Centinel Proposal

Name

Streaming Data Handler - Centinel (A distributed reliable framework for collection, aggregation and analysis of streaming data)

Repo Name

centinel

Description

The Centinel project aims at providing a distributed, reliable framework for efficiently collecting, aggregating and sinking streaming data across Persistence DB and stream analyzers (example: Graylog, Elastic search, Spark, Hive etc.).

This framework enables SDN applications/services to receive events from multiple streaming sources (example: Syslog, Thrift, Avro, AMQP, Log4j, HTTP /REST etc) and execute actions like network configuration/batch processing/real-time analytics.

Core features of Centinel framework are:

- Stream collector - Collecting, aggregating and sinking streaming data
- Log Service - Listen log stream events coming from log analyzer
- Log Service - Enables user to configure rules (example: alerts, diagnostic, health, dashboard etc.)
- Log Service - Performs event processing/analytics
- User Interface - Enable set-rule, search, visualize, alert, diagnostic, dashboard etc.
- Adaptor - Log analyzer plug-in to Graylog and a generic data-model to extend to other stream analyzers (Logstash etc.)
- REST Services - Northbound APIs for Log Service and Steam collector framework
- Leverages - TSDR persistence service, data query, purging and elastic search

The following architecure depicts the core components of the Centinel:

blocked URL

Scope

The scope of this project (Centinel for OpenDaylight Beryllium) is as follows:

- Flume based framework for efficiently collecting, aggregating, and moving streaming data into different storage destinations like ODL Persistence DB, stream analyzer
- Implementation of Log service to listen/processing for log events coming from open source Graylog analyser
- Implements GrayLog plugin extending an abstraction layer for log analyzer
- Integrated User interface for features provided by "log service"(configure-rules, alerts etc)

Functional overview of features provided by "log service" are give below:

Feature	Details
Streams	• Mechanism to route messages into categories in real time while they are processed like stream for audit logs(install bundle etc.) • Rule configuration includes message, level, source etc. • Streams are generated by GrayLog server as per user-defined rules. Event-handler module handles streams from GrayLog server and persist it into Centinel

Alerts	• Log alerts :
	  ◦ Alerts get generated based on specific event matching in real-time
	  ◦ Alert condition types : Message count condition, Field value condition, Field string value condition
	  ◦ Alerts get cleared if specified condition does not persist. Alert check interval time is configurable, default is 60 seconds.
	• Common alert operations like manual acknowledge , delete, filter , sort etc. will be supported

Search and Analyze	• Support for search query language. • Time range for search can be specified • Visualization includes histogram
Dashboard	• Build pre-defined views on data by adding widgets. Domain expert can define search query and save results on dashboard. • Search result type : Search result counts, Search result histogram charts, Field value charts,Quick value results
Diagnostic	• Enables , user to specify/configure group of log messages (events or specific conditions) as single rule. Order of messages and condition for each messages can also be configured. • Notification will be persisted on db if log messages are coming out of order/sequence • On clicking notification on UI, popup window will open which displays expected order and received order of log messages. • User will be able to scroll popup window to see individual received log message. • Event processor has intelligence to verify sequencing of messages and generate diagnostic reports • Success notification is displayed when all specified conditions are true.
Health	• User specify list of messages with conditions for feature. • Feature health will be changed to Critical, Major, Warn on reception of all messages specified

Resources Committed (developers committed to working)

Sumit Kapoor < sumit.kapoor@tcs.com>
Rajender Joshi <rajender.joshi@tcs.com>
Shreshtha Joshi <Shreshtha.Joshi@tcs.com>
Rattenpal Amandeep <Rattenpal.Amandeep@tcs.com>
Abhishek Abhi <Abhishek.Abhi@tcs.com>
Sunaina Khanna <Sunaina.Khanna@tcsin.com>
Himanshu Yadav <Yadav.Himanshu1@tcs.com>
Swati Tyagi <Tyagi.Swati@tcs.com>

Initial Committers

Sumit Kapoor < sumit.kapoor@tcs.com> <ODL ID: sumitkapoor>
Shreshtha Joshi <Shreshtha.Joshi@tcs.com> <ODL ID: shreshthajoshi>
Rajender Joshi <rajender.joshi@tcs.com> <ODL ID: rajenderjoshi1>

Vendor Neutral

The project is made from scratch, no vendor code, logos nor is anything included.

Meets Board Policy (including IPR)

New Project. No Inbound Code Review required

See also

- Centinel:Main
- Centinel:Beryllium_Release_Plan
- File:Centinel-08132015.pdf
- File:CentinelBoronArchitecture.pdf