**OPS漫谈之一：通过VLAN的配置流程来看OPS的系统架构**

陈玉强

OPS(OpenSwitch)是个开源交换机项目，其开源社区(www.openswitch.net)由惠普主导，目前得到了越来越多的关注和认可。本文旨在通过梳理配置VLAN的流程，帮助大家对OPS的系统软件架构有个更深入和具体的认识。

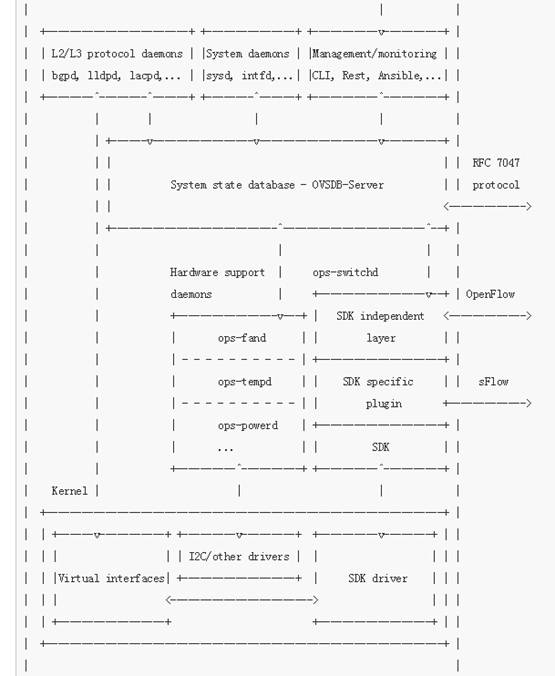
**OPS系统架构** 图1是OPS官方给出的OPS系统架构图，从图中可以看出，OPS系统架构最重要的特点是以OVSDB为核心，是新型的数据驱动操作系统。关于数据驱动的话题，我们在后续的文章再来谈。对OpenFlow熟悉的同学对OVSDB一定不陌生，OPS其实就是复用了OVS的OVSDB模块，并做了一定的改造。

图1-OPS系统架构图

**订阅-发布机制** 在给出配置VLAN的流程前，首先要讲下OPS的数据同步概念。简单的说，OPS的OVSDB内保存着系统所有的配置数据和运行数据，系统所有的其它模块要向OVSDB订阅表。比如，ops-vland这个进程是用来处理VLAN的，那么这个进程就要订阅VLAN这张表，而OVSDB内VLAN这张表一旦发生变化，OVSDB则把变化VLAN的数据发布给ops-vland，这就是“订阅-发布”机制。

**向DB配置VLAN** 完成VLAN的配置，涉及到下面几个进程。Vtysh，用于实现CLI；ops-vland，VLAN处理模块；ovsdb-server，OVSDB核心；ops-switchd，完成DB向芯片SDK的适配，实现芯片的配置。

图2描述了创建VLAN的配置流程，注意在这张图里还没有真正的配置芯片：

第一步：用户在CLI 的配置模式下输入创建VLAN命令行，vtysh则本地创建VLAN表，然后把该VLAN表同步给ovsdb-server。这里有人会有疑问，到底什么是VLAN表。其实VLAN表就是一个C语言的结构体，具体到VLAN这张表，或者说具体到VLAN这个结构体来说，它定义了id，name，description等元素(有兴趣的同学要研究下OVSDB关于表和列的源码才会这里有更正确的认识)，这我们容易理解，它还定义了hw\_vlan\_config元素，这个元素是用于控制是否配置芯片的。注意此时表VLAN的hw\_vlan\_config是没有设置的。

第二步：ovsdb-server同步好VLAN表后，向所有订阅该表的进程发布该数据。这个例子里就是把VLAN数据发布给ops-vland和ops-switchd两个进程。ops-switchd收到该VLAN数据后，检查’hw\_vlan\_config’没有置位，所以它不会向芯片添加该VLAN。



图2-向DB配置VLAN

**向芯片配置VLAN** 图3描述了完成配置芯片的流程。

第三步：当用户使能了这个VLAN，并且把某个端口加入这个VLAN后，其数据同步到ops-vland，ops-vland检查出此时应该向芯片添加该VLAN，就对该VLAN表置位’hw\_vlan\_config’。该置位数据同样会发布给OSVDB。

第四步：OVSDB把该VLAN表的’hw\_vlan\_config’置位数据同步给ops-switchd，ops-switchd检查’hw\_vlan\_config’置位后，通过plugin模块实现向芯片的配置。至此，芯片里真正完成了该VLAN的配置。



图3-向芯片配置VLAN

**多说一句** 图2和图3里都有一个’XXX Plugin’模块，这里有必要说下。Plugin模块是OPS系统实现软硬解耦的关键，不同厂家的芯片使用不同的Plugin代码。目前，Broadcom芯片使用的是opennsl plugin，Cavium芯片使用他们自己的xpliant plugin，Mellanox和盛科共同使用sai plugin。不同的plugin调用不同的芯片API接口，opennsl plugin调用OPENNSL接口，xpliant plugin调用cavium SDK接口，sai plugin调用SAI接口。

**结束语** OPS是一个巨大的开源项目，作为白牌交换机领域最具竞争力的开源系统软件，它有很多独到的特点和优势。本文旨在通过一个配置VLAN的例子，向同学们介绍OPS的系统架构。文中提到的一些概念，没有深入展开，这些概念就留给对数通感兴趣的同学们去研究了。