Configuring ISA Control Policies

The Intelligent Service Architecture (ISA) is a core set of Cisco IOS components that provide a structured framework in which edge access devices can deliver flexible and scalable services to subscribers. A Cisco device that is running a Cisco IOS image with ISA is called an Intelligent Service Gateway (ISG). ISA control policies are a means of defining the actions that your system will take in response to specified condition and events. A wide variety of system actions, conditions, and events can be combined using a consistent policy language, providing a flexible and precise way of configuring ISA. This module provides information about how to configure ISA control policies.

Module History

This module was first published on April 28, 2005, and last updated April 11, 2005.

Finding Feature Information in This Module

Your Cisco IOS software release may not support all features. To find information about feature support and configuration, use the "Feature Information for ISA Control Policies" section on page 35.

Contents

Prerequisites for Configuring ISA Control Policies

Authentication, authorization, and accounting (AAA) method lists must be configured prior to defining authentication and authorization actions.

CISCO SYSTEMS

Corporate Headquarters:
Cisco Systems, Inc., 170 West Tasman Drive, San Jose, CA 95134-1706 USA

Restrictions for Configuring ISA Control Policies

Control policies are activated for specific contexts, not directly on sessions. Control policies apply to all sessions hosted on the context.

Only one control policy map may be applied to a given context.

Control policies can only be defined through CLI.

Not all actions may be associated with all events.

A new control class may not be inserted between existing control classes once a control policy map has been defined.

Information About ISA Control Policies

Before you configure ISA control policies, you should understand the following concepts:

- Control Policies, page 18
- Uses of Control Policies, page 19

Control Policies

Control policies define the actions that the system will take in response to specified events and conditions. For example, a control policy can be configured to authenticate specific subscribers and then provide them with access to specific services.

A control policy is made of one or more control policy rules. A control policy rule is an association of a control class and one or more actions. The control class defines the conditions that must be met before the actions will be executed.

There are three steps involved in defining a control policy:

1. Create one or more control class maps.

   A control class map specifies the conditions that must be met for a policy to be activated, and, optionally, the event that causes the class to be evaluated. A control class map may contain multiple conditions, each of which will evaluate to either true or false. Match directives can be used to specify whether all, any, or none of the individual conditions must evaluate true in order for the class to evaluate true.

2. Create a control policy map.

   A control policy map contains one or more control policy rules. A control policy rule associates a control class map with one or more actions. Actions are numbered and executed sequentially.

3. Apply the control policy map.

   A control policy map is activated by applying it to a context. A control policy map can be applied to one or more of the following types of contexts:

- Global
- Interface
- Subinterface
- Virtual template
- Virtual circuit (VC) class

- Permanent virtual circuit (PVC)

  In general, control policy maps that are applied to more specific contexts take precedence over policy maps applied to more general contexts. In the list above, the context types are listed in order of precedence. For example, a control policy map that is applied to a PVC takes precedence over a control policy map that is applied to an interface.

Note Traffic policies are another type of policy used by ISA. Traffic policies define the handling of data packets and are configured in service policy maps or service profiles. For more information about traffic policies, see the "Configuring ISA Subscriber Services" module.

Uses of Control Policies

Use control policies to configure ISA to perform specific actions in response to specific events and conditions. For example, control policies could be used for the following purposes:

- To activate a default service when a subscriber session is first detected
- To sequence the gleaning of subscriber identity, where a control protocol exists on the access side
- To determine how the system responds to an idle timeout or to a subscriber who has run out of credit
- To enable transparent autologon, which enables authorization on the basis of an IP address or MAC address
- To configure the maximum amount of time a session can remain unauthenticated
- To send periodic session state information to other devices

How to Configure an ISA Control Policy

Perform the following tasks to configure an ISA control policy:

- Configuring a Control Class Map, page 19 (required)
- Configuring a Control Policy Map, page 24 (required)
- Applying the Control Policy Map, page 27 (required)
- Monitoring and Maintaining ISA Control Policies, page 31 (optional)

Configuring a Control Class Map

A control class map contains conditions that must be met for a control policy to be executed. A control class map can contain one or more conditions. Perform this task to configure a control class map.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **class-map type control** [**match-all match-any match-none**] *class-map-name*

4. **available** {**authen-status** | **authenticated-domain** | **authenticated-username** | **dnis** | **media** | **mlp-negotiated** | **nas-port** | **no-username** | **protocol** | **service-name** | **source-ip-address** | **timer** | **tunnel-name** | **unauthenticated-domain** | **unauthenticated-username**}
5. **greater-than** [**not**] **nas-port** {**adapter** *adapter-number* | **channel** *channel-number* | **ipaddr** *ip-address* | **port** *port-number* | **shelf** *shelf-number* | **slot** *slot-number* | **sub-interface** *sub-interface-number* | **type** *interface-type* | **vci** *vci-number* | **vlan** *vlan-id* | **vpi** *vpi-number*}
6. **greater-than-or-equal** [**not**] **nas-port** {**adapter** *adapter-number* | **channel** *channel-number* | **ipaddr** *ip-address* | **port** *port-number* | **shelf** *shelf-number* | **slot** *slot-number* | **sub-interface** *sub-interface-number* | **type** *interface-type* | **vci** *vci-number* | **vlan** *vlan-id* | **vpi** *vpi-number*}
7. **less-than** [**not**]**nas-port** {**adapter** *adapter-number* | **channel** *channel-number* | **ipaddr** *ip-address* | **port** *port-number* | **shelf** *shelf-number* | **slot** *slot-number* | **sub-interface** *sub-interface-number* | **type** *interface-type* | **vci** *vci-number* | **vlan** *vlan-id* | **vpi** *vpi-number*}
8. **less-than-or-equal** [**not**] **nas-port** {**adapter** *adapter-number* | **channel** *channel-number* | **ipaddr** *ip-address* | **port** *port-number* | **shelf** *shelf-number* | **slot** *slot-number* | **sub-interface** *sub-interface-number* | **type** *interface-type* | **vci** *vci-number* | **vlan** *vlan-id* | **vpi** *vpi-number*}
9. **match authen-status** {**authenticated** | **unauthenticated**}
10. **match authenticated-domain** {*domain-name* | **regexp** *regular-expression*}
11. **match authenticated-username** {*username* | **regexp** *regular-expression*}
12. **match dnis** {*dnis* | **regexp** *regular-expression*}
13. **match media** {**async** | **atm** | **ether** | **ip** | **isdn** | **mpls** | **serial**}
14. **match mlp-negotiated** {**no** | **yes**}
15. **match nas-port** {**adapter** *adapter-number* | **channel** *channel-number* | **ipaddr** *ip-address* | **port** *port-number* | **shelf** *shelf-number* | **slot** *slot-number* | **sub-interface** *sub-interface-number* | **type** {**async** | **atm** | **basic-rate** | **enm** | **ether** | **fxo** | **fxs** | **none** | **primary-rate** | **synch** | **vlan** | **vty**} | **vci** *vci-number* | **vlan** *vlan-id* | **vpi** *vpi-number*}
16. **match no-username** {**no** | **yes**}
17. **match protocol** {**atom** | **ip** | **pdsn** | **ppp** | **vpdn**}
18. **match service-name** {*service-name* | **regexp** *regular-expression*}
19. **match source-ip-address** *ip-address subnet-mask*
20. **match timer** {*timer-name* | **regexp** *regular-expression*}
21. **match tunnel-name** {*tunnel-name* | **regexp** *regular-expression*}
22. **match unauthenticated-domain** {*domain-name* | **regexp** *regular-expression*}
23. **match unauthenticated-username** {*username* | **regexp** *regular-expression*}

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**class-map type control** [**match-all match-any match-none**] *class-map-name* **Example:** `Router(config)# class-map type control match-all class1`	Creates or modifies a control class map, which defines the conditions under which the actions of a control policy map will be executed.
Step 4	**available** {**authen-status** | **authenticated-domain** | **authenticated-username** | **dnis** | **media** | **mlp-negotiated** | **nas-port** | **no-username** | **protocol** | **service-name** | **source-ip-address** | **timer** | **tunnel-name** | **unauthenticated-domain** | **unauthenticated-username**} **Example:** `Router(config-control-classmap)# available nas-port`	Creates a condition that will evaluate true if the specified subscriber identifier is locally available.
Step 5	**greater-than** [**not**] **nas-port** {**adapter** *adapter-number* | **channel** *channel-number* | **ipaddr** *ip-address* | **port** *port-number* | **shelf** *shelf-number* | **slot** *slot-number* | **sub-interface** *sub-interface-number* | **type** *interface-type* | **vci** *vci-number* | **vlan** *vlan-id* | **vpi** *vpi-number*} **Example:** `Router(config-control-classmap)# greater-than nas-port type atm vpi 200 vci 100`	Creates a condition that will evaluate true if the subscriber network access server (NAS) port identifier is greater than the specified value.
Step 6	**greater-than-or-equal** [**not**] **nas-port** {**adapter** *adapter-number* | **channel** *channel-number* | **ipaddr** *ip-address* | **port** *port-number* | **shelf** *shelf-number* | **slot** *slot-number* | **sub-interface** *sub-interface-number* | **type** *interface-type* | **vci** *vci-number* | **vlan** *vlan-id* | **vpi** *vpi-number*} **Example:** `Router(config-control-classmap)# greater-than-or-equal nas-port vlan 10`	Creates a condition that will evaluate true if the specified subscriber NAS port identifier is greater than or equal to the specified value.

	Command or Action	Purpose
Step 7	**less-than** [**not**] **nas-port** {**adapter** *adapter-number* | **channel** *channel-number* | **ipaddr** *ip-address* | **port** *port-number* | **shelf** *shelf-number* | **slot** *slot-number* | **sub-interface** *sub-interface-number* | **type** *interface-type* | **vci** *vci-number* | **vlan** *vlan-id* | **vpi** *vpi-number*} **Example:** `Router(config-control-classmap)# less-than nas-port type atm vpi 200 vci 105`	Creates a condition that will evaluate true if the specified subscriber NAS port identifier is less than the specified value.
Step 8	**less-than-or-equal** [**not**] **nas-port** {**adapter** *adapter-number* | **channel** *channel-number* | **ipaddr** *ip-address* | **port** *port-number* | **shelf** *shelf-number* | **slot** *slot-number* | **sub-interface** *sub-interface-number* | **type** *interface-type* | **vci** *vci-number* | **vlan** *vlan-id* | **vpi** *vpi-number*} **Example:** `Router(config-control-classmap)# less-than-or-equal nas-port ipaddr 10.10.10.10`	Creates a condition that will evaluate true if the specified subscriber NAS port identifier is less than or equal to the specified value.
Step 9	**match authen-status** {**authenticated** | **unauthenticated**} **Example:** `Router(config-control-classmap)# match authen-status authenticated`	Creates a condition that will evaluate true if a subscriber's authentication status matches the specified authentication status.
Step 10	**match authenticated-domain** {*domain-name* | **regexp** *regular-expression*} **Example:** `Router(config-control-classmap)# match authenticated-domain cisco.com`	Creates a condition that will evaluate true if a subscriber's authenticated domain matches the specified domain.
Step 11	**match authenticated-username** {*username* | **regexp** *regular-expression*} **Example:** `Router(config-control-classmap)# match authenticated-username regexp "admin@.*com"`	Creates a condition that will evaluate true if a subscriber's authenticated username matches the specified username.
Step 12	**match dnis** {*dnis* | **regexp** *regular-expression*} **Example:** `Router(config-control-classmap)# match dnis reg-exp 5551212`	Creates a condition that will evaluate true if a subscriber's Dialed Number Identification Service number (DNIS number, also referred to as *called-party number*) matches the specified DNIS number.
Step 13	**match media** {**async** | **atm** | **ether** | **ip** | **isdn** | **mpls** | **serial**} **Example:** `Router(config-control-classmap)# match media atm`	Creates a condition that will evaluate true if a subscriber's access media type matches the specified media type.

	Command or Action	Purpose
Step 14	**match mlp-negotiated** {**no** | **yes**} **Example:** `Router(config-control-classmap)# match mlp-negotiated yes`	Creates a condition that will evaluate true depending on whether or not the subscriber's session was established using multilink PPP negotiation.
Step 15	**match nas-port** {**adapter** *adapter-number* | **channel** *channel-number* | **ipaddr** *ip-address* | **port** *port-number* | **shelf** *shelf-number* | **slot** *slot-number* | **sub-interface** *sub-interface-number* | **type** {**async** | **atm** | **basic-rate** | **enm** | **ether** | **fxo** | **fxs** | **none** | **primary-rate** | **synch** | **vlan** | **vty**} | **vci** *vci-number* | **vlan** *vlan-id* | **vpi** *vpi-number*} **Example:** `Router(config-control-classmap)# match nas-port type ether slot 3`	Creates a condition that will evaluate true if a subscriber's NAS port identifier matches the specified value.
Step 16	**match no-username** {**no** | **yes**} **Example:** `Router(config-control-classmap)# match no-username yes`	Creates a condition that will evaluate true depending on whether or not a subscriber's username is available.
Step 17	**match protocol** {**atom** | **ip** | **pdsn** | **ppp** | **vpdn**} **Example:** `Router(config-control-classmap)# match protocol ip`	Creates a condition that will evaluate true if a subscriber's access protocol type matches the specified protocol type.
Step 18	**match service-name** {service-name | **regexp** *regular-expression*} **Example:** `Router(config-control-classmap)# match service-name gold`	Creates a condition that will evaluate true if the service name associated with a subscriber matches the specified service name.
Step 19	**match source-ip-address** *ip-address subnet-mask* **Example:** `Router(config-control-classmap)# match source-ip-address 10.10.10.10 255.255.255.255`	Creates a condition that will evaluate true if a subscriber's source IP address matches the specified IP address.
Step 20	**match timer** {*timer-name* | **regexp** *regular-expression*} **Example:** `Router(config-control-classmap)# match timer TIMERA`	Creates a condition that will evaluate true upon expiry of a specified policy timer.

	Command or Action	Purpose
Step 21	**match tunnel-name** {*tunnel-name* | **regexp** *regular-expression*} **Example:** `Router(config-control-classmap)# match tunnel-name regexp L.*`	Creates a condition that will evaluate true if a subscriber's virtual private dial-up network (VPDN) tunnel name matches the specified tunnel name.
Step 22	**match unauthenticated-domain** {*domain-name* | **regexp** *regular-expression*} **Example:** `Router(config-control-classmap)# match unauthenticated-domain abc.com`	Creates a condition that will evaluate true if a subscriber's unauthenticated domain name matches the specified domain name.
Step 23	**match unauthenticated-username** {username | **regexp** *regular-expression*} **Example:** `Router(config-control-classmap)# match unauthenticated-username regexp .*blue.*`	Creates a condition that will evaluate true if a subscriber's unauthenticated username matches the specified username.

Configuring a Control Policy Map

A control policy map contains one or more control policy rules, which associate a control class with one or more actions. Perform this task to configure a control policy map.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **policy-map type control** *policy-map-name*
4. **class type control** {*control-class-name* | **always**} [**event** {**account-logon** | **credit-exhausted** | **quota-depleted** | **service-start** | **service-stop** | **session-default-service** | **session-service-found** | **session-start** | **timed-policy-expiry**}
5. *action-number* **authenticate aaa list** *list-name*
6. *action-number* **authorize** [**aaa list** *list-name*] [**password** *password*] [**upon network-service-found** {**continue** | **stop**}] **identifier** {**authenticated-domain** | **authenticated-username** | **dnis** | **mac-address** | **nas-port** | **source-ip-address** | **tunnel-name** | **unauthenticated-domain** | **unauthenticated-username**}
7. *action-number* **collect** [**aaa list** *list-name*] **identifier** {**authen-status** | **authenticated-domain** | **authenticated-username** | **dnis** | **media** | **mlp-negotiated** | **nas-port** | **no-username** | **protocol** | **service-name** | **source-ip-address** | **timer** | **tunnel-name** | **unauthenticated-domain** | **unauthenticated-username**}
8. *action-number* **if upon network-service-found** {**continue** | **stop**}
9. *action-number* **service** [**disconnect** | **local** | **vpdn**]
10. *action-number* **service-policy type control** *policy-map-name*

11. *action-number* **service-policy type service** [**unapply**] [**aaa list** *list-name* service] {**name** *service-name* | **identifier** {**authenticated-domain** | **authenticated-username** | **dnis** | **nas-port** | **tunnel-name** | **unauthenticated-domain** | **unauthenticated-username**}}
12. *action-number* **set-timer** *name-of-timer minutes*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:s** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**policy-map type control** *policy-map-name* **Example:** `Router(config)# policy-map type control MY-POLICY`	Creates or modifies a control policy map, which is used to define a control policy.
Step 4	**class type control** {*control-class-name* | **always**} [**event** {**account-logon** | **credit-exhausted** | **quota-depleted** | **service-start** | **service-stop** | **session-default-service** | **session-service-found** | **session-start** | **timed-policy-expiry**} **Example:** `Router(config-control-policymap)# class type control always event session-start`	Specifies a control class for which actions may be configured. • A policy rule for which the control class is **always** will always be treated as the lowest priority rule within the control policy map.
Step 5	*action-number* **authenticate aaa list** *list-name* **Example:** `Router(config-control-policymap-class-control)# 1 authenticate aaa list LIST1`	Initiates an authentication request.
Step 6	*action-number* **authorize** [**aaa list** *list-name*] [**password** *password*] [**upon network-service-found** {**continue** | **stop**}] **identifier** {**authenticated-domain** | **authenticated-username** | **dnis** | **mac-address** | **nas-port** | **source-ip-address** | **tunnel-name** | **unauthenticated-domain** | **unauthenticated-username**} **Example:** `Router(config-control-policymap-class-control)# 1 authorize`	Initiates a request for authorization on the basis of the specified identifier.

	Command or Action	Purpose
Step 7	*action-number* **collect** [**aaa list** *list-name*] **identifier** {**authen-status** | **authenticated-domain** | **authenticated-username** | **dnis** | **media** | **mlp-negotiated** | **nas-port** | **no-username** | **protocol** | **service-name** | **source-ip-address** | **timer** | **tunnel-name** | **unauthenticated-domain** | **unauthenticated-username**} **Example:** `Router(config-control-policymap-class-control)# 1 collect identifier authen-status`	Collects the specified subscriber identifier from the access protocol.
Step 8	*action-number* **if upon network-service-found** {**continue** | **stop**} **Example:** `Router(config-control-policymap-class-control)# 2 if upon network-service-found stop`	Specifies whether the system should continue processing policy rules once the subscriber's network service has been identified.
Step 9	*action-number* **service** [**disconnect** | **local** | **vpdn**] **Example:** `Router(config-control-policymap-class-control)# 3 service disconnect`	Specifies a network service type for PPP sessions.
Step 10	*action-number* **service-policy type control** *policy-map-name* **Example:** `Router(config-control-policymap-class-control)# service-policy type control domain_based_access`	Nests the specified control policy map within a parent control policy map.
Step 11	*action-number* **service-policy type service** [**unapply**] [**aaa list** *list-name*] {**name** *service-name* | **identifier** {**authenticated-domain** | **authenticated-username** | **dnis** | **nas-port** | **tunnel-name** | **unauthenticated-domain** | **unauthenticated-username**}} **Example:** `Router(config-control-policymap-class-control)# 1 service-policy type service aaa list LISTA name REDIRECT`	Activates an ISA service. • Specifying an identifier instead of a service name will activate a service that has the same name as the specified identifier.
Step 12	*action-number* **set-timer** *name-of-timer minutes* **Example:** `Router(config-control-policymap-class-control)# 1 set-timer TIMERA 5`	Starts a named policy timer. • Expiration of the timer generates the event timed-policy-expiry.

Applying the Control Policy Map

A control policy map must be activated by applying it to a context. Perform one or more of the following tasks to apply a control policy to a context:

Applying a Control Policy Map Globally on the Router

Perform this task to apply a control policy globally.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **service-policy type control** *policy-map-name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**service-policy type control** *policy-map-name* **Example:** `Router(config)# service-policy type control policy1`	Applies a control policy.

Applying a Control Policy Map to an Interface or Subinterface

Perform this task to apply an ISA control policy to an interface or subinterface.

SUMMARY STEPS

1. **enable**
2. **configure terminal**

3. **interface** *type number*
4. **service-policy type control** *policy-map-name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**interface** *type number*[**.***subinterface-number*] **Example:** `Router(config)# interface gigabitethernet0/1`	Specifies an interface and enters interface configuration mode.
Step 4	**service-policy type control** *policy-map-name* **Example:** `Router(config-if)# service-policy type control policy1`	Applies a control policy.

Applying a Control Policy Map to a Virtual Template

Perform this task to apply an ISA control policy map to a virtual template.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface virtual-template** *number*
4. **service-policy type control** *policy-map-name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 3	**interface virtual-template** *number* **Example:** `Router(config)# interface virtual-template0`	Creates a virtual template interface and enters interface configuration mode.
Step 4	**service-policy type control** *policy-map-name* **Example:** `Router(config-if)# service-policy type control policy1`	Applies a control policy.

Applying a Control Policy Map to an ATM VC Class

A VC class is a set of preconfigured VC parameters that are configured and applied to a particular VC or ATM interface. Perform this task to apply an ISA control policy map to an ATM VC class.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface virtual-template** *number*
4. **service-policy type control** *policy-map-name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**vc-class atm** *vc-class-name* **Example:** `Router(config)# vc-class atm class1`	Creates an ATM VC class and enters ATM VC class configuration mode. • A VC class can be applied to an ATM interface, subinterface, or VC.
Step 4	**service-policy type control** *policy-map-name* **Example:** `Router(config-vc-class)# service-policy type control policy1`	Applies a control policy.

Applying a Control Policy Map to an ATM PVC

Perform this task to apply an ISA control policy to an ATM PVC.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface atm** *interface-number*[**.***subinterface-number* {**mpls** | **multipoint** | **point-to-point**}]
4. **pvc** *vpi*/*vci*
5. **service-policy type control** *policy-map-name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**interface atm** *interface-number*[**.***subinterface-number* {**mpls** | **multipoint** | **point-to-point**}] **Example:** `Router(config)# interface atm 5/0.1 multipoint`	Specifies an ATM interface or subinterface and enters interface configuration mode.
Step 4	**pvc** *vpi*/*vci* **Example:** `Router(config-if)# pvc 2/101`	Creates an ATM PVC and enters ATM virtual circuit configuration mode.
Step 5	**service-policy type control** *policy-map-name* **Example:** `Router(config-if-atm-vc)# service-policy type control policy1`	Applies a control policy.

Monitoring and Maintaining ISA Control Policies

Optionally, you can perform this task to monitor and maintain ISA control policy operation. Steps can be performed in any order.

SUMMARY STEPS

1. **enable**
2. **show class-map type control**
3. **show policy-map type control**
4. **clear class-map type control**
5. **clear policy-map type control**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**show class-map type control** **Example:** `Router# show class-map type control`	Displays information about ISA control class maps. • The display includes statistics on the number of times a particular class has been evaluated and what the results were.
Step 3	**show policy-map type control** **Example:** `Router# show policy-map type control`	Displays information about ISA control policy maps. • The display includes statistics on the number of times each policy rule within the policy map has been executed.
Step 4	**clear class-map type control** **Example:** `Router# clear class-map type control`	Clears the control class map counters.
Step 5	**clear policy-map type control** **Example:** `Router# clear policy-map type control`	Clears the control policy map counters.

Configuration Examples for ISA Control Policies

This section contains the following examples of ISA control policies:

Control Policy for Layer 2 Access and Service Provisioning: Example

The following example shows how to configure a control policy that produces the following results:

- VPDN forwarding is applied to anyone dialing in from “xyz.com”.
- Access to locally terminated Layer 3 network resources is provided to anyone dialing in from “def.com”.
- Anyone else is barred.

```
! Configure the control class maps.
class-map type control match-all MY-FORWARDED-USERS
 match unauthenticated-domain "xyz.com"
!
class-map type control match-all MY-LOCAL-USERS
 match unauthenticated-domain "def.com"
!
! Configure the control policy map.
policy-map type control MY-POLICY
 class type control MY-FORWARDED-USERS event session-start
  1 service-policy type service identifier nas-port
  2 service local
!
 class type control MY-LOCAL-USERS event session-start
  1 service local
!
 class type control always event session-start
  2 service disconnect
!
! Apply the control policy globally.
interface Dialer1
 service-policy type control MY-POLICY
```

Control Policy Restricting Access on the Basis of Interface and Access Media: Example

This example shows how to configure a control policy to allow access only to users who enter the router from a particular interface and access type. In this case, only PPPoE users will be allowed; everyone else gets barred.

The class map, “MATCHING-USERS”, will evaluate true only if all of the lines within it also evaluate true; however, within “MATCHING-USERS” is a nested class map, "NOT-ATM". This nested class map represents a subcondition that must also evaluate to true. Note that the class map “NOT-ATM” specifies “match-none”. This means that "NOT-ATM" evaluates to true only if each and every condition line within it evaluates to false.

The third condition specifies matching on the NAS port associated with this subscriber. Specifically, only subscribers that arrive on an Ethernet interface and on slot 3 will evaluate to true.

```
! Configure the control class maps.
class-map type control match-all MATCHING-USERS
 class type control NOT-ATM
 match media ether
 match nas-port type ether slot 3
!
class-map type control match-none NOT-ATM
 match media atm
!
```

If the conditions in the class map “MATCHING-USERS” evaluate to true, the first action to be executed is to authenticate the user. If authentication is successful, the service named “gold” will be downloaded and applied. Finally, a Layer 3 service is provided.

If “MATCHING-USERS” is not evaluated as true, the “always” class will apply, which results in barring anyone who does not match “MATCHING-USERS”.

```
! Configure the control policy map.
policy-map type control my-pppoe-rule
 class type control MATCHING-USERS event session-start
  1 authenticate aaa list XYZ
  2 service-policy type service gold
  3 service local
!
 class type control always
  1 service disconnect
!
! Apply the control policy to an interface.
interface ethernet3/0
 service-policy type control my-pppoe-rule
```

Finally, the policy is associated with an interface.

Control Policy for ISA Prepaid Billing Support: Example

The following example shows a control policy configured to redirect subscriber packets to the server group “redirect-sg” when the credit-exhausted event occurs:

```
service-policy type control RULEA
!
policy-map type control RULEA
 class type control always event credit-exhausted
  1 service-policy type service redirectprofile
!
policy-map type service redirectprofile
 class type traffic CLASS-ALL
  redirect to group redirect-sg

policy-map type service mp3
 class type traffic CLASS-ACL-101
  authentication method-list cp-mlist
  accounting method-list cp-mlist
  prepaid conf-prepaid

subscriber feature prepaid conf-prepaid
 threshold time 20
 threshold volume 0
 method-list accounting ap-mlist
 method-list authorization default
 password cisco
```

Control Policies for ISA Transparent Autologon: Example

In the following example, if the client is from the 1.1.1.0 subnet, ISA transparent autologon is applied and an authorization request is sent to the list TAL_LIST with the subscriber’s source IP address as the username. If the authorization request is successful, any automaticactivation services specified in the returned user profile are activated for the session and the execution of rules within the control-policy

stops. If the authorization is not successful, the rule execution proceeds, and the subscriber is redirected to the policy server to log in. If the subscriber does not log in within five minutes, the session is disconnected.

```
interface Ethernet0/0
 service-policy type control RULEA

aaa authentication login TAL_LIST group radius
aaa authentication login LOCAL local
access-list 100 permit ip any any

class-map type traffic match-any all-traffic
 match access-group input 100
 match access-group output 100

policy-map type service redirectprofile
 class type traffic all-traffic
  redirect to ip 10.0.0.148 port 8080

class-map type control match-all CONDA
 match source-ip-address 1.1.1.0 255.255.255.0
!
class-map type control match-all CONDF
 match timer TIMERB
 match authen-status unauthenticated

policy-map type control RULEA
 class type control CONDA event session-start
  1 authorize aaa list TAL_LIST password cisco identifier source-ip-address
  2 apply aaa list LOCAL service redirectprofile
  3 set-timer TIMERB 5 minutes
 class type control CONDF event timed-policy-expiry
  1 service disconnect
```

Additional References

The following sections provide references related to ISA control policies.

Related Documents

Related Topic	Document Title
ISA commands	*Cisco IOS Intelligent Service Architecture Command Reference*

Technical Assistance

Description	Link
Technical Assistance Center (TAC) home page, containing 30,000 pages of searchable technical content, including links to products, technologies, solutions, technical tips, and tools. Registered Cisco.com users can log in from this page to access even more content.	http://www.cisco.com/public/support/tac/home.shtml

Feature Information for ISA Control Policies

Table 3 lists the features in this module and provides links to specific configuration information. Only features that were introduced or modified in Cisco IOS Release 12.2(27)SBA or later appear in the table.

Not all commands may be available in your Cisco IOS software release. For details on when support for specific commands was introduced, see the command reference documents.

If you are looking for information on a feature in this technology that is not documented here, see the "Intelligent Service Architecture Features Roadmap"

Cisco IOS software images are specific to a Cisco IOS software release, a feature set, and a platform. Use Cisco Feature Navigator to find information about platform support and Cisco IOS software image support. Access Cisco Feature Navigator at http://www.cisco.com/go/fn. You must have an account on Cisco.com. If you do not have an account or have forgotten your username or password, click **Cancel** at the login dialog box and follow the instructions that appear.

Note Table 3 lists only the Cisco IOS software release that introduced support for a given feature in a given Cisco IOS software release train. Unless noted otherwise, subsequent releases of that Cisco IOS software release train also support that feature.

Table 3 Feature Information for ISA Control Policies

Feature Name	Releases	Feature Configuration Information
ISA:Policy Control: Policy: Domain Based (Auto-domain, Proxy)	12.2(27)SBA	ISA control policies manage the primary services and rules used to enforce particular contracts. These policies include programmable interfaces to dynamic triggers and conditional logic to be applied to flows within a session, or other characteristics of a session upon meeting the policy criteria. Polices can be configured to interpret the domain as a request to activate the service associated with that domain name, allowing users automatically receive services in accordance with the domain to which they are attempting to connect. The following sections provide more information about this feature: • Information About ISA Control Policies, page 18 • How to Configure an ISA Control Policy, page 19
ISA: Policy Control: Policy: Triggers (Time, Volume, Duration)	12.2(27)SBA	ISA control policies can be configured with time-based, volume-based,and duration-based policy triggers. Time-based triggers use an internal clock, allowing policies to be applied at specific times. Volume-based triggers are based on packet count; when the packet count reaches a specified value, the specified policy is applied.Duration-based triggers are based on an internal timer. Upon expiration of the timer, the specified policy is applied. The following sections provide more information about this feature: • Information About ISA Control Policies, page 18 • How to Configure an ISA Control Policy, page 19

Table 3 Feature Information for ISA Control Policies

Feature Name	Releases	Feature Configuration Information
ISA:Policy Control: Multidimensional Identity per Session	12.2(27)SBA	ISA control policies provide a flexible way to collect pieces of subscriber identity during session establishment. Control policies also allow session policy to be applied iteratively as more elements of identity become available to the system. The following sections provide more information about this feature: • Information About ISA Control Policies, page 18 • How to Configure an ISA Control Policy, page 19
ISA: Policy Control: Cisco Policy Language	12.2(27)SBA	ISA control policies are a structured replacement for feature-specific configuration commands and allow configurable functionality to be expressed in terms of an event, a condition, and an action. Control policies provide an intuitive and extensible framework, with a consistent set of CLI commands, for specifying system behavior. The ISA policy language is a aligned with the Cisco Common Classification Policy Language (C3PL). The following sections provide more information about this feature: • Information About ISA Control Policies, page 18 • How to Configure an ISA Control Policy, page 19

This module first published April 28, 2005. Last updated April 28, 2005.