

Laboratorio Controlador OpenDayLight

Contenido

Laboratorio – Controlador OpenDayLight.....	2
Objetivo Práctica Laboratorio	2
Requisitos Laboratorio	2
Actividades Laboratorio	2
Instalación OpenDayLight	2
Acceso Web.....	4
Simulación con Mininet y OpenDayLight	5
Bono	7
Informe Laboratorio	7
Referencias.....	8

Laboratorio – Controlador OpenDayLight

Objetivo Práctica Laboratorio

INTEGRAR el controlador SDN OpenDayLight con una red desplegada en la herramienta mininet.

Requisitos Laboratorio

Para el correcto desarrollo de este laboratorio es necesario contar con lo siguiente:

- Máquina virtual del Laboratorio - Preparación Ambiente.
- Mininet.
- Acceso a Internet.

Actividades Laboratorio

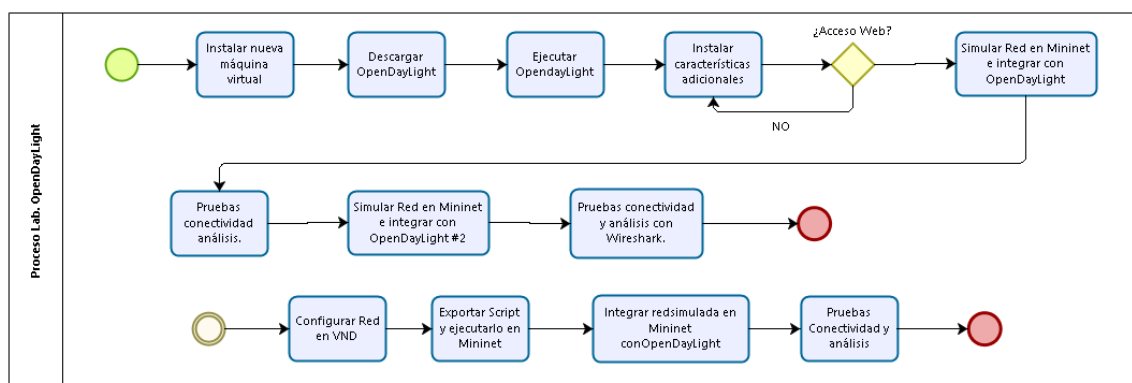


Diagrama 1 Proceso Laboratorio

Instalación OpenDayLight

OpenDaylight (ODL) es una plataforma abierta modular para personalizar y automatizar redes de cualquier tamaño y escala. El proyecto OpenDaylight surgió del movimiento SDN, con un claro enfoque programar una red de comunicaciones (OpenDayLight, 2017).

OpenDayLight es impulsado por una comunidad global colaborativa de proveedores y usuarios que se adapta continuamente para admitir el conjunto más amplio de casos de uso de SDN y NFV de la industria. Con más de 1000 desarrolladores, 50 organizaciones miembro y soporte para aproximadamente 1,000 millones de suscriptores en todo el mundo, OpenDaylight está desarrollando rápidamente cadenas de herramientas integradas para casos de uso líderes.

Es importante tener en cuenta, que esté controlador debe instalarse en una máquina virtual diferente, considerando los siguientes recursos:

- **CPU:** 1 Núcleo.
- **Memoria RAM:** 2 GB.
- **Almacenamiento:** 20 GB.
- **Red:** Modo NAT/Conexión Puente.

Para evitar problemas a la hora de ejecutar este controlador SDN, deberemos detener el servicio del ovs-testcontroller instalado en prácticas anteriores:

```
sudo systemctl stop openvswitch-testcontroller
```

Primero, debemos descargar el controlador OpenDayLight de la siguiente manera:

```
wget -nd -c
"https://nexus.opendaylight.org/content/groups/public/org/.opendaylight/integration/\
distribution-karaf/0.4.1-Beryllium-SR1/distribution-karaf-
0.4.1-Beryllium-SR1.tar.gz"
```

```
tar -xf distribution-karaf-0.4.1-Beryllium-SR1.tar.gz
cd distribution-karaf-0.4.1-Beryllium-SR1
```

Para ejecutar el controlador SDN OpenDayLight, ejecutamos el siguiente script:

```
sudo ./bin/karaf
```

Ahora bien, ya que la arquitectura de OpenDayLight se basa en los microservicios, debemos instalar una serie de características adicionales para lograr tener una operación optima del controlador:

```
feature:install -v odl-restconf odl-l2switch-switch odl-mdsal-
apidocs odl-dlux-core odl-dlux-all
```

- ? A partir de los componentes (todos) y sus relaciones, describa la arquitectura del controlador OpenDayLight.
- ? Describa la utilidad que tienen las características adicionales que estamos instalando en el controlador OpenDayLight.

Al finalizar la instalación, debemos tener el siguiente output:

```
opendaylight-user@root>feature:install -v odl-restconf odl-l2switch-all odl-mdsal-apidocs odl-dlux-core odl-dlux-all
Found installed feature odl-restconf 1.4.3-Boron-SR3
Installing feature odl-l2switch-all 0.4.3-Boron-SR3
Found installed feature odl-l2switch-switch 0.4.3-Boron-SR3
Found installed feature odl-mdsal-apidocs 1.4.3-Boron-SR3
Found installed feature odl-dlux-core 0.4.3-Boron-SR3
Found installed feature odl-dlux-all 0.4.3-Boron-SR3
opendaylight-user@root>
```

Imagen 1. Instalación Características

Acceso Web

Una de las características que instalamos, nos permite acceder via web al controlador OpenDayLight, para poder gestionarlo. Para acceder utilizamos la siguiente dirección:

http://Dir_IP_ODL:8181/index.html#/login

La siguiente es una captura de pantalla del acceso web a OpenDayLight:

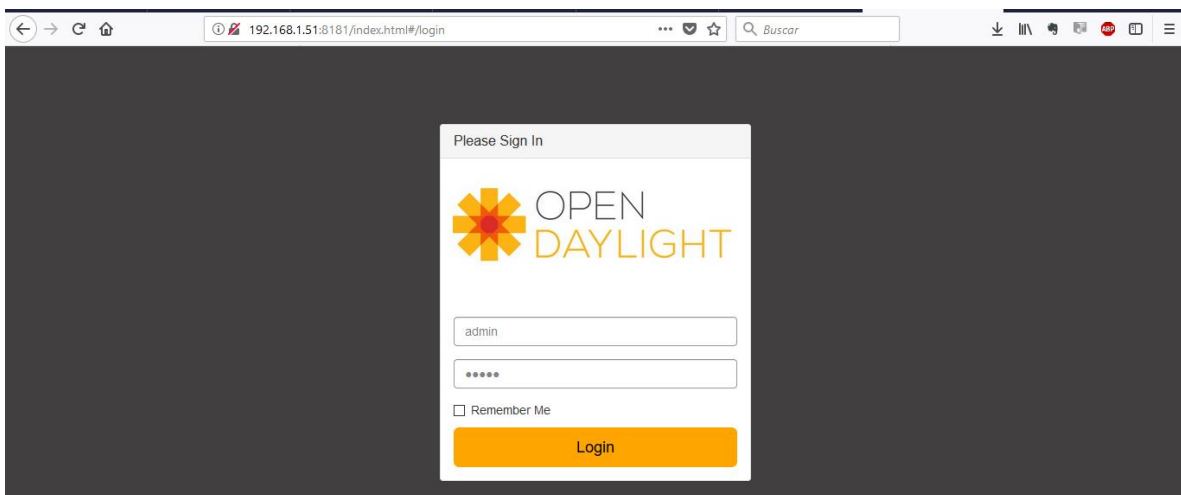


Imagen 2. Acceso Web

Las credenciales para el acceso son:

- Usuario: admin
- Contraseña: admin

- ? ¿Cuál es el componente que permite tener acceso web a OpenDayLight? ¿Cómo es su funcionamiento, considerando la arquitectura previamente descrita? ¿Por qué es necesario instalar esta característica adicional para lograr

Simulación con Mininet y OpenDayLight

Iniciamos una nueva instancia de OpenDayLight, y configuramos la siguiente topología de red en Mininet, esta topología debemos conectarla con el controlador OpenDayLight:

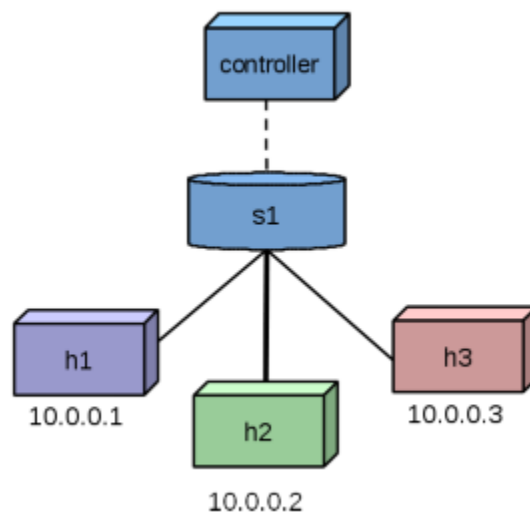


Diagrama 2 Topología de Red

📌 Recuerden utilizar los comandos aprendidos en la práctica de laboratorio de Mininet.

Una vez tengan simulada la red y conectada con OpenDayLight, realiza una prueba de conectividad con todos los nodos y revisa:

- ? ¿Qué cambios se pueden presenciar en el controlador OpenDayLight?

Ahora, terminamos la simulación de la red en Mininet y pasamos a emular la siguiente topología de red:

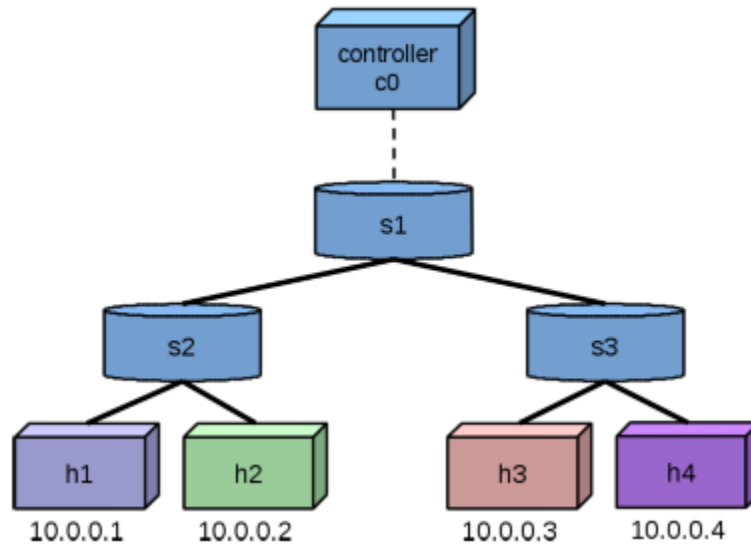


Diagrama 3. Topología de Red

- ❓ Analice: ¿Qué ocurre en el controlador OpenDayLight en el momento de terminar la primera simulación de red? Cuando simulamos la nueva topología, ¿Qué ocurre en el controlador OpenDayLight? Revise la topología y los nodos, en el menú lateral de la herramienta web.
- ❓ ¿Cuál considera usted, es la importancia de tener este comportamiento en un controlador SDN?

Ahora bien, iniciamos una captura de tráfico en Wireshark y pasamos a analizar el tráfico del protocolo OpenFlow en nuestra topología de red. Debemos obtener una captura como la siguiente:

203	5.343096773	10.0.2.15	192.168.206.130	OpenFlow	194	Type: OFPT_PACKET_IN	
204	5.343166619	10.0.2.15	192.168.206.130	OpenFlow	194	Type: OFPT_PACKET_IN	
205	5.343241692	192.168.206.130	10.0.2.15	TCP	60	6653 → 36710 [ACK] Seq=1087 Ack=15891 Win=65535 Len=0	
206	5.343249235	192.168.206.130	10.0.2.15	TCP	60	6653 → 36712 [ACK] Seq=1103 Ack=15907 Win=65535 Len=0	
207	5.348032324	10.0.2.15	192.168.206.130	OpenFlow	194	Type: OFPT_PACKET_IN	
208	5.348229245	192.168.206.130	10.0.2.15	TCP	60	6653 → 36710 [ACK] Seq=1087 Ack=16031 Win=65535 Len=0	
209	5.348316710	10.0.2.15	192.168.206.130	OpenFlow	194	Type: OFPT_PACKET_IN	
210	5.348931883	192.168.206.130	10.0.2.15	TCP	60	6653 → 36712 [ACK] Seq=1103 Ack=16047 Win=65535 Len=0	
211	5.499501799	192.168.206.130	10.0.2.15	OpenFlow	62	Type: OFPT_BARRIER_REQUEST	
212	5.499719429	10.0.2.15	192.168.206.130	OpenFlow	62	Type: OFPT_BARRIER_REPLY	
213	5.499851636	192.168.206.130	10.0.2.15	TCP	60	6653 → 36710 [ACK] Seq=1095 Ack=16039 Win=65535 Len=0	
214	6.487919389	192.168.206.130	10.0.2.15	OpenFlow	70	Type: OFPT_MULTIPART_REQUEST, OFPMP_TABLE	
215	6.488198138	10.0.2.15	192.168.206.130	OpenFlow	6166	Type: OFPT_MULTIPART_REPLY, OFPMP_TABLE	
216	6.488959289	192.168.206.130	10.0.2.15	TCP	60	6653 → 36710 [ACK] Seq=1111 Ack=18959 Win=65535 Len=0	
217	6.488971486	192.168.206.130	10.0.2.15	TCP	60	6653 → 36710 [ACK] Seq=1111 Ack=20419 Win=65535 Len=0	
218	6.488972395	192.168.206.130	10.0.2.15	TCP	60	6653 → 36710 [ACK] Seq=1111 Ack=21879 Win=65535 Len=0	
219	6.488973465	192.168.206.130	10.0.2.15	TCP	60	6653 → 36710 [ACK] Seq=1111 Ack=22151 Win=65535 Len=0	
220	6.494064124	192.168.206.130	10.0.2.15	OpenFlow	110	Type: OFPT_MULTIPART_REQUEST, OFPMP_FLOW	
221	6.495158608	10.0.2.15	192.168.206.130	OpenFlow	750	Type: OFPT_MULTIPART_REPLY, OFPMP_FLOW	
222	6.495494431	192.168.206.130	10.0.2.15	TCP	60	6653 → 36710 [ACK] Seq=1107 Ack=22847 Win=65535 Len=0	
223	6.517221354	192.168.206.130	10.0.2.15	OpenFlow	70	Type: OFPT_MULTIPART_REQUEST, OFPMP_GROUP_DESC	
224	6.517366539	10.0.2.15	192.168.206.130	OpenFlow	70	Type: OFPT_MULTIPART_REPLY, OFPMP_GROUP_DESC	
225	6.518582485	192.168.206.130	10.0.2.15	TCP	60	6653 → 36710 [ACK] Seq=1183 Ack=22863 Win=65535 Len=0	
226	6.519329947	192.168.206.130	10.0.2.15	OpenFlow	78	Type: OFPT_MULTIPART_REQUEST, OFPMP_GROUP	
227	6.519481666	10.0.2.15	192.168.206.130	OpenFlow	70	Type: OFPT_MULTIPART_REPLY, OFPMP_GROUP	
228	6.519651186	192.168.206.130	10.0.2.15	TCP	60	6653 → 36710 [ACK] Seq=1207 Ack=22879 Win=65535 Len=0	
229	6.520823288	192.168.206.130	10.0.2.15	OpenFlow	78	Type: OFPT_MULTIPART_REQUEST, OFPMP_PORT_STATS	
230	6.521062597	10.0.2.15	192.168.206.130	OpenFlow	518	Type: OFPT_MULTIPART_REPLY, OFPMP_PORT_STATS	
231	6.521186256	192.168.206.130	10.0.2.15	TCP	60	6653 → 36710 [ACK] Seq=1231 Ack=23343 Win=65535 Len=0	
232	6.523133240	192.168.206.130	10.0.2.15	OpenFlow	78	Type: OFPT_MULTIPART_REQUEST, OFPMP_QUEUE	
233	6.523284944	10.0.2.15	192.168.206.130	OpenFlow	70	Type: OFPT_MULTIPART_REPLY, OFPMP_QUEUE	
234	6.523401804	192.168.206.130	10.0.2.15	TCP	60	6653 → 36710 [ACK] Seq=1255 Ack=23359 Win=65535 Len=0	
235	6.988552654	192.168.206.130	10.0.2.15	OpenFlow	62	Type: OFPT_BARRIER_REQUEST	

Imagen 3. Captura Tráfico

- ❓ Describa el intercambio de paquetes que ocurre entre el controlador OpenDayLight y el resto de la red.

Bono

Integre una topología de red generada con el Virtual Network Descriptor, trabajado durante la clase, con el controlador OpenDayLight. Se debe presentar todo el proceso que llevo a cabo para lograr realizar el bono: diseño de la topología en el VND, ejecución del script de la topología en Mininet, integración con el controlador OpenDayLight y vista de los nodos y topología dentro del controlador SDN OpenDayLight.

NOTA: No se pueden utilizar las mismas topologías trabajadas en el desarrollo de este laboratorio.

Informe Laboratorio

Elabore un informe de laboratorio donde se evidencie lo siguiente:

- Trabajo realizado durante el laboratorio (instalaciones, configuraciones, diseños, comandos, entre otros). Elabore un diagrama de flujo, a través del cual, logré reflexionar respecto a los procedimientos realizados, tenga en cuenta mencionar las dificultades encontradas y como trabajo para resolverlas.
- Responda cada una de las preguntas aquí planteadas, teniendo en cuenta los procesos que siguió para dar respuesta a cada pregunta.
- Conclusiones del trabajo realizado. Tenga en cuenta que las conclusiones deben ser una síntesis del informe de laboratorio presentado, por lo que, en esta sección se deben incluir los resultados, un análisis de lo aprendido y el resumen del informe incluyendo los procesos ejecutados.

Referencias

Linux Foundation. (2014). OpenDayLight, 31(5), 1–58.

fontes, r. (24 de sep de 2014). *github.com/ramonfontes*. Obtenido de <https://github.com/ramonfontes/vnd-sdn-version>