

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 topic 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. Table 1 on page 1 shows the edge congestion points and the corresponding locations in the directory.

For information about the NAS port ID, see Using Flexible RADIUS Packet Definitions.

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

Table 1: Congestion Points Derived Through NAS Port ID

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

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.

In cases where the combination of VR and service name do not uniquely identify the backbone congestion point, you can use backbone congestion point expressions and Python scripts to classify the backbone congestion point. Python scripts are executed when evaluating the congestion point expression. The format of the backbone congestion point expression is similar to the expression used in the congestion point profile. You can embed Python expressions, such as service plug-in attributes, in the congestion point expression. As a result, you can derive multiple backbone congestion points from a single service session.

For example, you can have a video-on-demand service that uses multiple video servers. One label-switched path (LSP) with the same parameters is created for each link between a video server and an access router. SRC-ACP uses the network interface configuration information to generate the DN *interfaceName* = *< NetworkInterface >* , *orderedCimKeys* = *< NetworkDevice >* , *o* = *AdmissionControl* , *o* = *umc* as a template for the congestion point. When receiving a service request, the server activates the service for the subscriber on the appropriate congestion point. The backbone congestion point corresponds to the evaluation of the backbone congestion point expression.

- Related Topics**
- For more information about automatically deriving congestion points from a configured congestion point profile, see [Deriving Congestion Points Automatically](#).
 - [Defining a Congestion Point Profile](#)
 - [Configuring SRC-ACP](#)

