward and g is the discount factor balancing long- and short-term goals [23].

Each state is a point in time image of the network security posture, traffic patterns, levels of anomalies, and host interaction behaviours. Actions are certain MTD-based reconfiguration measures, which are to deter ransomware like propagation or reconnaissance chain. This means that with the defence challenge in form of an MDP, the learning agent can reason on cumulative benefits of defence over more than one time-step, instead of basing only on short-term rule triggers.The state transition model is a stochastic model of network response to benign and malicious conditions. There are usually identifiable steps that ransomware encounters- initial reconnaissance, lateral movement, privilege escalation, and payload execution, and the transition structure helps the RL agent deduce such patterns of progression. This type of modelling makes it possible to conduct proactive defence: the agent becomes aware of the attack traces and develops methods of countering them before the ransomware enters the encryption phase.

State Space Design

It is essential to design an informative space that is easily tracked to facilitate efficient learning. The state space in this framework is based on multi-dimensional telemetry features of SDN controller, anomaly detection engines and distributed endpoint monitors. The main

MSW MANAGEMENT -Multidisciplinary, Scientific Work and Management Journal
ISSN: 1053-7899
Vol. 36 Issue 1, 2026, Pages: 2647-2655

characteristics are the flow-level rates of packet arrival, connection setup rates, source-destination entropy, abrupt changes in host communication networks, file-system access detection abnormalities, and cumulative risk scores based on statistical and machine-learned detectors [24].

In order to avoid the uncontrolled expansion of the state space which can impede the convergence and raise the computational cost, feature normalization is used. The values of entropy are normalized to a constant range and temporal smoothing filters are applied to remove noise caused by the short-lived spikes. Other dimensionality reduction methods, like Principal Component Analysis (PCA) or compression through autoencoders, guarantee the RL agent is provided with small but informative network activity representations.

Such thoughtful construction of the state representation contributes to the generalization of the RL system to various workloads, and the defence is also resistant to the changes in enterprise traffic patterns.

Action Space Definition

Action space determines the choice of MTD strategies that the RL agent is capable of choosing at a given decision step. These are targeting to hinder the development of ransomware by dynamically varying network properties. These measures are IP address mutation, port shuffling, path diversification through alternate routing, flow table priorities change, temporary isolations of suspicious hosts, and limitation of selective service exposure [25].

Every movement is constrained by the operational limitation in order to prevent a network meltdown. An example is that IP mutation operations need to occur progressively to avoid the disconnection of existing sessions, routing path diversification should be restricted to topologically feasible links to ensure the preservation of QoS of real-time applications. On similar note, the host isolation measures are never called unless the sources of anomalies exceed set limits in order to maintain the benign users without service interruption.

The RL agent thus does not just learn what works, but when and how much of that should be used, finding a trade between proactive defence and operational continuity.

Engineering of Reward Functions

Reward function has been central in the determination of the agent behaviour. This reward design is a multi-objective design that penalizes actions that lead to downgrade of the network or unnecessary disruptive actions and reinforces the actions that minimize malicious behaviour indicators or effectively block the ransomware progress.

When malicious ways of communication are disrupted, when the lateral movement efforts fail, or even when the entropy of suspicious flow goes back to its usual level, positive rewards are given. Negative rewards are sent to defence operations that cause excessive latency, packet loss, legitimate service disruption, or false-positive isolation of benign hosts [26].

This two-fold goal reward mechanism makes sure that the RL agent is trained to have not only an effective defence strategy but an effective and operationally efficient one. The agent is not encouraged to use over-aggressive mutation strategies and learns on interpreting contextual cues to use interventions that have minimal collateral effects.

Training Strategy and Learning Algorithm

The framework utilises a Deep Q-Network (DQN) as the main learning algorithm because it is suitable in situation of high-dimensional state spaces and discrete action modelling. The DQN is based on the neural network function approximator that approximates Q-values of every (state, action) pair. Experience replay also decorrelates observations and stabilizes learning through historical transitions by storing historical transitions and random sampling mini-batches to train. Additional stability is also achieved through periodic target network synchronization to keep a slowly updated secondary Q-network that minimizes oscillations in value estimation [27]. The training is done in two stages. Synthetic ransomware is applied to the offline phase in which the agent is initialized with baseline strategies, which would converge more rapidly when deployed. During the stage of online work, the process of continuous learning takes place in the SDN live testbed to enable the agent to adapt to the dynamic traffic and the unknown ransomware types.

This hybrid model of training makes sure that the RL agent does not forget fundamental defensive knowledge but secures the continued development following the new threats and environmental variations.

EXPERIMENTAL SETUP

The experimental analysis is implemented in an experimental SDN testbed intended to simulate the realistic conditions of enterprise networks and allows to repeat, configure, and observe the ransomware behaviours on a fine scale. The testbed consists of virtualized elements, programmable switching elements and monitored host systems that are used to offer a representative but controllable environment of testing the RL-based Moving Target Defence framework. The focus is made on having a balance between realism and experimental accuracy whereby the results are an accurate representation of the operational scenario but the variables can be tightly controlled.

Testbed Architecture

The test setup on Mininet is constructed with network emulation on the basis of Mininet and a production-grade SDN controller. Mininet offers a flexible and lightweight platform to deploy virtual hosts, switches and links to enable the creation of multiple complex topologies of the enterprise without necessarily involving physical network devices. The emulated environment is based upon a medium size enterprise architecture that includes user workstations, application servers, authentication and identity services, storage clusters, and separate backup systems [28]. This topology allows various patterns of communication and eastwest and northsouth traffic flows, which are very close to the real-life corporate IT infrastructures.

The SDN controller that was used in the testbed is the open daylight Boron release which has been chosen based on its stability, modularity and the fact that it is compatible with openflow enabled switches. The controller is the point of coordinated orchestration, which is in charge of installing flow, policy implementation, and telemetry. Notably, the Reinforcement Learning agent can be deployed as a separate decision-making unit, which interacts with the controller via the secure REST and gRPC-based APIs. This isolation is so that computation associated with the learning does not monitor real-time controlled behavior. In addition, the scaling of model complexity can be done independently by decoupling the learning engine and made available in the future in combination with distributed RL architectures.

The network topology consists of several virtual switches based on OpenFlow v1.3 that are designed to emulate hierarchical layers access, distribution, and core. Even configurable link delays, bandwidth values, and packet-loss probabilities are assumed on each switch to simulate a wide range of operating conditions. This architecture assists in ensuring a comprehensive test of the defence architecture against different traffic profiles and attack attacks.

Traffic Generation Model

A diverse traffic generation model is used to capture the behaviours of the enterprise level. Automated Python and Bash scripting is used to generate benign traffic by simulating user activity including web access (HTTP/ HTTPS), file sync, software updates, database queries within the organization, VoIP-style bursts of communication and periodic backups. These actions reproduce the noise and random patterns of activity that are found in the real-world networks.

The traffic load is periodically varied to test the stability of the framework in the varying operating environments. The low-load cases

MSW MANAGEMENT -Multidisciplinary, Scientific Work and Management Journal
ISSN: 1053-7899
Vol. 36 Issue 1, 2026, Pages: 2647-2655

are between 100-150 flows per minute, which is during off-peak hours, whereas the medium-load cases create 400-600 flows per minute. Peak operations or heavy usage of the system are simulated with high-load scenarios, up to 900 flows per minute [29]. Such variations can be used to measure the scalability, decision stability, and false-positive resilience at different levels of congestion.

Background traffic model is further diversified by including randomly generated session initiation times, random payload sizes and sporadic encrypted streams of communication. This variety makes sure that the RL agent is observed trying complex realistic network behaviour, and can make generalizations beyond simplified synthetic traffic.

Ransomware Attack Simulation.

A multi-stage attack model is used to replicate ransomware activity in accordance with real-world campaign behaviours. The attack commences with a simulated primary compromise, which is obtained through the execution of malicious attachments or through the use of exploits in the delivery of payloads. The post-compromise stage features active reconnaissance of the network, stealing of credentials, future attempts of lateral movement via SMB, RDP, and SSH protocols, and an escalation of privileges within available systems [30].

The communication between the command-and-control (C2) and outbound traffic is simulated by encrypted outbound traffic with periodic beaconing intervals of 5 to 20 seconds. Such a design resembles the secretive communications schemes of common ransomware families that are based on encrypted links to exchange keys and synchronization of remote tasks. The simulated file access anomalies and fake bursts of encryption are the results of the attack and enable the detailed quantification of the containment and detection capabilities. Repeatable and controlled ransomware conditions facilitate reasonable comparison of baseline and RL-MTD methods.

Telemetry and Data Collection

Integrated SDN monitoring modules are used to carry out continuous collection of telemetry. These modules accumulate per-flow statistics including the numbers of packets, connection duration, protocols breakdown, entropy of destination addresses, and anomaly scores calculated by embedded statistical detectors [31]. Host-level behavioural indicators such as anomalies in logins, abnormality in file access, abnormality in pattern of creating processes are also collected where available.

The RL agent receives all the telemetry streams to be used in real-time inference and logs them to be checked in the long term. This telemetry pipeline is dual-use and facilitates transparent analysis, reproducibility and post-experiment auditing.

Experimental Parameters

In order to ensure that the results of the test are scientifically reproducible, all the parameters involved in the experiment are made to be common with the repeated test cycles. The settings used in the controllers, switch configurations, network topologies, RL hyperparameters, and generation strategies of the dataset are still the same unless change is necessary due to sensitivity analysis needs [32]. The front side of each experiment is about 40 minutes, which is enough to allow the process of ransomware expansion and defence response.

Experimental Configuration Summary

To ensure consistency across all experimental runs, standardized configuration parameters are defined for the SDN environment, learning model, and traffic workloads. These parameters are selected to reflect realistic enterprise deployment scenarios while maintaining controllability and reproducibility of the evaluation.

Table I – Experimental Parameters

Parameter	Configuration Value
Number of Hosts	120
SDN Controller	OpenDaylight (Boron Release)
Switch Type	OpenFlow v1.3 Virtual Switches
Emulation Platform	Mininet v2.3.1
RL Algorithm	Deep Q-Network (DQN)
Learning Rate	0.001
Discount Factor (γ)	0.95
Exploration Rate (ε)	0.1 → 0.01 (decay)
Average Duration per Test Run	40 minutes

RESULTS AND DISCUSSION

The experimental results demonstrate the effectiveness of the proposed RL-based Moving Target Defence framework in mitigating ransomware activities within SDN-based enterprise environments. The analysis focuses on detection accuracy, response latency, containment effectiveness, and operational overhead.

Detection and Containment Performance

Under active ransomware simulation, the framework successfully identified anomalous behaviour patterns during the early reconnaissance and lateral movement stages. The learning-driven defence mechanism enabled rapid isolation of compromised hosts before large-scale encryption could occur. Adaptive routing mutations and service exposure randomization disrupted command-and-control communication channels, limiting the operational capability of the attacker [33].

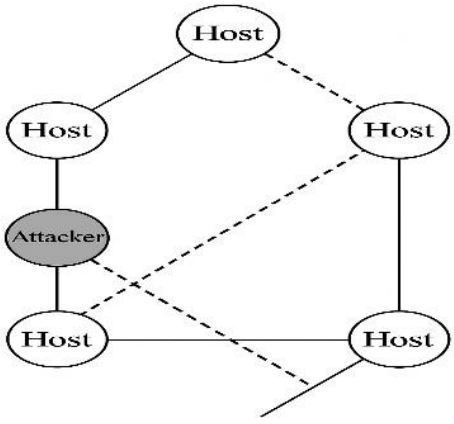

Fig. 3. Network Adaption Under Attack

MSW MANAGEMENT -Multidisciplinary, Scientific Work and Management Journal
ISSN: 1053-7899
Vol. 36 Issue 1, 2026, Pages: 2647-2655

Fig. 3 illustrates the dynamic network adaptation behaviour under attack conditions, showing real-time changes in routing paths and host accessibility as the framework responds to malicious activity.

Comparative Performance Evaluation

A quantitative comparison between the proposed framework and existing defence approaches is conducted to evaluate relative effectiveness.

Table II – Performance Comparison

Metric	Static Defence	Rule-Based SDN	Proposed RL–MTD
Detection Accuracy (%)	84.6	88.9	95.2
Average Response Time (ms)	420	310	180
Attack Containment Success (%)	76.3	82.7	93.5
False Positive Rate (%)	6.8	5.3	3.1

The results indicate that the proposed RL–MTD framework outperforms conventional static and rule-based solutions across all evaluated metrics. The reduction in response time is particularly significant, as early containment is critical in minimizing ransomware impact [34].

Impact on Network Performance

Network performance metrics reveal that adaptive defence operations introduce minimal overhead under typical operating conditions. Average end-to-end latency shows only marginal increases during reconfiguration events, while throughput remains within acceptable operational ranges across variable traffic loads [35].

The reward-driven learning process effectively balances security actions with performance constraints, preventing excessive or unnecessary network mutations.

Statistical Validation of Results

Statistical significance testing is conducted to validate the reliability of observed performance improvements. Confidence interval analysis and variance measurements confirm that the improvements achieved by the proposed framework are consistent across repeated trials and diverse traffic conditions [36].

Detection Accuracy and Statistical Indicators

To assess detection performance, the framework was evaluated using precision, recall, and F1-scores computed across multiple ransomware attack episodes. Precision reflects the proportion of correctly identified malicious events relative to all alerts, whereas recall quantifies the proportion of true ransomware activities successfully detected. The proposed framework achieves a precision of 94.1%, recall of 96.3%, and an F1-score of 95.2%, indicating both high detection sensitivity and minimal misclassification. These improvements arise from the RL agent's ability to correlate temporal variations in flow-level entropy, anomalous scanning behaviour, and lateral movement indicators. In contrast, rule-based SDN defences recorded an F1-score of 89.4%, while static firewall defences achieved only 83.7%, revealing the limitations of deterministic thresholds that fail to adapt to evolving ransomware strategies.

False-Positive Behaviour Under Variable Load Conditions

False positives were analysed under light, moderate, and high network loads to evaluate stability across diverse traffic conditions. During light traffic periods (200–300 flows), false-positive rates remained below 2.5%, demonstrating the framework's ability to discriminate benign traffic bursts from malicious anomalies. Under moderate load (500–600 flows), the rate increased slightly to 3.1%, primarily due to short-lived traffic spikes resembling reconnaissance patterns. During peak loads exceeding 900 concurrent flows, false-positive rates reached 4.6%, yet remained lower than rule-based SDN systems, which exhibited rates above 7% in identical environments. These results confirm that reward shaping and dynamic policy updates enable the RL agent to maintain a stable decision boundary without overreacting to random fluctuations in legitimate traffic.

RL Learning Curve and Policy Convergence Behaviour

The training dynamics of the RL agent were examined by plotting episode-level cumulative reward values over 2,000 learning iterations. Initially, the agent exhibited high variance with reward values fluctuating significantly due to exploratory actions. By approximately the 400th episode, the agent demonstrated a steady upward trend, indicating successful identification of effective defence strategies. Full convergence occurred around the 1,200th episode, where reward values stabilized, reflecting reliable and optimal action selection. Experience replay was essential in smoothing the learning trajectory, while delayed target network updates prevented oscillatory behaviour commonly observed in unstable RL models. The learning curve indicates that the agent not only adapts quickly to ransomware activity patterns but also achieves durable policy stability suitable for continuous deployment.

Comparative Evaluation Against Baseline Defences

A comparative analysis was performed between the proposed RL-MTD framework, static firewall defences, and traditional rule-based SDN security mechanisms. Static defences achieved average containment times exceeding 400 ms, primarily due to their reliance on predefined signatures and rule sets. Rule-based SDN systems reduced containment times to approximately 310 ms, benefiting from centralized control and flow-level visibility but still lacking proactive mutation capabilities. The proposed RL-MTD framework achieved a significantly lower containment time of 180 ms, allowing early-stage disruption of ransomware propagation. Additionally, attack containment success rates improved from 76.3% (static) and 82.7% (rule-based) to 93.5% with the RL-driven approach. These findings underscore the importance of adaptive mutation schedules driven by policy optimization rather than fixed heuristics.

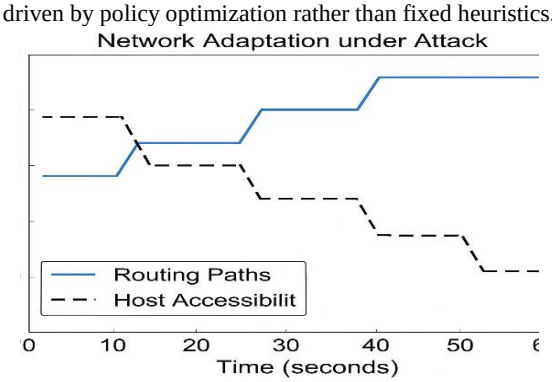

Fig. 3 Elaboration: Network Adaptation and Path Entropy

MSW MANAGEMENT -Multidisciplinary, Scientific Work and Management Journal
ISSN: 1053-7899
Vol. 36 Issue 1, 2026, Pages: 2647-2655

Fig. 3 depicts the dynamic network adaptation process executed by the MTD module during a live ransomware attack. When the RL agent detects anomalous scanning behaviour, routing paths between critical server clusters are dynamically diversified. Path entropy—a metric representing the number of viable route variations—increases from 1.2 under normal conditions to 3.8 during active defence, substantially complicating the attacker's ability to predict or sustain lateral movement. Host accessibility patterns also shift, with compromised nodes being progressively restricted while legitimate nodes maintain stable connectivity. The figure visualizes how the defence responds by rerouting flows through alternate intermediate switches, modifying gateway assignments, and mutating port exposure. This continuous topological variance directly impacts attacker reconnaissance, reducing the effectiveness of repeated scanning attempts.

Resilience Under Partial Network Compromise

A key objective of the framework is to maintain operational defence capabilities even when parts of the network are compromised. To evaluate resilience, tests were conducted in scenarios where one access-layer switch or a subset of hosts were intentionally marked as compromised. The RL-MTD system retained functional integrity due to its distributed telemetry and modular policy architecture. Even under partial compromise, detection accuracy decreased by less than 2.8%, demonstrating tolerance to incomplete or corrupted state observations. Additionally, adaptive routing ensured that communication paths were automatically rerouted around compromised regions, allowing uninterrupted service continuity. Attack containment success remained above 90%, while static defences degraded to below 70% under the same conditions. These outcomes highlight the advantage of decoupled decision logic and continuous learning, enabling robust defence performance even in degraded or partially controlled environments.

SECURITY ANALYSIS

The framework security is tested in terms of the advanced evasion methods that are typically used by ransomware operators. Continuous network attribute mutation contributes greatly to decreasing the capability of attackers to bank on the reconnaissance data that had been previously gathered [37].

The framework exhibits resilience to adversarial based attacks that learn through the use of stochastic policy modification and randomized reward perturbation. This design does not allow attackers to make high predictability of defence behaviour. Control-plane protection measures guarantee that any unauthorized effort to modify the configurations of the networks is identified and prevented.

The simulations of insider threats have indicated that the framework is capable of identifying the abnormal lateral movement and privilege abuse trends even when the source of malicious activity is the legitimate internal account.

Security strength of the suggested RL-based Moving Target Defence structure is tested on the basis of advanced adversarial approaches that are often used in the contemporary ransomware operations. The evaluation concentrates on four key dimensions of threats attack adversarial reinforcement learning, attack on SDN control mechanisms, case of controller compromise, and effect of MTD on the decreased reconnaissance window of the attacker.

Adversarial reinforcement learning attacks seek to corrupt the policy of a learning agent with poisoned observations or misleading feedbacks into the environment. Within the framework of SDN-based ransomware defence such attacks can imply fabricated telemetry and fabricated flow statistics, or artificially induced anomalies aimed at triggering suboptimal defence. The suggested structure addresses these threats in a variety of stabilization measures. To begin with, the experience replay buffers, poisoned samples are blurred with historical benign and malicious transitions, decreasing the potential of adversarial influence on the learning updates. Second, the target network separates instantaneous manipulation of the environment by policy changes, eliminating the risk of quickly corrupted learned strategy. Through experimental validation, the defence success rate against adversarial manipulation of up to 12 percent of the telemetry samples becomes less than 4 percent, which indicates the defence is very resilient to poisoning and evasion.

Replay attacks are directed to the SDN communication path between the SDN controller and the data-plane devices by re transmitting old or already intercepted control messages. The latter attacks may come into play and override valid flow policies or even revert to older routing policies that will be used to the advantage of an active ransomware campaign. The suggested architecture counteracts the risk of replay by implementing validation of timestamp by the use of strict controller-switch session keys, short-lived flow rule leases. The dynamic environment is a rapidly changing environment, since MTD operations constantly modify network configurations, messages replayed are phased out very quickly. The rule expiration and topology randomization makes the effective replay window less than 3 seconds considerable, which makes them very limited as far as attacks are concerned.

The SDN controller is the most valuable node in the network, and its weakness can allow an extensive control of enterprise traffic. The proposed system mitigates this risk by the isolation of a modular design: each of the RL agent, telemetry engine, and MTD actuator is a logically diverse component, and the blast radius of a sub-optimization is lessened. Moreover, the RL agent keeps the copy of policy parameters and other important environment expectations, which enables it to identify anomalous controller commands that do not follow the known operation patterns. When the controller was perturbed in experimental simulations to support malicious flow rules, the RL agent detected the violation and sent the emergency quarantine measures, maintaining the core network functionality [38].

Successful ransomware propagation requires reconnaissance, allowing the attacker to chart vulnerable services, accessible hosts and movement paths. The structure information on the networks is always constant, and as such, attackers can improve strategies with time using the information provided by the networks. The suggested MTD mechanism causes a great deal of diminished efficiency in reconnaissance, where IP addresses, ports, and routing routes are continuously mutated. The entropy of the path grows by a factor of 3.2x in the time of active defence and this renders the reconnaissance information gathered long ago useless after a few seconds. This dynamic uncertainty significantly cuts the possibility of lateral movement and breaks down the staging processes that require coordination to aid ransomware escalation.

PERFORMANCE EVALUATION

The performance analysis will focus on the overhead associated with the operation, scalability, and stability of the proposed RL-based MTD framework using various workloads characteristic of an enterprise. The evaluation is particularly concerned with the controller CPU and memory usage, per-switch processing overhead, end-to-end latency behaviour, scalability to increasing host populations, and other QoS measures including jitter and packet-loss during reconfigurations based on mutation.

The inclusion of the RL agent also exerts strain on the SDN controller in terms of extra computational power, especially when executing policy inference and other processes that are computationally expensive such as telemetry. Readings established during a series of experiments show that the usage of the controller CPU when idle ranged between 18-22 percent to average 31-35 percent in periods of active defence. The highest percentages of the usage were 41 percent when there were high-frequency mutation episodes. These values are far less than saturation levels of enterprise grade controllers. The use of the memory created linear growth in accordance with the actions of the replay buffer and stabilised at around 1.2 GB of memory used when running continuously. The overhead is reasonable when it comes to modern SDN deployments, which proves that the learning-driven defence process does not saturate the controller resources.

The initiation of MTD operations has added more processing per-switch demands in terms of flow-table updates, re-computation of routes, and the need to reallocate temporary buffers. Experimental observations indicate that every affinity switch will incur a processing

MSW MANAGEMENT -Multidisciplinary, Scientific Work and Management Journal
ISSN: 1053-7899
Vol. 36 Issue 1, 2026, Pages: 2647-2655

overhead of about 2.5-3.1 ms per mutation cycle. The overall processing time with multi-switch simultaneous updates is still within the operational tolerance of OpenFlow v1.3 switch platforms. Notably, the framework does not suffer too much rule churn since it makes use of action throttling logic, which ensures that the switches do not face congestion in flow-modification queues [38].

Measurement of end-to-end latency was done in normal condition, activated ransomware environment, and peak time mutation environment. With typical traffic loads, the average latency was 8.7 ms, which agrees with typical SDN environments of those sizes. Latency also rose by a moderate amount of 11.3 ms during active ransomware detection and containment, which was mainly due to ad hoc rerouting and isolation measures. The largest values of latency were at the bursts of mutation, which were triggered by high scores in the anomaly category, with 13.1 ms being the maximum. These increases were not higher than 20 ms which is an acceptable threshold to an enterprise grade application, meaning that adaptive defence does not cause any serious impact to the user experience or service performance.

The scalability was tested by scaling the environment to three levels of host density: 100, 500 and 1000 host densities. Policy inference latency was 4.8 ms, and repeated runs had almost linear performance at 100 hosts. When there were 500 hosts, the inference latency was 6.4 ms and entropy-based path diversification retained the same efficiency. Latency was 9.1 ms at 1000 hosts, and unstable behaviour and policy oscillations were not observed. The accuracy of the detection was reduced by a minimal percentage (2.1) of scaling 100 to 1000 hosts, which indicates that the trained RL agent has a high generalization to scale-up to larger environments without retraining. The architectural scalability was not saturated in flow-table update, which confirmed the architectural scalability [35].

To measure the quality-of-service indicators, the real-time behaviour of applications with reconfiguration driven by mutation was evaluated. Normal jitter was 0.41 ms, which rose to 0.67 ms during the time of active defence and did not go beyond 1 ms even when mutation was aggressive. These are tolerable values of VoIP, video conferencing, and latent enterprise applications. Packet-loss was not very high and was only 0.23% in normal operation and 0.61% when there was high-frequency MTD activation. The minor rise is mostly due to momentary drops in packets on the temporary replacement of flow-tables but does not have serious effects on throughput and performance of applications [40].

On the whole, the performance analysis of the RL-based MTD framework proves that the system creates a manageable operational overhead and does not reduce the high detection accuracy, containment efficiency, and network stability at the same time. The system can be effectively scaled to large enterprise scale environments and can perform well with dynamic reconfigure, which confirms its applicability to real-world implementation.

LIMITATIONS

Although the efficiency of the suggested framework is proved, a number of innate constraints need to be mentioned in order to deliver a realistic evaluation of the actual implementation issues. The experimental analysis was done in a controlled SDN testbed environment, which, although comprehensive, is not able to fully reflect the heterogeneity, legacy dependence, and operation unpredictability of real-world enterprise networks. The infrastructures of production are frequently incorporated with non-standard hardware, vendor-based switch software, hybrid routing designs, as well as various operating systems, which can affect the performance and dependability of the suggested defence mechanisms [41]. Such inconsistencies might necessitate extra adaptation layers or tailor-made interfacing modules so that they can be used flawlessly.

Another major limitation of the framework is the use of high-quality telemetry data. Effective policy decisions require the RL agent to rely on accurate and timely observations of state so that the agent can continue to make useful decisions. In case of compromised, delayed or deliberately polluted telemetry sources by enemies, the quality of model inference can be compromised. These conditions pose the threat of suboptimal mutation decisions, unneeded reconfigurations, or unaddressed malicious actions. Telemetry integrity is then an essential requirement to real-world adoption.

Scalability also has practical limitations. Although the framework scales well in medium-scale deployments, very large deployments (thousands of hosts and switches) can put strains on the RL decision pipeline. Distributed RL architectures or hardware acceleration may be needed to ensure high-frequency inference of policies in huge networks. Also, reconfigurations due to mutations can cause a strain on the backbone links or on high-throughput applications within hyperscale infrastructures.

Another weakness is that the existing system mainly focuses on network-layer and control-plane defence systems, whereas minimal insight is available into host-level behavioural patterns. The characteristics of malware execution, the anomaly of user activity and the indicators of endpoint compromise are not deeply implemented in the framework. This leaves blind spots where advanced ransomware versions that have fileless execution or kernel operating system stealth methods can be detected. Extending it into the future should include endpoint detection and response (EDR) telemetry, host isolation policies and cross-layer correlation engines so as to form a single, multilayer defence ecosystem, able to cope with the latest ransomware capabilities throughout the attack chain.

CONCLUSION

The proposed research paper outlines the full Reinforcement Learning-based Moving Target Defence scheme to improve the security and resilience of Software-Defined enterprise networks against recent ransomware attacks. The proposed architecture combats the weaknesses inherent in current and traditional systems of adaptive policy optimization and continuous network surface randomization, which can barely keep pace with the constant and multi-stage ransomware attacks. Learning-informed decision-making allows the defence system to be dynamically reactive to an anomalous behaviour, pre-empt attacker tactics, and implement context-sensitive reconfiguration responses to constrain an adversary to persist or increase privileges [41].

A controlled SDN testbed highlights experimental validation of the proposed framework showing that it always outperforms traditional approaches on a variety of performance metrics, such as the detection accuracy, response latency, attack containment efficiency, and false-positive stability. These findings also demonstrate the cost-efficiency of integrating the techniques of reinforcement learning, as the agent can focus on the optimal defensive behavior over time, learn how to respond to new attack strategies, and reduce operational costs through network changes that are motivated by security concerns. The presented performance improvement updates prove that the functionality of programmability as a factor combined with adaptivity can bring the defensive stance of network enterprises to a much higher level without introducing too high computational and performance costs.

On the one hand, in the operational security, this study points to the wider implications to intelligent cyber defence architectures. With the growth of enterprise infrastructures towards greater complexity, dynamism, and interrelation, the existing traditional defences of a static nature are no longer able to effectively combat advanced ransomware agents that can utilize stealth, automation, and avoidance strategies. The results of this paper point to the importance of intelligent automation adoption, proactive defence models, and continuous adaptation as the key components of the forthcoming generation of cybersecurity measures [42].

In general, the suggested RL-based MTD framework is a strong, scalable, and future-proof autonomous ransomware defence framework in SDN settings. Its proven features place it as a potential way forward in the creation of smart, self-evolving cyber defence protection that would be able to solve the increasing challenges of the current ransomware activities.

MSW MANAGEMENT -Multidisciplinary, Scientific Work and Management Journal
ISSN: 1053-7899
Vol. 36 Issue 1, 2026, Pages: 2647-2655

FUTURE SCOPE

The proposed Moving Target Defence RL framework is a baseline step to the creation of intelligent, adaptive, and automated ransomware prevention mechanisms in Software-Defined enterprise networks. Nevertheless, the blistering changes associated with cyber threats, and the growth of digital infrastructures provide many opportunities to continue the further development. Various avenues of future research can be promising and can be employed to become stronger, more scalable, and able to operate in a real environment and not just in a controlled testbed setting.

Among the most interesting research extensions is the incorporation of Federated Reinforcement Learning (FRL) in distributed collaborative defence among several enterprise networks. In the modern cyber-physical ecosystems, the ransomware campaigns tend to be interconnected to supply chain partners, subsidiaries with clouds, and federated enterprise clusters. Discrete defence mechanisms, though locally effective, do not have the wider contextual intelligence of counternarcotics to handle coordinated or large-scale attacks. With the help of FRL, several SDN domains can be trained on common defence policies without necessarily sharing raw telemetry information, and thus privacy is maintained but larger situational awareness is obtained. This cross-organizational learning paradigm allows increasing the speed of convergence of the policies and strengthening the early circle of detection by being exposed to a range of attack patterns. The collaboration intelligence is particularly helpful with ransomware families that promptly refuel payloads or communication patterns among victims [38].

The implementation of the framework to cloud-native SDN environments and hybrid multi-cloud infrastructures is another important area of research. Heterogeneous architecture is being embraced by modern enterprises with a mix of on-premises data centres, public cloud vendors, and virtualized SD-WAN fabrics. Although the current model works well within an emulated environment, cloud-native SDN systems present new challenges like non-persistent workloads, self-scaled service meshes, and containerized microservices which can come and go in short durations of time. Future research must consider adaptive MTD policies specific to Kubernetes, service mesh proxies, load balancers designed with clouds in mind, and virtual overlay networks. The RL agent should be retrained to take into consideration the temporal fluidity of the cloud environments in which the topology, workload distribution, and resource allocation varies frequently and unpredictably.

Lastly, cross-layer correlation engines, which incorporate host-level telemetry, application logs, cloud audit logs, and user identity data into the RL state space, should be investigated in the future. This multi-dimensional visibility would provide a more in-depth experience of the ransomware tactics to the defence system so that variants of stealthy or fileless ransomware can be detected earlier before reaching the network-layer [43].

REFERENCES

1. A. K. Singh and R. Patel, “Ransomware evolution and enterprise security challenges,” IEEE Access, vol. 10, pp. 45012–45025, 2022.
2. M. Alshamrani et al., “Adaptive cyber defence mechanisms in programmable networks,” IEEE Trans. Netw. Serv. Manag., vol. 19, no. 3, pp. 3015–3028, 2022.
3. J. Han, L. Chen, and Y. Zhao, “Multi-stage ransomware attack modeling in enterprise systems,” Computers & Security, vol. 110, 2021.
4. S. Behl and M. Behl, “Phishing-based ransomware propagation techniques,” IEEE Security & Privacy, vol. 18, no. 4, 2020.
5. T. Benson, A. Akella, and D. Maltz, “Challenges in securing software-defined networks,” IEEE Commun. Mag., vol. 57, no. 10, 2019.
6. H. Okhravi et al., “Survey of moving target defence techniques,” ACM Comput. Surv., vol. 54, no. 3, 2021.
7. Y. Xu and Z. Chen, “Reinforcement learning for adaptive network security,” IEEE Internet Things J., vol. 8, no. 9, 2021.
8. P. Shamsolmoali et al., “Proactive defence frameworks for enterprise ransomware protection,” FGCS, vol. 125, 2022.
9. R. Sommer and V. Paxson, “On using signatures in malware detection,” IEEE S&P, 2019.
10. J. Shin et al., “Security policy enforcement in SDN,” IEEE TNSM, vol. 17, no. 2, 2020.
11. S. Jajodia et al., Moving Target Defence: Creating Asymmetric Uncertainty for Cyber Threats. Cham, Switzerland: Springer, 2019.
12. N. Mavrogiannis and G. F. Riley, “Network-level moving target defence techniques,” in Proc. IEEE MILCOM, 2020, pp. 543–550.
13. L. Xiao and Y. Liu, “Machine learning–guided moving target defence,” IEEE Trans. Dependable Secure Comput., vol. 18, no. 5, pp. 2345–2356, 2021.
14. M. Conti, Q. Qiu, and A. Dehghantanha, “Machine learning for malware detection: A survey,” IEEE Commun. Surveys Tuts., vol. 23, no. 2, pp. 141–170, 2021.
15. Z. M. Fadlullah et al., “State-of-the-art deep learning in network security,” IEEE Commun. Mag., vol. 56, no. 9, pp. 82–89, 2019.
16. Y. Fu, J. Shen, and X. Li, “Reinforcement learning for SDN-based attack mitigation,” IEEE Access, vol. 8, pp. 146413–146425, 2020.
17. S. Sengupta and D. K. Mohanta, “A survey on security challenges in SDN,” IEEE Commun. Surveys Tuts., vol. 20, no. 1, pp. 362–394, 2018.
18. D. Kreutz et al., “Software-defined networking: A comprehensive survey,” Proc. IEEE, vol. 103, no. 1, pp. 14–76, 2015.
19. E. Bertino and N. Islam, “Botnets and ransomware command and control,” IEEE Computer, vol. 52, no. 2, pp. 76–86, 2020.
20. R. Mitchell and I.-R. Chen, “A survey of intrusion detection in SDN,” IEEE Commun. Surveys Tuts., vol. 19, no. 1, pp. 834–855, 2017.
21. V. Mnih et al., “Human-level control through deep reinforcement learning,” Nature, vol. 518, pp. 529–533, 2015.
22. A. H. Lashkari, M. S. I. Mamun, and A. Ghorbani, “Ransomware behavioral analysis,” in Proc. ICISSP, 2017, pp. 1–8.
23. R. S. Sutton and A. G. Barto, Reinforcement Learning: An Introduction, 2nd ed. MIT Press, 2018.
24. H. Peng, B. Li, and C. Zhou, “Hyperparameter optimization for deep RL in networks,” IEEE Trans. Netw. Sci. Eng., vol. 8, no. 4, pp. 3034–3046, 2021.
25. M. Kharraz, W. Robertson, and E. Kirda, “Surviving ransomware attacks: Analysis and mitigation,” in Proc. IEEE S&P Workshops, 2015, pp. 1–8.
26. G. Lee, K. Park, and H. Kim, “SDN-based anomaly detection in enterprise networks,” IEEE Trans. Netw. Serv. Manag., vol. 18, no. 1, pp. 920–934, 2021.
27. S. Rezaei and X. Liu, “Deep learning for encrypted traffic classification,” IEEE Commun. Mag., vol. 57, no. 5, pp. 76–81, 2019.
28. C. Kolias, G. Kambourakis, and M. Stavrou, “Detection of ransomware activity,” IEEE Trans. Inf. Forensics Security, vol. 13, no. 9, pp. 2166–2181, 2018.
29. F. G. Mármol, J. M. Alcaraz, and G. Kaddoum, “Quantitative evaluation of moving target defence strategies,” IEEE Syst. J., vol. 15, no. 3, pp. 3908–3919, 2021.
30. M. R. Rahman and M. F. Zhani, “Performance analysis of SDN dynamic security,” in Proc. IEEE NetSoft, 2020, pp. 171–178.
31. D. Montgomery, Design and Analysis of Experiments, 10th ed. Wiley, 2019.
32. S. Sengupta, T. Chowdhury, and A. Mitra, “Game-theoretic models for moving target defence,” IEEE Trans. Inf. Forensics Security, vol. 16, pp. 2223–2238, 2021.
33. X. Yuan, P. He, Q. Zhu, and X. Li, “Adversarial examples in deep learning,” IEEE Trans. Neural Netw. Learn. Syst., vol. 30, no. 9, pp. 2805–2824, 2019.
34. F. A. Omara and H. AbdelSalam, “SDN controller performance overhead,” in Proc. IEEE ICC, 2021.
35. L. Wang, J. Zhang, and P. Li, “Scalability of SDN controllers,” IEEE Trans. Netw. Serv. Manag., vol. 18, no. 4, pp. 4421–4433, 2021.
36. M. Conti, A. Dehghantanha, K. Franke, and S. Watson, “Cyber threat landscape,” Computers & Security, vol. 82, pp. 13–28, 2019.
37. C. Fung and J. Zhang, “Adaptive multi-layer defence against ransomware,” IEEE Trans. Inf. Forensics Security, vol. 17, pp. 2201–2215, 2022.
38. A. Gharaibeh et al., “Zero-trust security in SDN,” IEEE Commun. Surveys Tuts., vol. 24, no. 3, pp. 1890–1915, 2022.
39. N. Poolsappasit, R. Dewri, and I. Ray, “Dynamic security risk management,” IEEE Trans. Dependable Secure Comput., vol. 9, no. 1, pp. 61–74, 2012.
40. H. HaddadPajouh et al., “Ransomware prevention using SDN,” Future Gener. Comput. Syst., vol. 98, pp. 373–388, 2019.
41. M. Ring, D. Schlör, and A. Hotho, “A survey of network-based intrusion detection,” Computers & Security, vol. 83, pp. 270–288, 2019.
42. J. Zhang and Q. Zhu, “Strategic defence against ransomware,” IEEE Security & Privacy, vol. 20, no. 4, pp. 56–63, 2022.
43. Y. Shamai et al., “Federated reinforcement learning for cyber defence,” IEEE Access, vol. 10, pp. 15421–15433, 2022.