

# Entendendo o Open vSwitch e o OpenWRT Network

Christiano Machado da Costa

<sup>1</sup>Ciência da Computação – Instituto Federal do Ceará – Campus Maracanaú  
Av. Contorno Norte, 1753 – 61936-000 – Maracanaú – CE – Brasil

christianomachado10@gmail.com

**Resumo.** Este artigo trata de entender melhor o funcionamento do Open vSwitch, bem como sua configuração básica. Também trata a configuração da ala de rede do OpenWRT, como a adição, subtração e alteração de suas interfaces. A forma de representação das interfaces físicas nas interfaces virtuais. Como a rede sem fio é interpretada pelo sistema e como configurar seu comutador interno. **Palavras-chave:** OpenWRT, Open vSwitch, Configurar Open vSwitch, Configurar OpenWRT, Interfaces OpenWRT, Rede OpenWRT.

**Abstract.** This article tries to understand the best operation of Open vSwitch, as well as its basic configuration. It also addresses an OpenWRT network configuration, such as adding, subtracting, and changing its interfaces. A way of representing physical interfaces in virtual interfaces. How a wireless network is interpreted by the system and how to configure its internal hub. **Keywords:** OpenWRT, Open vSwitch, Configure Open vSwitch, Configure OpenWRT, OpenWRT Interfaces, OpenWRT Network.

## 1. Introdução

Ao iniciarmos o processo de entendimento do Open vSwitch é necessário que alguns procedimentos sejam tomados. Usaremos como modelo o roteador TP-LINK TL 1043ND, é para ele que iremos compilar o *firmware* com o Open vSwitch. Na descrição estará o artigo feito por mim que mostra todo o processo de compilação do OpenWRT com o Open vSwitch e para que este artigo seja melhor entendido, é necessário que o leitor tenha um conhecimento no qual ele irá encontrar no documento anterior que pode ser encontrado aqui [Machado 2017].

Após a compilação ser feita e o *firmware* ser instalado no roteador, será necessário que entendamos melhor como funciona o Open vSwitch e para isto, um pequeno exemplo é demonstrado passo a passo na seção 2. Para que a experiência seja completa, é necessário entender também como funciona a parte de rede do OpenWRT, como ele trata suas VLANs, interfaces físicas e interfaces de rede sem fio.

Diferente de alguns *firmwares* proprietários, o OpenWRT é bem flexível com relação as suas configurações, pois ele é baseado em linux, o que nos permite realizar qualquer modificação necessária sem nenhum problema. Existem diversos documentos relacionados à configuração do OpenWRT com o Open vSwitch. Sendo este um pequeno resumo de toda sua complexidade [Burson 2015].

## 2. Open vSwitch

O Open vSwitch (OVS) é um comutador virtual que suporta o protocolo *OpenFlow* e é normalmente usado com hipervisores para interconectar máquinas virtuais dentro de

um *host* e máquinas virtuais entre diferentes *hosts* em toda rede. Também é usado em equipamentos dedicados à comutação. Ele pode ser uma peça chave em uma solução SDN.

Abaixo estão os comandos utilizados para instalação do Open vSwitch em distribuições do Debian. Existem pacotes para outras distribuições do Linux, porém não serão abordadas neste artigo. [Linux 2016]

#### Listing 1. Instalando Open vSwitch

```
$ apt-get update  
$ apt-get install openvswitch-common
```

### 2.1. Entendendo Open vSwitch

Agora iremos realizar um pequeno teste para compreender melhor o funcionamento do OVS. O estudo consistirá em colocar um comutador virtual como um intermediário entre a entrada e saída de uma Interface de Rede (NIC), por convenção *eth0* irá representar virtualmente a NIC, e a nossa pilha *IP*, em seguida conectaremos uma máquina virtual com o OVS. Para que todo o estudo seja feito é necessário que alguns pré-requisitos sejam instalados. Neste caso foi utilizado o **VirtualBox**, **OVS** e pacote **iproute2**. A Figura 1 mostra o estado inicial da máquina.

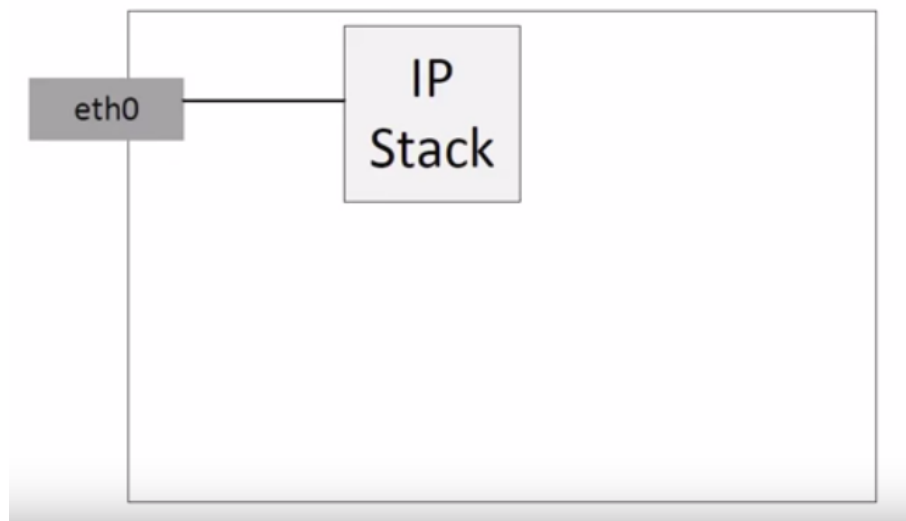


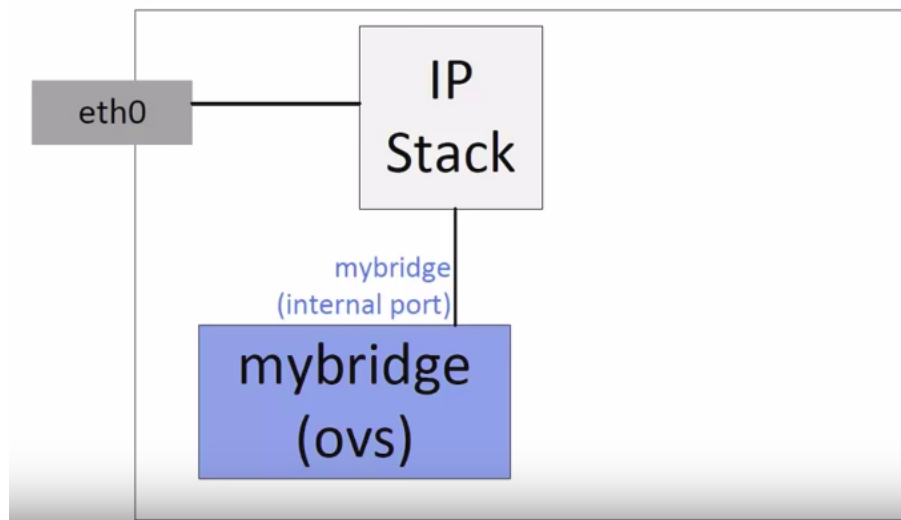
Figura 1. *eth0* e pilha IP [Malher 2013].

Agora iremos criar um *switch* virtual utilizando o OVS. O nosso *switch* se chamará **mybridge**, portanto, quando o segundo comando for executado deverão aparecer algumas informações relativas ao *switch* criado. Logo em seguida ele será ativado e já estará disponível como uma interface virtual. Agora o nosso diagrama está um pouco diferente, vejamos na Figura 2. Abra o terminal e digite os seguintes comandos:

#### Listing 2. Criando Switch Virtual

```
$ ovs-vsctl add-br mybridge  
$ ovs-vsctl show  
$ ifconfig mybridge up
```

O nosso *switch* se chamará **mybridge**, portanto, quando o segundo comando for executado deverão aparecer algumas informações relativas ao *switch* criado. Logo em seguida ele será ativado e já estará disponível como uma interface virtual. Agora o nosso diagrama está um pouco diferente, vejamos na Figura 2



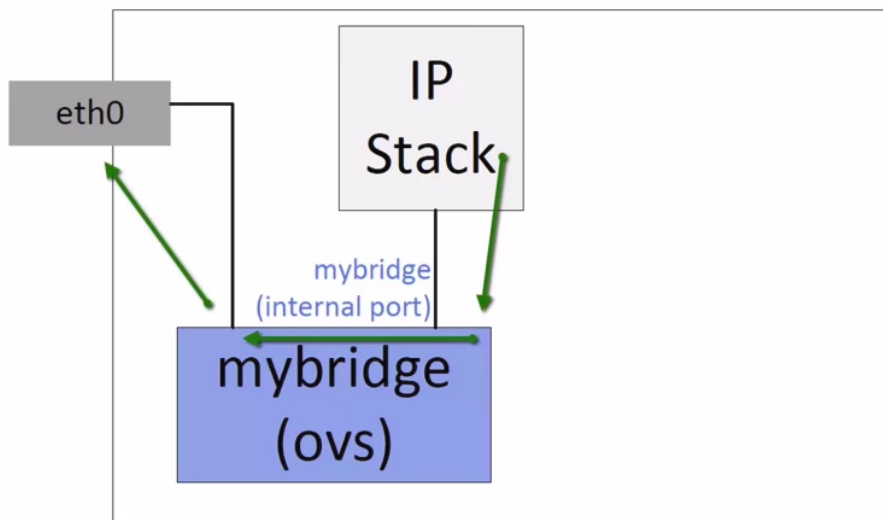
**Figura 2. Switch Virtual adicionado [Malher 2013].**

Pela Figura 2 podemos ver que a interface **mybridge** não está conectada à internet, ou seja, até o momento nada mudou no funcionamento da máquina. Agora nosso objetivo é conectar a interface recém criada à **eth0**. Os comandos abaixo são para adicionar a interface **eth0** ao OVS e mostrar o resultado, respectivamente.

### **Listing 3. Adicionando eth0 ao OVS**

```
$ ovs-vsctl add-port mybridge eth0  
$ ovs-vsctl show
```

Ao executar os comandos acima a conexão com a internet será perdida, pois agora a interface **eth0** está redirecionada para o *switch* virtual **mybridge**, e a pilha IP tentará acessar diretamente a **eth0**. Faremos com que a pilha IP tente acessar pelo OVS, a Figura 3 mostra o que deverá acontecer após realizar tais modificações.



**Figura 3. Máquina acessando primeiro o OVS [Malher 2013].**

Para consertar isso, faremos duas coisas, remover configuração IP do **eth0** e fazer a porta interna do meu *switch* virtual um cliente DHCP. Executaremos dois comandos no terminal que estarão disponíveis abaixo. Respectivamente, removerá a configuração IP da interface **eth0** e adicionará um IP padrão ao *switch* virtual *mybridge*.

**Listing 4. Removendo IP da eth0 e adicionando à mybridge**

```
$ ifconfig eth0 0
$ dhclient mybridge
```

Agora adicionaremos duas interfaces TUN/TAP para máquinas virtuais. Estas interfaces são dispositivos do *kernel* de rede virtual inteiramente em software e funcionam similar às camadas de Enlace e de Redes.

**Listing 5. Adicionando dispositivos TUN/TAP**

```
$ ip tuntap add mode vport1
$ ip tuntap add mode vport2
$ ifconfig vpor1 up
$ ifconfig vpor2 up
```

Após criados, devemos adicioná-los ao OVS para que o tráfego gerado pelas máquinas virtuais passe pelo nosso *switch*. A Figura 4 mostra como ficará o nosso diagrama.

**Listing 6. Adicionando dispositivos ao OVS**

```
$ ovs-vsctl add-port mybridge vport1
$ ovs-vsctl add-port mybridge vport2
```

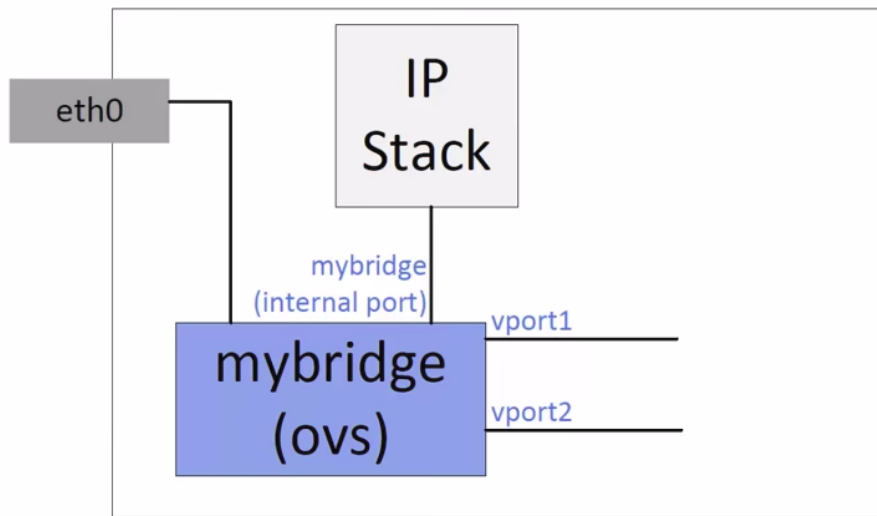


Figura 4. Portas TUN/TAP adicionadas [Malher 2013].

Para o próximo passo será necessário criar duas máquinas virtuais e modificar seus adaptadores de rede virtual para os que criamos aqui. Irei supor que as máquinas foram criadas, agora é só ir em **Configurações – Redes – Mudar o adaptador**, a Figura 5 ilustra os passos citados.

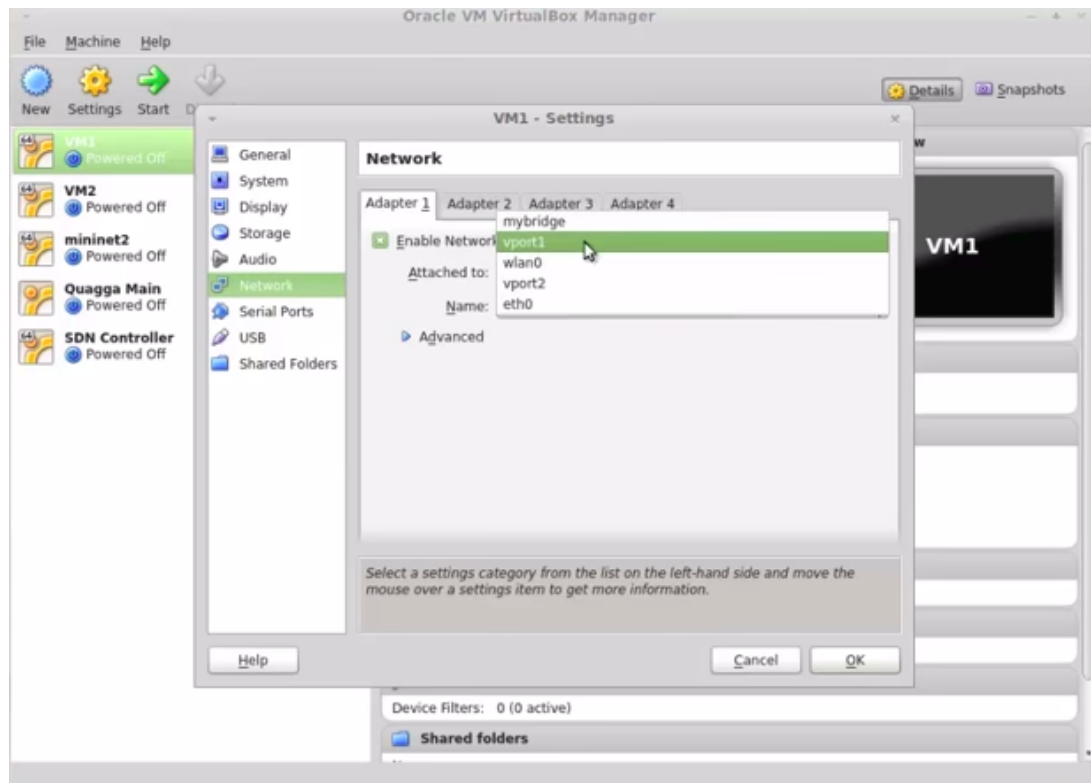


Figura 5. Configurando Máquinas Virtuais [Malher 2013].

Agora inicie as máquinas virtuais e verifique as configurações básicas de rede e

test o comando *ping* com um *host* externo.

## 2.2. Open vSwitch no OpenWRT

Como modelo de testes iremos utilizar o mesmo roteador no qual compilamos o OpenWRT com OVS [Machado 2017]. Caso o roteador não possua o OVS instalado, será necessário baixar o seu pacote. Para fazer isto é bem simples, basta acessar, como *root*, o roteador pelo terminal via **SSH** e em seguida digitar os comandos abaixo.

### Listing 7. Baixando Pacote OVS

```
$ dpkg update
$ dpkg install openvswitch
```

Feito isso, reinicie o equipamento para evitar qualquer imprevisto com o OVS. Como o objetivo é configurar o roteador para que ele dê suporte ao paradigma SDN, faremos uma configuração básica do OVS que é adicionar um controlador. Os comandos a seguir devem ser digitados dentro do roteador.

### Listing 8. Adicionando Controlador ao OVS

```
$ ovs-vsctl set-controller 'seu-ovs' tcp:'ip-controlador':'porta-contro'
```

## 3. Network no OpenWRT

O OpenWRT possui um *switch* virtual interno próprio que é responsável por controlar as interfaces físicas através de VLANs. Por padrão teremos duas VLANs principais, que são denominadas **eth0.1** e **eth0.2**, elas são associadas a interface WAN e as interfaces LANs do TP-LINK TL-1043ND. Iremos acessar o nosso roteador pela interface LuCI para facilitar o entendimento, porém, toda configuração feita no roteador é baseada em arquivo, ou seja, não é necessário que haja uma interface web, podemos simplesmente manipular os arquivos de tal forma que obteremos o mesmo resultado. Acesse o roteador através de um navegador com o seu IP, geralmente **192.168.0.1** ou **192.168.1.1**. Ao entrar nele vá em *Switch* na aba *Network*, podemos criar mais VLANs que podem representar alguma interface específica do roteador, faremos isto com finalidade explicativa.

A Figura 6 ilustra o resultado final. Lembrando que devemos ter muito cuidado ao relacionar as VLANs com as interfaces físicas, pois podemos perder o acesso ao roteador, mas isso pode ser resolvido apenas dando um *factory reset*. Então como exemplo criaremos uma VLAN para a interface física 4, para fazer isto basta clicar no botão **Add** e no campo ao lado indicar qual será o VLAN ID, depois na devemos marcar a *Port 4* como *untagged* e a CPU como *tagged*, o resto será marcado como *off*. Isso irá associar a interface 4 com a VLAN de ID descrito e esta VLAN estará associada ao *switch* virtual interno do equipamento.

#### VLANs on "switch0" (RTL8366RB)

VLAN ID	Port 0	Port 1	Port 2	Port 3	Port 4	CPU
Port status:						
1	off	untagged	untagged	untagged	off	tagged
2	untagged	off	off	off	off	tagged
3	off	off	off	off	untagged	tagged

Figura 6. Nova VLAN criada.

Também é possível criar interfaces virtuais, na aba *Network* em *Interfaces*. Essas interfaces virtuais nós iremos associar com as interfaces físicas que são representadas pelas VLANs. A Figura 7 mostra o resultado final deste exemplo. Ao clicar em **Add New Interface** criaremos uma interface, será necessário definir seu nome, no qual chamaremos de **vlan3**. Podemos selecionar alguns protocolos, como DHCP Client ou PPPoE, porém iremos selecionar o de endereço estático, mais abaixo selecione a VLAN **eth0.3**, depois clique em *submit*. Feito isto poderemos configurar um endereço IP, para as duas versões, também um *gateway* ou até mesmo um servidor DHCP. Nesse caso colocaremos apenas um endereço IP 10.10.3.1/24, ou seja com a máscara de rede 255.255.255.0 e ativaremos o servidor DHCP. Agora podemos nos conectar com o roteador utilizando a interface física 4 que estará associada à uma VLAN, que por sua vez também estará associada a uma interface virtual chamada VLAN3. As outras interfaces são criadas por padrão pelo sistema.

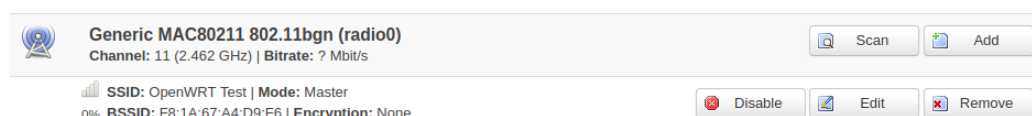
#### Interface Overview

Network	Status	Actions
<b>VLAN3</b> 	Uptime: 0h 0m 26s MAC-Address: 00:00:00:00:00:00 RX: 0.00 B (0 Pkts.) TX: 738.00 B (7 Pkts.) IPv4: 10.10.3.1/24	
<b>LAN</b> 	Uptime: 2h 27m 14s MAC-Address: F8:1A:67:A4:D9:F6 RX: 29.02 MB (288351 Pkts.) TX: 554.85 MB (440847 Pkts.) IPv4: 192.168.1.1/24 IPv6: FD72:17D1:661F::1/60	
<b>WAN</b> 	Uptime: 2h 26m 50s MAC-Address: 00:00:00:00:00:00 RX: 546.82 MB (434530 Pkts.) TX: 31.15 MB (272543 Pkts.) IPv4: 10.50.21.189/24	
<b>WAN6</b> 	Uptime: 0h 0m 0s MAC-Address: 00:00:00:00:00:00 RX: 546.82 MB (434530 Pkts.) TX: 31.15 MB (272543 Pkts.) IPv4: 10.50.21.189/24	

Figura 7. Nova interface VLAN 3 criada.

Assim como a rede cabeada, podemos representar a rede sem fio do mesmo jeito. Neste caso na aba *Network* em *Wifi* iremos configurar uma interface virtual referente a uma rede sem fio. Para fazer isto basta clicar em **Add** e em seguida, logo abaixo definir algumas configurações padrões de uma rede sem fio, como por exemplo o **SSID**, neste caso chamarei de OpenWRT Teste, então definiremos o modo, neste caso será como *Access Point* e por sua vez podemos associar ela a uma interface já criada, associaremos a interface VLAN3, na parte de *Wireless Security* podemos definir algumas senhas de acesso e formas de criptografia, porém, deixaremos esta rede livre. A Figura 8 mostra o resultado final deste exemplo.

### Wireless Overview



### Associated Stations

SSID	MAC-Address	IPv4-Address	Signal	Noise	RX Rate	TX Rate
No information available						

**Figura 8. Interface Wireless.**

Agora qualquer dispositivo que se conecte ao *Wifi* OpenWRT Teste estará na mesma subrede que o dispositivo conectado via cabo à interface física 4. Todas estas configurações podem ser feitas através de arquivos que estão dentro do roteador. Estas interfaces criadas podem ser adicionadas ao OVS.

## 4. Conclusão

Portanto, podemos montar uma rede de acordo com suas necessidades. Com a manuseabilidade das interfaces virtuais do sistema podemos conectar o OpenWRT ao OVS de diversas formas, dependendo apenas do objetivo final da nossa rede. Um claro exemplo, separar as interfaces físicas, ou seja, a interface WAN poderia receber dados vindos diretamente do controlador, a interface LAN 1 poderia ser a responsável pelo servidor DHCP externo.

Com esta flexibilidade nas configurações do roteador, podemos facilmente transformá-lo em um roteador *OpenFlow-enabled* com ajuda do OVS. Sendo o OVS responsável pelo controle de encaminhamento de todo o fluxo gerado, bem como a comunicação com o controlador. Tudo isso se aplica à roteadores que dão suporte ao OpenWRT e que possuem capacidade de armazenamento suficiente para o OVS, contudo, nem sempre a configuração poderá ser feita através de uma interface web e deverá ser feita manipulando os arquivos manualmente, ou até mesmo por um *script* que gere os arquivos de configuração.



## Referências

- Burson, L. (2015). Compilando e instalando openwrt com open vswitch. Turning TP-LINK WR1043NDv2.1 router into OpenFlow-enabled switch. Material disponível em : <http://blog.ljdelight.com/turning-tp-link-wr1043ndv2-1-router-into-openflow-enabled-switch/>.
- Linux, F. (2016). Distributions packaging open vswitch.
- Machado, C. (2017). Compilando e instalando openwrt com open vswitch. Compilando e Instalando OpenWRT com Open vSwitch. Material disponível em : [https://github.com/chrismachado/mininet/blob/master/Compilando\\_e\\_Instalando\\_OpenWRT.pdf](https://github.com/chrismachado/mininet/blob/master/Compilando_e_Instalando_OpenWRT.pdf).
- Malher, D. (2013). Introduction to open vswitch (ovs). Introdução ao Open vSwitch. Material disponível em : <https://www.youtube.com/watch?v=rYW7kQRyUvA&t=737s>.