DOI: http://dx.doi.org/10.26483/ijarcs.v17i1.7405

ISSN No. 0976-5697

Volume 17, No. 1, January-February 2026

International Journal of Advanced Research in Computer Science

RESEARCH PAPER

Available Online at www.ijarcs.info

AI-POWERED MALICIOUS IP DETECTION AND INTERACTIVE CYBER SECURITY CHATBOT ASSISTANT

A. Selva Priya
Assistant Professor/ Department of Information Technology
Sri Ramakrishna Engineering College, Coimbatore, India

S. Sakthivel*
UG student / Department of Information Technology
Sri Ramakrishna Engineering College, Coimbatore, India

C. S. Sakthivel
UG student / Department of Information Technology
Sri Ramakrishna Engineering College, Coimbatore, India

R. Sanjay
UG student / Department of Information Technology
Sri Ramakrishna Engineering College, Coimbatore, India

Abstract: Because more people now use online services, attacks from harmful IP addresses are growing harder to ignore. Old rule-driven defenses struggle to keep up with how these threats change over time. A new method uses artificial intelligence to spot dangerous IPs by studying data flow, actions across systems, and known risks. Instead of relying only on fixed conditions, it learns from repeated behaviors that signal danger. What sets it apart is a smart chat helper built into the system. This part talks back when asked, offering clear details about possible threats. Learning happens continuously, adjusting as fresh attack styles appear. Responses come in natural language, making findings easier to grasp.

Keywords: AI-powered; malicious IP; interactive cyber security; chatbot assistant

I. INTRODUCTION

When people rely more on digital systems, cyber threats grow stronger. Hackers often hide behind certain IP addresses to launch actions such as overwhelming networks, guessing passwords, stealing data through fake pages, or spreading harmful software. These moves put network safety at risk. Spotting risky IPs early helps keep connections secure. Fixing problems fast supports stable and trustworthy operations.

Not often do old methods catch what slips through. When threats evolve fast, slow reactions fail. Hidden patterns? They demand smarter eyes. Machines now notice odd shifts before harm spreads. False alarms drop when learning systems take over. Speed improves once static rules fade out. Today's chaos needs tools that adapt mid-step. Guesswork fades where experience builds silently.

Even though machines now learn patterns, some digital guards still miss alerts. Because systems watch endless data flows, spotting odd behavior becomes possible. Yet delays happen when warnings reach human reviewers too late. While smart software catches threats fast, people sometimes respond slow. Although tech improves daily, timing gaps remain a real problem. When signals pile up, workers struggle to keep pace despite help from code. Since algorithms detect risks early, reaction speed should match - often it does not.

Dealing with this challenge takes shape through an approach laid out in the pages ahead. Built around artificial intelligence, it spots harmful IP addresses without relying on old patterns. Instead of just alerts, help arrives in the form of a live chat guide that responds in real time.

II. LITERATURE REVIEW

Some research looks at how machines learn to spot bad internet addresses and break-ins. Instead of just following set rules, these systems study past activity to tell what's risky. One method uses decision trees that vote together to make a choice about threats. Another relies on lines drawn through data points to separate normal from suspicious actions. Some models even detect oddities without knowing if they're dangerous ahead of time. Patterns hidden in web traffic can give away new kinds of attacks. These smart methods adjust themselves when faced with unfamiliar behaviors.

Out there, threat intel tools pull in outside info - things like bad IP tags or past attack clues - to beef up alarms. Yet they usually run solo, leaving humans to piece together what it means.

Lately, studies point to chat-based AI stepping into cyber safety roles - especially when things go wrong or someone needs help. These bots, built on language software, reply to security questions, break down warnings, maybe even walk a person step-by-step through fixes. Usability gets better with them around, yet live threat scanners seldom link up with these tools directly. Real-time spotting and chat helpers still run on separate tracks more often than not.

III. System Architecture

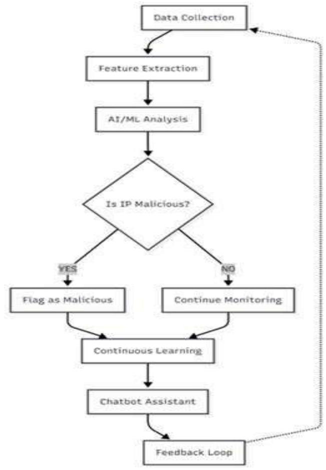


Fig 1 Flow Chart

One way to start: signals move fast through the framework, feeding into modules tuned for spotting harmful IPs without delay. After that comes live adjustments - each event shapes how future patterns get analysed, so performance shifts subtly over time. Instead of static rules, learning happens quietly behind scenes, nudged forward whenever new threats appear online. A chat-style front end opens access - not everyone types like experts, yet clarity still emerges from messy questions. Insights form not just from code but dialogue, where user intent helps sharpen what alerts matter most. Fig. 1 shows steps linked loosely, not rigidly - one stage passes results onward, though earlier parts may re-engage if needed. Responses grow sharper because loops exist; feedback reshapes detection precision bit by bit. Speed meets understanding here, avoiding blunt flags in favour of context- aware warnings. Large traffic loads enter one end, exit as distilled risks, trimmed down by layered checks along the path. Human input weaves into analysis early, making outputs easier to grasp when urgency strikes. Even complex behaviors become traceable since explanations tag along with every result produced.

Right off the bat, raw network data flows into the system through the Data Collection Module. Instead of one fixed source, inputs come from places like firewall reports, IDS warnings, server log entries, plus traffic histories. On top of that, key details tag along - source and target IPs show up alongside ports, protocols, time stamps. Visibility stays sharp because live tracking follows every packet moving in or out. This piece acts first, grabbing everything before anything else happens downstream. From the moment it arrives, data moves straight into the system - delay stays low. Because of this, odd patterns show up sooner rather than later. When load spikes happen, throughput doesn’t buckle; instead, performance holds steady. Even with constant surges, operation stays smooth without slowing down.

Because plain network records aren’t useful on their own, the Feature Extraction Module takes log entries and pulls out signs that might point to harmful activity. Out of simple event details, it builds clear summaries - looking at how often traffic appears, how fast requests come in, how long connections last. Session numbers, attempts to probe ports, which protocols show up most, past access habits - all turned into organized formats here. These outputs reflect quick actions as well as routines built over time, tied to each IP address. Instead of feeding messy logs straight into analysis, they become number-based snapshots. That shift helps machine learning spot faint irregularities, once older systems based on fixed rules tend to overlook.

Right in the middle sits the AI/ML Analysis Module, feeding smart judgment into spotting harmful IPs. Built using supervised models, it learns from data tagged clearly as safe or dangerous online behavior. When live traffic flows in, features pulled from network signals get scanned on the fly. Out comes a risk score per IP - higher means more suspicious. Because it studies past examples, it catches familiar attacks but also spots new twists in threats. Unlike old-style blocklists that just match names, this method adapts, cutting down wrong alarms. Accuracy climbs when patterns shift and noise fades, making choices about safety sharper. Decisions rest on learned trends rather than rigid rules stuck in the past.

Out of the AI/ML Analysis Module flows data, then steered into the Decision Engine for sorting suspicious IPs. Hitting a preset line triggers instant tagging - too high a score means flagged as dangerous. Below that mark, addresses escape immediate penalties but stay under watch. No person needs to step in; choices happen on their own, swift and steady. Speed here shrinks how long systems face danger, lessening harm before it spreads.

Right after an IP gets marked as harmful, a special alert system kicks in. That bad address goes straight into the system's log showing its risky history. At once, warnings pop up so IT staff and cybersecurity experts know what happened. Alongside those alerts, every move tied to the attack gets written down carefully. Later, investigators pull these notes when reviewing how things went wrong. Based on company rules, that flagged address might get shut out completely or slowed way down at entry points. Automation handles most of this instantly, stopping danger early. Because of this setup, damage often stops before it spreads too far.

Even if an IP seems harmless at first, the system does not dismiss it. It gets sent onward - into a process where behavior shifts are watched across days or weeks. The aim here is spotting quiet threats, ones that move slowly enough to slip past early checks. Normal routines form baselines; anything drifting from them raises flags automatically. Over time, what once looked safe might reveal subtle signs of risk. Long-running dangers, patient and hidden, stand less chance when every detail stays under review. Security grows stronger not through one scan but steady attention.

Fresh insights flow into the system through a built-in learning component, adjusting AI and ML models as new data appears along with how items get classified. When suspicious IP addresses are marked by analysts, they’re folded directly into the training pool - shaping model behavior around real- world shifts in attack styles. Retrained at regular intervals, these models sharpen their

© 2025-2027, IJARCS All Rights Reserved

judgment gradually, staying sharp when facing unseen threats or inventive breach attempts. Performance climbs steadily because yesterday's anomalies become today's signals.

One key part of the design focuses on people: the Cybersecurity Chatbot Assistant Module connects users - like analysts and admins - to the system through conversation. Instead of static reports, it talks back, using natural language understanding to grasp what someone is asking. When an alert pops up about a suspicious IP, the bot gives reasons behind the flag - not just raw data. Questions about active threats? It responds with specific details rather than vague warnings. From suggesting ways to reduce risks to turning dense findings into plain summaries, it shapes answers based on context. Because it explains things clearly, those using it spend less time digging and more time acting. Awareness grows when information flows smoothly, so decisions happen faster. Clarity like that changes how teams interact with complex signals across the network. Efficiency shifts not because tools get louder - but because they speak plainly.

Starting fresh each time, user actions feed into the system's brain, creating a cycle of adjustment. Because people reply, correct, or add details during chats, those inputs shape how alerts are set and responses formed. Responses grow sharper over time since every message helps retrain the models. When updates happen regularly, alignment with actual threats and daily operations stays strong.

One big plus? The setup spots bad IPs right away. It gives clear reasons how it knows, using chat-style talks with users instead of hidden logic. Learning shifts as threats change, staying sharp over time. Parts snap together easily, letting teams adjust without full rebuilds. Rules that never change matter less here - smart tools team up with people instead. This mix handles today's risks well while leaving room to grow when new dangers show up.

IV. NOVELTY

A new system uses artificial intelligence to spot harmful internet addresses while offering real-time help through a chat-based assistant. Instead of working alone like older tools that just send out technical warnings, it connects smart detection with a talkative interface. Because it explains threats in clear terms, people understand risks faster. Working together, these parts make responses quicker without needing deep training in cyber defence. Understanding grows when machines show not just what happened, but why it matters.

What stands out about this system is how it studies what bad IPs actually do. Most older tools rely on fixed lists or set-in- stone rules, yet those fail when facing new kinds of threats never seen before. Instead, this approach watches live actions

- like how often an address sends data, when it connects, how long sessions last, or which protocols it uses. By focusing on real-world activity, it catches unknown attacks, hidden breaches, and quiet hostile moves that slip past traditional filters. Detection becomes flexible, tougher, able to adjust without needing constant updates.

What stands out most is how the design focuses on decisions, keeping track of IP addresses as time passes. Not stuck with just a single judgment, it watches what IPs do over days. Even those first seen as harmless get checked again and again, catching threats that show up late or change shape slowly. Because of this steady watch, fewer dangers slip through unnoticed - especially ones shaped by attackers shifting tactics to dodge old defences.

What sets it apart is how it keeps learning nonstop, thanks to a design built around feedback loops. From detections to observations and even how people interact with it, every bit flows back into the training cycle to sharpen performance. Because of this shifting approach, it grows smarter along with emerging threats, needing fewer hands-on fixes. When fresh tactics show up in attacks, it quietly upgrades its knowledge, making future spotting more exact.

What sets this framework apart is how clearly it shows its reasoning. Most AI tools in cybersecurity work like closed systems, giving almost no insight into their choices. Instead, this approach opens up the process using everyday language to reveal what's happening behind the scenes. When an IP gets flagged, the chatbot breaks down the reasons, pointing out suspicious actions and red flags in plain terms. Seeing these explanations helps users rely on the outcomes more easily. Quick understanding leads to quicker responses during security events.

What makes this setup different? It keeps learning without needing constant updates. Instead of working in steps, each part talks back to the others - detection shapes monitoring, which tweaks how it learns, while what users do steers adjustments behind the scenes. When someone confirms an alert or replies during a conversation, those moments shift how sensitive the system becomes. Accuracy grows not from data volume, but because behavior pulls the tuning knobs. Not only does the new setup cut down on manual effort, it reshapes how threats are handled day to day. Rather than sifting through piles of data themselves, security teams get clear summaries by chatting with an assistant. Outcomes appear faster now - guidance shows up alongside context, not after long searches. What once took hours now unfolds in moments, turning isolated warnings into steps anyone can follow. The move isn't just about speed; it changes when help arrives during a crisis.

One thing leads to another - the setup grows as needs shift. Built this way, it takes in new threat data without skipping a beat. Learning systems plug in smoothly, just like automated defences do. Change comes fast in cyber space, yet the core stays steady. Even when networks stretch wider, the framework holds. New risks emerge, tools evolve, still it keeps pace. Not every design allows that kind of movement.

What stands out about this system? It weaves together behavior-driven threat spotting, ongoing adaptation, clear reasoning behind decisions, along with guided support - forming something cohesive. Instead of just relying on code, it blends smart analysis with how people actually work. This mix pushes past older models that stay rigid. Transparency grows. So does flexibility. Effectiveness shifts into sharper focus. The result feels less like automation, more like partnership.

V. RESULTS AND DISCUSSIONS

Out of real-life web activity logs and open-source sites labelled safe or harmful, the new tool that spots bad IPs with artificial intelligence came together alongside a talking helper bot. Not just spotting threats but explaining them clearly became its core task during testing. Accuracy in telling good from bad online behavior sat at the centre of scrutiny - how well it reads actions matters just as much as what users get back when they ask. Sometimes answers landed right, sometimes clues helped more than conclusions. Through each test run, clarity shaped up slowly without flash or guesswork guiding results.

Midway through testing, regular websites sitting on reliable IP addresses got visited just like everyday users would. Even though nothing suspicious showed up, their behavior stayed steady - timing between requests even, protocols running normally. Without hesitation, the system flagged each of these as safe. Into the ongoing tracking section they went, quietly joining the watchlist.

Picture a browser window showing regular site visits - everything loads just fine. Connectivity stays solid throughout, no hiccups at all. The system runs smooth, acting exactly how it should. That quiet confidence comes from knowing real sites aren't wrongly tagged. Legitimate addresses slip through untouched, unnoticed in the background. Few mistakes like that mean users won't get tripped up by alerts. Trust builds slowly when things work without drama. Usability thrives where errors are rare.

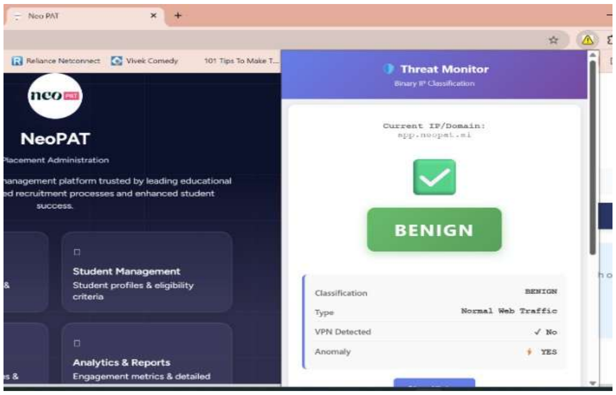
Unlike normal sites, harmful IP addresses got checked through recognized attack zones plus shady network origins. Odd actions stood out - like too many requests, strange timing between visits, multiple tries to reach different entry points. Spotting those quirks happened fast because the learning software caught each red flag correctly. Pages from bad IPs appear locked down, alerts pop up right away, proof sitting clear in Figure 3. Threats light up on screen the moment they hit, response already moving before danger spreads.

Right away, if an IP got flagged as harmful, alerts fired off while the incident went into the logbook. At that point, the automated helper stepped in, explaining what led to the flag. Not through piles of code lines but by breaking down causes- like odd data flows or actions straying from normal habits. That shift made things clearer, cutting down how much work people needed when checking threats.

Kept under watch, the system once again proved it works well when tested. At first, certain IP addresses looked harmless - then slowly started acting strange. Once noticed, those addresses got another look from the software and ended up marked as threats. Because of this shift in judgment, the setup shows it adapts on the fly, driven by decisions made mid-process. Attacks that unfold over time tend to slip past rigid tools, yet here they were caught.

What helped boost how well the system worked was its way of using feedback. When users confirmed answers or asked new questions, those moments shaped what it learned next. Gradually, sorting threats and explaining them grew more reliable - a sign that learning kept moving forward. Since cyberattacks shift all the time, being able to adjust like these matters most when used out in actual security settings.

The tests show it works - spotting bad sites without mistaking good ones too often. Because it combines smart detection with a helpful bot, people find it easier to use and more precise at the same time. Real images of safe and harmful pages were tested, proving it holds up outside lab conditions. What stands out is how well it performs when actually needed.



NeoPAT

Placement Administration

anagement platform trusted by leading educational
ed recruitment processes and enhanced student
success.

Student profiles & eligibility criteria

Engagement metrics & detailed

Threat Monitor

Binary IP Classification

Current IP/Domain:
app.neopat.ai

BENIGN

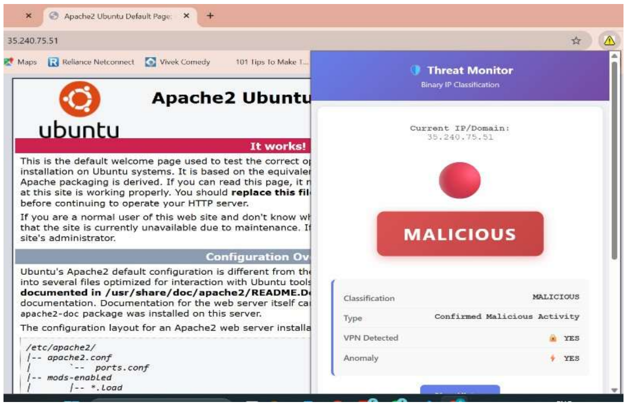
Classification BENIGN

Type Normal Web Traffic

VPN Detected ✓ No

Anomaly ⚡ YES

Fig 2 Benign Website Access



35.240.75.51

Apache2 Ubuntu

It works!

This is the default welcome page used to test the correct o
installation on Ubuntu systems. It is based on the equivale
Apache packaging is derived. If you can read this page, it n
at this site is working properly. You should **replace this fil**
before continuing to operate your HTTP server.

If you are a normal user of this web site and don't know wh
that the site is currently unavailable due to maintenance. If
site's administrator.

Configuration Ov

Ubuntu's Apache2 default configuration is different from the
into several files optimized for interaction with Ubuntu tools
documented in /usr/share/doc/apache2/README.D
documentation. Documentation for the web server itself ca
apache2-doc package was installed on this server.

The configuration layout for an Apache2 web server installa

```
/etc/apache2/
|-- apache2.conf
|       `--  ports.conf
|-- mods-enabled
|       |-- *.load
```

Threat Monitor

Binary IP Classification

Current IP/Domain:
35.240.75.51

MALICIOUS

Classification MALICIOUS

Type Confirmed Malicious Activity

VPN Detected 🔒 YES

Anomaly ⚡ YES

Fig 3 Malicious Website Access

VI. MERITS AND DEMERITS

A big plus here? It spots bad IPs by learning how they act, instead of just checking old lists. Because it watches things like unusual traffic spikes or odd login times, surprises pop up less often. Not stuck using only fixed rules means it adapts when new tricks appear online. Watching how data moves help catch sneaky attacks regular tools miss. Accuracy climbs since it isn't waiting for a perfect match in a database somewhere. Old ways relied too much on past examples - this one looks at what's actually happening now. Seeing patterns others ignore makes the whole thing tougher to fool. What matters shows up differently each time something feels off. Fewer false alarms happen because context gets considered every step. Most importantly, it works even when hackers change their methods overnight.

What stands out about the system is how it spots threats as they happen. Built to watch network flow nonstop, it handles information swiftly - catching harmful IP addresses without delay. Because responses come fast, harm from online attacks like service overload, forced break-ins, or spreading viruses gets blocked early. Spotting danger and sending warnings instantly makes protection far more effective for any organization.

A chatbot built into the system makes cybersecurity easier to interact with. Rather than showing warnings full of jargon, it translates threats into everyday language. Because people see what is happening in plain terms, they tend to trust the technology more. Understanding the type and seriousness of an attack becomes faster this way. So whether someone knows coding or not, reacting to breaches gets simpler. Fewer experts are needed when guidance comes through clearly.

Every now and then, the system grows smarter just by doing its job. Because it listens closely to what happens after each alert, it tweaks how it spots dangers without waiting around for human help. Over time, that means fewer old-school updates done by hand. Watching actual behavior out in the wild helps it stand firm when sneaky new attacks show up unexpectedly.

One big plus? The setup breaks into separate pieces that still work together smoothly. Each part - gathering info, pulling out key details, artificial smarts, tracking activity, chatting back - runs on its own without dragging down the rest. Need more threat feeds or smarter prediction tricks? Just plug them in. Even tossing in new automated helpers fits right into place. Expansion feels natural, never forced. So it works just as well on a few machines as it does across thousands, fitting into different setups without needing a redesign.

Even with clear advantages, the model isn't free of flaws - its need for clean training examples stands out. Because outcomes rely heavily on varied, well-tagged information, gaps show up fast when inputs lack balance. When details go unchecked or carry slant, errors creep into how addresses get sorted. Accuracy holds only if updates flow steadily and checks happen without pause.

When the internet connection wavers, problems start to show up. Real-time tasks demand steady speed plus strong computing power. If either one falters, everything slows down. Places stuck with old networks often struggle to keep pace. Weak hardware handles the load poorly. Running this smoothly gets harder where tools are outdated.

Now here's something to think about - the chatbot helper, useful as it is for making things clearer, comes with built-in boundaries. Sometimes language models stumble when faced with unclear questions from users, or hand out broad answers when dealing with intricate cyberattacks. It helps break down threats, sure, yet falls short where seasoned human insight matters most during serious breaches. When situations grow too big or too urgent, that's when people must step in, because automation alone can't carry the load.

Finding balance between safety and information control gets tricky sometimes. Even though its main job is spotting harmful IP addresses, moving tons of traffic around can open doors to leaks or abuse. Staying protected means strict login rules, solid scrambling methods for data, plus following digital safety laws - efforts that tend to make setup more tangled than expected.

What stands out is how this setup uses smart spotting, live tracking, clear reasoning from machines, along with ongoing adjustments, giving it strength in today's cyber protection efforts. Yet hurdles pop up when looking at reliance on data, needs for hardware support, and weaknesses in chat interfaces - each needing fixes before solid deployment happens. When fine-tuned properly, backed by strong safety layers, what it offers ends up far surpassing what holds it back.

VII. APPLICATIONS

Out there where networks need constant guarding, this tool fits right into many different setups. Instead of just flagging threats, it shows why an IP acts suspiciously - clear as day. Picture a setup that works smoothly whether protecting one office or a sprawling online infrastructure. What stands out is how fast it spots danger while talking back in ways people actually understand. Automation meets conversation here, helping teams stay ahead without getting lost in noise. Security feels less like guesswork when responses make sense and arrive on time. Not every defence system explains itself - this one does, quietly changing how warnings are handled. Behind the scenes, smart analysis pairs with straightforward replies, keeping things running. It adapts because today's threats don't follow old patterns. Protection grows smarter not by doing more, but by thinking clearer.

Midway through a busy workday, network flows get checked nonstop by the setup inside company systems. When banks, tech developers, or large firms handle private records, suspicious IP behaviors like forced entry tries or stealthy data leaks are spotted early on. Instead of digging hard through logs, staff receive clear alert summaries via a chat helper built into the tool. That sudden ping might just point to an ongoing breach attempt caught before damage spreads.

Midway through a hectic shift, an analyst clicks open the interface - suddenly, scattered warnings begin to group themselves into clear patterns. Instead of wading through noise, they ask a question in plain words and get back concise answers about what matters most. Alerts that once piled up like unread mail now shrink under smart filtering; each one weighed for urgency. Responses come threaded with context: how an intrusion unfolded, where it hit hardest, what step comes next. Fatigue eases when decisions gain clarity, even during peak storms of activity. Efficiency rises not from speed alone but from knowing exactly where to look.

When it comes to web hosting services or companies handling many sites and apps, this approach fits right in. Because threats like flooding servers, brute force logins, or bot-driven hacks are real risks, spotting bad IPs matters a lot. Instead of waiting, the setup notices odd patterns from certain addresses - then helps tech teams respond fast. Keeping things online and stable becomes easier when warnings come early.

Schools and research groups also rely on this kind of technology. Cyber threats frequently hit university systems, including course websites and private databases. The setup spots suspicious behavior early, giving warnings before damage occurs. Instead of just blocking attacks, it opens a way for tech teams or learners to ask questions through an assistant built into the interface. Knowledge about digital safety grows naturally when people get answers during real tasks. Protection becomes part of daily work, not a separate chore. Over time, that strengthens how well campuses handle online risks.

Now picture an online store moving fast, handling dozens of purchases every minute. Security matters most when names, addresses, and card numbers flow through digital paths. Instead of waiting for trouble, the system watches how each connection behaves across the web. Unusual login patterns? Repeated failed tries at checkout? These signs trigger quiet alerts behind the scenes. Fraud schemes slow down because the software spots odd rhythms before damage spreads. Even bots pretending to be shoppers get flagged without slowing real customers. During busy sale events, risks grow - yet help arrives instantly through a built-in helper that responds within seconds. Trust stays intact not by promises but steady actions working day and night.

Out there among busy data flows, Internet Service Providers tap into the system to spot odd traffic shapes across many users. Because suspicious IP addresses show up clearly, problems get flagged before they spread too far. When threats emerge, response happens fast - right where the network moves most of the load. Handling massive streams feels natural here, since the design grows smoothly with demand. Heavy loads? That is what it was built for, quietly working inside complex provider setups.

Because governments handle sensitive information, protection matters more than ever. When attacks emerge, smart software spots them fast - no delays. A talking interface then breaks down what happened, so teams understand next steps without confusion. Digital systems stay up, people keep trusting services, threats get blocked before harm spreads. Every alert gets explained plainly, never buried in jargon.

Out there among security service teams, this setup fits right in. With it, companies offer sharper watch systems, smarter risk checks, one step ahead of threats. Clients get updates that respond, not just repeat data. Less handwork shows up when sorting through alerts now. Better results come without piling on hours. Work flows easier once the system takes hold. Expert shops find it blends well with what they already do. Reporting wakes up - more alive than static charts ever were.

REFERENCES

[1] M. Alazab and A. Awajan,"IP SafeGuard—An AI-Driven Malicious IP Detection Framework," IEEE Access, vol.13,pp.9024990267,2025,doi:10.1109/ACCESS.2025.356 9289.

[2] Y. Lin and Y. Lu, "Evolving ML-Based Intrusion Detection: CyberThreat Intelligence for Dynamic Model Updates,"IEEE Transactions on Network and Service Management, vol. 20, no. 2, pp. 2335–2352, June 2025.

[3] A. Saim, "A Comprehensive Review of AI's Current Impact and Future Prospects in Cybersecurity, "IEEE Access, vol. 13, pp. 87500– 87526, 2025.

[4] D. Lazar, K. Cohen and A. Ron, "IMDoC: Identification of Malicious Domain Campaigns via DNS and Communicating Files," IEEE Access, vol. 9, pp. 45242– 45256, 2021, doi: 10.1109/ACCESS.2021.3066957.

[5] A. Jain and S. Verma, "Rule-Based Risk Scoring for Malware and System Threat Detection," International Journal of Cyber Security and Digital Forensics, vol. 14, no. 1, 2025.

[6] L. Orevi,"IPvest: Clustering the IP Traffic of Network Entities Hidden Behind a Single IP Address Using Machine Learning,"IEEE Transactions on Network and Service Management, vol. 18, no. 3, pp. 3647–3661, Sept. 2021, doi:10.1109/TNSM.2021.3062488.

[7] F. Le, E. Valdez, and P. Cheng, "Counting devices: Revisiting existing approaches in today's settings," in Proc. IEEE Int. Conf. Big Data (Big Data), Los Angeles, CA, USA, Dec. 2022, pp. 4032–4037.

[8] B. Zhang, Y. Guan, W. Niu, J. Tan, and Z. Mao, "A hybrid packet clustering approach for NAT host analysis," in Proc. IEEE Int. Conf.Commun. Softw. Netw. (ICCSN), 2021, pp. 432–438.

[9] A. S. Khatouni, L. Zhang, K. Aziz, I. Zincir, and A. N. Zincir-Heywood,"Exploring NAT detection and host identification using machine learning," in Proc. 15th Int. Conf. Netw. Service Manag. (CNSM), Halifax,NS, Canada, Oct. 2021, pp. 1–8.

[10] E. Valdez, D. Pendarakis, and H. Jamjoom, "How to discover IoT devices when network traffic is encrypted," in Proc. IEEE Int. Congr. Internet Things (ICIOT), Milan, Italy, Jul. 2022, pp. 17–24.