5/10/2021

Nadir Hussain

023-18-0025

Computer Networks

Lab 08

**Lab Objectives**

* Configure FTP Server and explain its working.
* Configure Link aggregation at layer 2
  + Create trunk with 3 physical interfaces, create multiple trunks and make physical links part of different trunks
* Configure Link aggregation at layer 3
  + Create trunk with multiple physical interfaces, create multiple trunks and make physical links part of different trunks

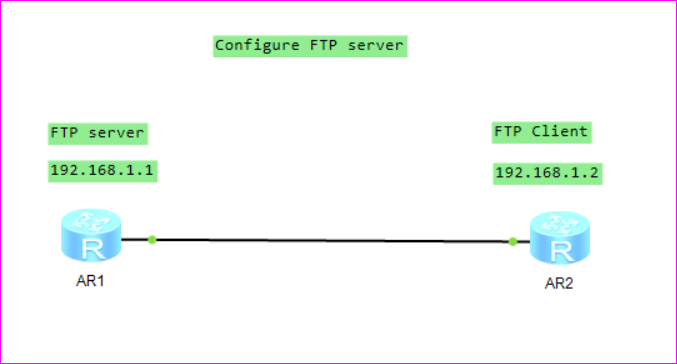
**Configure FTP Server and explain its working.**

***Why do we need FTP server and client?***

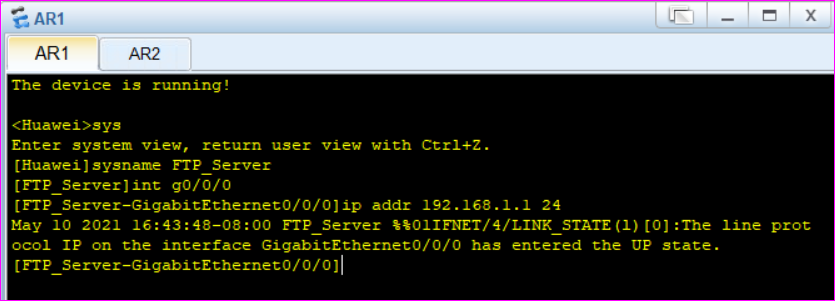
* We know router is layer 3 device, its main functions are routing and forwarding.

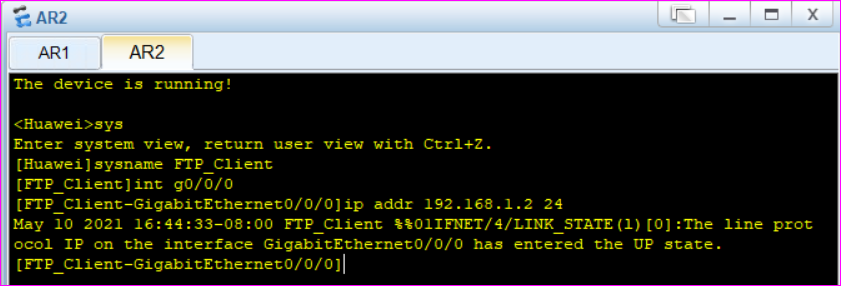
But this device can provide some application layer services for sake of control plane. Some of application layer protocols help Network layer to complete its functionality. FTP is one of them in which one node sends file and other receives the file in FTP. So we implement FTP here.

**Design the network**



**Configure interfaces**





To accomplish FTP server functionality with routers, we need to enable FTP server, but before that we want to create an account so client have user name and password to access the resources on server.

There is directory service to create account, called **aaa.**

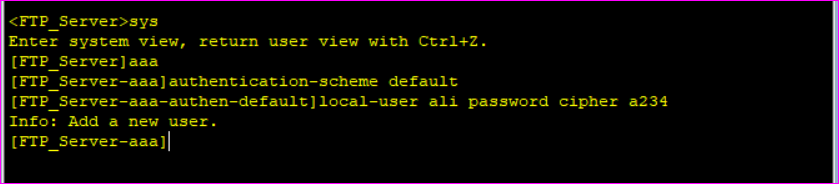
**aaa** is directory service that allows to create account with authentication, authorization and accounting.

Aaa=authentication, authorization and accounting,

Client will request the server with user id and password when server receives the credentials it will authenticate and authorize the client moreover, server keeps the record when user was logged in and logged out(accounting) etc.

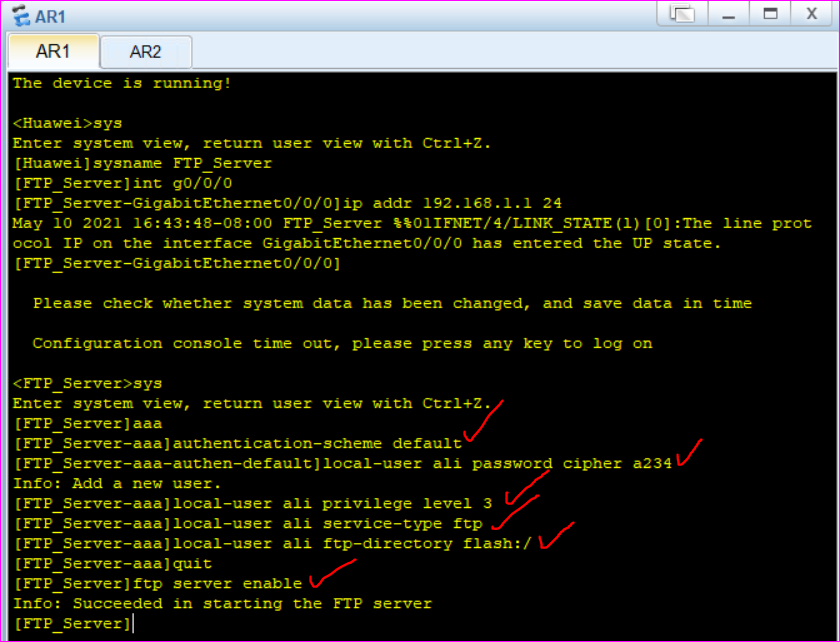
**Now creating aaa client account**

**Enter into aaa service, define authentication scheme and create a local user**



User is added to **aaa** directory

Now define the privileges of user and enable ftp server



We have assigned the user level previlidges to created account

Defined the ftp service of aaa for user or we can define as http as well

And got access to flash:/ directory, its by default directory on huwei routers

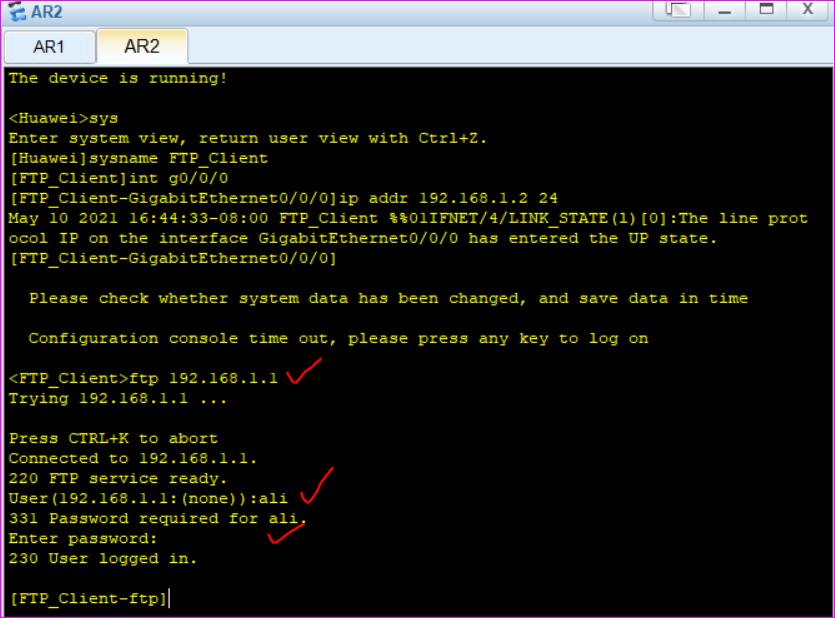
For admin we use http, for other users we enable ftp generally.

Then enabled the ftp server, which is now listening for the ftp requests from client

**Note :**

The number of possible connection with a local user account is can be limited by applying the command access-limit no\_of\_active\_connection

**Now on FTP client, we request access from server**



We can display the flash directory contents by dir command.

Now we can execute Operating System commands here, like copy directory from one location to another, change file permissions etc.

**Link aggregation**

Why link aggregation?

1. Balance load : weight is distributed on both links, we can change weight as well.
2. Backup : In case if one link on router/switch is failed, then other takes the load

Now we want to aggregate, or speedup the bandwidth of routers then we aggregate the links between routers. Exp: We increase the capacity by adding or doubling the links.

But it will not benefit because these are physically 2 links but logically it should be one.

We need mapping of physical links to logical links.

Logical link is named as trunk in networking terms so we create trunks.

Link Aggregation supports two modes of implementation

**Manual load balancing mode:**

* Member interfaces are manually added to a LAG.
* All the interfaces configured with load balancing are set in forwarding state.
* The manual load balancing mode does not use the Link Aggregation Control Protocol (LACP).

**Static LACP mode:**

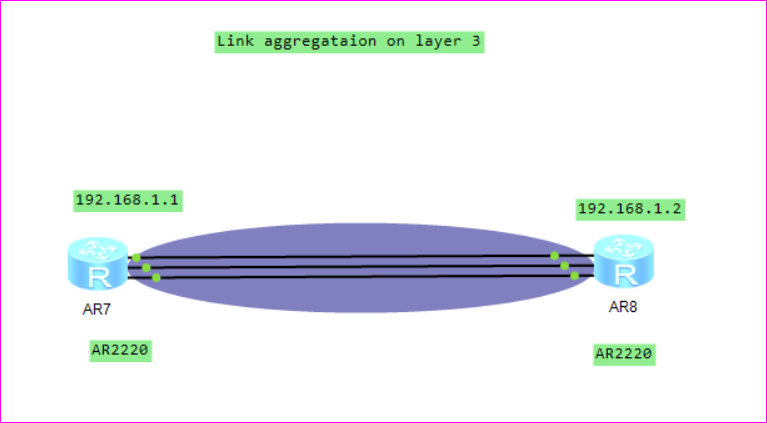
* The devices at the two ends of a link negotiate aggregation parameters by exchanging LACP packets.
* Then two devices determine the active interface and the inactive interface.
* It is necessary to manually create an Eth-Trunk and add members to it.

If an active link fails, data forwarding is switched to the backup link with the highest priority and the status of backup link charges to active

**Task**

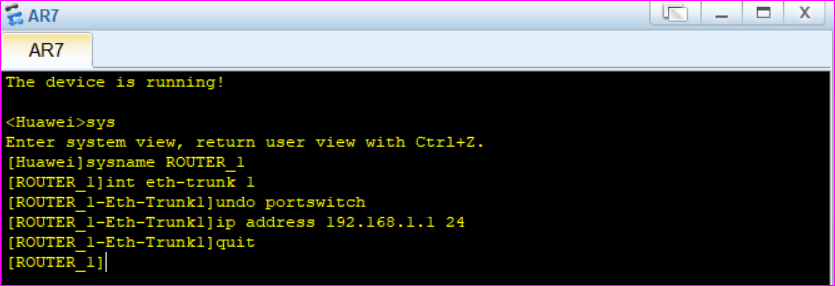
**Create trunk with 3 physical interfaces on layer 3**

**Design**



We don’t need to configure interfaces, rather we create trunk(s) on each router and interfaces into it. Trunk is logical link

**On router 1**

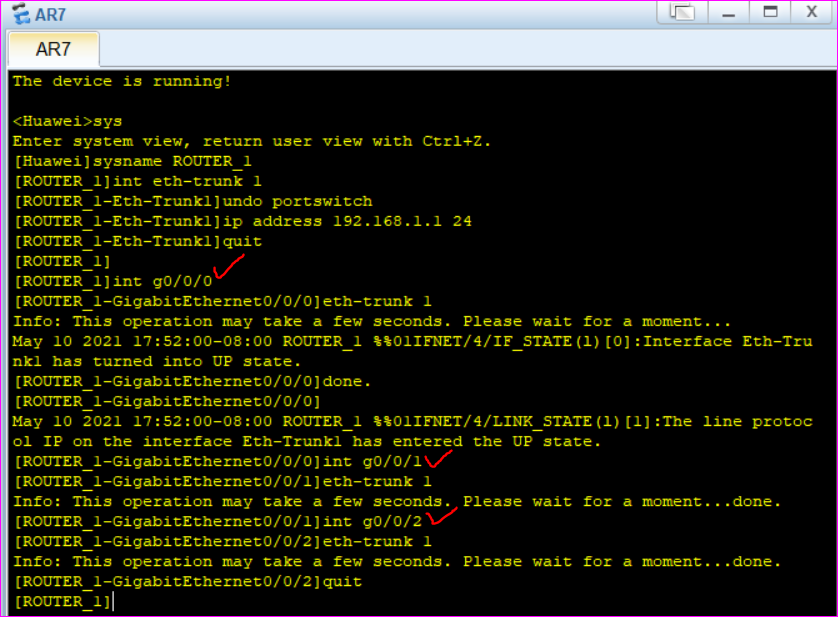


Why undo port switch?

When we are using port of router as switch, so we need to turn it off so as it should work as router port. Then it will accept the ip.

We have enabled trunk by eth-trunk on router 1.

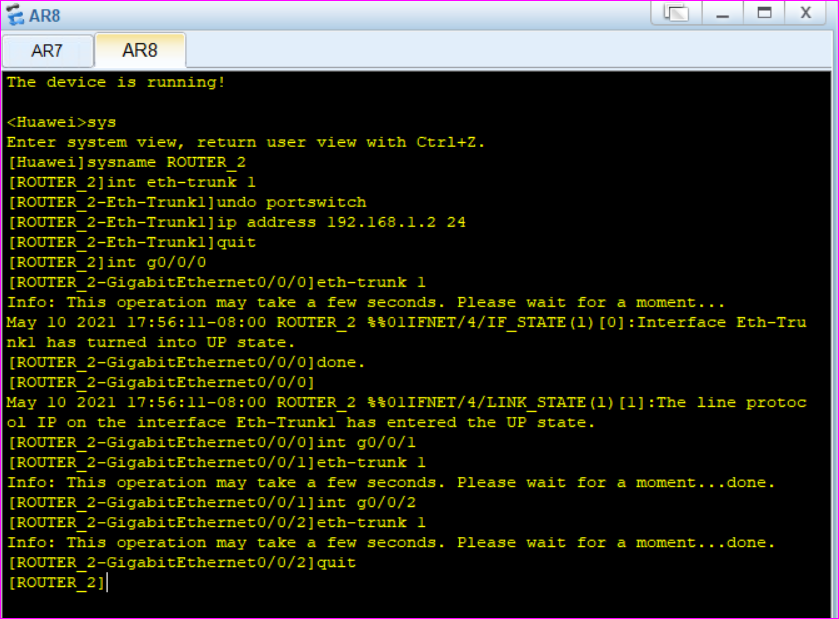
Now make interface1, interface2 and interface3 Of AR7 as parts of trunk that we created.



If we want interface to work as router port interface, the we need to undo its port switch, which is default.

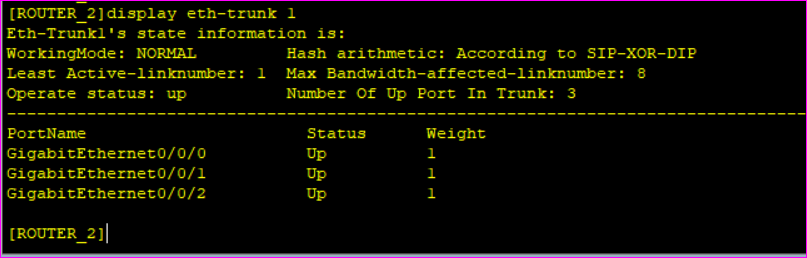
All three interfaces are part of trunk 1

**On other router we do the same**



All interfaces on router 2 are part of eth-trunk 1

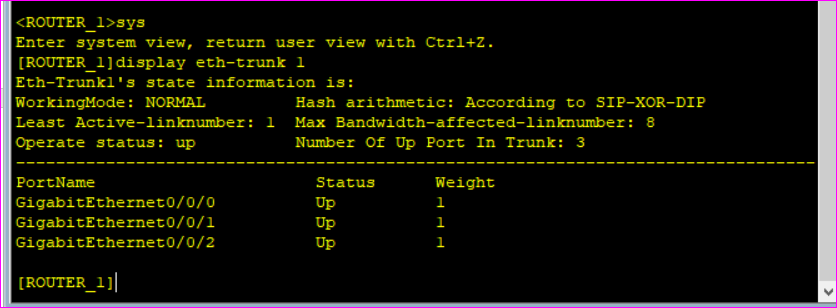
**Now we display link aggregation on both routers**



All three interfaces are UP and working fine on router 2.

We can change the weight, by which we mean we shift the load to other interface.

**Now we look aggregation on router 2**

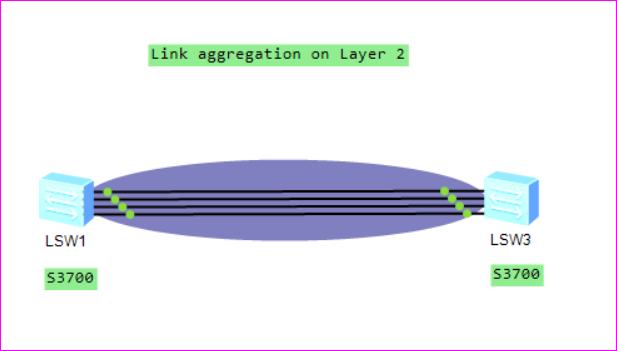


All three ports are UP.

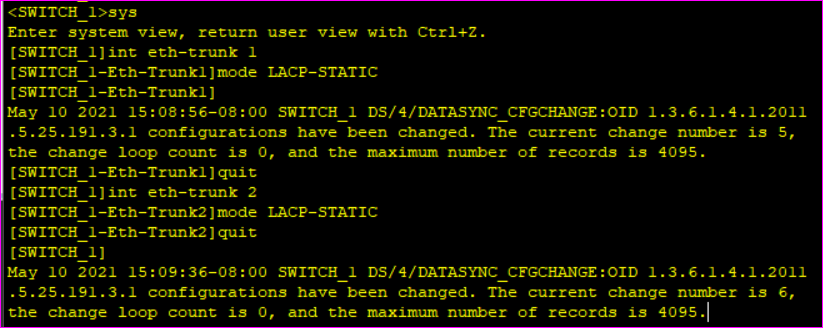
**Link aggregation on Layer 2**

**Create trunk with 4 physical interfaces, create multiple trunks and make physical links part of different trunks on layer 2**

**Design**



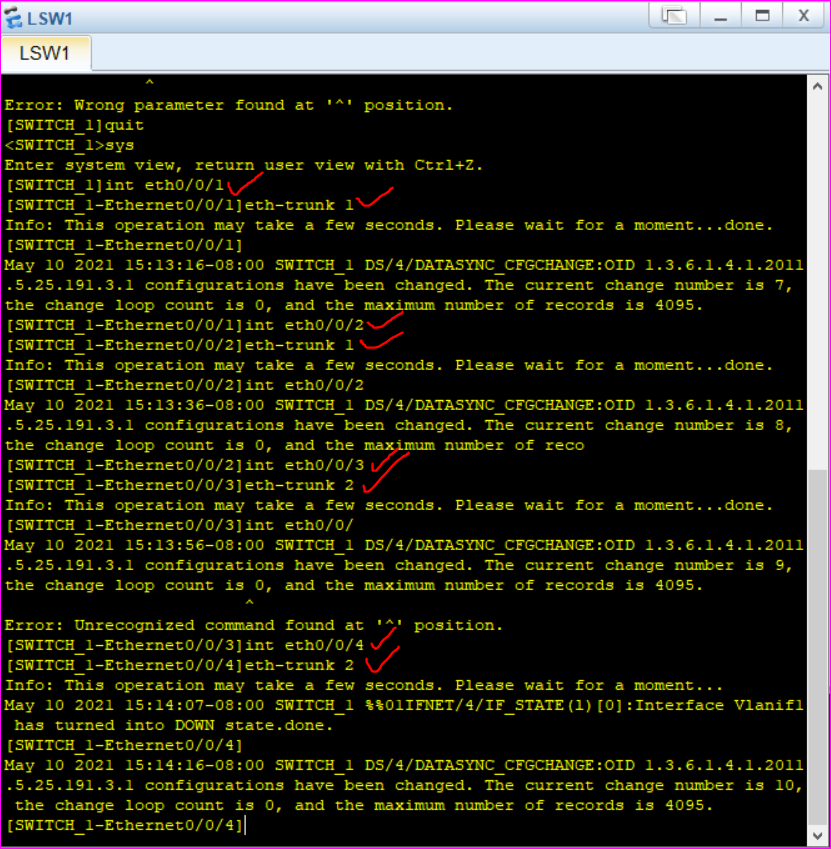
**Now we create 2 trunks on switch 1**



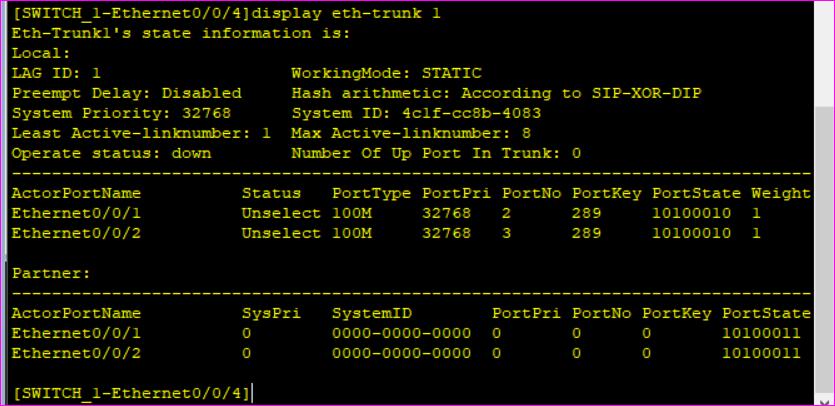
**Why do we used LACP-Static here?**

* The devices at the two ends of a link negotiate aggregation parameters by exchanging LACP packets.
* Then two devices determine the active interface and the inactive interface.
* It is necessary to manually create an Eth-Trunk and add members to it.
* If an active link fails, data forwarding is switched to the backup link with the highest priority and the status of backup link charges to active.

**Now we add interfaces to these trunks on switch 1**

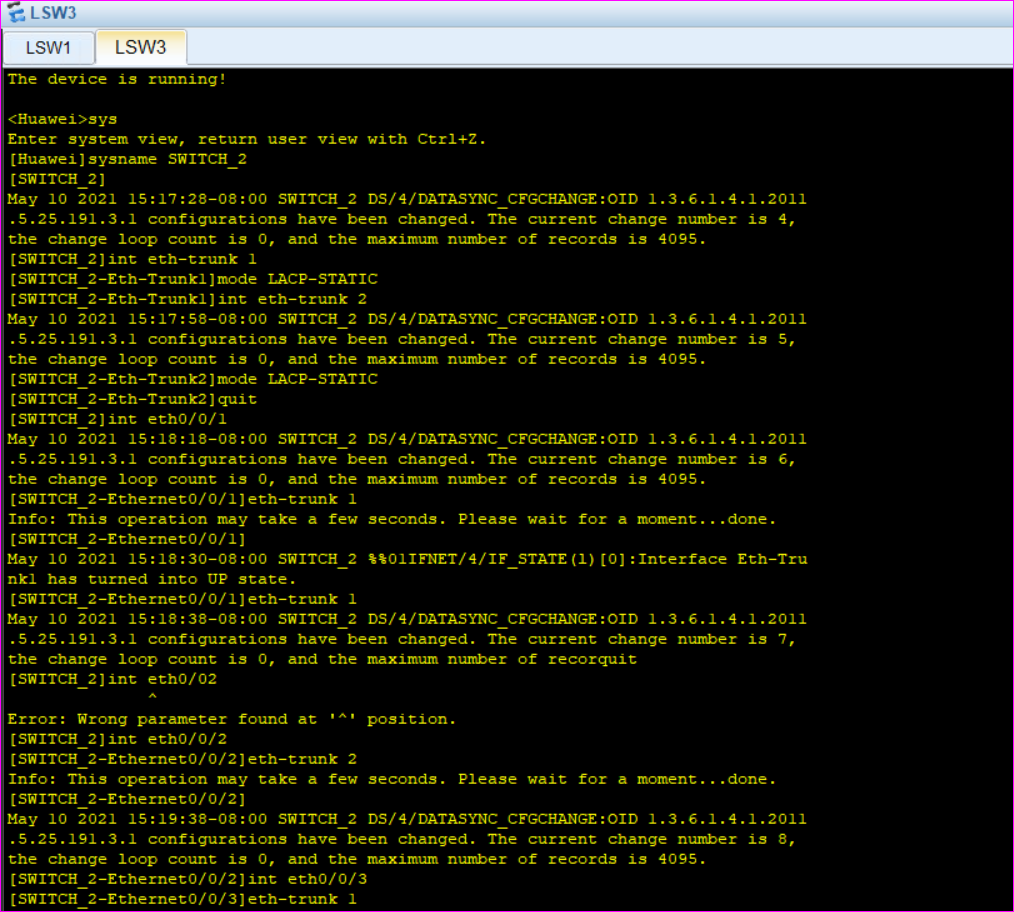
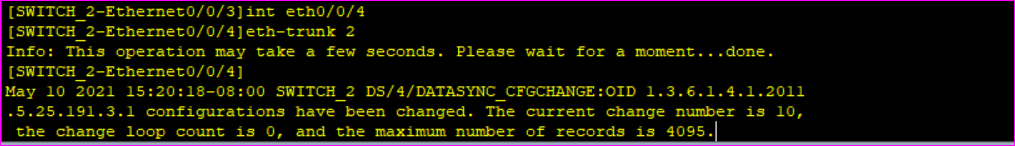


Now we display trunk 1

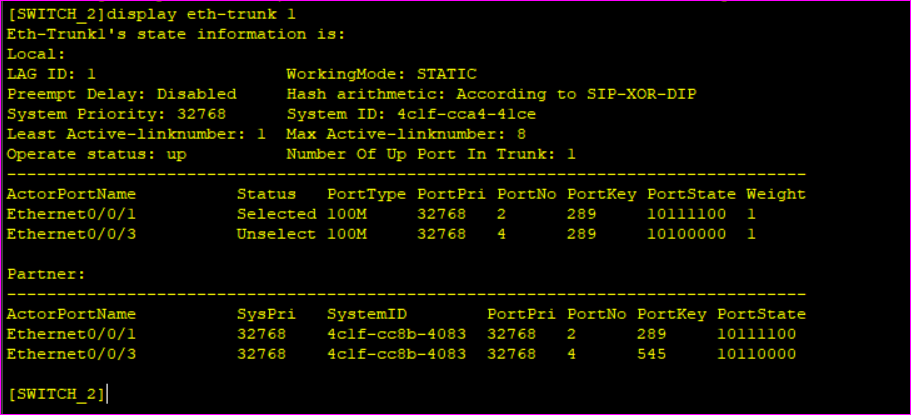


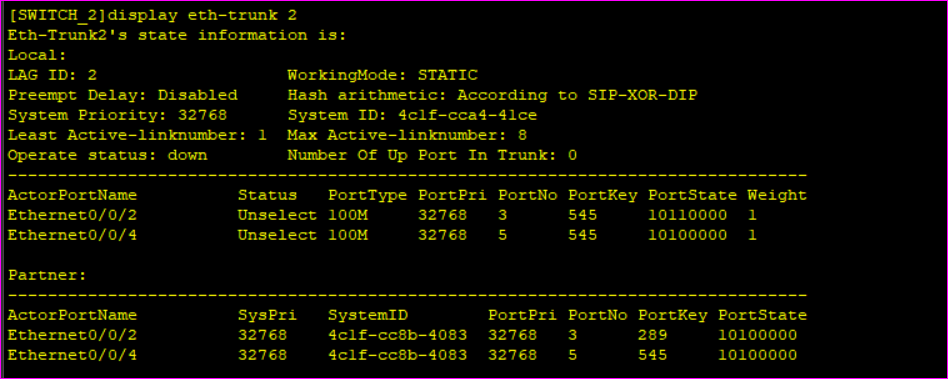
None of ports/interface is up.

**Now we create 2 trunks on Switch 2 and add interface 1 and 3 to trunk 1 and interface 2 and 4 to trunk 2**

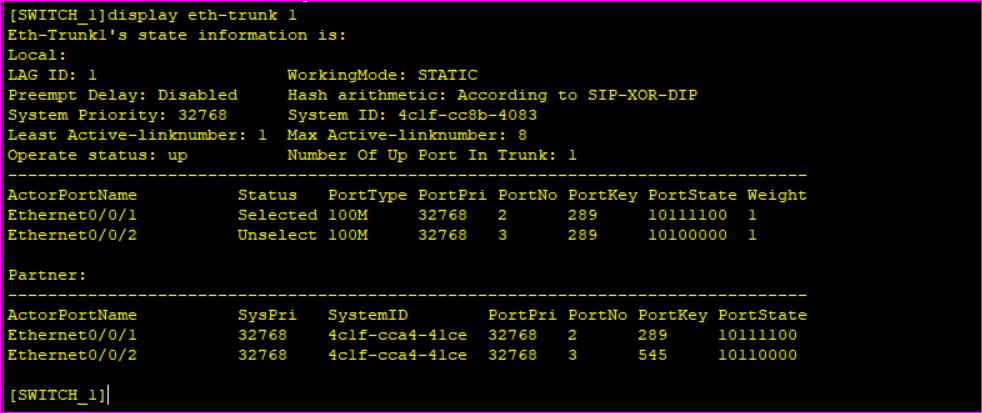
 

**Now we display aggregtion trunk 1 and trunk 2 of switch 2**

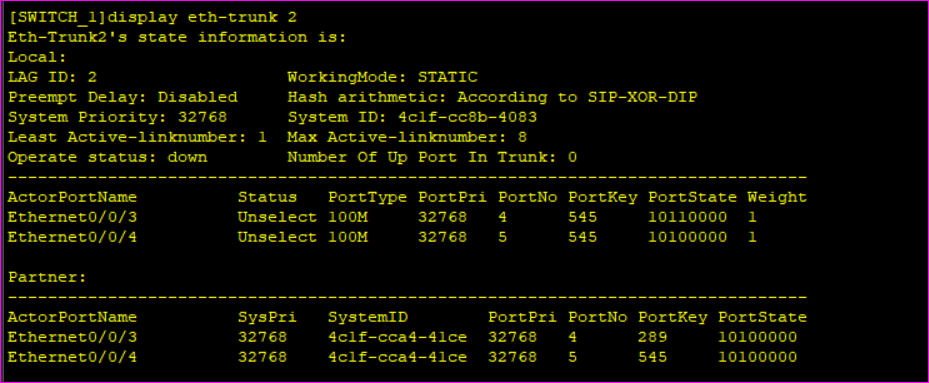




**Now again we display trunk 1 and trunk 2 on switch 2 and check the UP ports.**



1 port Is UP now in trunk 1.



**The End**