Multiple Spanning Tree Protocol

The Multiple Spanning Tree Protocol (MSTP) is an STP variant that allows multiple and independent spanning trees to be created over the same physical network. The parameters for each spanning tree can be configured separately, so as to cause a different network devices to be selected as the root bridge or different paths to be selected to form the loop-free topology. Consequently, a given physical interface can be blocked for some of the spanning trees and unblocked for others.

Having set up multiple spanning trees, the set of VLANs in use can be partitioned among them; for example, VLANs 1 - 100 can be assigned to spanning tree 1, VLANs 101 - 200 can be assigned to spanning tree 2, VLANs 201 - 300 can be assigned to spanning tree 3, and so on. Since each spanning tree has a different active topology with different active links, this has the effect of dividing the data traffic among the available redundant links based on the VLAN - a form of load balancing.

Restrictions for configuring MSTP

- RSTP is not supported. To support RSTP, all vlans are mapped to MSTI 0 when no instance is created for MSTP.
- PVSTP is *not* supported.
- Supports only 16 instances.
- Untagged EVCs do not participate in MST loop detection.

How to Configure MST Protocol

This section describes the procedure for configuring MSTP:

Enabling Multiple Spanning Tree Protocol

By default, MSTP is disabled on all interfaces. MSTP need not be enabled explicitly on each interfaces. By turning the global configuration on, it is enabled on all interfaces.

Configuring Multiple Spanning Tree Protocol

Describes steps to configure MST

SUMMARY STEPS

1. **configure**
2. **spanning-tree mode mst**
3. **spanning-tree mst configuration**
4. **instance** *vlan-id* **vlan** *vlan-range*
5. **name** *region*
6. **revision** *revision -number*
7. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure** **Example:** `Device> configure`	Enters global configuration mode.
Step 2	**spanning-tree mode mst** **Example:** `Device> spanning-tree mode mst`	Enables MSTP configuration mode.
Step 3	**spanning-tree mst configuration** **Example:** `Device(config)#spanning-tree mst configuration`	Enters the MSTP configuration submode.
Step 4	**instance** *vlan-id* **vlan** *vlan-range* **Example:** `Device(config-mstp-inst)# instance 1 vlan 450-480`	Maps the VLANs to an MST instance
Step 5	**name** *region* **Example:** `Device(config-mstp)# name m1`	Sets the name of the MSTP region.
Step 6	**revision** *revision -number* **Example:** `Device(config-mstp)#)revision 1`	Sets the revision level of the MSTP region.
Step 7	**end** **Example:** `Device(config-mstp-if)# end`	Returns to privileged EXEC mode.

Configuring Untagged EFP over MST Interface

Describes steps to configure untagged EFP over MST:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface number*
4. **no ip address**
5. **service instance** *number* **ethernet** *[name]*
6. **bridge-domain** *bridge-id*
7. **encapsulation untagged dot1q** {any|vlan-id [,vlan-id [-vlan-d]]}
8. **l2protocol peer stp**
9. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**interface** *interface number* **Example:** `Router(config)# interface gigabitEthernet 0/0/5`	Specifies the Gigabit Ethernet interface to configure, where: slot/subslot/port-Specifies the location of the interface.
Step 4	**no ip address** **Example:** `Router (config-if)# no ip address`	Disables the IP address on the interface.
Step 5	**service instance** *number* **ethernet** *[name]* **Example:** `Router (config-if)#service instance 200 ethernet`	Configure an EFP (service instance) and enter service instance configuration mode.
Step 6	**bridge-domain** *bridge-id* **Example:** `Router (config-if-srv)# bridge-domain from-encapsulation`	Creates a list of bridge domains for an EFP trunk port using the bridge-domain IDs derived from the encapsulation VLAN numbers.
Step 7	**encapsulation untagged dot1q** {any|vlan-id [,vlan-id [-vlan-d]]} **Example:**	Configures the encapsulation. Defines the matching criteria that maps the ingress dot1q or untagged frames on an interface for the appropriate service instance.

	Command or Action	Purpose
	`Router (config-if-srv)# encapsulation dot1q 20`	
Step 8	**l2protocol peer stp** **Example:** `Router (config-if-srv)# l2protocol peer stp`	Configures STP to peer with a neighbor on a port that has an EFP service instance.
Step 9	**end** **Example:** `Device(config-mstp-if)# end`	Returns to privileged EXEC mode.

Configuration Example

This example shows how to configure STP to peer with a neighbor on a service instance.

```
interface GigabitEthernet0/0/0
no ip address
negotiation auto
service instance trunk 10 ethernet
  encapsulation dot1q 10-20
  bridge-domain from-encapsulation
!
service instance 1024 ethernet
  encapsulation untagged
  l2protocol peer stp
  bridge-domain 1024
!
end
```