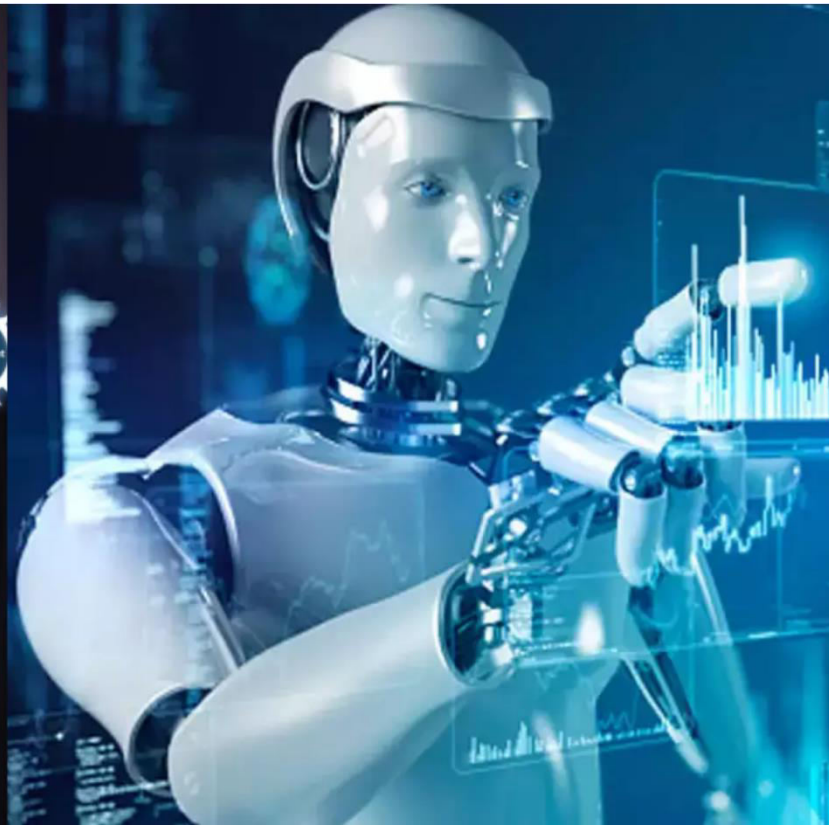
IJIRSET
International Journal of Innovative Research in Science | Engineering | Technology

ISSN(O): 2319-8753
ISSN(P): 2347-6710

International Journal of Innovative Research in Science Engineering and Technology (IJIRSET)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)



Impact Factor: 8.699

Volume 15, Special Issue 1, January 2026

ISSN (Online) : 2319 - 8753
ISSN (Print) : 2347 - 6710

International Journal of Innovative Research in Science, Engineering and Technology

International Conference on Emerging Technologies and Pharmaceutical Research (ICETPR 2026)

17–18 January 2026

Indala College of Engineering, Polytechnic & Pharmacy, Kalyan, MH, India

|Volume 15, Special Issue 1, January 2026|

Deceptive Cyber Defense using Honeypots for Attack Monitoring & Mitigation

Asst. Prof. Gauri Bhosale[1], Ojasvi Telang[2], Divesh Patare[3], Tanavi Pendurkar[4]

Asst. Prof., Dept. of Computer Engineering, Indala College, Mumbai, India[1]

Dept. of Computer Engineering, Indala College, Mumbai, India [2]

Dept. of Computer Engineering, Indala College, Mumbai, India[3]

Dept. of Computer Engineering, Indala College, Mumbai, India[4]

ABSTRACT: This manuscript sketches a security design that tries to feel alive rather than stiff. Instead of stacking tools in a neat but disconnected row, it blends layered defense, deceptive traps, intrusion monitoring, careful human judgment, and a rhythm of continuous feedback. The older habits of the industry and the gaps left by academic studies hover in the background while the proposed design steps forward with a more grounded attitude. Honeypot activity shapes intrusion rules, not as a gimmick but as a stream of raw experience that keeps the system from growing stale. A small two-list structure keeps decisions crisp: one group for what must be stopped, another for what can be ignored, and everything unfamiliar moves toward examination inside controlled decoys. TLS flows meet a proxy that opens the traffic for inspection, letting the IDS see what would otherwise pass unseen. The intent is to build something people can actually run without wrestling with proprietary bloat or labyrinthine wiring, something that grows steadily and adapts without drama. [1] [6] [9] [10] [35] [37] [40] [41] [3] [5] [14] [28] [13] [21] [20]

KEYWORDS: Layered defense, intrusion monitoring, honeypots, deception systems, machine learning elements, network zoning, governance loops, signature lifecycle, TLS termination, Suricata, OPNsense, T Pot.

I. INTRODUCTION

1. Purpose of this Document

This document tries to gather scattered ideas about modern network defense and shape them into something useful and practical. Instead of marching through rigid definitions, it offers a sense of direction. The goal is to show why such an architecture matters, what motivates it, and what it should accomplish before anyone calls it successful. While the text touches on technical details, it behaves less like a dry instruction manual and more like a guidebook you keep nearby while navigating an unfamiliar trail. Every section leans toward clarity rather than perfection, and the writing wanders occasionally to keep the subject from feeling like a chore. The message underneath remains steady: define the aim clearly, understand the reasoning behind each choice, and stay honest about what the system should deliver.

1.1 Scope and Audience

The scope stretches across theory, field practice, academic influence, and hands on implementation. It avoids drifting into pure theory and at the same time refuses to reduce everything into oversimplified checklists. It looks at layered defense, real world pain points, academic blind corners, and the structure of an implementable model.
[35] [37] [40] [41]

Readers may come from different positions. A researcher might look at this and notice ways their ideas could be tested in an environment closer to reality. A security practitioner might treat it as a guide to assembling an open source stack that works without feeling brittle. A decision maker, possibly less interested in packet minutiae, may look for clarity about cost, strategy, and long term sustainability. Each group sees the same text from a different angle, but the intention is that none should feel out of place.

1.2 Structure of the Document

The flow of the document builds upward, beginning with the fundamentals of strong network defense and the ways one can measure whether a design works as promised. After that come the academic threads: where they shine, where they

ISSN (Online) : 2319 - 8753
ISSN (Print) : 2347 - 6710

International Journal of Innovative Research in Science, Engineering and Technology
International Conference on Emerging Technologies and Pharmaceutical Research (ICETPR 2026)
17–18 January 2026
Indala College of Engineering, Polytechnic & Pharmacy, Kalyan, MH, India

|Volume 15, Special Issue 1, January 2026|

drift away from the trenches, and where they leave questions unanswered. The narrative then shifts toward the problems found in actual deployments before settling into the core architecture being proposed. That portion travels through logic, components, and the rhythm of governance. [35] [41]

Later sections explore scaling paths, governance patterns, and evaluation methods that make sense in both technical and managerial contexts. The closing thoughts pull together the scattered points and nudge the reader toward refining, extending, and using the model in the wild. It ends with an open door rather than a full stop, because a security design that pretends to be finished is already aging. [35] [41]

IIPRINCIPLES OF GOOD NETWORK SECURITY IMPLEMENTATION

1.1 Defense in Depth as a Philosophy

Strong defense rarely comes from a single mighty shield. It comes from layers placed intentionally, each with a purpose that complements the others. Imagine several rings around a valuable center. If one ring fails, the others do not collapse with it. This layered mindset accepts that every barrier can eventually crack, and instead of pretending otherwise, it builds redundancy and breathing room. Human judgment, firewalls, policies, monitoring tools, and alerts all play roles. Instead of chasing fantasies of perfect protection, the aim becomes endurance. Slow the attacker. Limit the spread. Buy time. Keep the environment predictable when chaos knocks on the door. The approach feels pragmatic rather than dramatic, almost like choosing sturdy boots instead of ornate armor.

1.2 Key Pillars: Prevention, Detection, Response, Recovery

A mature security setup rests on four pillars that shape everyday behavior.

Prevention

These measures try to stop trouble at the first fence. Firewalls, strict access rules, sane configurations, and timely patches reduce the number of openings an attacker can probe. In practice it is less about declaring absolute safety and more about shrinking the number of ways a mistake can turn into damage.

Detection

Even with strong preventive habits, something slips through eventually. Detection tools watch for the strange and the subtle. Logs, patterns, correlations, and alerts catch hints that something is off. Breadth matters. Depth matters. Curiosity matters even more, because sometimes the odd signal hides beneath ordinary traffic.

Response

When trouble shows itself, speed becomes its own shield. Automated blocks, quick rule updates, and coordinated actions help contain the situation before it spreads. Response is where planning meets reality. A calm hand here matters more than glamorous tools.

Recovery

Despite all effort, impact cannot always be avoided. Recovery restores services, validates integrity, and reviews lessons. It brings back trust and order. The process is both technical and human, because confidence needs rebuilding just as much as servers do.

Together these pillars form a loop. Prevent, detect, respond, restore, then refine the previous steps. The cycle repeats quietly in the background, keeping the system from drifting into complacency.

1.3 Measuring Security Effectiveness: Parameters and KPIs

Security cannot rely on confidence alone. Numbers help reveal whether a system performs well or simply feels like it does.

Detection Accuracy

Precision tells you how many alerts truly matter. Recall shows how many real attacks were caught before they caused harm. Too many false alarms pull analysts into a fog where important details are lost. Too many misses leave cracks unnoticed.

ISSN (Online) : 2319 - 8753
ISSN (Print) : 2347 - 6710

International Journal of Innovative Research in Science, Engineering and Technology

International Conference on Emerging Technologies and Pharmaceutical Research (ICETPR 2026)

17–18 January 2026

Indala College of Engineering, Polytechnic & Pharmacy, Kalyan, MH, India

|Volume 15, Special Issue 1, January 2026|

Time to Detect and Time to Block
Speed shapes the outcome. The quicker suspicious activity is noticed, the fewer steps an attacker can take. The shorter the block time, the smaller the window for escalation.

Resource Efficiency
A hungry security tool can strangle the very platform it is supposed to protect. Balanced use of memory, CPU, and bandwidth keeps defenses from turning into hidden bottlenecks.

Governance and Audit Trails [35] [41]
A good system leaves a clear record of change. Who approved a rule, when it was added, when it expired, and why it exists in the first place. Without transparency, trust fades and people hesitate to rely on the system.

Resilience and Ability to Grow
A design that cannot endure stress is fragile. A design that cannot scale is temporary. A strong architecture bends without breaking and grows without losing its shape.

Put together, these ideas form a picture of security as a living structure. Something that evolves, measures its own performance, learns from mistakes, and gains strength from iteration.

STATE OF ACADEMIC RESEARCH

1.4 Current Focus Areas in Academia

Research circles continue to explore deceptive systems that lure intruders with convincing illusions. Honeypots have evolved from basic traps into entire staged environments that mimic real networks with surprising depth. Scholars examine not only what malicious actors drop into these systems, but also how they move, what they attempt, and how long they persist when they believe they have found something valuable. The study of deception grows more adventurous each year, almost like watching a theater where the actors do not know they are performing.
[1] [6] [9] [10]

Another area centers on the creation of attack signatures without endless manual effort. Many researchers try to build engines that can read captured traffic, observe harmful behavior, and carve out patterns that become security rules. Sometimes the systems use simple rules. Sometimes they use complex pattern recognition. The shared goal is to help defenders turn raw observations into rules that are ready for tools like Suricata or Snort without weeks of human labor. [11] [15] [17] [33]

Academic interest also leans heavily toward learning systems that recognize unusual behavior. Instead of searching for explicit signatures, these models study normal behavior and then alert when something strays from the expected. From modest statistical approaches to deep learning ensembles, experiments attempt to detect stealthy intrusions that hide behind everyday traffic. Yet the same challenge appears repeatedly. These systems may become too sensitive and drown analysts in alerts, or they may become too cautious and let threats pass quietly. [11] [15] [17] [33] [13] [21] [20]

A fourth area blends human judgment with machine capability. Automation can be fast but can also misread context. This leads researchers to propose mixed workflows where analysts review, adjust, or cancel automated suggestions. Such approaches try to keep context, reason, and adaptability in the loop. Human intuition remains essential, even in environments where algorithms do most of the heavy lifting.

1.5 Key Findings and Contributions

Academic work has produced several meaningful lessons. Deception systems stretch out the time intruders remain engaged, giving defenders more opportunities to observe. Automated signature creation shortens the delay between seeing a threat and blocking it. Learning based systems recognize patterns that simple signatures would never notice, although they must be handled carefully to avoid being misled. When human reviewers stay involved, false alarms drop and trust in the system rises. These findings show consistent progress toward defenses that respond smoothly and observe with greater clarity. [1] [6] [9] [10] [11] [15] [17] [33]

ISSN (Online) : 2319 - 8753
ISSN (Print) : 2347 - 6710

International Journal of Innovative Research in Science, Engineering and Technology

International Conference on Emerging Technologies and Pharmaceutical Research (ICETPR 2026)

17–18 January 2026

Indala College of Engineering, Polytechnic & Pharmacy, Kalyan, MH, India

|Volume 15, Special Issue 1, January 2026|

1.6 Open Datasets and Benchmarking Efforts

Despite the progress, one issue returns again and again. Many studies rely on old datasets that no longer resemble the busy realities of modern networks. Traffic today mixes encrypted flows, cloud services, mobile devices, and rapid bursts of application calls. Older datasets, though convenient, cannot capture these complexities. A few initiatives attempt to release more current datasets that include encrypted exchanges, device traffic, and cloud centered patterns. Competitions and shared platforms also try to test algorithms with more realistic scenarios. [3] [5] [14] [28]

Still, the shortage of varied and modern data makes it hard for promising academic models to prove their strength in genuine environments. Without realistic evaluation grounds, models may appear strong in controlled experiments yet fall short when introduced to production networks. The need for shared, continually updated data grows louder, almost like a background hum in every research discussion.

II. CHALLENGES IN CURRENT NETWORK SECURITY PRACTICES

2.1 Fragmentation and Compartmentalization of Tools

The modern security environment often looks like an assortment of mismatched parts. Firewalls stand alone. Intrusion systems stand alone. End point agents stand alone. Logging dashboards float somewhere else entirely. The pieces rarely speak to each other in a natural way. Analysts dart between screens, trying to stitch together a picture that never fully aligns. This scattered structure creates shadows where intruders slip through, and it slows everything from alert handling to routine troubleshooting. [13] [21] [20]

2.2 Excessive False Positives and Alert Fatigue

Security teams frequently face an overwhelming stream of alarms that do not point to real danger. When a tool is too sensitive, it becomes noisy. When it is too relaxed, it misses threats. Analysts spend their days sifting through repeated messages trying to find the few that matter. After long enough, even serious alerts blend into background clutter. Fatigue takes hold, and the system begins to lose its ability to notice the important signals.

2.3 Encryption Blind Spots

The growing use of encrypted traffic strengthens privacy but leaves defenders without visibility. TLS, encrypted DNS, and QUIC turn large portions of network flow into unreadable blocks. Intrusion systems that rely on inspecting content cannot see what hides within these encrypted containers. Without creative methods of gaining controlled visibility, defenders are left guessing. Attackers use this blind space to move quietly while defenders watch only the outlines of what passes by. [3] [5] [14] [28] [13] [21] [20]

Scalability Bottlenecks in Inline Intrusion Systems [13] [21] [20]
Inline inspection tools carry a heavy burden. They must examine everything without slowing anything down. As networks grow faster and more crowded, these systems strain under the load. Large encrypted packets, complex protocols, and constant bursts of traffic can overwhelm processing capacity. When that happens, packets drop or inspection logic lags behind. The defender is forced to choose between clarity and performance, a choice that should never be necessary but appears often. [3] [5] [14] [28]

2.4 Disconnect Between Academic Models and Operational Reality

Many academic papers present creative solutions. Some introduce unusual algorithms. Others reshape familiar ideas into new forms. Yet many of these models never see the turbulence of real world environments. Production networks contain inconsistencies, legacy systems, odd traffic spikes, and unpredictable behavior that no controlled experiment can fully capture. As a result, tools that look impressive on paper may struggle when exposed to the messiness of daily operations.

These challenges do not arise from a lack of imagination. They come from friction between scattered tools, noisy alerts, encrypted obscurity, scaling limitations, and a gap between theory and practice. Until these gaps narrow, defenders will keep wrestling with systems that feel stretched thin and occasionally brittle. [3] [5] [14] [28]**WHERE ACADEMIA FALLS SHORT**

2.5 Lack of Reproducible End to End Pipelines

Academic studies often present bright ideas supported by graphs and evaluation tables. Yet many omit the working code, real datasets, or instructions needed to reproduce the results. Without these details, practitioners cannot rebuild the model

ISSN (Online) : 2319 - 8753
ISSN (Print) : 2347 - 6710

International Journal of Innovative Research in Science, Engineering and Technology
International Conference on Emerging Technologies and Pharmaceutical Research (ICETPR 2026)
17–18 January 2026
Indala College of Engineering, Polytechnic & Pharmacy, Kalyan, MH, India

|Volume 15, Special Issue 1, January 2026|

in a field environment. The ideas remain attractive but impractical, like blueprints missing key pages.

2.6 Insufficient Emphasis on Governance and Rule Lifecycle [35] [41]
A frequent oversight in academic work is the lack of attention to how rules live, change, and expire. Many papers focus entirely on detection performance but ignore questions like who reviews a rule, how its relevance is tracked, or when it should be retired. Rules age just like software. They become noisy, outdated, or misleading. Governance is not glamorous, but it is essential for any system that intends to survive outside a lab. [35] [41]

2.7 Limited Validation on Modern Traffic
Real networks today generate complex flows that include multiplexed protocols, encrypted exchanges, and high velocity application calls. Many academic experiments still rely on datasets that reflect none of this. Models trained on outdated corpora stumble when introduced to modern traffic. Promising techniques may collapse when confronted with encrypted DNS or dense API traffic patterns. [3] [5] [14] [28]

2.8 Weak Integration with Existing Open Source Tools
Organizations rely heavily on well established platforms such as Suricata, Snort, Zeek, OPNsense, or ELK. Academic proposals often appear as isolated prototypes written in uncommon languages or frameworks. They rarely integrate naturally with these widely used systems. Without compatibility, adoption becomes difficult. Even strong ideas are left unused because they cannot fit into the existing environment. [11] [15] [17] [33]

2.9 Failure to Deliver Deployable Models
Many academic contributions stop at theory. They offer detection rates but not deployment instructions. They describe performance but not throughput under stress. They present algorithms but not how to fit them alongside firewalls, proxies, or monitoring stacks. Industry needs to be ready to use components that can survive high volume networks, not only conceptual insights.

Academia brings imagination and exploration, but practical adoption requires reproducibility, governance, realistic validation, compatibility, and packaging. Without these elements the bridge remains incomplete. Ideas gather on one side, needs gather on the other, and both continue waiting for a structure that connects them. [35] [41]

III. OUR APPROACH: A UNIFIED, IMPLEMENTABLE MODEL

3.1 Philosophy of an Adaptive Feedback Loop
The spirit of this model comes from a cycle that never really rests. Instead of imagining defenses as cold stone walls, this approach treats them like a creature that listens, reacts, and grows wiser with every confrontation. Attacks are not simply turned away. They are examined carefully, almost like studying tracks in soft soil. Each pattern reveals something, and those lessons return to strengthen the system. Over time the design behaves less like a rigid machine and more like a living guard that sharpens its instincts with every encounter.

3.2 The Two List Policy
Operational clarity often disappears when categorization becomes too clever. Here the system keeps things surprisingly simple. There are two lists. One list stops threats the moment they appear. The other list identifies noise that can be safely ignored. Anything that sits outside these two categories is sent toward the honeypot so it can be observed without harming production. Analysts manage only what matters. Everything unfamiliar becomes a learning opportunity rather than an unresolved mystery. The simplicity feels intentional and refreshing. [1] [6] [9] [10]

3.3 Human Guided Governance with Expiry and Burn In
Automation moves fast, but it has limits. This model insists that humans remain part of the flow. When new signatures appear from deception logs or anomaly sensors, they are not granted immediate authority. They spend time in an alert only stage. During this burn in period analysts study them and decide where they belong. Should they block? Should they be ignored? Or should they stay in observation for a while longer? Any signature that does not receive judgment within a fixed period fades from the queue so the system avoids clutter and stale rules. [1] [6]
[9] [10] [2] [3] [5] [7] [18] [22] [11] [15] [17] [33]

ISSN (Online) : 2319 - 8753
ISSN (Print) : 2347 - 6710

International Journal of Innovative Research in Science, Engineering and Technology
International Conference on Emerging Technologies and Pharmaceutical Research (ICETPR 2026)
17–18 January 2026
Indala College of Engineering, Polytechnic & Pharmacy, Kalyan, MH, India

|Volume 15, Special Issue 1, January 2026|

3.4 Deception Driven Intelligence for IDS Rules

Honeypots here are more than bait. They are a workshop where real attacker behavior is shaped into new defenses. Every interaction leaves behind trails of behavior that scripts collect and carve into candidate rules. These rules enter the governance pipeline where human reviewers study them. Slowly the attacker becomes an unintentional contributor. Their attempts forge the very shields that will counter the next wave. [1] [6] [9] [10] [35] [41]

3.5 Inline Visibility Through Proxy Terminated TLS and IDS Inspection

Encryption hides activity from prying eyes, which is good for privacy but troublesome for defenders. The design places a reverse proxy at the point where encrypted connections are opened. Once traffic becomes readable inside the trusted zone, Suricata can observe it with full clarity. Legitimate traffic continues to the application servers, and harmful flows are stopped or redirected. Privacy remains intact for external users while the internal defense stack sees what it must in order to function. [3] [5] [14] [28] [11] [15] [17] [33]

3.6 Benefits Compared to Academic Proposals

This approach keeps its feet planted on the ground. Everything rests on proven open source components like OPNsense, Suricata, Squid, and T Pot. No exotic systems. No elaborate theoretical engines. The model works because it was built with deployment in mind. People can replicate it easily. Every rule change is recorded. Every decision has a trail. The two list structure avoids confusing classification schemes and helps analysts remain confident even in noisy environments. It does not stay trapped in conceptual diagrams. It becomes something you can run, evaluate, and refine. [11] [15] [17] [33]

IV. DETAILED ARCHITECTURE OF OUR SETUP

4.1 Network Segmentation Across Zones

At the foundation sits a careful division of the network into four regions with distinct levels of trust and exposure. External Zone

This is where the outside world meets the firewall. Every request from the public begins here, much like a busy gate at the edge of a city. Server DMZ

This zone contains the reverse proxy and Suricata sensors. It protects critical backend systems while still allowing necessary communication with the outside world. [11] [15] [17] [33] Honeypot DMZ [1] [6] [9] [10]

This segment hosts the T Pot platform. It is a crafted environment built to attract malicious interest and watch it unfold safely.

Internal Zone

This is where sensitive systems reside. Access is deliberate and restricted. It represents the protected heart of the design. By splitting the network into these regions the system prevents a single breach from spilling everywhere at once.

4.2 Core Components and Their Purposes OPNsense Firewall

It stands at the border and controls what enters or leaves. It segments zones, enforces the least privilege principle, and applies block rules that evolve with new intelligence. Suricata IDS or IPS [11] [15] [17] [33]

After the proxy reveals readable traffic, Suricata examines it for known patterns and unusual behavior. It blocks confirmed threats and sends all alert information to the dashboard for deeper study. [11] [15] [17] [33]

Squid Reverse Proxy

It accepts encrypted sessions and opens them inside the trusted environment. This allows Suricata to analyze data that would otherwise remain hidden. [3] [5] [14] [28] [11] [15] [17] [33]

T Pot Honeypot Suite [1] [6] [9] [10]

A cluster of deceptive services waits here. Unknown or suspicious flows are sent into this zone so the system can observe what intruders try to accomplish. Custom Automation Tools

The severity engine decides whether something should be blocked, ignored, or diverted. The signature generator crafts new rule proposals from honeypot logs. A reporting tool compiles all events into summaries that help analysts make decisions. [1] [6] [9] [10] [11] [15] [17] [33]

Central Dashboard

Based on ELK with added governance features, this dashboard gathers logs from every component. Analysts review signatures, track rule decisions, manage expirations, and view historical context for any rule or event. [35] [41] [11] [15] [17] [33]

ISSN (Online) : 2319 - 8753
ISSN (Print) : 2347 - 6710

International Journal of Innovative Research in Science, Engineering and Technology

International Conference on Emerging Technologies and Pharmaceutical Research (ICETPR 2026)

17–18 January 2026

Indala College of Engineering, Polytechnic & Pharmacy, Kalyan, MH, India

|Volume 15, Special Issue 1, January 2026|

4.3 Traffic Flow and Decision Making

A packet moves through several stages. It reaches the firewall first, where basic policies shape its path. Approved packets move to the reverse proxy where encrypted sessions are opened. The plaintext flow moves to Suricata, which tests it against the rule lists. Block listed signatures stop traffic immediately. Ignore listed ones pass without interruption. Anything not classified moves toward the honeypot. Clean traffic reaches backend servers. Everything that enters the honeypot becomes part of new intelligence. [1] [6] [9] [10] [3] [5] [14] [28] [11] [15] [17] [33]

4.4 Governance Workflow

Every proposed signature appears in a queue that analysts must review. They choose whether the rule should block, be ignored, or remain in the divert category for further observation. If analysts do not act within seven days, the proposal disappears to prevent clutter. This cycle keeps the rule set healthy and ensures that human reasoning remains part of every important decision. [11] [15] [17] [33]

4.5 Logging, Monitoring, and Dashboards

The system records nearly everything. The firewall logs decisions and blocked addresses. Suricata sends alerts and metadata. The proxy provides session and handshake information. The honeypot logs attacker actions in detail. These streams merge in the ELK dashboard. Analysts can explore patterns, study events across time, watch global attack origins, and inspect the life of any rule. The governance layer adds decision tracking and expiration alerts. Together they create a transparent ecosystem where nothing disappears into obscurity. [1] [6] [9] [10] [35] [41] [11]
[15] [17] [33]

V. WHY THIS APPROACH IS UNIQUE

5.1 Bridging the Academia and Industry Gap

This approach does not float in academic clouds nor hide in corporate machinery. It carries ideas from research halls and shapes them into something that can survive in busy production networks. Instead of presenting another polished theory that never meets real traffic, it stitches academic imagination directly into familiar open source tools. Deception methods, anomaly detection, and adaptive learning do not remain abstract. They become features in systems people already trust. This continuity makes the model feel grounded rather than experimental, transforming scholarly creativity into something a network team can actually run without hesitation. [1] [6] [9] [10] [2] [3] [5] [7] [18] [22]

5.2 Defense in Depth with a Shared Feedback Rhythm

Layered defense often becomes a pile of separate walls that never speak to each other. This design encourages conversation between them. The firewall shapes boundaries. The proxy grants clarity. Suricata inspects with purpose. The honeypot learns quietly in the background and sends what it discovers back into the rule system. Each attempted intrusion strengthens the next cycle. The layers become partners instead of strangers, turning defense in depth from a static monument into a shifting, learning rhythm. [1] [6] [9] [10] [35] [37] [40] [41] [11] [15] [17] [33] [13] [21] [20]

5.3 Governance Focused and Human Guided

Many modern systems trust machines too much and people too little. When algorithms churn out rules without context, oversight becomes impossible. This design avoids the trap of opaque automation. Humans decide which rules matter, which should be ignored, and which need more study. Every rule ages. Every rule has a record. Analysts sit at the center, guiding decisions with evidence rather than guesswork. The result is a transparent system where judgment remains visible and accountable.

5.4 Reproducible, Scalable, and Built Entirely on Open Source

The architecture does not rely on secret platforms or costly hardware. It stands on open systems anyone can inspect or extend. OPNsense, Suricata, Squid, T Pot, and ELK form the backbone. Reproducibility becomes natural. People can build it in a lab, scale it across sites, and tune it without waiting for vendors. The freedom to adjust and experiment strengthens the entire model. Communities can contribute improvements, making evolution a shared effort rather than a closed process. [11] [15] [17] [33]

5.5 A Clear Alternative to Heavy Enterprise Stacks

Many enterprise solutions feel like tangled forests of features, licenses, and orchestration. They deliver power but demand complexity in return. This model chooses clarity instead. Two lists guide decisions. Traffic paths remain

ISSN (Online) : 2319 - 8753
ISSN (Print) : 2347 - 6710

International Journal of Innovative Research in Science, Engineering and Technology

International Conference on Emerging Technologies and Pharmaceutical Research (ICETPR 2026)

17–18 January 2026

Indala College of Engineering, Polytechnic & Pharmacy, Kalyan, MH, India

|Volume 15, Special Issue 1, January 2026|

readable. Every component has a clear purpose. Nothing feels bloated. Yet the depth remains. Firewalling, proxying, inspection, deception, and governance still form a complete defense. It shows that strong security does not require massive, intricate towers. It only needs thoughtful assembly and steady feedback. [1] [6] [9] [10] [35] [41] The uniqueness of this model does not come from inventing new parts. It comes from arranging familiar tools into a system that actually moves. It connects theory with field practice, coordinates layers so they teach each other, places humans in control of decisions, embraces open foundations, and offers a simpler alternative to the sprawling stacks many companies battle today.

VI. INDUSTRY BENEFITS

6.1 Reduced Alert Fatigue and Less Rule Overgrowth

Security teams often face alarms that repeat endlessly or carry little value. This model trims the noise. The two list method keeps rules either decisively blocking or clearly ignored. Anything uncertain moves into observation. This prevents endless rule expansion and keeps analysts from drowning in alarms. Their attention shifts toward meaningful events, easing pressure on already strained teams.

6.2 Faster Blocking of Confirmed Threats

Once a threat earns trust as a genuine risk, the model wastes no time. Approved signatures act instantly at the firewall. There is no delay, no extra process, no cautious waiting. Attackers meet firm resistance almost immediately. The quickness reduces their freedom to maneuver and limits the damage they can attempt. [11] [15] [17] [33]

6.3 Safe Analysis of New Threats Inside the Honeypot Space [1] [6] [9] [10]

Suspicious behavior is not always harmful. Sometimes it is unusual enough to deserve deeper study. The honeypot zone becomes a safe place where these ambiguous flows can act freely without harming production systems. Analysts gain insight, watching new techniques unfold in isolation. This gradual understanding turns uncertainty into confidence and fuels new defenses. [1] [6] [9] [10]

6.4 Compliance Friendly Logging and Auditable Decisions

Regulated industries demand full traceability. This architecture naturally produces detailed logs of rule creation, analyst review, expiration, and decision history. Every change becomes part of a permanent record. Auditors see a clear chain of responsibility without requiring extra tools. Compliance checks become simpler and less stressful.

6.5 Cost Efficiency Through Open Source Components

Security budgets often struggle under licensing fees and hardware renewals. Here the heavy lifting is done by mature open source systems with strong communities. Scaling the setup does not require vendor negotiation. Organizations gain advanced capability without the weight of recurring license costs. The economics shift in favor of flexibility rather than fixed contracts.

6.6 Tangible KPIs That Measure Actual SOC Improvement

Effectiveness becomes measurable. False positives drop as noise is removed. Detection time decreases because inspection sits inline. Blocking time shrinks because decisions are enforced instantly. Analysts have a clear window of seven days to process new proposals, giving structure to their workflow. Rule lifecycle metrics reveal how healthy the system remains over time. The SOC gains numbers that prove progress instead of relying on instinct.

Taking together these benefits create a security posture that protects more while demanding less. It sharpens focus, increases speed, deepens understanding, satisfies auditors, lowers costs, and backs every claim with measurable data.

VII. SCALABILITY AND FUTURE DIRECTIONS

7.1 Splitting Suricata and the Proxy Into Separate Machines

Early deployments keep the proxy and Suricata on a single host for simplicity. As traffic grows, the two can separate into distinct machines. The proxy continues to open encrypted sessions. Suricata receives mirrored plaintext streams. This separation avoids competition for resources and increases inspection capacity. Additional proxies can support heavy traffic while a centralized inspection cluster handles analysis at scale. [3] [5] [14] [28] [11] [15] [17] [33]

ISSN (Online) : 2319 - 8753
ISSN (Print) : 2347 - 6710

International Journal of Innovative Research in Science, Engineering and Technology

International Conference on Emerging Technologies and Pharmaceutical Research (ICETPR 2026)

17–18 January 2026

Indala College of Engineering, Polytechnic & Pharmacy, Kalyan, MH, India

|Volume 15, Special Issue 1, January 2026|

7.2 Expanding Into Proxy and IDS Farms

When demand rises sharply, one proxy and one sensor are no longer enough. Farms of proxies and farms of Suricata instances can grow horizontally. A load balancer distributes inbound connections across multiple proxies. Each proxy mirrors data to a pool of IDS nodes. Growth becomes a matter of adding more machines, not redesigning architecture. This elasticity supports global deployments, large user bases, and busy application landscapes. [11] [15] [17] [33]

7.3 High Availability With Redundant Firewalls and Sensors

Stability matters just as much as performance. OPNsense supports redundancy through CARP, allowing another firewall to take over instantly if one fails. Suricata sensors can also run in parallel. If one falls silent, others continue watching. This shift transforms the network from a fragile system into one capable of withstanding sudden failures without losing visibility. [11] [15] [17] [33]

7.4 Extending Protection Into Endpoints, Applications, and Identity

The current architecture focuses strongly on the perimeter and the network path. Future versions stretch outward and inward. Endpoints can join the same intelligence loop through EDR or XDR tools. A web application firewall protects application behaviors. API security modules guard modern traffic flows. Zero Trust concepts enforce identity based access throughout the environment. These additions fill protective gaps and connect the network layer with application and identity layers.

7.5 A Shared Path for Researchers and Practitioners

One of the design's quiet strengths is its openness for collaboration. Researchers can test new ideas in a realistic sandbox without starting from scratch. Practitioners gain insight from shared datasets and field experiences. A collaborative roadmap includes shared honeypot data, public rulesets derived from real adversarial behavior, joint performance evaluations, and training sessions where academic techniques meet operational challenges. Together these create a cycle where research informs practice and practice refines research. [1] [6] [9] [10]

Growth and evolution sit at the heart of this model. It scales gracefully from a small setup to a broad distributed environment. It welcomes new components without breaking its shape. It invites cooperation and constant improvement. In doing so it positions itself not as a finished design, but as a foundation that learns with every challenge it meets.

VIII. EVALUATION FRAMEWORK

8.1 Metrics for Benchmarking

A security system earns its credibility when numbers support its claims. Five measures guide the evaluation.

Precision

This ratio tells how often alerts point to real danger. When precision rises, analysts waste less time chasing illusions.

Recall

This measure reflects how many true attacks the system actually catches. High recall shows that adversaries find fewer hiding places.

False Positive Rate

Noise weakens defenders. Lowering this rate lightens the workload and keeps attention sharp.

Mean Time to Block

This value expresses how quickly the system shuts down a confirmed threat. Shorter intervals trim the attacker's freedom to act.

Throughput

A strong design handles heavy traffic without stumbling. This measure reveals whether performance holds steady under pressure.

Together these indicators create a clear picture of accuracy, sensitivity, efficiency, speed, and endurance.

ISSN (Online) : 2319 - 8753
ISSN (Print) : 2347 - 6710

International Journal of Innovative Research in Science, Engineering and Technology

International Conference on Emerging Technologies and Pharmaceutical Research (ICETPR 2026)

17–18 January 2026

Indala College of Engineering, Polytechnic & Pharmacy, Kalyan, MH, India

|Volume 15, Special Issue 1, January 2026|

8.2 Ablation Studies

To understand how each architectural choice shapes performance, components can be removed or altered in controlled trials.

With and Without Honeypot Feedback [1] [6] [9] [10]

By observing how detection quality changes without input from deception logs, evaluators can see how much the adaptive loop contributes to real blocking speed and rule strength. [1] [6] [9] [10]

Two List and Three List Policies

Testing both approaches reveals whether the simpler structure truly improves governance or whether an extra category offers value. These experiments replace guesswork with measured insight. [35] [41]

8.3 Testing with Real Traffic

Synthetic tests rarely capture the turbulence of production networks.

Encrypted Traffic [3] [5] [14] [28]

Modern networks rely on encrypted flows. The model must prove it can examine decrypted traffic after proxy termination without slowing or weakening inspection. [3] [5] [14] [28]

API Workloads

Enterprise systems rely heavily on rapid API exchanges. Stress tests using these patterns reveal whether inspection and governance remain steady under constant, high frequency requests. [35] [41]

Real traffic trials ensure the model evolves from theoretical promise into dependable practice.

8.4 Comparison Against Academic and Enterprise Designs A model's strength becomes clear when measured beside its peers. Academic Prototypes

Many academic designs focus on pattern recognition or learning based detection. Few integrate deception, governance, and operational traceability. Comparison exposes where academic creativity excels and where it falters under real conditions. [1] [6] [9] [10] [35] [41]

Enterprise Implementations

Commercial stacks often deliver strength wrapped in complexity and cost. By measuring throughput, accuracy, governance load, and financial impact, evaluators can see whether a leaner architecture can match or exceed traditional giants. [35] [41]

Evaluation therefore depends on both hard data and contextual comparisons, testing whether the architecture performs well not only in theory but also in the realities of field deployment.

IX. CONCLUSION

9.1 Key Contributions

The system described here does more than combine familiar security tools. It fuses governance, deception, adaptive learning, and human review into a single moving structure. Every intrusion attempt becomes part of a continuous learning process. The design grows over time, guided by observation rather than rigid assumptions. It behaves less like a static wall and more like an organism that adjusts, remembers, and strengthens itself. [1] [6] [9] [10] [35] [41]
[13] [21] [20]

9.2 What This Means for Academia

Researchers gain a platform where ideas can be tested in conditions closer to real deployments. Instead of experimenting only on curated datasets, they can study behavior inside a reproducible environment that reflects modern network flows. This encourages research that steps beyond theory, giving scholars a chance to test assumptions in a living ecosystem.

9.3 What This Means for Industry

Enterprises gain a practical, transparent, and economical alternative to heavy proprietary stacks. The architecture relies on open foundations and keeps governance visible, avoiding the hidden behaviors that plague opaque automation. SOC teams gain relief through reduced alert noise, clearer workflows, and measurable improvements. Strong defense becomes attainable without excessive cost or vendor dependency. [35] [41]

9.4 Adoption and Future Standardization

Adoption begins with controlled pilots. These pilots measure false positive reductions, blocking speed, and analyst efficiency. Successful results lead to case studies, community documentation, and broader acceptance. Over time

ISSN (Online) : 2319 - 8753
ISSN (Print) : 2347 - 6710

International Journal of Innovative Research in Science, Engineering and Technology

International Conference on Emerging Technologies and Pharmaceutical Research (ICETPR 2026)

17–18 January 2026

Indala College of Engineering, Polytechnic & Pharmacy, Kalyan, MH, India

|Volume 15, Special Issue 1, January 2026|

shared standards may form around governance patterns, signature lifecycles, and deception integrated workflows. This strengthens the field and encourages a cultural shift toward adaptive, transparent security strategies. [1] [6] [9] [10] [35] [41] [11] [15] [17] [33]

The model challenges complacent traditions. It shows that security can be responsive, open, and practical without sacrificing power. Its path from prototype to standard depends on collaboration and refinement, but its potential points toward a future where defensive systems learn as quickly as the threats they face.

APPENDIX A

Sample Rules and Metadata

Example Rules & Metadata with Categories (Block, Divert, Ignore)

Below are illustrative Suricata-style rule snippets annotated with metadata tags to indicate their governance category. These serve as templates for how signatures are maintained and classified. [35] [41] [11] [15] [17] [33]

Block Example

alert http any any -> any any (msg:"Exploit Attempt – CVE-2023-12345 RCE"; flow:to_server,established; content:"/vulnerable_endpoint"; http_uri; classtype:web-application-attack; sid:500001; rev:1; metadata: deployment BLOCK;source honeypot; confidence 0.95; created_at 2025-09-03;) [1] [6] [9] [10]

Action: Blocked immediately at firewall/IDS.

Justification: Confirmed remote code execution vulnerability actively exploited in the wild. **Divert Example**

alert tcp any any -> any 445 (msg:"Suspicious SMB Behavior – Unclassified"; flow:to_server,established; content:"\IPC$"; nocase; classtype:attempted-recon; sid:500002; rev:1; metadata: deployment DIVERT; source tpot; confidence 0.6; created_at 2025-09-03;)

Action: Redirect to honeypot for deeper observation. [1] [6] [9] [10]

Justification: Pattern indicates reconnaissance, but no confirmed exploit. **Ignore Example**

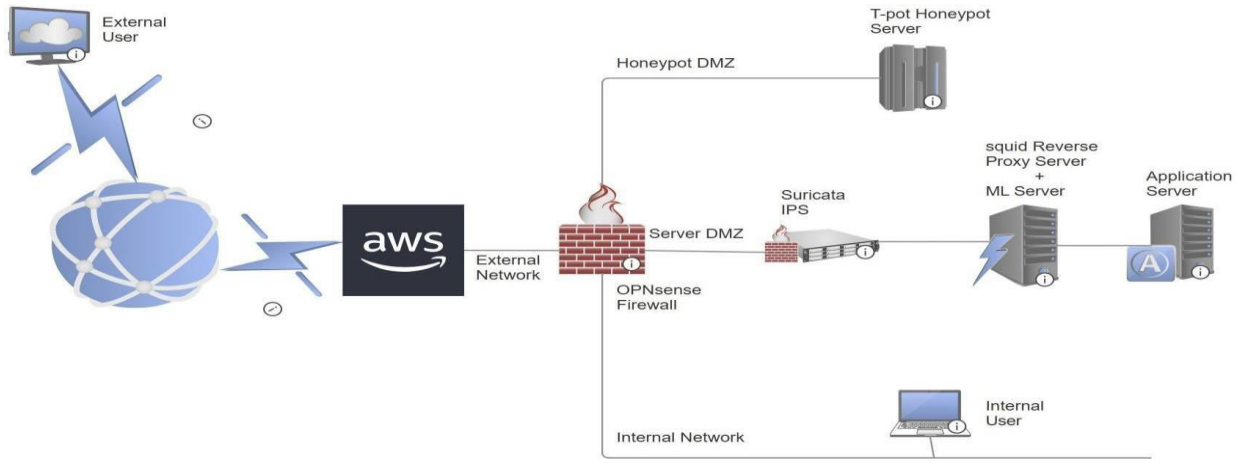
alert dns any any -> any any (msg:"Malformed DNS Query – Benign in this Environment"; dns_query; content:"example.internal"; nocase; sid:500003; rev:1; metadata: deployment IGNORE; source analyst; confidence 0.2; created_at 2025-09-03;)

Action: Suppressed from further analysis.

Justification: Known internal application quirk with no security impact.

APPENDIX B

Deployment Topologies Baseline Topology



Internet enters through the firewall, then reaches the proxy layer. Decrypted traffic flows to Suricata for inspection. Suspicious items move to the honeypot zone, while safe items reach backend systems. Logs gather in the ELK environment for analysis. [1] [6] [9] [10] [11] [15] [17] [33]

ISSN (Online) : 2319 - 8753
ISSN (Print) : 2347 - 6710

International Journal of Innovative Research in Science, Engineering and Technology

International Conference on Emerging Technologies and Pharmaceutical Research (ICETPR 2026)

17–18 January 2026

Indala College of Engineering, Polytechnic & Pharmacy, Kalyan, MH, India

|Volume 15, Special Issue 1, January 2026|

REFERENCES

[1] Spitzner, L. Honeypots: Tracking Hackers. Addison-Wesley, 2003. [1] [6] [9] [10]
[2] Perdisci, R., Gu, G., & Lee, W. "Using an Ensemble of One-Class SVM Classifiers to Harden Payload-based Anomaly Detection Systems." IEEE ICDM, 2006. [2] [3] [5] [7] [18] [22]
[3] Sommer, R., & Paxson, V. "Outside the Closed World: On Using Machine Learning for Network Intrusion Detection." IEEE S&P, 2010. [2] [3] [5] [7] [18] [22] [13] [21] [20]
[4] Alshamrani, A., Myneni, S., Chowdhary, A., & Huang, D. "A Survey on Advanced Persistent Threats: Techniques, Solutions, Challenges, and Research Opportunities." Computers & Security, 2019.
[5] Stiborek, J., Bartos, K., & Rehak, M. "Advanced Network Traffic Anomaly Detection Using Machine Learning." International Journal of Network Management, 2018. [2] [3] [5] [7] [18] [22]
[6] Rajab, M. A., Zarfoss, J., Monrose, F., & Terzis, A. "A Multifaceted Approach to Understanding the Botnet Phenomenon." IMC, 2006. [6] [9] [10] [16] [25] [26]
[7] Wang, K., & Stolfo, S. J. "Anomalous Payload-Based Network Intrusion Detection." RAID, 2004. [13] [21] [20]
[8] Moore, D., Shannon, C., Voelker, G. M., & Savage, S. "Internet Quarantine: Requirements for Containing SelfPropagating Code." IEEE INFOCOM, 2003.
[9] Bailey, M., Cooke, E., Jahanian, F., Xu, Y., & Karir, M. "A Survey of Botnet Technology and Defenses." Cybersecurity Applications & Technology Conference for Homeland Security (CATCH), 2009. [6] [9] [10] [16] [25] [26]
[10] Li, Z., Sanghi, M., Chen, Y., Kao, M. Y., & Chavez, R. "Honeynet-based Botnet Scan Traffic Analysis." ITCC, 2006. [6] [9] [10] [16] [25] [26]
[11] Jiang, N., Chen, J., & Wang, Y. "Rule Generation for Network Intrusion Detection Based on Attack Behavior Features." Security and Communication Networks, 2017. [13] [21] [20]
[12] Garfinkel, T., Pfaff, B., Chow, J., Rosenblum, M., & Boneh, D. "Terra: A Virtual Machine-Based Platform for Trusted Computing." ACM SIGOPS Operating Systems Review, 2003.
[13] Paxson, V. "Bro: A System for Detecting Network Intruders in Real-Time." Computer Networks, 1999.
[14] Tellenbach, B., Burkhart, M., Schatzmann, D., & Hofmann, M. "Accurate Host-Based Flow Rate Limiting Using Host Operating Systems." RAID, 2009.
[15] Lee, W., & Stolfo, S. J. "A Framework for Constructing Features and Models for Intrusion Detection Systems." ACM Transactions on Information and System Security (TISSEC), 2000. [13] [21] [20]
[15] Yegneswaran, V., Barford, P., & Ullrich, J. "Internet Intrusions: Global Characteristics and Prevalence." ACM SIGMETRICS Performance Evaluation Review, 2003. [13] [21] [20]
[16] Dagon, D., Zou, C. C., & Lee, W. "Modeling Botnet Propagation Using Time Zones." NDSS, 2006. [6] [9] [10] [16] [25] [26]
[17] Anagnostopoulos, M., Giannetsos, T., & Lambrinoudakis, C. "Intrusion Detection Systems and Anomaly Detection: An Overview." Information Security Journal, 2011. [2] [3] [5] [7] [18] [22] [13] [21] [20]
[18] Creech, G., & Hu, J. "A Semantic Approach to Host-based Intrusion Detection Systems Using Contiguous and Discontiguous System Call Patterns." IEEE Transactions on Computers, 2014. [13] [21] [20]
[19] Mukkamala, S., Sung, A. H., & Abraham, A. "Intrusion Detection Using Ensemble of Soft Computing Paradigms." Springer Advances in Soft Computing, 2005. [13] [21] [20]
[20] Porras, P., & Valdes, A. "Live Traffic Analysis of TCP/IP Gateways." NDSS, 1998.
[21] Vigna, G., & Kemmerer, R. A. "NetSTAT: A Network-Based Intrusion Detection Approach." Computer Security Applications Conference, 1998. [13] [21] [20]
[22] Kim, H., Kim, J., & Kim, H. "An Adaptive Methodology for Intrusion Detection with High Detection Rates and Low False Alarm Rates." Expert Systems with Applications, 2015. [13] [21] [20]
[23] Chaboya, L., & Mukherjee, B. "Performance of Anomaly Detection Systems in Anomaly Detection Approaches." IEEE ICC, 2001. [2] [3] [5] [7] [18] [22]
[24] Shameli-Sendi, A., et al. "Taxonomy of Intrusion Response Systems." Computers & Security, 2012. [13] [21] [20]
[25] Karasaridis, A., Rexroad, B., & Hoeflin, D. "Wide-Scale Botnet Detection and Characterization." HotBots, 2007. [6] [9] [10] [16] [25] [26]
[26] Shin, S., & Gu, G. "Conficker and Beyond: A Large-Scale Empirical Study." ACSAC, 2010.
[27] Grimes, R. A. Malicious Mobile Code: Virus Protection for Windows. O'Reilly Media, 2001.
[28] Marchetti, M., & Messori, M. "Anomaly Detection of Web-based Attacks." Journal of Information Security and Applications, 2015. [2] [3] [5] [7] [18] [22]
[29] Pletka, R., Weiler, N., & Hutchison, D. "A Survey of Self-healing Systems: Taxonomy and Future Directions."

ISSN (Online) : 2319 - 8753
ISSN (Print) : 2347 - 6710

International Journal of Innovative Research in Science, Engineering and Technology

International Conference on Emerging Technologies and Pharmaceutical Research (ICETPR 2026)

17–18 January 2026

Indala College of Engineering, Polytechnic & Pharmacy, Kalyan, MH, India

|Volume 15, Special Issue 1, January 2026|

International Journal of Computer Applications, 2008.
[30] Heidemann, J., et al. “Census and Survey of the Visible Internet.” ACM SIGCOMM, 2008.
[31] Tuncer, S., et al. “Cybersecurity Data Sources for Dynamic Threat Modeling.” IEEE Symposium on Security and Privacy Workshops, 2018.
[32] Caviglione, L., Gaggero, M., & Lalande, J. F. “Survey of Network Covert Channels: Techniques, Countermeasures, and Challenges.” Computers & Security, 2017.
[33] Maggi, F., Zanero, S., & Cavallaro, L. “Detecting Intrusions through Semantic Correlation of Audit Data.” Computers & Security, 2009. [13] [21] [20]
[34] Pei, K., et al. “Hunting for Undocumented System Behaviors in Binary Code with Machine Learning.” IEEE S&P, 2020. [2] [3] [5] [7] [18] [22] **Industry Frameworks and Standards**
[35] NIST Cybersecurity Framework (CSF) – National Institute of Standards and Technology, 2018.
[36] NIST Special Publication 800-61r2 – Computer Security Incident Handling Guide, 2012. [36] [43]
[37] NIST Special Publication 800-53r5 – Security and Privacy Controls for Federal Information Systems and Organizations, 2020.
[38] NIST Special Publication 800-171r2 – Protecting Controlled Unclassified Information, 2020.
[39] MITRE ATT&CK Framework – MITRE Corporation, 2023.
[40] CIS Critical Security Controls v8 – Center for Internet Security, 2021.
[41] ISO/IEC 27001:2013 – Information Security Management Systems – Requirements.
[42] ISO/IEC 27002:2022 – Information Security Controls.
[43] ISO/IEC 27035:2016 – Information Security Incident Management. [36] [43]
[44] ISO/IEC 15408 (Common Criteria) – Evaluation Criteria for Information Technology Security.
[45] ENISA Threat Landscape Report 2023 – European Union Agency for Cybersecurity. [46] OWASP Top Ten 2021 – Open Web Application Security Project.

INTERNATIONAL STANDARD SERIAL NUMBER INDIA

INNO SPACE SJIF Scientific Journal Impact Factor

doi cross ref

निस्केयर NISCAIR

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN SCIENCE | ENGINEERING | TECHNOLOGY

9940 572 462 6381 907 438 ijirset@gmail.com

www.ijirset.com

Scan to save the contact details