# Making TF Cloud Native

## Date

25 Jun 2020

## Attendees

- @Prasad Miriyala (Presenter)
- Sukhdev Kapur (Presenter)
- Casey Cain (Host)
- sagar Chitnis
- Derek Siiwa
- Frederick Kautz
- Herakliusz Lipiec
- Prabhjot Singh Sethi
- Krzysztof Kajkowski
- Sai Pujita T
- Mehmet Toy
- Victor Morales
- Stephan Garcia
- Soujanya R M
- Sivakumar Ganapathy
- Sangarshan P
- Richard Roberts
- Pranavdatta DN
- Parth yadav
- Mahdi Ahmed Jama
- Eric Tang
- Kiran KN
- Sudheendra Rao

## Overview

We will cover several short term and long term initiatives to make TF cloud-native, such as CRDfication, Operator Framework, TF Mesh (service mesh), etc

- Schedule: https://teamup.com/event/show/id/gUkLBrcDcwSyyefzWSw9yz7GNzsYLN
- Recording: Making TF Cloud Native.mp4
- Presentation: https://drive.google.com/file/d/16LE21d8FKMbjIsQfE9-FSDXBbSna27O5/view?usp=sharing

## Minutes

Prasad and Sukhdev presented the motivation for making TF cloud native and a bunch of possible items we could pursue to achieve the same.

Sukhdev elaborated on the following –

1. Importance of Cloud nativeness in the context of 5G as the whole industry and telcos are moving towards this.
2. Brief summary of what is needed to make TF cloud native.
3. Various components that make 5g work
    a. AMF, SMF and UPF are the critical components. Minimum needed for 5G to work.
    b. All interfaces among the compoents are standards based. They could come from various vendors and hence it is a completely disaggregated architecture. These are actually a bunch of pods that could be running anywhere.
4. Imagine these are pods/VMs running on TF. How do you facilitate the communication?
    a. Enhance TF datapath to support these next-gen applications.
    b. We have to build envoy kind of proxy.

Prasad then elaborated on each of the possible areas we could pursue to achieve cloud nativeness.

Prasad also suggested to the community that we could create Jira tickets and take these items forward.

**Life Cycle Management**

Operator framework: Bring up contrail in a simple manner. TF operator will be open source.

We could look into Operator Framework not just for Contrail, but also for services (CNF and VNF).

**Control Plane**

Service discovery: This plays a role in cloud nativeness.

CRDfication:

1. Service Chaining
    a. Especially in 5G we can concatenate services that we already have in TF.
    b. Not just network based Service Chaining. But also tag based Service Chaining.
2. Zero trust policy - With tag based policies, TF already has the building blocks for this.

1. Generating policies by looking at traffic.
2. Adding new intents. Does it disrupt? Verify them before committing. Verify against already seen traffic.

iii. Policies into draft mode and getting it approved by an admin. Don't disrupt an existing app.

1. Tag based encryption: Already supported between vRouters. All traffic goes through encryption. We may not want to encrypt all. We could selectively encrypt traffic using tags or CRDfication.
2. Service mesh integration with envoy proxy: Send using CRDs to Envoy proxy from TF.
3. Operator framework for services: Today a lot of functions need to be done when defining Service Chaining between two 2 VN, eg., port-tuples. We can have some kind of automation. Especially in case of k8s, lots of nuances where it fails. Operator framework could help in creating services, monitors them and automatically create port tuples if they fail. Helps onboarding customer services to TF.

Light wight TF in k8s

Consume VMs also in k8s using Kubevirt

Explore Kata for achieving security of VMs but agility of containers

**Datapath Enhancements**

Need higher throughputs

1. SRIOV would give a lot of benefits
2. DPDK for containers
3. Smart NIC integration (working, need more enhancements)

Nested K8s

1. vCNFs in an intermediary in the move from VNF to CNF.
2. When we do vCNF functions, there are cases where we have to learn mac-ip. This is needed for giving better options for vCNF.

Envoy proxy integration

1. We need to plugin Envoy proxy into datapath.
2. Can be done at a higher level with CRDfication or with HBS we already have.

vRouter to vRouter Encryption

1. We have IPsec based approach way. We could explore using Wireguard.
2. How can we use tag based encryption? Tag based encryption rather than encrypting all.

**Action Items**

Frederick suggested exploring SPIFFE and SPIRE (https://spiffe.io). It provides secure identities in the form of certificates for workloads.

## Action items

- [ ]