Chapter 8
Overview of Plug-Ins Included with the SAE

This chapter gives an overview of the features of the SAE. Topics include:

- How Internal Plug-Ins Work on page 121

- Types of Internal Plug-Ins on page 122

- Assigning DHCP Addresses to Subscribers on page 124

- Creating and Tracking Subscriber Sessions on page 126

- Activating and Tracking Service Sessions on page 127

How Internal Plug-Ins Work

Plug-ins work with the SAE through events. Events such as subscriber logins and logouts, as well as service activation and deactivation, trigger the SAE to create event objects and send them to plug-in instances that are configured to receive the events. When a plug-in receives an event, it processes the event. For example, when a subscriber logs in, the SAE sends the username and password to an authentication plug-in that compares the username and password with data stored in a directory.

The plug-in configuration is made up of a plug-in pool and event publishers.

Plug-In Pool

The plug-in pool consists of plug-in instances. A plug-in instance describes a particular plug-in that can handle events that it receives from the SAE. An authorization plug-in instance might be set up to perform RADIUS authentication when it receives a subscriber login event. A tracking plug-in instance might be set up to write accounting information to a file when it receives service session events.

For each type of plug-in you can create multiple instances that contain different configurations of the plug-in.

If you have multiple retailers, you might use different authentication methods and servers to authenticate each retailer's subscribers. In this case you could set up an authentication plug-in instance for each retailer.

You could also set up a tracking plug-in instance to write certain accounting information to a file whenever it receives an event. Then you could set up another instance that writes different accounting information to a different file. You could then use one instance to track subscriber sessions and another to track service sessions. Or you could set up plug-in instances to track different types of services.

Event Publishers

Event publishers tell the SAE which events to send to which plug-in instances. There are four types of event publishers. Each type determines the scope of events that are sent to plug-in instances.

- Service-specific publishers—Authenticate subscribers of a particular service, authorize sessions for the service, and track subscriber activity related to the service

- Retailer-specific publishers—Authenticate and track subscribers and authorize DHCP address allocations for subscribers who log in to the domain(s) of a particular retailer

- Virtual router–specific publishers—Authenticate and track managed interfaces on a particular virtual router

- Global publishers—Authorize all subscriber sessions, track all subscriber and service sessions, authorize DHCP address allocations for all DHCP subscribers, and authorize all subscribers to change their subscriptions; authenticate subscribers and authorize DHCP address allocations for subscribers who log in to a retailer domain for which no retailer-specific authentication plug-ins are specified; and track all router interfaces that the SAE manages

Each publisher can notify a number of plug-in instances when an event occurs, and each plug-in instance can be registered with a number of publishers.

Types of Internal Plug-Ins

There are two main types of plug-ins: authorization plug-ins and tracking plug-ins.

Authorization Plug-Ins

Authorization plug-ins can perform both authentication (that is, verify the originator of a request) and authorization. Authorization can include the setting of service session parameters such as session timeout or authorizing services based on the current load of the router.

You can set up authorization plug-ins to:

- Globally authorize all subscriber sessions.

- Authenticate subscribers who belong to a particular retailer's domain.

- Globally authenticate and/or authorize all service sessions.

- Authenticate and/or authorize sessions for a particular service.

- Globally authorize DHCP address allocations.

- Authorize DHCP address allocation for subscribers who log in to a particular retailer's domain.

- Globally authorize subscribers to change their subscriptions.

- Authenticate administrators so that they can access SDX Web Admin.

☞ **NOTE:** Event publishers send events to all configured plug-in instances. For authentication to succeed, all authentication plug-ins that receive the authentication request must grant authentication.

Tracking Plug-Ins

Tracking plug-ins track activity or log accounting information. You can set up tracking plug-ins to:

- Globally track all subscribers.

- Track subscribers who belong to a particular retailer's domain.

- Globally track all service sessions.

- Track service sessions for individual services.

- Track QoS service sessions for individual services and attach the required QoS profile to the JUNOSe subscriber interface.

Tracking plug-ins keep the state of active sessions and provide usage and accounting data. For each subscriber and service session, plug-ins can track when the session is activated and deactivated and can keep interim updates. For example, when the SAE activates a service, it sends a Service Session Start event to tracking plug-in instances that are registered to receive events for that service. When the service is stopped, the SAE sends a Service Session Stop event to all tracking plug-ins that received the Service Session Start event. If interim accounting is configured, service session interim update events are sent at regular intervals to all tracking plug-ins that are registered to receive the event.

One application of tracking plug-ins is to keep usage records, such as session time and volume counters. Service-tracking plug-ins can set a timeout for a service session in response to start and interim updates that the plug-in receives for the session. When a service session is active longer than the defined timeout, the SAE stops the session and sends service session stop events to the tracking plug-ins.

Another application is to track QoS services and attach the required QoS profile to the subscriber interface. See *SRC-PE Solutions Guide, Chapter 1, Managing Tiered and Premium Services with QoS on JUNOSe Routers with the SRC CLI.*

Customizing RADIUS Packets with Plug-Ins

RADIUS internal plug-ins include flexible RADIUS plug-ins and custom RADIUS plug-ins that let you customize RADIUS authentication and accounting packets that the SAE sends to RADIUS servers. You can specify which fields are included in various types of RADIUS packets and what information is contained in the fields.

For example, you can specify values in authentication response packets that will set session and idle timeouts, set the RADIUS class, and set the session volume quota. For accounting packets, you can specify which fields to include in accounting records.

For DHCP subscribers, you can set up RADIUS authorization plug-ins to return to the router attributes that can be used to select a DHCP address or select a fixed address for each subscriber.
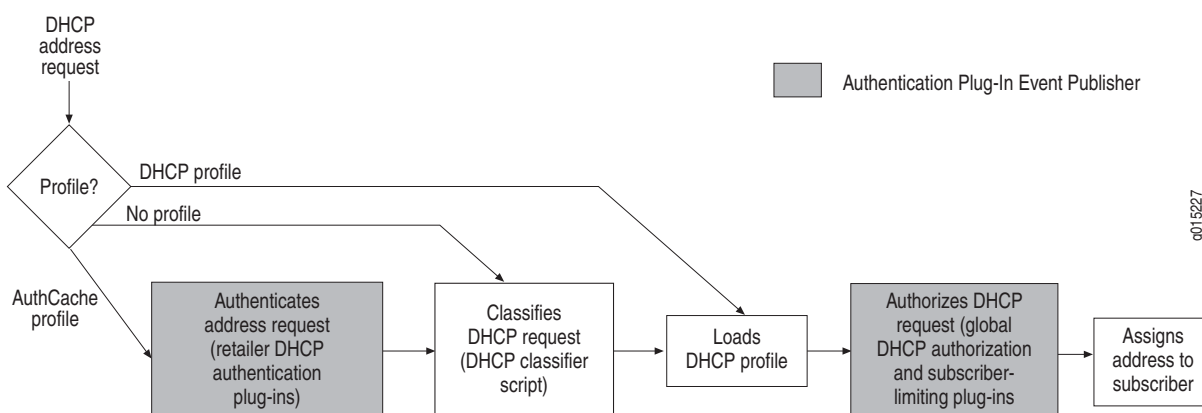
The main difference between flexible RADIUS plug-ins and custom RADIUS plug-ins is that custom plug-ins are designed to deliver better system performance than the flexible RADIUS plug-ins. To use a custom plug-in, you must provide a Java class that implements the SPI defined in the RADIUS client library. Use this SPI to specify which fields and field values to include in RADIUS accounting packets. The RADIUS client library is part of the SAE core API.

To customize RADIUS packets with a flexible RADIUS plug-in, see *Defining RADIUS Packets for Flexible RADIUS Plug-Ins with the SRC CLI* on page 163.

Assigning DHCP Addresses to Subscribers

Figure 24 shows the process that the SAE uses to assign addresses to DHCP subscribers.

Figure 24: DHCP Address Assignment

To create and track a subscriber session for DHCP subscribers, the SAE:

1. Uses the client's media access control (MAC) address to look up a profile in cache or in the directory.

    a. If the SAE finds an authCache profile, it continues with Step 2.

    b. If the SAE does not find a profile, it skips to Step 3.

    c. If the SAE finds a DHCP profile, it skips to Step 4.

2. Authenticates the address request.

    The SAE authenticates the request by using the configured DHCP authentication plug-ins. The DHCP authentication plug-ins are configured in the Retailer object in the directory. The SAE selects the retailer based on the domain name of the login request. If the Retailer object does not specify a DHCP authentication plug-in, the default retailer authentication plug-in is used for authentication.

    If authentication fails, the SAE sends a discover decision with accept = false to the router.

3. Classifies the DHCP request.

    The SAE runs a DHCP classification script to select the DHCP profile to load. If it does not find a profile, the SAE sends a discover decision with accept = false to the router.

4. Loads a DHCP profile.

    The SAE loads the selected DHCP profile from the directory.

5. Authorizes the DHCP request.

    The SAE authorizes the request by using the globally configured DHCP authorization plug-ins, which can include a subscriber-limiting plug-in.

    Note that if the DHCP profile contains configuration parameters and the DHCP authorization plug-ins also return parameters, the plug-in parameters take precedence.
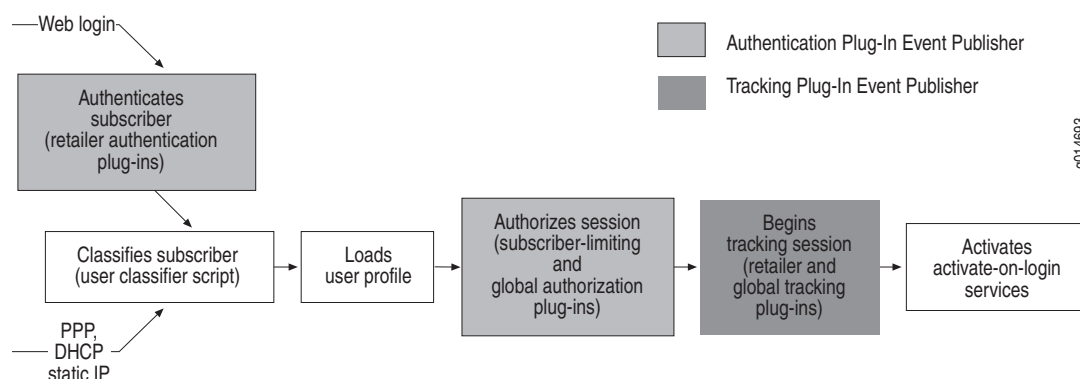
6. Assigns the address to the subscriber.

    The SAE sends a DHCP discover decision to the router, which enables the router to assign an address to the subscriber. When the subscriber accepts the assigned address, the router sends an address request to the SAE, and the SAE starts processing a DHCP login request. See *Creating and Tracking Subscriber Sessions* on page 126.

Creating and Tracking Subscriber Sessions

Figure 25 shows the process that the SAE uses to create and begin tracking subscriber sessions.

Figure 25: Creating and Tracking Subscriber Sessions

To create and track a subscriber session, the SAE:

1. Authenticates the login request.

   a. Web logins are authenticated by the SAE directly. The SAE maps the login request to a retailer object in the directory by matching the requested domain name. If the retailer object:

      ■ Has an authentication plug-in configured, the SAE asks the plug-in to authenticate the subscriber.

      ■ Does not have an authentication plug-in configured, the SAE sends the authentication request to the default retailer authentication plug-in.

   b. PPP and static IP interface addresses are authenticated by the router using the RADIUS setup configured in the router. The SAE is notified only after the authentication is completed successfully.

2. Classifies the subscriber.

   The SAE runs a subscriber classification script to select the subscriber profile to load.

3. Loads a subscriber profile.

   The SAE loads the selected subscriber profile from the directory.

4. Authorizes the subscriber session.

   The SAE authorizes the subscriber session before it starts the session:

   a. The SAE checks the number of concurrent logins of the subscriber profile and its parent and sibling profiles and sends an event to the subscriber-limiting plug-in. If the maximum number of allowed concurrent logins configured in the plug-in is exceeded, the subscriber session is not authorized.

   b. The SAE calls the global subscriber authorization plug-in instances, which can perform custom authorization.

   If any of the previous steps fail, the SAE either keeps the currently active subscriber profile (in case of a Web login) or loads the unauthenticated subscriber profile. The reason for the failure is stored in the unauthenticated profile and can be displayed when the subscriber eventually connects to the portal.

5. Sends start subscriber tracking events.

   The SAE sends subscriber session start events to tracking plug-ins configured for the associated retailer and to global subscriber tracking plug-in instances.

   When a subscriber session is closed, the SAE sends subscriber session stop tracking events to the same plug-ins that received the subscriber session start events.
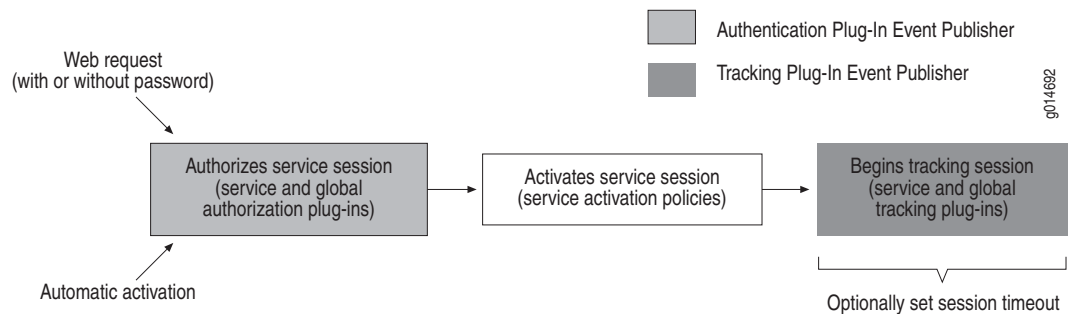
   The SAE does not create subscriber session interim update events.

6. Activates services for the subscriber that are set up to activate on login.

Activating and Tracking Service Sessions

Figure 26 shows the process that the SAE uses to activate and then track services. The SAE can activate services in one of two ways:

- Automatically—After the SAE creates a subscriber session, it activates all activate-on-login service subscriptions.

- Manually—Through a call of the portal application programming interface (API) method Subscription.setActive. This method is typically provided in the form of a Web portal and allows interaction with the subscriber.

Figure 26: Activating and Tracking Service Sessions

To activate and begin tracking a service session, the SAE:

1. Authorizes the service session.

   The SAE sends events to authorization plug-in instances configured for the service and to global service authorization plug-in instances.

   Service authorization plug-ins may perform authentication as well as authorization. If you define a plug-in instance to perform authentication, the portal developer must set username and password values before subscribers try to activate the service. Because the subscriber must provide the username and password, it is not possible to automatically activate a service that requires authentication.

2. Activates the services by applying service activation policies.

3. Begins tracking the service.

   Sends a service session start event to the tracking plug-in instances configured for the service and to the global service tracking plug-in instances. If interim accounting is configured, a service session interim update event is sent at regular intervals to all tracking plug-ins that are registered to receive the event.

When a service is stopped (either explicitly through a call to the portal API, or implicitly through the termination of the associated subscriber session or through a timeout), a service session stop event is sent to all tracking plug-ins that received the service session start event.

Service-tracking plug-ins can set the session timeout of a service session in response to Service Session Start and Service Session Interim Update events. When a service session is active longer than the defined timeout, the SAE closes the session and sends the appropriate Service Session Stop events.