Chapter 20

Overview of Providing Admission Control with SRC-ACP

This chapter describes the SRC Admission Control Plug-In (SRC-ACP) application, which provides admission control. Topics include:

Overview of SRC-ACP

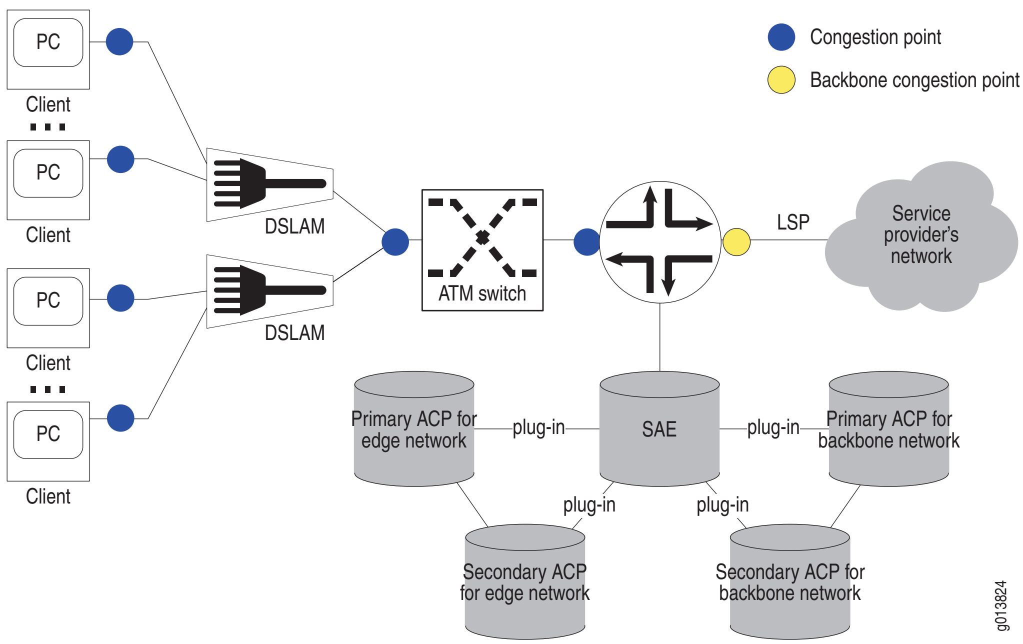
SRC-ACP is an external plug-in for the SAE. SRC-ACP authorizes and tracks subscribers' use of network resources associated with services that the SRC software manages. Service providers can implement SRC-ACP configurations for both residential and enterprise subscribers. Consequently, both JUNOSe routers and JUNOS routing platforms are compatible with SRC-ACP. References to virtual routers (VRs) in this documentation refer to an actual VR on a JUNOSe router or the single VR called default that the SRC software associates with each JUNOS routing platform (see *Part 2*, *Using Juniper Networks Routers in the SRC Network*).

SRC-ACP operates in two separate regions of the SRC network: the *edge* network and the *backbone* network. The edge network is the layer 2 access network through which subscribers connect to the router. The backbone network is the region between the router and the service provider's network.

Congestion often occurs in the network at points where connections are aggregated. SRC-ACP monitors congestion points at interfaces between devices in the edge network. In the backbone network, SRC-ACP monitors one congestion point, a point-to-point label-switched path (LSP) between the router and the service provider's network.

Figure 45 shows a typical network topology.

Figure 45: Position of SRC-ACP in Network



In the edge network, SRC-ACP performs the following procedures to determine whether there are sufficient resources to activate a service:

- Tracks active services for each subscriber and the guaranteed traffic rate (bandwidth) at the congestion points associated with a subscriber.
- Tracks the rate of traffic between the subscriber and the network (upstream bandwidth) and the rate of traffic between the network and subscriber (downstream bandwidth).
- Monitors new requests for activation of services.
- Compares the resources required for the new services with the resources available for the subscriber and the congestion points.
- Activates the service if sufficient resources are available, and prevents activation of the service if sufficient resources are not available.

In the backbone network, SRC-ACP performs the following procedures to determine whether there are sufficient resources to activate a service:

- Tracks the guaranteed traffic rate for a service at the congestion point.
- Tracks the actual traffic rate for the service at the congestion point.

- Monitors new requests for activation of services.
- Compares the resources required for the new services with the resources available at the congestion point.
- Activates the service if sufficient resources are available, and prevents activation of the service if sufficient resources are not available.

Typically, network administrators use their own network management applications and external applications to provide data for SRC-ACP. SRC-ACP first obtains updates from external applications through its remote CORBA interface, and then obtains updates from the directory by means of LDAP. For information about developing external applications that send data to SRC-ACP, see *API for ACP* on page 281. SRC-ACP does not interact directly with the network to assess the capacity of a congestion point or actual use of network resources.

In the backbone network, SRC-ACP can also execute applications defined in the action congestion point. For information about defining applications in congestion points, see *Configuring Action Congestion Points* on page 311. Some applications require real-time congestion point status. If SRC-ACP must provide real-time congestion point status to the application, state synchronization must be enabled to handle interface tracking events so that the congestion points are updated properly.

Deriving Congestion Points Automatically

SRC-ACP can derive some congestion points automatically. Depending on your network configuration and requirements, however, you may need to enter congestion points manually. This section describes the conditions and requirements for SRC-ACP to derive congestion points automatically.

Deriving Edge Congestion Points

For SRC-ACP to derive edge congestion points, subscribers must always connect through the same interface on the router. In addition, SRC-ACP requires one of the following conditions to derive edge congestion points if you are not using a congestion point profile:

- Access to subscriber profiles that define bandwidth values and a list of the distinguished names (DNs) of congestion points between the subscriber and the router.
- An ATM access network between the subscriber and the router for which all the traffic coming from one DSLAM travels on a single virtual path. In this case, SRC-ACP automatically derives three congestion points through the network access server (NAS) port ID. (For information about the NAS port ID, see *SRC-PE Subscribers and Subscriptions Guide, Chapter 11, Configuring Accounting and Authentication Plug-Ins with the SRC CLI*.) Table 17 on page 276 shows the edge congestion points and the corresponding locations in the directory.

For more information about automatically deriving congestion points from a configured congestion point profile, see *Deriving Congestion Points from a Profile* on page 276.

SRC-ACP does not use bandwidth statistics from subscriber profiles when it derives congestion points, because the congestion points already use that data.

Table 17: Congestion Points Derived Through NAS Port ID

Congestion Points	Location of Object in Directory
Physical interface on router	interfaceName=ATM<slot>/<port>, *orderedCimKeys=<routerName>, o=AdmissionControl, o=umc* <slot>—Number of port on router <port>—Number of port on router <routerName>—Hostname configured for router
ATM virtual path	interfaceName=ATM<slot>/<port>:<vpi> *orderedCimKeys=<routerName>, o=AdmissionControl, o=umc* <vpi>—Number of virtual path on router
ATM virtual connection	interfaceName=ATM<slot>/<port>:<vpi>.<vci> *orderedCimKeys=<routerName>, o=AdmissionControl, o=umc* <vci>—Number of virtual connection on router

Deriving Congestion Points from a Profile

If you configure a congestion point profile, SRC-ACP can automatically derive congestion points for cases in which:

- There is no subscriber profile.
- The congestion points can be derived from information provided by the access interface on B-RAS. For example, in an ATM or VLAN connection, you can derive congestion points representing physical interfaces and intermediate switches based on the NAS port ID reported by B-RAS.

When SRC-ACP receives notification to start subscriber tracking and to load congestion points for a subscriber, it runs a congestion point classification and accesses the configured congestion point profile. Congestion point classification uses the same classification engine as subscriber and interface classification in the SAE.

For this feature to operate correctly, you create a congestion point profile that automatically performs congestion point classification. For more information about this topic, see *Defining a Congestion Point Profile* on page 321 (using the SRC CLI) or *Configuring SRC-ACP* on page 358 (using SRC configuration applications on Solaris platforms).

Deriving Backbone Congestion Points

SRC-ACP can automatically derive backbone congestion points if you specify the setting <-vrName->/<-serviceName-> for the congestion point associated with a service. When the SRC-ACP starts operating, it will substitute the name of the VR and the service name from the activation request.

For example, you can specify the setting <-vrName->/<-serviceName-> for the congestion point associated with a service called News. Then, when a subscriber who connects to the network through a VR called boston requests activation of this service, SRC-ACP receives the request and proceeds as follows:

1. SRC-ACP reads the congestion point specification, <-vrName->/<-serviceName->, from the congestion point defined for the service News.

2. SRC-ACP substitutes the actual information, boston/News, in the variables.

3. SRC-ACP uses this information to generate the DN *cn=News, cn=boston, o=CongestionPoints, o=umc*.

4. SRC-ACP uses this DN to obtain from the directory the network interface, which defines the location of the congestion point, for this DN.

For this feature to operate correctly, you must configure the DN for each combination of VR and service to point to an actual network interface. For more information about this topic, see *Configuring SRC-ACP to Manage the Backbone Network* on page 310.

Allocating Bandwidth to Applications Not Controlled by SRC-ACP

If you control the bandwidth of some applications by means of SRC-ACP, you can accommodate the applications that are not controlled by SRC-ACP by assigning *background* bandwidths for the edge congestion points. The background bandwidth is the total bandwidth allocated to the applications for which bandwidth is not controlled by SRC-ACP.

Because the total background bandwidth is unlikely to be used at a particular time, you can also specify a tuning factor that provides an estimation of the fraction of the background bandwidth that will be used. You can configure multiple values for the background bandwidth with corresponding tuning factors.

Use of Multiple SRC-ACPs

An SRC-ACP can support one or more SAEs. Similarly, multiple SRC-ACPs can support one SAE; for example, if an SAE is managing multiple VRs, you may have an SRC-ACP for each VR. However, only one SRC-ACP can manage a particular congestion point.

Interactions Between SRC-ACP and Other Components

This section describes how SRC-ACP interacts with other components to track data.

1. (Edge and dual mode only) When a subscriber connects to the router, SRC-ACP loads the subscriber profile from the directory. If the subscriber profile contains provisioned and actual traffic rates for the subscriber's interface and the set of congestion points between the subscriber and the router, SRC-ACP caches the information while the subscriber is connected to the router. SRC-ACP automatically updates the subscriber's actual upstream and downstream rates if the subscriber profile changes in the directory.

2. (Backbone mode only) When a subscriber activates a service, SRC-ACP loads the network interfaces defined in the service and caches the information.

3. (Optional) SRC-ACP obtains through its remote CORBA interface data from external applications about subscribers and congestion points. If a congestion point is unavailable, SRC-ACP denies service activation requests on the associated network interface until the interface is available again.

4. If SRC-ACP does not receive data from an external application, SRC-ACP loads data about congestion points from the directory. For each congestion point the following data is retrieved:

   - Provisioned bandwidth

   - Background bandwidths (if used for edge congestion points)

   SRC-ACP caches this information and automatically updates the cache when the information changes in the directory.

5. (Edge and dual modes) If SRC-ACP does not receive data from an external application, SRC-ACP loads a subscriber's provisioned or actual bandwidth from the subscriber profile. If the actual bandwidth is available, SRC-ACP ignores the provisioned bandwidth.

   SRC-ACP caches this information and automatically updates the cache when the information changes in the directory.

6. (Backbone and dual modes only) Using a hosted plug-in, the SAE monitors the states of router interfaces associated with backbone congestion points. The SAE sends relevant data to SRC-ACP through the SRC-ACP's remote interface.

7. When the subscriber requests activation of a service subscription (either through the SAE core API or automatically for activate-on-login services), the SAE notifies SRC-ACP to authorize and track the service usage.

   a. The SAE sends the requested bandwidth to SRC-ACP.

   b. SRC-ACP authorizes or denies service activation.

      If SRC-ACP authorizes the service activation, the SAE activates the service and sends a tracking event to SRC-ACP. SRC-ACP updates the current bandwidth for all congestion points with the requested bandwidth.

If SRC-ACP authorizes the service activation with state synchronization enabled, SRC-ACP reserves the requested bandwidth on all congestion points until the reservation expires. You can specify the reservation timeout value when configuring SRC-ACP operation.

- For each congestion point, SRC-ACP verifies whether:

  (current bw + reserved bw + requested bw) >
  [provisioned bw - (background bw x tuning factor)]

  If the desired bandwidth exceeds the allocated bandwidth, SRC-ACP denies service activation.

- When SRC-ACP receives a service start tracking event, the requested bandwidth is committed. That is, for each congestion point, the requested bandwidth reservation is removed and the requested bandwidth is added to the current bandwidth.

- When the bandwidth reservation expires, the reserved bandwidth is released.

If SRC-ACP does not authorize the service activation, the SAE delivers a message detailing the reason to the originator of the activation request.

SRC-ACP distinguishes between bandwidth exceeded on the subscriber interface (first congestion point) and bandwidth exceeded on a network interface by sending two different messages back to the SAE. In the first case, the subscriber may resolve the bandwidth problem by deactivating another service.

8. When a service is deactivated (either through the SAE core API or because a session times out), SRC-ACP updates the current bandwidth for all congestion points by removing the original requested bandwidth.

9. SRC-ACP stores all information about subscribers, services, and congestion points in a set of files.

   SRC-ACP continually adds data to these files, but does not delete old data. Consequently, the sizes of the files continue to increase. SRC-ACP does, however, reorganize the files when the sum of their sizes increments by a specified value. Reorganizing the files reduces their sizes. You can also reorganize the files by using the SRC CLI (see *Reorganizing the File That Contains ACP Data* on page 325.)

Redundancy

You can configure SRC-ACP redundancy for a region of the network by installing SRC-ACP on two different hosts, installing a naming service application on the SAE, and connecting both SRC-ACP hosts to the SAE (see Figure 45 on page 274). One SRC-ACP acts as the primary application, and the other as the secondary application.

NOTE: Both SRC-ACPs in a redundant pair must operate in the same mode. You cannot configure an SRC-ACP in edge mode and an SRC-ACP in backbone mode as a redundant pair.

The primary and secondary SRC-ACPs communicate with each other through a CORBA interface. When you start each SRC-ACP (see *Starting SRC-ACP* on page 325), it will register its redundancy CORBA interface with the naming service application, and import the interface for the other SRC-ACP from the naming service application.

Each SRC-ACP continuously monitors the other's availability. The primary SRC-ACP receives data from the SAE and sends any changes to the secondary SRC-ACP. If the secondary SRC-ACP is unavailable, the primary SRC-ACP caches the data to send when the secondary SRC-ACP becomes available.

If the primary SRC-ACP becomes unavailable, the secondary SRC-ACP immediately notifies the naming service application and assumes the primary role. If the former primary SRC-ACP recovers very quickly, it will again assume the primary role. However, if the former primary SRC-ACP recovers more slowly, it will assume the primary role only if the former secondary SRC-ACP becomes unavailable.

Fault Recovery

If the SAE cannot reach SRC-ACP, the SAE will deny all service activation requests. As soon as it reaches SRC-ACP, the SAE again sends authorization requests to SRC-ACP.

SRC-ACP keeps the state of the congestion points in persistent storage, and if SRC-ACP becomes unavailable, the service authorization can continue in the correct state. Because service activation requests are automatically denied when the SAE cannot reach SRC-ACP, SRC-ACP does not miss any active service sessions. The SAE will resend all service deactivation requests after SRC-ACP is reachable again.

SRC-ACP monitors SAE synchronization events for information about VR availability and SAE availability. If a VR reboots or an SAE becomes unavailable, SRC-ACP updates the states of congestion points associated with those devices accordingly.

If the SAE becomes unavailable, the router will automatically reestablish connection to either the redundant SAE or, if a redundant SAE is not available, to the original SAE when it again becomes available. The new SAE notifies SRC-ACP that the original SAE failed and specifies which subscriber and service sessions were logged during this time. SRC-ACP uses this information to update its state.

State Synchronization

You can configure SRC-ACP to synchronize states with the SAE.

If state synchronization is enabled, the current state can be transferred when SRC-ACP has started up or lost its state. SRC-ACP does not have to keep a local and persistent copy of the state. However, SRC-ACP requires additional bandwidth to transfer state information that can affect performance.

Both SRC-ACP redundancy and state synchronization can be enabled at the same time. In this situation, the primary and secondary SRC-ACPs are set up as a community and will communicate with each other to determine the primary SRC-ACP. The primary SRC-ACP registers its interoperable object reference (IOR) with the SAE so that the SAE will communicate only with the primary SRC-ACP. When the primary SRC-ACP becomes unavailable, the secondary SRC-ACP assumes the role of the primary SRC-ACP and performs state synchronization if necessary.

API for ACP

You can develop your own application to update information about subscribers and congestion points for SRC-ACP. The application can call one method to interact with SRC-ACP. This method is called:

update (in RemoteUpdateType rut, in TagValueList attrs)

The method takes a property-value pair and passes the information to SRC-ACP. For information about the properties and values you can pass to SRC-ACP, see the file *acpPlugin.idl* in the folder *SDK/idl* in the SRC software distribution.

To create an application that updates SRC-ACP remotely:

1. Compile the IDL file, and generate the code in the language in which you want to write the application.

2. Write the application, and include the generated code for the IDL file.

3. Use the CORBA object reference defined in the property ACP.syncRateAdaptor.ior to send data from the application to SRC-ACP.

For information about the interfaces, properties, and methods available in the CORBA remote API for ACP, see the documentation in the SRC software distribution at *SDK/doc/idl/acp/html/index.html*.