LogLogic
Web Services API
Implementation Guide

Software Release: 5.1
Document release: December 2010
Part No: LL71000-00E05100000

This manual supports LogLogic software release 4.9 and later releases until replaced by a newer edition.
Proprietary Information
This document contains proprietary and confidential information of LogLogic, Inc. and its licensors. In accordance with the license, this document may not be copied, disclosed, modified, transmitted, or translated except as permitted in writing by LogLogic, Inc.

Trademarks
"LogLogic" and the LogLogic logo are trademarks of LogLogic, Inc. in the United States and/or foreign countries. All other company product names are trademarks or registered trademarks of their respective owners.

Notice
The information contained in this document is subject to change at any time without notice. All warranties with respect to the software and accompanying documentation are set out exclusively in the Software License Agreement or in the Product Purchase Agreement that covers the documentation.
## Contents

### Preface:
- About This Guide
  - Audience .................................................. 9
  - Related Documents ....................................... 9
  - Technical Support ........................................ 10
  - Documentation Support .................................. 10
  - Conventions ............................................. 11

### Part I: Web Services API Overview and Setup

#### Chapter 1: Getting Started
- Overview of the Web Services API .................................. 15
  - Administration Service ...................................... 15
  - Report and Search Service .................................... 15
  - Requirements ............................................... 15

### Part II: Administration Services

#### Chapter 2: Alert Service Operations
- Overview ..................................................... 19
- Implementation Guidelines ................................... 20
- Alert Service Operation Definitions .......................... 20
  - createAlert Operation ....................................... 21
  - createAlertRemote Operation ............................... 22
  - readAlert Operation ....................................... 22
  - readAlertRemote Operation ............................... 23
  - updateAlert Operation ..................................... 23
  - updateAlertRemote Operation ............................ 24
  - deleteAlert Operation ...................................... 24
  - deleteAlertRemote Operation ............................ 25
  - getList Operation ......................................... 25
  - getListRemote Operation ................................. 25
  - alertResponse Type ....................................... 26
  - getAlertHistory Operation ................................ 27
  - acknowledgeAlertHistoryByKey Operation ............... 27
  - removeAlertHistoryByKey Operation ....................... 28
  - removeAlertHistory Operation ........................... 28
  - alertHistoryResponse Type ............................... 29
- Status Codes .................................................. 30
- Common Request Parameters .................................. 30
## Contents

Alert-Specific Request Parameters ............................................... 32  
Adaptive Baseline Alert .......................................................... 33  
Cisco PIX Message Alert .......................................................... 34  
Message Volume Alert .............................................................. 35  
Network Policy Alert ............................................................... 36  
Pre-Defined Search Filter Alert ................................................... 39  
Ratio Based Alert ................................................................. 40  
System Alert .............................................................................. 42  
VPN Connections Alert .............................................................. 45  
VPN Messages Alert ................................................................. 46  
VPN Statistics Alert ................................................................. 47

### Chapter 3: Device Service Operations

Overview ................................................................................. 51  
Implementation Guidelines ......................................................... 51  
Device Service Operation Descriptions ........................................ 52  
  createDevice Operation ........................................................... 52  
  createDeviceRemote Operation ................................................... 53  
  readDevice Operation ............................................................. 53  
  readDeviceRemote Operation ..................................................... 54  
  updateDevice Operation .......................................................... 54  
  updateDeviceRemote Operation ................................................... 54  
  deleteDevice Operation ........................................................... 55  
  deleteDeviceRemote Operation ..................................................... 55  
  getList Operation .................................................................. 55  
  getListRemote Operation ......................................................... 56  
  deviceResponse Type .............................................................. 56  
Status Codes ............................................................................. 57  
Common Request Parameters ..................................................... 57

### Chapter 4: Device Group Service Operations

Overview ................................................................................. 59  
Implementation Guidelines ......................................................... 59  
Device Group Service Operation Descriptions ................................... 59  
  createDeviceGroup Operation ...................................................... 61  
  createDeviceGroupRemote Operation .............................................. 61  
  addDevices Operation ............................................................... 61  
  addDevicesRemote Operation ....................................................... 62  
  removeDevices Operation .......................................................... 62  
  removeDevicesRemote Operation ................................................. 63  
  readDeviceGroup Operation ....................................................... 63  
  readDeviceGroupRemote Operation .............................................. 64
Get Custom Report List Operation ........................................ 107
Get Custom Report List Request Parameter .......................... 108
Get Custom Report List Response Attributes ........................ 108
Get Report Meta Info Operation ......................................... 108
Get Report Meta Info Request Parameter ............................ 108
Get Report Meta Info Response Attributes .......................... 109
Get Detail Report Meta Info Operation ................................. 109
Get Detail Report Meta Info Response Attributes .................... 110

Part IV: Utilities Services and Appendices

Chapter 10: Authentication Service API

GetAuthenticationToken Operation ................................. 113
Request Parameters ...................................................... 113
Response ................................................................. 113
Status Codes ............................................................ 113

Appendix A: Reference Lists

Alert Types .............................................................. 115
Message Areas .......................................................... 116
User Administration Privileges ........................................ 116
PREFACE:

About This Guide

The LogLogic™ Appliance-based solution lets you capture and manage log data from all types of sources in your enterprise. LogLogic Appliances install within 10 minutes and begin collecting and aggregating data from connected log sources immediately.

This document describes the LogLogic Web Services APIs that enable you to interface with the LogLogic Appliances to manage reports and alerts as well as to perform searches.

Audience

This document is intended for LogLogic customers who are developing a Web Services application to interface with the LogLogic solution. Users should be familiar with developing applications using Web Services APIs as well as either developing in Java or Perl.

Related Documents

The LogLogic documentation is available on the Solutions CD or on the LogLogic Technical Support website – http://www.loglogic.com/services/support. The documentation includes Portable Document Format (PDF) files and Online Help accessible from the LogLogic user interface.

To read the PDF documentation, you need a PDF file viewer such as Adobe Acrobat Reader. You can download the Adobe Acrobat Reader at http://www.adobe.com.

The following documents contain additional information about the LogLogic Appliances:

- **LogLogic Release Notes** — Provides information specific to the release including product information, new features and functionality, resolved issues, known issues and any late-breaking information. Check the LogLogic support web site periodically for further updates.

- **LogLogic Upgrade Guide** — Describes how to upgrade the LogLogic Appliance software.

- **LogLogic Quick Start Guide** — Describes how to get started with your LogLogic Appliance. In addition, the guide includes details about the Appliance hardware.

- **LogLogic LX 2010N Quick Start Guide** — Describes how to get started with the LogLogic LX 2010N NEBS-compliant Appliance, and includes details about the Appliance hardware.

- **LogLogic Administration Guide** — Describes how to administer the LogLogic solution including managing users, managing log data storage, and managing new log sources (devices).

- **LogLogic User Guide** — Describes how to use the LogLogic solution, including for managing reports, managing alerts, and performing searches.
Technical Support

LogLogic is committed to the success of our customers and to ensuring our products improve customers’ ability to maintain secure, reliable networks. Although LogLogic products are easy to use and maintain, occasional assistance might be necessary. LogLogic provides timely and comprehensive customer support and technical assistance from highly knowledgeable, experienced engineers who can help you maximize the performance of your LogLogic Appliances.

To reach the LogLogic Support team:

**Telephone:**
Toll Free — 1-800-957-LOGS
Local — 1-408-834-7480
Europe, Middle East, Africa (EMEA) or Asia Pacific (APAC): +44 (0) 207 1170075 or +44 (0) 8000 669970

**Email:** support@loglogic.com

**Support Website:** [http://www.loglogic.com/services/support](http://www.loglogic.com/services/support)

When contacting Customer Support, be prepared to provide the following information:

- Your name, e-mail address, phone number, and fax number
- Your company name and company address
- Your machine type and release version
- Serial number located on the back of the Appliance or the eth0 MAC address
- A description of the problem and the content of pertinent error messages (if any)

Documentation Support

Your feedback on LogLogic documentation is important to us. Send e-mail to DocComments@loglogic.com if you have questions or comments. Your comments will be reviewed and addressed by the LogLogic technical writing team.

In your e-mail message, please indicate the software name and version you are using, as well as the title and document date of your documentation.
Conventions

LogLogic documentation uses the following conventions:

**Caution:** Highlights important situations that could potentially damage data or cause system failure.

**IMPORTANT!** Highlights key considerations to keep in mind.

**Note:** Provides additional information that is useful but not always essential.

**Tip:** Highlights guidelines and helpful hints.

This guide also uses the following conventions to highlight code and command-line elements:

- **Monospace** is used for programming elements (such as code fragments, objects, methods, parameters, and HTML tags) and system elements (such as file names, directories, paths, and URLs).

- **Monospace bold** is used to distinguish system prompts or screen output from user responses, as in this example:
  
  username: **system**
  
  home directory: *home\app*

- **Monospace italic** is used for placeholders, which are general names that you replace with names specific to your site, as in this example:
  
  `LogLogic_home_directory\upgrade`

- Straight brackets signal options in command-line syntax.
  
  `ls [-AabCcdFfgilmnopqRrstux1] [-X attr] [path ...]`
Part I: Web Services API Overview and Setup
### Overview of the Web Services API

The API is XML Web Services-based to provide a standardize API transport.

LogLogic provides two main Services:

- **Administration Service** - for managing administrative tasks
- **Report and Search Service** - for accessing report and search capabilities

### Administration Service

The LogLogic Administration Service API lets you manage LogLogic Appliances via the following operations:

- Alert Service Operations .......................................................... 19
- Device Service Operations ......................................................... 51
- Device Group Service Operations ............................................... 59
- Message Routing Service Operations ....................................... 69
- Search Filter Service Operations .............................................. 79
- User Administration Service Operations .................................. 87

### Report and Search Service

The LogLogic Search Service API lets you manage reports and search queries on LogLogic Appliances. Managing report and search queues includes running reports, viewing reports, and searching reports to return specific data.

- Report and Search Service ....................................................... 99

### Requirements

To use the LogLogic Web Services API, you need:
Getting Started: Requirements

- a software tool to integrate your applications with the LogLogic Web Services API (such as Apache Axis (for Java and C++ clients) or SOAP::Lite (for Perl clients)).
- a software tool to create the client side API code (such as wsdl2java or wsdl2perl).
- LogLogic WSDL
- a software tool to generate client stubs
- LogLogic Appliance UserID with “access web services” enabled

Port 443 is the required port on the LogLogic Appliance for accessing the Web Services API calls.
Part II: Administration Services
CHAPTER 2:

Alert Service Operations

The Alert Service operations let you manage alerts in the LogLogic Appliance.

- **Overview** ................................................................. 19
- **Implementation Guidelines** .......................................... 20
- **Alert Service Operation Definitions** .............................. 20
- **Status Codes** ............................................................. 30
- **Common Request Parameters** ....................................... 30
- **Alert-Specific Request Parameters** ............................... 32

**Overview**

The Alert Service operations let you create, read, update, and delete alerts as well as view all alerts in the LogLogic Appliance.

Using the `CreateAlert` and `UpdateAlert` operations, you can define and update rules to detect unusual traffic on your network or detect Appliance system anomalies. Alerts can be configured to generate SNMP events and/or send an email notification when the alert rule is triggered for a specific type of alert.

The alert types are Adapter Baseline, Cisco Pix Messages, Message Volume, Network Policy, Pre-defined Search Filter, Ratio Based, System, VPN Connections, VPN Messages, and VPN Statistics. For more information on supported alerts, see Alert Types on page 115 or the online help for each specific alert. In the LogLogic Appliance, to view the user interface implementation navigate to Alerts.

When creating (`CreateAlert`) or updating (`UpdateAlert`) an alert, you must specify a value for the `alertRules` Common Request Parameter. The `alertRules` value is used to define alert rules for a specific alert.

Figure 1 on page 20 provides a graphical view of the Common and Alert-Specific Parameters. The example displays an implementation of the `createAlert` operation specifying the VPN Messages alert type for the `alertRules`.
Implementation Guidelines

The following are general implementation guidelines for the Alert Service operations:

- A set of **Common Request Parameters** are required for each Alert Service operation.

- The `createAlert` Operation, `createAlertRemote` Operation, `updateAlert` Operation, and `updateAlertRemote` Operation require that you specify Common and Alert-Specific Request Parameters. Alert-Specific Request Parameters are specified using the `alertRules` Common Request Parameter.

- Alert Rules, defined in the `alertRules` Common Request Parameter, are specified as a string in the format:

  ```
  /parameter1/valueA/parameter2/valueD/valueE/
  ```

  For example, a rule for the Network Policy alert is:

  ```
  "fewerThan/100//moreThan/10//alertFilter/False//policyAction/Accept//srcIPMin/10.1.2.3//srcIPMax/255.255.255.255//srcPortMin/0//srcPortMax/100//destIPMin/10.1.1.123//destIPMax/255.255.255.255//destPortMin/0//destPortMax/100//protocol/all"
  ```

  For specific usage rules, see **Common Request Parameters** on page 30 and **Alert-Specific Request Parameters** on page 32.

Alert Service Operation Definitions

There are two kinds of operations:

- **local** - operation is performed on the local Appliance itself
- **remote** - operations (names ending with Remote) performed on a specified remote Appliance

The Alert Service Operations are as follows:
createAlert Operation

The createAlert operation lets you create new alert rules in the LogLogic Appliance.

When using createAlert, you must specify:

- **Common Request Parameters** on page 30
- **Alert-Specific Request Parameters** on page 32 (includes alertRules format description)

### Request Parameters

- authToken
- alertType
- name
- desc
- priorityName
- enabled
- deviceNames
- usernames
- trapIds
- resetTime
- trackIndividualDevice
- alertRules
- snmpOId

For more information on each Common Request Parameter, see **Common Request Parameters** on page 30.

### Response

- alertResponse (see **alertResponse Type** on page 26)

**Note:** The Alert-Specific Request parameters specified in the alertRules parameter are also returned. The response depends on the alert type used.
Example

To create a VPN Connection Alert named MyAlertName:

```plaintext
createAlert authstr "VPN Connection Alert" "MyAlertName" "VPN Connection Alert Description" "low" "10.1.2.3_04" "admin" "400" "yes" "VPNUser/LogLogicUser//VPNGroup/LogLogicGroup//disconnectReason/is denied access" ""
```

createAlertRemote Operation

The `createAlertRemote` operation lets you create new alert rules on a managed LogLogic Appliance from a Management Station.

When using `createAlertRemote`, you must specify:

- **Common Request Parameters** on page 30
- **Alert-Specific Request Parameters** on page 32

Request Parameters

- `authToken`, `applianceIP`, `alertType`, `name`, `desc`, `priorityName`, `enabled`, `deviceNames`, `usernames`, `trapIds`, `resetTime`, `trackIndividualDevice`, `alertRules`, `snmpOId`

For more information on each Common Request Parameter, see **Common Request Parameters** on page 30.

Response

- `alertResponse` (see **alertResponse Type** on page 26)

**Note:** The Alert-Specific Request parameters specified in the `alertRules` parameter are also returned. The response depends on the alert type used.

Example

To create a VPN Connection Alert named MyAlertName on remote Appliance 1.2.20.100:

```plaintext
createAlertRemote authstr 1.2.20.100 "VPN Connection Alert" "MyAlertName" "VPN Connection Alert Description" "low" "10.1.2.3_04" "admin" "400" "yes" "VPNUser/LogLogicUser//VPNGroup/LogLogicGroup//disconnectReason/is denied access" ""
```

readAlert Operation

The `readAlert` operation lets you view the details of existing alerts in the LogLogic Appliance.

Request Parameters

- `authToken`, `alertName`
Alert Service Operations : Alert Service Operation Definitions

Response
alertResponse (see alertResponse Type on page 26)

Example
To view the details of the MyAlertName alert:
readAlert authstr "MyAlertName"

readAlertRemote Operation

The readAlertRemote operation lets you view the details of existing alerts on a managed LogLogic Appliance from a Management Station.

Request Parameters
authToken, applianceIP, alertName

Response
alertResponse (see alertResponse Type on page 26)

Example
To view the details of the MyAlertName alert on remote Appliance 1.2.20.100:
readAlertRemote authstr 1.2.20.100 "MyAlertName"

updateAlert Operation

The updateAlert operation lets you update existing alerts in the LogLogic Appliance.

When using updateAlert, you must specify:
- Common Request Parameters on page 30
- Alert-Specific Request Parameters on page 32

Request Parameters
authToken, alertType, name, desc, priorityName, enabled, deviceNames, usernames, trapIds, resetTime, trackIndividualDevice, alertRules, snmpOId, changeNameTo

Response
alertResponse (see alertResponse Type on page 26)

Note: The Alert-Specific Request parameters specified in the alertRules parameter are also returned. The response depends on the alert type used.

Example
To update a VPN Connection Alert named MyAlertName to be named NewAlertName with the priority set at high:
updateAlert authstr "VPN Connection Alert" "MyAlertName" "VPN Connection Alert Description" "low" "10.1.2.3_04" "admin" "400" "yes" "VPNUser/LogLogicUser//VPNGroup/LogLogicGroup///disconnectReason/is denied access" “NewAlertName”

**updateAlertRemote Operation**

The updateAlertRemote operation lets you update existing alerts on a managed LogLogic Appliance from a Management Station.

When using updateAlertRemote, you must specify:

- **Common Request Parameters** on page 30
- **Alert-Specific Request Parameters** on page 32

**Request Parameters**

- authToken, applianceIP, alertType, name, desc, priorityName, enabled, deviceNames, usernames, trapIds, resetTime, trackIndividualDevice, alertRules, snmpOId, changeNameTo

**Response**

- alertResponse (see alertResponse Type on page 26)

**Note:** The Alert-Specific Request parameters specified in the alertRules parameter are also returned. The response depends on the alert type used.

**Example**

To update a VPN Connection Alert named MyAlertName to be named NewAlertName with the priority set at high, on remote Appliance 1.2.20.100:

updateAlert authstr 1.2.20.100 "VPN Connection Alert" "MyAlertName" "VPN Connection Alert Description" "low" "10.1.2.3_04" "admin" "400" "yes" "VPNUser/LogLogicUser//VPNGroup/LogLogicGroup///disconnectReason/is denied access" “NewAlertName”

**deleteAlert Operation**

The deleteAlert operation lets you delete existing alerts in the LogLogic Appliance.

**Request Parameters**

- authToken, alertName

**Response**

- alertResponse (see alertResponse Type on page 26)

**Example**

To delete the alert named MyAlertName:

deleteAlert authstr “MyAlertName”
deleteAlertRemote Operation

The deleteAlertRemote operation lets you delete existing alerts on a managed LogLogic Appliance from a Management Station.

Request Parameters
authToken, applianceIP, alertName

Response
alertResponse (see alertResponse Type on page 26)

Example
To delete the alert named MyAlertName on remote Appliance 1.2.20.100:
deleteAlertRemote authstr 1.2.20.100 "MyAlertName"

ggetList Operation

The getList operation lets you retrieve the complete list of all alerts currently defined in the LogLogic Appliance.

Request Parameter
authToken

Response
If resultCount is greater than 0 and statusCode is 2000 (successful), the response returns a list of all alerts (total number indicated by resultCount) currently configured in the remote LogLogic Appliance.

If resultCount is 0 and statusCode is not 2000 (successful), an error is returned in statusMessage.

Example
To retrieve a list of all alerts defined in the Appliance:
ggetList authstr

ggetListRemote Operation

The getListRemote operation lets you retrieve the complete list of all alerts currently defined on a managed LogLogic Appliance from a Management Station.

Request Parameter
authToken, applianceIP
Response

If resultCount is greater than 0 and statusCode is 2000 (successful), the response returns a list of all alerts (total number indicated by resultCount) currently configured in the remote LogLogic Appliance.

If resultCount is 0 and statusCode is not 2000 (successful), an error is returned in statusMessage.

Example

To retrieve a list of all alerts defined in remote Appliance 1.2.20.100:

getListRemote authstr 1.2.20.100

alertResponse Type

alertResponse is returned for all alert operations except getList and getListRemote.

alertResponse always contains the following common elements:

- authToken
- resultCount
- statusCode
- statusMessage
- summaryOnly

If resultCount is 1 and statusCode is 2000 (successful), the resultSet element is included after resultCount listing the following Alert details:

- alertType
- name
- desc
- priorityName
- enabled
- deviceNames
- usernames
- trapIds
- resetTime
- trackIndividualDevice
- alertRules
- snmpOId (createAlert and createAlertRemote only)
- changeNameTo (updateAlert and updateAlertRemote only)

Note: The Alert-Specific Request parameters specified in the alertRules parameter are also returned. The response depends on the alert type used.
If resultCount is 0 and statusCode is not 2000 (successful), an error is returned in statusMessage.

**getAlertHistory Operation**

The getAlertHistory operation lets you retrieve all alert logs currently on the LogLogic Appliance. You may also use the filters to narrow down the result list.

**Request Parameter**

authToken, applianceIP, filters

**Response**

alertHistoryResponse (see *alertHistoryResponse Type* on page 29)

If resultCount is greater than 0 and statusCode is 2000 (successful), the response returns a list of alert logs (total number indicated by resultCount) currently on the remote LogLogic Appliance.

If resultCount is 0 and statusCode is not 2000 (successful), an error is returned in statusMessage.

If the appliance is a management station, you may get the aggregated alert logs by specifying “All” in applianceIP.

**Example**

To retrieve the list of all high priority alert logs on Appliance 1.2.20.100:

getAlertHistory authstr 1.2.20.100 /Priority=/High/

To retrieve all alert logs on Appliance 1.2.20.100 with empty filters:

getAlertHistory authstr 1.2.20.100 ""

To retrieve all new alert logs on Appliance 1.2.20.100:

getAlertHistory authstr 1.2.20.100 /Type=/Unacknowledged/

**acknowledgeAlertHistoryByKey Operation**

The acknowledgeAlertHistoryByKey operation lets you acknowledge a list of alert logs currently on the LogLogic Appliance. You have to obtain the key list from getAlertHistory operation.

If you have obtained a list from the aggregated alert logs, you must specify “All” in applianceIP. Otherwise, it will be processed in single appliance mode and only alert logs on current appliance will be affected.

If resultCount is not 0, it means there are invalid keys in your key list. You can browse the returned resultSet for these keys.

**Request Parameter**

authToken, applianceIP, keyList
Response
alertHistoryResponse (see alertHistoryResponse Type on page 29)

Example
To acknowledge one or more alert logs on the Appliance 1.2.20.100:
acknowledgeAlertHistoryByKey authstr 1.2.20.100  keyList

removeAlertHistoryByKey Operation

The removeAlertHistoryByKey operation lets you remove a list of alert logs currently on the LogLogic Appliance. You have to obtain the key list from getAlertHistory operation.

If you have obtained a list from the aggregated alert logs, you must specify “All” in applianceIP. Otherwise, it will be processed in single Appliance mode and only alert logs on the current Appliance will be affected.

If resultCount is not 0, it means there are invalid keys in your key list. You can browse the returned resultSet for these keys.

Request Parameter
authToken, applianceIP, keyList

Response
alertHistoryResponse (see alertHistoryResponse Type on page 29)

Example
To remove one or more alert logs on the Appliance 1.2.20.100:
removeAlertHistoryByKey authstr 1.2.20.100  keyList

removeAlertHistory Operation

The removeAlertHistory operation lets you remove all alert logs currently on the LogLogic Appliance. You may also use the filters to narrow down the list.

Request Parameter
authToken, applianceIP, filters

Response
alertHistoryResponse (see alertHistoryResponse Type on page 29)

Example
To remove all medium priority alert logs on Appliance 1.2.20.100:
removeAlertHistory authstr 1.2.20.100 /Priority/=/Medium/
To remove all alert logs on Appliance 1.2.20.100 with empty filters:
removeAlertHistory authstr 1.2.20.100 ""

To remove all acknowledged alert logs on Appliance 1.2.20.100:
removeAlertHistoryRemote authstr 1.2.20.100 /Type=/Acknowledged/

**alertHistoryResponse Type**

alertHistoryResponse is returned for all alert history operations.
alertHistoryResponse always contains the following common elements:
- authToken
- resultCount
- statusCode
- statusMessage
- summaryOnly

In `getAlertHistory` operation, if `resultCount` is greater then 0 and `statusCode` is 2000 (successful), the `resultSet` element that holds alert logs is included after `resultCount`.

The attributes of an alert log are:

<table>
<thead>
<tr>
<th>Attributes</th>
<th>Description</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Key</td>
<td>Used to identify a unique log. It can be used in <code>acknowledgeAlertHistoryByKey</code> or <code>removeAlertHistoryByKey</code> operations. The user can obtain this by calling <code>getAlertHistory</code>.</td>
<td>string</td>
</tr>
<tr>
<td>time</td>
<td>Corresponds to the “Time” field on Alert Viewer page.</td>
<td>date</td>
</tr>
<tr>
<td>sourceIp</td>
<td>Corresponds to the “Source IP” field on Alert Viewer page.</td>
<td>string</td>
</tr>
<tr>
<td>msgType</td>
<td>Corresponds to the “Type” field on Alert Viewer page.</td>
<td>string</td>
</tr>
<tr>
<td>notifyType</td>
<td>Possible values are 1, 2, 3 for email alert, snmp alert, and no notification, respectively.</td>
<td>number</td>
</tr>
<tr>
<td>emailRcpt</td>
<td>Corresponds to the “Alert Destination” field on Alert Viewer page when email alert is used.</td>
<td>string</td>
</tr>
<tr>
<td>trapReceiver</td>
<td>Corresponds to the “Alert Destination” field on Alert Viewer page when snmp alert is used.</td>
<td>string</td>
</tr>
<tr>
<td>message</td>
<td>The alert message body.</td>
<td>string</td>
</tr>
<tr>
<td>priority</td>
<td>Possible values are 0, 1, 2 for priority low, medium, and high, respectively.</td>
<td>number</td>
</tr>
<tr>
<td>ArchiveFlag</td>
<td>Possible values are 0, 1 for unacknowledged alert logs and acknowledged alert logs, respectively.</td>
<td>number</td>
</tr>
<tr>
<td>ApplianceIp</td>
<td>Corresponds to the “Appliance” field on Alert Viewer page. This will be visible on Alert Viewer page when you view alert logs from a Management Station.</td>
<td>string</td>
</tr>
</tbody>
</table>

If `resultCount` is 0 and `statusCode` is not 2000 (successful), an error is returned in `statusMessage`. 
Status Codes

The Status Codes are:

<table>
<thead>
<tr>
<th>Status Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>2000</td>
<td>Server success</td>
</tr>
<tr>
<td>4000</td>
<td>Unauthorized request</td>
</tr>
<tr>
<td>5000</td>
<td>Invalid parameter, getStatusMessage() contains detail information about the error</td>
</tr>
</tbody>
</table>

Common Request Parameters

A set of Common Request Parameters are required for each of the Alert Service operations. When using the CreateAlert or UpdateAlert operations, you must specify the alertRules parameter.

Common Request Parameters usage must follow several rules:

- You must specify a value for all Required Common Request Parameters.
- All Common Request parameters must be implemented in the order in which they appear in the Request Parameters section for each of the Alert Service operations.
- For Common Request Parameters, you must specify the value of the parameter only. Note that LogLogic expects the values in the order defined in this document.
- All values for Common Request Parameters must be enclosed by double quotation marks (“value”).

Table 1 Alert Service Common Request Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
<th>Values</th>
<th>Required</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>authToken</td>
<td>Token string returned from the authentication service or the “username/password”.</td>
<td>IP address of a managed Appliance. To specify an IP address, use the standard IP address format. For example: 10.1.2.3</td>
<td>yes</td>
<td>string</td>
</tr>
<tr>
<td>applianceIp</td>
<td>The managed Appliance on which you perform the operation. If the value is blank, it retrieves the Appliance IP address from the local Appliance. This parameter is available only for Management Station Appliances using operations with Remote in the name.</td>
<td>IP address of a managed Appliance. To specify an IP address, use the standard IP address format. For example: 10.1.2.3</td>
<td>yes</td>
<td>string</td>
</tr>
<tr>
<td>alertType</td>
<td>Type of alert, such as Network Policy Alert or System Alert.</td>
<td>For a list of alert types, see Alert Types on page 115.</td>
<td>yes</td>
<td>string</td>
</tr>
<tr>
<td>name</td>
<td>Name of the alert.</td>
<td>Any text up to 64 characters in length.</td>
<td>yes</td>
<td>string</td>
</tr>
<tr>
<td>desc</td>
<td>Description for the remote device.</td>
<td>Any text up to 64 characters in length.</td>
<td>optional</td>
<td>string</td>
</tr>
<tr>
<td>priorityName</td>
<td>Priority level of the alert.</td>
<td>Possible values: low, medium, and, high</td>
<td>yes</td>
<td>string</td>
</tr>
<tr>
<td>enabled</td>
<td>Determines if the alert is enabled or disabled. (Appears as enabled or disabled in returned value.)</td>
<td>Possible values: yes — enabled no — disabled The default is no.</td>
<td>yes</td>
<td>string</td>
</tr>
</tbody>
</table>
Table 1  Alert Service Common Request Parameters (Continued)

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
<th>Values</th>
<th>Required</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>deviceNames</td>
<td>List of devices. Valid entries contain one or more devices and/or device groups. To see a list of all available devices and device groups, use the Devices tab in the LogLogic Appliance user interface. To access the Devices tab, click Alerts &gt; Manage Alerts, click the Add New button, select an alert type, and then click the Devices tab.</td>
<td>List of valid devices and/or groups. Use a forward slash (/) as a delimiter for multiple entries. For example: 10.1.1.1/10.1.1.7 If a device has a forward slash (/) in the name, such as HP/UX or IBM i5/OS, you must replace the forward slash with %2F. (The F is case-sensitive.) For example: HP%2FUX</td>
<td>yes</td>
<td>string</td>
</tr>
<tr>
<td>usernames</td>
<td>User names for the alerts. Specify a single user or a user group.</td>
<td>Use a forward slash (/) as a delimiter for multiple entries. For example: user1/usergroup7</td>
<td>yes</td>
<td>string</td>
</tr>
<tr>
<td>trapIds</td>
<td>Trap name or IP Address to send the SNMP messages when the alert is triggered.</td>
<td>Use a forward slash (/) as a delimiter for multiple entries. For example: trap1/trap2/trap3</td>
<td>yes</td>
<td>string</td>
</tr>
<tr>
<td>resetTime</td>
<td>Time to wait between alerts that are generated. The Appliance does not issue an additional alert of the same type until the resetTime elapses.</td>
<td>Any positive integer. The value is in seconds. For example, the value 120 represents two minutes.</td>
<td>yes</td>
<td>number</td>
</tr>
<tr>
<td>trackIndividualDevice</td>
<td>Enables or disables individual device tracking.</td>
<td>Possible values: yes—enabled no — disabled The default is no.</td>
<td>yes</td>
<td>string</td>
</tr>
<tr>
<td>alertRules</td>
<td>Alert rule specific to the alert type.</td>
<td>See Alert-Specific Request Parameters on page 32 for a list of specific alert rules for each alert type.</td>
<td>yes</td>
<td>string</td>
</tr>
<tr>
<td>snmpOId</td>
<td>Specifies an SNMP OID to identify the originator of this alert.</td>
<td>Any valid SNMP OID</td>
<td>no</td>
<td>string</td>
</tr>
<tr>
<td>changeNameTo</td>
<td>New name of the alert. If empty, the object name is unchanged.</td>
<td>Any text up to 64 characters in length.</td>
<td>yes</td>
<td>string</td>
</tr>
</tbody>
</table>
Alert Service Operations: Alert-Specific Request Parameters

### Alert-Specific Request Parameters

You must specify the alert rules for each specific alert type you are managing. Alert Rules are defined using the `alertRules` Common Request Parameter. The following section contains alert-specific parameters for each of the alert types. The alert types are Adapter Baseline, Cisco Pix Messages, Message Volume, Network Policy, Pre-defined Search Filter, Ratio Based, System, VPN Connections, VPN Messages, and VPN Statistics.

Alert-Specific Request Parameters usage must follow several rules:

- All Alert-Specific Parameters can be implemented in any order. LogLogic recommends that you implement the alert rules in a consistent order and format to make managing the alert rules easier.

- Alert-Specific parameter values must include double quotation marks around the entire alert rule. For example:

  "param1/valueA//param2/valueC"

### Table 1  Alert Service Common Request Parameters (Continued)

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
<th>Values</th>
<th>Required</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>filters</td>
<td>List of expressions applied to narrow down affected alert logs. Filters are used only in <code>getAlertHistory</code> and <code>removeAlertHistory</code> operations. The priority and type filters work the same way as the drop-down boxes in alert viewer. For example, <code>/Priority/=/All_System/</code> returns all system alerts. The <code>New_Entry</code>, <code>Offset</code> and <code>Count</code> filters are used only in <code>getAlertHistory</code> operation. When <code>New_Entry</code> is set to true, it will return only new logs since the last call to <code>getAlertHistory</code> with <code>New_Entry</code> turned on. If this is the first time, then all alert logs will be returned. <code>Count</code> allows you to specify how many alert logs will be returned. The maximum count is 10,000. <code>Offset</code> allows you to specify the start offset. It is zero-based. Because you cannot return all alert logs at once if the total amount exceeds the maximum value. You have to use offset to get remaining alert logs.</td>
<td>Values must use the format: <code>/filtername/=/Value/</code> The valid filter names are &quot;Type&quot;, &quot;Priority&quot;, &quot;Offset&quot;, &quot;Count&quot; and &quot;New_Entry&quot;. &quot;Type&quot; supports &quot;Unacknowledged&quot;, &quot;Acknowledged&quot; and &quot;All&quot;. &quot;Priority&quot; supports &quot;High&quot;, &quot;Medium&quot;, &quot;Low&quot;, &quot;All_System&quot; and &quot;All&quot;. &quot;Count&quot; and &quot;Offset&quot; can not be negative. &quot;New_Entry&quot; supports &quot;True&quot; or &quot;False&quot;. If the filters are not present, the default is all types, all priorities, 0, 1000 and New_Entry set to false.</td>
<td>no</td>
<td>Array of string</td>
</tr>
<tr>
<td>keyList</td>
<td>A list consists of keys returned from <code>getAlertHistory</code> operation.</td>
<td>With <code>getAlertHistory</code> operation, you will retrieve a list of alert logs. The key value can be obtained from the key attribute of an alert log.</td>
<td>yes</td>
<td>Array of string</td>
</tr>
</tbody>
</table>
If the name of a device or the parameter value includes a forward slash (/), such as HP/UX, IBM i5/OS, or Accept/Total, you must replace the forward slash with %2F. (The F is case-sensitive.)

Examples: HP%2FUX, IBM i5%2FOS, or Accept%2FTotal

Use forward slash marks as delimiters when specifying alert rules. Use a single forward slash mark (/) as a delimiter to define multiple values for a parameter. Use double forward slash marks (//) as delimiters for parameters. For example:

```
param1/valueA/param2/valueC/valueD/param3/valueE
```

where param1, param2, and param3 are parameters and valueA, valueC, valueD, and valueE are values for param1, param2, and param3, respectively.

The example assigns the following name/value pairs:

```
param1 = valueA
param2 = valueC, valueD
param3 = valueE
```

**Adaptive Baseline Alert**

The Adaptive Baseline Alert lets you be notified if message rates fall above or below your average baseline range for a specified day and time of the week.

**Request Parameters**

fewerThan, moreThan

**Example**

“fewerThan/100//moreThan/10”

Table 2 lists the Adaptive Baseline Alert-specific parameters. You must include the parameters as inputs for the alertRules parameter.

**Table 2**  Adaptive Baseline Alert-specific parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
<th>Values</th>
<th>Required</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>fewerThan</td>
<td>Minimum percentage of messages that must be received within a time period (timeSpan parameter) before an alert is generated. If the number of messages drops below the fewerThan value, an alert is generated. The fewerThan and moreThan parameters make up the alert range.</td>
<td>Any positive integer between 1 and 100. The fewerThan value must be greater than the moreThan value.</td>
<td>yes</td>
<td>string</td>
</tr>
<tr>
<td>moreThan</td>
<td>Maximum percentage of messages that must be received within a time period (timeSpan parameter) before an alert is generated. If the number of messages exceeds the moreThan value, an alert is generated. The fewerThan and moreThan parameters make up the alert range.</td>
<td>Any positive integer between 1 and 100. The moreThan value must be less than the fewerThan value.</td>
<td>yes</td>
<td>string</td>
</tr>
</tbody>
</table>
Cisco PIX Message Alert

The Cisco PIX Messages alert allows for triggering on PIX message criticality, code, and message rate. Since this alert is specific to Cisco PIX messages, the alert device selection is limited to Cisco PIX devices.

Request Parameters

criticality, fewerThan, moreThan, messageCode, timeSpan

Example

"criticality/1//fewerThan/100//moreThan/10//messageCode/1-709006 //TimeSpan/60"

Table 3 lists the Cisco PIX Message Alert-specific parameters. You must include the parameters as inputs for the alertRules parameter.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
<th>Values</th>
<th>Required</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>criticality</td>
<td>Criticality for the alert. See your firewall documentation for details about the values in the list.</td>
<td>Enter a numeric value from the following list: 0 — emergency 1 — alert 2 — critical 3 — error 4 — warning 5 — notice 6 — information 7 — debug The default is 1.</td>
<td>yes</td>
<td>string</td>
</tr>
<tr>
<td>fewerThan</td>
<td>Minimum number of messages that must be received within a time period (timeSpan parameter) before an alert is generated. If the number of messages drops below the fewerThan value, an alert is generated. The fewerThan and moreThan parameters make up the alert range. You do not have to specify both fewerThan and moreThan.</td>
<td>Any positive integer between 1 and 100.</td>
<td>yes</td>
<td>string</td>
</tr>
<tr>
<td>moreThan</td>
<td>Maximum number of messages that can be received within a time period (timeSpan parameter) before an alert is generated. If the number of messages exceeds the moreThan value, an alert is generated. The fewerThan and moreThan parameters make up the alert range. You do not have to specify both fewerThan and moreThan.</td>
<td>Any positive integer between 1 and 100.</td>
<td>yes</td>
<td>string</td>
</tr>
</tbody>
</table>
Alert Service Operations : Alert-Specific Request Parameters

The Message Volume-based alert allows alerting when message volume falls below, or is above, preset messages-per-second thresholds. The alert applies to all devices.

The Message Volume-based alert also supports Zero Message Alert by using the timeSpan parameter. The time granularity of Zero Message Alert is in minutes, so the timeSpan has to be 60 seconds or greater. When timeSpan is present, you do not have to provide fewerThan and moreThan parameters as they will be ignored.

Table 4 lists the Message Volume Alert-specific parameters. You must include the parameters as inputs for the alertRules parameter.

**Request Parameters**

fewerThan, moreThan

**Example**

"fewerThan/100//moreThan/10"

---

Table 3  Cisco PIX Message Alert-specific parameter (Continued)

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
<th>Values</th>
<th>Required</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>messageCode</td>
<td>Message code for which an alert is generated.</td>
<td>Valid Cisco PIX message code. Message codes must match the criticality parameter. For example, if criticality is set to 3, you can specify any message code that starts with 3-. The default is 1-709006.</td>
<td>yes</td>
<td>string</td>
</tr>
<tr>
<td>timeSpan</td>
<td>Period of time that must be exceeded by the fewerThan and moreThan thresholds before an alert is triggered. If the fewerThan and moreThan thresholds are met for the specified timeSpan, an alert is generated.</td>
<td>Any positive integer. The value is in seconds. For example, the value 120 represents two minutes. The default is 60.</td>
<td>yes</td>
<td>string</td>
</tr>
</tbody>
</table>
Network Policy Alert


Table 4 lists the Network Policy Alert-specific parameters. You must include the parameters as inputs for the alertRules parameter.

Request Parameters

fewerThan, moreThan, alertFilter, policyAction, srcIPMin, srcIPMax, srcPortMin, srcPortMax, destIPMin, destIPMax, destPortMin, destPortMax, protocol

Example

"fewerThan/100//moreThan/10//alertFilter/False//policyAction/Accept//srcIPMin/10.1.2.3//srcIPMax/255.255.255.255//srcPortMin/0 //srcPortMax/100//destIPMin/10.1.1.123//destIPMax/255.255.255.255 //destPortMin/0//destPortMax/100//protocol/all"
# Network Policy Alert-specific parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
<th>Values</th>
<th>Required</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>alertFilter</td>
<td>Alert filter used for the alert.</td>
<td>Possible values:</td>
<td>yes</td>
<td>string</td>
</tr>
<tr>
<td></td>
<td></td>
<td>None, False Acceptance, False Rejection</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>None — Report on both False Rejection and False Acceptance traffic.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>False Acceptance — Report only the traffic that passed the firewall,</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>but should have been rejected according to this policy.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>False Rejection — Report only the traffic that the firewall denied,</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>but should have been accepted according to this policy.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>policyAction</td>
<td>Type of policy rules.</td>
<td>Possible values:</td>
<td>yes</td>
<td>string</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Accept, Deny</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Accept - policy rules that define network traffic that the firewall</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>should accept.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Deny — policy rules that define network traffic that the firewall</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>should reject.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>srcIPMin</td>
<td>The minimum limit for your Source IP addresses. This is for incoming and</td>
<td>Standard IP address format. For example:</td>
<td>yes</td>
<td>string</td>
</tr>
<tr>
<td></td>
<td>outgoing traffic that accesses your firewall.</td>
<td>0.0.0.0.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>srcIPMax</td>
<td>The maximum limit for your Source IP addresses. This is for incoming and</td>
<td>Standard IP address format. For example:</td>
<td>yes</td>
<td>string</td>
</tr>
<tr>
<td></td>
<td>outgoing traffic that accesses your firewall.</td>
<td>255.255.255.255</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Alert Service Operations : Alert-Specific Request Parameters

Table 5  Network Policy Alert-specific parameters (Continued)

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
<th>Values</th>
<th>Required</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>srcPortMin</td>
<td>The lower limit range for your source ports. This is for incoming and outgoing traffic that accesses your firewall. The srcPortMin and srcPortMax parameters make up the source port range.</td>
<td>Valid ports are ports 0 through 65,535.</td>
<td>yes</td>
<td>string</td>
</tr>
<tr>
<td>srcPortMax</td>
<td>The upper limit range for your source ports. This is for incoming and outgoing traffic that accesses your firewall. The srcPortMin and srcPortMax parameters make up the source port range.</td>
<td>Valid ports are ports 0 through 65,535.</td>
<td>yes</td>
<td>string</td>
</tr>
<tr>
<td>destIPMin</td>
<td>The minimum limit for your destination IP addresses. This is for incoming and outgoing traffic that accesses your firewall. The destIPMin and destIPMax parameters make up the destination IP range.</td>
<td>Standard IP address format. For example: 10.1.2.3</td>
<td>yes</td>
<td>string</td>
</tr>
<tr>
<td>destIPMax</td>
<td>The maximum limit for your destination IP addresses. This is for incoming and outgoing traffic that accesses your firewall. The destIPMin and destIPMax parameters make up the destination IP range.</td>
<td>Standard IP address format. For example: 255.255.255.255</td>
<td>yes</td>
<td>string</td>
</tr>
<tr>
<td>destPortMin</td>
<td>The lower limit range for your destination ports. This is for incoming and outgoing traffic that accesses your firewall. The destPortMin and destPortMax parameters make up the destination port range.</td>
<td>Valid ports are ports 0 through 65,535.</td>
<td>yes</td>
<td>string</td>
</tr>
<tr>
<td>destPortMax</td>
<td>The upper limit range for your destination ports. This is for incoming and outgoing traffic that accesses your firewall. The destPortMin and destPortMax parameters make up the destination port range.</td>
<td>Valid ports are ports 0 through 65,535.</td>
<td>yes</td>
<td>string</td>
</tr>
<tr>
<td>protocol</td>
<td>Protocol associated with the specified IP address LogLogic Appliances support ICMP, TCP, and UDP protocols.</td>
<td>Possible values: tcp, udp, icmp, tcp.udp, tcp.icmp, udp.icmp, tcp.udp.icmp, all The default is all, for all protocols.</td>
<td>yes</td>
<td>string</td>
</tr>
</tbody>
</table>
Pre-Defined Search Filter Alert

The Pre-Defined Search Filter Alert allows for alert notification when a text search match occurs within the received log message. This alert leverages the Log Appliance search filters for the text search match definitions.

Table 6 lists the Pre-Defined Search Filter Alert-specific parameters. You must include the parameters as inputs for the alertRules parameter.

Request Parameters

searchFilterName, fewerThan, moreThan, timeSpan

Example

"searchFilterName/MySearchFilter//fewerThan/100//moreThan/10 //timeSpan/60"

Table 6  Pre-Defined Search Filter Alert-specific parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
<th>Values</th>
<th>Required</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>searchFilterName</td>
<td>Name of the search filter.</td>
<td>Any text up to 64 characters in length.</td>
<td>yes</td>
<td>string</td>
</tr>
<tr>
<td>fewerThan</td>
<td>Minimum number of messages that must be received within a time period (timeSpan parameter) before an alert is generated. If the number of messages drops below fewerThan, an alert is generated. The fewerThan and moreThan parameters make up the alert range. You do not have to specify both fewerThan and moreThan.</td>
<td>Any positive integer between 1 and 100.</td>
<td>yes</td>
<td>string</td>
</tr>
<tr>
<td>moreThan</td>
<td>Maximum number of messages that can be received within a time period (timeSpan parameter) before an alert is generated. If the number of messages exceeds moreThan, an alert is generated. The fewerThan and moreThan parameters make up the alert range. You do not have to specify both fewerThan and moreThan.</td>
<td>Any positive integer between 1 and 100.</td>
<td>yes</td>
<td>string</td>
</tr>
<tr>
<td>timeSpan</td>
<td>Period of time that must be exceeded by the fewerThan and moreThan thresholds before an alert is triggered. If the fewerThan and moreThan thresholds are met for the specified timeSpan, an alert is generated.</td>
<td>Any positive integer. The value is in seconds. For example, the value 120 represents two minutes. The default is 60.</td>
<td>yes</td>
<td>string</td>
</tr>
</tbody>
</table>
Ratio Based Alert

The Ratio Based Alert triggers when the percentage of a specified message type exceeds or falls below specified percentages. For example, the Denied/(Accept+Denied) Alert Ratio can be used to trigger an alert when the number of Denied messages exceeds 90% of the Accept and Denied message count.

Table 7 lists the Ratio Based Alert-specific parameters. You must include the parameters as inputs for the alertRules parameter.

Request Parameters

c fewerThan, moreThan, ratio

Example

“fewerThan/100//moreThan/10//ratio/Accept%2FTotal”

Table 7  Ratio Based Alert-specific rules

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
<th>Values</th>
<th>Required</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>fewerThan</td>
<td>Minimum percentage of messages (by ratio specified by ratio parameter) that must be received before an alert is generated. If the number of messages drops below the fewerThan value, then an alert is generated. The fewerThan and moreThan parameters make up the alert range for the value specified by the ratio parameter.</td>
<td>Any positive integer between 1 and 100.</td>
<td>yes</td>
<td>string</td>
</tr>
</tbody>
</table>
### Alert Service Operations: Alert-Specific Request Parameters

#### moreThan

Maximum percentage of messages (by ratio specified by the ratio parameter) that must be received before an alert is generated. If the number of messages drops below the fewerThan value, then an alert is generated.

The fewerThan and moreThan parameters make up the alert range for the value specified by the ratio parameter.

Any positive integer between 1 and 100.

**Required**: yes  
**Type**: string

#### ratio

Message count ratio for the specified alert.

Possible values:
- Accept/Total
- Deny/Total
- Login Success/Total
- Login Failure/Total
- Accept/ (Accept+Denied)
- Denied/ (Accept+Denied)
- Login Success/ (Success+Failure)
- Denied Success/ (Success+Failure)

You must substitute %2F for each forward slash. (The F is case-sensitive.) For example: Accept%2FTotal.

**Required**: yes  
**Type**: string

### Table 7  Ratio Based Alert-specific rules

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
<th>Values</th>
<th>Required</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>moreThan</td>
<td>Maximum percentage of messages (by ratio specified by ratio parameter) that must be received before an alert is generated. If the number of messages drops below the fewerThan value, then an alert is generated. The fewerThan and moreThan parameters make up the alert range for the value specified by the ratio parameter.</td>
<td>Any positive integer between 1 and 100.</td>
<td>yes</td>
<td>string</td>
</tr>
</tbody>
</table>
| ratio     | Message count ratio for the specified alert. | Possible values:  
- Accept/Total  
- Deny/Total  
- Login Success/Total  
- Login Failure/Total  
- Accept/ (Accept+Denied)  
- Denied/ (Accept+Denied)  
- Login Success/ (Success+Failure)  
- Denied Success/ (Success+Failure)  
You must substitute %2F for each forward slash. (The F is case-sensitive.) For example: Accept%2FTotal. | yes | string |
System Alert

The System Alert allows for notification when system health and status criteria exceed acceptable bounds.

Table 8 lists the System Alert-specific parameters. You must include the parameters as inputs for the alertRules parameter.

Request Parameters

alertCriteria, lowThreshold, highThreshold, drive, peerIP

Example

>alertCriteria/CPU Temperature/highThreshold/80"
### Table 8 System Alert-specific Rules

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
<th>Values</th>
<th>Required</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>alertCriteria</td>
<td>The criteria used to generate an alert.</td>
<td>Possible values: CPU Temperature, Disk Usage, Dropped Messages, Fail-over, Network Connection Speed, Network Interface, Synchronization Failure, TCP Forward Connection Status</td>
<td>yes</td>
<td>string</td>
</tr>
<tr>
<td></td>
<td>CPU Temperature — CPU temperature, in Celsius degrees, under which the CPU temperature must remain.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Disk Usage — Drive and percentage under which the disk usage must remain.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Dropped Messages — Maximum dropped message rate (messages per second).</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Fail-over — Identifies when fail-over is enacted.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Network Connection Speed — Minimum throughput threshold for the network connection.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Network Interface — Identifies that a network interface (for example, eth0 or eth1) fails to function.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Synchronization Failure — Identifies if the data synchronization process fails after failover occurs.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>TCP Forward Connection Status — Downstream device IP address. Forwarding Rules are required for this alert criteria. Use the Message Routing Service Operations on page 69 or specify the message routing rules in the LogLogic User interface (Administration &gt; Message Routing).</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>interface</th>
<th>Specify the interfaces that the system should alert on.</th>
<th>Possible values: eth0, eth1 or eth1</th>
<th>yes</th>
<th>string</th>
</tr>
</thead>
<tbody>
<tr>
<td>(Network Interface only)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>lowThreshold</th>
<th>Minimum connection speed. Specify a value if implementing the Network Connection Speed alert criteria (AlertCriteria parameter)</th>
<th>Possible values: 10-half, 100-half, 100-full, 1000-full. The default is 10-half.</th>
<th>yes</th>
<th>string</th>
</tr>
</thead>
<tbody>
<tr>
<td>(Network Connection Speed only)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### Table 8  System Alert-specific Rules (Continued)

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
<th>Values</th>
<th>Required</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>highThreshold (CPU Temperature only)</td>
<td>CPU temperature, in Celsius degrees, under which the CPU temperature must remain. Specify a value if implementing the CPU Temperature alert criteria (AlertCriteria parameter)</td>
<td>Any positive integer between 1 and 100. The value is in Celsius degrees. The default is 80 degrees Celsius.</td>
<td>yes</td>
<td>string</td>
</tr>
<tr>
<td>highThreshold (Disk Usage only)</td>
<td>The percentage of allowable disk space usage on the specified disk drive. Use in conjunction with the drive parameter. Specify only if the alertCriteria value is Disk Usage.</td>
<td>Any positive integer between 1 and 100. The default is 90 (90%).</td>
<td>yes</td>
<td>string</td>
</tr>
<tr>
<td>highThreshold (Dropped Messages only)</td>
<td>Maximum dropped message rate (messages per second). Specify only if the alertCriteria value is Dropped Messages.</td>
<td>Any positive integer. The value is in seconds. The default is 100 messages per second.</td>
<td>yes</td>
<td>string</td>
</tr>
<tr>
<td>drive (Disk Usage only)</td>
<td>The disk drive under which usage must remain below the value specified in the highThreshold (Disk Usage) parameter. Use in conjunction with the highThreshold (Disk Usage) parameter. Specify only if the alertCriteria value is Disk Usage.</td>
<td>Possible values: /*, /failsafe, /tmp, /loglogic.</td>
<td>yes</td>
<td>string</td>
</tr>
<tr>
<td>peerIP (TCP Forward Connection Status only)</td>
<td>Downstream device IP address. The IP address is created using the destinationIP parameter in Message Routing Service Operations on page 69 or by specifying the Destination IP in the LogLogic User interface (Administration &gt; Message Routing). Specify only if the alertCriteria value is TCP Forward Connection Status.</td>
<td>Valid IP address associated with a message routing rule.</td>
<td>yes</td>
<td>string</td>
</tr>
</tbody>
</table>
VPN Connections Alert

The VPN Connection Alert triggers when a VPN connection is denied access and/or disconnected. The VPN Connection alert is only applicable to Cisco VPN, Radius, and Nortel Contivity devices.

Table 9 lists the VPN Connections Alert-specific parameters. You must include the parameters as inputs for the alertRules parameter.

Request Parameters

VPNUser, VPNGroup, VPNIP, disconnectReason

Example

"VPNUser/User1//VPNGroup/Group1//disconnectReason/both"

or

"VPNIP/10.1.2.3//disconnectReason/isdisconnected"

Table 9  VPN Connections Alert-specific rules

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
<th>Values</th>
<th>Required</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>VPNUser</td>
<td>Defines the VPN user from where the connection originates. If you specify a value for the VPNUser parameter, you must also specify a value for the VPNGroup parameter. You can specify a value for the VPNIP parameter instead of the VPNUser and VPNGroup parameters together.</td>
<td>optional</td>
<td>string</td>
<td></td>
</tr>
<tr>
<td>VPNGroup</td>
<td>Defines the VPN Group from where the connection originates. You must specify a value for the VPNGroup parameter if you specify a value for the VPNUser parameter. You can specify a value for the VPNIP parameter instead of the VPNUser and VPNGroup parameters together.</td>
<td>optional</td>
<td>string</td>
<td></td>
</tr>
</tbody>
</table>
VPN Messages Alert

The VPN Message Alert triggers on combinations of specific VPN message area, severity, and code. This alert is applicable to Cisco VPN devices.

Table 10 lists the VPN Messages Alert-specific parameters. You must include the parameters as inputs for the `alertRules` parameter.

**Request Parameters**

`messageArea, messageCode, severityFrom, severityTo`

**Example**

```
"messageArea/ANY//messageCode/0//severityFrom/2//severityTo/6"
```
VPN Statistics Alert

The VPN Statistics Alert triggers when recorded statistics on VPN or Radius messages match relative or absolute criteria. For example, an alert can be configured to trigger when the Number of Bytes Received per day for a specific user exceeds, say, 1 Mb per day, which is an absolute value. The alert rule could also be configured as a relative rule, for example “grows by 10%.”

Table 11 lists the VPN Statistics Alert-specific parameters. You must include the parameters as inputs for the alertRules parameter.

Request Parameters

VPNUser, VPNGroup, VPNIP, statistic, matchCount, perTimeUnit, measureBy, changeAs, changeValue
### Example

"VPNUser/10.1.2.3//statistic/Number of Denies//matchCount/5//perTimeUnit/minute//measureBy/1//changeAs/falls below 5//changeValue/5"

or

"VPNUser/User1//VPNGroup/Group1//statistic/Number of Connections//matchCount/25//perTimeUnit/minute//measureBy/5//changeAs/exceeds 5//changeValue/5"

---

**Table 11** VPN Statistics Alert-specific rules

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
<th>Values</th>
<th>Required</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>VPNUser</td>
<td>Defines the VPN user from where the connection originates. If you specify a value for the VPNUser parameter, you must also specify a value for the VPNGroup parameter. You can specify a value for the VPNIP parameter instead of the VPNUser and VPNGroup parameters together.</td>
<td>Optional string</td>
<td>Optional</td>
<td>String</td>
</tr>
<tr>
<td>VPNGroup</td>
<td>Defines the VPN Group from where the connection originates. You must specify a value for the VPNGroup parameter if you specify a value for the VPNUser parameter. You can specify a value for the VPNIP parameter instead of the VPNUser and VPNGroup parameters together.</td>
<td>Optional string</td>
<td>Optional</td>
<td>String</td>
</tr>
<tr>
<td>VPNIP</td>
<td>Defines the VPN IP address from where the connection originates. You can use the VPNIP parameter as an additional filter instead of using the VPNUser and VPNGroup parameters.</td>
<td>Standard IP address format. For example: 10.1.2.3</td>
<td>Optional</td>
<td>String</td>
</tr>
<tr>
<td>statistic</td>
<td>Identifies the type of statistic. Specify the perTimeUnit parameter with the statistic parameter. If you specify the value as Connection Duration, the perTimeUnit parameter is not necessary as the value defaults to seconds.</td>
<td>Possible values: Number of Connections, Number of Denies, Bytes Sent, Bytes Received, Connection Duration.</td>
<td>Yes</td>
<td>String</td>
</tr>
</tbody>
</table>
### Table 11  VPN Statistics Alert-specific rules

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
<th>Values</th>
<th>Required</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>matchCount</td>
<td>Rate at which the statistic type (statistic parameter) is sampled. This field is not applicable to if the value of the statistic parameter is Connection Duration.</td>
<td>Possible values: second, minute, hour, day, week, none. If you specify none, statistics are measured regardless of rate.</td>
<td>yes</td>
<td>string</td>
</tr>
<tr>
<td>perTimeUnit</td>
<td>Defines the number of times a match must occur before an alert is sent. The match is determined by the combination of the fields you define for this type of alert. The measureBy parameter represents the Threshold field in the user interface.</td>
<td>Any positive integer.</td>
<td>yes</td>
<td>string</td>
</tr>
<tr>
<td>changeAs</td>
<td>Defines the percentage of increase or decrease of the alert type (statistic parameter). For example, to be alerted when a number of Denied Connections per second grows by 400% from the average, enter the value “grows by 400”. The average is taken from the previous time period and varies depending on the type of information you view.</td>
<td>Possible values: grows by ##% drops by ##% Where ## is a positive integer. The ## value is a percentage. For example: grows by 400 or drops by 200</td>
<td>yes</td>
<td>string</td>
</tr>
<tr>
<td>changeValue</td>
<td>Defines the absolute number of denied connections per second that an alert type (statistic parameter) must change by for an alert to be generated. For example, to be alerted when a number of Denied Connections per second increases by 400 from the average, enter the value “increases by 400”.</td>
<td>Possible values: exceeds ## falls below ## equals ## increase by ## decrease by ## Where ## is a positive integer. The ## value is the number of denied connections per second. For example: exceeds 400 or equals 100</td>
<td>yes</td>
<td>string</td>
</tr>
</tbody>
</table>
CHAPTER 3:

Device Service Operations

The Device Service operations enable you to manage devices in your LogLogic Appliance.

- Overview .......................................................... 51
- Implementation Guidelines ........................................ 51
- Device Service Operation Descriptions ............................ 52
- Status Codes ..................................................... 57
- Common Request Parameters .................................... 57

Overview

The Device Service operations let you create, read, update, and delete devices as well as view all devices in the LogLogic Appliance.

Devices are any source of log data that you want captured by a LogLogic Appliance. In the LogLogic Appliance, to view the user interface implementation navigate to Administration > Manage Devices.

Implementation Guidelines

The following are general implementation guidelines for Device Service operations:

- A set of Common Request Parameters (on page 57) are required for each operation. You must specify a value for all Required parameters.

- You must specify the value of the Common Request Parameters only.

- All parameters must be implemented in the order in which they appear in the Request Parameters section for each of the operations.

- All values for Common Request Parameters must be enclosed by double quotation marks (“value”). If the name of a device or a parameter value includes a forward slash (/), such as HP/UX, or IBM i5/OS, you must replace the forward slash with %2F. (The F is case-sensitive.)

Examples: HP%2FUX or IBM %2Fi5%2FOS

- When using the createDevice Operation, createDeviceRemote Operation, updateDevice Operation, or updateDeviceRemote Operation, if the dnsRefreshEnabled parameter is enabled, the value for the deviceName parameter can be overwritten by the value on your DNS server. If this occurs, use the DNS name of the device as the value for the deviceName parameter. You can retrieve the DNS name by looking up the IP address of the device.
Device Service Operation Descriptions

There are two kinds of operations:

- Local - operation is performed on the local Appliance itself
- Remote - operations (names ending with `Remote`) performed on a specified remote Appliance

The Device Service operations are as follows:

Contents

- createDevice Operation .............................................................. 52
- createDeviceRemote Operation .................................................. 53
- readDevice Operation ............................................................... 53
- readDeviceRemote Operation ..................................................... 54
- updateDevice Operation ............................................................ 54
- updateDeviceRemote Operation .................................................. 54
- deleteDevice Operation ............................................................. 55
- deleteDeviceRemote Operation ................................................... 55
- getList Operation ..................................................................... 55
- getListRemote Operation .......................................................... 56
- deviceResponse Type .................................................................. 56

createDevice Operation

The `createDevice` operation lets you add a new device to be supported by the LogLogic Appliance.

**Note:** Managing File Transfer Rules is not supported through the Web Services API. You must use the LogLogic user interface to manage File Transfer Rules for each device.

Request Parameters

`authToken`, `deviceName`, `deviceType`, `description`, `enabled`, `dnsRefreshEnabled`, `deviceIp`

For more information on each Common Request Parameter, see *Common Request Parameters* on page 57.

Response

`deviceResponse` (see `deviceResponse Type` on page 56)
Example

To add a Cisco ACS device named Cisco ACS Sample Device, with a description of Cisco ACS Description, having IP address 10.1.2.3, and enabling the Appliance to retrieve log messages from the device but not enabling DNS name refresh:

createDevice authstr “Cisco ACS Sample Device” “Cisco ACS” “Cisco ACS Description” “yes” “no” “10.1.2.3”

createDeviceRemote Operation

The createDeviceRemote operation lets you add a new device to be supported by a managed LogLogic Appliance from a Management Station.

Note: Managing File Transfer Rules is not supported through the Web Services API. You must use the LogLogic user interface to manage File Transfer Rules for each device.

Request Parameters

authToken, applianceIP, deviceName, deviceType, description, enabled, dnsRefreshEnabled, deviceIp

For more information on each Common Request Parameter, see Common Request Parameters on page 57.

Response

deviceResponse (see deviceResponseType on page 56)

Example

To add (on remote Appliance 1.2.20.100) a Cisco ACS device named Cisco ACS Sample Device, with a description of Cisco ACS Description, having IP address 10.1.2.3, and enabling the Appliance to retrieve log messages from the device but not enabling DNS name refresh:

createDeviceRemote authstr 1.2.20.100 “Cisco ACS Sample Device” “Cisco ACS” “Cisco ACS Description” “yes” “no” “10.1.2.3”

readDevice Operation

The readDevice operation lets you view the configuration of an existing LogLogic supported device based on the deviceName.

Request Parameters

authToken, deviceName

Response

deviceResponse (see deviceResponseType on page 56)

Example

readDevice authstr “MyOracleDBName”
readDeviceRemote Operation

The readDeviceRemote operation lets you view the configuration of an existing LogLogic supported device based on the deviceName on a managed LogLogic Appliance from a Management Station.

Request Parameters

authToken, applianceIP, deviceName

Response

deviceResponse (see deviceResponse Type on page 56)

Example

readDeviceRemote authstr 1.2.20.100 "MyOracleDBName"

updateDevice Operation

The updateDevice operation lets you update the configuration of an existing device.

Request Parameters

authToken, deviceName, deviceType, description, enabled, dnsRefreshEnabled, deviceIp, changeNameTo

Response

deviceResponse (see deviceResponse Type on page 56)

Example

To update the CiscoACSName device name to NewCiscoACSName:

updateDevice authstr “CiscoACSName” “Cisco ACS” “CiscoACSDescription” “yes” “no” “10.1.2.9” “NewCiscoACSName”

updateDeviceRemote Operation

The updateDeviceRemote operation lets you update the configuration of an existing device on a managed LogLogic Appliance from a Management Station.

Request Parameters

authToken, applianceIP, deviceName, deviceType, description, enabled, dnsRefreshEnabled, deviceIp, changeNameTo

Response

deviceResponse (see deviceResponse Type on page 56)

Example

To update the CiscoACSName device name to NewCiscoACSName:
updateDeviceRemote authstr 1.2.20.100 "CiscoACSName" "Cisco ACS"
"CiscoACSDescription" "yes" "no" "10.1.2.9" "NewCiscoACSName"

**deleteDevice Operation**

The deleteDevice operation lets you delete an existing device.

**Request Parameters**

authToken, deviceName

**Response**

deviceResponse (see *deviceResponse Type* on page 56)

**Example**

To delete the device MyDeviceName from the Appliance:

deleteDevice authstr "MyDeviceName"

**deleteDeviceRemote Operation**

The deleteDeviceRemote operation lets you delete an existing device on a managed LogLogic Appliance from a Management Station.

**Request Parameters**

authToken, applianceIP, deviceName

**Response**

deviceResponse (see *deviceResponse Type* on page 56)

**Example**

To delete the device MyDeviceName from remote Appliance 1.2.20.100:

deleteDeviceRemote authstr 1.2.20.100 "MyDeviceName"

**getList Operation**

The getList operation lets you retrieve a list of all devices currently configured for the LogLogic Appliance. This returns all devices, not just enabled devices.

**Request Parameter**

authToken

**Response**

If resultCount is greater than 0 and statusCode is 2000 (successful), the response returns a list of all devices (total number indicated by resultCount) currently configured for the LogLogic Appliance.
If resultCount is 0 and statusCode is not 2000 (successful), an error is returned in statusMessage.

**Example**

To list all devices currently configured for the Appliance:

```
getList authstr
```

**getListRemote Operation**

The `getListRemote` operation lets you retrieve a list of all devices currently configured for a managed LogLogic Appliance from a Management Station. This returns all devices, not just enabled devices.

**Request Parameter**

authToken, applianceIP

**Response**

If resultCount is greater than 0 and statusCode is 2000 (successful), the response returns a list of all devices (total number indicated by resultCount) currently configured for the remote LogLogic Appliance.

If resultCount is 0 and statusCode is not 2000 (successful), an error is returned in statusMessage.

**Example**

To list all devices currently configured for remote Appliance 1.2.20.100:

```
getListRemote authstr 1.2.20.100
```

**deviceResponse Type**

deviceResponse is returned for all device operations except `getList` and `getListRemote`.

deviceResponse always contains the following common elements:

- authToken
- resultCount
- statusCode
- statusMessage
- summaryOnly
If `resultCount` is 1 and `statusCode` is 2000 (successful), the `resultSet` element is included after `resultCount` listing the following device details:

1. applianceIp
2. applianceName
3. description
4. deviceId
5. deviceIp
6. deviceName
7. deviceType
8. dnsRefreshEnabled
9. enabled

If `resultCount` is 0 and `statusCode` is not 2000 (successful), an error is returned in `statusMessage`.

### Status Codes

The Status Codes are:

<table>
<thead>
<tr>
<th>Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>2000</td>
<td>Server success</td>
</tr>
<tr>
<td>4000</td>
<td>Unauthorized request</td>
</tr>
<tr>
<td>5000</td>
<td>Invalid parameter, <code>getStatusMessage()</code> contains detail information about the error</td>
</tr>
</tbody>
</table>

### Common Request Parameters

A set of Common Request Parameters are required for each Device Service operation. Table 12 lists all of the Common Request parameters for the Device Service operations. For more information on the parameters, view the LogLogic Appliance user interface for managing devices as well as the online help.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
<th>Values</th>
<th>Required</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>authToken</td>
<td>Unique authentication token.</td>
<td></td>
<td>yes</td>
<td>string</td>
</tr>
<tr>
<td>applianceIp</td>
<td>The managed Appliance on which you perform the operation.</td>
<td>IP address of a managed Appliance. To specify an IP address, use the standard IP address format. For example: 10.1.2.3</td>
<td>yes</td>
<td>string</td>
</tr>
<tr>
<td></td>
<td>If the value is blank, it retrieves the Appliance IP address from the local Appliance.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>This parameter is available only for Management Station Appliances using operations with Remote in the name.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>deviceName</td>
<td>Name of the device from which you intend to transfer log data.</td>
<td>Any text up to 64 characters in length.</td>
<td>yes</td>
<td>string</td>
</tr>
</tbody>
</table>
### Table 12  Device Service Operation Common Request Parameters (Continued)

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
<th>Values</th>
<th>Required</th>
<th>Type</th>
</tr>
</thead>
</table>
| deviceType      | Type of device or application generating the logs to be transferred. The deviceType value cannot be changed once the device profile is added. | For example "Cisco Pix"  
See the Administration > Manage Devices user interface for a list of available device type values. | yes      | string |
| description     | Description for the remote device. If you do not provide a description, you must at least specify the value as an empty string. | Any text up to 64 characters in length.  
Type double quotation marks (""") to specify an empty string or no description text. | yes      | string |
| enabled         | Indicates whether retrieval of log files from this device is enabled.  
(Appears as enabled or disabled in returned value.) | Possible values:  
yes — enable  
no — disable | yes      | string |
| dnsRefreshEnabled| Enables the Device Name to be refreshed through DNS lookups.  
The DNS name might override any name you assign in the deviceName. | Possible values:  
yes — enable  
no — disable | yes      | string |
| deviceIp        | IP address of the device from which you want to transfer files. | Standard IP address format. For example:  
10.1.2.3 | yes      | string |
| changeNameTo    | New name of the device.  
If empty, the object name is unchanged. | Any text up to 64 characters in length.  
(forupdateDevice and updateDeviceRemote only) | yes      | string |
CHAPTER 4: Device Group Service Operations

The Device Group Service operations enable you to manage device groups in your LogLogic Appliance.

- Overview ................................................................. 59
- Implementation Guidelines ............................................. 59
- Device Group Service Operation Descriptions ................. 59
- Status Codes ............................................................. 67
- Common Request Parameters ....................................... 67

Overview

The Device Group Service operations let you create, read, update, and delete device groups as well as view all device groups in the LogLogic Appliance.

Device groups are groups configured on the LogLogic Appliance of any source of log data that you want captured by the Appliance. In the LogLogic Appliance, to view the user interface implementation navigate to Administration > Manage Devices > Groups.

Implementation Guidelines

The following are general implementation guidelines for Device Group Service operations:

- A set of Common Request Parameters (on page 67) are required for each operation. You must specify a value for all Required parameters.
- You must specify the value of the Common Request Parameters only.
- All parameters must be implemented in the order in which they appear in the Request Parameters section for each of the operations.
- All values for Common Request Parameters must be enclosed by double quotation marks ("value"). If the name of a device or a parameter value includes a forward slash (/), such as HP/UX, or IBM i5/OS, you must replace the forward slash with %2F. (The F is case-sensitive.)

Examples: HP%2FUX or IBM i5%2FOS

Device Group Service Operation Descriptions

There are two kinds of operations:

- local - operation is performed on the local Appliance itself
- remote - operations (names ending with Remote) performed on a specified remote Appliance
There are two kinds of device groups:

- local - a group of log sources on the local Appliance (that is, the Appliance you are logged into)
- global - a group of log sources on multiple Appliances. Global groups are created and managed only from Management Station Appliances.

The Device Group Service operations are as follows:

- `createDeviceGroup Operation` .................................................. 61
- `createDeviceGroupRemote Operation` ..................................... 61
- `addDevices Operation` ................................................................. 62
- `addDevicesRemote Operation` .................................................. 62
- `removeDevices Operation` ......................................................... 62
- `removeDevicesRemote Operation` ............................................. 63
- `readDeviceGroup Operation` ...................................................... 63
- `readDeviceGroupRemote Operation` ....................................... 64
- `updateDeviceGroup Operation` .................................................. 64
- `updateDeviceGroupRemote Operation` ..................................... 64
- `deleteDeviceGroup Operation` .................................................. 65
- `deleteDeviceGroupRemote Operation` ..................................... 65
- `getList Operation` ................................................................. 65
- `getListRemote Operation` ......................................................... 66
- `deviceGroupResponse Type` ..................................................... 66
createDeviceGroup Operation

The createDeviceGroup operation lets you add a new device group to the LogLogic Appliance.

Request Parameters

authToken, deviceGroupName, description, groupType, enabled, deviceNames

For more information on each Common Request Parameter, see Common Request Parameters on page 67.

Response
deviceGroupResponse (see deviceGroupResponse Type on page 66)

Example

To add a local device group named Cisco ACS Devices, with a description of Cisco ACS Description, including log sources 10.1.2.3 and CiscoACS1, and enabling the group:
createDeviceGroup authstr 1.2.20.100 "Cisco ACS Devices" "Cisco ACS Description" "local" "yes" "10.1.2.3,CiscoACS1"

createDeviceGroupRemote Operation

The createDeviceGroupRemote operation lets you add a new device group to a managed LogLogic Appliance from a Management Station.

Request Parameters

authToken, applianceIP, deviceGroupName, description, groupType, enabled, deviceNames

For more information on each Common Request Parameter, see Common Request Parameters on page 67.

Response
deviceGroupResponse (see deviceGroupResponse Type on page 66)

Example

To add (on remote Appliance 1.2.20.100) a global device group named Cisco ACS Devices, with a description of Cisco ACS Description, including log sources 10.1.2.3 and CiscoACS1, and enabling the group:
createDeviceGroupRemote authstr 1.2.20.100 "Cisco ACS Devices" "Cisco ACS Description" "global" "yes" "10.1.2.3,CiscoACS1"
addDevices Operation

The addDevices operation lets you add one or more devices to a device group on a LogLogic Appliance.

Request Parameters

authToken, deviceGroupName, deviceNames

For more information on each Common Request Parameter, see Common Request Parameters on page 67.

Response

deviceGroupResponse (see deviceGroupResponse Type on page 66)

Example

To add devices CiscoACS2 and 10.1.4.6 to the device group Cisco ACS Devices:

```
addDevices authstr "Cisco ACS Devices" "CiscoACS2,10.1.4.6"
```

addDevicesRemote Operation

The addDevicesRemote operation lets you add one or more devices to a device group on a managed LogLogic Appliance from a Management Station.

Request Parameters

authToken, applianceIP, deviceGroupName, deviceNames

For more information on each Common Request Parameter, see Common Request Parameters on page 67.

Response

deviceGroupResponse (see deviceGroupResponse Type on page 66)

Example

To add (on remote Appliance 1.2.20.100) devices CiscoACS2 and 10.1.4.6 to the device group Cisco ACS Devices:

```
addDevicesRemote authstr 1.2.20.100 "Cisco ACS Devices" "CiscoACS2,10.1.4.6"
```

removeDevices Operation

The removeDevices operation lets you remove one or more devices from a device group on a LogLogic Appliance.

Request Parameters

authToken, deviceGroupName, deviceNames
For more information on each Common Request Parameter, see Common Request Parameters on page 67.

**Response**

deviceGroupResponse (see deviceGroupResponse Type on page 66)

**Example**

To remove devices CiscoACS2 and 10.1.4.6 from the device group Cisco ACS Devices:
removeDevices authstr "Cisco ACS Devices" "CiscoACS2,10.1.4.6"

**removeDevicesRemote Operation**

The removeDevicesRemote operation lets you remove one or more devices from a device group on a managed LogLogic Appliance from a Management Station.

**Request Parameters**

authToken, applianceIP, deviceGroupName, deviceNames

For more information on each Common Request Parameter, see Common Request Parameters on page 67.

**Response**

deviceGroupResponse (see deviceGroupResponse Type on page 66)

**Example**

To remove (on remote Appliance 1.2.20.100) devices CiscoACS2 and 10.1.4.6 from the device group Cisco ACS Devices:
removeDevicesRemote authstr 1.2.20.100 "Cisco ACS Devices" "CiscoACS2,10.1.4.6"

**readDeviceGroup Operation**

The readDeviceGroup operation lets you view the configuration of an existing LogLogic Appliance device group based on the deviceGroupName.

**Request Parameters**

authToken, deviceGroupName

**Response**

deviceResponse (see deviceGroupResponse Type on page 66)

**Example**

readDeviceGroup authstr "MyOracleDBGroup"
readDeviceGroupRemote Operation

The readDeviceGroupRemote operation lets you view the configuration of an existing device group based on the deviceGroupName on a managed LogLogic Appliance from a Management Station.

Request Parameters

authToken, applianceIP, deviceGroupName

Response

deviceResponse (see deviceGroupResponse Type on page 66)

Example

readDeviceGroupRemote authstr 1.2.20.100 "MyOracleDBGroup"

updateDeviceGroup Operation

The updateDeviceGroup operation lets you update the configuration of an existing device group.

Request Parameters

authToken, deviceGroupName, description, groupType, enabled, deviceNames, changeNameTo

If deviceNames is empty, the group’s devices are unchanged. If deviceNames are specified, the group is updated to include the listed devices.

Response

deviceResponse (see deviceGroupResponse Type on page 66)

Example

To update the global CiscoACSGroup device group name to NewCiscoACSGroup:
updateDeviceGroup authstr "CiscoACSGroup" "CiscoACSDescription" "global" "yes" "NewCiscoACSGroup"

updateDeviceGroupRemote Operation

The updateDeviceGroupRemote operation lets you update the configuration of an existing device group on a managed LogLogic Appliance from a Management Station.

Request Parameters

authToken, applianceIP, deviceGroupName, description, groupType, enabled, deviceNames, changeNameTo

If deviceNames is empty, the group’s devices are unchanged. If deviceNames are specified, the group is updated to include the listed devices.
Response
deviceResponse (see deviceGroupResponse Type on page 66)

Example
To update the global CiscoACSGroup device group name on LogLogic Appliance 10.1.20.200 to NewCiscoACSGroup:

updateDeviceGroupRemote authstr 10.1.20.200 "CiscoACSGroup" "CiscoACSDescription" "global" "yes" "" "NewCiscoACSGroup"

deleteDeviceGroup Operation
The deleteDeviceGroup operation lets you delete an existing device group.

Request Parameters
authToken, deviceGroupName

Response
deviceResponse (see deviceGroupResponse Type on page 66)

Example
To delete the device group MyDeviceGroup from the Appliance:
deleteDevice authstr "MyDeviceGroup"

deleteDeviceGroupRemote Operation
The deleteDeviceGroupRemote operation lets you delete an existing device group on a managed LogLogic Appliance from a Management Station.

Request Parameters
authToken, applianceIP, deviceGroupName

Response
deviceResponse (see deviceGroupResponse Type on page 66)

Example
To delete the device group MyDeviceGroup from remote Appliance 1.2.20.100:
deleteDeviceGroupRemote authstr 1.2.20.100 "MyDeviceGroup"

getList Operation
The getList operation lets you retrieve a list of all device groups currently configured for the LogLogic Appliance. This returns all device groups, not just enabled device groups.
Request Parameter

authToken

Response

If resultCount is greater than 0 and statusCode is 2000 (successful), the response returns a list of all device groups (total number indicated by resultCount) currently configured for the LogLogic Appliance.

If resultCount is 0 and statusCode is not 2000 (successful), an error is returned in statusMessage.

Example

To list all device groups currently configured for the Appliance:

g getList authstr

gGetListRemote Operation

The getListRemote operation lets you retrieve a list of all device groups currently configured for a managed LogLogic Appliance from a Management Station. This returns all device groups, not just enabled device groups.

Request Parameter

authToken, applianceIP

Response

If resultCount is greater than 0 and statusCode is 2000 (successful), the response returns a list of all device groups (total number indicated by resultCount) currently configured for the remote LogLogic Appliance.

If resultCount is 0 and statusCode is not 2000 (successful), an error is returned in statusMessage.

Example

To list all device groups currently configured for remote Appliance 1.2.20.100:

g getListRemote authstr 1.2.20.100

deviceGroupResponse Type

deviceGroupResponse is returned for all deviceGroup operations except getList and getListRemote.
deviceGroupResponse always contains the following common elements:

- authToken
- resultCount
- statusCode
- statusMessage
- summaryOnly

If resultCount is 1 and statusCode is 2000 (successful), the resultSet element is included after resultCount listing the following device group details:

- deviceGroupName
- description
- groupType
- enabled
- devices

If resultCount is 0 and statusCode is not 2000 (successful), an error is returned in statusMessage.

### Status Codes

The Status Codes are:

<table>
<thead>
<tr>
<th>Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>2000</td>
<td>Server success</td>
</tr>
<tr>
<td>4000</td>
<td>Unauthorized request</td>
</tr>
<tr>
<td>5000</td>
<td>Invalid parameter, getStatusMessage() contains detail information about the error</td>
</tr>
</tbody>
</table>

### Common Request Parameters

A set of Common Request Parameters are required for each Device Group Service operation. Table 13 lists all of the Common Request parameters for the Device Group Service operations. For more information on the parameters, view the LogLogic Appliance user interface for managing device groups as well as the online help.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
<th>Values</th>
<th>Required</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>authToken</td>
<td>Unique authentication token.</td>
<td></td>
<td>yes</td>
<td>string</td>
</tr>
<tr>
<td>applianceIp</td>
<td>The managed Appliance on which you perform the operation.</td>
<td>IP address of a managed Appliance. To specify an IP address, use the standard IP address format. For example: 10.1.2.3</td>
<td>yes</td>
<td>string</td>
</tr>
<tr>
<td>deviceGroupName</td>
<td>Name of the device group on the LogLogic Appliance.</td>
<td>Any text up to 64 characters in length.</td>
<td>yes</td>
<td>string</td>
</tr>
</tbody>
</table>
## Device Group Service Operations: Common Request Parameters (Continued)

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
<th>Values</th>
<th>Required</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>description</td>
<td>Description for the device group. If you do not provide a description, you must at least specify the value as an empty string.</td>
<td>Any text up to 64 characters in length. Type double quotation marks (&quot;&quot;&quot;) to specify an empty string or no description text.</td>
<td>yes</td>
<td>string</td>
</tr>
<tr>
<td>groupType</td>
<td>Type of device group, either local or global. The groupType value cannot be changed once the device profile is added.</td>
<td>local or global</td>
<td>yes</td>
<td>string</td>
</tr>
<tr>
<td>enabled</td>
<td>Indicates whether retrieval of log files from this device is enabled. (Appears as enabled or disabled in returned value.)</td>
<td>Possible values: yes — enable no — disable</td>
<td>yes</td>
<td>string</td>
</tr>
<tr>
<td>deviceNames</td>
<td>Names or IP addresses (comma-separated) of devices in the group.</td>
<td>Standard IP address format or device name. For example: 10.1.2.3 or CiscoACS2</td>
<td>yes</td>
<td>string</td>
</tr>
<tr>
<td>changeNameTo</td>
<td>New name of the device group. If empty, the object name is unchanged.</td>
<td>Any text up to 64 characters in length. (for updateDeviceGroup and updateDeviceGroupRemote only)</td>
<td>yes</td>
<td>string</td>
</tr>
</tbody>
</table>
CHAPTER 5:

Message Routing Service Operations

The Message Routing Service operations enable you to manage message routing rules for your LogLogic Appliance.

- Overview ................................................................. 69
- Implementation Guidelines ........................................ 69
- Message Routing Service Operation Descriptions ............ 69
- Status Codes ........................................................... 75
- Common Request Parameters .................................... 76

Overview

The Message Routing Service operations enable you to create, read, update, and delete message routing rules as well as view all message routing rules in your LogLogic Appliance.

Message Routing Rules let you forward a copy of incoming log data from one LogLogic Appliance to another LogLogic Appliance or 3rd party device. In the LogLogic Appliance you can view the user interface implementation by navigating to Administration > Message Routing.

Implementation Guidelines

The following are general implementation guidelines for the Message Routing Service operations:

- A set of Common Request Parameters (on page 76) are required for each operation. You must specify a value for all Required parameters.
- You must specify the value of the Common Request Parameters only.
- All parameters must be implemented in the order in which they appear in the Request Parameters section for each of the operations.
- All values for Common Request Parameters must be enclosed by double quotation marks ("value"). If the name of a device or a parameter value includes a forward slash (/), such as HP/UX, or IBM i5/OS, you must replace the forward slash with %2F. (The F is case-sensitive.)

Examples: HP%2FUX or IBM \ i5\%2FOS

Message Routing Service Operation Descriptions

There are two kinds of operations:

- local - operation is performed on the local Appliance itself
The Message Routing Service contains the following operations:

- **createMsgRouting Operation** .................................................. 70
- **createMsgRoutingRemote Operation** ..................................... 70
- **readMsgRouting Operation** .................................................... 71
- **readMsgRoutingRemote Operation** ....................................... 71
- **updateMsgRouting Operation** ................................................. 72
- **updateMsgRoutingRemote Operation** ..................................... 72
- **deleteMsgRouting Operation** ................................................ 73
- **deleteMsgRoutingRemote Operation** ...................................... 73
- **getList Operation** .............................................................. 73
- **getListRemote Operation** ..................................................... 74
- **msgRoutingResponse Type** ..................................................... 74

### createMsgRouting Operation

The `createMsgRouting` operation lets you create message routing rules.

**Request Parameters**

- `authToken`, `routingName`, `sourceDevice`, `destinationIP`, `destinationPort`, `routingProtocol`, `enabled`, `severity`, `facility`, `searchFilterName`, `compressionEnabled`, `encryptionEnabled`, `authenticationEnabled`

For more information on each Common Request Parameter, see *Common Request Parameters* on page 76.

**Response**

`msgRoutingResponse` (see `msgRoutingResponse Type` on page 74)

**Example**

To add a message routing rule named `MyRoutingRuleName` for device 10.1.2.3 to route to 10.2.3.4 port 4433 using the LogLogic TCP protocol, for all Syslog messages using the search filter named `MySearchFilter`, with encryption and authentication enabled:

```
createMsgRouting authstr "MyRoutingRuleName" "10.1.2.3" "10.2.3.4" "4433" "LogLogic TCP" "yes" "All Syslog" "MySearchFilter" "no" "yes" "yes"
```

### createMsgRoutingRemote Operation

The `createMsgRoutingRemote` operation lets you create message routing rules on a managed LogLogic Appliance from a Management Station.
Request Parameters
authToken, applianceIP, routingName, sourceDevice, destinationIP, destinationPort, routingProtocol, enabled, severity, facility, searchFilterName, compressionEnabled, encryptionEnabled, authenticationEnabled

For more information on each Common Request Parameter, see Common Request Parameters on page 76.

Response
msgRoutingResponse (see msgRoutingResponse Type on page 74)

Example
To add a message routing rule (on remote Appliance 1.2.20.100) named MyRoutingRuleName for device 10.1.2.3 to route to 10.2.3.4 port 4433 using the LogLogic TCP protocol, for all Syslog messages using the search filter named MySearchFilter, with encryption and authentication enabled and compression disabled:
createMsgRoutingRemote authstr 1.2.20.100 “MyRoutingRuleName” “10.1.2.3” “10.2.3.4” “4433” “LogLogic TCP” “yes” “All Syslog” “MySearchFilter” “no” “yes” “yes”

readMsgRouting Operation
The readMsgRouting operation lets you view the configuration of an existing message routing rule.

Request Parameters
authToken, routingName

Response
msgRoutingResponse (see msgRoutingResponse Type on page 74)

Example
To view the configuration of message routing rule MyRoutingRule:
readMsgRouting authstr “MyRoutingRule”

readMsgRoutingRemote Operation
The readMsgRoutingRemote operation lets you view the configuration of an existing message routing rule on a managed LogLogic Appliance from a Management Station.

Request Parameters
authToken, applianceIP, routingName

Response
msgRoutingResponse (see msgRoutingResponse Type on page 74)
Example

To view the configuration of message routing rule MyRoutingRule on remote Appliance 1.2.20.100:
readMsgRoutingRemote authstr 1.2.20.100 "MyRoutingRule"

updateMsgRouting Operation

The updateMsgRouting operation lets you update the configuration of an existing message routing rule.

Request Parameters

authToken, routingName, sourceDevice, destinationIP, destinationPort, routingProtocol, enabled, severity, facility, searchFilterName, compressionEnabled, encryptionEnabled, authenticationEnabled, changeNameTo

Response

msgRoutingResponse (see msgRoutingResponse Type on page 74)

Example

To update MyRoutingRuleName to NewRoutingRuleName:
updateMsgRouting authstr "MyRoutingRuleName" "10.1.2.3" "10.2.3.4" "4433" "LogLogic TCP" "yes" "All Syslog" "MySearchFilter" "no" "yes" "yes" "NewRoutingRuleName"

updateMsgRoutingRemote Operation

The updateMsgRoutingRemote operation lets you update the configuration of an existing message routing rule on a managed LogLogic Appliance from a Management Station.

Request Parameters

authToken, applianceIP, routingName, sourceDevice, destinationIP, destinationPort, routingProtocol, enabled, severity, facility, searchFilterName, compressionEnabled, encryptionEnabled, authenticationEnabled, changeNameTo

Response

msgRoutingResponse (see msgRoutingResponse Type on page 74)

Example

To update MyRoutingRuleName to NewRoutingRuleName on remote Appliance 1.2.20.100:
updateMsgRoutingRemote authstr 1.2.20.100 "MyRoutingRuleName" "10.1.2.3" "10.2.3.4" "4433" "LogLogic TCP" "yes" "All Syslog" "MySearchFilter" "no" "yes" "yes" "NewRoutingRuleName"
deleteMsgRouting Operation

The deleteMsgRouting operation lets you delete an existing message routing rule.

**Request Parameters**

authToken, routingName

**Response**

msgRoutingResponse (see `msgRoutingResponse Type` on page 74)

**Example**

To delete message routing rule MyRoutingRuleName from the Appliance:

delemeMsgRouting authstr “MyRoutingRuleName”

deleteMsgRoutingRemote Operation

The deleteMsgRouting operation lets you delete an existing message routing rule on a managed LogLogic Appliance from a Management Station.

**Request Parameters**

authToken, applianceIP, routingName

**Response**

msgRoutingResponse (see `msgRoutingResponse Type` on page 74)

**Example**

To delete message routing rule MyRoutingRuleName from remote Appliance 1.2.20.100:

delemeMsgRoutingRemote authstr 1.2.20.100 “MyRoutingRuleName”

ggetList Operation

The getList operation lets you retrieve a list of all message forwarding rules currently configured in the LogLogic Appliance.

**Request Parameter**

authToken

**Response**

If `resultCount` is greater than 0 and `statusCode` is 2000 (successful), the response returns a list of all message forwarding rules (total number indicated by `resultCount`) currently configured in the LogLogic Appliance.

If `resultCount` is 0 and `statusCode` is not 2000 (successful), an error is returned in `statusMessage`. 
Example

To list all message forwarding rules currently configured on the Appliance:

```bash
getList authstr
```

**getListRemote Operation**

The `getListRemote` operation lets you retrieve a list of all message forwarding rules currently configured on a managed LogLogic Appliance from a Management Station.

**Request Parameter**

authToken, applianceIP

**Response**

If `resultCount` is greater than 0 and `statusCode` is 2000 (successful), the response returns a list of all message forwarding rules (total number indicated by `resultCount`) currently configured in the remote LogLogic Appliance.

If `resultCount` is 0 and `statusCode` is not 2000 (successful), an error is returned in `statusMessage`.

**Example**

To list all message forwarding rules currently configured on remote Appliance 1.2.20.100:

```bash
getListRemote authstr 1.2.20.100
```

**msgRoutingResponse Type**

`msgRoutingResponse` is returned for all `msgRouting` operations except `getList` and `getListRemote`.

`msgRoutingResponse` always contains the following common elements:

- authToken
- resultCount
- statusCode
- statusMessage
- summaryOnly
If `resultCount` is 1 and `statusCode` is 2000 (successful), the `resultSet` element is included after `resultCount` listing the following Message Forwarding Rule details:

- `routingName`
- `sourceDevice`
- `destinationIP`
- `destinationPort`
- `routingProtocol`
- `enabled`
- `severity`
- `facility`
- `searchFilterName`
- `compressionEnabled`
- `encryptionEnabled`
- `authenticationEnabled`
- `changeNameTo` (updateMsgRouting and updateMsgRoutingRemote only)

If `resultCount` is 0 and `statusCode` is not 2000 (successful), an error is returned in `statusMessage`.

### Status Codes

The Status Codes are:

<table>
<thead>
<tr>
<th>Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>2000</td>
<td>Server success</td>
</tr>
<tr>
<td>4000</td>
<td>Unauthorized request</td>
</tr>
<tr>
<td>5000</td>
<td>Invalid parameter, <code>getStatusMessage()</code> contains detail information about the error</td>
</tr>
</tbody>
</table>
Common Request Parameters

A set of Common Request Parameters are required for each Message Routing Service operation. Table 14 lists all of the Message Routing Service Common Request Parameters. For more information on the parameters, view the LogLogic Appliance user interface for message routing as well as the online help.

Table 14  Message Routing Service Common Request Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
<th>Values</th>
<th>Required</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>authToken</td>
<td>Unique authentication token.</td>
<td></td>
<td>yes</td>
<td>string</td>
</tr>
<tr>
<td>applianceIp</td>
<td>The managed Appliance on which you perform the operation. If the value is blank, it retrieves the Appliance IP address from the local Appliance. This parameter is available only for Management Station Appliances using operations with Remote in the name.</td>
<td>IP address of a managed Appliance. To specify an IP address, use the standard IP address format. For example: 10.1.2.3</td>
<td>yes</td>
<td>string</td>
</tr>
<tr>
<td>routingName</td>
<td>Name of the routing or forwarding rule.</td>
<td>Any text up to 64 characters in length.</td>
<td>yes</td>
<td>string</td>
</tr>
<tr>
<td>sourceDevice</td>
<td>Source device from where messages are routed.</td>
<td>Any text up to 64 characters in length.</td>
<td>yes</td>
<td>string</td>
</tr>
<tr>
<td>destinationIP</td>
<td>IP address for the message forwarding.</td>
<td>Standard IP address format. For example: 10.1.2.3</td>
<td>yes</td>
<td>string</td>
</tr>
<tr>
<td>destinationPort</td>
<td>Port for the message forwarding.</td>
<td>Any positive integer.</td>
<td>yes</td>
<td>number</td>
</tr>
<tr>
<td>routingProtocol</td>
<td>Protocol used for message forwarding.</td>
<td>Possible values: UDP, Syslog, Raw TCP, Syslog, SNMP, LogLogic TCP For more information on each protocol, see the online help.</td>
<td>yes</td>
<td>string</td>
</tr>
<tr>
<td>enabled</td>
<td>Enables message routing. (Appears as enabled or disabled in returned value.)</td>
<td>Possible values: yes — enable, No — disable</td>
<td>yes</td>
<td>string</td>
</tr>
<tr>
<td>severity</td>
<td>Message type used to filter messages</td>
<td>Possible values: All, Emergency, Alert, Critical, Error, Warning, Notice, Informational, Debug For more information on each severity level, see the online help.</td>
<td>no</td>
<td>string</td>
</tr>
</tbody>
</table>
Table 14  Message Routing Service Common Request Parameters (Continued)

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
<th>Values</th>
<th>Required</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>facility</td>
<td>The facility specifies the subsystem that produced the message. For example, all mail programs log with the mail facility (LOG_MAIL) if they log using syslog.</td>
<td>Possible values: auth, auth2, Clock, Clock2, FTP, Kernel, Local0, Local1, Local3, Local4, Local5, Local6, Local7, Log Alert, Log Audit, Mail, News, NTP, Printer, Syslog, System, User-Level, UUCP For more information on each facility, see the online help.</td>
<td>no</td>
<td>string</td>
</tr>
<tr>
<td>searchFilterName</td>
<td>Name of the search filter.</td>
<td>Any text up to 64 characters in length.</td>
<td>yes</td>
<td>string</td>
</tr>
<tr>
<td>compression</td>
<td>Enables or disables compression for LogLogic TCP message routing.</td>
<td>Possible values: yes — enable, no — disable</td>
<td>yes</td>
<td>string</td>
</tr>
<tr>
<td>encryptionEnabled</td>
<td>Enables or disables encryption of the message routing rule.</td>
<td>Possible values: yes — enable, no — disable</td>
<td>yes</td>
<td>string</td>
</tr>
<tr>
<td>authentication</td>
<td>Enables and disables authentication of the log message.</td>
<td>Possible values: yes — enable, no — disable</td>
<td>yes</td>
<td>string</td>
</tr>
<tr>
<td>changeNameTo</td>
<td>New name of the routing or forwarding rule.</td>
<td>Any text up to 64 characters in length.</td>
<td>yes</td>
<td>string</td>
</tr>
</tbody>
</table>
CHAPTER 6:

Search Filter Service Operations

The Search Filter Service operations enable you to manage search filters in your LogLogic Appliance.

**Overview**

The Search Filter Service operations let you create, read, update, and delete search filters as well as view all search filters in your LogLogic Appliance.

Search filters let you define search patterns to view specific data and to define alerts. In the LogLogic Appliance you can view the user interface implementation by navigating to **Search > Search Filters**.

**Implementation Guidelines**

The following are general implementation guidelines for the Search Filter Service operations:

- A set of Common Request Parameters (on page 85) are required for each operation. You must specify a value for all Required parameters.
- You must specify the value of the Common Request Parameters only.
- All parameters must be implemented in the order in which they appear in the Request Parameters section for each of the operations.
- All values for Common Request Parameters must be enclosed by double quotation marks ("value"). If the name of a device or a parameter value includes a forward slash (/), such as HP/UX, or IBM i5/OS, you must replace the forward slash with %2F. (The F is case-sensitive.)

Examples: HP%2FUX or IBM i5%2FOS

To search on the expression %2F itself, replace the % with %25. The resulting expression to use is %252F.
Search Filter Service Operation Descriptions

The Search Filter Service operations let you create, read, update, and delete search filters in your Appliance. In the LogLogic Appliance user interface, you can view all actions accessible through the Search Filter Service operations by navigating to the Search Filter screen from the navigation tree.

There are two kinds of operations:

- **local** - operation is performed on the local Appliance itself
- **remote** - operations (names ending with `Remote`) performed on a specified remote Appliance

- `createSearchFilter Operation` .................................................. 80
- `createSearchFilterRemote Operation` .................................... 81
- `readSearchFilter Operation` .................................................... 81
- `readSearchFilterRemote Operation` ....................................... 81
- `updateSearchFilter Operation` ................................................ 82
- `updateSearchFilterRemote Operation` ..................................... 82
- `deleteSearchFilter Operation` ................................................ 83
- `deleteSearchFilterRemote Operation` ..................................... 83
- `getList Operation` ............................................................... 83
- `getListRemote Operation` ..................................................... 84
- `searchFilterResponse Type` .................................................. 84

**createSearchFilter Operation**

The `createSearchFilter` operation lets you create search filters with associated search terms.

**Request Parameters**

- `authToken`, `searchFilterName`, `searchFilterType`, `description`, `sharedWithOtherUsers`, `expression1`, `expression2`, `expressionOperator`

For more information on each Common Request Parameter, see *Common Request Parameters* on page 85.

**Response**

- `searchFilterResponse` (see `searchFilterResponse Type` on page 84)

**Example**

To add a search filter named My Search Filter Name using the BooleanSearchFilter1 filter, of the Boolean Search type, described as My Search Filter Description, sharing the filter with other users, and searching on the term failed:
createSearchFilter authstr "My Search Filter Name"
"BooleanSearchFilter1" "Boolean Search" "My Search Filter Description" "yes" "failed" "" ""

**createSearchFilterRemote Operation**

The `createSearchFilterRemote` operation lets you create search filters with associated search terms on a managed LogLogic Appliance from a Management Station.

**Request Parameters**

`authToken`, `applianceIP`, `searchFilterName`, `searchFilterType`,
`description`, `sharedWithOtherUsers`, `expression1`, `expression2`,
`expressionOperator`

For more information on each Common Request Parameter, see Common Request Parameters on page 85.

**Response**

`searchFilterResponse` (see `searchFilterResponse Type` on page 84)

**Example**

To add a search filter (to remote Appliance 1.2.20.100) named My Search Filter Name using the BooleanSearchFilter1 filter, of the Boolean Search type, described as My Search Filter Description, sharing the filter with other users, and searching on the term failed:

createSearchFilterRemote authstr 1.2.20.100 "My Search Filter Name"
"BooleanSearchFilter1" "Boolean Search" "My Search Filter Description" "yes" "failed" "" ""

**readSearchFilter Operation**

The `readSearchFilter` operation lets you view the details of a specific search filter defined in your LogLogic Appliance.

**Request Parameters**

`authToken`, `searchFilterName`

**Response**

`searchFilterResponse` (see `searchFilterResponse Type` on page 84)

**Example**

To view the details of the search filter named My Search Filter Name:

readSearchFilter authstr "My Search Filter Name"

**readSearchFilterRemote Operation**

The `readSearchFilterRemote` operation lets you view the details of a specific search filter defined on a managed LogLogic Appliance from a Management Station.
**Request Parameters**
authToken, applianceIP, searchFilterName

**Response**
searchFilterResponse (see *searchFilterResponse Type* on page 84)

**Example**
To view the details of the search filter named My Search Filter Name on remote Appliance 1.2.20.100:
readSearchFilterRemote authstr "My Search Filter Name"

**updateSearchFilter Operation**
The updateSearchFilter operation lets you update an existing search filter.

**Request Parameters**
authToken, searchFilterName, searchFilterType, description, sharedWithOtherUsers, expression1, expression2, expressionOperator, changeNameTo

**Response**
searchFilterResponse (see *searchFilterResponse Type* on page 84)

**Example**
To update an existing search filter named My Search Filter Name to My New Search Filter Name:
updateSearchFilter authstr "My Search Filter Name"
"BooleanSearchFilter1" "Boolean Search" "My Search Filter Description" "yes" "failed" "" ""My New Search Filter Name"

**updateSearchFilterRemote Operation**
The updateSearchFilterRemote operation lets you update an existing search filter on a managed LogLogic Appliance from a Management Station.

**Request Parameters**
authToken, applianceIP, searchFilterName, searchFilterType, description, sharedWithOtherUsers, expression1, expression2, expressionOperator, changeNameTo

**Response**
searchFilterResponse (see *searchFilterResponse Type* on page 84)
Example
To update an existing search filter named My Search Filter Name to My New Search Filter Name on remote Appliance 1.2.20.100:
updateSearchFilterRemote authstr 1.2.20.100 "My Search Filter Name" "BooleanSearchFilter1" "Boolean Search" "My Search Filter Description" "yes" "failed" "" "My New Search Filter Name"

deleteSearchFilter Operation

The deleteSearchFilter operation lets you remove an existing search filter from your LogLogic Appliance.

Request Parameters
authToken, searchFilterName

Response
searchFilterResponse (see searchFilterResponse Type on page 84)

Example
To remove the search filter named My Search Filter Name from the Appliance:
deleteSearchFilter authstr "My Search Filter Name"

deleteSearchFilterRemote Operation

The deleteSearchFilterRemote operation lets you remove an existing search filter on a managed LogLogic Appliance from a Management Station.

Request Parameters
authToken, applianceIP, searchFilterName

Response
searchFilterResponse (see searchFilterResponse Type on page 84)

Example
To remove the search filter named My Search Filter Name from remote Appliance 1.2.20.100:
deleteSearchFilterRemote authstr 1.2.20.100 "My Search Filter Name"

getList Operation

The getList operation lets you retrieve a list of all search filters currently configured in the LogLogic Appliance.

Request Parameters
authToken
Response

If resultCount is greater than 0 and statusCode is 2000 (successful), the response returns a list of all search filters (total number indicated by resultCount) currently configured in the LogLogic Appliance.

If resultCount is 0 and statusCode is not 2000 (successful), an error is returned in statusMessage.

Example

To view a list of all search filters on the Appliance:

g getList authstr

getListRemote Operation

The getListRemote operation lets you retrieve a list of all search filters currently configured on a managed LogLogic Appliance from a Management Station.

Request Parameters

authToken, applianceIP

Response

If resultCount is greater than 0 and statusCode is 2000 (successful), the response returns a list of all search filters (total number indicated by resultCount) currently configured in the remote LogLogic Appliance.

If resultCount is 0 and statusCode is not 2000 (successful), an error is returned in statusMessage.

Example

To view a list of all search filters on remote Appliance 1.2.20.100:

g getListRemote authstr 1.2.20.100

searchFilterResponse Type

searchFilterResponse is returned for all searchFilter operations except getList and getListRemote.

searchFilterResponse always contains the following common elements:

- authToken
- resultCount
- statusCode
- statusMessage
- summaryOnly
If `resultCount` is 1 and `statusCode` is 2000 (successful), the `resultSet` element is included after `resultCount` listing the following Search Filter details:

- `expression1`
- `expression2`
- `expressionOperator`
- `expressionType`
- `filterDescription`
- `filterName`
- `sharedWithOtherUsers`

If `resultCount` is 0 and `statusCode` is not 2000 (successful), an error is returned in `statusMessage`.

### Status Codes

The Status Codes are:

<table>
<thead>
<tr>
<th>Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>2000</td>
<td>Server success</td>
</tr>
<tr>
<td>4000</td>
<td>Unauthorized request</td>
</tr>
<tr>
<td>5000</td>
<td>Invalid parameter, <code>getStatusMessage()</code> contains detail information about the error</td>
</tr>
</tbody>
</table>

### Common Request Parameters

A set of Common Request Parameters are required for each Search Filter operation. Table 15 lists all of the Search Filter parameters. For more information on the parameters, view the LogLogic Appliance user interface for search filters as well as the online help.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
<th>Default Value</th>
<th>Required</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>authToken</code></td>
<td>Unique authentication token.</td>
<td></td>
<td>yes</td>
<td>string</td>
</tr>
<tr>
<td><code>applianceIp</code></td>
<td>The managed Appliance on which you perform the operation.</td>
<td>IP address of a managed Appliance. To specify an IP address, use the standard IP address format. For example: 10.1.2.3</td>
<td>yes</td>
<td>string</td>
</tr>
<tr>
<td><code>searchFilterName</code></td>
<td>Name of the search filter.</td>
<td>Any text up to 64 characters in length.</td>
<td>yes</td>
<td>string</td>
</tr>
<tr>
<td><code>searchFilterType</code></td>
<td>Type of search filter.</td>
<td>Use Words, Use Exact Phrase, Regular Expression, or Boolean Search</td>
<td>yes</td>
<td>string</td>
</tr>
<tr>
<td><code>description</code></td>
<td>Description of the search filter name (searchFilterName parameter)</td>
<td>Any text up to 64 characters in length.</td>
<td>yes</td>
<td>string</td>
</tr>
<tr>
<td><code>sharedWithOtherUsers</code></td>
<td>Identifies if this search filter is available to other users. If yes, it is shared with others.</td>
<td>yes</td>
<td>yes</td>
<td>string</td>
</tr>
</tbody>
</table>
### Table 15  Search Filter operation Common Request Parameters (Continued)

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
<th>Default Value</th>
<th>Required</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>expression1</td>
<td>The actual search terms for the search filter.</td>
<td>Any valid search criteria up to 276 characters in length.</td>
<td>yes</td>
<td>string</td>
</tr>
<tr>
<td>expression2</td>
<td>The actual search terms for the search filter.</td>
<td>Any valid search criteria up to 270 characters in length.</td>
<td>yes</td>
<td>string</td>
</tr>
<tr>
<td>expressionOperator</td>
<td>The actual search terms for the search filter.</td>
<td></td>
<td>yes</td>
<td>string</td>
</tr>
<tr>
<td>changeNameTo</td>
<td>New name of the search filter.</td>
<td>Any text up to 64 characters in length.</td>
<td>yes (for updateSearchFilter and updateSearchFilterRemote only)</td>
<td>string</td>
</tr>
</tbody>
</table>
CHAPTER 7:

User Administration Service Operations

The User Administration Service operations enable you to manage users in your LogLogic Appliance.

- Overview ................................................................. 87
- Implementation Guidelines ........................................... 87
- User Administration Service Operation Descriptions .......... 87
- Status Codes ............................................................ 93
- Common Request Parameters ....................................... 94

Overview

The User Administration Service operations enable you to create, read, update, and delete users in your system. In the LogLogic Appliance you can view the user interface implementation by navigating to Administration > Manage Users.

Implementation Guidelines

The following are general implementation guidelines for the User Administration Service operations:

- A set of Common Request Parameters (on page 94) are required for each of the operations. You must specify a value for all Required parameters.
- You must specify the value of the Common Request Parameters only.
- All parameters must be implemented in the order in which they appear in the Request Parameters section for each of the operations.
- Use forward slash marks as delimiters when specifying multiple values for the privileges Common Request Parameter. For example:
  “valueA/valueB/ValueD”
- All values for Common Request Parameters must be enclosed by double quotation marks (“value”). If the name of a device or a parameter value includes a forward slash (/), such as HP/UX, or IBM i5/OS, you must replace the forward slash with %2F. (The F is case-sensitive.)
  Examples: HP%2FUX or IBM i5%2FOS

User Administration Service Operation Descriptions

There are two kinds of operations:

- local - operation is performed on the local Appliance itself
remote - operations (names ending with Remote) performed on a specified remote Appliance

The User Administration Service operation contains the following operations:

- createUser Operation ........................................ 88
- createUserRemote Operation .................................. 88
- readUser Operation .................................................. 89
- readUserRemote Operation ........................................ 89
- updateUser Operation ............................................ 90
- updateUserRemote Operation .................................... 90
- deleteUser Operation ............................................... 90
- deleteUserRemote Operation ...................................... 91
- getList Operation .................................................. 91
- getListRemote Operation .......................................... 92
- userResponse Type .................................................. 92

**createUser Operation**

The createUser operation lets you create new users in the LogLogic Appliance.

**Request Parameters**

authToken, loginName, firstName, lastName, loginPassword, phone, email, enabled, privileges, devices, appliances

For more information on each Common Request Parameter, see Common Request Parameters on page 94.

**Response**

userResponse (see userResponse Type on page 92)

**Example**

To create a user named MyFirstName MyLastName with email address me@loglogic.com, phone number 1-777-7777, authenticating locally, with an initial password MyPassword, and with Report Administrator access to all general syslog devices:

createUser authstr "me" "MyFirstName" "MyLastName" "MyPassword" "1-777-7777" "me@loglogic.com" "yes" "Report Administrator" "All General Syslog" ""

**createUserRemote Operation**

The createUserRemote operation lets you create new users on a managed LogLogic Appliance from a Management Station.
**Request Parameters**

authToken, applianceIP, loginName, firstName, lastName, loginPassword, phone, email, enabled, privileges, devices, appliances

For more information on each Common Request Parameter, see *Common Request Parameters* on page 94.

**Response**

userResponse (see *userResponse Type* on page 92)

**Example**

To create a user (on remote Appliance 1.2.20.100) named MyFirstName MyLastName with email address me@loglogic.com, phone number 1-777-7777, authenticating locally, with an initial password MyPassword, and with Report Administrator access to all general syslog devices:

creatUserRemote authstr 1.2.20.100 "me" "MyFirstName" "MyLastName" "MyPassword" "1-777-7777" "me@loglogic.com" "yes" "Report Administrator" "All General Syslog" ""

**readUser Operation**

The readUser operation lets you view the configuration of an existing user.

**Request Parameters**

authToken, loginName

**Response**

userResponse (see *userResponse Type* on page 92)

**Example**

To view the configuration of user UserName:

readMsgRouting authstr "UserName"

**readUserRemote Operation**

The readUserRemote operation lets you view the configuration of an existing user on a managed LogLogic Appliance from a Management Station.

**Request Parameters**

authToken, applianceIP, loginName

**Response**

userResponse (see *userResponse Type* on page 92)

**Example**

To view the configuration of user UserName on remote Appliance 1.2.20.100:
updateUser Operation

The updateUser operation lets you update the configuration of an existing user.

**Request Parameters**

authToken, loginName, firstName, lastName, loginPassword, phone, email, enabled, privileges, devices, appliances, changeLoginNameTo

**Response**

userResponse (see *userResponse Type* on page 92)

**Example**

To update the createuser example, by removing the Configuration Administrator as a privilege for the user:

```
updateUser authstr "me" "MyFirstName" "MyLastName" "MyPassword"
"1-777-7777" "me@loglogic.com" "yes" "Report Administrator" "all"
"" ""
```

updateUserRemote Operation

The updateUserRemote operation lets you update the configuration of an existing user. on a managed LogLogic Appliance from a Management Station.

**Request Parameters**

authToken, applianceIP, loginName, firstName, lastName, loginPassword, phone, email, enabled, privileges, devices, appliances, changeLoginNameTo

**Response**

userResponse (see *userResponse Type* on page 92)

**Example**

To update the createuserRemote example, by removing the Configuration Administrator as a privilege for the user on remote Appliance 1.2.20.100:

```
updateUserRemote authstr 1.2.20.100 "me" "MyFirstName" "MyLastName"
"MyPassword" "1-777-7777" "me@loglogic.com" "yes" "Report Administrator" "all"
"" ""
```

deleteUser Operation

The deleteUser operation lets you delete an existing user.

**Request Parameters**

authToken, loginName
Response
userResponse (see userResponse Type on page 92)

Example
To delete user UserName from the Appliance:
deleteUser authstr “UserName”

deleteUserRemote Operation
The deleteUser operation lets you delete an existing user on a managed LogLogic Appliance from a Management Station.

Request Parameters
authToken, applianceIP, loginName

Response
userResponse (see userResponse Type on page 92)

Example
To delete user UserName from remote Appliance 1.2.20.100:
deleteUserRemote authstr 1.2.20.100 “UserName”

ggetList Operation
The getList operation lets you retrieve a list of all users currently configured in the LogLogic Appliance.

Request Parameter
authToken

Response
If resultCount is greater than 0 and statusCode is 2000 (successful), the response returns a list of all users (total number indicated by resultCount) currently configured in the LogLogic Appliance.

If resultCount is 0 and statusCode is not 2000 (successful), an error is returned in statusMessage.

Example
To view a list of all users configured on the Appliance:
ggetList authstr
**getListRemote Operation**

The `getListRemote` operation lets you retrieve a list of all users currently configured on a managed LogLogic Appliance from a Management Station.

**Request Parameter**

- authToken, applianceIP

**Response**

If `resultCount` is greater than 0 and `statusCode` is 2000 (successful), the response returns a list of all users (total number indicated by `resultCount`) currently configured in the remote LogLogic Appliance.

If `resultCount` is 0 and `statusCode` is not 2000 (successful), an error is returned in `statusMessage`.

**Example**

To view a list of all users configured on remote Appliance 1.2.20.100:

```
getListRemote authstr 1.2.20.100
```

**userResponse Type**

`userResponse` is returned for all user operations except `getList` and `getListRemote`.

`userResponse` always contains the following common elements:

- authToken
- `resultCount`
- `statusCode`
- `statusMessage`
- summaryOnly
If `resultCount` is 1 and `statusCode` is 2000 (successful), the `resultSet` element is included after `resultCount` listing the following User details:

- `loginName`
- `firstName`
- `lastName`
- `loginPassword`
- `phone`
- `email`
- `enabled`
- `privileges`
- `devices`
- `appliances` (createUser, createUserRemote, updateUser, and updateUserRemote only)
- `changeLoginName` (updateUser and updateUserRemote only)

If `resultCount` is 0 and `statusCode` is not 2000 (successful), an error is returned in `statusMessage`.

**Status Codes**

The Status Codes are:

<table>
<thead>
<tr>
<th>Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>2000</td>
<td>Server success</td>
</tr>
<tr>
<td>4000</td>
<td>Unauthorized request</td>
</tr>
<tr>
<td>5000</td>
<td>Invalid parameter, <code>getStatusMessage()</code> contains detail information about the error</td>
</tr>
</tbody>
</table>
Common Request Parameters

A set of Common Request Parameters are required for each User Administration Service operation. Table 16 lists all of the User Administration Service Common Request Parameters. For more information on the parameters, view the LogLogic Appliance user interface for Manage Users as well as the online help.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
<th>Values</th>
<th>Required</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>authToken</td>
<td>Unique authentication token.</td>
<td></td>
<td>yes</td>
<td>string</td>
</tr>
<tr>
<td>applianceIp</td>
<td>The managed Appliance on which you perform the operation.</td>
<td>IP address of a managed Appliance. To specify an IP address, use the standard IP address format. For example: 10.1.2.3</td>
<td>yes</td>
<td>string</td>
</tr>
<tr>
<td>loginName</td>
<td>Login name for the user you are creating.</td>
<td>Any text up to 16 characters in length.</td>
<td>yes</td>
<td>string</td>
</tr>
<tr>
<td>firstName</td>
<td>First name of the user.</td>
<td>Any text up to 64 characters in length.</td>
<td>No</td>
<td>string</td>
</tr>
<tr>
<td>lastName</td>
<td>Last name of the user.</td>
<td>Any text up to 64 characters in length.</td>
<td>No</td>
<td>string</td>
</tr>
<tr>
<td>loginPassword</td>
<td>Password used for user login. In addition, you can use the loginPassword parameter to update/change a user’s password. For Active Directory (AD) users, enclose the password in single-quotes (‘) when it is an input parameter to a shell script. AD passwords can contain special characters (such as !, @, or *) which the shell does not properly interpret unless within single-quotes.</td>
<td>The password must be at least 6 characters long, contain at least one non-alphabetic character, and cannot be the same as the loginName.</td>
<td>yes</td>
<td>string</td>
</tr>
<tr>
<td>phone</td>
<td>Contact phone number of user.</td>
<td>Any valid phone number up to 32 characters in length.</td>
<td>No</td>
<td>string</td>
</tr>
<tr>
<td>email</td>
<td>Email address of the user.</td>
<td>Any valid email address up to 64 characters in length.</td>
<td>Yes</td>
<td>string</td>
</tr>
</tbody>
</table>
| enabled    | Enable or disable the user specified with the loginName parameter. (Appears as enabled or disabled in returned value.)                                                                                       | Possible values: yes — enable
no — disable                                                                                                                  | yes      | string   |
| privileges | Identifies the access privileges for the specified user. You can specify a User Type (contains a pre-defined list of privileges) or specific privileges.  
To view the user interface implementation, navigate to the Administration > Manage Users > Privileges tab.  
The Import/Export privilege contains a forward slash (/). To specify it here, replace / with %2F. | For a list of possible values, see User Administration Privileges on page 116.  
To specify multiple values, you must separate entries with a forward slash (/). For example: User Administrator/Manage Alerts/Manage Devices | yes      | string   |
### User Administration Service operations Common Request Parameters (Continued)

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
<th>Values</th>
<th>Required</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>devices</td>
<td>Identifies the devices that the user can access. You can specify All for provide access to all devices in the Appliance or you can identify specific devices. You can use the <code>getList Operation</code> on page 55 to retrieve a list of device names in your LogLogic Appliance. Alternatively, you can view use the user interface to view all devices by navigating to Administration &gt; Manage Devices. Possible values: All - all devices <code>deviceName</code> - specify each device name as it appears in the LogLogic Appliance. To specify multiple values, you must separate entries with a forward slash (/). For example: &quot;Device1/Device2/Device7/Device9&quot;</td>
<td>yes</td>
<td>string</td>
<td></td>
</tr>
<tr>
<td>appliances</td>
<td>(Management Station only) Identifies the Appliance in the Management Station cluster that this user accesses.</td>
<td>Any valid IP address</td>
<td>yes (for <code>createUser</code>, <code>createUser Remote</code>, <code>updateUser</code> and <code>updateUser Remote</code> only)</td>
<td>number</td>
</tr>
<tr>
<td>changeLogin Name</td>
<td>New login name for a user. If empty, the login name is unchanged.</td>
<td>Any text up to 64 characters in length.</td>
<td>yes (for <code>updateUser</code> and <code>updateUser Remote</code> only)</td>
<td>string</td>
</tr>
</tbody>
</table>
CHAPTER 8:

System Service

The System Service provides programmatic access to the system functionality on the LogLogic Appliance.

Overview

The System Service API provides these operations:

- `getApplianceSystemInfo`, which lets your program retrieve the system information on an Appliance.

Status Codes

The Status Codes are:

<table>
<thead>
<tr>
<th>Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>No error</td>
</tr>
<tr>
<td>1</td>
<td>No data available for the query parameters provided</td>
</tr>
<tr>
<td>5</td>
<td>Insufficient Privilege for requested operation</td>
</tr>
<tr>
<td>7</td>
<td>Invalid input</td>
</tr>
<tr>
<td>8</td>
<td>Persistence exception</td>
</tr>
</tbody>
</table>

Get Appliance System Info Operation

The `getApplianceSystemInfo` operation returns a result containing the system information on an Appliance.

Get Appliance System Info Request Parameters

The `getApplianceSystemInfo` operation request parameters are:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
<th>Values</th>
<th>Required</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>authToken</td>
<td>Unique authentication token.</td>
<td></td>
<td>Yes</td>
<td>String</td>
</tr>
<tr>
<td>applianceIp</td>
<td>The Appliance from which you retrieve the system information.</td>
<td>IP address. For example: 10.1.2.3</td>
<td>Yes</td>
<td>String</td>
</tr>
</tbody>
</table>
Get Appliance System Info Response Attributes

The `getApplianceSystemInfo` response attributes are:

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>applianceIp</td>
<td>Appliance IP. For example, &quot;10.0.80.14&quot;.</td>
<td>String</td>
</tr>
<tr>
<td>applianceModel</td>
<td>Appliance model. For example, &quot;LX2010&quot;.</td>
<td>String</td>
</tr>
<tr>
<td>applianceTitle</td>
<td>Appliance title. For example, &quot;LX 2010 (Management Station)&quot;.</td>
<td>String</td>
</tr>
<tr>
<td>applianceType</td>
<td>Appliance type. For example, &quot;LX&quot;.</td>
<td>String</td>
</tr>
<tr>
<td>applianceVersion</td>
<td>Appliance version. For example, &quot;5.0.0&quot;.</td>
<td>String</td>
</tr>
<tr>
<td>buildNumber</td>
<td>Build number. For example, &quot;201008071649\n&quot;.</td>
<td>String</td>
</tr>
<tr>
<td>lspVersion</td>
<td>LSP version. For example, &quot;17.1.201003171111&quot;.</td>
<td>String</td>
</tr>
<tr>
<td>lspVersionShort</td>
<td>LSP version short string. For example, &quot;17.1&quot;.</td>
<td>String</td>
</tr>
<tr>
<td>logLineDelimiter</td>
<td>Logline delimiter. For example, &quot;\r\n&quot;.</td>
<td>String</td>
</tr>
<tr>
<td>runningMode</td>
<td>Values are one of: &quot;ms&quot;, &quot;non-ms&quot;, &quot;strict-non-ms&quot;.</td>
<td>String</td>
</tr>
<tr>
<td>underRemoteControl</td>
<td>Indicate the appliance is under remote control or not.</td>
<td>Boolean</td>
</tr>
<tr>
<td>applianceTimeZoneId</td>
<td>Time zone ID. For example, “UTC”.</td>
<td>String</td>
</tr>
<tr>
<td>currentApplianceTimeEpoch</td>
<td>Current appliance time Epoch. For example, 1282241419646.</td>
<td>Number</td>
</tr>
<tr>
<td>applianceTimeZoneOffsetSeconds</td>
<td>Appliance timezone offset in seconds.</td>
<td>Number</td>
</tr>
<tr>
<td>errorCause</td>
<td>Error cause code of the error.</td>
<td>Number</td>
</tr>
<tr>
<td>errorCode</td>
<td>Error code for the request number.</td>
<td>Number</td>
</tr>
</tbody>
</table>
Part III: Report and Search Service
CHAPTER 9:

Report and Search Service

The Report and Search Service provides programmatic access to the Reporting and Index Search functionality on the LogLogic Appliance.

- Overview ................................................................. 101
- Status Codes .......................................................... 102
- Get Report Operation ................................................. 102
- Get Template Report List Operation ............................. 107
- Get Custom Report List Operation ............................... 107
- Get Report Meta Info Operation ................................. 108
- Get Detail Report Meta Info Operation ......................... 109

Overview

The Report and Search Service API provides these operations:

- getReport, which lets your program run a:
  - Real-Time report query
  - Pre-defined Custom Report query
  - Index Search query
- getTemplateReportList, which lets you list all Real-Time report templates on the Appliance.
- getCustomReportList, which lets you list all Custom Reports defined on the Appliance.
- getDetailReportMetaInfo, which lets you retrieve the type information for columns in a report on the Appliance via the given detail report token.
Status Codes

The Status Codes are:

<table>
<thead>
<tr>
<th></th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>No error</td>
</tr>
<tr>
<td>1</td>
<td>No data available for the query parameters provided</td>
</tr>
<tr>
<td>2</td>
<td>Invalid time range</td>
</tr>
<tr>
<td>3</td>
<td>Invalid report name</td>
</tr>
<tr>
<td>4</td>
<td>Unknown device</td>
</tr>
<tr>
<td>5</td>
<td>Insufficient Privilege for requested operation</td>
</tr>
<tr>
<td>6</td>
<td>Invalid device for report</td>
</tr>
<tr>
<td>7</td>
<td>Invalid input</td>
</tr>
<tr>
<td>8</td>
<td>Persistence exception</td>
</tr>
<tr>
<td>9</td>
<td>Invalid token reference for detail report operation</td>
</tr>
</tbody>
</table>

Get Report Operation

The `getReport` operation returns a result set containing the records from a Real-Time report query, custom report query, Index Search, or Index Report.

Get Report Request Parameters

The `getReport` operation common request parameters are:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
<th>Values</th>
<th>Required</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>authToken</td>
<td>Unique authentication token.</td>
<td></td>
<td>Yes</td>
<td>String</td>
</tr>
<tr>
<td>applianceIp</td>
<td>The Appliance from which you retrieve the report.</td>
<td>IP address of a managed Appliance or All for retrieving an aggregated report from all managed Appliances. To specify an IP address, use the standard IP address format. For example: 10.1.2.3</td>
<td>No</td>
<td>String</td>
</tr>
</tbody>
</table>
### Report and Search Service: Get Report Operation

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
<th>Values</th>
<th>Required</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>reportType</td>
<td>Type of report.</td>
<td>Possible values:</td>
<td>Yes</td>
<td>String</td>
</tr>
<tr>
<td></td>
<td></td>
<td>1—reportName is a Real-Time report template</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>3—reportName is a name of the custom report</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>4—Report is a detail report and the detail report token is specified</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>in the filters</td>
<td></td>
<td></td>
</tr>
<tr>
<td>reportName</td>
<td>Name of the report template or custom report.</td>
<td>For index search:</td>
<td>Yes</td>
<td>String</td>
</tr>
<tr>
<td></td>
<td></td>
<td>SearchExpression</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Hits</td>
<td></td>
<td></td>
</tr>
<tr>
<td>sourceDeviceName</td>
<td>The name of the device, device group, or IP address to use in the report query.</td>
<td>Possible values:</td>
<td>Yes, except for custom reports</td>
<td>String</td>
</tr>
<tr>
<td></td>
<td></td>
<td>“ALL”, “ALL Cisco PIX”, or “10.1.1.80”</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>The default is “All” for all devices; except, for custom reports, the default is the device list stored in the custom report.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>timeRange</td>
<td>A predefined time range specification, or an indication to use the time range specified by the fromTime and toTime parameters.</td>
<td>Possible values:</td>
<td>No</td>
<td>Date</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Pre-defined — Last 1 Hour, Last 2 Hours, Last “n” Hours, Yesterday, Today</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Specific time— SpecificTime</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>The default is “Last 1 Hour”</td>
<td></td>
<td></td>
</tr>
<tr>
<td>fromTime</td>
<td>Start time for the time range of the report query.</td>
<td>See Date/Time Formats for getReport on page 105</td>
<td>No</td>
<td>Date</td>
</tr>
<tr>
<td></td>
<td></td>
<td>This parameter is required only if the value for timeRange is a SpecificTime.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
## Parameter Description | Values | Required | Type
--- | --- | --- | ---
**toTime** | End time for the time range of the report query. This parameter is required only if the value for `timeRange` is a `SpecificTime`. | See Date/Time Formats for `getReport` on page 105 | No | Date
**sortColumn** | Column to use as the records sorting key. | | No | String
**filters** | List of expressions applied to the query to receive records passing the filter criteria. For example, for the `activeConnections` report template, the filter "/direction=/INBOUND/" retrieves only inbound connection records. This is similar functionality as using a "where" clause in SQL. See the Advanced Options section for the report you specified using the `reportName` parameter. For examples of using filters for Index Search or Index Report, see Specifying Filters for Index Searches on page 105. For examples of using filters for Detail Report, see Specifying Filters for Detail Report on page 105. | Values must use the format: `Column/Operator/Value` | No | Array of String
**startRow** | Report query result row to use as the first row of the report result set. Use the `startRow` and `numberOfRowsToRetrieve` to retrieve a specific range of records. The first row of the query result is zero (0). | A numeric value. | Yes | Number
**numberOfRowsToRetrieve** | Maximum number of rows to return in the report result set. Use the `startRow` and `numberOfRowsToRetrieve` to retrieve a specific range of records. | A numeric value. | Yes | Number
**sortColumn** | Column in the generated report to sort on, and the direction of the sort (up or down). | Values must use the format: `column-name/direction` where `direction` is `up` or `down` | No | String
Specifying Filters for Index Searches

The following are examples of how to use the getReport filters parameter to specify search filters for Index Search or Index Report queries.

Index Search

There are two ways to specify search filters with Index Search:

- Specify a pre-defined search filter that contains a Boolean search expression:
  `/predefined/=Boolean-search-filter-name/`

- Specify a Boolean expression:
  `/fullTextSrchCriteria/=Boolean-search-expression/`
  For example, in the user interface for Index Search, you might specify “admin AND login” for the search terms. To specify the query using the Web Services API, add a filter with the string:
  `/fullTextSrchCriteria/=admin AND login/`

The Boolean expression is case-insensitive.

Index Report

The Index Report lets you specify multiple pre-defined Boolean search filters:

`/filter_selection/=search-filter_name1, search-filter_name2, ../`

Specifying Filters for Detail Report

When the target report is a summary report, an extra column, LLDetailTokenRef will be returned in the response. You can use this value to retrieve the detail report of a specified row that you want to drilldown. To do so, set the report type to 4 and provide the detail token in the filters parameter. The values in reportName and sourceDeviceName parameters are ignored in this report type. The following are examples of how to use the getReport filters parameter to specify filter for Detail Report queries:

`/LLDetailTokenRef/=Token value from previous getReport operation/`

Date/Time Formats for getReport

The following formats must be used when specifying fromTime and toTime to designate a specific timeRange for the getReport query.

Year:

`YYYY`

Example: 2009

Year and month:

`YYYY-MM`

Example: 2009-03
Complete date:

\[ YYYY-MM-DD \]

Example: 2009-03-16

Complete date plus hours and minutes:

\[ YYYY-MM-DDThh:mmTZD \]

Example: 2009-03-16T19:20+01:00

Complete date plus hours, minutes, and seconds:

\[ YYYY-MM-DDThh:mm:ssTZD \]

Example: 2009-03-16T19:20:30+01:00

Complete date plus hours, minutes, seconds, and a decimal fraction of a second:

\[ YYYY-MM-DDThh:mm:ss.sTZD \]

Example: 2009-03-16T19:20:30.45+01:00

where:

- \[ YYYY \] = four-digit year
- \[ MM \] = two-digit month (01=January, etc.)
- \[ DD \] = two-digit day of month (01 through 31)
- \[ T \] = literal delimiter in the string, indicating the beginning of the time element
- \[ hh \] = two digits of hour (00 through 23; am/pm are not allowed)
- \[ mm \] = two digits of minute (00 through 59)
- \[ ss \] = two digits of second (00 through 59)
- \[ s \] = one or more digits representing a decimal fraction of a second
- \[ TZD \] = time zone designator (Z or +hh:mm or -hh:mm)

**Get Report Response Attributes**

The `getReport` operation returns a tabular set of rows and columns similar to the UI-based report result.

**Note:** When the target report is a summary report, an extra column `LLDetailTokenRef` is added to `columnNames` and extra detail tokens are added to each record in `records` accordingly.

The attributes of a `LogLogicReport` are:

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>errorCode</code></td>
<td>Error code for the query failure.</td>
<td>Number</td>
</tr>
<tr>
<td><code>reportName</code></td>
<td>Name of the report.</td>
<td>String</td>
</tr>
<tr>
<td><code>startTime</code></td>
<td>Start time for the report query.</td>
<td>Date</td>
</tr>
<tr>
<td><code>endTime</code></td>
<td>End time for the report query.</td>
<td>Date</td>
</tr>
<tr>
<td><code>numberOfColumnsPerRow</code></td>
<td>Number of columns in each row in the result set.</td>
<td>Number</td>
</tr>
</tbody>
</table>
Get Template Report List Operation

The `getTemplateReportList` operation returns a list of the templates reports configured in the Appliance, identical to the Real-Time Reports in the Appliance navigation menu.

Get Template Report List Request Parameter

The `getTemplateReportList` operation request parameter are:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
<th>Values</th>
<th>Required</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>authToken</td>
<td>Unique authentication token.</td>
<td></td>
<td>Yes</td>
<td>String</td>
</tr>
</tbody>
</table>

Get Template Report List Response Attributes

The `getTemplateReportList` response attributes are:

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>errorCode</td>
<td>Error code for the request number.</td>
<td>Number</td>
</tr>
<tr>
<td>recordCount</td>
<td>Number of custom report names.</td>
<td>Number</td>
</tr>
<tr>
<td>reportList</td>
<td>An array of <code>reportTitle</code> (string) and <code>reportType</code> (string). The <code>reportType</code> value can be used in the <code>getReport</code> operation’s <code>reportName</code> parameter.</td>
<td>Array</td>
</tr>
</tbody>
</table>

Get Custom Report List Operation

The `getCustomReportList` operation returns a list of the custom reports configured in the Appliance.
Get Custom Report List Request Parameter

The `getCustomReportList` operation request parameter are:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
<th>Values</th>
<th>Required</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>authToken</td>
<td>Unique authentication token.</td>
<td></td>
<td>Yes</td>
<td>String</td>
</tr>
</tbody>
</table>

Get Custom Report List Response Attributes

The `getCustomReportList` response attributes are:

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>errorCode</td>
<td>Error code for the request number.</td>
<td>Number</td>
</tr>
<tr>
<td>recordCount</td>
<td>Number of custom report names.</td>
<td>Number</td>
</tr>
<tr>
<td>reportList</td>
<td>An array of <code>reportTitle</code> (string) and <code>reportType</code> (string). The custom report's <code>reportTitle</code> value can be used in the <code>getReport</code> operation's <code>reportName</code> parameter.</td>
<td>Array</td>
</tr>
</tbody>
</table>

Get Report Meta Info Operation

The `getReportMetaInfo` operation retrieves the report column metadata for use together with the results of a `getReport` call using the same parameters. When the target report is a summary report, an extra column `LLDetailTokenRef` is added to `columnInfo` to indicate that it is a summary report.

**Note:** Only displayed columns in a custom report will be returned, not all columns.

Get Report Meta Info Request Parameter

The `getReportMetaInfo` operation request parameters are:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
<th>Values</th>
<th>Required</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>authToken</td>
<td>Unique authentication token.</td>
<td></td>
<td>Yes</td>
<td>String</td>
</tr>
<tr>
<td>applianceIP</td>
<td>The Appliance from which you retrieve the metadata.</td>
<td>IP address of a managed Appliance or All for retrieving an aggregated report from all managed Appliances. To specify an IP address, use the standard IP address format. For example: 10.1.2.3</td>
<td>Yes</td>
<td>String</td>
</tr>
<tr>
<td>reportType</td>
<td>Type or report to run.</td>
<td>1 = Real Time report, 2 = Summary report, 3 = Custom report</td>
<td>Yes</td>
<td>Number</td>
</tr>
</tbody>
</table>

Report and Search Service : Get Report Meta Info Operation
Get Report Meta Info Response Attributes

The `getReportMetaInfo` response attributes are:

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>reportName</td>
<td>Name of the report to run.</td>
<td></td>
</tr>
<tr>
<td>sourceDeviceName</td>
<td>Name of the log source to include in the report.</td>
<td></td>
</tr>
<tr>
<td>errorCode</td>
<td>Error code for the request number.</td>
<td>Number</td>
</tr>
<tr>
<td>errorCause</td>
<td>Error cause code of the error.</td>
<td>Number</td>
</tr>
<tr>
<td>columnCount</td>
<td>Number of columns returned in the report.</td>
<td>Number</td>
</tr>
<tr>
<td>columnInfo</td>
<td>An array of <code>reportColumnInfo</code> (complex type). <code>reportColumnInfo</code> is a structure containing:</td>
<td>Array</td>
</tr>
<tr>
<td></td>
<td>■ displayName (string)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>■ shortDisplayName (string)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>■ internalName (string)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>■ columnType (string)</td>
<td></td>
</tr>
</tbody>
</table>

Get Detail Report Meta Info Operation

The `getDetailReportMetaInfo` command performs the same functionality as `getReportMetaInfo`. However, instead of `ReportType/LogSource`, it takes an `LLDetailTokenRef` string as input to retrieve the report column metadata.

**Note:** Only displayed columns in a detail report will be returned, not all columns.
Get Detail Report Meta Info Request Parameter

The `getDetailReportMetaInfo` operation request parameters are:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
<th>Values</th>
<th>Required</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>authToken</td>
<td>Unique authentication token.</td>
<td></td>
<td>Yes</td>
<td>String</td>
</tr>
<tr>
<td>applianceIP</td>
<td>The Appliance from which you retrieve the metadata.</td>
<td>IP address of a managed Appliance or All for retrieving an aggregated report from all managed Appliances. To specify an IP address, use the standard IP address format. For example: 10.1.2.3</td>
<td>Yes</td>
<td>String</td>
</tr>
<tr>
<td>LLDetailTokenRef</td>
<td>The detail report token reference.</td>
<td>The detail token string received in previous <code>getReport</code> operation.</td>
<td>Yes</td>
<td>String</td>
</tr>
</tbody>
</table>

Get Detail Report Meta Info Response Attributes

The `getDetailReportMetaInfo` response attributes are:

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>errorCode</td>
<td>Error code for the request number.</td>
<td>Number</td>
</tr>
<tr>
<td>errorCause</td>
<td>Error cause code of the error.</td>
<td>Number</td>
</tr>
<tr>
<td>columnCount</td>
<td>Number of columns returned in the report.</td>
<td>Number</td>
</tr>
<tr>
<td>columnIndex</td>
<td>An array of <code>reportColumnInfo</code> (complex type). <code>reportColumnInfo</code> is a structure containing: displayName (string)</td>
<td>Array</td>
</tr>
<tr>
<td></td>
<td>shortDisplayName (string)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>internalName (string)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>columnType (string)</td>
<td></td>
</tr>
</tbody>
</table>
Part IV: Utilities Services and Appendices
The Authentication Service gives the client program the ability to log in to the Appliance. LogLogic Web Services methods require an authentication token to validate that the client program has access to the Appliance. Instead of using a username and password (or basic authentication) with every Web Service call, the LogLogic Web Services infrastructure implements a “token” based authentication scheme. Token-based authentication schemes help prevent passwords from being exposed. Only the “token” needs to be included in code, or cached in memory by the client side application. Additionally, the authentication “token” is not applicable, or usable, by the Appliance’ user interface.

GetAuthenticationToken Operation

The GetAuthenticationToken lets the Web Services program authenticate and retrieve an authentication token from the Appliance.

Request Parameters

The GetAuthenticationToken request parameters are:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
<th>Default Value</th>
<th>Required</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>username</td>
<td>A valid user account on the Appliance with “Web Service Access” enabled.</td>
<td></td>
<td>No</td>
<td>String</td>
</tr>
<tr>
<td>password</td>
<td>The password for the user account specified by username.</td>
<td></td>
<td>No</td>
<td>String</td>
</tr>
</tbody>
</table>

Response

The attributes of a GetAuthenticationToken response is:

errorCode—An AuthenticationResult containing the authentication token string; an error code is included in the case of authentication failure. The response is number.

Status Codes

The Status Codes are:

<table>
<thead>
<tr>
<th>Status Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>No error</td>
</tr>
<tr>
<td>1</td>
<td>Invalid username and/or password</td>
</tr>
<tr>
<td>2</td>
<td>Invalid authorization token</td>
</tr>
</tbody>
</table>
APPENDIX A:

Reference Lists

- Alert Types .................................................. 115
- Message Areas ............................................. 116
- User Administration Privileges ....................... 116

Alert Types

The alert types are:

**Adapter Baseline**—Use the Adaptive Baseline Alert to notify you if message rates fall above or below your average baseline range for the specified day and time of the week.

**Cisco Pix Messages**—The Cisco PIX Messages alert allows for triggering on PIX message criticality, code, and message rate. Since this alert is specific to Cisco PIX messages, the alert device selection is limited to Cisco PIX devices.

**Message Volume**—The Message Volume-Based alert allows alerting when message volume falls below, or is above, preset messages-per-second thresholds. This alert applies to all devices.


*Important*: Network Policy Alert Rules (Rules Tab) are required for this alert to trigger.

**Pre-defined Search Filter**—The Pre-Defined Search Filter Alert allows for alert notification when a text search match occurs within the received log message. This alert leverages the Log Appliance search filters for the text search match definitions. To define the text match for the alert, use Search Filters on the navigation tree.

**Ratio Based**—The Ratio Based Alert triggers when the percentage of a specified message type exceeds or falls below specified percentages.

For example, the Denied/(Accept+Denied) Alert Ratio can be used to trigger an alert when the number of Denied messages exceeds 90% of the Accept and Denied message count.

**System**—The System Alert allows for notification when system health and status criteria exceed acceptable bounds.

**VPN Connections**—The VPN Connection Alert triggers when a VPN connection is denied access and/or disconnected. The VPN Connection alert is only applicable to Cisco VPN, Radius, and Nortel Contivity devices.
VPN Messages—The VPN Message Alert triggers on combinations of specific VPN message area, severity, and code. This alert is applicable to Cisco VPN devices.

VPN Statistics—The VPN Statistics Alert triggers when recorded statistics on VPN or Radius messages match relative or absolute criteria.

For example, you can configure an alert to trigger when the Number of Bytes Received per day for a specific user exceeds 1Mb per day, which is an absolute value. The alert rule can also be configured as a relative rule, such as “grows by 10%.”

Message Areas

The following is a complete list of available message areas that you can define in the messageArea parameter for the VPN Messages Alert. You can also view the list in the LogLogic Appliance user interface.

ANY, AUTH, AUTHDBG, AUTHDECODE, AUTOUPDATE, BKPLN, BMGT, BMGTDDBG, BUFFER, CAPI, CERT, CIFS, CIFSDBG, CLIENT, CONFIG, DHCP, DHCPDBG, DHCPDECODE, DIAG, DM, DNS, DNSDBG, DNSDECODE, EMAILPROXY, EVENT, EVENTDBG, EVENTMIB, EXPANSIONCARD, FILTER, FILTERDBG, FIPS, FIPSDECODE, FSM, FTPD, FW, FWDBG GWDECODE, GENERAL, GRE, GREDBG, GREDECODE, H323, H323DBG, HARDWAREMON, HDLC, HTTP, HWDIAG, IKE, IKEDBG, IP, IPDBG, IPDECODE, IPSEC, IPSECDBG, IPSECDECODE, L2TP, L2TPDBG, LBSS, MIB2TRAP, NETBIOS, OSPF, PPP, PPPDBG, PPPDECODE, PPPOE, PPPOEDBG, PPPOEDECODE, PPTP, PPTPDBG, PPTPDECODE, PSH, PSOS, QUEUE, RADIUSACCT, REBOOT, RM, SMTP, SNMP, SSH, SSL, SYSTEM, T1E1, TCP TELNET, TELNETDBG, TELNETDECODE, TIME, VRRP, WAN, WEBVPN

User Administration Privileges

The following table lists the User Type (User Admin, Report Admin, and Config Admin) and the associated privileges. You can specify either the UserType and/or the associated privileges.

<table>
<thead>
<tr>
<th>User Type</th>
<th>Privileges Granted</th>
</tr>
</thead>
<tbody>
<tr>
<td>Administrator</td>
<td>All privileges</td>
</tr>
<tr>
<td>User Admin</td>
<td>Manage Users</td>
</tr>
<tr>
<td></td>
<td>Manage Administrators</td>
</tr>
<tr>
<td></td>
<td>Replicate Users</td>
</tr>
</tbody>
</table>

116 Web Services API Implementation Guide
<table>
<thead>
<tr>
<th>User Type</th>
<th>Privileges Granted</th>
</tr>
</thead>
<tbody>
<tr>
<td>Report Admin</td>
<td>Custom Reports</td>
</tr>
<tr>
<td></td>
<td>Real-Time Reports</td>
</tr>
<tr>
<td></td>
<td>Summary Reports</td>
</tr>
<tr>
<td></td>
<td>Search Archived Data</td>
</tr>
<tr>
<td></td>
<td>Replay</td>
</tr>
<tr>
<td>Config Admin</td>
<td>Manage Devices</td>
</tr>
<tr>
<td></td>
<td>Port Configuration</td>
</tr>
<tr>
<td></td>
<td>Message Routing Configuration</td>
</tr>
<tr>
<td></td>
<td>Manage Alerts</td>
</tr>
<tr>
<td></td>
<td>Manage Check Point Devices</td>
</tr>
<tr>
<td></td>
<td>Manage PIX Msg Codes</td>
</tr>
<tr>
<td></td>
<td>System Configuration</td>
</tr>
<tr>
<td></td>
<td>Access control</td>
</tr>
<tr>
<td></td>
<td>Backup/NAS Configuration</td>
</tr>
<tr>
<td></td>
<td>Access Management station</td>
</tr>
<tr>
<td></td>
<td>Manage File Transfer Rules</td>
</tr>
<tr>
<td></td>
<td>Import/Export</td>
</tr>
<tr>
<td></td>
<td>Manage Packages</td>
</tr>
</tbody>
</table>