TF DDF 2019: Contributor Experience discussion and feedback

Date

18 Nov 2019

Minutes

- VM (Vicky) Brasseur (she/her) & Will Stevens
- Syed Ahmed
- Lisa Caywood
- Casey Cain
- Ian Rae
- Darien Hirotsu
- We're opening tickets as soon as they come up! See our Jira.
  - https://jira.tungsten.io
- Today's focus will be a discussion on contributing to TF.
  - Docs needs a LOT of help. 🙂
  - Let's focus on what we can do to make it suck less
- Leading question: How many of you have installed Tungsten Fabric?
  - A number of "yes's"
  - Tools used:
    - tf-devstack
    - Contrail Ansible & fab scripts
    - CloudOps Automation Tooling
      - https://github.com/cloudops/tf-demo
- Next leading question: Were you able to get it to work twice in a row?
  - Mostly "no's"
- What should the process be?
  - Submit a Blueprint to Jira
  - Defend the Blueprint at the Technical Worksteam Meeting
  - Once the Blueprint is accepted, create a Spec and land it in Gerrit
  - The PTL for the project in question reviews the Spec and approves it, so it can be implemented
    - Note that there are challenges with this since well... we don't have PTLs (also we need to determine what the projects are)
  - Implement the Spec once the Technical Worksteam
    - If you are a first time contributor, submit an ICLA or CCLA via Gerrit (assumes that all repositories have been moved)
  - Once the initial implementation is complete, test the implementation locally
    - This is a challenge given the state of tf-devstack, but let's start here (see
  - Submit a Gerrit patch once the implementation is complete and tested locally
    - Note that there are challenges with docs given the state of Zuul
  - Once you have +2, the code is merged
  - Note here, there is no developer guide to navigate this
    - Developer guide should:
      - Where and how to setup a bug/blueprint
        - Templates
      - Have a section to test a change locally (See: https://jira.tungsten.io/projects/TFP/issues/TFP-89?filter=allopenissues)
      - Have a Gerrit reviewer section (how to review a patch)
      - Have a patch submission (Gerrit) section
- Many of us have tried to install tf-devstack to push a patch
  - WS: I've tried to VERY hard to get tf-devstack in a place where I can test my patches to feel confident that my code works
    - Lots of contorting and pain result from this exercise
    - The best place we landed was building the artifacts and pointing Contrail Ansible Deployer to a ~private instance of Docker Hub with those artifacts
  - WS: Also, there are two architectures that result from the deployers.
    - Docker compose (Contrail Ansible Deployer)
      - HA is an issue with this deployment
      - No other CNI does it this way
      - Our guess is that Kubernetes was not in the picture when the deployer was created
    - Kubernetes manifest (One click guide)
      - These manifests haven't been updated in a year
      - They don't work any longer for Kubernetes 1.16 (Stateful and Daemon sets changed)
      - This is the way all other CNIs are built (and how TF should do it)
  - WS: Another challenge is building the manifest (see BluePrint)
    - https://jira.tungsten.io/projects/TFP/issues/TFP-89?filter=allopenissues
    - There is a shell script to help but the variables are inconsistent and it has to run on a host where the target CNI shall be installed
  - WS: If we can systematically build a manifest based on variables, then it's much easier to create a local dev environment
    - As an artifact, it also helps us integrate with deployers that DevOps engineers use like RKE
      - https://github.com/rancher/rke
    - It was not clear that the BluePrint created needed to be discussed in the Technical Workstream Meeting (oops)
      - Also, the TSC and the TWS members need better accountability to ensure that BluePrints don't die on the vine
  - Blueprint process issues

- - - Need clarification on the mapping of Blueprints to specs and releases
    - Specs are the expected implementation of a Blueprint
      - PTLs will then have input on the spec
      - PTLs have the responsibility of ensuring technical issues like backwards compatibility
- CLA
  - We mostly have this figured out (thank you VM (Vicky) Brasseur (she/her) and Will Stevens)

Action items

- ☑ Casey Cain work with the TSC and TWS to drive the process with Prabhjot Singh Sethi to ensure BluePrints move through the process correctly (example issue: Will Stevens was not aware that he had to defend his blueprint with the TWS)