Automated Alert Classification Using Machine learning and NLP

Problem Specification

Security Operations Centers (SOC) and Network Operation Center (NOC) are inundated with high volumes of alerts, which causes analysts to experience alert fatigue and struggle with timely incident response. The current reliance on manual triage processes amplifies false positive rates and risks critical security incidents being overlooked. Existing automated triage tools often fail to integrate unstructured alert descriptions with structured metadata, thereby missing opportunities to enhance classification accuracy through Natural Language Processing (NLP).As stated in Figure 1.1 Majority of NOC/SOC alerts are unable to handle and insufficient information also with Estimated false positive rate 40 % and false negative rate (failure to take action) 30 %

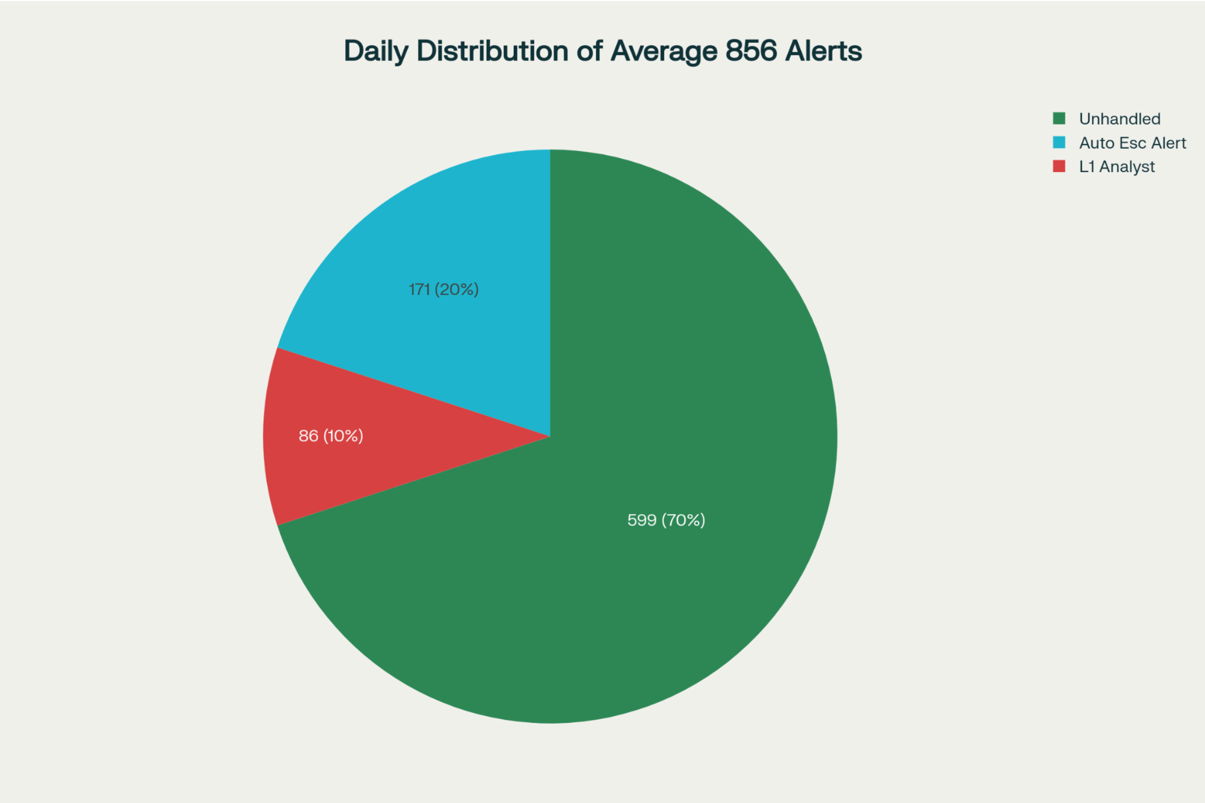
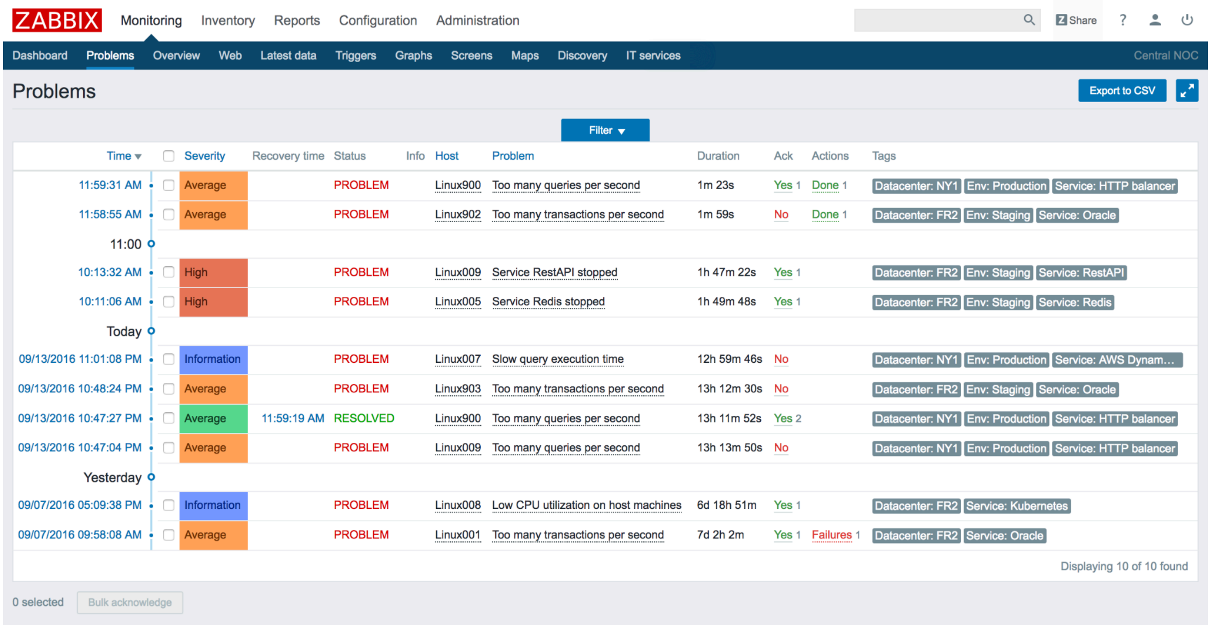


Figure 1.1 Daily Alert Distribution of NOC/SOC



Problems

Time ▾	□	Severity	Recovery time	Status	Info	Host	Problem	Duration	Ack	Actions	Tags
11:59:31 AM	□	Average		PROBLEM		Linux900	Too many queries per second	1m 23s	Yes 1	Done 1	Datacenter: NY1 Env: Production Service: HTTP balancer
11:58:55 AM	□	Average		PROBLEM		Linux902	Too many transactions per second	1m 59s	No	Done 1	Datacenter: FR2 Env: Staging Service: Oracle
11:00											
10:13:32 AM	□	High		PROBLEM		Linux009	Service RestAPI stopped	1h 47m 22s	Yes 1		Datacenter: FR2 Env: Staging Service: RestAPI
10:11:06 AM	□	High		PROBLEM		Linux005	Service Redis stopped	1h 49m 48s	Yes 1		Datacenter: FR2 Env: Staging Service: Redis
Today											
09/13/2016 11:01:08 PM	□	Information		PROBLEM		Linux007	Slow query execution time	12h 59m 46s	No		Datacenter: NY1 Env: Production Service: AWS Dynam…
09/13/2016 10:48:24 PM	□	Average		PROBLEM		Linux903	Too many transactions per second	13h 12m 30s	No		Datacenter: FR2 Env: Staging Service: Oracle
09/13/2016 10:47:27 PM	□	Average	11:59:19 AM	RESOLVED		Linux900	Too many queries per second	13h 11m 52s	Yes 2		Datacenter: NY1 Env: Production Service: HTTP balancer
09/13/2016 10:47:04 PM	□	Average		PROBLEM		Linux009	Too many queries per second	13h 13m 50s	No		Datacenter: NY1 Env: Production Service: HTTP balancer
Yesterday											
09/07/2016 05:09:38 PM	□	Information		PROBLEM		Linux008	Low CPU utilization on host machines	6d 18h 51m	Yes 1		Datacenter: FR2 Service: Kubernetes
09/07/2016 09:58:08 AM	□	Average		PROBLEM		Linux001	Too many transactions per second	7d 2h 2m	Yes 1	Failures 1	Datacenter: FR2 Service: Oracle

Displaying 10 of 10 found

0 selected

Figure 1.2 Typical events from monitoring system

Figure 1.2 explain traditional alerts are base on objects and trigger on certain threshold which work well on time series data . The **objective of this project is to develop a hybrid system that combines traditional machine learning with NLP to automatically classify cybersecurity and performance problem alerts**. By leveraging both structured features—such as severity, timestamp, source and destination IP addresses, user and system entities and NLP-derived features—such as alert title and description embeddings, TF-IDF vectors, named entities, and sentiment or urgency scores—this solution aims to reduce analyst workload while preserving high accuracy and low false negative rates.

Data Source

Data will be sourced through 2 possible approach. Option 1 Public datasets such as the AIT alert collection, CICIDS network intrusion logs, the NapierOne threat indicator repository, will provide a foundational corpus. Option 2, partnerships with private NOC providers and managed SOC services will supply anonymized,

Sample Data

This system will process a stratified sample of approx 2,000 alerts drawn from enterprise SOC/NOC environments,. Stratification will ensure balanced representation across critical, high, medium, and low-severity categories. The model's output will consist of a severity prediction—categorized as Minor, Major, or Critical—accompanied by probability scores.

Feature	**Encoding Method**
Time	Extract → Hour, DayOfWeek, Month;
Severity	Ordinal Encoding (0=Info→3=Average)
Status	One-Hot Encoding
Host	Label Encoding (Sensitive Data)
Problem	TF-IDF (max 100) with NER

Figure 1.3 Alert Features and target encoding method

Expectation

The alert-classification model—built on static features and enhanced with basic NLP signals—can be expected. with precision of .70–.80, while a recall of .70 This operating point corresponds to an overall F1-score of approximately 0.65–0.75,