# 1.基础知识介绍

1. **Netconf可以理解为一种通用的协议（给网络设备发送配置）。**

这样我的配置就可以写成脚本，关注点变成了配置数据而不是设备的相关命令。

理性情况，所有的设备都用相同的配置，netconf会自己处理他们到相应的机器上，但是网络设备的配置结构往往不同，比如，思科设备配置一个address只要知道接口名，ip版本和地址掩码就够了，而juniper的机器上不仅需要知道这些，还要额外提供一个unit号来标识逻辑接口。**这样导致实现同样功能的不同设备需要配置结构也往往不同。**

**Yang模型是netconf协议的一种数据建模语言。**

1. Yang model即是给对应设备所需的配置结构来个模型，接着就是如同填写完型填空一样。
2. Yang model模式的节点主要分为四种类型：set/get/rpc/notification(配置节点，状态节点，操作节点，通知节点)

交换机和控制机进行交互，首先控制器会下发基于yang模型生成的报文，naas收到以后进行处理，感知到有变化的节点时，会调用我们注册的对应节点的回调函数进行处理，从而实现功能。（如果是通知节点，是交换机组装通知报文，通过naas提供的接口发送到控制器，再由控制器解析）

NaaS配置管理系统：

1. 配置管理子系统模块层次图：
2. 建链过程概述：

Step1：callhome进程用于建立controller和云盒直接的TCP链接，并告诉controller：“嘿，我在这儿，快来向我注册吧”

Step2：TCP链接建立后，会在云盒上起SSHD进程，由controller主动向云盒发起SSH连接申请。（在服务器上开启了sshd服务，其他pc上才能ssh进来，ssh为一个类似telnet的服务）

Step3：SSH通道建立后，就可以在上面跑起netconf协议，下发一些xml格式的报文（yang模型），进行业务交互。

业务plugin概述：业务plugin作为配置管理子系统和业务应用之间的桥梁，负责完成两边接口和数据是适配转换工作，包括：

* 从配置管理子系统订阅其关心的配置节点的数据变化，并将这些改变的内容转化为业务应用的配置；
* 提供配置管理子系统查询相关状态节点的接口，让配置管理子系统能够获得业务应用的运行状态数据；
* 从配置管理子系统订阅其关心的rpc操作，并响应这些rpc操作；
* 当业务应用检测到相关事件告警发生，通过配置管理子系统产生响应的notification；

# 2.Yang 模型开发

## 2.1 NETCONF

Netconf协议规定了client与server两种角色。Client可以与server建立网络连接，建立连接后，client可以基于NETCONF协议完成对server的管理。例如，下发配置，完成特定操作等。对于我们的应用场景，交换机为server角色，并遵循标准的NETCONF协议，因此用户可以使用遵循标准的NETCONF协议的client完成对交换机的管理。

## 2.2交换机上NETCONF的实现

交换机本质上是一台Linux主机，上面运行的进程完成了交换机的业务逻辑，交换机上的主要进程有：

* VRP进程：交换机各种转发业务，对外提供命令行界面
* RPA进程(remote plugin agent)：python实现的NETCONF协议南向进程
* Confd进程：NETCONF协议解析者，NETCONF协议引擎。（是一个统一配置管理工具）

Confd与RPA之间,RPA与VRP之间通过进程间通信建立连接，用户可以通过telnet/ssh等方式登录到VPR命令行界面。因此，NETCONF整体的概念图为：

## 一条config的下发

sysname

VRP对通过telnet对外提供命令行界面时，命令行提示符中的内容是VRP的sysname，如下图所示：

方框中的HUAWEI为VRP的sysname。

在中大型网络中，交换机作为接入设备，可能被放置在不同的物理位置，例如一座10层的大厦，每一层都有一台接入交换机，如何方便区分这些交换机？将交换机的sysname命名为不同的楼层名可能为一个好办法，例如5层的交换机可以叫HUAWEI-F5：

**通过NETCONF修改sysname**

上面一节介绍了如何通过VRP提供的命令行修改sysname，如果通过NETCONF修改sysname，是如何做的呢？

1. 找到sysname的YANG模型定义

定义设备sysname的YANG模型的xpath为：huawei-device:device/name，关于xpath的写法，参考：[http://rnd-isourceb.huawei.com/s00201741/Switch\_Product\_YANG\_development\_manual/tree/master/%E9%99%84%E5%BD%95/YANG%E6%A8%A1%E5%9E%8B%E8%AE%BE%E8%AE%A1%E8%BE%93%E5%87%BA%E8%A7%84%E8%8C%83.md#%E8%8A%82%E7%82%B9%E7%9A%84xpath](http://rnd-isourceb.huawei.com/s00201741/Switch_Product_YANG_development_manual/tree/master/%E9%99%84%E5%BD%95/YANG%E6%A8%A1%E5%9E%8B%E8%AE%BE%E8%AE%A1%E8%BE%93%E5%87%BA%E8%A7%84%E8%8C%83.md" \l "%E8%8A%82%E7%82%B9%E7%9A%84xpath)

打开huawei-device的YANG模型文件，找到name节点，可以看到name节点的描述如下：

leaf name {

type string;

description "The name of device, for example SZ-LG-001.";

}

可以知道，YANG模型是一个字符串类型。

1. 使用yangsh修改这个YANG模型。

Yangsh是华为自研的一个NETCONF的调试clinet，使用方法见：[http://rnd-isourceb.huawei.com/s00201741/Switch\_Product\_YANG\_development\_manual/tree/master/02%E4%BA%A4%E6%8D%A2%E6%9C%BA%E5%9F%BA%E6%9C%AC%E6%93%8D%E4%BD%9C/%E4%BA%A4%E6%8D%A2%E6%9C%BA%E5%9F%BA%E6%9C%AC%E6%93%8D%E4%BD%9C.md#yangsh%E5%9F%BA%E6%9C%AC%E6%93%8D%E4%BD%9C](http://rnd-isourceb.huawei.com/s00201741/Switch_Product_YANG_development_manual/tree/master/02%E4%BA%A4%E6%8D%A2%E6%9C%BA%E5%9F%BA%E6%9C%AC%E6%93%8D%E4%BD%9C/%E4%BA%A4%E6%8D%A2%E6%9C%BA%E5%9F%BA%E6%9C%AC%E6%93%8D%E4%BD%9C.md" \l "yangsh%E5%9F%BA%E6%9C%AC%E6%93%8D%E4%BD%9C)。使用yangsh改变name的值如下图所示：

1. Sysname是如何下发的？

需要注意的是，由于VRP是交换机的业务主体，因此修改sysname的操作实际是从VRP进程中生效。让我们回想NETCONF的整体概念图，可知sysname的下发分如下几步：

1. yangsh(NETCONF client)中，响应commit命令，发现sysname发生变化。yangsh将变化组织为符合NETCONF协议的XML报文，发送至confd进程
2. confd进程解析收到的NETCONF协议报文，并与数据库中的配置比对，发现sysname发生变更。
3. confd检索关注/huawei-device:device/name节点变更的南向plugin，发现RPA关注此变更。通知RPA进程节点变更事件。
4. RPA收到此节点变更事件后，与confd进行多次交互，得知sysname变化为HUAWEI-F5。
5. RPA进程中调用具体的plugin实现，并将sysname变更的xml作为参数传递给plugin。
6. plugin解析此变化，并把sysname组织成命令行sysname HUAWEI-F5配置到VRP进程，同时将结果返回。
7. RPA进程收到plugin的返回结果，再将结果通知给confd进程。
8. confd进程根据RPA的结果刷新数据库，并将结果组织成标准的NETCONF协议发回yangsh
9. yansh显示出成果的消息。

## 2.3 RPA进程的实现

RPA进程北向链接confd，南向连接VRP，是YANG模型到VRP配置的中转站。

RPA的主要功能为：接受confd的YANG模型变更消息，将其转换成VRP命令下发，并将结果返回给confd。同时处理过程中的各种异常场景。因此要正确实现RPA，至少需要两部分知识：1.与confd的通讯方法 2:VRP的命令配置方法。