Savanna Test Plan

Rev. 1

MIRANTIS

Contents

1 Revision History

Date	Author	Comments
12/03/2013	QA team	Initial Draft

2 Introduction

The main goal of this document is to present a detailed test cases for Savanna testing. This test plan includes functional test cases for the features and non-functional test cases for performance, load and compatibility testing types. This document is divided to sections by the testing types.

3 System Under Test Specifications

- OpenStack Folsom
- Savanna version 0.1
- Python version 2.7.6

4 QA Team Responsibilities

QA team is responsible to:

- make a test plan contains acceptance, functional, performance, scalability, reliability and compatibility test cases
- add regression test cases to the test plan in case of regression
- execute acceptance and regression tests for release candidate
- execute all automated functional, scalability, reliability and compatibility test cases on stable nightly builds when corresponding features will be implemented
- automate as many acceptance tests as possible
- maintain infrastructure for making nightly builds and run automated tests
- notify Dev team about found bugs via bug tracking system
- help Dev team with bug reproduction
- verify bugs fixed by Dev team
- community support about QA and automation on the project

4.1 Test Deliverables

The following items should be delivered after product release:

1. test plan
2. test cases specification
3. test cases execution reports
4. project status and metrics

4.2 Test Schedule

1. Basic Cluster Provisioning - end of April 2013
2. Cluster Operations - middle of July 2013
3. Analytics as a Service - middle of October 2013

4.3 Test Cycles

1. Implemented functionality (pass 1)
2. Working functionality (pass 2)
3. All critical bugs and bugs related to functionality fixed - acceptance for demo (pass 3)

5 Suspension Criteria and Resumption Requirements

The main criteria to suspend the testing for candidate build is having at least one 'Blocker' bug. Resumption criteria in that case is fixing all 'Blocker' bugs and make a new candidate build. The second criteria to suspend the testing is instability or unavailability of the test lab. Resumption criteria in that case is fixing all issues and problems with the test lab. Performance testing can be suspended if the product is unstable on the load. Resumption criteria for that case is fixing corresponding bugs and make a new stable build.

6 Risk Areas Evaluation

Risk Areas were evaluated by their severity and priority. These potential risks will be addressed by tests. Number of test cases for each feature is determined by the priorities defined on the basis of severity and probability of risks, related to this feature.

Each test case has a priority according to and probability of a particular risk, related to this test case. Also each test case has a priority for automation.

#	Risk	Severity	Probability	Comments
1	Difference between REST API requirements and implementation	High	Low	The following items must be tested: • Node template requests: create, modify, delete, get, list • Cluster requests: create, modify, delete, get, list
2	Errors in the REST API implementation	High	High	The following items must be tested: • Deploy acceptance • Hadoop acceptance • REST API requests both for node templates and clusters: get, list
3	Errors in the plugin API functionality	High	High	REST API requests should be tested for the following Hadoop implementations: • Apache Hadoop • Hortonworks Ambari • Clouderra Manager • Intel Hadoop
4	Errors in the UI functionality	High	High	The following items must be tested: • Node template requests through Horizon UI: create, modify, delete • Cluster requests through Horizon UI: create, modify, delete • TBD
5	Savanna is not compatible with custom Hadoop images	Medium	High	Create a Hadoop cluster using Hadoop images provided by Hortonworks Ambari, Clouderra Manager, Intel Hadoop, etc
6	Savanna is not compatible with different OpenStack releases	Medium	High	Savanna should be installed on the upcoming OpenStack releases(TBD which)
7	Savanna is not compatible with different versions of Python	High	High	Savanna should be tested on Python 2.6 and 2.7. Also it should be tested on 3.x versions when the OpenStack does support them.
8	Installation problems	High	Low	Savanna supports 2 installation modes: from the sources and using pip utility. Both of them should be tested.
9	Lack of UI performance	Medium	Medium	TBD

10	Low performance of Hadoop deploying	Medium	Medium	TBD
11	Low performance of Apache Hadoop HDFS	Medium	Medium	TBD
12	Low performance of Apache Hadoop MapReduce	Medium	Medium	The following scenarios must be tested: • Pi evaluation • Load test script(TBD name?)

7 Features To Be Tested

Basic Cluster Provisioning

- Cluster provisioning
- Deployment Engine implementation for pre-installed images
- Templates for Hadoop cluster configuration
- REST API for cluster startup and operations
- UI integrated into Horizon

Cluster Operations

- Manual cluster scaling (add/remove nodes)
- Hadoop cluster topology configuration parameters
  - Data node placement control
  - HDFS location
  - Swift integration
- Integration with vendor specific deployment/management tooling
- Monitoring support - integration with 3rd-party monitoring tools (Zabbix, Nagios)

Analytics as a Service

- API to execute Map/Reduce jobs without exposing details of underlying infrastructure (similar to AWS EMR)
- User-friendly UI for ad-hoc analytics queries based on Hive or Pig

Further Roadmap

- HDFS and Swift integration
  - Caching of Swift data on HDFS
  - Avoid issues with Swift eventual consistency while running job
- HBase support

8 Features NOT to be tested

Feature	**Reason**
Full regression of the Hadoop	Scope of Savanna project doesn't include implementation of the Hadoop functionality

9 Test lab network topology

9.1 Descriptions of network topology and servers destination

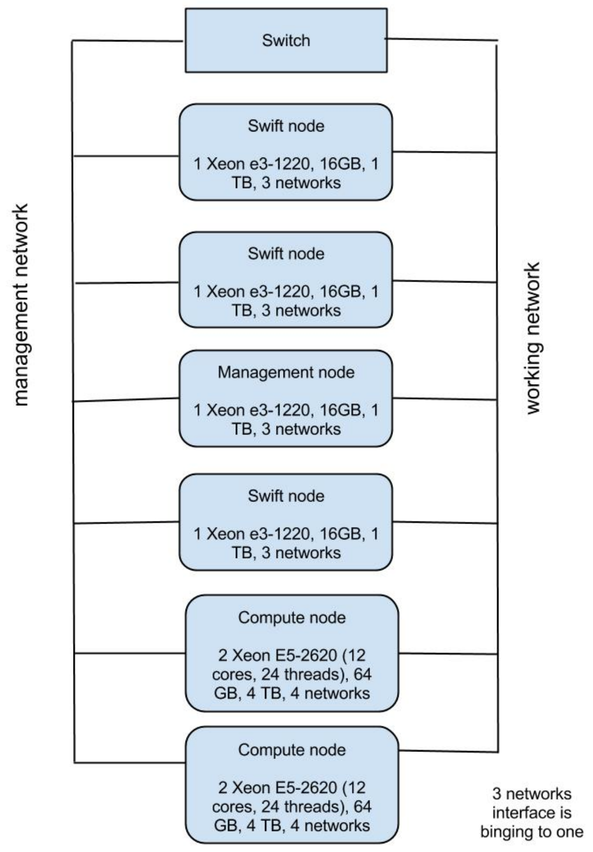
Virtualized infrastructure is used for Savanna testing presented on the Figure 1.

- *Switch* - usual a switch-device.
- *Swift node* - on this nodes can run the virtual machine.
- *Management node* - this node manage lab.
- *Compute node* - on this node run the virtual machines.

9.2 Network scheme for tests

This scheme is used in all test cases for Savanna.

Figure 1: Network scheme for tests


Switch
Swift node
1 Xeon e3-1220, 16GB, 1 TB, 3 networks
Swift node
1 Xeon e3-1220, 16GB, 1 TB, 3 networks
Management node
1 Xeon e3-1220, 16GB, 1 TB, 3 networks
Swift node
1 Xeon e3-1220, 16GB, 1 TB, 3 networks
Compute node
2 Xeon E5-2620 (12 cores, 24 threads), 64 GB, 4 TB, 4 networks
Compute node
2 Xeon E5-2620 (12 cores, 24 threads), 64 GB, 4 TB, 4 networks
management network
working network
3 networks interface is binging to one

10 Test Cases Requirements

- All possible test cases should be automated.
- At least 70% of the test cases should be automated.