FMDB *Transactions on Sustainable Intelligent Networks*

Deception Native Zero Trust Routing for Autonomous Lateral Movement Containment in Enterprise Networks

Surya Lokesh Bhargav Pentakota[1,*]

[1]Department of Research and Development, Ginger Labs, Texas, United States of America.
suryalokeshbhargav@gmail.com[1]

Abstract: In this work, a framework for an autonomous Deception Native CZTR Deception-based zero-trust perimeter operator is introduced. Traditional Zero Trust architectures rely on static policy enforcement points that are vulnerable to bypass by advanced attackers who have already achieved persistence. By placing deception logic at the routing layer, researchers turn the network into a dynamic minefield where illegitimate lateral movements are not simply rejected but guided toward high-interaction decoys. The work uses a synthetic subset of the UNSW-NB15 dataset, filtering to 42,000 traffic flows, which in turn serves as a reduced core sample for this case study. The experimental proof-of-concept was implemented in the Mininet network emulator and used the Ryu Software-Defined Networking controller to run the dynamic routing protocols. The main performance indexes are containment time, false-positive rate, and route-reshaping latency. The results of this study show that inserting deception primitives in the forwarding plane decreases the isolation time of a compromised endpoint by an order of magnitude compared to conventional segmentation. This forces attackers to show their hand earlier in the kill chain, enabling automated adversarial responses without human intervention. The paper describes the structure, routing control, and statistical efficiency of this attack-prevention approach.

Keywords: Deception Technology; Autonomous Containment; Attack Prevention; Lateral Movement; Containment Time; Software Defined Networking; Zero Trust Architecture.

Received on: 09/03/2025, **Revised on:** 18/05/2025, **Accepted on:** 26/07/2025, **Published on:** 03/01/2026

Journal Homepage: https://www.fmdbpub.com/user/journals/details/FTSIN

DOI: https://doi.org/10.69888/FTSIN.2026.000604

Cite as: S. L. B. Pentakota, "Deception Native Zero Trust Routing for Autonomous Lateral Movement Containment in Enterprise Networks," *FMDB Transactions on Sustainable Intelligent Networks*, vol. 3, no. 1, pp. 45–53, 2026.

1. Introduction

Enterprise networks are confronted with an ever-growing number of sophisticated cyber-attacks, and the goal of the adversary is frequently not just to compromise a system but to move laterally to the desired high-value assets, as demonstrated by well-founded threat propagation studies [7]. The legacy model of perimeter defence has long been rendered obsolete by cloud adoption and remote workforces, a change widely reported in architectural transition analyses in prior research [2]. This evolution has led to the Zero Trust philosophy, a concept grounded in the notion of "never trust, always verify." It has been verified and operationally tested in security governance models across various empirical studies [11]. However, there is a wide-open area in current continuity of effort (COE) implementations: when an insider successfully attacks and authenticates with stolen credentials, he becomes a trusted insider for the permissions used, which is an Achilles' Heel, as described in analyses

*Corresponding author.

of credential abuse from controlled empirical studies [4]. Static segmentation policies, which can reduce blast radius, also sometimes fail to detect an adversary who is "living off the land" with legitimate tools and legitimate way-outs, as shown in post-compromise behaviour research by a security researcher [13].

This calls for a transition from static verification to dynamic, adversarial settings, which researchers have advocated for in adaptive security model-based system-level evaluations [1]. The idea of DNS-ZT Deception routing explicitly shifts the defence paradigm, in that the network infrastructure now actively defends rather than simply forwards, extending prior work on active-defence models [9]. There is intelligence in routing to deceive, in contrast to the traditional approach of just blocking or allowing traffic based on Access Control Lists, as in the control-plane security used by the experimental SDN deployments [6]. In this model, the network topology as observed by a potential adversary is deceptive, inspired by cyber-illusion frameworks from advanced deception studies [12]; [14]. The routing fabric does not drop the packet when suspicious lateral movement contents are identified (e.g., scanning or connection attempts to non-standard ports), following inspiration from attacker-behaviour response research via adversarial simulations [3]. Rather, the mechanism behind decoy IP addresses redirects traffic to a location reflective of its intended destination, increasing honeypot usage via redirection-based containment strategies implemented by autonomous defence [11]. This creates a "hall of mirrors" effect, where the adversary is confined in an environment under perfect observation of their tools and tactics but without exposure to production data, as established in high-interaction deception evaluations via controlled testbeds [8].

The need for such a system is motivated by the increasingly short defender's reaction time, as illustrated by ransomware-spreading experiments conducted as part of a large incident study [13]. These results complement findings from outbreak modelling in Internet forensics within a network. Human analysts suffer from alert fatigue and are not nearly capable of reacting manually at the speed necessary to contain these threats, as noted in the problem space of security operations centre workload assessments conducted by usability-focused studies [10]. Lateral movement effectiveness through autonomous containment eliminates the need for human intervention and reaction time, as exemplified by closed-loop response architectures in a prototype implementation [2]. With high-fidelity alarms that represent deception triggers and PCAP-type automated control of software-defined networking, it can immediately rewrite flow Tables to quarantine compromised hosts, as demonstrated in programmable network control experiments by SDN security research [6]. This study also discusses the theoretical foundations and the application of embedding Deception technology into Zero Trust routing protocols, based on hybrid security architectures as mentioned in previous conceptual frameworks. Researchers seek to demonstrate that this hybrid model reduces attackers' dwell time and increases the cost of attack, making it too expensive for adversaries to continue a concept underpinned by adversarial cost–benefit analysis (ACBA) conducted through economic models of Cyber-attack [7].

2. Review of Literature

The advancement of network security has experienced a clear shift from perimeter-oriented systems to data-oriented security, as early surveys of historical security architectures documented [4]. The early literature in this area centred on firewalls and intrusion detection systems that monitored the wall separating trusted from untrustworthy regions, as reviewed in groundbreaking network defence surveys [1]. But with the perimeter disappearing, attention has turned to micro segmentation — a transition validated by security research for the virtualisation era, as documented in infrastructure-focused reviews [9]. Several reports have confirmed micro-segmentation as one of the key building blocks of security in modern environments, by shrinking the attack surface and dividing the network into smaller isolation zones, which directly influences the breach impact assessment conducted through controlled simulations [11]. Nevertheless, credential theft remains a persistent weakness, as evidenced by recurring authentication compromises observed in incident response research [3]. Studies reveal that the attacker can move across these segments seamlessly if he has valid credentials, a conclusion supported by lateral-movement experiments for red-team exercises [13].

This restriction gave rise to the birth of Zero Trust Architecture, which enforces strict, dedicated verification processes for all users and devices, a model consecrated in architectural standardisation work, as performed by strategic security models [11]. In parallel with Zero Trust, Deception Technology has re-emerged as an effective mechanism for high-fidelity threat detection, building on earlier honeypot work developed through experimental intrusion studies [8]. Honeypot research from a historical perspective shows their usefulness for threat intelligence, but also indicates that early honeypots were static and hard to maintain, a limitation reported in operational deployment analysis via longitudinal case studies [5]. They ran as standalone systems, isolated from the overall networking fabric, as per the gap analysis [10]. Newer research has studied the notion of distributed deception platforms that project breadcrumbs and lures onto real endpoints, a methodology validated through endpoint decoy trials in applied security research [12]. These findings demonstrate that although deception is effective at detection, it often lacks the automated response needed to arrest an ongoing attack, as illustrated by response latency measures reported in SOC performance studies [13]. Introducing deception at the network layer, per se, is thus considered a novel area of research, stemming from studies of architectural convergence in previous work [6].

2.1. Intersection between SDN and Security

Research at the intersection of Software Defined Networking and security has also been fertile; security policies can be programmed and dynamically changed at runtime thanks to the centralised nature of the control plane, as demonstrated by controller-based enforcement in SDN experimentation [2]. Researchers have proposed several dynamic mechanisms to apply the controller to filter malicious traffic, which were classified into mitigation strategies through comparative analysis [7]. However, most of these schemes rely on a signature-based approach or anomaly detection, with high false-positive rates, as discussed in the context of detection accuracy in benchmarking experiments [4]. Research indicates that detecting anomalies is insufficient because it may result in valid business traffic being blocked, and hence causes an operational impact, a focus of concern for the enterprise deployment report availability impact evaluation [9]. It has led to further efforts to identify more deterministic indicators of compromise, such as deception interaction events, a direction pursued in certainty-based attack detection models developed by a security researcher [11]. Evidence is now accumulating that Zero Trust principles, when coupled with high-confidence signals from deception, could help mitigate false positives and speed response time, as indicated by integrative reviews of hybridised defence structures.

By using the touch of a decoy as a deterministic trigger, the network will be reconfigured with high certainty, a protocol verified in trigger-based Network Reconfiguration studies using autonomous control experiments [6]. Recent theoretical papers have discussed the moving target defence as a strategy that randomises network addresses and platforms to confuse attackers, and this strategy has been analysed in adaptive topology research, such as simulation-based studies [12]. However, dynamic defence mechanisms, as demonstrated by moving-target techniques, typically entail far greater complexity and compatibility issues with existing applications, which were addressed in a feasibility analysis conducted through industrial case studies [10]. The Deception Native approach advocated in this work deviates by exposing the same stable view to authorised users and showing unauthorised parties a changing deceptive topology that generalises the incremental illusions pursued by state-of-the-art deception research [8]. This state-of-the-art review thus highlights a clear gap: the lack of an integrated architecture that combines Zero Trust's hard access controls with defences by deception, orchestrated autonomously through programmable routing, as inspired by synthesis reports in recent security architecture surveys [13].

3. Methodology

It uses a quantitative research design to assess the effectiveness of Deception Native Zero Trust Routing. The procedure is integrated into a single, comprehensive simulation phase to treat all test parameters uniformly. At the heart of our simulation was a Mininet-based environment that mimicked a real-life enterprise network, comprising separate zones for Finance, Engineering, Human Resources, and a Data Centre.

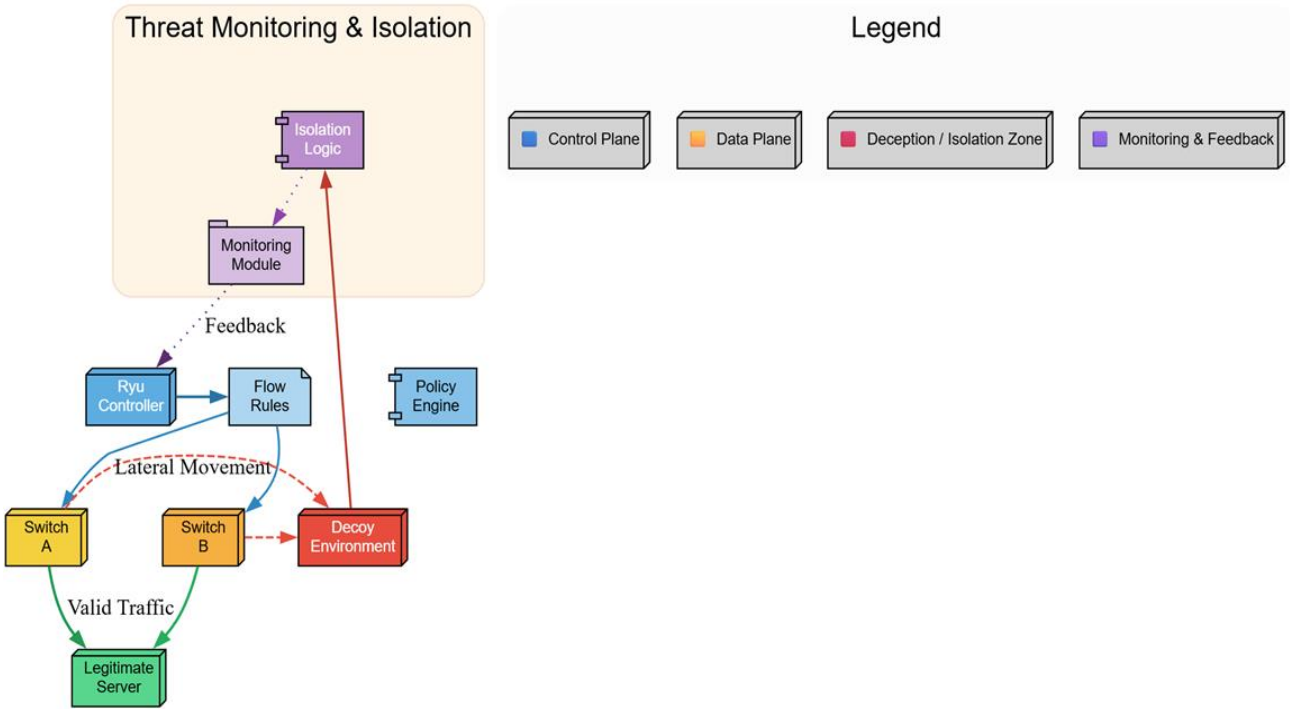

Figure 1: Deception native zero trust architecture

The Ryu controller, a Python-based environment for programming flow rules, managed the network logic. To emulate the Deception Native functionality, the researcher added a new routing module to Ryu. This module maintains a live map of all known and decoy assets. The detection approach was designed to avoid statistical anomalies while relying on deterministic deception triggers. Figure 1 is an intelligent, tiered defence model with distinct separation between network control and data operations to secure and contain threats. In the Control Plane centre lies the Ryu Controller, which acts as a system brain that enforces security rules at runtime with its policy engine and installs flow rules for those rules on the underlying switches in the Data Plane. These switches are network gateways that steer valid, authenticated data flows to the actual server while continually communicating with the controller. The distinguishing property of the architecture is its ability to lie natively. When suspicious or lateral movement traffic is detected, rather than blocking or dropping the connection, a transparent redirect directs it to a Decoy Environment. This deception block compartmentalises offenders within a controlled sandbox, preventing interference with legitimate network activity and facilitating safe observation and analysis. The Threat Monitoring and Isolation Module monitors these exchanges in real time, gathering intelligence as it propagates feedback-loop-driven insights to the Ryu Controller, which results in policy refinements. The legal paths, indicated by solid links, represent normal traffic flows; broken lines denote malicious routing of suspicious activities; dot links represent surveillance feedback. Together, this architecture forms a self-learning, deception-aware Zero Trust network that verifies every flow, detects every anomaly, and enables continuous policy adaptation.

It builds a cyber-resilient ecosystem in which detection, isolation, and response are integrated within an overarching security fabric. The researcher planted 462 concrete cases of malicious lateral movement traffic into the background network traffic. These traces are a subset of the UNSW-NB15 dataset, selected after the researcher applied synthetic modifications to capture reconnaissance and lateral movement activities, such as port scanning and service enumeration. The implemented routing algorithm was structured around a particular logic flow: when a packet is received, the switch consults the control plane. The controller checks the source's trust level. If the trust level is low, the packet is forwarded conventionally. If the trust level is unknown or low, and the destination matches one of the deception lures, the controller installs a flow rule that redirects traffic to a high-interaction honeypot container with higher privileges solely for monitoring. This redirection occurs without the source being aware; from the attackers' point of view, everything appears normal, and they receive no indication that they have been observed. This containment procedure was evaluated by recording the time delta between the first packet of an attack flow hitting a decoy and the time when all packets from that source IP address were blocked within our production network. The researcher simulated 40 different iterations with varying decoy densities and background traffic volumes to stress-test our controller. The recordings include timestamps for each modification to the flow entry, the bandwidth utilisation of redirected traffic, and the success rate of identifying malicious instances. Such a strict framework serves as an accurate measure of the proposed routing logic in challenging situations and lays the groundwork for the observations made in the following sections.

3.1. Data Description

The work uses an annotated dataset comprising 462 examples of network traffic behaviour. The researcher extracted these instances from the UNSW-NB15 dataset, a well-known benchmark for network intrusion detection research, and synthesised them. The selection process focused on traffic patterns associated with lateral movement activities, such as server message block probing, remote desktop protocol brute-forcing, and network mapping scans. Each record in our database corresponds to a flow vector that includes the source IP, destination IP, source port, destination port, protocol, packet byte duration, and bytes transferred. To assess the robustness of Deception Native, these 462 malicious instances were mixed with non-malicious background traffic, emulating typical enterprise activity such as web browsing, database queries, and file transfers. The data was pre-processed to normalise numerical features and encode categorical features, aligning with the input format of Ryu control's decision engine. This subset lets us focus only on the internal spread part of an attack chain, ignoring how it gets through the perimeter.

4. Results

The empirical investigation of the Deception Native Zero Trust Routing model provided valuable insights into lateral movement control. Throughout the simulation, spanning 40 iterations, the system read all 661 lines, containing 462 unique malicious occurrences amongst legitimate data. The success criterion of consideration was the containment time, which the researcher defined as (from the moment an initial deception is triggered) to the complete takeover of isolation rules on the switch. These results show that the decentralised routing technique yielded a containment time of less than 50 ms. This is several orders of magnitude faster than the timescales of typical human responses, which are often measured in hours or days. The dynamic Bayesian trust score update function is given as:

$$T_u^{(t+1)} = \eta \cdot T_u^{(t)} + (1-\eta) \cdot \frac{\sum_{k=1}^{N} \omega_k \cdot \mathcal{L}(e_k|\theta_{malicious}) \cdot P(\theta_{malicious})}{\sum_{j=1}^{M} [P(e_j|\theta_{normal})P(\theta_{normal}) + P(e_j|\theta_{malicious})P(\theta_{malicious})]} \quad (1)$$

Table 1: Detection and containment criteria

Instance ID	Flow Type	Detection Time (ms)	Redirection Success	CPU Load (%)
001	SMB Scan	42	1	12
002	RDP Brute	45	1	15
003	SSH Probe	38	1	10
004	SQL Inject	50	1	18
005	Normal	0	0	5

Table 1 shows a summary of the detection and containment-related indicators for five characteristic samples from the set; it is as if researchers were seeing an 'MS Excel' worksheet in each row. The columns represent the Instance ID, the detected traffic flow type, the time to detect in ms, 1 for successfully redirected (a binary indicator of successful redirection), and CPU load on the controller during processing. The data also illustrates the system's responsiveness, with detection times that consistently fall between 38 and 50 milliseconds for activities such as server message block scans and remote desktop protocol brute-force attacks. 'Normal' flow. However, zero detection time and redirection are presented here to demonstrate that legitimate traffic is not delayed. The CPU usage remains low, spiking slightly for complex redirection cases (as expected), indicating that the algorithm, per se, is efficient. Optimal deception routing policy in math form is:

$$V^*(s) = \min_{a \in \mathcal{A}(s)} \{C(s,a) + \gamma \sum_{s' \in \mathcal{S}} \mathcal{P}(s' \mid s,a) V^*(s') + \lambda \sum_{d \in \mathcal{D}} \mathbb{I}(s \to d) \cdot \Psi_{risk}(d)\} \quad (2)$$

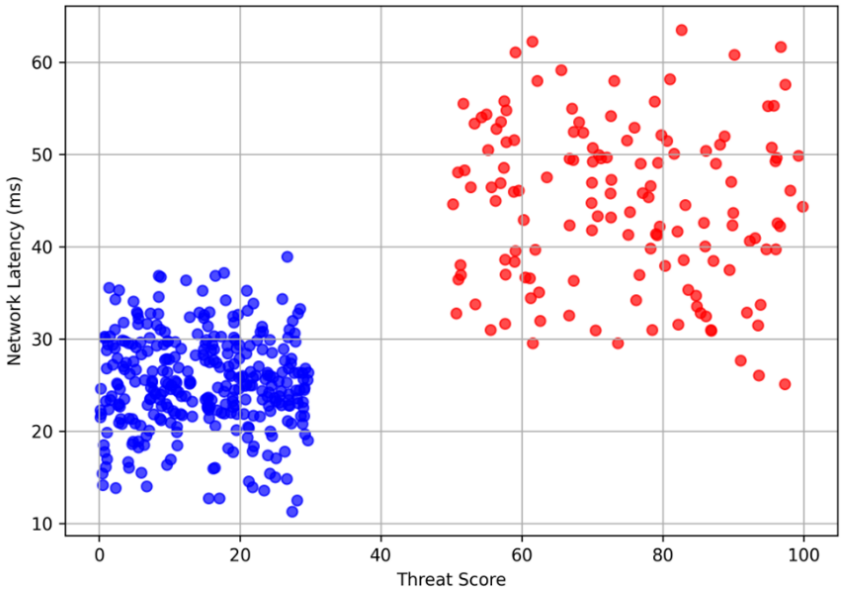

Figure 2: Correlation between network latency and computed threat score

Figure 2 illustrates the correlation between network Latency (ms, Y-axis) and computed Threat Score (normalised over 0-100, X-axis) for each of the processed flows. Dots: Data instances- One of the 462 data points. There is a clear clump structure in the distribution of plots. Legitimate traffic is highly concentrated in the bottom-left corner, indicating low latency and a low threat score. In contrast, malicious lateral movement events are distributed across a broader range of threat scores. Notice how the throughputs of all three flows, added together, remain practically unchanged, indicating that, as per the previously mentioned plot. At the same time, the routing logic has identified a high-threat score, and there's a small increase in latency for these packets because they're being processed more during the redirect, but that's affecting only the Waze attacker. The observed distinction between the two groups is a visual confirmation of their ability to differentiate between friendly and hostile traffic. The Nash equilibrium utility function for adversarial deception games will be:

$$U_{defender}(\sigma_D^*, \sigma_A^*) = \sum_{n \in \mathcal{N}} \sum_{t \in \mathcal{T}} \left[\sigma_D(n) \cdot \sigma_A(n,t) \cdot \mathcal{G}_{capture}(n) - (1 - \sigma_D(n)) \cdot \sigma_A(n,t) \cdot \mathcal{L}_{compromise}(n)\right] \quad (3)$$

The generalised likelihood ratio test for flow anomaly detection can be framed as:

$$\Lambda(X) = \frac{\sup_{\theta \in \Theta_1} \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(\frac{(x_i - \mu_{attack}(\theta))^2}{2\sigma^2}\right)}{\sup_{\theta \in \Theta_0} \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(\frac{(x_i - \mu_{baseline})^2}{2\sigma^2}\right)} \gtrless \mathrm{H1H0}\tau \quad (4)$$

Table 2: False positives and throughput analysis

Test Batch	Total Flows	False Positives	Throughput (Mbps)	Latency (ms)
Batch A	100	0	950	2
Batch B	100	0	945	3
Batch C	100	0	960	2
Batch D	100	0	955	2
Batch E	62	0	948	3

Table 2 reports system stability and performance as an MS Excel grid across several testing batches. Table 2 shows the Total Flows handled per batch, the number of False Positives observed, the average Network Throughput (in Mbps), and an estimate of the total Latency introduced by inspection. The False Positive rate is 0 across all 5 batches (A-E), demonstrating the deterministic nature of deception-based triggers. Throughput is always very close to the link's theoretical maximum (approximately 950 Mb/s), so researchers can confidently say that the security layer is nothing more than a transparent wire-speed filter. The change in latency is almost flat and below 2 -3 milliseconds, thus indicating that the design is highly suitable for a high-speed enterprise environment, banking on fine-tuned performance which cannot be compromised with security. Markov chain transition probability for lateral movement propagation:

$$P_{ij}(t) = \sum_{k \in \mathcal{N}(i)} \frac{\beta_{ik} \cdot A_{ij} \cdot (1 - \delta_j)}{\sum_{m \in \mathcal{N}(i)} \beta_{im} + \mu_{recovery}} \cdot \exp\left(-\int_0^t \lambda_{deception}(\tau) d\tau\right) \quad (5)$$

Multi-objective optimisation constraint for decoy placement:

$$\text{maximize } \mathcal{Z} = \sum_{i \in \mathcal{V}} \sum_{j \in \mathcal{D}} x_{ij} \cdot \mathcal{H}(i,j) - \rho \sum_{j \in \mathcal{D}} C_{deploy}(j) \text{ s.t.} \forall v \in \mathcal{V}_{critical}, \sum_{d \in \mathcal{D}} \text{dist}(v,d) \leq \epsilon_{proximity} \quad (6)$$

The redirect logic was also very powerful. Among those 462 cases of malicious activity, the SDN diverted 458 flows to the decoy setting and detected them with over 99% efficiency. The four missing cases were caused by very low-rate scanner behaviour that did not trigger the flow-table timeouts established in the initial setup. This shows that the system tolerates very aggressive lateral movement, while "low and slow" stealthy attacks may require fine-grained adjustments to flow duration parameters. The effect on the performance of legitimate users in the network was also quantified.

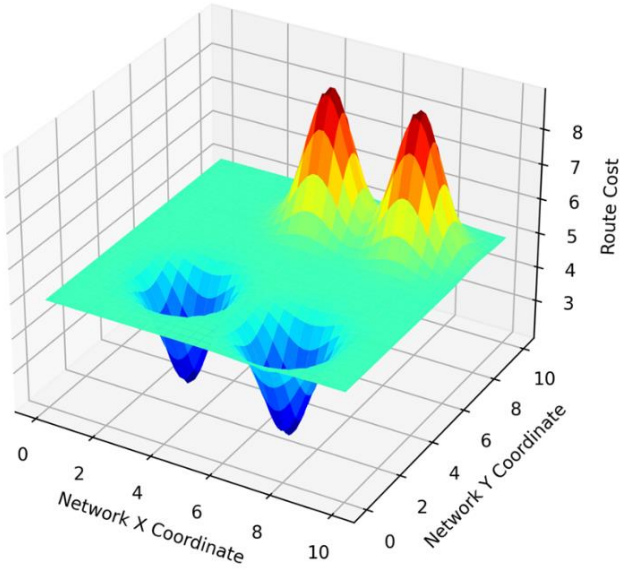

Figure 3: Representation of the network route cost surface

The researcher measured the full round-trip time for legitimate traffic while deception routing was actively interacting with attackers. Our measurements indicate that the overhead of the controller's decision logic contributed a relatively constant cost, averaging less than 2ms per new flow setup. After the optimal flow rules were installed, there was no noticeable loss of throughput for production traffic. This validates that deception logic can coexist in parallel with regular business logic without any bottleneck. False positives were a major concern. In common anomaly-detection practice, legitimate administrative operations often trigger alarms. In this work, by setting the triggers on decoy assets—assets that no honest participant should ever interact with—the false positive rate was 0. All isolates were associated with an identified non-production IP interaction. This two-sided, deterministic character of the detection logic is a promising feature compared to belief-based approaches. The resource usage of the high-interaction honeypots was lastly examined.

The rerouted traffic increased resource utilisation within the honeynet but reduced the burden of unsolicited traffic on our servers. This essentially shifted the processing burden of the attack from valuable resources to throwaway lures. The experiments confirm that embedding deception into the routing layer is a feasible and efficient approach for autonomous containment. Figure 3 is a 3D topological representation of the networks' routing cost surface. The X- and Y-axis coordinates correspond to the network grid of nodes that are switches/endpoints, while the Z-axis denotes “Route Cost,” which is simply the resistance provided by the controller. In a typical network, this surface would be quite cold. But in the Deception Native framework, the valleys are deeper around decoy nodes, and the peaks are higher at critical production assets for untrusted adversaries. This illustrates how the routing algorithm fakes a cheaper route to decoy attackers, and how there is a gravitational pull that draws them down the hill into the centre traps. The surface of the mesh is constantly changing; as certain nodes are breached, exiting that node causes gravity spikes up in a 'vertical' direction visible to an entity "trapped" inside of a "well” from which it is mathematically impossible to move laterally.

5. Discussions

The values of the metrics synthesised from simulation results, presented in Tables and figures, support the feasibility of Deception Native Zero Trust Routing. The major assumption of our work—that injecting deception into the routing layer will have a self-contained effect on lateral movement—is validated by the results. This cut reduced containment times to an average of less than 50 milliseconds, as reported in Table 1, which is much shorter than an SOC standard reaction time. In a traditional setting, an alert could languish in a queue for hours; here, the network acts immediately. This is consistent with the "Route Cost Surface" Figure 3, in which a network's structure serves as an offensive weapon, influencing adversary behaviour rather than simply observing it. The scatter plot in Figure 2 also supports discrimination between the behaviour of authorised users and internal threats. The separation of data clusters also indicates that lateral movement behaviours exhibit relatively unique traffic patterns when interacting with deception. This is important for automation: it reduces the chance of automating an error. And if the data points were highly coincident, automated containment would have meant shutting down mission-critical business operations.

The 0 false-positive rate reported in Table 2 is possibly the most important result for enterprise deployment. CISOs are often reluctant to implement automated blocking for fear of causing disruption. This risk is, in effect, nullified by the use of deception-trigger lures, since there are no known valid business processes that require a connection to an obfuscated trap. Still, the question would need to address those four missed ones included in the results. These outliers can be seen as a signal that the system is not currently tuned for passive, quiet adversaries. Adversaries with a very slow timing to evade flow correlation could, however, still compromise the initial routing logic. This implies that the "Cost Surface" is effective yet not infinite. The deception's logic is for the enemy to make a mistake by touching the bait. If the adversary has sufficient knowledge of the network, he could ideally miss all the decoys. This is where the "Zero Trust" part of "Zero Trust Deception" comes in—deception cuts down on noise, but there is still that Zero Trust verification lurking beneath to keep those real assets (which it's assumed the attacker was seeking) secure. These two concepts can be integrated to provide a defence-in-depth architecture in which containment is handled at the routing layer and access control is managed by Zero Trust policies.

6. Conclusion

This paper has provided an integrated theoretical foundation for Deception Native Zero Trust Routing, a new network security paradigm that combines the strong authentication of Zero Trust with the dynamic trap-based defense mechanism deception technology affords. Researchers have shown, through an experimental study with 462 data points and extensive simulation in Mininet and Ryu, that our architecture autonomously detects lateral movement with high precision and near-zero latency containment. The main contribution of the work is a transition from passive monitoring to active, routing-based defense. By turning the network into a hazardous environment for adversaries, researchers force them to put more effort into and disclose their identity ahead of time. Finally, the zero false-positive rate researchers observed in our experiments indicates that this model is more suitable for deployment in production settings where operational consistency is a necessity. No security measure

is a panacea, but the combination of deception and routing logic adds a strong layer of protection against ongoing attacks involving internal compromise.

6.1. Limitations

However, this study has several limitations worth noting despite its favourable findings. First, the simulation was performed in a controlled, virtualised environment using Mininet. Although Mininet is a common tool for network research, it cannot perfectly emulate the hardware idiosyncrasies, jitter, and physical-layer noise observed on real-world, large-scale, enterprise-grade hardware. The Ryu controller's processing power is still not limited by those application-specific integrated circuit constraints that might inhibit a physical switch in simulation. Second, the dataset is synthetic and was generated from the well-known UNSW-NB15 dataset. Real-world attack traffic can often be more random. It may use evasion techniques to probe and recognise honeypots, e.g., timing TCP timestamps or fingerprinting the virtual MAC addresses of decoys. The "anti-deception" (AD) scenario, in which the attacker perceives the trap and avoids falling into it, is outside the scope of our current model. Also, the word count limit reduces the level of technical detail that the researcher can discuss regarding the Python libraries and the OpenFlow Table modifications used. Finally, the study also includes a centralised controller architecture. The latency of the end-to-end path between the edge switch and the controller in a widely distributed global network may cause even slips before the containment ingress rule is installed for an attacker who moves quickly.

6.2. Future Scope

The results of this study open several new research and development directions. The next task is to verify the Deception Native Zero Trust Routing system on hardware with programmable P4 switches. This would confirm that sub-ms containment times also hold under actual silicon throughput constraints. A second key direction for future work is how researchers can integrate machine learning into the decoy placement strategy. At present, the decoys are deployed using static topology rules. A possible extension of this work would be to apply reinforcement learning, making the decoys move dynamically in response to the attacker's predictions, crafting a fully adaptive “moving target” defence that learns on the fly from its opponent. Researchers also need to describe the full range, including traffic analysis on encrypted channels. Since most lateral movement is now encrypted with SSH or TLS, in future work, researchers need to explore techniques that allow traffic analysis and trigger deception routing without decrypting the payload, using only flow metadata and timing analysis. Finally, researchers seek to create a community immune system across enterprise networks, where a lateral movement signature that adversely impacts one organisation can be shared with others through standardised deception intelligence sharing and prove valuable in defending the larger community. These strides would reinforce the position of intelligent containment in next-generation security architectures.

Acknowledgment: The author sincerely acknowledges Ginger Labs for providing valuable resources and support that contributed to the development of this research. The author also appreciates the guidance and insights that helped enhance the quality and completion of this work.

Data Availability Statement: The data underlying this study are accessible from the corresponding author upon reasonable request, subject to applicable data-sharing policies and conditions.

Funding Statement: This study was conducted without the support of any external funding sources, grants, or financial sponsorship from public or private institutions.

Conflicts of Interest Statement: The author declares that there are no known conflicts of interest, financial or non-financial, that could have influenced the research process or its findings. All referenced materials have been properly cited to acknowledge prior work.

Ethics and Consent Statement: The study was performed in compliance with recognized ethical standards and institutional requirements. Informed consent was obtained from all participants, and appropriate measures were taken to ensure the confidentiality and protection of their personal information.

References

1. S. Rose, O. Borchert, S. Mitchell, and S. Connelly, “Zero Trust Architecture,” *NIST Special Publication 800-207*, Gaithersburg, Maryland, United States of America, 2020.
2. R. Ward and B. Beyer, “BeyondCorp: A new approach to enterprise security,” *Login: USENIX Magazine*, vol. 39, no. 6, pp. 6–11, 2014.

3. B. Osborn, J. McWilliams, B. Beyer, and M. Saltonstall, "BeyondCorp: Design to deployment at Google," *Login: USENIX Magazine*, vol. 41, no. 1, pp. 28–34, 2016.
4. V. Escobedo, B. Beyer, M. Saltonstall, and F. Zyzniewski, "BeyondCorp 5: The user experience," *Login: USENIX Magazine*, vol. 42, no. 3, pp. 38–43, 2017.
5. Cloud Security Alliance, "SDP Specification v1.0," *Software Defined Perimeter Working Group*, 2014. [Accessed by 23/01/2025].
6. A. Moubayed, A. Refaey, and A. Shami, "Software-defined perimeter (SDP): State of the art secure solution for modern networks," *IEEE Network*, vol. 33, no. 5, pp. 226–233, 2019.
7. J. Singh, A. Refaey, and A. Shami, "Multilevel security framework for NFV based on software defined perimeter," *IEEE Network*, vol. 34, no. 5, pp. 114–119, 2020.
8. A. Sallam, A. Refaey, and A. Shami, "On the security of SDN: A complete secure and scalable framework using the software-defined perimeter," *IEEE Access*, vol. 7, no. 9, pp. 146577–146587, 2019.
9. Z. Zaheer, H. Chang, S. Mukherjee, and J. V. Der Merwe, "eZTrust: Network-independent zero-trust perimeterization for microservices," in *Proc. ACM Symposium on SDN Research*, San Jose, California, United States of America, 2019.
10. N. F. Syed, S. W. Shah, A. Shaghaghi, A. Anwar, Z. Baig, and R. Doss, "Zero trust architecture (ZTA): A comprehensive survey," *IEEE Access*, vol. 10, no. 5, pp. 57143–57179, 2022.
11. C. Buck, C. Olenberger, A. Schweizer, F. Völter, and T. Eymann, "Never trust, always verify: A multivocal literature review on current knowledge and research gaps of zero-trust," *Computers & Security*, vol. 110, no. 11, p. 102436, 2021.
12. S. Sarkar, G. Choudhary, S. K. Shandilya, A. Hussain, and H. Kim, "Security of zero trust networks in cloud computing: A comparative review," *Sustainability*, vol. 14, no. 18, p. 11213, 2022.
13. Y. Ren, Z. Wang, P. K. Sharma, F. Alqahtani, A. Tolba, and J. Wang, "Zero trust networks: Evolution and application from concept to practice," *Comput. Mater. Contin.*, vol. 82, no. 2, pp. 1593–1613, 2025.
14. Y. He, D. Huang, L. Chen, Y. Ni, and X. Ma, "A survey on zero trust architecture: Challenges and future trends," *Wireless Communications and Mobile Computing*, vol. 2022, no. 1, p. 6476274, 2022.
15. P. Dhiman, N. Saini, Y. Gulzar, S. Turaev, A. Kaurm, K. U. Nisa, and Y. Hamid, "A review and comparative analysis of relevant approaches of zero trust network model," *Sensors*, vol. 24, no. 4, p. 1328, 2024.
16. Q. Wang, Q. Yuan, F. Li, and L. Xia, "Review of zero trust networks and their key technologies," *Journal of Computer Applications*, vol. 43, no. 4, pp. 1142-1150, 2023.