

Contents

1	Entorno de emulacion	2
1.1	Motivacion	2
1.2	Herramientas	2
1.3	Eleccion de la herramienta	3

Chapter 1

Entorno de emulacion

1.1 Motivacion

Uno de los principales objetivos del proyecto es realizar pruebas funcionales y de escala sobre el prototipo. Para poder realizar estas pruebas, resulta necesario construir un buen entorno de simulación o emulación que cumpla con los requerimientos del prototipo.

1.2 Herramientas

Se estudió el estado del arte en lo que respecta a opciones de emulación o simulación para SDN. A continuación se detallan las principales.

NS-3

Estinet

Mininet

Mininet es un emulador de redes SDN que permite emular hosts, switches, controladores y enlaces. Mininet utiliza virtualización basada en procesos para ejecutar múltiples instancias (hasta 4096) de hosts y switches en un único kernel de sistema operativo. También utiliza una capacidad de Linux denominada *network namespace* que permite crear "interfaces de red virtuales", y de esta manera dotar a los hosts con sus propias interfaces, tablas de ruteo y tablas ARP. Lo que en realidad hace Mininet es utilizar la arquitectura *Linux container*, que tiene la capacidad de proveer virtualización completa, pero de un modo reducido ya que no requiere de todas sus capacidades. Mininet también utiliza *virtual ethernet (veth)* para conectar los

nodos.

Tambien se estudiaron posibilidades no orientadas a priori a SDN, pero con potencial dada su alta configurabilidad.

LXC

Esta

Máquinas virtuales

1.3 Eleccion de la herramienta

Una de las primeras observaciones que se hizo fue que ninguna herramienta de las estudiadas cumplia todos los requerimientos que el prototipo necesita para ser emulado o simulado correctamente. Los principales requerimientos que debia cumplir el entorno eran:

- Soporte para Quagga: Dado que la arquitectura del prototipo es dependiente en que los routers puedan correr esta suite de ruteo, resulta crucial que el entorno de emulación pueda cumplir con dicho comportamiento. Sin embargo, el concepto de ruteo de capa 3 es algo ajeno al paradigma SDN, y por lo tanto no es contemplado por las principales herramientas de simulación o emulación para SDN.
- OpenFlow 1.3: La aplicación que implementa VPNs sobre el prototipo depende de que los switches OpenFlow tengan soporte para MPLS. OpenFlow ofrece MPLS a partir de la versión 1.3, pero no todas las herramientas ofrecen soporte para OF 1.3.
- Facil de usar: Es importante que el entorno pueda generar distintas topologias y escenarios sin demasiado esfuerzo de configuracion.
- Escalabilidad: Dado que uno de los objetivos de las pruebas es realizar pruebas de carga, el entorno deberia ofrecer buena escalabilidad.

ns-3 fue descartado debido a que no ofrece soporte para Quagga ni OpenFlow 1.3 al momento de realizar esta investigacion.

La opcion de crear nodos con Linux containers o maquinas virtuales resuelve el problema de Quagga y OF 1.3, pero llevaria una gran cantidad de trabajo construir distintas topologias (sobre todo si son grandes), ya que casi todo debe ser configurado manualmente por el usuario.

Estinet requiere licencias pagas, y se opto por elegir herramientas open source.

La herramienta que se eligió fue Mininet. *Out of the box*, Mininet ya cumple tres de los cuatro requerimientos explicados anteriormente. Está diseñada para ser escalable, ya que usa containers reducidos, tiene soporte para OpenFlow 1.3, y es muy fácil de usar. El aspecto en el que falla es soporte para Quagga. Dado que los switches de Mininet no tienen su propio network namespace, es imposible hacer que corran la suite de ruteo. Por otro lado, los hosts de Mininet sí lo tienen. Por lo tanto, la alternativa que se eligio es la de usar los containers reducidos que Mininet ofrece como hosts, y agregarles Quagga y OVS para que se comporten como los routers del prototipo. Mininet esta programado con orientacion a objetos y tiene una buena API, asi que alcanza con definir nuestras propias clases de nodos (RAUSwitch, RAUHost, RAUController y QuaggaRouter) derivandolas desde la clase Host de Mininet y agregandoles los servicios o características que falten.