



Configuring Ethernet Virtual Connections

Ethernet Virtual Connection (EVC) as an association between two or more user network interfaces that identifies a point-to-point or multipoint-to-multipoint path within the service provider network. An EVC is a conceptual service pipe within the service provider network. A bridge domain is a local broadcast domain that is VLAN-ID-agnostic. An ethernet flow point (EFP) service instance is a logical interface that connects a bridge domain to a physical port or to an EtherChannel group in a router.

The Cisco ME 1200 NID supports the application software control modules and interfaces related to EVC.

- [How to Configure Ethernet Virtual Circuit, page 1](#)
- [Configuring Ethernet Virtual Circuit, page 1](#)
- [Creating a Policer, page 2](#)
- [EVC Control Entry \(ECE\) Configuration, page 4](#)
- [Ethernet Private Line or E-LAN, page 14](#)
- [Ethernet Virtual Private Line, page 16](#)
- [Other Commands For EVC Configuration, page 17](#)

How to Configure Ethernet Virtual Circuit

Configuring Ethernet Virtual Circuit

DETAILED STEPS

	Command or Action	Purpose
Step 1	ProvisionEVC Example: Switch# ProvisionEVC	Enters the ProvisionEVC mode.

	Command or Action	Purpose
Step 2	addEVC evcConfiguration {instance <i>evc-stance-id</i> learning {enable disable} nni-ports nni-vid <i>nni-vid-outer-tag</i> policer-id <i>policer-id</i>} Example: Switch(ProvisionEVC)# addEVC evcConfiguration nni-vid 101 Switch(ProvisionEVC)# addEVC evcConfiguration learning enable Switch(ProvisionEVC)# addEVC evcConfiguration nni-ports GigabitEthernet-6-NNI enable Switch(ProvisionEVC)# addEVC evcConfiguration policer-id 1	Adds the EVE configuration.
Step 3	addEVC review Example: Switch(ProvisionEVC)# addEVC review	Reviews the addEVC configuration.
Step 4	addEVC commit Example: Switch(ProvisionEVC)# addEVC commit	Sends the addEVC configuration to the Cisco ME 1200 NID.
Step 5	exit Example: Switch(ProvisionEVC)# exit Switch#	Exits the ProvisionEVC mode.

Example

```
Switch# ProvisionEVC
Switch(ProvisionEVC)# addEVC evcConfiguration instance 7
Switch(ProvisionEVC)# addEVC evcConfiguration nni-vid 101
Switch(ProvisionEVC)# addEVC evcConfiguration learning enable
Switch(ProvisionEVC)# addEVC evcConfiguration nni-ports GigabitEthernet-6-NNI enable
Switch(ProvisionEVC)# addEVC evcConfiguration policer-id 1
Switch(ProvisionEVC)# addEVC review
Switch(ProvisionEVC)# addEVC commit
```

AddEVC Commit Success!!!

Creating a Policer

DETAILED STEPS

	Command or Action	Purpose
Step 1	ProvisionEVC Example: Switch# ProvisionEVC	Enters the ProvisionEVC mode.

	Command or Action	Purpose
Step 2	<p>addPolicerEVC <i>evc-policer</i> {cbs <i>cbs-id</i> cir <i>committed-information-rate</i> ebs <i>excess-burst-size</i> eir <i>excess-information-rate</i> policer-id <i>policer-id</i> policer mode {color-aware coupled} policer-type {mef single} rate-type {data line} state {enabled disabled}}</p> <p>Example: Switch(ProvisionEVC) # addPolicerEVC evc-policer cir 20000 Switch(ProvisionEVC) # addPolicerEVC evc-policer ebs 30000 Switch(ProvisionEVC) # addPolicerEVC evc-policer eir 40000 Switch(ProvisionEVC) # addPolicerEVC evc-policer policer-id 1 Switch(ProvisionEVC) # addPolicerEVC evc-policer state enabled</p>	<p>Adds the EVC Policer.</p> <ul style="list-style-type: none"> • cbs—Specifies the committed burst size in bytes. • cir—Specifies the committed information rate. Multiply by 1000 to get rate in BPS. • ebs—Specifies the excess burst size in bytes. • eir—Specifies the excess information rate. • policer-id—Specifies the Policer ID. The valid values are from 1 to 1022. • policer-mode—Specifies the Policer mode—whether color-aware or coupled. • policer-type—Specifies the Policer mode—whether mef or single. • rate-type—Specifies the rate type policing—whether data or line. • state—Specifies the policer state—whether enabled or disabled.
Step 3	<p>addPolicerEVC review</p> <p>Example: Switch(ProvisionEVC) # addPolicerEVC review</p>	Displays the addPolicerEVC configuration.
Step 4	<p>addPolicerEVC commit</p> <p>Example: Switch(ProvisionEVC) # addPolicerEVC commit AddPolicerEVC Commit Success!!!</p>	Sends the configuration to the NID.
Step 5	<p>exit</p> <p>Example: Switch(ProvisionEVC) # exit Switch#</p>	Exits the ProvisionEVC mode.

Number of policers allowed are 1022. Use the following scale numbers for the ECE or EVC configuration with or without configuring QoS with tag pop 0, 1, or 2:

- Maximum 510 ECEs can be configured with or without configuring QoS (0-7 COS) with one NNI port to one UNI port.

- Maximum of 340 ECEs can be configured with or without configuring QoS (0-7 COS) with two NNI ports to one UNI or one NNI port to two UNI ports.
- Maximum of 255 ECEs can be configured with or without configuring QoS (0-7 COS) with three NNI ports to one UNI port or one NNI port to three UNI ports.
- Maximum of 170 ECEs can be configured with or without configuring QoS (0-7 COS) with four NNI ports to one UNI port or one NNI port to four UNI ports.
- Maximum of 128 ECEs can be configured with or without configuring QoS (0-7 COS) with five NNI ports to one UNI port or one NNI port to five UNI ports.

If OAM, HQoS, or EFP is configured on the Cisco ME 1200 NID, you can configure the following maximum service instances on every UNI interface:

- 64 ECE or EVC with eight COS classes.
- 104 ECE or EVC with four COS classes.
- 104 ECE or EVC with two COS classes.

EVC Control Entry (ECE) Configuration

ECE rules are used to divide the UNI traffic into two service classes.

This division of UNI traffic is achieved through:

- Simple NNI: All EVCs on the NNI port use the same QoS mapping and statistics.



Note This method requires fewer resources.

- Advanced NNI: Each EVC on the NNI port has separate QoS mapping and statistics.

In the following example, multiple ECE rules are created:

Configuring ECE Sample Rule 1

For rule 1, frames received on the UNI port with PCP 4-7 values are mapped to class 4 and sent with PCP 4 in the outer tag on the NNI port.

DETAILED STEPS

	Command or Action	Purpose
Step 1	ProvisionEVC Example: Switch# ProvisionEVC	Enters the ProvisionEVC mode.

	Command or Action	Purpose
Step 2	<p>addECE ece-configuration ece-id ece-id</p> <p>Example:</p> <pre>Switch(ProvisionEVC)# addECE ece-configuration ece-id 2</pre>	Adds ECE configuration.
Step 3	<p>addECE ece-configuration control actions {class {disabled specific specific-id} direction {bothnni-to-uni uni-to-nni} drop-precedence {disabled one zero} evc-id {none specific specific-ecv-id} policer-id {discard evc none specific specific-id} policy-id acl-policy-id tag-pop-count tag-pop-count}</p> <p>Example:</p> <pre>Switch(ProvisionEVC)# addECE ece-configuration control actions evc-id specific 7 Switch(ProvisionEVC)# addECE ece-configuration control actions tag-pop-count 1 Switch(ProvisionEVC)# addECE ece-configuration control actions policer-id specific 1 Switch(ProvisionEVC)# addECE ece-configuration control actions class specific 4</pre>	<p>Adds the ECE control action configuration.</p> <ul style="list-style-type: none"> • class—Specifies the ECE class. • direction—Specifies the direction of flow of traffic. • drop-precedence—Specifies the drop precedence (higher value means more dropping). • evc-id—Specifies the EVC ID. The valid specific values are from 1 to 1024. • policer-id—Specifies the policer ID. The valid specific values are from 1 to 1022. • policy-id—Specifies the ACL policy ID. The valid values are from 0 to 63. • tag-pop-count—Specifies the tagged VLAN count to be removed (either one or two outermost tags).
Step 4	<p>addECE ece-configuration control egress-inner-tag addECE ece-configuration control egress-inner-tag {dei-mode {classified drop-prec fixed} dei-value dei pcp-mode {classified fixed mapped} pcp-value pcp-value type type vlan-id vlan-id}</p> <p>Example:</p> <pre>Switch(ProvisionEVC)# addECE ece-configuration control egress-inner-tag dei-mode classified Switch(ProvisionEVC)# addECE ece-configuration control egress-inner-tag type none Switch(ProvisionEVC)# addECE ece-configuration control egress-inner-tag vlan-id 3</pre>	<p>Adds the ECE control egress inner tag rewrite configuration.</p> <ul style="list-style-type: none"> • dei-mode—Specifies the DEI mode—whether classified, drop precedence, or fixed. • dei-value—Specifies the DEI value. The valid values are 0 and 1. • pcp-mode—Specifies the PCP mode—whether classified, fixed, or mapped. • pcp-value—Specifies the PCP value. The valid values are from 1 to 7. • type—Specifies the type—whether c-tagged, none, s-custom, or s-tagged. • vlan-id—Specifies the VLAN ID. The valid values are from 1 to 4095.
Step 5	<p>addECE ece_configuration control egress-outer-tag {dei-mode {classified drop-prec fixed} dei-value dei-value mode {enabled disabled} pcp-mode {classified fixed mapped} pcp-value pcp-value vlan-id vlan-id</p>	<p>Adds the ECE control egress outer tag rewrite configuration.</p> <ul style="list-style-type: none"> • dei-mode—Specifies the DEI mode—whether classified, drop precedence, or fixed.

	Command or Action	Purpose
	<p>Example:</p> <pre>Switch(ProvisionEVC)# addECE ece-configuration control egress-outer-tag pcp-mode fixed Switch(ProvisionEVC)# addECE ece-configuration control egress-outer-tag pcp-value 4</pre>	<ul style="list-style-type: none"> • dei-value—Specifies the DEI value. The valid values are 0 and 1. • mode—Specifies the mode—whether enabled or disabled. • pcp-mode—Specifies the PCP mode—whether classified, fixed, or mapped. • pcp-value—Specifies the PCP value. The valid values are from 1 to 7. • vlan-id—Specifies the VLAN ID. The valid values are from 1 to 4095.
Step 6	<p>addECE ece-configuration control ingress-match {frame-type {any ipv4 {dest-ip-filter source-ip-filter} ipv6 {dest-ip-filter source-ip-filter}} inner-tag-match {match-fields match-type} mac-params {dmac-filer smac-filter} outer-tag-match {match-fields match-type} uni-ports {GigabitEthernet-1-UNI GigabitEthernet-2-UNI GigabitEthernet-3-UNI GigabitEthernet-4-UNI GigabitEthernet-5-UNI GigabitEthernet-6-UNI}}</p> <p>Example:</p> <pre>Switch(ProvisionEVC)# addECE ece-configuration control ingress-match uni-ports GigabitEthernet-2-UNI enable Switch(ProvisionEVC)# addECE ece-configuration control ingress-match outer-tag-match match-type c-tagged Switch(ProvisionEVC)# addECE ece-configuration control ingress-match outer-tag-match match-fields vlan-id-filter specific 100 Switch(ProvisionEVC)# addECE ece-configuration control ingress-match outer-tag-match match-fields inner-pcp val-4-7</pre>	<p>Adds the ECE control ingress inner tag rewrite configuration.</p> <ul style="list-style-type: none"> • frame-type—Specifies the type of frame relay. • inner-tag-match—Specifies the inner tag match value. • mac-params—Specifies the DMAC and SMAC default values. • outer-tag-match—Specifies the outer tag match value. • uni-ports—Specifies the GigabitEthernet UNI ports.
Step 7	<p>addECE review</p> <p>Example:</p> <pre>Switch(ProvisionEVC)# addECE review</pre>	<p>Reviews the addECE configuration.</p>
Step 8	<p>addECE commit</p> <p>Example:</p> <pre>Switch(ProvisionEVC)# addECE commit</pre>	<p>Sends the configuration to the NID.</p>
Step 9	<p>exit</p> <p>Example:</p> <pre>Switch(ProvisionEVE)# exit</pre>	<p>Exits the ProvisionEVE mode.</p>

Example

```

Switch# ProvisionEVC
Switch(ProvisionEVC)# addECE ece-configuration ece-id 1
Switch(ProvisionEVC)# addECE ece-configuration ece-id 1
Switch(ProvisionEVC)# addECE ece-configuration control actions evc-id specific 777
Switch(ProvisionEVC)# addECE ece-configuration control actions tag-pop-count 1
Switch(ProvisionEVC)# addECE ece-configuration control actions policer-id none
Switch(ProvisionEVC)# addECE ece-configuration control ingress-match uni-ports
GigabitEthernet-2-UNI enable
Switch(ProvisionEVC)# addECE ece-configuration control ingress-match outer-tag_match
match-type c-tagged
Switch(ProvisionEVC)# addECE ece-configuration control ingress-match outer-tag_match
match-fields vlan-id-filter specific 100
Switch(ProvisionEVC)# addECE ece-configuration control ingress-match outer-tag_match
match-fields inner-dei any
Switch(ProvisionEVC)# addECE ece-configuration control ingress-match outer-tag_match
match-fields inner-pcp val-any
Switch(ProvisionEVC)# addECE review
Commands in queue:
  addECE ece-configuration ece-id 1
  addECE ece-configuration ece-id 1
  addECE ece-configuration control actions evc-id specific 777
  addECE ece-configuration control actions tag-pop-count 1
  addECE ece-configuration control actions policer-id none
  addECE ece-configuration control ingress-match uni-ports GigabitEthernet-2-UNI enable
  addECE ece-configuration control ingress-match outer-tag-match match-type c-tagged
  addECE ece-configuration control ingress-match outer-tag-match match-fields vlan-id-filter
  specific 100
  addECE ece-configuration control ingress-match outer-tag-match match-fields inner-dei any

  addECE ece-configuration control ingress-match outer-tag-match match-fields inner-pcp
  val-any

Switch(ProvisionEVC)# addECE commit

Clearing Socket 4 Clearing Socket 4
AddECE Commit Success!!!

```

Configuring ECE-v3

SUMMARY STEPS

1. ProvisionEVC
2. addECE-v3 eceConfiguration-v3 ece-id ece-id
3. addECE-v3 eceConfiguration-v3 control action {class {disabled | specific *specific-id*} | direction {bothnni-to-uni | uni-to-nni} | drop-precedence {disabled | one | zero} | evc-id {none | specific *specific-ecv-id*} | l2cp-dmac {cisco | custom} | l2cp-mode {discard | forward | peer | tunnel} | policer-id {discard | evc | none | specific *specific-id*} | policy-id *acl-policy-id* | rule-type {both | rx | tx} | tag-pop-count {tag-pop-count} | tx-lookup {isdx | vid-only | vid-ppp}}
4. addECE-v3 eceConfiguration-v3 control egress-inner-tag {dei-mode {classified | drop-prec | fixed} | dei-value *dei* | pcp-mode {classified | fixed | mapped} | pcp-value *pcp-value* | type *type* | vlan-id *vlan-id*}
5. addECE-v3 eceConfiguration-v3 control egress-outer-tag {dei-mode {classified | drop-prec | fixed} | dei-value *dei-value* | mode {enabled | disabled} | pcp-mode {classified | fixed | mapped} | pcp-value *pcp-value* | vlan-id *vlan-id*}
6. addECE-v3 eceConfiguration-v3 control ingress-match {frame-type {any | ipv4 {dest-ip-filter | source-ip-filter} | ipv6 {dest-ip-filter | source-ip-filter}}} | inner-tag-match {match-fields | match-type} | mac-params {dmac-filer | smac-filter} | outer-tag-match {match-fields | match-type} | uni-ports {GigabitEthernet-1-UNI | GigabitEthernet-2-UNI | GigabitEthernet-3-UNI | GigabitEthernet-4-UNI | GigabitEthernet-5-UNI | GigabitEthernet-6-UNI}}
7. addECE-v3 eceConfiguration-v3 control l2cp-parameters {cdp | cisco-cfm | cisco-stp | cisco-vlan | dot1x | dtp | elmi | gmrp | gvrp | lacp | lamp | lldp | loam | pagp | pause | pb | pb-gvrp | pvst | stp | uld | vtp}
8. addECE-v3 review
9. addECE-v3 commit
10. exit

DETAILED STEPS

	Command or Action	Purpose
Step 1	ProvisionEVC Example: Switch# ProvisionEVC	Enters the ProvisionEVC mode.
Step 2	addECE-v3 eceConfiguration-v3 ece-id ece-id Example: Switch(ProvisionEVC)# addECE ece-configuration ece-id 2	Adds ECE configuration.
Step 3	addECE-v3 eceConfiguration-v3 control action {class {disabled specific <i>specific-id</i> } direction {bothnni-to-uni uni-to-nni} drop-precedence {disabled one zero}	Adds the ECE control action configuration. • class—Specifies the ECE class.

	Command or Action	Purpose
	<p>evc-id { none specific <i>specific-evt-id</i> } l2cp-dmac { cisco custom } l2cp-mode { discard forward peer tunnel } policer-id { discard evc none specific <i>specific-id</i> } policy-id <i>acl-policy-id</i> rule-type { both rx tx } tag-pop-count { <i>tag-pop-count</i> } tx-lookup { isdx vid-only vid-pcp }</p> <p>Example:</p> <pre>Switch(ProvisionEVC)# addECE ece-configuration control actions evc-id specific 7 Switch(ProvisionEVC)# addECE ece-configuration control actions tag-pop-count 1 Switch(ProvisionEVC)# addECE ece-configuration control actions policer-id specific 1 Switch(ProvisionEVC)# addECE ece-configuration control actions class specific 4</pre>	<ul style="list-style-type: none"> • direction—Specifies the direction of flow of traffic. • drop-precedence—Specifies the drop precedence (higher value means more dropping). • evc-id—Specifies the EVC ID. The valid specific values are from 1 to 1024. • policer-id—Specifies the policer ID. The valid specific values are from 1 to 1022. • policy-id—Specifies the ACL policy ID. The valid values are from 0 to 63. • tag-pop-count—Specifies the tagged VLAN count to be removed (either one or two outermost tags). • l2cp-dmac—Specifies the L2CP tunnel DMAC. <ul style="list-style-type: none"> ◦ cisco—Specifies Cisco Generic BPDU Tunneling DMAC. ◦ custom—Specifies custom DMAC. • l2cp-mode—Specifies the L2CP mode. <ul style="list-style-type: none"> ◦ discard—Discard L2CP frames. ◦ forward—Forward L2CP frames. ◦ peer—Peer L2CP frames. ◦ tunnel—Tunnel L2CP frames. • rule-type—Specifies the rule type. <ul style="list-style-type: none"> ◦ both—Specifies both rule type. ◦ rx—Specifies rx rule type . ◦ tx—Specifies tx rule type . • tx-lookup—Specifies tx-lookup. <ul style="list-style-type: none"> ◦ isdx—Specifies isdx lookup. ◦ vid-only—Specifies vid-only lookup . ◦ vid-pcp—Specifies vid-pcp lookup .
<p>Step 4</p>	<p>addECE-v3 eceConfiguration-v3 control egress-inner-tag { dei-mode { classified drop-prec fixed } dei-value <i>dei</i> pcp-mode { classified fixed }</p>	<p>Adds the ECE control egress inner tag rewrite configuration.</p> <ul style="list-style-type: none"> • dei-mode—Specifies the DEI mode—whether classified, drop precedence, or fixed.

	Command or Action	Purpose
	<p>mapped pcp-value <i>pcp-value</i> type <i>type</i> vlan-id <i>vlan-id</i>}</p> <p>Example:</p> <pre>Switch(ProvisionEVC)# addECE-v3 ece-configuration control egress-inner-tag dei-mode classified Switch(ProvisionEVC)# addECE-v3 ece-configuration control egress-inner-tag type none Switch(ProvisionEVC)# addECE-v3 ece-configuration control egress-inner-tag vlan-id 3</pre>	<ul style="list-style-type: none"> • dei-value—Specifies the DEI value. The valid values are 0 and 1. • pcp-mode—Specifies the PCP mode—whether classified, fixed, or mapped. • pcp-value—Specifies the PCP value. The valid values are from 1 to 7. • type—Specifies the type—whether c-tagged, none, s-custom, or s-tagged. • vlan-id—Specifies the VLAN ID. The valid values are from 1 to 4095.
Step 5	<p>addECE-v3 eceConfiguration-v3 control egress-outer-tag {dei-mode {classified drop-prec fixed} dei-value <i>dei-value</i> mode {enabled disabled} pcp-mode {classified fixed mapped} pcp-value <i>pcp-value</i> vlan-id <i>vlan-id</i></p> <p>Example:</p> <pre>Switch(ProvisionEVC)# addECE-v3 ece-configuration control egress-outer-tag pcp-mode fixed Switch(ProvisionEVC)# addECE-v3 ece-configuration control egress-outer-tag pcp-value 4</pre>	<p>Adds the ECE control egress outer tag rewrite configuration.</p> <ul style="list-style-type: none"> • dei-mode—Specifies the DEI mode—whether classified, drop precedence, or fixed. • dei-value—Specifies the DEI value. The valid values are 0 and 1. • mode—Specifies the mode—whether enabled or disabled. • pcp-mode—Specifies the PCP mode—whether classified, fixed, or mapped. • pcp-value—Specifies the PCP value. The valid values are from 1 to 7. • vlan-id—Specifies the VLAN ID. The valid values are from 1 to 4095.
Step 6	<p>addECE-v3 eceConfiguration-v3 control ingress-match {frame-type {any ipv4 {dest-ip-filter source-ip-filter} ipv6 {dest-ip-filter source-ip-filter}} inner-tag-match {match-fields match-type} mac-params {dmac-filer smac-filter} outer-tag-match {match-fields match-type} uni-ports {GigabitEthernet-1-UNI GigabitEthernet-2-UNI GigabitEthernet-3-UNI GigabitEthernet-4-UNI GigabitEthernet-5-UNI GigabitEthernet-6-UNI}}</p> <p>Example:</p> <pre>Switch(ProvisionEVC)# addECE-v3 ece-configuration control ingress-match uni-ports GigabitEthernet-2-UNI enable Switch(ProvisionEVC)# addECE-v3 ece-configuration control ingress-match outer-tag-match match-type c-tagged Switch(ProvisionEVC)# addECE-v3 ece-configuration control ingress-match outer-tag-match match-fields</pre>	<p>Adds the ECE control ingress inner tag rewrite configuration.</p> <ul style="list-style-type: none"> • frame-type—Specifies the type of frame relay. • inner-tag-match—Specifies the inner tag match value. • mac-params—Specifies the DMAC and SMAC default values. • outer-tag-match—Specifies the outer tag match value. • uni-ports—Specifies the GigabitEthernet UNI ports.

	Command or Action	Purpose
	<pre>vlan-id-filter specific 100 Switch(ProvisionEVC)# addECE-v3 ece-configuration control ingress-match outer-tag-match match-fields inner-pcp val-4-7</pre>	
Step 7	<p>addECE-v3 eceConfiguration-v3 control l2cp-parameters { cdp cisco-cfm cisco-stp cisco-vlan dot1x dtp elmi gmrp gvrp lacp lamp lldp loam pagp pause pb pb-gvrp pvst stp uld vtp }</p> <p>Example:</p> <pre>Switch(ProvisionEVC)# addECE-v3 ece-configuration control l2cp-parameters dot1x</pre>	<p>Adds the ECE control ingress inner tag rewrite configuration.</p> <ul style="list-style-type: none"> • cdp—Specifies cdp frames. • cisco-cfm —Specifies Cisco CFM frames. • cisco-stp—Specifies the Cisco STP Uplink Fast frames. • cisco-vlan —Specifies Cisco VLAN bridge frames. • dot1x —Specifies the 802.1X frames. • dtp—Specifies the DTP frames. • elmi—Specifies the E-LMI frames. • gmrp —Specifies the GMRP frames. • gvrp —Specifies the GVRP frames. • lacp —Specifies the LACP frames. • lamp —Specifies the LAMP frames. • lldp —Specifies the LLDP frames. • loam —Specifies the Link OAM frames. • pagp—Specifies the PAGP frames. • pause —Specifies the Pause frames. • pb—Specifies the PB frames. • pb-gvrp —Specifies the PB GVRP frames. • pvst—Specifies the PVST frames. • stp —Specifies the STP frames. • uld —Specifies the ULD frames. • vtp —Specifies the VTP frames.
Step 8	<p>addECE-v3 review</p> <p>Example:</p> <pre>Switch(ProvisionEVC)# addECE-v3 review</pre>	Reviews the addECE configuration.
Step 9	<p>addECE-v3 commit</p> <p>Example:</p> <pre>Switch(ProvisionEVC)# addECE-v3 commit</pre>	Sends the configuration to the NID.

	Command or Action	Purpose
Step 10	exit Example: Switch(ProvisionEVE)# exit Switch(config-controller)#	Exits to the controller configuration mode.

Example

```
(ProvisionEVE)# addece-v3 review
Commands in queue: 7
    addECE-v3 eceConfiguration-v3 ece-id 98
    addECE-v3 eceConfiguration-v3 control actions evc-id specific 97
    addECE-v3 eceConfiguration-v3 control ingress-match uni-ports 6-4
    addECE-v3 eceConfiguration-v3 control ingress-match frame-type l2cp
    addECE-v3 eceConfiguration-v3 control actions l2cp-mode tunnel
    addECE-v3 eceConfiguration-v3 control actions l2cp-dmac cisco
    addECE-v3 eceConfiguration-v3 control l2cp-parameters dot1x

(ProvisionEVC)# addece-v3 commit

AddECE-v3 Commit Success!!!(ProvisionEVC)#
```

Configuring the ECE Sample Rule 2

For rule 2, other frames received on the UNI port are mapped to class 0 and sent with PCP 0 in the outer tag on the NNI port.



Note

The configuration steps are similar to the ones mentioned in the [Configuring ECE Sample Rule 1](#) section.

Example

```
Switch# ProvisionEVC
Switch(ProvisionEVC)# addECE ece-configuration control actions evc-id specific 7

Switch(ProvisionEVC)# addECE ece-configuration control actions tag-pop-count 1
```

```

Switch(ProvisionEVC) # addECE ece-configuration control actions policer-id specific 1
Switch(ProvisionEVC) # addECE ece-configuration control actions class specific 0
Switch(ProvisionEVC) # addECE ece-configuration control ingress-match uni-ports
GigabitEthernet-2-UNI enable
Switch(ProvisionEVC) # addECE ece-configuration control ingress-match outer-tag-match
match-type c-tagged
Switch(ProvisionEVC) # addECE ece-configuration control ingress-match outer-tag-match
match-fields vlan-id-filter specific 99
Switch(ProvisionEVC) # addECE ece-configuration control ingress-match outer-tag-match
match-fields inner-pcp val-0-3
Switch(ProvisionEVC) # addECE ece-configuration control egress-outer-tag pcp-mode fixed
Switch(ProvisionEVC) # addECE ece-configuration control egress-outer-tag pcp-value 0
Switch(ProvisionEVC) # addECE commit

```

Configuring ECE Sample Rule 3

For rule 3, frames received on the NNI port 6 with S-tag 101 and C-tag 100 with any PCP values can be mapped to class 4 and sent with PCP 4 on the UNI port.



Note

The configuration steps are similar to the ones mentioned in the [Configuring ECE Sample Rule 1](#) section.

Example

```

Switch# ProvisionEVC
Switch(ProvisionEVC) # addECE ece-configuration ece-id 3
Switch(ProvisionEVC) # addECE ece-configuration control actions evc-id specific 7
Switch(ProvisionEVC) # addECE ece-configuration control actions class specific 4
Switch(ProvisionEVC) # addECE ece-configuration control ingress-match uni-ports
GigabitEthernet-2-UNI enable
Switch(ProvisionEVC) # addECE ece-configuration control egress-inner-tag pcp-mode fixed
Switch(ProvisionEVC) # addECE ece-configuration control egress-inner-tag pcp-value 4
Switch(ProvisionEVC) # addECE commit

```

Configuring ECE Sample Rule 4

For rule 4, insert a new c-tag in frames forwarding to the NNI port.



Note

The configuration steps are similar to the ones mentioned in the [Configuring ECE Sample Rule 1](#) section.

Example

```

Switch# ProvisionEVC
Switch(ProvisionEVC) # addECE ece-configuration ece-id 4
Switch(ProvisionEVC) # addECE ece-configuration control actions evc-id specific 7
Switch(ProvisionEVC) # addECE ece-configuration control actions tag-pop-count 1
Switch(ProvisionEVC) # addECE ece-configuration control actions policer-id specific 1
Switch(ProvisionEVC) # addECE ece-configuration control ingress-match uni-ports
GigabitEthernet-2-UNI enable
Switch(ProvisionEVC) # addECE ece-configuration control ingress-match outer-tag-match
match-type c-tagged
Switch(ProvisionEVC) # addECE ece-configuration control ingress-match outer-tag-match
match-fields vlan-id-filter specific 99
Switch(ProvisionEVC) # addECE ece-configuration control egress-inner-tag type c-tagged
Switch(ProvisionEVC) # addECE ece-configuration control egress-inner-tag vlan-id 77
Switch(ProvisionEVC) # addECE commit

```

Configuring ECE Sample Rule 5

For rule 5, insert a new tag in frames forwarding to the UNI port, the outer tag for NNI - UNI direction for the ECE.



Note

The configuration steps are similar to the ones mentioned in the [Configuring ECE Sample Rule 1](#) section.

Example

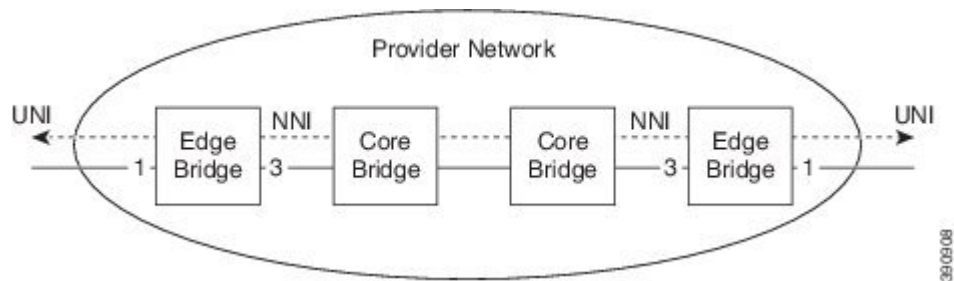
```
Switch# ProvisionEVC
Switch(ProvisionEVC) # addECE ece-configuration ece-id 5
Switch(ProvisionEVC) # addECE ece-configuration control actions direction nni-to-uni
Switch(ProvisionEVC) # addECE ece-configuration control actions direction nni-to-uni
                        ---> This field is mandatory to pass
Switch(ProvisionEVC) # addECE ece-configuration control actions direction nni-to-uni
                        ---> Map this ECE rule to an EVC configured
                        above.
Switch(ProvisionEVC) # addECE ece-configuration control ingress-match uni-ports
GigabitEthernet-2-UNI enable
Switch(ProvisionEVC) # addECE ece-configuration control egress-outer-tag enabled
Switch(ProvisionEVC) # addECE ece-configuration control egress-outer-tag vlan-id 78
Switch(ProvisionEVC) # addECE commit
```

Ethernet Private Line or E-LAN

Ethernet Private Line (EPL) or E-LAN and Ethernet Virtual Private Line (EVPL) are Carrier Ethernet data services defined by the Metro Ethernet Forum. E-LAN provides a point-to-point Ethernet virtual connection (EVC) between a pair of dedicated user-network interfaces (UNIs), with a high degree of transparency.

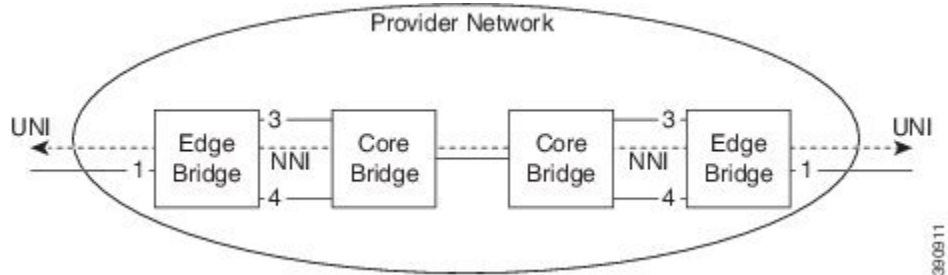
The following diagrams show a Provider Network offering various types of E-LAN between two UNIs.

Figure 1: An Unprotected Ethernet-LAN



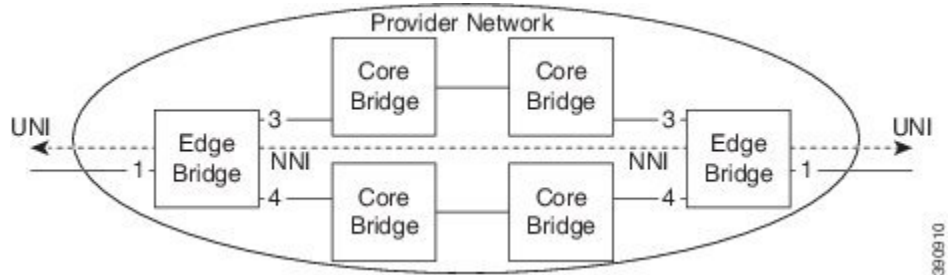
The following diagram shows an ethernet private (EP) line with 1-to-1 port protection on the network-network interface (NNI) side. This setup requires more resources compared to the unprotected EP-Line, because rules must be added for both NNI ports.

Figure 2: Port Protected E-LAN



The following diagram shows an ethernet LAN with ring protection on the network-network interface (NNI) side. The resource consumption is similar to the port protection scenario, because rules are added for each NNI port.

Figure 3: Ring-Protected E-LAN



The following sections describe how to configure the Edge Bridges.

Configuring ECE for E-LAN Between Two UNI and NNI Ports

For more information on configuring ECE, see the [EVC Control Entry \(ECE\) Configuration](#) section.

ECE Rule 1 on UNI Interface 3

```
Switch # ProvisionEVC
Switch(ProvisionEVC) # addECE ece-configuration ece-id 6
Switch(ProvisionEVC) # addECE ece-configuration control actions tag-pop-count 1
Switch(ProvisionEVC) # addECE ece-configuration control actions evc-id specific 9

Switch(ProvisionEVC) # addECE ece-configuration control actions policer-id specific 1
Switch(ProvisionEVC) # addECE ece-configuration control ingress-match uni-ports
GigabitEthernet-3-UNI enable
Switch(ProvisionEVC) # addECE ece-configuration control ingress-match outer-tag_match
match-type c-tagged
Switch(ProvisionEVC) # addECE ece-configuration control ingress-match outer-tag-match
match-fields vlan-id-filter specific 500
Switch(ProvisionEVC) # addECE review
Switch(ProvisionEVC) # addECE commit
```

ECE Rule 1 on UNI Interface 2

```

Switch# ProvisionEVC
Switch(ProvisionEVC)# addECE ece-configuration ece-id 6
Switch(ProvisionEVC)# addECE ece-configuration control actions tag-pop-count 1
Switch(ProvisionEVC)# addECE ece-configuration control actions evc-id specific 9

Switch(ProvisionEVC)# addECE ece-configuration control actions policer-id specific 1
Switch(ProvisionEVC)# addECE ece-configuration control ingress-match uni-ports
GigabitEthernet-2-UNI enable
Switch(ProvisionEVC)# addECE ece-configuration control ingress-match outer-tag-match
match-type c-tagged
Switch(ProvisionEVC)# addECE ece-configuration control ingress-match outer-tag-match
match-fields vlan-id-filter specific 600
Switch(ProvisionEVC)# addECE review
Switch(ProvisionEVC)# addECE commit

```

**Note**

Tag POP count is 1 for E-LAN service, that is, all frames are passed to the EVC popping one tag in the direction from UNI to NNI and pushing one tag in the other direction.

Configuring EVC for E-LAN

For more information on configuring EVC, see the [Configuring Ethernet Virtual Circuit](#) section.

Example

```

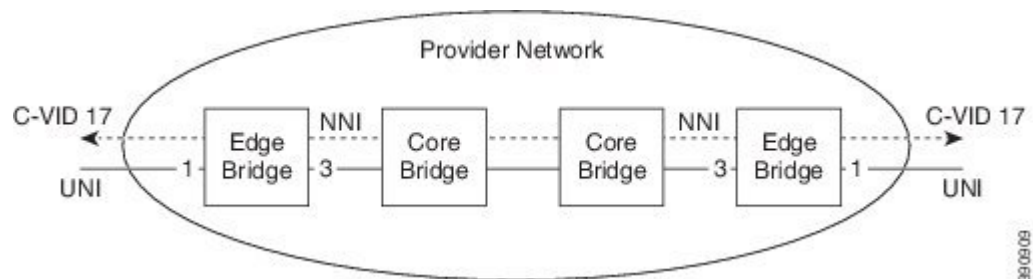
Switch# ProvisionEVC
Switch(ProvisionEVC)# addEVC evcConfiguration instance 9
Switch(ProvisionEVC)# addEVC evcConfiguration internal-vid 400
Switch(ProvisionEVC)# addEVC evcConfiguration nni-vid 400
Switch(ProvisionEVC)# addEVC evcConfiguration learning enable
Switch(ProvisionEVC)# addEVC evcConfiguration nni-ports GigabitEthernet-6-NNI enable
Switch(ProvisionEVC)# addEVC evcConfiguration policer-id 1
Switch(ProvisionEVC)# addEVC review
Switch(ProvisionEVC)# addEVC commit

```

Ethernet Virtual Private Line

The following diagram shows an unprotected ethernet virtual private line (EVP-Line) forwarding frames with C-VID = 17 between the user-network interface (UNI) ports.

Figure 4: Unprotected EVP-Line



This following section describes the configuration of the EVPL service between the UNI and NNI ports.

Configuring ECE For EVPL Service

For more information on configuring ECE, see the [EVC Control Entry \(ECE\) Configuration](#) section.

Example

```
Switch# ProvisionEVC
Switch(ProvisionEVC)# addECE ece-configuration ece-id 6
Switch(ProvisionEVC)# addECE ece-configuration control actions evc-id specific 8

Switch(ProvisionEVC)# addECE ece-configuration control actions policer-id specific 1
Switch(ProvisionEVC)# addECE ece-configuration control ingress-match uni-ports
GigabitEthernet-3-UNI enable
Switch(ProvisionEVC)# addECE ece-configuration control ingress-match outer-tag-match
match-type c-tagged
Switch(ProvisionEVC)# addECE ece-configuration control ingress-match outer-tag-match
match-fields vlan-id-filter range 300-350
Switch(ProvisionEVC)# addECE review
Switch(ProvisionEVC)# addECE commit
```



Note

The above ECE rule allows all VLANs ranging from 300 to 350. However, if you need to filter specific VLANs then you must create individual ECE rules. For more information, see [Configuring ECE Sample Rule 1](#).

Configuring EVC For EVPL Service

For more information on configuring EVC, see the [Configuring Ethernet Virtual Circuit](#) section.

Example

```
Switch# ProvisionEVC
Switch(ProvisionEVC)# addEVC evcConfiguration instance 8
Switch(ProvisionEVC)# addEVC evcConfiguration internal-vid 200
Switch(ProvisionEVC)# addEVC evcConfiguration nni-vid 200
Switch(ProvisionEVC)# addEVC evcConfiguration learning enable
Switch(ProvisionEVC)# addEVC evcConfiguration nni-ports GigabitEthernet-5-NNI enable
Switch(ProvisionEVC)# addEVC evcConfiguration policer-id 1
Switch(ProvisionEVC)# addEVC review
Switch(ProvisionEVC)# addEVC commit
```

Other Commands For EVC Configuration

Clearing EVC Statistics

clearEVCStatistics clear-evc-stats {all | ece | evc-id | physical-port}

```
Switch(ProvisionEVC)# clearEVCStatistics clear-evc-stats all
Switch(ProvisionEVC)# clearEVCStatistics clear-evc-stats ece ece-id <1-1024>
Switch(ProvisionEVC)# clearEVCStatistics clear-evc-stats ece physical-port <1-6>
Switch(ProvisionEVC)# clearEVCStatistics clear-evc-stats evc-id <1-1024>
Switch(ProvisionEVC)# clearEVCStatistics clear-evc-stats physical-port <1-6>
Switch(ProvisionEVC)# clearEVCStatistics review
Switch(ProvisionEVC)# clearEVCStatistics commit
```

Using the Default Configuration

default

```
Switch(ProvisionEVC)# default
```



Note

This command resets all configuration to default values.

Deleting Configuration

Use this command to delete the ECE configuration.

deleteECE delete-ece-request *ece-id*

```
Switch(ProvisionEVC)# deleteECE delete-ece-request <1-1024>
Switch(ProvisionEVC)# deleteECE review
Switch(ProvisionEVC)# deleteECE commit
```

Use this command to delete the EVC configuration.

deleteEVC deleteEVCrequest *evc-id*

```
Switch(ProvisionEVC)# deleteEVC deleteEVCrequest <1-1024>
Switch(ProvisionEVC)# deleteEVC review
Switch(ProvisionEVC)# deleteEVC commit
```

Use this command to delete the EVC Policer request.

deletePolicerEVC *evc-policer-id evc-policer-id*

```
Switch(ProvisionEVC)# deletePolicerEVC evc-policer-id <1-1024>
Switch(ProvisionEVC)# deletePolicerEVC review
Switch(ProvisionEVC)# deletePolicerEVC commit
```

Editing Configuration

Use this command to edit the ECE configuration.

editECEConfiguration ece-update-configuration {*ece-id ece-id* | update {class | direction | drop-precedence | *evc-id* | *policer-id* | rule-type | tag-pop-count | tx-lookup } }

```
Switch(ProvisionEVC)# editECEConfiguration ece-update-configuration ece-id <1-1024>
Switch(ProvisionEVC)# editECEConfiguration ece-update-configuration update class disabled
Switch(ProvisionEVC)# editECEConfiguration ece-update-configuration update class specific
<0-7>
Switch(ProvisionEVC)# editECEConfiguration ece-update-configuration update direction both
Switch(ProvisionEVC)# editECEConfiguration ece-update-configuration update direction
nni-to-uni
Switch(ProvisionEVC)# editECEConfiguration ece-update-configuration update direction
uni-to-nni
Switch(ProvisionEVC)# editECEConfiguration ece-update-configuration update drop-precedence
disabled
Switch(ProvisionEVC)# editECEConfiguration ece-update-configuration update drop-precedence
one
Switch(ProvisionEVC)# editECEConfiguration ece-update-configuration update drop-precedence
zero
Switch(ProvisionEVC)# editECEConfiguration ece-update-configuration update evc-id none
Switch(ProvisionEVC)# editECEConfiguration ece-update-configuration update evc-id specific
<1-1024>
Switch(ProvisionEVC)# editECEConfiguration ece-update-configuration update policer-id discard
Switch(ProvisionEVC)# editECEConfiguration ece-update-configuration update policer-id evc
Switch(ProvisionEVC)# editECEConfiguration ece-update-configuration update policer-id none
Switch(ProvisionEVC)# editECEConfiguration ece-update-configuration update policer-id
specific
Switch(ProvisionEVC)# editECEConfiguration ece-update-configuration update policy-id <0-63>
Switch(ProvisionEVC)# editECEConfiguration ece-update-configuration update rule-type both
Switch(ProvisionEVC)# editECEConfiguration ece-update-configuration update rule-type rx
Switch(ProvisionEVC)# editECEConfiguration ece-update-configuration update rule-type tx
Switch(ProvisionEVC)# editECEConfiguration ece-update-configuration update tag-pop-count
<0-2>
Switch(ProvisionEVC)# editECEConfiguration ece-update-configuration update tx-lookup isdx
```

```
Switch(ProvisionEVC) # editECEConfiguration ece-update-configuration update tx-lookup vid-only
Switch(ProvisionEVC) # editECEConfiguration ece-update-configuration update tx-lookup vid-pc
Switch(ProvisionEVC) # editECEConfiguration review
Switch(ProvisionEVC) # editECEConfiguration commit
```

Use this command to edit the EVC configuration.

editEVCConfiguration evcupdateConfiguration {instance *instance_id* | update {internal-vid | learning | nni-ports | nni-vid | policer-id}}

```
Switch(ProvisionEVC) # editEVCConfiguration evcupdateConfiguration instance <1-1024>
Switch(ProvisionEVC) # editEVCConfiguration evcupdateConfiguration update internal-vid
<1-4095>
Switch(ProvisionEVC) # editEVCConfiguration evcupdateConfiguration update learning
Switch(ProvisionEVC) # editEVCConfiguration evcupdateConfiguration update nni-ports
Switch(ProvisionEVC) # editEVCConfiguration evcupdateConfiguration update nni-vid
Switch(ProvisionEVC) # editEVCConfiguration evcupdateConfiguration update policer-id
Switch(ProvisionEVC) # editEVCConfiguration review
Switch(ProvisionEVC) # editEVCConfiguration commit
```

Enabling/Disabling/Modifying EVC Policer

Use this command to enable the EVC Policer.

enableEVCpolicer evc-policer-enable *policer-id*

```
Switch(ProvisionEVC) # enableEVCpolicer evc-policer-enable <1-1024>
Switch(ProvisionEVC) # enableEVCpolicer review
Switch(ProvisionEVC) # enableEVCpolicer commit
```

Use this command to disable the EVC Policer.

disableEVCpolicer evc-policer-enable *policer-id*

```
Switch(ProvisionEVC) # disableEVCpolicer evc-policer-enable <1-1024>
Switch(ProvisionEVC) # disableEVCpolicer review
Switch(ProvisionEVC) # disableEVCpolicer commit
```

Use this command to modify the EVC Policer.

modifyEVCpolicer evc-policer-enable *policer-id*

```
Switch(ProvisionEVC) # modifyEVCpolicer evc-policer-modify modify cbs <0-100000>
Switch(ProvisionEVC) # modifyEVCpolicer evc-policer-modify modify cir <0-10000000>
Switch(ProvisionEVC) # modifyEVCpolicer evc-policer-modify modify ebs <0-100000>
Switch(ProvisionEVC) # modifyEVCpolicer evc-policer-modify modify eir <0-10000000>
Switch(ProvisionEVC) # modifyEVCpolicer evc-policer-modify modify policer-mode colour-aware
Switch(ProvisionEVC) # modifyEVCpolicer evc-policer-modify modify policer-mode coupled
Switch(ProvisionEVC) # modifyEVCpolicer evc-policer-modify modify policer-type mef
Switch(ProvisionEVC) # modifyEVCpolicer evc-policer-modify modify policer-type single
Switch(ProvisionEVC) # modifyEVCpolicer evc-policer-modify modify rate-type data
Switch(ProvisionEVC) # modifyEVCpolicer evc-policer-modify modify rate-type line
Switch(ProvisionEVC) # modifyEVCpolicer evc-policer-modify modify state
Switch(ProvisionEVC) # modifyEVCpolicer evc-policer-modify policer-id <1-1022>
Switch(ProvisionEVC) # modifyEVCpolicer review
Switch(ProvisionEVC) # modifyEVCpolicer commit
```

Fetching EVC and ECE Configuration

Use the following commands to fetch the ECE configuration:

- **getECEBlankForm getECEForm**

```
Switch(ProvisionEVC) # getECEBlankForm getECEForm
Switch(ProvisionEVC) # getECEBlankForm review
Switch(ProvisionEVC) # getECEBlankForm commit
```

- **getECECounters ece-statistics-req *ece-statistics-request***

```
Switch(ProvisionEVC) # getECECounters ece-statistics-req <1-1024>
Switch(ProvisionEVC) # getECECounters review
Switch(ProvisionEVC) # getECECounters commit
```

- **getECEConfiguration getECEconfig** *ece-configuration*
Switch(ProvisionEVC) # getECEConfiguration getECEconfig <1-1024>
Switch(ProvisionEVC) # getECEConfiguration review
Switch(ProvisionEVC) # getECEConfiguration commit

Use the following commands to fetch the EVC configuration:

- **getEVBlankForm getEVForm**
Switch(ProvisionEVC) # getEVBlankForm getEVForm
Switch(ProvisionEVC) # getEVBlankForm review
Switch(ProvisionEVC) # getEVBlankForm commit
- **getEVC-Counters evc-statistics-req** *evc-statistics-request*
Switch(ProvisionEVC) # getEVCounters evc-statistics-req <1-1024>
Switch(ProvisionEVC) # getEVCounters review
Switch(ProvisionEVC) # getEVCounters commit
- **getEVConfiguration getEVconfig** *evc-configuration*
Switch(ProvisionEVC) # getEVConfiguration getEVconfig <1-1024>
Switch(ProvisionEVC) # getEVConfiguration review
Switch(ProvisionEVC) # getEVConfiguration commit