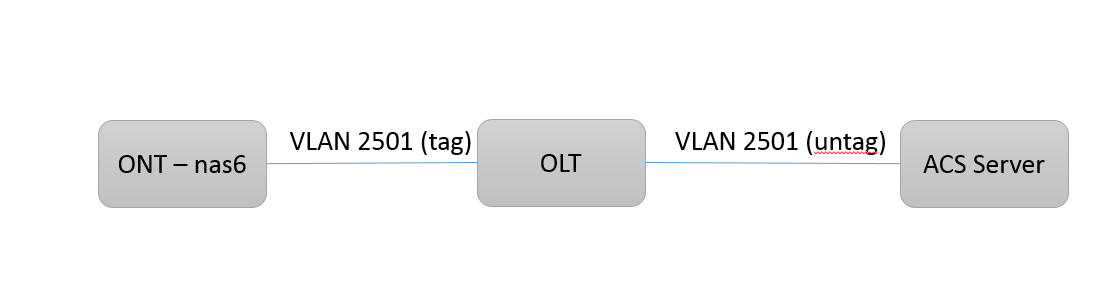
# Setup topo for TR069

## Structure of vACS



## Configure ACS

## Check Ip of server ACS:

$ifconfig

eth0 Link encap:Ethernet HWaddr 74:86:7a:dd:ab:6e

inet addr:11.0.0.33 Bcast:192.168.2.255 Mask:255.255.255.0

UP BROADCAST MULTICAST MTU:1500 Metric:1

RX packets:0 errors:0 dropped:0 overruns:0 frame:0

TX packets:0 errors:0 dropped:0 overruns:0 carrier:0

collisions:0 txqueuelen:1000

RX bytes:0 (0.0 B) TX bytes:0 (0.0 B)

Interrupt:16

## Install ACS (Refer <https://github.com/genieacs/genieacs/wiki/GenieACS-v1.1.x-installation-in-Ubuntu-14.04-Server> )

### Install Node.js

Version v1.1.x of GenieACS is targeted at NodeJS v7.x:

curl -sL https://deb.nodesource.com/setup\_7.x | sudo -E bash -

sudo apt-get install -y nodejs build-essential

### Install Redis

The best-working Redis server is the one that is in the Ubuntu repositories:

apt -y install redis-server

### Install MongoDB

apt-key adv --keyserver hkp://keyserver.ubuntu.com:80 --recv 0C49F3730359A14518585931BC711F9BA15703C6

echo "deb http://repo.mongodb.org/apt/debian jessie/mongodb-org/3.4 main" | sudo tee /etc/apt/sources.list.d/mongodb-org-3.4.list

apt update

apt install mongodb-org -y

mkdir -p /data/db

### Install GenieACS

curl -sS https://dl.yarnpkg.com/debian/pubkey.gpg | sudo apt-key add -

echo "deb https://dl.yarnpkg.com/debian/ stable main" | sudo tee /etc/apt/sources.list.d/yarn.list

sudo apt update

sudo apt install yarn -y

yarn global add genieacs --prefix /opt

cd /opt

ln -s /root/.config/yarn/global/node\_modules/genieacs/ .

### Install Ruby, Bundler, Rails for the GenieACS GUI install

apt install -y git-core curl zlib1g-dev build-essential libssl-dev libreadline-dev libyaml-dev libsqlite3-dev sqlite3 libxml2-dev libxslt1-dev libcurl4-openssl-dev python-software-properties libffi-dev

cd

git clone git://github.com/sstephenson/rbenv.git .rbenv

echo 'export PATH="$HOME/.rbenv/bin:$PATH"' >> ~/.bash\_profile

echo 'eval "$(rbenv init -)"' >> ~/.bash\_profile

git clone git://github.com/sstephenson/ruby-build.git ~/.rbenv/plugins/ruby-build

echo 'export PATH="$HOME/.rbenv/plugins/ruby-build/bin:$PATH"' >> ~/.bash\_profile

source ~/.bash\_profile

rbenv install -v 2.3.4

rbenv global 2.3.4

gem install bundler

gem install rails

rbenv rehash

### Install GenieACS GUI

cd /opt/

git clone -b master https://github.com/zaidka/genieacs-gui

cd genieacs-gui

cp config/graphs-sample.json.erb config/graphs.json.erb

cp config/index\_parameters-sample.yml config/index\_parameters.yml

cp config/summary\_parameters-sample.yml config/summary\_parameters.yml

cp config/parameters\_edit-sample.yml config/parameters\_edit.yml

cp config/parameter\_renderers-sample.yml config/parameter\_renderers.yml

cp config/roles-sample.yml config/roles.yml

cp config/users-sample.yml config/users.yml

bundle

rails db:migrate RAILS\_ENV=development

### Launch GenieACS automatically at boot up

To have it running in the background at boot up, put the following in the Ubuntu crontab (Access to it via crontab -e). If you cannot find the redis-server binary in /usr/bin, it will be in /home/bin/, so change it if you need it:

@reboot /usr/bin/mongod > /var/log/mongod.log 2>&1

@reboot /usr/bin/redis-server > /var/log/redis-server.log 2>&1

@reboot /opt/bin/genieacs-cwmp > /var/log/genieacs-cwmp.log 2>&1

@reboot /opt/bin/genieacs-nbi > /var/log/genieacs-nbi.log 2>&1

@reboot /opt/bin/genieacs-fs > /var/log/genieacs-fs.log 2>&1

@reboot sh /startup.sh > /var/log/rails.log 2>&1

In /startup.sh put the following commands:

cd /opt/genieacs-gui && /opt/genieacs-gui/bin/rails s -b 0.0.0.0 -p 3000

In /opt/genieacs-gui/bin/rails change the first line. It then will seem equal as:

#!/root/.rbenv/versions/2.3.4/bin/ruby

begin

load File.expand\_path('../spring', \_\_FILE\_\_)

rescue LoadError => e

raise unless e.message.include?('spring')

end

APP\_PATH = File.expand\_path('../config/application', \_\_dir\_\_)

require\_relative '../config/boot'

require 'rails/commands'

If you don't change this line, rails app won't work

### Authentication configuration

vim /usr/local/share/.config/yarn/global/node\_modules/genieacs/config/config.json

|  |
| --- |
| {  "MONGODB\_CONNECTION\_URL" : "mongodb://127.0.0.1/genieacs",  "CWMP\_INTERFACE" : "0.0.0.0",  "CWMP\_PORT" : 7547,  "CWMP\_SSL" : false,  "NBI\_INTERFACE" : "0.0.0.0",  "NBI\_PORT" : 7557,  "FS\_INTERFACE" : "0.0.0.0",  "FS\_PORT" : 7567,  **"FS\_HOSTNAME" : "file\_server\_ip",**  "DEBUG" : false  } |

Noted: Edit **"file\_server\_ip" to "11.0.0.33",**

vim /usr/local/share/.config/yarn/global/node\_modules/genieacs/config/auth.js

|  |
| --- |
| "use strict";  function connectionRequest(deviceId, url, username, password, callback) {  **return callback('cpe\_username','cpe\_password');**  }  exports.connectionRequest = connectionRequest; |

## Configuration for OLT

## OLT Dasan

vlan create 2501 // Tạo vlan 2501

!

vlan add br2501 9 untagged // Gói tin vlan 2501 đi qua port 9 sẽ được untagged

!

interface br2501

no shutdown

ip address 11.0.0.1/24 // Tạo IP cho interface vlan 2501

!

!

extended-vlan-tagging-operation 2501\_FTP create

downstream-mode enable

untagged-frame 1

treat inner vid 2501 cos 4 tpid 0x8100

apply

!

extended-vlan-tagging-operation HSI\_35 create

downstream-mode enable

single-tagged-frame 1

filter inner vid 35 cos 0 tpid 0x8100

treat remove single

treat inner vid 35 cos copy-inner tpid copy-inner

apply

!

!

traffic-profile DS\_TH create

tcont 1

gemport 1/1

dba-profile BE

tcont 2

gemport 2/1

dba-profile FTP

tcont 3

gemport 3/1

dba-profile BE

mapper 1

gemport count 1

mapper 2

gemport count 1

mapper 3

gemport count 1

bridge 1

ani mapper 1

uni virtual-eth 1

extended-vlan-tagging-operation HSI\_35

bridge 2

ani mapper 2

link ip-host-config 1

ip-host-config 1

ip address static

extended-vlan-tagging-operation 2501\_FTP

apply

!

!

gpon-olt 1

onu add 8 DSNW26acf4f8 auto-learning

onu encryption 8 enable

onu-profile 8 DS\_TH

onu static-ip 8 ip-host 1 11.0.0.88/24 gw 11.0.0.1

!

## Configuration for ONT

Step 1: Telnet to ONT

Step 2: Setup ULR for ACS server

tcapi set Cwmp\_Entry acsUrl http://11.0.0.33:7547

tcapi commit Cwmp\_Entry

tcapi save Cwmp\_Entry

## Check result

Open the Web to obverse info on ACS Server

<http://11.0.0.33>

## Troubleshoot

Using Wireshark to capture packet at ACS server or ONT side.

# How to test TR069

Refer testcases at \\10.61.63.11\cdbr-projects\ONT\6.Archive\PARTNER\DASAN\5.TEST CASE\1.TOTAL TEST CASE\18. TR069 test cases.pdf