Centinel: Beryllium: Release Plan

Contents

Introduction

In Beryllium, we plan to develop "Log Service" and plug-in for log analyzer (i.e. Graylog). Log service will do processing of real time events coming from log analyzer. Additionally, stream collector (flume and sqoop based) that will collect logs from Opendaylight and sink it to persistence service (integrated with TSDR). Also includes Restconf interface to inject events to north bound applications for real-time analytic/network configuration. Centinel User Interface (web interface) will be available to operators to enable rules/alerts/dashboard etc.

Release Deliverable

Name	Description
Log Service	Plug-in to log analyzer i.e. Graylog.
Log Service	Log analyzer abstraction layer to support multiple plug-ins.
Log Service	Features support for rule configurations, streams, alerts.
Centinel North Bound	Provides REST interface to applications above Centinel's framework.
Web Interface	for visualization of data from Centinel.
Stream Collector	Flume based stream collection of semi-structure/unstructured data. Collects logs from OpenDaylight
Stream Collector	Sqoop based stream collection of structured data. Support for MySQL.
Stream Collector	Sink streaming data to persistence service.

(Conforming to Beryllium dates)

Release Milestones

Milestone	Offset 1 Date	Deliverables			
M1	7/30/2015	**Name**	**Status**	**Description**	
		Intent to participate	**Done**	Intent to participate in Beryllium Simultaneous Release	
		Candidate Release Plan	**Done**	Candidate Release Plan	
M2	8/27/2015	**Name**	**Status**	**Description**	
		Release Plan	**Done**	Final Release Plan	
		Centinel design document	**Done**	- Design document with reference to Centinel plug-in, "log service" and "log analyser abstraction" - with more emphasis on Alert feature - Define Yang model for configuration rules and events coming from Log analyzer and TSDR collaboration. Submit skeleton patch.	

M3	10/1/2015			

Name	Status	Description
Centinel skeleton	**Done**	Submit skelton for Centinel having plug-in, "log service" and "log analyzer abstraction"
Yang modelling	**Done**	Submit Yang models as applicable for Centinel
Centinel design document (incremental update)	**Done**	• Wire frame • Centinel - TSDR collaboration data model • Design approach for additional features - Diagnostics, Health, Dashboard, Stream and Steam Collector
Feature(s)	**Done**	• Log Service: Implementation of configuration module with Restconf support • Log Service: Event handler - Implementation of aggregate module • Log analyzer abstraction layer: Listen configuration rules and send it to plugin • Plug-in to Graylog: Send rules to Graylog server. Support for streams • Web Interface : Web interface development of configuration complete. • Data Change listener : support for alert and stream
Unit Test	**Done**	Unit Test development of each module
Integration and System test	**Done**	Simple system test on a karaf distribution with the project's recommended features installed

M4	11/12/2015			

Name	Status	Description
API Freeze	**Done**	• REST API from all Centinel services freeze
Feature(s)	**Done**	• Log service - Event handler : Aggregator complete. Event processor major functionality complete. • Plug-in to Graylog complete • Log analyzer abstraction layer : Publish all type of events to MD-SAL • Stream collector: Persist OpenDaylight logs to persistence DB • Stream Collector : Logback-flume client implementation(Collection of ODL/SFLOW logs) • Web interface : Support for streams , alert and dashboard. • Subscribe : Support for subscription of events • Data change listener : Integration of listener with persistence service and Web interface complete
Unit Test	**Done**	Unit Test development of each module
Integration testing	**Done**	Manual integration testing among stream collector, persistence, log service and Web interface
System test	**Done**	Run a simple system test on a karaf distribution with the project's recommended features installed on Code Merge
Documentation	**Done**	Create draft documentation of Centinel modules/interfaces

M5	12/17/2015			

Name	Status	Description
Code Freeze	**Done**	Centinel committed features all applicable code submitted
Documentation	**Done**	• Update wiki documentation to reflect changes/new features • Available user-guide with operational instructions on how to use Centinel for stream data handler
Feature(s)	**Done**	• Log service – Event handler : Aggregator , event processor and persist module complete • Integration with persistence service complete. • Stream Collector : Sqoop integration with persistence service complete
Unit Test	**Done**	Unit Test development of each module
Integration testing	**Done**	Manual integration testing among stream collector, TSDR persistence, log service, stream collector and Web interface
System test	**Done**	Run system test for Centinel and integrated with TSDR feature set

RC0	TBD			

Name	Status	Description
Deliverable Name		Deliverable Description

RC1	TBD			

Name	Status	Description
Deliverable Name		Deliverable Description

RC2	TBD	Name	Status	Description	
		Release Review		Release Review Description	
		Deliverable Name		Deliverable Description	
Formal Release	TBD	Name	Status	Description	
		Deliverable Name		Deliverable Description	

Externally Consumable APIs

- None Listed. Centinel creates own APIs based upon relevant data required.

Expected Dependencies on Other Projects

- Dependency on TSDR project

Expected Incompatibilities with Other Projects

None

Compatibility with Previous Releases

Changed APIs and/or Functionality

- Not applicable. This would be first release of Centinel in OpenDaylight Beryllium release.

Themes and Priorities

Requests from Other Projects

None Listed

Test Tools Requirements

- Java unit and integration tests

Other

See also

- Centinel Home Page
- Centinel Project Proposal
- **Centinel Beryllium Release Plan**
- Centinel-dev Archives
- File:Centinel - Design document M2Sep072015-InitialDraft.pdf