Real-time Streaming Analytics & Alerting

Team - Anonymous
Srinivas
Surya
Amitav

April 2015

Agenda

Problem and Solution Overview

System Demo

Summary

What is the problem statement?
Create a real-time log stream monitoring and alerting system which can process continuous stream of log data, monitor it for user defined keywords and generate alerts when you detect the key words.

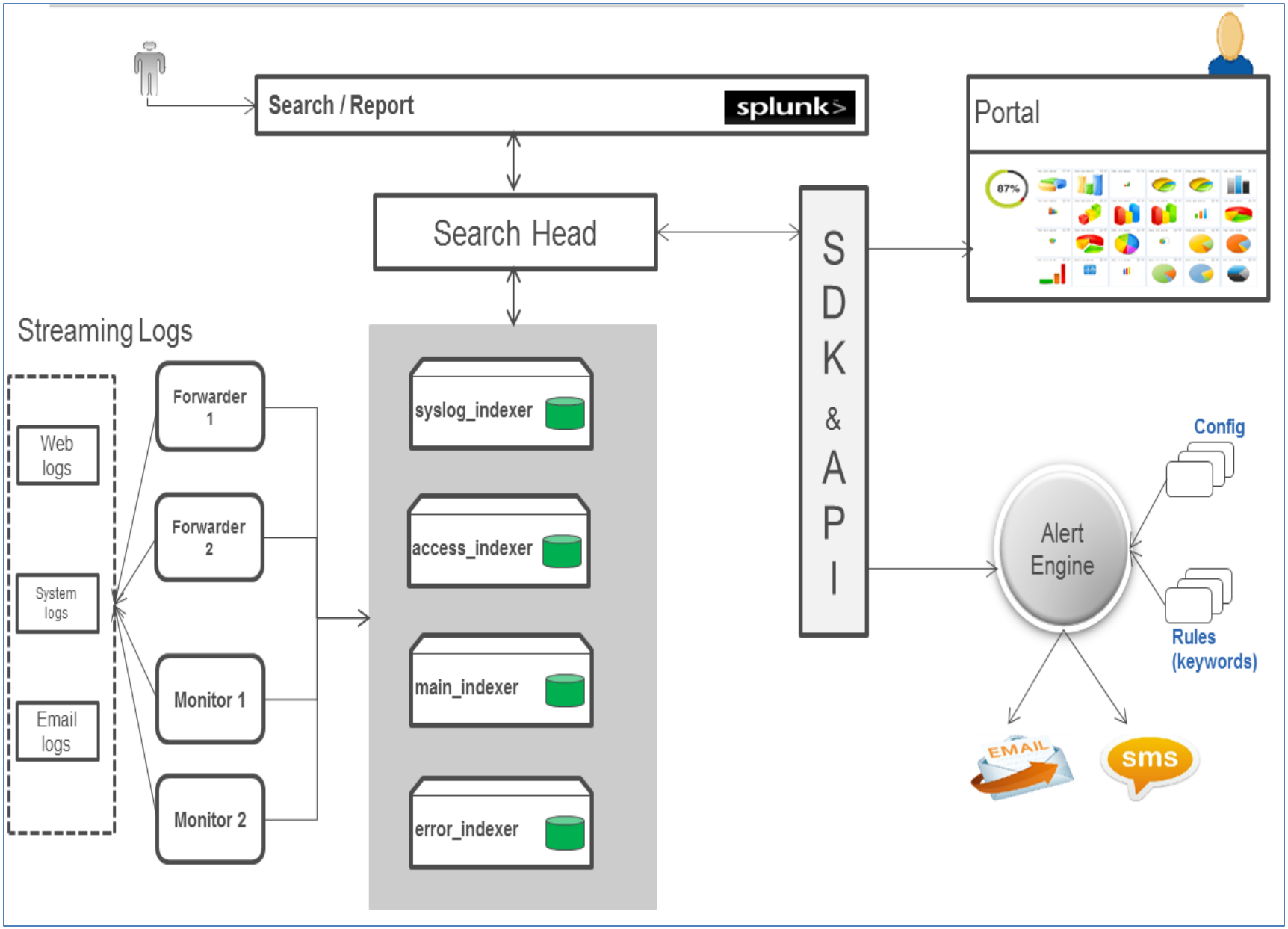
What are the Use Cases?
Development of a scalable system for the below 4 use cases -
Real-time streaming log monitoring and collection from multiple applications
Real-time streaming log processing in online mode and generation of events/incidents based on rules/keywords
Querying on the real-time data for finding events/incidents, reporting and visualization
Real-time log alerting/notification through different modes (Email/SMS)

Software and Hardware Details
Splunk Enterprise 6.2.x
Splunk Development Platform – SDK/APIs
Bootstrap 3.0
Jboss 7.1.1
JDK 7
Perl
Linux OS

What is our Solution?
Developed a scalable platform using the above softwares. we have used four different streaming log sources from accesslog,syslog,emaillog. Logs are being generated with the help of log simulators. Each of the streaming log source is being accessed/monitored by a forwarder or a monitor which reads the streaming logs using pull mechanism. Forwarders are created in Perl. The streaming data is ingested by respective indexers and stored in MongoDB. Here we have enabled a custom Dashboard for the users.

We have created a custom portal in Bootstrap which provides visualization and reporting capabilities for users. The custom Portal displays the total events, critical incidents with real-time graphs. Custom Portal is integrated using SDK/APIs and hosted on Jboss app server. The solution is integrated with an Alert Engine which sends the alerts based on the configured keywords.

Design and Implementation Architecture



How our solution meets the problem statement?

A. Scalability

- Scalability if streaming dataflow increases : By using Forwarders/Monitors – If Data volume increases we can increase the instances of forwarders and monitors
- Scalability if data volume increases : Increasing the capacity of the commodity hardware of Splunk platform– CPU,RAM, Disk
- Scalability if the users increases – By using Load balancers and multiple Splunk core engines
- Scalability if the Downstream systems increases – By using more scalable APIs

B. Realtime query on the streaming data

- By using Splunk Search and Reporting App and SPL

C. Realtime Alerting

- Using Splunk APIs
- Using Splunk Alerting which is part of the licensed version

D. User Experience and visualization

- Using custom Dashboard based on the user requirements
- Providing savedQueries, report Acceleration
- Realtime visualization with various types of graphs

Scalable Architecture