Bolted System: Auto-deployment Cloud Project

Vidya Anandamurali
Pei Jia
Yuxi Jiang
Jiangnan Zou

Project Description (Recap)

Automate the deployment of Bolted which consists of:

- Installation and configuration of each component of the Bolted system (HIL, BMI, Keylime and orchestration)
- Integrating all the components together
- Bolted system should be able to be installed on any cloud platform

BU

Use Case

As a cloud provider, I want to deploy Bolted System on my cloud platform easily without having to call upon a personnel to configure the components.

Minimum Viable Product

- To deliver a system that doesn't require separate installation and configuration procedure for each component of the bolted system
- Develop an ansible playbook for each component first and then an overall playbook for all four components to work together.

BU

Last Sprint Report

- ➢ Install HIL on Virtual Machine (CentOS/VMware)
  - ✓ Install through terminal command line
  - ➢ Automate installation using Ansible
    - ○ Unittest
- ➢ Install Keylime on Virtual Machine (CentOS/VMware)
  - ✓ Install through terminal command line
  - ✓ Bash script automated installation
  - ➢ Test script in another virtual machine
- ➢ Install BMI on Virtual Machine (CentOS/VMware)
  - ➢ Install BMI on a CentOS environment contains HIL
    - ○ Automate installation using Ansible
    - ○ Unittest
- ✓ Learning Ansible

BU

Project Progress of Keylime

- TPM (Trusted Platform Module) This is the core module of Keylime, because keylime uses this module to collect node information, (operating system, application and etc), using [quote].
- For now, installation is under virtual machine, which doesn't contain a TPM module. Testing is needed for later test.
- A bash script is developed to automated the installation procedure, targeted for [CentOS VM using vmware]
- Testing passed, bash to ansible script is in progress.

Project Progress of HIL

- Installing CentOS on VMware workstation.
- Setting up the server upon which HIL runs. (epel and python - using pip)
- Configure HIL (hil.cfg)
- Setting up the HIL database
- Starting the server
- Testing the setup

```
(.venv)[Vidya@rfc1918 hil]$ cd hil/
(.venv)[Vidya@rfc1918 hil]$ ls
api.py              cli.py          dev_support.pyc  __init__.py     network_allocator.py
api.pyc             cli.pyc         errors.py        __init__.pyc    network_allocator.pyc
auth.py             commands        errors.pyc       migrations      rest.py
auth.pyc            config.py       ext              migrations.py   rest.pyc
class_resolver.py   config.pyc      flaskapp.py      migrations.pyc  server.py
class_resolver.pyc  deferred.py     flaskapp.pyc     model.py        server.pyc
client              dev_support.py  hil.db           model.pyc       test_common.py
(.venv)[Vidya@rfc1918 hil]$ which hil
~/hil/.venv/bin/hil
(.venv)[Vidya@rfc1918 hil]$ _
```

BU

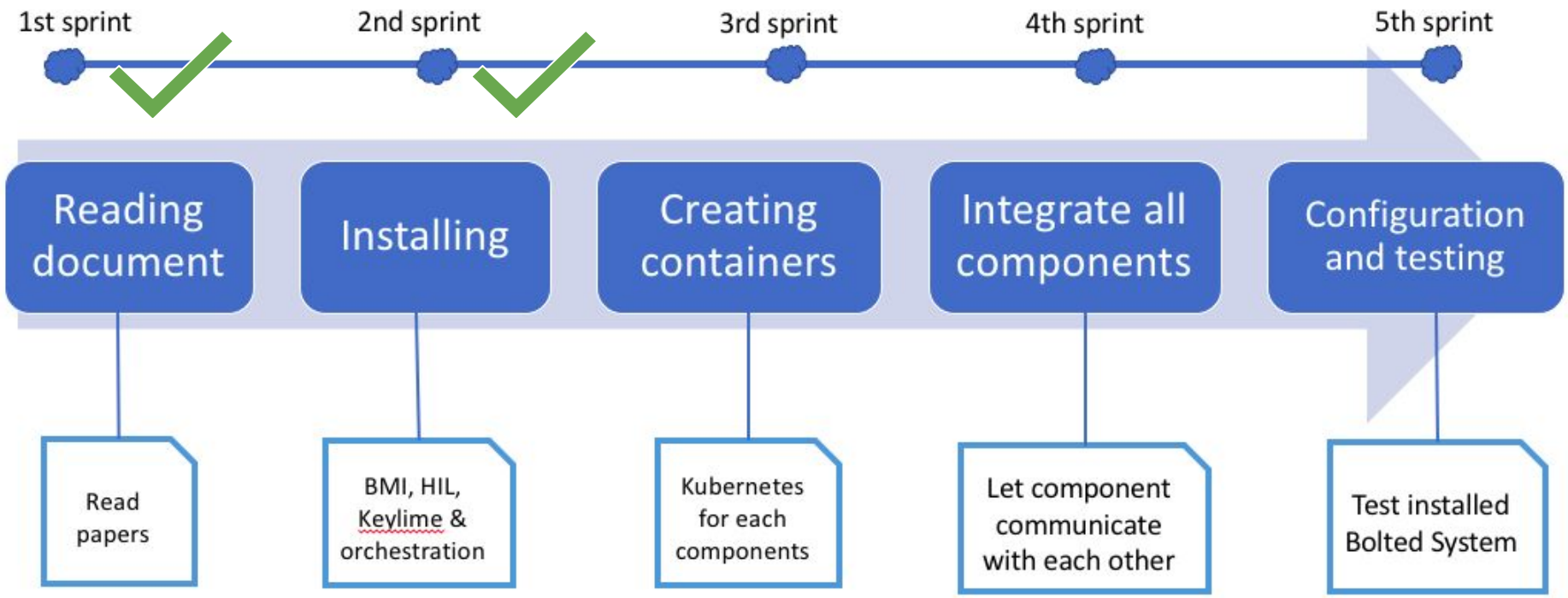
Project Progress of BMI

- Set up CentOS 7.0 on virtual machine
- HIL environment checking
- Installing the Ceph Client
  - Learn Ceph server and client simultaneously
- Configuring iSCSI Server
  - Deal with system version inconsistency
- Configuring DHCP Server
- Installing BMI
  - Did not settle File missing problem yet

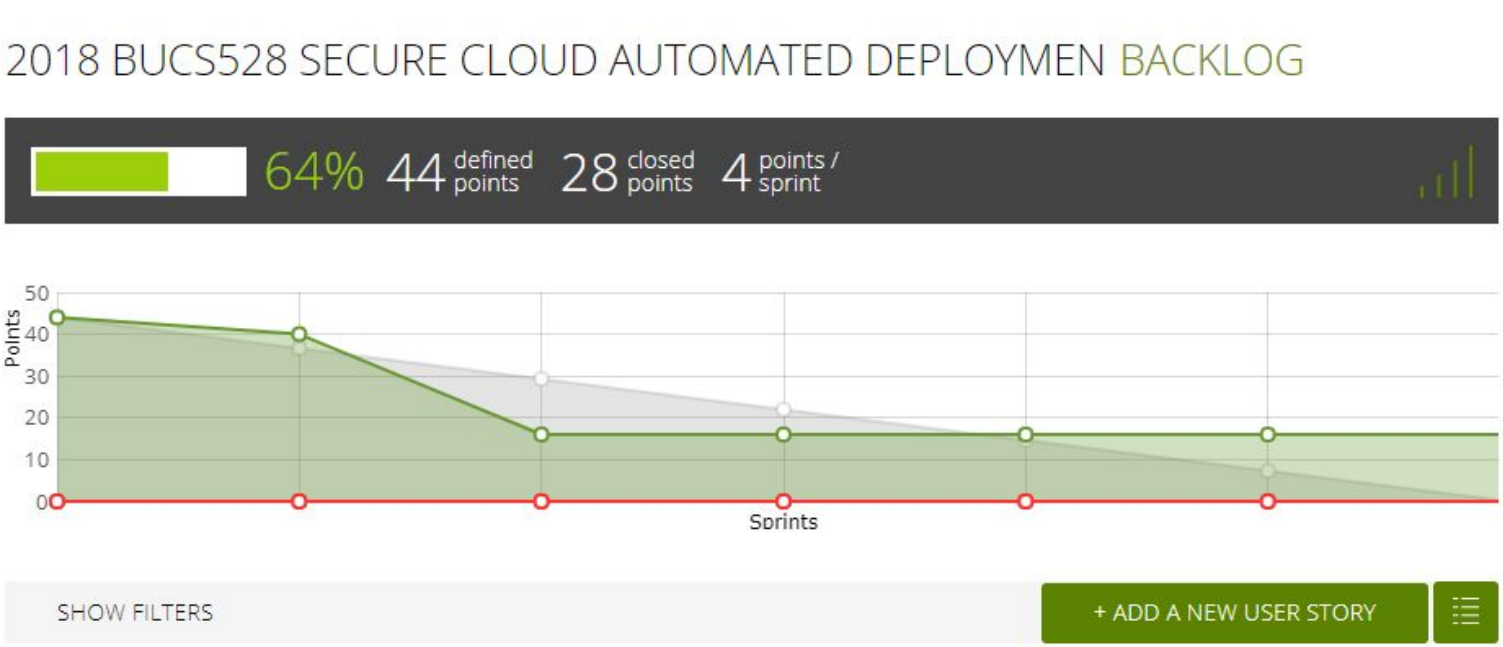
BU

Demo

Project Plan



Burndown Chart



Responsibilities for next sprint

- Write Ansible playbook script for installation of each component in CentOS virtual machine.
- Learn Kubernetes.
- Creating containers for each component using Kubernetes

Thank you.

Question ?