

Overview of Flexible RADIUS Plug-Ins

Flexible RADIUS accounting and authentication plug-ins allow you to define the content of RADIUS packets that the SAE sends to RADIUS servers. You can specify which attributes are included in different types of RADIUS packets (for example, session start or stop requests, or accounting on or off requests). You can also specify what information is contained in the attribute fields.

A RADIUS attribute configuration consists of RADIUS attribute instances. Each instance defines attributes for a specific type of packet—For example, start requests or accounting off requests.

Within each attribute instance, you define individual RADIUS attributes. The following is a RADIUS attribute instance for authentication requests:

```
radius-attributes auth {  
  attributes {  
    User-Name loginId;  
    User-Password password;  
    NAS-Identifier localNasId;  
    NAS-IP-Address localNasIp;  
    NAS-Port nasPort;  
  }  
}
```

Each RADIUS packet template can consist of multiple RADIUS attribute instances.

Using Default RADIUS Templates

The SRC software comes with two default templates:

- **stdAcct**—Defines RADIUS accounting packets and is used in the default RADIUS flexible accounting plug-in instance **flexRadiusAcct**.
- **stdAuth**—Defines RADIUS authentication packets and is used in the default RADIUS flexible authentication plug-in instance **flexRadiusAuth**.

Naming RADIUS Attribute Instances

Attribute instances define attributes for a specific type of RADIUS packet. The name that you assign to an attribute instance specifies the type of packet to which the attribute definition is applied. Table 1 on page 1 lists the available packet types.

Table 1: RADIUS Attribute Instance Names

| Attribute Instance (Packet-Type) | Type of RADIUS Packet to Which Attribute Definition Is Applied |
|-------------------------------------|--|
| acct | Any accounting request |
| auth | Any authentication request |
| authresp | Any authorization response |

Table 1: RADIUS Attribute Instance Names *(continued)*

| Attribute Instance (Packet-Type) | Type of RADIUS Packet to Which Attribute Definition Is Applied |
|-------------------------------------|--|
| dhcprsp | DHCP response |
| off | Accounting-Off requests |
| on | Accounting-On requests |
| onoff | Accounting-On or Accounting-Off requests |
| start | Start requests |
| startstop | Start, Stop, or Interim Update requests |
| stop | Stop or Interim Update requests |
| svcacct | Service Session Start, Stop, or Interim requests |
| svcresp | Any service authorization response |
| svcstart | Service Session Start requests |
| svcstop | Service Session Stop or Interim requests |
| useracct | Subscriber Session Start, Stop, or Interim requests |
| userresp | Any subscriber authorization response |
| userstart | Subscriber Session Start requests |
| userstop | Subscriber Session Stop, or Interim requests |

Defining RADIUS Attributes

RADIUS attribute definitions consist of a RADIUS attribute and a value for the RADIUS attribute.

You can define values for standard RADIUS attributes or JUNOS vendor-specific attributes (VSAs).

Standard RADIUS Attributes

For standard RADIUS attributes, use a name or number as defined in *RFC 2865—Remote Authentication Dial In User Service (RADIUS) (June 2000)*, *RFC 2866—RADIUS Accounting (June 2000)*, or *RFC 2869—RADIUS Extensions (June 2000)*. For a full list, see www.iana.org/assignments/radius-types.

Juniper Networks VSAs

For Juniper Networks VSAs, use one of the following formats:

- Vendor-Specific.4874. <vsa#> . <type>
- 26.4874. <vsa#> . <type>

where <type> is one of the following:

- text—Indicates that the value is 1–253 octets containing UTF-8 encoded characters
- string—Indicates that the value is 1–253 octets containing binary data
- address—Indicates that the value is a 32-bit value
- integer—Indicates that the value is a 32-bit unsigned value
- time—Indicates that the value is a 32-bit unsigned value, seconds since 00:00:00 UTC, January 1, 1970

The following is an example of RADIUS attribute instances that define RADIUS VSAs.

```
radius-attributes svcresp {
  attributes {
    Session-Timeout setSessionTimeout(ATTR);
    Idle-Timeout setIdleTimeout(ATTR);
    vendor-specific.Juniper.Sdx-Session-Volume-Quota setSessionVolumeQuota(ATTR);
    vendor-specific.WISPr.Redirection-URL "setProperty(\"startURL=%s\" % ATTR)";
    vendor-specific.WISPr.Bandwidth-Min-Up "setSubstitution(\"min_up_rate=%s\" % ATTR)";
    vendor-specific.WISPr.Bandwidth-Min-Down "setSubstitution(\"min_down_rate=%s\" % ATTR)";
    vendor-specific.WISPr.Bandwidth-Max-Up "setSubstitution(\"max_up_rate=%s\" % ATTR)";
    vendor-specific.WISPr.Bandwidth-Max-Down "setSubstitution(\"max_down_rate=%s\" % ATTR)";
  }
}
radius-attributes dhcpresp {
  attributes {
    Framed-Pool setPoolName(ATTR);
    Framed-IP-Address setUserIpAddress(ATTR);
    26.4874.1.text setAuthVirtualRouterName(ATTR);
    26.4874.2.text setPoolName(ATTR);
    26.4874.31.text setServiceBundle(ATTR);
  }
}
```

- Related Topics**
- Using Flexible RADIUS Packet Definitions
 - Configuring a RADIUS Packet Template
 - Configuring UDP Ports for RADIUS Plug-Ins
 - Defining the Values of RADIUS Attributes

