Application-Level Session Tracking and QoS Control

Modified: 2019-06-21

Juniper Networks, Inc.
1133 Innovation Way
Sunnyvale, California 94089
USA
408-745-2000
www.juniper.net

Application-Level Session Tracking and QoS Control

The information in this document is current as of the date on the title page.

YEAR 2000 NOTICE

Juniper Networks hardware and software products are Year 2000 compliant. Junos OS has no known time-related limitations through the year 2038. However, the NTP application is known to have some difficulty in the year 2036.

END USER LICENSE AGREEMENT

The Juniper Networks product that is the subject of this technical documentation consists of (or is intended for use with) Juniper Networks software. Use of such software is subject to the terms and conditions of the End User License Agreement ("EULA") posted at https://support.juniper.net/support/eula/. By downloading, installing or using such software, you agree to the terms and conditions of that EULA.

Table of Contents

List of Figures

List of Tables

About the Documentation

Documentation and Release Notes

To obtain the most current version of all Juniper Networks® technical documentation, see the product documentation page on the Juniper Networks website at https://www.juniper.net/documentation/.

If the information in the latest release notes differs from the information in the documentation, follow the product Release Notes.

Juniper Networks Books publishes books by Juniper Networks engineers and subject matter experts. These books go beyond the technical documentation to explore the nuances of network architecture, deployment, and administration. The current list can be viewed at https://www.juniper.net/books.

Documentation Conventions

Table 1 on page x defines notice icons used in this guide.

Table 1: Notice Icons

Icon	Meaning	Description
	Informational note	Indicates important features or instructions.
	Caution	Indicates a situation that might result in loss of data or hardware damage.
	Warning	Alerts you to the risk of personal injury or death.
	Laser warning	Alerts you to the risk of personal injury from a laser.
	Tip	Indicates helpful information.
	Best practice	Alerts you to a recommended use or implementation.

Documentation Conventions

Table 1 on page x defines the notice icons used in this guide. Table 3 on page xi defines text conventions used throughout this documentation.

Table 2: Notice Icons

Icon	Meaning	Description
	Informational note	Indicates important features or instructions.
	Caution	Indicates a situation that might result in loss of data or hardware damage.
	Warning	Alerts you to the risk of personal injury or death.
	Laser warning	Alerts you to the risk of personal injury from a laser.
	Tip	Indicates helpful information.
	Best practice	Alerts you to a recommended use or implementation.

Table 3: Text Conventions

Convention	Description	Examples
Bold text like this	• Represents keywords, scripts, and tools in text. • Represents a GUI element that the user selects, clicks, checks, or clears.	• Specify the keyword **exp-msg**. • Run the **install.sh** script. • Use the **pkgadd** tool. • To cancel the configuration, click **Cancel**.
Bold text like this	Represents text that the user must type.	**user@host# set cache-entry-age** *cache-entry-age*

`Fixed-width text like this`	Represents information as displayed on your terminal's screen, such as CLI commands in output displays.	`nic-locators {`
		`  login {`
		`  resolution {`
		`  resolver-name /realms/`
		`  login/A1;`
		`  key-type LoginName;`
		`  value-type SaeId;`
		`  }`

Regular sans serif typeface	• Represents configuration statements. • Indicates SRC CLI commands and options in text. • Represents examples in procedures. • Represents URLs.	• **system ldap server{** **stand-alone;** • Use the **request sae modify device failover** command with the **force** option • **user@host# . . .** • https://www.juniper.net/techpubs/software/management/sdx/api-index.html

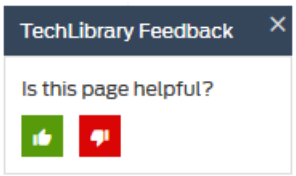
Table 3: Text Conventions (continued)

Italic sans serif typeface	Represents variables in SRC CLI commands.	**user@host# set local-address *local-address***
Angle brackets	In text descriptions, indicate optional keywords or variables.	Another runtime variable is <gfwif>.
Key name	Indicates the name of a key on the keyboard.	Press Enter.
Key names linked with a plus sign (+)	Indicates that you must press two or more keys simultaneously.	Press Ctrl + b.
Italic typeface	• Emphasizes words. • Identifies book names. • Identifies distinguished names. • Identifies files, directories, and paths in text but not in command examples.	• There are two levels of access: *user* and *privileged*. • *SRC-PE Getting Started Guide.* • *o=Users, o=UMC* • The */etc/default.properties* file.
Backslash	At the end of a line, indicates that the text wraps to the next line.	Plugin.radiusAcct-1.class=\ net.juniper.smgt.sae.plugin\ RadiusTrackingPluginEvent
Words separated by the | symbol	Represent a choice to select one keyword or variable to the left or right of this symbol. (The keyword or variable may be either optional or required.)	diagnostic | line

Documentation Feedback

We encourage you to provide feedback so that we can improve our documentation. You can use either of the following methods:

- Online feedback system—Click TechLibrary Feedback, on the lower right of any page on the Juniper Networks TechLibrary site, and do one of the following:



  - Click the thumbs-up icon if the information on the page was helpful to you.
  - Click the thumbs-down icon if the information on the page was not helpful to you or if you have suggestions for improvement, and use the pop-up form to provide feedback.

- E-mail—Send your comments to **techpubs-comments@juniper.net**. Include the document or topic name, URL or page number, and software version (if applicable).

Requesting Technical Support

Technical product support is available through the Juniper Networks Technical Assistance Center (JTAC). If you are a customer with an active Juniper Care or Partner Support Services support contract, or are covered under warranty, and need post-sales technical support, you can access our tools and resources online or open a case with JTAC.

- JTAC policies—For a complete understanding of our JTAC procedures and policies, review the *JTAC User Guide* located at https://www.juniper.net/us/en/local/pdf/resource-guides/7100059-en.pdf.
- Product warranties—For product warranty information, visit https://www.juniper.net/support/warranty/.
- JTAC hours of operation—The JTAC centers have resources available 24 hours a day, 7 days a week, 365 days a year.

Self-Help Online Tools and Resources

For quick and easy problem resolution, Juniper Networks has designed an online self-service portal called the Customer Support Center (CSC) that provides you with the following features:

- Find CSC offerings: https://www.juniper.net/customers/support/
- Search for known bugs: https://prsearch.juniper.net/
- Find product documentation: https://www.juniper.net/documentation/
- Find solutions and answer questions using our Knowledge Base: https://kb.juniper.net/
- Download the latest versions of software and review release notes: https://www.juniper.net/customers/csc/software/
- Search technical bulletins for relevant hardware and software notifications: https://kb.juniper.net/InfoCenter/
- Join and participate in the Juniper Networks Community Forum: https://www.juniper.net/company/communities/
- Create a service request online: https://myjuniper.juniper.net

To verify service entitlement by product serial number, use our Serial Number Entitlement (SNE) Tool: https://entitlementsearch.juniper.net/entitlementsearch/

Creating a Service Request with JTAC

You can create a service request with JTAC on the Web or by telephone.

- Visit https://myjuniper.juniper.net.
- Call 1-888-314-JTAC (1-888-314-5822 toll-free in the USA, Canada, and Mexico).

For international or direct-dial options in countries without toll-free numbers, see https://support.juniper.net/support/requesting-support/.

PART 1

Overview

CHAPTER 1

Overview of SRC Application Library

SRC Component Overview

The SRC software is a dynamic system. It contains many components that you use to build a subscriber management environment. You can use these tools to customize and extend the SRC software for your use and to integrate the SRC software with other systems. The SRC software also provides the operating system and management tools for C Series Controllers.

Table 4 on page 3 gives a brief description of the components that make up the SRC software.

Table 4: Descriptions of SRC Components

Component	Description
Server Components	
Service activation engine (SAE)	• Authorizes, activates, and deactivates subscriber and service sessions by interacting with systems such as Juniper Networks routers, cable modem termination system (CMTS) devices, RADIUS servers, and directories. • Collects accounting information about subscribers and services from routers, and stores the information in RADIUS accounting servers, flat files, and other accounting databases. • Provides plug-ins and application programming interfaces (APIs) for starting and stopping subscriber and service sessions and for integrating with systems that authorize subscriber actions and track resource usage.
Subscriber Information Collector (SIC)	The SIC listens for RADIUS accounting events from IP edge devices (accounting clients) and forwards them to a remote AAA server, allowing the SRC software to gain increased subscriber awareness. Additionally, the SIC can optionally edit accounting events before routing them.
Network information collector (NIC)	Collects information about the state of the network and can provide a mapping from a given type of network data to another type of network data.
Redirect Server	Redirects HTTP requests received from IP Filter to a captive portal page.

Table 4: Descriptions of SRC Components (continued)

Component	Description
3GPP Gateway	The SRC Third-Generation Partnership Project (3GPP) gateway is a Diameter-based component in the SRC software, which provides integration with 3GPP Policy and Charging Control environments, to provide fixed-mobile convergence (FMC). The SRC 3GPP gateway provides Gx-based integration with the Policy and Charging Rules Function (PCRF). The SRC 3GPP gateway uses the northbound Gx interface to mediate between the PCRF and Juniper Networks routers like the E Series Broadband Services routers and MX Series routers. The northbound Gx interface on the SRC 3GPP gateway communicates with the PCRF using the Diameter protocol.
3GPP Gy	The SRC 3GPP Gy is a Diameter-based component in the SRC software, which provides Gy-based integration with the Online Charging System (OCS), to provide FMC. The SRC 3GPP Gy uses the northbound Gy interface to handle charging-related information between the OCS and Juniper Networks routers like the E Series Broadband Services routers and MX Series routers. The northbound Gy interface communicates with the OCS using the Diameter protocol.
Web Application Service	The SRC software includes a Web application server that hosts the Web Services Gateway and the Volume Tracking Application (SRC VTA). In production environments, this application server is designed to host only these applications. However, you can load your own applications into this server for testing or demonstration purposes.
Web Services Gateway	Allows a gateway client—an application that is not part of the SRC network—to interact with SRC components through a Simple Object Access Protocol (SOAP) interface. The Web Services Gateway provides the Dynamic Service Activator which allows a gateway client to dynamically activate and deactivate SRC services for subscribers and to run scripts that manage the SAE.
Monitor Components Connectivity (MCC)	Monitors the connectivity state between SAEs in a community and between SAE and RADIUS server periodically and collects diagnostic information about the connectivity state of components, such as connection error, connection timeout, and socket read/write timeout.
Repository	
Directory	The SRC software includes the Juniper Networks database, which is a built-in Lightweight Directory Access Protocol (LDAP) directory for storing all SRC data including services, policies, and small subscriber databases. For large subscriber databases, you must supply your own directory.
SRC Configuration and Management Tools	
SRC command line interface (CLI)	Provides a way to configure the SRC software on a C Series Controller from a Junos OS–like CLI. The SRC CLI includes the policies, services, and subscribers CLI, which has separate access privileges.
C-Web interface	Provides a way to configure, monitor, and manage the SRC software on a C Series Controller through a Web browser. The C-Web interface includes a policies, services, and subscribers component, which has separate access privileges.
Simple Network Management Protocol (SNMP) agent	Monitors system performance and availability. It runs on all the SRC hosts and makes management information available through SNMP tables and sends notifications by means of SNMP traps.

Table 4: Descriptions of SRC Components (continued)

Component	Description
Service Management Applications (Run on external system)	
IMS Services Gateway	Integrates into an IP multimedia system (IMS) environment. The SRC software provides a Diameter protocol-based interface that allows the SRC software to integrate with services found on the application layer of IMS.
SRC Programming Interfaces	
NETCONF API	Allows you to configure or request information from the NETCONF server on a C Series Controller that runs the SRC software. Applications developed with the NETCONF API run on a system other than a C Series Controller.
CORBA plug-in service provider interface (SPI)	Tracks sessions and enables linking the rest of the service provider's operations support system (OSS) with the SRC software so that the OSS can be notified of events in the life cycle of SAE sessions. Hosted plug-ins only.
CORBA remote API	Provides remote access to the SAE core API. Applications that use these extensions to the SRC software run on a system other than a C Series Controller.
NIC access API	Performs NIC resolutions. Applications that use these extensions to the SRC software run on a system other than a C Series Controller.
SAE core API	Controls the behavior of the SRC software. Applications that use these extensions to the SRC software run on a system other than a C Series Controller.
Script services	Provides an interface to call scripts that supply custom services such as provisioning policies on a number of systems across a network.
VTA API	The Volume Tracking Application (VTA) API is a Simple Object Access Protocol (SOAP) interface that allows developers to create gateway clients and that administrators use to manage VTA subscribers and sessions. The SRC Web Services Gateway allows a gateway client—an application that is not part of the SRC network—to interact with SRC components, such as the VTA, through a SOAP interface.
Authorization and Accounting Applications	
AAA RADIUS servers	Authenticates subscribers and authorizes their access to the requested system or service. Accepts accounting data—time active and volume of data sent—about subscriber and service sessions. RADIUS servers run on a system other than a C Series Controller.
SRC Admission Control Plug-In (SRC ACP)	Authorizes and tracks subscribers' use of network resources associated with services that the SRC application manages.
Flat file accounting	Stores tracking data to accounting flat files that can be made available to external systems that send the data to a rating and billing system.

Table 4: Descriptions of SRC Components (continued)

Component	Description
Volume Tracking Application	The SRC Volume Tracking Application (SRC VTA) is an SRC component that allows service providers to track and control the network usage of subscribers and services. You can control volume and time usage on a per-subscriber or per-service basis. This level of control means that service providers can offer tiered services that use volume as a metric, while also controlling abusive subscribers and applications. When a subscriber or service exceeds bandwidth limits (or quotas), the SRC VTA can take actions including imposing rate limits on traffic, sending an e-mail notification, or charging extra for additional bandwidth consumed.
Demonstration Applications (available on the Juniper Networks Website)	
Enterprise Audit Plug-In	Defines a callback interface, which receives events when IT managers complete specified operations.
Enterprise Manager Portal	Allows service providers to provision services for enterprise subscribers on routers running JunosE or Junos OS and allows IT managers to manage services. Enterprise Manager Portal can be used with NAT Address Management Portal to allow service providers to manage public IP addresses for use with NAT services on routers running Junos OS and to all IT managers to make requests about public IP addresses through the Enterprise Manager Portal.
Monitoring Agent application	Integrates IP address managers, such as a DHCP server or a RADIUS server, into an SRC-managed network so that the SAE is notified about subscriber events. The Monitoring Agent application runs on a Solaris platform.
Residential service selection portals	Provides a framework for building Web applications that allow residential and enterprise subscribers to manage their own network services. It comes with several full-featured sample Web applications that are easy to customize and suitable for deployment. The Residential service selection portals run on a Solaris platform.
Sample enterprise service portal	Lets service providers supply an interface to their business customers for managing and provisioning services.

Related Documentation

- *SRC Product Description*

CHAPTER 2

Overview of Application-Level Session Tracking and QoS Control

Overview of Application-Level Session Tracking and QoS Control

The SRC software has been integrated with the Ellacoya Networks Deep Packet Inspection (DPI) platform to provide a traffic management solution that combines the advanced traffic identification and reporting features of the Ellacoya DPI with the SRC software's intelligent service policy enforcement. With this solution, providers can identify, monitor, and control traffic on a per-application or per-subscriber basis.

Application traffic such as peer-to-peer file sharing or instant messaging, which in many cases originates or terminates outside a provider's network, can cause abusive or indiscriminate consumption of bandwidth and affect a provider's ability to deliver its own services. In particular, services that require higher, guaranteed levels of performance, such as voice over IP (VoIP) or video on demand, can be affected. Having visibility into applications that are transported over the network and their associated bandwidth consumption at various times is important, as is the ability to control those applications.

The DPI solution allows providers to implement service control policies on specific traffic flows quickly and effectively. Such policies include throttling back, capping volume, or even enhancing bandwidth or service quality for sanctioned peer-to-peer applications.

Benefits of Application-Level Session Tracking and QoS Control

By identifying and effectively controlling traffic at the application level, service providers can:

- Put usage controls on applications on a subscriber basis. For example, you can put a quota limit on the amount of peer-to-peer traffic that a subscriber can consume in a month.

  Once subscribers have used their quota, you can apply a policy that throttles back or blocks a subscriber's peer-to-peer traffic, bill the subscriber for additional usage, or allow the subscriber to purchase additional quota.
- Limit the total percentage of network resources that a specific type of traffic is allowed to consume.
- Provide higher or guaranteed levels of performance for premium services by applying QoS control to application sessions. For example, two subscribers start an Xbox Live session. The Ellacoya DPI platform detects activity for this application, and sends application usage counters to the SRC software. The SRC software pushes policies that deliver a specific level of QoS for this application session to a router or other network device.
- Charge subscribers based on their usage of premium content-based services.
- Offer and charge for tiered Internet services based on both speed and application.
- Better support network planning functions by gaining an in-depth understanding of traffic flows and patterns on a per-subscriber and per-application basis.

Related Documentation

- Integration of the SRC Software and the Ellacoya DPI Platform on page 8
- Subscriber Login and Logout in a DPI Environment on page 12
- Loading the Sample Data for the DPI on page 27
- Configuring the Script Service on page 30

Integration of the SRC Software and the Ellacoya DPI Platform

This topic describes the integration of the SRC software with the Ellacoya DPI platform.

Ellacoya Networks DPI Platform

The SRC software is integrated with the Ellacoya IP Service Control System (IPSCS), which delivers comprehensive monitoring tools and extensive reporting that gives providers network visibility into and control over their subscribers, network, and service offerings (see Figure 1 on page 9). The IPSCS includes the following components:

- IPSCS e30 Switch, which provides application session and usage information based on real-time deep inspection of network traffic.
- Service logic engine (SLE), which is the control component of the IPSCS software. The SLE includes the Usage Quota Management System (UQMS) applications programming interfaces (API) that the SRC software can call to collect usage data and to indicate that subscriber sessions have started, changed, or stopped. Calls for the API are carried between the SLE and the SRC software by the Remote Method Invocation (RMI) protocol.

The IPSCS allows the creation of Usage Quota Management Systems, which allows third-party systems, such as the SRC software, to be integrated with the IPSCS. The UQMS is a provisioning system that determines business rules and communicates them to the IPSCS and other network components.

Service providers provision business rules on the UQMS. Based on the configuration instructions from the UQMS, the IPSCS collects usage statistics on specified intervals and applications, and then forwards the data to the UQMS for processing. The UQMS sends authentication and configuration information for subscribers to the IPSCS.

Juniper Networks Platforms

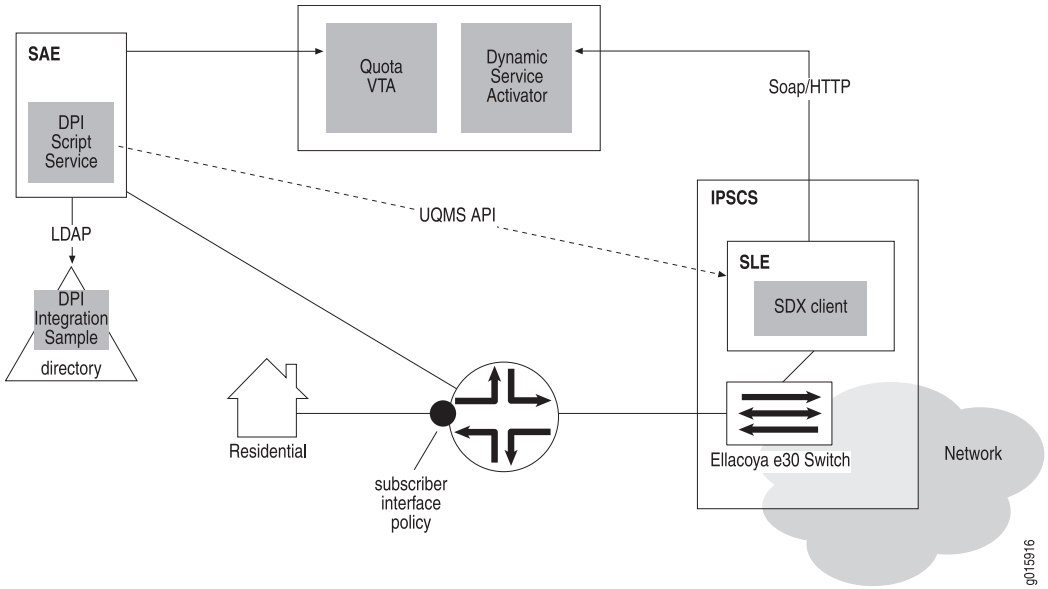
As shown in Figure 1 on page 9, the SRC software is integrated with the Ellacoya DPI platform through an SAE script service that works as a UQMS to the IPSCS. The script service sends event messages to and receives messages from the SLE through the UQMS API. The script service translates subscriber logins and logouts and service activation and deactivation to corresponding RMI function calls in the SLE. The script service collects accounting data from the SLE and maps the data to SRC service session usage data. The SRC software is integrated with the Ellacoya platform in a way that makes services implemented by the e30 Switch just like any other SRC service.

The SRC software supports the DPI solution on routers running JunosE or Junos OS, and on PacketCable Multimedia Specification (PCMM) policy servers and cable modem termination system (CMTS) devices.

You can integrate the SLE with the SRC software through a SOAP interface to the NIC and the SAE.

Figure 1 on page 9 shows the components involved in the SRC-DPI integration.

Figure 1: DPI Integration Overview



IPSCS Service Offers and Service Bundles

The Ellacoya IPSCS has the concept of service offers that contain a collection of service bundles. Service bundles are applied to a subscriber and contain a set of policy rules that define the patterns needed to recognize application sessions between specified source and destination networks. The service bundle includes the actions to take on application sessions, including marking specified streams with a DiffServ code point. Each rule in a service bundle can also have an activation period to control the time it is active.

A service offer also allows the association of traffic-accounting profiles (TAPs) with service bundles. TAPs are used to interface with external accounting systems. If a TAP is applied to a service bundle for a subscriber, then the counters for that service bundle are accumulated in accounting records indexed by TAP name.

Mapping Service Offers and Service Bundles to SRC Concepts

A service offer applied to a subscriber in the IPSCS corresponds to the SRC concept of the set of services active for a subscriber. A service bundle in an applied service offer corresponds to an SRC service activation.

A TAP applied to a service bundle roughly corresponds to accounting being enabled on an SRC policy.

The DPI integration maps SRC service activations and deactivations to changing the service offer applied to the subscriber so that it contains the corresponding set of service bundles. For example, a service provider offers three SRC services—Web access, e-mail, and peer-to-peer file sharing—to its subscribers. Each service can be dynamically enabled or disabled. The Ellacoya DPI system provides eight service offers that accommodate all possible combinations of the three SRC services.

Synchronization Between the SRC Software and the Ellacoya System

Table 5 on page 10 outlines the interactions that must be synchronized between the SRC software and the Ellacoya system.

Table 5: Synchronization Between the SRC Software and the Ellacoya System

Object	Source	Description
Service offer	Provisioned on the Ellacoya system	Contains the definition of the profile set in terms of network policies. For example, rate limits and ToS marking.
Subscriber	Dynamic	Subscribers are dynamically created in the Ellacoya database as the SRC software provisions them through the UQMS API.
Subscriber/Service offer binding	Dynamic	The binding of the subscriber to a service offer is done dynamically when the SRC software provisions the subscriber through the UQMS API.

Table 5: Synchronization Between the SRC Software and the Ellacoya System (continued)

Object	Source	Description
Subscriber/IP address binding	Dynamic	The binding between the subscriber and IP address is a transient binding communicated from the SRC software when the subscriber session starts.

Collecting Accounting Data

The IPSCS has a bulk statistics architecture where usage counters are collected for all active subscribers and services in bulk, and is controlled by a global accounting interval. The DPI script service polls for new usage data through the UQMS API based on the SLE global accounting interval. The SLE returns a list of file Uniform Resource Identifiers (URIs) of bulkstats files that have not been processed by the DPI script service.

After the DPI script service collects accounting data from the SLE, it maps the accounting data to usage data for an SRC service, and provides the usage data to the SAE. Because the SLE reports accounting data in the last accounting interval and the SAE expects service usage data since the service started, the script service adds the usage data in each accounting interval since the script service started. The script service keeps the counters in memory and also stores them so they can be recovered if there is an SAE failover.

The script service also stores a timestamp for the last time new data was reported to the SAE. This way, if there is a failover, the script service can determine when the doneWithFile method can be called for a file and whether or not a piece of usage data has been processed. The new SAE (after failover) requests the accounting data for sessions where the data in the file was reported to the previous SAE. At this time, the script service can compare timestamps (records in the file are also timestamped) to recognize that the data was previously reported and to recognize when all of the data in a file has been processed.

The IPSCS provides only one accounting record per accounting interval, per service offer, and per subscriber. This behavior means that if a DPI script service is stopped and restarted during one IPSCS accounting interval (usually 15 minutes), service usage of the two service sessions within that interval are allocated to one of the service sessions—the first service session for which the SAE requests an accounting update.

Related Documentation

- Overview of Application-Level Session Tracking and QoS Control on page 7
- Provisioning the IPSCS on page 33
- Subscriber Login and Logout in a DPI Environment on page 12
- Service Activation and Deactivation in a DPI Environment on page 12
- SRC Script Services for DPI on page 13

Subscriber Login and Logout in a DPI Environment

Login events are not directly handled by the SAE, and there is no need for subscriber tracking plug-ins. When a script service is activated, the script service determines the Ellacoya service offer name based on the SRC services that are to be activated on login for the subscriber. Then, the script service calls the startSubscriberRadiusSession method of the UQMS API, which starts a subscriber session, and provides the subscriber's IP address, ID, and service offer name to the SLE.

When a script service is deactivated because the subscriber logs out, the script service notifies the SLE to remove the subscriber session, and it ignores all subsequent service deactivations for the subscriber.

Related Documentation

- Overview of Application-Level Session Tracking and QoS Control on page 7
- Service Activation and Deactivation in a DPI Environment on page 12
- SRC Script Services for DPI on page 13
- Configuring Subscriptions to DPI Services on page 32
- Loading the Sample Data for the DPI on page 27

Service Activation and Deactivation in a DPI Environment

When the SAE activates the first DPI script service or deactivates the last DPI script service, the script service calls the startSubscriberRadiusSession() or stopSubscriberRadiusSession() methods (which start and stop subscriber sessions) in the UQMS API.

You can set up the script service so that when the first service that a script service implements is activated at login, the script service starts all other activate-on-login and persistently activated services for the subscriber that are also implemented by the script service. The combination of all activate-on-login and persistent services maps to one service offer in the SLE.

When there is a change to the subscriber's services that triggers a service activation—for example, the subscriber activates a service with a portal, the service provider adds new services, or the subscriber's quota is replenished—the script service calls the SLE changeSubscriberSession() method (which changes a subscriber session) with the new service offer name that corresponds to the combination of active DPI script services of the subscriber.

When deactivation of script services is triggered by changes to a subscriber's services—for example, the scheduled time for a service has ended, the service provider removes a service, or a subscriber exceeds the quota—the DPI script service calls the SLE changeSubscriberSession() method with the new service offer name that corresponds to the combination of active DPI script service of the subscriber.

The SAE expects the script service to return its final usage data when it stops the service. However, the SLE does not return the final accounting record synchronously when the changeSubscriberSession() method is called. When a DPI script service is stopped, the script service signals to the SAE that accounting data is to be collected at a later time.

Related Documentation

- Integration of the SRC Software and the Ellacoya DPI Platform on page 8
- Subscriber Login and Logout in a DPI Environment on page 12
- SRC Script Services for DPI on page 13
- Configuring Subscriptions to DPI Services on page 32
- Adding a Service Scope on page 28
- Configuring the Script Service on page 30

SRC Script Services for DPI

Every subscriber must have at least one DPI script service active during subscriber login. This DPI script service corresponds to a service bundle in the SLE that provides the default policy for processing subscriber traffic. To match the service bundle with the script service, the service bundle must have the same name as the DPI script service and be prefixed by SDX_. For example, if the script service is called dpiService1, the service bundle must be called SDX_dpiService1.

You can create a service scope to hold your script services. Putting a script service in a service scope allows the service to be used in different regions of the network. (See "Configuring a Virtual Router Object for DPI" on page 32 for more information.) The DPI sample data contains a service scope called DPI. You can model your script services after the sample data.

After you configure a service scope, you need to configure script services for the scope and configure parameters for the script service. For more information about configuring service scopes, script services, and parameters, see *Configuring Service Scopes (SRC CLI)*.

Related Documentation

- Integration of the SRC Software and the Ellacoya DPI Platform on page 8
- Provisioning the IPSCS on page 33
- Adding a Service Scope on page 28
- Creating a DPI Script Service on page 28
- Configuring the Script Service on page 30
- Loading the Sample Data for the DPI on page 27

PART 2

Installation

CHAPTER 3

Installation Tasks for SRC Application Library

SRC Application Library Software

You can access the software for the SRC application library, and the product *Release Notes* on the Juniper networks Web site at: https://www.juniper.net/support/products/src/index.html. The application library files are located in the **SRC_APLIB.tar.gz** file. Table 6 on page 17 lists the applications provided in the application library.

Table 6: Application Library Applications

Application	Type of Application	File or Directory in Archive File
Deep Packet Inspection Script Service	Solaris package	UMCdpi-ss
Threat Mitigation Application	Solaris package	**UMCthma**

The SRC software also provides sample applications in the **SDK+AppSupport+Demos+Samples.tar.gz** on the Juniper networks Web site at: https://www.juniper.net/support/products/src/index.html .

For information about the sample applications, see the *SRC PE Sample Applications Guide* on the Juniper Networks Web site at https://www.juniper.net/support/products/src/index.html .

The SRC Admission Control Plug-In (SRC ACP) component is installed on C Series Controllers as part of the SRC core software. The documentation for SRC ACP is in the *SRC PE Network Guide*.

Related Documentation

- Before You Install the SRC Applications on page 18
- Installing Sample SRC Data for Applications in the Application Library Guide on page 20
- Solaris Packages and Installation Folders for the SRC Application Library on page 18
- Installing SRC Application Packages on page 19

Before You Install the SRC Applications

Before you install applications, install necessary Solaris patches on the installation host, make sure that you understand if you want to root or nonroot users to have access to install and configure the application, and establish users and groups for software administration.

Before you install the UMCthma, install JBoss on the Solaris host. You can obtain the JBoss software in the **SDK+AppSupport+Demos+Samples.tar.gz** file available on the Juniper networks Web site at: https://www.juniper.net/support/products/src/index.html.

The SRC applications are distributed as Solaris packages or Web application archive (WAR) files.

Related Documentation

- SRC Application Library Software on page 17
- Installing SRC Application Packages on page 19
- Solaris Packages and Installation Folders for the SRC Application Library on page 18

Solaris Packages and Installation Folders for the SRC Application Library

Table 7 on page 18 lists the components for each application, their Solaris package names, and the directories where each component is installed by default. In Table 7 on page 18, the directories listed are all subordinate to */opt/UMC*.

Table 7: Solaris Packages and Installation Folders for Application Library

Application	Components Supplied with SRC	Package	Installation Directory
Threat Mitigation Application	• Threat Mitigation Application • Python Runtime Environment	• UMCthma • SMCpython	• conf/thma • python

Related Documentation

- SRC Application Library Software on page 17
- Before You Install the SRC Applications on page 18
- Installing SRC Application Packages on page 19
- Installing Sample SRC Data for Applications in the Application Library Guide on page 20
- Uninstalling SRC Packages on page 20

Installing SRC Application Packages

To install an application library package:

1. On the UNIX host where you will install the application library software, log in as root.
2. Copy the Solaris package for an application to a directory such as **/tmp**.
3. Launch the **pkgadd** tool.

   ```
   pkgadd -d /tmp
   ```

   The tool displays the available Solaris packages.
4. Enter the desired package(s).

   You can enter the name or number for a single package, multiple packages separated by spaces, or the keyword **all** to select all the packages.

   The tool displays the license agreement.
5. Press Enter to move through the agreement, and then enter **y** to accept the license agreement when prompted by the tool.
6. Follow the prompt directions to accept the installation directory for the package, to permit the use of superuser scripts required for the package, and so on.

NOTE: You can use the UNIX **swmtool** command to install the application packages, but this method requires that you install each application separately. If you use **admintool** directly, you can install multiple applications at the same time.

Related Documentation

- SRC Application Library Software on page 17
- Before You Install the SRC Applications on page 18
- Installing SRC Web Applications on page 20
- Uninstalling SRC Packages on page 20
- Solaris Packages and Installation Folders for the SRC Application Library on page 18
- Reviewing SRC Port Settings for SRC Applications on page 23

Uninstalling SRC Packages

Use the **pkgrm** command to uninstall application library components. For example, to remove the SRC Threat Mitigation package, issue the following command, and respond as prompted by the process:

```
pkgrm UMCvtacnf
```

Related Documentation

- SRC Application Library Software on page 17
- Installing SRC Application Packages on page 19
- Solaris Packages and Installation Folders for the SRC Application Library on page 18

Installing Sample SRC Data for Applications in the Application Library Guide

You can install sample data from the SRC CLI for the following applications:

- Dynamic Service Activator application
- Traffic-Mirroring Application

Related Documentation

- SRC Application Library Software on page 17
- Before You Install the SRC Applications on page 18
- Installing SRC Application Packages on page 19
- *Loading Sample Data into a Juniper Networks Database (SRC CLI)*
- *Dynamic Service Activator Overview*

Installing SRC Web Applications

We supply one WAR file for each Web application in the application library. Web applications must be deployed in a Web application server.

The exact way you install Web applications depends on the Web application server you are using and the particular Web application.

The following procedure provides general steps for installing a Web application:

1. Install the Web application server on the host.
2. If the Web application requires configuration of a properties file, complete the following procedure:
   a. Copy the WAR file from the archive file to a temporary folder on the host.
   b. Unpack the WAR file.

For information about unpacking and packing WAR files, see http://java.sun.com/j2se/1.4/docs/guide/jar/.

c. Edit the properties file for the Web application.

d. Repack the WAR file.

3. Deploy the WAR file by using the procedure appropriate for your Web application server.

   For information about deploying WAR files, see the documentation for your Web application software.

Related Documentation

- SRC Application Library Software on page 17
- Before You Install the SRC Applications on page 18
- Installing Web Applications Inside JBoss on page 21
- Removing SRC Web Applications on page 23
- Reviewing SRC Port Settings for SRC Applications on page 23

Installing Web Applications Inside JBoss

We provide the JBoss Web application server for Solaris with the sample and demonstration applications in the **SDK+AppSupport+Demos+Samples.tar.gz** file. available from the Juniper Web site at: https://www.juniper.net/support/products/src/index.html.

JBoss is an open-source Java application server that provides full support for J2EE application programming interfaces (APIs). The installation directory of the JBoss Application Server is **/opt/UMC/jboss**.

For Solaris, the JBoss Web application server package is called **UMCjboss.pkg**. The following shows an example procedure for installing the Solaris JBoss Web application server:

1. On the UNIX host where you want to install JBoss, log in as root.
2. Copy the Solaris package UMCjboss.pkg to a directory such as **/tmp**.
3. Launch the **pkgadd** tool.

   pgkgadd –d ***path to*****/UMCjboss.pkg**

   The tool displays the available Solaris packages.

```
The following packages are available:
  1  UMCjboss     JBoss J2EE Application Server
                  (sparc) Release jboss-6.0.0

Select package(s) you wish to process (or 'all' to process all packages).
(default: all) [?,??,q]:
```

4. Enter the desired packages.

   You can enter the name or number for a single package or multiple packages separated by spaces. Alternatively, enter the keyword “all” to select all the packages.

```
Processing package instance <UMCjboss> from /tmp/UMCjboss.pkg
JBoss J2EE Application Server(sparc)
Release jboss-6.0.0 http://jboss.org

Which configuration do you want to use? (default|all|minimal) [default] [?,q]all
```

5. If you want to deploy the VTA in a clustered Application Server environment, enter **all** as the configuration. By default, the default JBoss configuration is used. This is fine for demonstration purposes.

```
Which user-id shall be running the JBoss server [nobody] [?,q]
```

6. For this example, press **Enter** to select **nobody**.

```
Configuration: "all"
User:          "nobody"

Is this correct? [y,n,?,q] y
```

7. Enter **y**.

```
*********************************************************************
* If you want to change these values later, please edit the file *
* /etc/init.d/jboss                                                 *
*********************************************************************

The selected base directory </opt/UMC/jboss> must exist before installation
is attempted.

Do you want this directory created now [y,n,?,q] y
```

8. Enter **y** to create the directory.
9. Customize the properties file for the Web application.
10. Deploy the WAR file by copying it into the JBoss **default/deploy** directory.

    cp <filename>.war /opt/UMC/jboss/server/default/deploy

    JBoss automatically starts the Web application when a new WAR file is copied into the **deploy** directory.

11. (Optional) Start the JBoss application server.

    /etc/init.d/UMCjboss start

12. (Optional) Stop the JBoss application server.

```
/etc/init.d/UMCjboss stop
```

Related Documentation

- SRC Application Library Software on page 17
- Before You Install the SRC Applications on page 18
- Installing SRC Web Applications on page 20
- Removing SRC Web Applications on page 23
- Reviewing SRC Port Settings for SRC Applications on page 23

Removing SRC Web Applications

The way you remove a Web application depends on the Web application server that you are using. Refer to the documentation on removing Web applications for your server.

To undeploy a Web application from JBoss:

- Remove the WAR file from the JBoss *default/deploy* directory.

Related Documentation

- SRC Application Library Software on page 17
- Installing SRC Web Applications on page 20
- Installing Web Applications Inside JBoss on page 21

Reviewing SRC Port Settings for SRC Applications

If you use firewall software within your internal network, ensure that firewall settings allow traffic to and from the following ports for the applications that you implement in your environment. Table 8 on page 23 lists the default port settings for SRC applications.

Table 8: Default Port Settings for SRC Applications

Component	Type of Communication	Default Port Setting
Threat Mitigation Application	Oracle (if used)	TCP 1521
	PostgreSQL (if used)	TCP 5432
	MySQL (if used)	TCP 3306

Related Documentation

- SRC Application Library Software on page 17
- Before You Install the SRC Applications on page 18
- Installing SRC Application Packages on page 19
- Installing Sample SRC Data for Applications in the Application Library Guide on page 20
- Installing SRC Web Applications on page 20

PART 3

Configuration

CHAPTER 4

Configuration Tasks for the SRC Software for DPI Integration

Loading the Sample Data for the DPI

The DPI sample data includes directory entries for service and policy configurations that are provided as a guide for how to set up an IPSCS-based service to be used as a quota service for the SRC VTA.

To load the sample data for the DPI feature:

1. Go to the dpi folder. The default folder is:

   /opt/UMC/sae/var/scriptservice/dpi

2. Enter the following command, and respond to the prompts.

   ```
   # ./loadsample
   Please enter the authentication dn [cn=umcadmin,o=umc]:
   Please enter the authentication password [admin123]:
   Please enter the directory host address [127.0.0.1]:
   Please enter the directory port [389]:
   ```

The software adds the following sample data:

```
adding new entry l=dpi, o=Scopes, o=UMC

adding new entry serviceName=service1, l=dpi, o=Scopes, o=UMC
```

```
adding new entry serviceName=service2, l=dpi, o=Scopes, o=UMC
adding new entry serviceName=service3, l=dpi, o=Scopes, o=UMC
adding new entry retailername=SP-DPI, o=Users, o=umc
adding new entry ou=local, retailername=SP-DPI, o=Users, o=umc
adding new entry uniqueID=jane,ou=local, retailername=SP-DPI, o=Users, o=umc
adding new entry serviceName=service3,ou=local, retailername=SP-DPI, o=Users, o=umc
adding new entry serviceName=service1,ou=local, retailername=SP-DPI, o=Users, o=umc
adding new entry serviceName=service2,ou=local, retailername=SP-DPI, o=Users, o=umc
```

Related Documentation

- Integration of the SRC Software and the Ellacoya DPI Platform on page 8
- Provisioning the IPSCS on page 33
- Subscriber Login and Logout in a DPI Environment on page 12
- SRC Script Services for DPI on page 13
- Configuring a Virtual Router Object for DPI on page 32

Adding a Service Scope

To add a service scope:

1. Add a service scope with the SRC CLI or the C-Web interface.

   See *Configuring Service Scopes (SRC CLI)* and *Configuring Service Scopes (C-Web Interface)*

2. Optionally, configure a precedence for the service scope.

Related Documentation

- SRC Script Services for DPI on page 13
- Service Activation and Deactivation in a DPI Environment on page 12
- Configuring Subscriptions to DPI Services on page 32
- Creating a DPI Script Service on page 28
- Configuring the Script Service on page 30

Creating a DPI Script Service

To create a script service within the service scope:

1. Within the service scope, create a script service with the SRC CLI or the C-Web interface.

   See *Customizing Service Implementations* or *Configuring Script Services (C-Web Interface)*

2. Specify the values for the following options for the service.

Script Type

- Type of script that the script service uses.
- Value—You must use URL for DPI script services

Class Name

- Name of the class that implements the ScriptService SPI. The SAE instantiates this class when it starts the script service.
- Value—You must enter net.juniper.smgt.dpi.sle.EllacoyaScriptService for DPI script services

File/URL

- URL to the Java archive (JAR) files that are associated with the script.
- Value—For a DPI script service, you must include all of the following JAR files.
  - *file:///opt/UMC/sae/var/scriptservice/dpi/lib/dpi-ss.jar,*
  - *file:///opt/UMC/sae/var/scriptservice/dpi/lib/sle-client.jar,*
  - *file:///opt/UMC/sae/var/scriptservice/dpi/lib/jbossall-client.jar,*
  - *file:///opt/UMC/sae/var/scriptservice/dpi/lib/jdom.jar,*
  - *file:///opt/UMC/sae/var/scriptservice/dpi/lib/jnet.jar,*
  - *file:///opt/UMC/sae/var/scriptservice/dpi/lib/log4j.jar*

NOTE: All JAR files except dpi-ss.jar are Ellacoya SLE files, and must be compatible with the version of the SLE that is being used. Check the release notes for the version of the SLE that the JAR files packaged with the SRC software are from. If in doubt, replace the JAR files from the SLE installation, which can be found in *<SLE install dir>/lib* (for example: */opt/ellacoya/lib*).

Related Documentation

- SRC Script Services for DPI on page 13
- Service Activation and Deactivation in a DPI Environment on page 12
- Configuring Subscriptions to DPI Services on page 32
- Adding a Service Scope on page 28
- Configuring the Script Service on page 30

Configuring the Script Service

To configure the script service, you provide parameter substitutions with the values of the configuration parameters that are in the service definitions. To do so:

1. Configure parameters for the script service. See *Customizing Service Implementations* or *Configuring Script Services (C-Web Interface)*
2. Configure all the parameters listed in Table 9 on page 30.

Table 9: Parameter Definitions for DPI Services

Parameter Name	Description
dpi_sle_connection_servers	IP address or hostname of the Ellacoya SLE server. If there is more than one server, separate entries with a comma. Enter the SLE addresses in priority order, with the primary server first. When the script service makes a connection to an SLE, it tries the addresses in the order that you entered them until it succeeds in connecting. If the connection is broken, the script service attempts to connect again, beginning with the first address in the list. Once the script connects to an SLE, it does not test the connection to the primary server and fail back to the primary server if the server recovers from a failure.
dpi_sle_subscriber_nameattr	Attribute that is used to identify a subscriber. This parameter instructs the SLE how to identify the subscriber. Usually subscribers are identified by their login name. You can also use the MAC address, primary user, NAS-Port ID, or interface name. The values for the parameter definition are: • loginname • primaryusername • macaddr • interfacename • nasportid
dpi_sle_subscriber_ipv6supported	Specifies whether or not IPv6 is supported for the DPI service. Must currently be false because IPv6 is not supported on Ellacoya systems.
dpi_sle_connection_domainName	Name of the domain that contains the service bundles for this service. The Ellacoya software configures service bundles by domain. This parameter value should match the domain that is configured in the Ellacoya Service Creation Manager.
dpi_sle_connection_username	Name used for authentication with the SLE. It must match the username that is configured in the Ellacoya Service Creation Manager.
dpi_sle_connection_password	Password used for authentication with the SLE.

Table 9: Parameter Definitions for DPI Services (continued)

Parameter Name	Description
dpi_sle_accountingmgr_storesize	Size limit of usage and accounting data that is stored. (A script services loads usage data from the SLE.) All script services in a virtual router share the same store. When the SAE requests usage data more slowly than the script service loads it from the SLE, the data accumulates in the store. When the store is filled, the script service suspends loading usage data from the SLE. However, as the SAE requests usage data, space in the store is freed, allowing the script service to resume loading data from the SLE.
dpi_sle_activate_inoneshot	The values for the parameter are true or false. The default value is true. If true, when a DPI script service is activated as an AOL subscription during subscriber login, the script service finds all AOL subscriptions and persistent sessions of the subscriber. The script service assumes that the SAE will activate all the subscriptions and sessions, and it starts an SLE subscriber session with a service offer that corresponds to the combination of the AOL subscriptions and persistent sessions. This optimization avoids repeatedly changing the subscriber's service offer while multiple AOL subscription or persistent sessions are activated one by one. Set this parameter to false if an AOL subscription or persistent session is not guaranteed to be activated during subscriber login; for example, an authorization plug-in or mutex group could prevent the activation of the services.
dpi_sle_subscriber_prefix	Set this parameter to match the subscriber prefix configured in the SLE. The DPI script service adds this prefix to subscriber IDs used in SLE subscriber sessions that the DPI script service starts. The default value is BBIP, which is also the default value used in the SLE. You can change the SLE subscriber prefix in the SubscriberPrefix field in the Accounting Service window of the SLE Configuration Console.

Related Documentation

- SRC Script Services for DPI on page 13
- Service Activation and Deactivation in a DPI Environment on page 12
- Configuring Subscriptions to DPI Services on page 32
- Adding a Service Scope on page 28
- Creating a DPI Script Service on page 28

Configuring a Virtual Router Object for DPI

You must create a virtual router object in the router to which the subscriber logs in, and attach the DPI service scope to the virtual router.

Each time a DPI script service is invoked by an activation, deactivation, or interim accounting request, the script service checks the service involved to see if the configuration in the service's parameter substitutions is different or newer than the parameters in the script service's configuration. If so, the script service instance updates its configuration according to the service. This functionality means you can change the configuration of the script service by updating the parameter substitutions in the service.

If you want to use the same DPI configuration throughout your network, make sure that all DPI script services have the same configuration. If you have several SLEs in your network that are responsible for the traffic coming from different sets of routers, you must ensure that the DPI script service gets the appropriate configuration on a per router basis. You can accomplish this by putting the script services into a scope and attaching the scope to the router for an SLE. Every service in the scope must have the same configuration so that the correct SLE handles all subscribers on routers that use the scope.

To attach a service scope to a virtual router configuration, configure the name of the DPI service scope for the DPI for the Scope field option of the virtual router configuration. See *SRC PE Network Guide*.

Related Documentation

- Integration of the SRC Software and the Ellacoya DPI Platform on page 8
- Subscriber Login and Logout in a DPI Environment on page 12
- Service Activation and Deactivation in a DPI Environment on page 12
- SRC Script Services for DPI on page 13
- Loading the Sample Data for the DPI on page 27

Configuring Subscriptions to DPI Services

You need to configure subscriptions to the DPI services. You can set up the subscriptions to activate immediately on login.

Related Documentation

- Service Activation and Deactivation in a DPI Environment on page 12
- Subscriber Login and Logout in a DPI Environment on page 12
- SRC Script Services for DPI on page 13
- Configuring the Script Service on page 30
- Creating a DPI Script Service on page 28
- *SRC PE Subscribers and Subscriptions Guide*

CHAPTER 5

Configuration Tasks for the Ellacoya DPI Platform for Integration

Provisioning the IPSCS

For integration with the SRC software, the IPSCS must be provisioned in a specific way. All IPSCS service objects, including application signatures, service elements, TAPs, service bundles, and service offers, must be created.

Before you introduce a new service that will be implemented by a DPI script service, you must create the service bundle and service offers. Before you remove a service, you must deactivate all sessions for the service before removing the corresponding service offers and bundles from the IPSCS.

See "Integration of the SRC Software and the Ellacoya DPI Platform" on page 8.

Service Bundles

Enable usage collection on service bundles by checking Collect Start/Stop Usage Data and the Collect Transmit and Receive Byte Counts checkboxes in Service Creation Manager.

Every DPI script service must have a corresponding ISPCS service bundle. To match the service bundle with the script service, the service bundle must have the same name as the DPI script service, but be prefixed by SDX_. For example, if the script service is called dpiService1, the service bundle must be called SDX_dpiService1.

Service Offers

Enable usage collection on service offers by checking the Enable Usage Collection checkbox in Service Creation Manager.

For every combination of implemented services that could be active at the same time, there must be a service offer with the name prefixed by " SDX_" configured in the IPSCS. The service offer must contain the service bundles that correspond to the services.

When a script service activates or deactivates a service session, it checks the list of services that will be active after the activation or deactivation. The script service then searches for a service offer with the right set of service bundles. The script service keeps a local cache of all service offer names and which bundles they contain. If the cache does not contain a service offer with the right set of bundles, the script service reloads the service offers from the SLE. If necessary, the script service queries the SLE for all service offers in the configured domain. If the appropriate offer is not found after a reload, the service activation or deactivation fails with an exception.

Traffic-Accounting Profiles

For every service bundle, there must be a TAP with the same name as the service bundle.

Related Documentation

- Overview of Application-Level Session Tracking and QoS Control on page 7
- Configuring the SLE on page 34
- Synchronizing System Clocks on page 37

Configuring the SLE

In Service Creation Manager, configure the following:

1. Create passive authentication configuration, and enable subscriber learning.
2. Create a switch configuration, and apply the passive authentication configuration to the switch configuration.
3. Create a switch that corresponds to your e30 Switch, and apply the switch configuration *to the switch.

You also need to enable accounting in the SLE. To do so, log in to the SLE Configuration Console console, and check the AccountingEnabled service attribute in the AccountingService window.

Related Documentation

- Provisioning the IPSCS on page 33
- Integration of the SRC Software and the Ellacoya DPI Platform on page 8
- Configuring a Virtual Router Object for DPI on page 32

PART 4

Administration

CHAPTER 6

Integrating the SRC Software and the DPI Software

Synchronizing System Clocks

The system clock in the e30 Switch must keep synchronized with the SAE. Make sure that the system clock of the Ellacoya switch does not use daylight savings mode.

Related Documentation

- Provisioning the IPSCS on page 33
- Integration of the SRC Software and the Ellacoya DPI Platform on page 8