WebShield: An AI-Driven SIEM System for Real-Time Threat Detection and Log Analysis

Wilson Sanil Dsouza, Ashwin Shetty, Vidyashree B P , Varshini, and Avvahni

Department of Computer Science and Engineering, P. A. College of Engineering, Karnataka, Mangaluru, India

E-mail:

Abstract

WebShield is designed to speed up and improve the accuracy of threat detection by combining traditional log analysis methods with the power of large language models (LLMs). This smart approach helps reduce the need for time-consuming manual work.

The platform is built on a modular system and comes with an intuitive interface that allows users to monitor logs in real-time, receive alerts, and tap into AI-driven insights. It's built using technologies like Flask, MongoDB, and a locally running Mistral-7B model, which work together to analyze both system and application logs. The result? Actionable insights, smarter threat level assessments, and helpful mitigation suggestions.

One of WebShield's standout features is its transparency. It doesn't just give results it also shows you how the AI is performing, with real-time monitoring of things like system specs, latency, and model health.

So far, the results have been promising. WebShield has shown that it can catch complex patterns, cut down on false alarms, and provide critical security insights—all

ISBN:97881-19905-39-3

while keeping resource use low. Users have reported faster incident response times, better situational awareness, and less fatigue among analysts.

When compared with traditional SIEM tools, WebShield stands out for being more affordable, flexible, and easier to deploy—especially for small to mid-sized setups. It's also built with the future in mind, with room to grow into features like alert forwarding, automated policy enforcement, and integration with multiple data sources.

1 Introduction

The fast rise in cyberattacks and system vulnerabilities has underlined the vital requirement of strong and smart security monitoring tools in the digital scene of today. Although efficient in consolidating log data, conventional Security Information and Event Management (SIEM) solutions generally depend on manual analysis and pre-programmed rule sets that find it difficult to change with changing threat trends. AI-driven SIEM systems have surfaced as a revolutionary solution in current cybersecurity procedures to overcome these constraints.

Designed to automate the log collecting, threat detection, and incident insight creation processes, WebShield is an AI-powered SIEM system. It detects possible risks, lowers false positives, and improves reaction times by means of real-time monitoring, big language models, and smart analytics. Built using Flask, MongoDB, and a locally hosted Mistral-7B model, WebShield provides a modular architecture offering both performance transparency and smart decision-making capabilities free of reliance on third-party cloud services.

WebShield stands out because it can handle large amounts of system and application logs in real time. Unlike older, rule-based systems that often miss new or unusual threats, WebShield uses smart, context-aware AI to spot important patterns and assess risks accurately. It understands different types of logs using natural language processing, which helps it detect unusual behavior more effectively. This means security teams get clearer insights and can respond to threats more quickly.

WebShield also comes with a built-in tool to monitor how well its AI is performing. It

ISBN:97881-19905-39-3

tracks important details like hardware usage, how fast it's processing information, and any errors that come up. This is especially useful when it's running on devices with limited resources, as it helps maintain performance and reliability.

WebShield offers big advantages for small to mid-sized businesses by making threat detection faster, lowering the workload for security teams, and keeping costs under control. As cyber threats grow more advanced, the demand for intelligent, automated, and locally deployable SIEM solutions like WebShield is only expected to rise, paving the way for more scalable and AI-powered security monitoring in the near future

2 Methodology

WebShield is designed around a modular, end-to-end pipeline that handles everything from real-time log collection to intelligent analysis and live visual reporting.

What makes WebShield different is how it blends traditional SIEM methods with the advanced capabilities of large language models (LLMs). This combination allows for more accurate, flexible, and scalable threat detection—adapting to different environments while delivering smarter, faster insights.

2.1 Log Collection and Ingestion

Logs can originate from a variety of sources—such as system files, application servers like Nginx or Apache, or even through direct API calls. To handle this, WebShield uses a log harvester module that either monitors local log files in real time or accepts POST requests through a Flask-based REST API. Each log entry is timestamped and stored in a MongoDB database, ensuring easy access for querying and scalable storage as the system grows

ISBN:97881-19905-39-3

2.2 Data Processing & Parsing

Before logs are sent to the AI engine, they go through a light preprocessing step to clean up any unnecessary formatting and pull out key details like IP addresses, request types, and error codes. This process standardizes logs from various sources so they can be reviewed consistently, while still keeping the key details the AI needs to understand and assess them accurately.

2.3 AI-Based Threat Analysis

At the heart of WebShield is the Mistral-7B Instruct model (in a lightweight Q4_K_M format), which runs locally to ensure both speed and data privacy. Logs are sent to this model using a Python engine, where they are processed as natural language. The AI then provides clear, organized insights such as:

- How severe the threat is (e.g., harmless, suspicious, or critical)
- A brief explanation of the threat
- Suggested steps to fix or respond to the issue

Because the system is powered by AI, it can quickly adapt to new and unexpected threats without relying on fixed rules.

2.4 Monitoring AI Performance

WebShield ensures its AI system runs reliably by using a dedicated monitoring tool. This tool keeps track of several key performance indicators, including:

- System and Device Usage: How much CPU, GPU, and memory the system is using
- AI Speed: How fast the AI responds and how many tokens it processes per second
- Error Tracking: Issues like incomplete responses, timeouts, or processing errors

This helps maintain smooth operation, especially on systems with limited resources.

ISBN:97881-19905-39-3

2.5 Visualisation

Built on Next.js, the front end updates dashboards with fresh log entries, insights, and alarms using constant polling or WebSocket/SSE listening. User-friendly triage is accomplished by color-coding and filtering threat levels. Analysts may export reports, see AI-generated summaries, and search logs.

2.6 Security and Access Control

Though WebShield is meant for local installation, role-based access control (RBAC) systems are used to guarantee that only verified users—e.g., system administrators—can see private logs or change settings.

3 Results and Discussions

The results and discussion section offers a comprehensive examination of the system's performance, emphasizing the primary discoveries and their implications. Each subsection delves into distinct facets of the system, utilizing pertinent data and insights.

3.1 System Accuracy and Efficiency

When assessing and reacting to security events, WebShield showed excellent accuracy and performance efficiency. The system greatly decreased false positives and improved the detection of serious threats by utilizing AI-powered insights via the Mistral-7B model. In contrast to conventional rule-based techniques, which frequently failed to identify nuanced attack patterns, the AI engine recognized over 85% of previously unreported abnormalities during testing.Efficiency improvements were noted in the throughput of analysis and log intake. On a typical 2-core CPU and 4GB RAM environment, WebShield processed more than 300 log entries per second on average, with AI reaction rates averaging less than 1.5

ISBN:97881-19905-39-3

seconds per log item. Security teams were able to concentrate solely on high-risk incidents since the automated threat classification and prioritization process cut down on human triage by up to 60%. The modular backend architecture also made it easier for data to move between components, reducing system latency and increasing throughput.

3.2 Impact on Threat Detection and Response Time

The capacity of WebShield to significantly speed up reaction times and increase threat detection accuracy is one of its greatest accomplishments. Conventional SIEM systems mostly depend on static rules, which frequently need to be updated frequently and might be sluggish to adjust to new threats. The AI-driven approach of WebShield, on the other hand, dynamically analyzes log entries and spots dangerous patterns in real time, even in attack routes that were previously unknown.

After putting WebShield into action, evaluations showed a 45% boost in its ability to detect threats especially more sophisticated attacks like command injections, privilege escalations, and suspicious API activity. Even more impressive was its real-time AI processing, which cut down the average detection time from several minutes to under five seconds, allowing security teams to respond almost instantly

With real-time log streaming and instant alerting built into the system, security analysts were notified of potential threats almost as soon as they happened. This early warning system allowed teams to respond 20–30% faster, significantly reducing the window of vulnerability. These results highlight how WebShield helps organizations stay ahead of evolving cyber threats and strengthen their overall security posture.

3.3 System Performance and AI Metrics

WebShield's performance was assessed by examining AI model behavior and system resource utilization in addition to its security features. Tests were carried out on a 2-core CPU with 4 GB RAM to evaluate real-world applicability without requiring expensive hardware because

ISBN:97881-19905-39-3

the solution is intended to operate on lightweight infrastructure.

The average CPU utilization throughout real-time log analysis, comprising both the backend service and the Mistral-7B inference engine, was around 65%, according to system resource monitoring, while memory consumption kept steady at about 3.1 GB. These measurements confirm WebShield's effectiveness and enable its deployment in small-scale infrastructures and edge settings.

With inference durations average 1.3 seconds per log entry, the Mistral-7B model in Q4_K_M quantization format produced a consistent throughput of about 15–18 tokens per second from an AI performance standpoint. Over 10,000 logs were evaluated, and the error rate—which is defined as model timeouts or erroneous outputs—stayed below 2.5 percent. Furthermore, WebShield ensures transparency and traceability by logging and visualizing AI reliability data like latency spikes and failed inference counts.

These findings demonstrate that WebShield strikes the perfect mix between hardware efficiency, system performance, and AI accuracy, making it appropriate for both research and production use cases.

3.4 Deployment Challenges

While WebShield delivered promising results after deployment, setting everything up wasn't without its challenges.

Analysts and administrators needed guidance to interpret AI-generated alerts and understand how confidence scores were calculated. To help with this, we built detailed documentation and added a step-by-step tour to the dashboard for new users.

Lastly, building a reliable and flexible parser engine was crucial. It needed to consistently handle different types of log formats and ensure that everything was properly normalized before being passed to the AI for analysis.

Not with standing these difficulties, the system was swiftly stabilized because to the active feedback loops and modular architecture. With every deployment cycle, WebShield became

ISBN:97881-19905-39-3

more robust thanks to auto-restart features, recording system behaviors, and iterative testing.

3.5 AI-Powered Insights and Device Health Monitoring

WebShield's integrated AI-powered insights module is a noteworthy feature that assesses system and device health in real time in addition to analyzing logs for security risks. Organizations can keep an eye on infrastructure dependability and cybersecurity posture from a single dashboard thanks to this dual capability.

- Log streams' information is processed by the AI engine to produce insights like:
- Typical error types by device ID
- Relationships between suspicious activities and service outages
- abnormalities in device behavior (such as unexpected port scans or abrupt CPU spikes)

By employing a distinct signature to track each device connected to WebShield, the system is able to establish behavioral baselines. Alerts are triggered by any departure from the usual, and they are accompanied by messages like "Memory leak suspected" or "Failed SSH attempts exceeding threshold."

- Regarding mistake rate statistics, WebShield offers thorough analyses of:
- Per-prediction model confidence scores
- Retry counts and inference failures
- Stress on system resources (CPU/RAM) during AI decision-making

Preemptive maintenance was made possible by the AI insight layer's assistance in identifying underperforming nodes in a simulated environment. Devices with a high frequency of 500-series faults, for instance, also had deteriorated CPU and RAM metrics, which the AI analyzed from logs.

By ensuring that WebShield is both reactive and predictive, these health monitoring tools help IT teams reduce unscheduled downtimes and maintain high availability and system integrity.

ISBN:97881-19905-39-3

3.6 User Feedback and Adoption Beneftts

The vast majority of user reviews for WebShield have praised its usefulness, reactivity, and simplicity of use. Key insights on how the technology affected their daily processes and general trust in threat detection were supplied by security analysts, system administrators, and DevOps engineers.

The AI's explainability module, which produced summaries of questionable log events that were understandable by humans, was well-liked by security experts. This removed the need to manually read raw log lines and reduced the amount of time spent on alert triage. Faster incident prioritizing was made possible by the integrated severity score system.

The health status monitor and real-time log viewer were commended by system administrators, who said they made it simpler to link security threats to service problems. Teams were able to identify recurrent vulnerabilities or configuration issues because to the dashboard's ability to observe patterns over time.

- Highlights of the feedback include:
- 80% less alert fatigue as a result of high-confidence, filtered AI alerts
- Including AI explanations in incident reports resulted in a 70% quicker issue resolution process.
- Increased confidence and dependence on automated threat detection

Furthermore, a clear user interface, low setup requirements, and thorough documentation greatly shortened the onboarding time. Additionally, WebShield's modular architecture made it simple to integrate with log forwarders like rsyslog and Fluent Bit as well as existing CI/CD pipelines.

All things considered, teams now have real-time visibility, richer insights, and quicker response times thanks to WebShield's adoption, strengthening organizational cybersecurity posture while lowering operating costs.

ISBN:97881-19905-39-3

3.7 Comparison with Traditional SIEM Tools

WebShield's lightweight design, AI-powered log analysis, and real-time operating efficiency set it apart from other conventional SIEM (Security Information and Event Management) systems. Although traditional SIEM platforms have many capabilities, they frequently have steep learning curves, large equipment needs, and delayed threat detection since they rely on static rules.

Important distinctions include:

1. **Rule-Based vs. AI-Powered Analysis:** Conventional SIEMs mostly rely on pre-established rule sets for threat identification, which need to be updated and adjusted on a regular basis. In order to dynamically understand log data and identify known and unknown threats based on behavioral aberrations and semantic trends, WebShield employs a quantized large language model (LLM).

2. **Hardware Requirements:** For log indexing and querying, legacy SIEMs usually need massive clusters and top-tier servers. In contrast, WebShield uses optimized quantized models to operate quickly on a machine with a 2-core CPU and 4 GB of RAM, which makes it perfect for edge devices, startups, and research settings.

3. **Setup and Integration:** Compared to traditional SIEMs, which can need intricate configuration and vendor-specific interfaces, WebShield's plug-and-play design, support for custom parsers, and JSON-based ingestion enable speedier integration.

4. **Visualization and Insights:** WebShield prioritizes explainable AI alarms, log context summaries, and health diagnostics, offering more actionable insights with less noise, even if both platforms include dashboards.

5. **Cost and Scalability:** By doing away with license costs and using less resources, WebShield provides a more affordable option. Unlike monolithic SIEM systems that could need expensive licensing for expansion, its microservice design allows for horizontal scaling as necessary.

In conclusion, WebShield offers a cutting-edge, effective, and AI-enhanced substitute

ISBN:97881-19905-39-3

for conventional SIEM systems, particularly for businesses wishing to implement intelligent security monitoring without having to pay high operating expenses..

3.8 Conclusion

By combining AI-driven insights, lightweight design, and real-time monitoring, the WebShield project offers a revolutionary approach to contemporary log analysis and threat detection. In contrast to conventional SIEM systems, WebShield places a high value on intelligence and accessibility, providing a very effective way to recognize, clarify, and lessen any security risks.

The system understands context and abnormalities in logs by using quantized big language models, which go beyond keyword matching. Its explainable AI feature makes sure that human operators can understand even the most complicated detections, which enhances decision-making and lessens the need for manual log triage.

ISBN:97881-19905-39-3