Secure Networking with Kubernetes, OpenStack, and Bare Metal

Gregory Elkinbard
ONS Amsterdam, September 2018
Agenda

- Brief Overview of Tungsten Fabric and Community
- New Security Model for TF 5.x
- TF OpenStack Integration
- TF Kube Integration
- Bare Metal Support
Mission

Build the world’s most ubiquitous, easy-to-use, scalable, secure, and cloud-grade SDN stack, providing a secure network fabric connecting all environments, all clouds, all people.

https://tungsten.io/
• 2013-Today: >300 years of work
• 200-300 developer contributions
• ~100 active developers
• Languages: C++, Python, Node, Go
• Apache 2.0 license
• Part of the Linux Foundation Networking
• GitHub repositories
• Gerrit review processes
• Launchpad bug tracking and blueprints
• Other OSS used: Cassandra, Kafka, HAproxy, Docker, Keystone
Features

Routing & Switching
(IPv4, v6)

Network Services
(IPAM, DNS, DHCP, SNAT, FIP, QoS, BGPaaS)

Load Balancing
(customizable ECMP, LBaaS)

Security & Policies
(Policy Enf., Distributed FW, Sec Grp, XMPP Encrypt.)

Perf & Scale
(DPDK / SRIOV, Smart NIC, Infra scale)

Gateway Services
(L2, L3, Software GW)

Rich Analytics,
(Alerts, Overlay-Underlay Correlation, multi-region)

Service Chaining
(PNF, VNF, v6, 3rd party / TAP, Health-check, policy-based, SFC Failover)

HA, Upgrades, Prov.
(Infra Failover, ISSU)

APIs & Orchestration
(multi-vendor Orch., SDN-U, OpenStack, K8s, vCenter)
Tungsten Fabric as SDN Controller

RULE THEM ALL WITH ONE
automated secure open SDN Controller

Public & Private
IaaS

openstack
aws
kubernetes

CaaS & PaaS
MESOS
OPENSHIFT

VMs or Metal
vmware

KVM

Public & Private
IaaS

CaaS & PaaS
VMs or Metal
Tungsten Fabric Single SDN for VMs & PODs

Basic Networking:
- L2/L3 or L2/L3 Network
- IPAM/DHCP, DNS, Multi-Tenancy

Advance Networking:
- VLAN-ID, VRRP, VIP, LB, Routes Advertisement,
- GW Function, Service Chaining, Traffic Steering, Flow awareness,
- QoS, SR-IOV/DPDK, BGP-VPN,
- Inter Site Federation, Health Checks, FW, IPSec/TLS Support
Uniform Network and Security Policy

Consistent security and network functionality between VMs, containers, or bare metal.

L4 Policy

Svc Chain Policy

Tungsten Fabric network and security policies provide fine grain traffic control, while abstracting away the underlay topology.
Tungsten Fabric Deployment Model

- Delivered as MicroServices
  - Docker Containers
  - Host dependencies in Privileged Installer Containers
- Common Installers
  - Helm
  - Ansible
  - Kolla
  - OpenStack Platform Director/TripleO
  - Mirantis MCP
  - Juju/Charms
  - OpenShift
- Latest Release on DockerHub
  - https://hub.docker.com/u/tungstenfabric/

DaemonSet, Ingress Services with Host Networking with choice of run single or multiple containers per PODs
Intent Based Security Policy

Policy Example:

allow web-traffic-group tier=web > tier=app match deployment && site

Objects at different levels can be tagged

Tags can be defined at different levels
- Global
- Project
- Network
- VM / Container / BMS
- Interface

Policies will finally be enforced at the interface level
CONSISTENT POLICY ENFORCEMENT

Tungsten fabric provides a rich, consistent set of security policy capabilities across multiple platforms.

1. Simplified Manageability *(change control, etc. is much easier)*
2. Improved Scalability
3. Define / Review / Approve Once → Use Everywhere
Tungsten Fabric Bare Metal Support
Bare Metal Integration

- TF enables Legacy VLAN based architecture **interconnecting** with a Cloud architecture
- Does not need a gateway when going from one VN to another on the TF overlay
Tungsten Fabric and OpenStack
TF and OpenStack Integration

Orchestration System (Open Stack) - Neutron

OSS/BSS - GUI

Application 1 - Application N

REST APIs

Contrail System
  - Configuration
  - Analytics
  - Control

East-West Peering Interface (BGP)

Management layer

Virtualized Server
  - VM
  - VM
  - VM

vRouter - Hypervisor

Gateway Router

IP fabric (underlay network)

XMPP

BGP + NetConf

MPLS over GRE, MPLS over UDP, or VXLAN
Integration Details

- Neutron Plugin
  - Production Stable
  - New for TF 5.0 - Direct Connect non-overlay mode

- ML2 Driver
  - Supports multi-SDN in OpenStack
  - Code is stable but not production tested
  - Lacks feature parity
Tungsten Fabric
Kubernetes Support
Tungsten Fabric Integration with k8s

Namespace: kube-system

- API Server
- Controller/Replication Manager
- etcd
- Scheduler
- Discovery
- Dashboard
- Contraill-kube-mgr
- Contraill Controller
- Contraill Analytics

* Contraill-Kube-manager listens to K8s API Server and conveys the API request to Contraill Controller

POD 1
- Kubelet
- CNI Plugin
- vRouter (replaces kube-proxy)

POD 2
- Kubelet
- CNI Plugin
- vRouter (replaces kube-proxy)

POD 3
- Kubelet
- CNI Plugin

POD 4
- Kubelet
- CNI Plugin
DIFFERENT LEVELS OF ISOLATION

DEFAULT CLUSTER MODE
- This is how Kubernetes networking works today
- Flat subnet where -- Any workload can talk to any other workload

NAMESPACE ISOLATION
- In addition to default cluster, operator can add isolation to different namespaces transparent to the developer
- In this mode, each POD is isolated from one another
- Note that all three modes can co-exist

POD / SERVICE ISOLATION
- In this mode, each POD is isolated from one another
- Note that all three modes can co-exist
Getting Started with Tungsten Fabric
Getting Started

https://tungsten.io/start/

Carbide SandBox for Amazon AWS
- Quick Kube Testbed for public clouds

Onprem with OpenStack
- Easy TF/OpenStack Deployment and Integration
Join the Community

Help Drive the Future