

Centinel: Boron: Release Plan

Contents

- [Introduction](#)
- [Target Use-Cases](#)
- [Release Deliverable](#)
- [Release Milestones](#)
- [Externally Consumable APIs](#)
- [Expected Dependencies on Other Projects](#)
- [Expected Incompatibilities with Other Projects](#)
- [Compatibility with Previous Releases](#)
 - [Changed APIs and/or Functionality](#)
- [Themes and Priorities](#)
- [Requests from Other Projects](#)
- [Test Tools Requirements](#)
- [Other](#)
- [See also](#)

Introduction

Aim of Centinel is to assist operators by providing visibility into the OpenDaylight management software defined network ecosystem (network via plugins, apps and infrastructure). To accomplish this, Centinel - Boron aims at a multi phased approach that includes:

- Data Collection: IPFIX, OpenFlow (from ODL-OF Plugin) and SNMP (from ODL-SNMP Plugin)
- Data Analysis: Using Kafka, Storm/Spark, Flume (replace Graylog)
- Analytics Dashboard: Inter-op with DLUX framework and ensure UX/UI (Graphana/Kibana)
- Machine Learning: Leverage Apache Mahout/SparkMlib to enable machine learning based on above analysis
- Additionally from usage-deployment, Embedded installation is proposed (Support for Embedded flume agent. Support for embedded installation of Query engine)
- Framework enhancement -

Enhance Centinel architecture to have two branches

A. Batch processing system(BPS)-BPS comprise of flume, hbase.

B. Real time analytics system(RAS)- RAS comprise of apache storm (Embed and distributed mode) for real time prediction.

In embed mode, storm will be used instead of Graylog server. In distributed mode, user will have both the options for log analysis.

Target Use-Cases

- Bandwidth utilization across WAN link
- Performance monitoring

Release Deliverable

Name	Description
Collectors	• IPFIX Collector: Collects IPFIX datagrams from Network for further analytic. • SNMP Collector: Collects data from SNMP plugin. • OpenFlow Collector: Collects OpenFlow datagrams from Network for further analytic.
Embedded installation and Framework enhancement support	• Embedded installation of Flume agent and query engine • Enhance Centinel architecture and replace Graylog with Storm. • Support for Distributed and Co-resident (Embedded) mode installation
Monitoring and Analytics	• Provides SDN infrastructure (Server, Network ,Application etc) monitoring and analysis • Dashboard to present Analytics and monitoring data
Predictive Analytics	• Machine Learning: Leverage Apache Mahout/SparkMlib to enable machine learning based on above analysis

Release Milestones

Milestone	Offset 2 Date	Deliverables		
M1	3/24/2016	Name	Status	Description
		Intent to participate	<i>Completed</i>	Intent to participate in Boron Simultaneous Release
		Candidate Release Plan	<i>Completed</i>	Candidate Release Plan
M2	5/5/2016	Name	Status	Description
		Release Plan		Final Release Plan
		Design document	<i>Completed'</i>	- Design document - Collectors (IPFIX collector, SNMP collector, OpenFlow collector) - Design document - Monitoring and Analytics.
		Feature(s)	<i>Completed</i>	IPFIX collector: Implementation of IPFIX collector.
M3	6/2/2016	Name	Status	Description
		Code skeleton	<i>Completed</i>	Submit skeleton for Enhancement features.
		Implementation	<i>Completed</i>	Submit architectural diagrams for the Centinel enhancement features.
		Feature(s)	<i>Completed</i>	- Framework Enhancement: Kafka consumer/publisher, Strom topology for real-time data analytics - Bandwidth utilization across WAN link, Performance monitoring: REST API defined
		Unit Test	<i>Completed</i>	Unit Test development of each module
		Integration and System test	<i>Completed</i>	Simple system test on a karaf distribution with the project's recommended features installed
M4	6/30/2016	Name	Status	Description
		API Freeze		REST API from all Centinel Services freeze
		Feature(s)		- Enhance Centinel architecture and replace Graylog with Storm. - Support for Distributed and Co-resident (Embedded) mode installation - Dashboard to present Analytics and monitoring data
		Unit Test		Unit Test development of each module
		Integration testing		Manual integration testing
		System test		Run a simple system test on a karaf distribution with the project's recommended features installed on Code Merge
		Documentation		Create draft documentation of Centinel modules/interfaces

M5	8/4/2016	<table><tr><th>Name</th><th>Status</th><th>Description</th></tr><tr><td>Code Freeze</td><td></td><td></td></tr><tr><td>Documentation</td><td></td><td> - Update Project wiki - User Guide, Installation Guide, Getting Started Guide and Others - Updated System Test approach and test cases Wiki page </td></tr><tr><td>Feature(s)</td><td></td><td> - Machine Learning (post Network monitoring and Analysis) and Predictive Analytics - End to End use-cases integration - Bandwidth utilization across WAN link, Performance monitoring </td></tr><tr><td>Unit Test</td><td></td><td>Unit Test development of each module</td></tr><tr><td>Integration testing</td><td></td><td>Manual integration testing.</td></tr><tr><td>System test</td><td></td><td>Run system test for Centinel</td></tr></table>	Name	Status	Description	Code Freeze			Documentation		- Update Project wiki - User Guide, Installation Guide, Getting Started Guide and Others - Updated System Test approach and test cases Wiki page	Feature(s)		- Machine Learning (post Network monitoring and Analysis) and Predictive Analytics - End to End use-cases integration - Bandwidth utilization across WAN link, Performance monitoring	Unit Test		Unit Test development of each module	Integration testing		Manual integration testing.	System test		Run system test for Centinel
		Name	Status	Description																			
		Code Freeze																					
		Documentation		- Update Project wiki - User Guide, Installation Guide, Getting Started Guide and Others - Updated System Test approach and test cases Wiki page																			
		Feature(s)		- Machine Learning (post Network monitoring and Analysis) and Predictive Analytics - End to End use-cases integration - Bandwidth utilization across WAN link, Performance monitoring																			
		Unit Test		Unit Test development of each module																			
		Integration testing		Manual integration testing.																			
System test		Run system test for Centinel																					
RC0	TBD	<table><tr><th>Name</th><th>Status</th><th>Description</th></tr><tr><td>Deliverable Name</td><td></td><td>Deliverable Description</td></tr></table>	Name	Status	Description	Deliverable Name		Deliverable Description															
Name	Status	Description																					
Deliverable Name		Deliverable Description																					
RC1	TBD	<table><tr><th>Name</th><th>Status</th><th>Description</th></tr><tr><td>Deliverable Name</td><td></td><td>Deliverable Description</td></tr></table>	Name	Status	Description	Deliverable Name		Deliverable Description															
Name	Status	Description																					
Deliverable Name		Deliverable Description																					
RC2	TBD	<table><tr><th>Name</th><th>Status</th><th>Description</th></tr><tr><td>Release Review</td><td></td><td>Release Review Description</td></tr><tr><td>Deliverable Name</td><td></td><td>Deliverable Description</td></tr></table>	Name	Status	Description	Release Review		Release Review Description	Deliverable Name		Deliverable Description												
Name	Status	Description																					
Release Review		Release Review Description																					
Deliverable Name		Deliverable Description																					
Formal Release	TBD	<table><tr><th>Name</th><th>Status</th><th>Description</th></tr><tr><td>Deliverable Name</td><td></td><td>Deliverable Description</td></tr></table>	Name	Status	Description	Deliverable Name		Deliverable Description															
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

None

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 Boron 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 Boron Release Plan**
- [Centinel-dev Archives](#)
- [File:Centinel - Design document M2Sep072015-InitialDraft.pdf](#)