SkyShark: Real-Time Kernel Behavioral Monitoring in Cloud-Native Systems

Rebecca Moroz, Dr. Brian Mitchell

College of Computing & Informatics, Drexel University, Philadelphia, PA

Abstract

Cloud-native systems are increasingly vulnerable to API-driven data breaches that bypass traditional defenses, often exploiting built-in functionality without deploying malware, evading traditional tools.

We simulate realistic API-based breaches in a cloud-native environment and introduce a platform built on NATS, OpenTelemetry, and Grafana that uses machine learning to baseline normal application behavior and detect deviations in real time.

Results show that combining kernel-level observability with modern telemetry tools uncovers stealthy threats missed by conventional monitoring.

Motivation

Modern cloud-native systems rely on APIs and microservices, which expand the attack surface, Many modern breaches exploit **legitimate system functionality** in abnormal ways, like over-permissioned API calls or misconfigured endpoints, making them invisible to signature-based tools.

Traditional solutions struggle because they:

- Rely on **known attack signatures**
- Miss subtle runtime anomalies
- Can't keep up with dynamic, short-lived workloads

There is a critical need for solutions that monitor runtime behavior, detect malware-free, logic-level threats, and integrate with existing SecOps workflows.

Methodology

SkyShark uses **eBPF/XDP probes** to collect kernel-level telemetry (syscalls, file, and network activity) without code changes.

An **unsupervised autoencoder**, trained on benign workloads, scores microbatches in real time using **reconstruction loss** to detect anomalies. Loss metrics are emitted using OpenTelemetry + Prometheus.

We tested SkyShark on a Kubernetes-based banking app with real APIs. Using **Locust**, we simulated **500 normal users** and **1 attacker**. The attacker exploited API misconfigurations to exfiltrate data at varying rates over 15-minute sessions.

Attack Pattern	Normal Users	Attackers	Data Leak per Request
Leak	**500**	**1**	**1-25 records**
Slow	**500**	**1**	**1-250 records**
Moderate	**500**	**1**	**250-500 records**
Aggressive	**500**	**1**	**500-1000 records**

Table: Attack Experimental Models

Contributions & Results

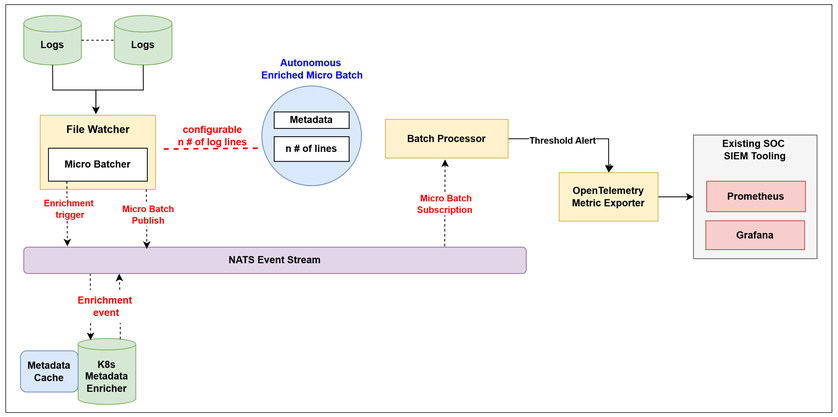

Figure: SkyShark System Architecture

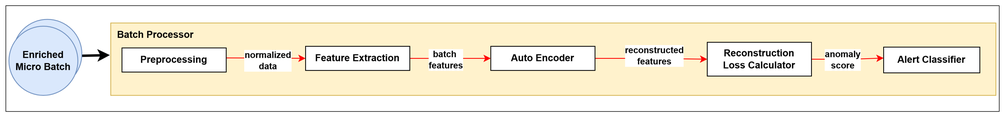

Figure: SkyShark Anomaly Detection

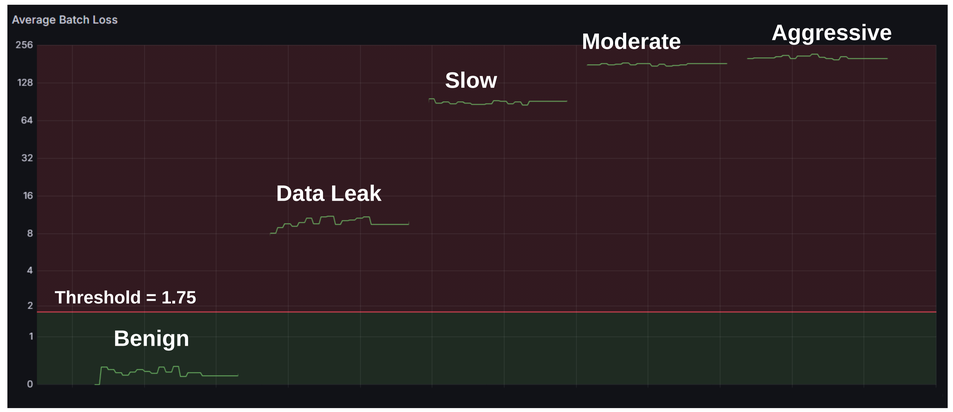

Figure: SkyShark Average Batch Loss

- SkyShark consistently flagged attacker behavior across all four attack types with no false negatives
- Anomalous microbatches exceeded the reconstruction loss threshold, triggering alerts.
- Benign sessions remained well below the threshold, confirming low false positive rates.

Despite only one attacker hidden among 500 normal users, SkyShark reliably detected data exfiltration ranging from small leaks to aggressive theft and as attacker strength increased, reconstruction loss increased proportionately.

Conclusions

SkyShark proves that real-time kernel-level behavioral monitoring can effectively detect modern, malware-free cloud threats, especially those that exploit legitimate functionality, like API misuse. By learning normal system behavior through unsupervised modeling, it flags subtle anomalies missed by signature- or rule-based tools.

- Across all scenarios, SkyShark consistently detected a single attacker hidden among 500 users, with **no false negatives** and **low false positive rates**. As attacker intensity increased, anomaly scores increased, demonstrating the system's sensitivity and robustness.
- By maintaining a low false positive rate, SkyShark reduces **alert fatigue** by limiting noisy or redundant alerts and allowing teams to focus on truly anomalous behavior.
- SkyShark also fills a critical operational gap: it integrates into existing production environments using a **cloud-native event stream pipeline (OpenTelemetry + NATS)**. This allows for real-time alerting and monitoring **without introducing new agents or requiring code changes,** a key advantage over traditional tools in **real-world SecOps** environments.

Future Work

- Explore transformer-based architectures to better capture long-range dependencies in application behavior
- Contrastive Loss approach to improve anomaly separation and adaptability for novel attack types
- Simulate a wider variety of targeted attacks to pinpoint model strengths and weaknesses.

Acknowledgements

Thank you to Dr. Mitchell and the College of Computing & Informatics.