



HP MSM313/MSM323 Integrated Services Access Points

Network Access Configuration Guide

HP MSM313/MSM323 Integrated Services Access Points

Network Access Configuration Guide

© Copyright 2010 Hewlett-Packard Development Company, L.P. The information contained herein is subject to change without notice.

This document contains proprietary information, which is protected by copyright. No part of this document may be photocopied, reproduced, or translated into another language without the prior written consent of Hewlett-Packard.

Publication Number

5998-0449

July 2010

Trademark Credits

Windows NT®, Windows®, and MS Windows® are US registered trademarks of Microsoft Corporation.

Disclaimer

HEWLETT-PACKARD COMPANY MAKES NO WARRANTY OF ANY KIND WITH REGARD TO THIS MATERIAL, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. Hewlett-Packard shall not be liable for errors contained herein or for incidental or consequential damages in connection with the furnishing, performance, or use of this material.

The only warranties for Hewlett-Packard products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. Hewlett-Packard shall not be liable for technical or editorial errors or omissions contained herein.

Hewlett-Packard assumes no responsibility for the use or reliability of its software on equipment that is not furnished by Hewlett-Packard.

Warranty

See the Customer Support/Warranty booklet included with the product. A copy of the specific warranty terms applicable to your Hewlett-Packard products and replacement parts can be obtained from your Hewlett-Packard Sales and Service Office or authorized dealer.

Open Source Software Acknowledgement Statement

This software incorporates open source components that are governed by the GNU General Public License (GPL), version 2. In accordance with this license, Hewlett-Packard will make available a complete, machine-readable copy of the source code components covered by the GNU GPL upon receipt of a written request. Send a request to:

Hewlett-Packard Company, L.P. GNU GPL Source Code
Attn: HP Support
Roseville, CA 95747 USA

www.hp.com

Contents

Chapter 1

Introduction 5

About this guide.....	6
Products covered.....	6
Important terms.....	6
Conventions.....	7
Cautions.....	7
Public/guest network access deployment.....	7
About the public and protected networks.....	8
Network access.....	8
Sample deployment.....	10
Contacting support.....	10
Online documentation.....	10

Chapter 2

Customizing the public access network 11

Configuring the public access network.....	12
Activating the public access network.....	12
Defining and retrieving service controller attributes.....	12
Retrieve attributes using RADIUS.....	13
Configured attributes.....	14
Attribute summary.....	15
Standard RADIUS attributes.....	15
Vendor-specific attributes.....	16
WISPr vendor-specific attributes.....	17
HP vendor-specific attributes.....	17
Maximum attribute size.....	19
Acct-Terminate-Cause values.....	20
Service controller attributes.....	21
Standard RADIUS attributes.....	21
Access request.....	21
Access accept.....	22
Access reject.....	22
Access challenge.....	22
Accounting request.....	22
Accounting response.....	23
HP vendor-specific attributes.....	24
Access lists.....	25
Default setting.....	25
How access lists work.....	25
Accounting support.....	26
Tips on using the access list.....	26
Defining access lists.....	27
Activating access lists.....	27
Access list example.....	29
Redirect URL.....	31
Custom SSL certificate.....	33
Configuration file.....	34
MAC authentication.....	35
Default user idle timeout.....	36
Default user session timeout.....	36
Default user SMTP server.....	37
Default user interim accounting update interval.....	37
Default user one-to-one NAT.....	38
Default user quotas.....	38
Multiple login servers.....	39
Traffic forwarding (dnat-server).....	40
Multiple DNAT servers.....	41
User configuration attributes.....	43
Standard RADIUS attributes.....	43
Access request.....	43

Access accept.....	45
Access reject.....	45
Access challenge.....	46
Accounting request.....	46
Accounting response.....	47
Attribute settings after reauthentication.....	47
Wi-Fi Alliance vendor-specific attribute.....	48
Access request and Accounting Request.....	48
HP vendor-specific attributes.....	48
Access request.....	48
Access accept.....	49
Group name.....	49
NAT port range.....	49
SSID.....	50
Incoming VLAN ID.....	50
Access list.....	50
Bandwidth level.....	51
Colubris-Intercept.....	51
Data rate.....	51
One-to-one NAT.....	52
Quotas.....	52
SMTP redirection.....	53
Station polling.....	54
Redirect URL.....	54
Administrator configuration attributes.....	54
Supported RADIUS attributes.....	55
Admin Access Request.....	55

Chapter 3

Customizing the public access interface 57

Overview.....	58
Common configuration tasks.....	58
Site map.....	59
Structure.....	59
Internal pages.....	60
Login page.....	60
Transport page.....	61
Session page.....	61
Fail page.....	62
External pages.....	62
Welcome page.....	63
Goodbye page.....	63
Login error page.....	63
Remote Login page.....	63
Customizing the internal pages.....	63
Creating new internal pages.....	63
Important restrictions.....	63
Loading new internal pages.....	64
Colubris-AVPair value strings.....	64
Placeholders.....	65
Example.....	65
Sample files.....	65
Changing the login page and logo.....	65
Customizing error messages.....	66
Customizing the external pages.....	66
Creating new external pages.....	66
Activating new external pages.....	66
Customization examples.....	68
Displaying custom welcome and goodbye pages.....	68
Delivering dynamically generated content.....	69
Supporting PDAs.....	69

Using a remote login page	70
How it works	70
Activating a remote login page	71
Security issues	72
Example	72
WISPr support	73
WISPr vendor-specific attributes	73
WISPr-Location-Name	73
WISPr-Location-ID	73
WISPr-Logoff-url	73
HP vendor-specific attributes	74
WISPr login URL	74
WISPr abort login URL	74
WISPr redirect page	74
Location-aware authentication	75
How it works	75
Returned information	76
Example	77
Security	77
iPass support	78
iPass login URL	78
ASP functions	78
Errors	78
RADIUS	78
Page URLs	80
Session status and properties	80
Session time	80
Session input/output/totals	82
Other	82
Session quotas	83
iPass support	84

Chapter 4

NOC authentication	87
Main benefits	88
How it works	88
Activating a remote login page with NOC authentication	89
Addressing security concerns	91
Securing the remote login page	91
Authenticating with the login application	91
Authenticating the service controller	92
NOC authentication list	92
Setting up the certificates	92
Install certificates on the web server	92
Define attributes	93
Install a certificate on service controller	93
Authenticating users	93
Returned values	95
Examples of returned HTML code	97
Simple NOC authentication example	97
Forcing user logouts	99

1

Introduction

Contents

About this guide - - - - -	6
Public/guest network access deployment - - - - -	7
Contacting support - - - - -	10
Online documentation - - - - -	10

About this guide

This guide explains how to configure and operate the public/guest network access feature.

Products covered

This guide covers the following products:

- MSM710, MSM730, MSM750
- MSM313, MSM313-R, MSM323, MSM323-R

Important terms

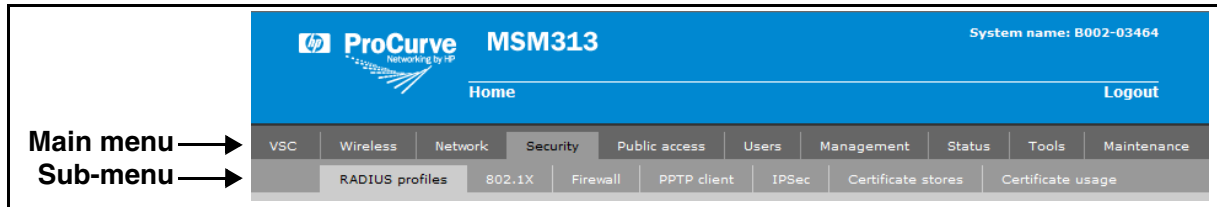
The following terms are used in this guide.

Term	Description
AP	Refers to HP MultiService Access Points: MSM335, MSM310, MSM310-R, MSM320, and MSM320-R.
service controller	Refers to the HP MSM Integrated Services Access Points, comprised of the MSM313, MSM313-R, MSM323, and MSM323-R.
Local mesh	In previous versions of the management tool and all former documentation, “local mesh” was known as “DWDS” (dynamic wireless distribution system).

Conventions

Management tool

This guide uses specific syntax when directing you to interact with the management tool user interface. Refer to this image for identification of key user-interface elements and then the table below showing example instructions:



Example directions in this guide	What to do in the user interface
Select Security > RADIUS profiles .	On the main menu select Security and then select RADIUS profiles on the sub-menu.
For Password specify secret22 .	In the field Password enter the text secret22 exactly as shown.

Commands and program listings

Monospaced text identifies commands, and program listings as follows:

Example	Description
<code>use-access-list</code>	Command name. Specify it as shown.
<code><i>ip_address</i></code>	Items in italics are parameters for which you must supply a value.
<code>ssl-certificate=URL [%s]</code>	Items enclosed in square brackets are optional. You can either include them or not. Do not include the brackets. In this example you can either include the “%s” or omit it.
<code>[ONE TWO]</code>	Items separated by a vertical line indicate a choice. Specify only one of the items. Do not include the vertical line.

Cautions

Caution: Cautions must be heeded to avoid loss of data or configuration information and to avoid improperly-configured networks.

Public/guest network access deployment

The public/guest network access feature enables the service controller to provide controlled network access for a variety of applications. Some common uses of this feature are:

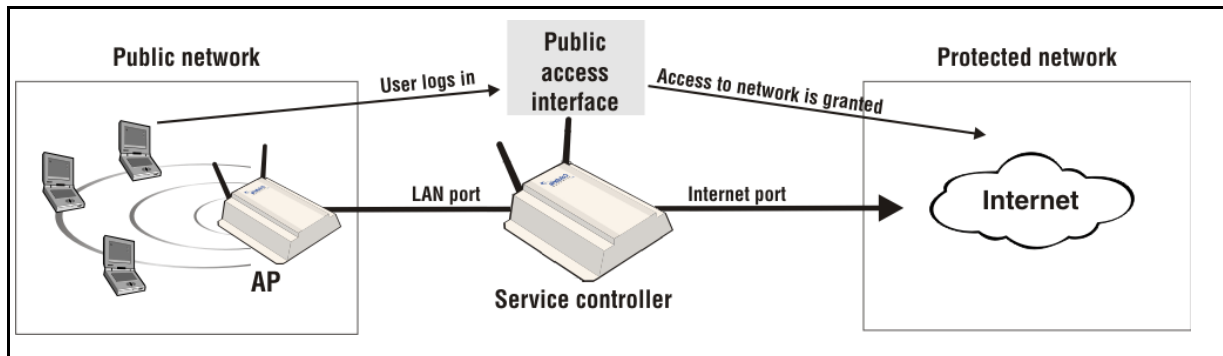
- Providing Internet access to wireless users in airports, restaurants, train stations, conference halls, etc.
- Providing wireless and wireline access to guests in hospitals, corporations, government buildings.
- Providing wireless and wireline access to students and teachers in schools and universities.
- Providing outdoor wireless access for an entire town, enabling city workers, police, fire, public security, and the general public to connect.

About the public and protected networks

When using the public/guest network access feature, the service controller acts as the gatekeeper between two distinct network segments: the public network and the protected network. In a typical setup:

- Access to the public network and its resources is available to all users once they successfully associate with the wireless network.
- Access to the private network is restricted by the service controller and requires that users be authenticated by the service controller before they gain access. In most setups, users login via the public access interface Login page via their web browser. Other authentication methods (802.1X, MAC) are available.

The following diagram shows the public and private networks for a simple deployment.



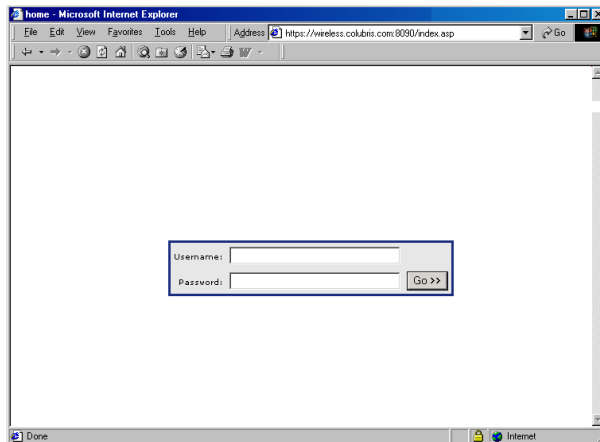
Network access

To reach private network resources, users must login using the public access interface and be authenticated by the service controller. The service controller can authenticate users using locally defined user accounts or by using the services of a third-party authentication server.

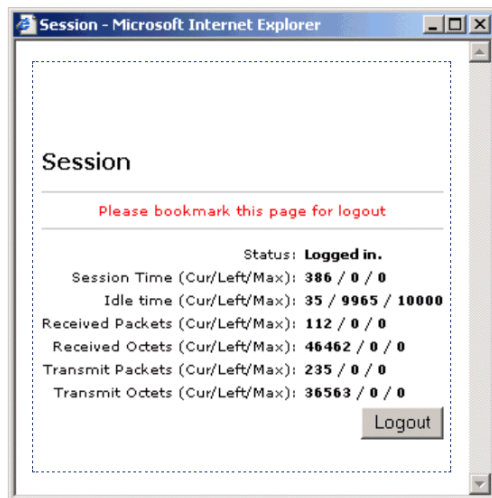
Note: The public access interface is only active on access-controlled VSCs that have HTML authentication enabled.

The public access interface is automatically displayed when a user attempts to browse a resource on the protected network. Initially the user sees the login page.

For example, the default login page:



Once the user is authenticated, the Session page appears, and the user is redirected either to the originally requested URL or to the Welcome page if defined. For example, this is default Session page. There is no default Welcome page, as it is an optional page.



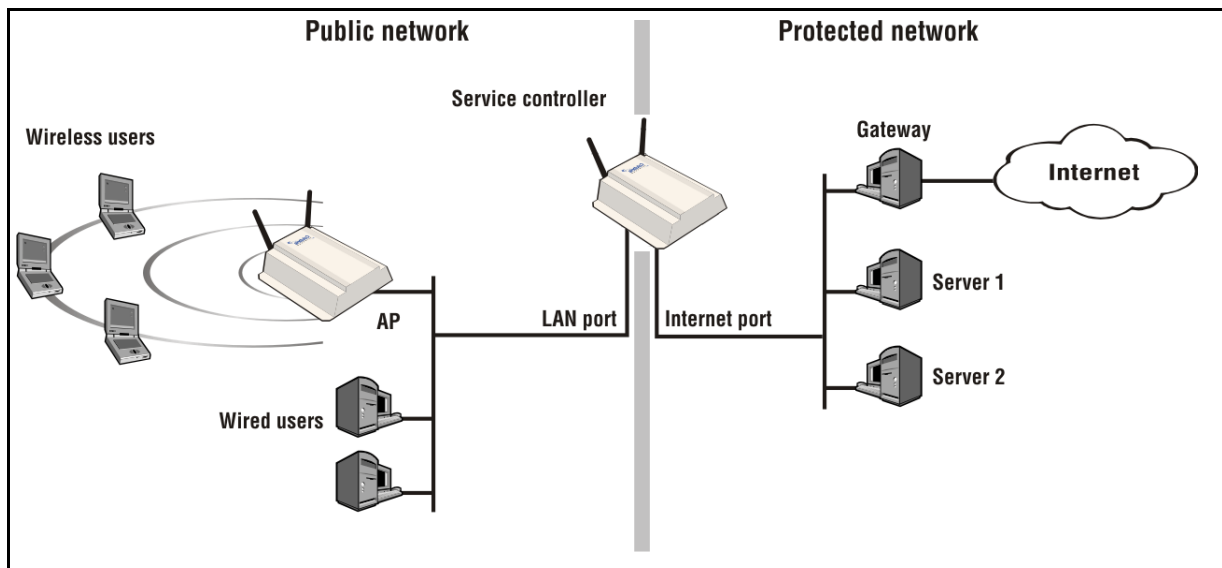
Note: A popup blocker will prevent the session page from being displayed.

Note: Users that are authenticated via MAC address are automatically logged in and do not see the Login page.

The service controller ships with default configuration settings for the public access network. Use the information in this guide to customize these settings to meet the needs of your installation. See also [“Configuring the public access network” on page 12](#).

Sample deployment

The following example shows a simple public/guest access network created using a service controller in combination with an autonomous AP.



In this example, wireless users can access the network in two different areas:

- access to the public network is available to all wireless and wired users
- access to the protected network (Server 1, Server 2, and the Internet) is only available to authenticated wireless users

For more scenarios, refer to the *MSM313/MSM323 Deployment Guide*. It contains a variety of examples that illustrate how to deploy and configure the public/guest network access feature.

Contacting support

The HP Web site, www.hp.com/networking/support provides up-to-date support information.

Additionally, your HP-authorized network reseller can provide you with assistance, both with services that they offer and with services offered by HP.

Online documentation

The latest documentation is available on the HP Support Web page at:

www.hp.com/networking/support.

2

Customizing the public access network

Contents

Configuring the public access network - - - - -	12
Attribute summary - - - - -	15
Service controller attributes - - - - -	21
User configuration attributes - - - - -	43
Administrator configuration attributes - - - - -	54

Configuring the public access network

Configuration of the public access network is controlled with RADIUS attributes. These attributes can be specified directly on the service controller or stored on a RADIUS server. Attributes are grouped into the following three categories.

- **Service controller attributes:** These attributes are used to configure global public access settings. They can be defined in the RADIUS account for the service controller or locally. Refer to [“Service controller attributes” on page 21](#) for a complete list.
- **User attributes:** These attributes are used to customize settings on a per-user basis. They can be defined in the RADIUS account for each user or locally on the service controller (MSM7xx series only). Refer to [“User configuration attributes” on page 43](#) for a complete list.
- **Administrator attributes:** These attributes are used to define settings for administrator accounts. They must be defined in the RADIUS account for each administrator. They cannot be defined locally on the service controller. Refer to [“Administrator configuration attributes” on page 54](#) for a complete list.

Activating the public access network

The public access network is only available for access-controlled VSCs.

Important: To disable access control on a MSM7xx series product, disable the **Access control** option under **General** for each VSC, AND disable the **Access control** option on the **Service Controller >> Public Access > Access control** page.

Defining and retrieving service controller attributes

RADIUS attributes are used to define a number of features for the public access interface.

Attributes can be retrieved from a third-party RADIUS server or defined directly on the service controller. For more information on these attributes, refer to [“Attribute summary” on page 15](#).

Select **Public Access > Attributes** to open the **RADIUS Attributes** page.

Any change to the local site config will only get apply at the next re-authentication.

RADIUS attributes

☐ Retrieve attributes using RADIUS

RADIUS profile: Profile 1

RADIUS username:

RADIUS password:

Confirm RADIUS password:

☐ Accounting

☒ Retrieved attributes override configured attributes

Retrieval interval: 720 minutes

Last retrieved: 7:46:20 ago

Retrieve Now

Save

Configured attributes

Attribute	Value	Action
VSA-WISPR-ACCESS-PROCEDURE	1.0	

Add New Attribute...

Configurable parameters on the **RADIUS Attributes** page include those described in the following sections.

Retrieve attributes using RADIUS

If you enable this option, the service controller will use the services of a third-party RADIUS server to retrieve configuration settings for customization of the public access interface. The settings must be defined in a RADIUS account for the service controller.

The retrieved settings will be added to those defined in the **Configured attributes** table (if any) to build the complete list of defined attributes. If the same attribute is defined on both the RADIUS server and in the Configured attributes table, the setting of **Retrieved attributes override configured attributes** determines which setting is used.

Enable the **Retrieve attributes using RADIUS** checkbox to configure the following parameters:

- **RADIUS profile:** Select a previously configured RADIUS profile to use to authenticate the service controller.
- **RADIUS username:** Specify the username of the RADIUS account assigned to the service controller.
- **RADIUS password / Confirm password:** Specify the password of the RADIUS account assigned to the service controller.
- **Accounting:** Enable this option to have the service controller generate a RADIUS accounting request ON/OFF each time its authentication state changes.

- **Retrieved attributes override configured attributes:** Enable this option to have attributes retrieved from the RADIUS server overwrite settings defined in the **Configured attributes** table.
- **Retrieval interval:** Specify the number of minutes to use for a retrieval interval. The service controller retrieves configuration settings each time this interval expires. This enables the service controller to retrieve updated operating information at regular intervals.
- **Last retrieved:** Shows the amount of time that has passed since the service controller successfully authenticated.

To avoid potential service interruptions that may occur when new operating information is activated by the service controller, it is strongly recommends that you use a large interval (12 hours or more).

You can override this value using the RADIUS attribute Session-timeout, which enables the following effective strategy: Configure Retrieval interval to a small value (10 to 20 minutes) and set the RADIUS attribute Session-timeout to override it with a large value (12 hours) when authentication is successful. Since the Retrieval interval is also respected for Access Reject packets, this configuration results in a short reauthentication interval in the case of failure, and a long one in the case of success.

- **Retrieve Now:** Select to force the service controller to contact the RADIUS server and retrieve configuration settings.

Configured attributes

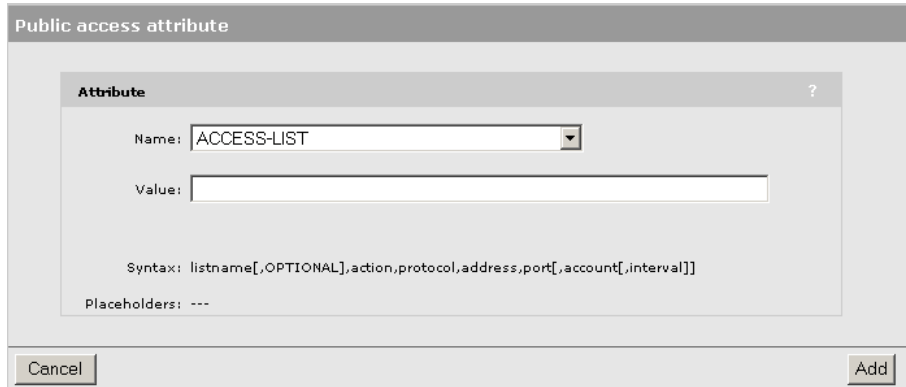
This table lists the locally configured attributes that define settings for the public access interface and user accounts.

This enables you to run the service controller without setting up a RADIUS server to store this configuration information, which is convenient for experimenting with the service controller feature set before deploying it.

If you enable the **Retrieve attributes using RADIUS** option, these same attributes can be defined in the RADIUS account for the service controller. When the service controller authenticates itself, it retrieves the attributes. These attributes will overwrite the configured attributes in this table if the **Retrieved attributes override configured attributes** is enabled.

To add a new attribute:

1. Select **Add New Attribute**. The **Public access attribute** page opens.



2. Under **Name**, select a type of local configuration attribute, as shown in the following figure.
3. Once you select a **Name**, information appears regarding the correct syntax to specify under **Value**. Use the correct syntax to specify the desired **Value**.
4. Select **Add**.

Attribute summary

This section provides a summary of all supported attributes.

Standard RADIUS attributes

The service controller supports the following standard RADIUS attributes. The names in boldface type (**Access Request**, **Access Accept**, and so on) refer to standard RADIUS message types.

Note: Attributes that begin with the letters MS are defined by Microsoft and are not RADIUS standard attributes.

Access Request

- Acct-Session-Id
- NAS-Port
- NAS-Port-Type
- User-Name
- Calling-Station-Id
- Called-Station-Id
- Framed-Ip-Address
- User-Password
- CHAP-Password
- CHAP-Challenge
- MSCHAP-Challenge
- MSCHAP-Response
- MSCHAPv2-Response
- EAP-Message
- State
- NAS-Identifier
- NAS-Ip-Address
- Framed-MTU
- Connect-Info
- Service-Type
- Message-Authenticator
- Chargeable-User-Identity
(only if enabled for the VSC)

Access Accept

- MS-MPPE-Recv-Key
- MS-MPPE-Send-Key
- EAP-Message
- Class
- Idle-Timeout
- Session-Timeout
- Acct-Interim-Interval
- Reply-Message
- Tunnel-Type
- Tunnel-Medium-Type
- Tunnel-Private-Group-ID
- Termination-Action
- Chargeable-User-Identity

Access Reject

- MSCHAP-Error
- Reply-Message
- EAP-Message

Access Challenge

- EAP-Message
- State

Accounting Request

- User-Name
- NAS-Port
- NAS-Port-Type
- NAS-Identifier
- NAS-Ip-Address
- Acct-Status-Type
- Calling-Station-Id
- Called-Station-Id
- Acct-Event-Timestamp
- Acct-Delay-Time
- Acct-Session-Id
- Acct-Authentic
- Acct-Session-Time
- Acct-Input-Octets
- Acct-Input-Gigawords
- Acct-Input-Packets
- Acct-Output-Octets
- Acct-Output-Gigawords
- Acct-Output-Packets
- Acct-Terminate-Cause
- Class
- Framed-Ip-Address
- Chargeable-User-Identity

Accounting Response

No attributes supported.

Vendor-specific attributes

The RADIUS standard allows vendors to create their own *vendor-specific* attributes to support additional features. HP supports vendor-specific attributes defined by the Wi-Fi Alliance, called WISPr (Wireless Internet Service Project Roaming) attributes, and attributes that HP has defined on its own.

WISPr vendor-specific attributes

HP supports three Wi-Fi Alliance vendor-specific attributes for Access Request and Accounting Request. These attributes are:

`wispr-location-name=location_name`

`wispr-location-id=location_id`

`wispr-logout-url=URL`

WISPr-Location-Name

- SMI network management private enterprise code = 14122
- Vendor-specific attribute type number = 2
- Attribute type = string

WISPr-Location-ID

- SMI network management private enterprise code = 14122
- Vendor-specific attribute type number = 1
- Attribute type = string

WISPr-Logout-url

- SMI network management private enterprise code = 14122
- Vendor-specific attribute type number = 3
- Attribute type = string

HP vendor-specific attributes

HP has defined two vendor-specific attributes, **Colubris-AVPair** and **Colubris-Intercept**, to support special features on the service controller such as the customization of the web interface and the security certificate. These attributes conform to RADIUS RFC 2865.

You may need to define these attributes on your RADIUS server if they are not already present. In this case you need to specify the following:

Colubris-AVPair

- SMI network management private enterprise code = 8744
- Vendor-specific attribute type number = 0
- Attribute type = string

Colubris-Intercept

- SMI network management private enterprise code = 8744
- Vendor-specific attribute type number = 1
- Attribute type = integer

HP vendor-specific attribute summary

The following values are permitted for the Colubris-AVPair attribute.

Service controller attributes	Service controller attributes
<ul style="list-style-type: none"> • access-list • use-access-list • ssl-certificate • ssl-noc-certificate • ssl-noc-ca-certificate • configuration-file • mac-address • default-user-idle-timeout • default-user-session-timeout • default-user-smtp-redirect • default-user-acct-interim-update • default-user-one-to-one-nat • default-user-max-input-packets • default-user-max-output-packets • default-user-max-input-octets • default-user-max-output-octets • default-user-max-total-octets • default-user-max-total-packets • primary-web-server-status-url • secondary-web-server-status-url 	<ul style="list-style-type: none"> • dnat-server • primary-dnat-server-status-url • secondary-dnat-server-status-url • login-page • transport-page • session-page • fail-page • logo • redirect-page • goodbye-url • welcome-url • messages • login-err-url • login-url • ipass-login-url • wispr-login-url • wispr-abort-login-url • colubris-wispr-access-procedure

User attributes	User attributes
<ul style="list-style-type: none">• group• nat-port-range• ssid• incoming-vlan-id• use-access-list• bandwidth-level• max-input-rate• max-output-rate• one-to-one-nat	<ul style="list-style-type: none">• max-input-packets• max-output-packets• max-input-octets• max-output-octets• max-total-packets• max-total-octets• smtp-redirect• polling-arp-interval• polling-max-arp-count

Maximum attribute size

The maximum attribute size that the service controller can receive in a single RADIUS request is 4096 bytes.

Note: Some networks may limit RADIUS request size to around 1500 bytes because they discard UDP fragments.

Acct-Terminate-Cause values

RADIUS standard Acct-Terminate-Cause values are supported as follows:

ID	Cause	Notes
1	User Request	Supported. Indicates that the user logged out.
2	Lost Carrier	Supported. Indicates that the client station is no longer alive.
3	Lost Service	Supported. When location-aware is enabled and a user switches access points, the service controller re-authenticates the user. If authentication fails due to timeout, this code is returned.
4	Idle Timeout	Supported. User exceeded the idle timeout value defined for the session.
5	Session Timeout	Supported. User exceeded maximum time defined for the session.
6	Admin Reset	Supported. User session was terminated by the service controller administrator via SNMP or the management tool.
7	Admin Reboot	Not Supported. (not applicable)
8	Port Error	Supported. If two users are detected using the same IP address, both are logged out with this error. Another cause is if an error is encountered in an access list definition. For example, an invalid host was specified.
9	NAS Error	Not Supported. (not applicable)
10	NAS Request	Not Supported. (not applicable)
11	NAS Reboot	Supported. User was logged out because the service controller was restarted.
12	Port Unneeded	Not Supported. (not applicable)
13	Port Preempted	Supported. When a user switches AP or SSID with incompatible configurations (authentication type), they are logged out with this code. Also if the user changes authentication type of the same AP.
14	Port Suspended	Not Supported. (not applicable)
15	Service Unavailable	Not Supported. (not applicable)
16	Callback	Not Supported. (not applicable)
17	User Error	Supported. An 801.1x client initiated a second authentication request for a user, and this request was refused.
18	Host Request	Not Supported. (not applicable)
0x8744 (34628 decimal)	Termination	Termination cause.

Service controller attributes

This section provides complete descriptions for all supported service controller attributes.

The attributes described in this section can be defined in the RADIUS account for the service controller (if you use a RADIUS server), or some can be locally configured. See [“Configuring the public access network” on page 12](#) for configuration instructions.

Standard RADIUS attributes

The following standard RADIUS attributes are supported.

Note: Strings are defined as 1 to 253 characters long.

Access request

- **Acct-Session-Id** (32-bit unsigned integer): Random value generated per authentication by the service controller.
- **NAS-Identifier** (string): The NAS ID set on the **Security > RADIUS > Add New Profile** page for the RADIUS profile being used.
- **NAS-Ip-Address** (32-bit unsigned integer): The IP address of the port the service controller is using to communicate with the RADIUS server.
- **NAS-Port** (32-bit unsigned integer): Always 0.
- **NAS-Port-Type** (32-bit unsigned integer): Always set to 19, which represents WIRELESS_802_11.
- **Calling-Station-Id** (string): The MAC address of the service controller's LAN port in IEEE format. For example: 00-02-03-5E-32-1A.
- **Called-Station-Id** (string): By default, this is set to the MAC address of the service controller's wireless/LAN port in IEEE format. For example: 00-02-03-5E-32-1A. To use the MAC address of the Internet port, you must edit the config file and change the setting of **radius-called-station-id-port** to **WAN** in the <ACCESS-CONTROLLER> section.
- **Framed-IP-Address** (32-bit unsigned integer): IP Address of the service controller's LAN port.
- **User-Name** (string): The RADIUS username assigned to the service controller on the **Public access > Attributes** page.
- **State** (string): As defined in RFC 2865.
- **Framed-MTU** (32-bit unsigned integer): Hard-coded to 1496 (802.1X).
- **Connect-Info** (string): The string “HTTPS” or “IEEE802.1X”.
- **Service-Type** (32-bit unsigned integer): RADIUS service type.
- **Message-Authenticator** (string): As defined in RFC 2869. Always present even when not doing an EAP authentication. length = 16 bytes.
- **Colubris-AVPair**: See the description in the section that follows.

The following attributes are mutually exclusive depending on the RADIUS authentication method.

- **User-Password** (string): The password assigned to the service controller on the **Public access > Attributes** page. Encoded as defined in RFC 2865. Only present when the authentication method for the RADIUS profile is set to PAP.
- **CHAP-Password** (string): The password assigned to the service controller on the **Public access > Attributes** page. Encoded as defined in RFC 2865. Only present when the authentication method for the RADIUS profile is set to CHAP.
- **CHAP-Challenge** (string): Randomly generated by the product. As defined in RFC 2865. Only present when the authentication method for the RADIUS profile is set to CHAP. Length = 19 bytes.
- **MSCHAP-Challenge** (string): As defined in RFC 2433. Only present when the authentication method for the RADIUS profile is set to MSCHAPv1 or MSCHAPv2. Length = 8 bytes.
- **MSCHAP-Response** (string): As defined in RFC 2433. Only present when the authentication method for the RADIUS profile is set to MSCHAPv1. Length = 49 bytes.
- **MSCHAPv2-Response** (string): As defined in RFC 2759. Only present when the authentication method for the RADIUS profile is set to MSCHAPv2. Length = 49 bytes.
- **EAP-Message** (string): As defined in RFC 2869. Only present when the authentication method for the RADIUS profile is set to EAP-MD5.

Access accept

- **Session-Timeout** (32-bit unsigned integer): Maximum time a session can be active. The service controller re-authenticates itself when this timer expires. Omitting this attribute or specifying 0 disables the feature. (Note that the authentication interval is also configurable on the **Public access > Attributes** page.
- **Class** (string): As defined in RFC 2865. Multiple instances are supported.
- **EAP-Message** (string): Only supported when authentication is EAP-MD5. Note that the content will not be read as the RADIUS Access Accept is overriding whatever indication contained inside this packet.
- **Colubris-AVPair**: See [“HP vendor-specific attributes” on page 24](#).

Access reject

None.

Access challenge

None.

Accounting request

- **Acct-Session-Id** (32-bit unsigned integer): Random value generated by the service controller.
- **NAS-Identifier** (string): The NAS ID set on the **Security > RADIUS > Add New Profile** page for the profile being used.
- **NAS-Ip-Address** (32-bit unsigned integer): The IP address of the port the service controller is using to communicate with the RADIUS server.
- **NAS-Port** (32-bit unsigned integer): Always 0.

- **NAS-Port-Type** (32-bit unsigned integer): Always set to 19, which represents WIRELESS_802_11.
- **Calling-Station-Id** (string): The MAC address of the service controller's LAN port in IEEE format. For example: 00-02-03-5E-32-1A.
- **Called-Station-Id** (string): The MAC address of the service controller's LAN port in IEEE format. For example: 00-02-03-5E-32-1A.
- **User-Name** (string): The RADIUS username assigned to the service controller on the **Public access > Attributes** page.
- **Class** (string). As defined in RFC 2865. Multiple instances are supported.
- **Framed-IP-Address** (32-bit unsigned integer): IP Address of the service controller's LAN port.
- **Acct-Status-Type** (32-bit unsigned integer): Supported values are Accounting-On (7) and Accounting-Off (8).
- **Acct-Event-Timestamp** (32-bit unsigned integer): As defined in RFC 2869.
- **Acct-Delay-Time** (32-bit unsigned integer): As defined in RFC 2869.
- **Acct-Authentic** (32-bit unsigned integer): Always set to 1 which means RADIUS.

Accounting response

None.

HP vendor-specific attributes

The following list summarizes all supported attributes by feature.

Feature	Description	See
Custom HTML pages and URLs, and supporting files	Enables you to customize the public access interface.	Chapter 3
Access lists	Enables you to create one or more access groups which define the set of network resources that are available to authenticated and unauthenticated users.	Page 25
Redirect URL	Enables you to redirect users with an access list definition.	Page 31
Custom security certificate	Enables you to replace the HP SSL certificate with your own.	Page 33
Configuration file	Enables you to store a configuration file at a central location to automatically update all your service controllers.	Page 34
MAC authentication	Enables you to authenticate devices based on their MAC addresses.	Page 35
Default user idle timeout	Default idle timeout for all users.	Page 36
Default user session timeout	Default session timeout for all users.	Page 36
Default user SMTP server	Default SMTP server to use for email redirection for all users.	Page 37
Default user interim accounting update interval	Default interval for user interim accounting updates.	Page 37
Default user one-to-one NAT	Defines the default setting for the one-to-one NAT option. This feature only applies to traffic using PPTP on the Internet port.	Page 38
Default user quotas	Enables default upload and download limits to be set for all users.	Page 38
Multiple login servers	This feature lets you dynamically set the URL used for retrieving custom external pages or a remote login page based on the status of a primary or secondary web server.	Page 39
WISPr support	Provides support for WISPr compliant client stations.	Chapter 3

The value of a Colubris-AVPair attribute is always a string. These strings are always of the form:
`<item>=<value>`

Access lists

Access lists enable you to create public areas on your network that all users can browse, and protected areas that are restricted to specific user accounts or groups.

Each access list is a set of rules that governs how the service controller controls access to network resources. You can create multiple access lists, each with multiple rules to manage the traffic on your public access network.

Default setting

By default no access lists are defined. This means that:

- If authentication (802.1X, WPA, HTML, MAC) is not enabled on a VSC, all users that connect to the VSC have access to the protected network.
- If authentication (802.1X, WPA, HTML, MAC) is enabled on a VSC, then:
 - Unauthenticated users only reach the public access login page. Access to the protected network is blocked, except for **register.coluris.com** which enables product registration.
 - Authenticated users have access to the protected network.

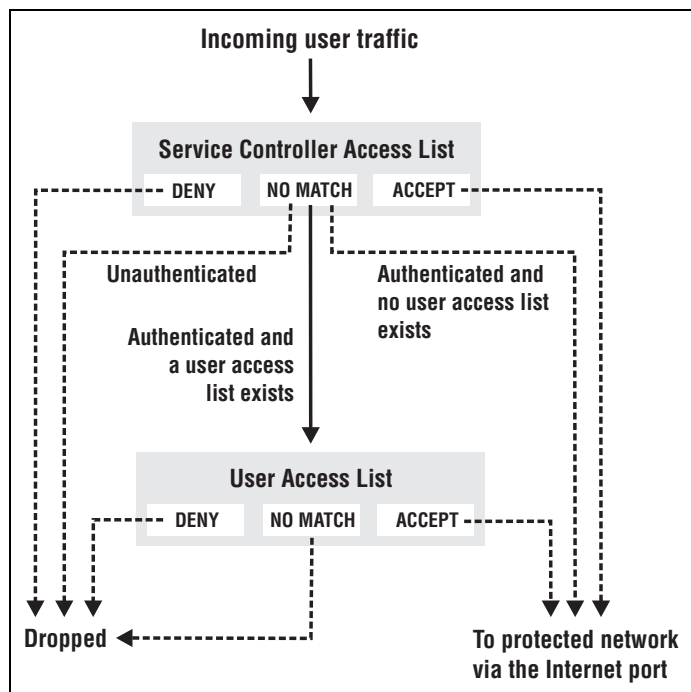
Note: If you only enable MAC-based authentication on a VSC, wired users are not required to authenticate and gain access to the protected network. In this case it is important to restrict network access with the appropriate access list definition.

How access lists work

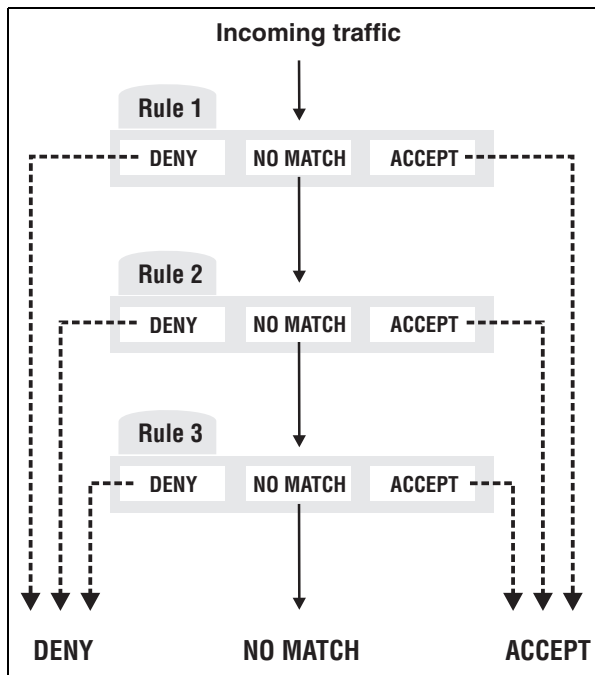
Access lists can be defined on both the service controller and individually for each user.

Incoming traffic cascades through the currently active lists. Traffic that is accepted or denied by a list is not available to the list that follows it. Traffic that passes through all lists without being accepted or denied is dropped.

The following diagram illustrates how incoming traffic from a user session is processed by the access list mechanism.



Within each access list, traffic cascades through the list rules in a similar manner.



Rules are numbered according to the order in which they are added. Only data that is not accepted or denied by a rule is available to the next rule in the list.

Accounting support

Each rule in an access list can be configured with an account name for billing purposes. The service controller sends billing information based on the amount of traffic matched by the rule.

This lets you create rules to track and bill traffic to particular destinations.

Tips on using the access list

With certificates

- If you replaced the default SSL certificate on the service controller with one signed by a well-known CA, you should define the access list to permit access to the CA certificate for all non-authenticated users. This enables the user's browser to verify that the certificate is valid without displaying any warning messages.
- Users may have configured their web browsers to check all SSL certificates against the Certificate Revocation List (CRL) maintained by the CA that issued the certificate. The location of the CRL may be configured in the browser, or embedded in the certificate. The access list should be configured to permit access to the CRL, otherwise the user's browser times out before displaying the login page.

Remote login page

If you are using the remote login page feature, make sure that access to the web server hosting the page must be granted to all unauthenticated users.

SMTP redirect

If an unauthenticated user establishes a connection to their email server, the SMTP redirect feature will not work once the user logs in. The user's email is still sent to the original email server.

To avoid this, do not use an access list to open TCP port 25 for unauthenticated users.

Critical access list definitions (such as for a remote login page, certificates) should not use the **OPTIONAL** setting because if these definitions fail to initialize there is no indication in the log.

Defining access lists

Access lists are defined by adding the following Colubris-AVPair value string to the RADIUS profile for a service controller or to the local list (**Public access > Attributes** page).

Each value string defines one rule. Up to 99 rules can be defined for an access list.

```
access-list=value
```

All rules that make up an access list must be initialized without error for the list to be active. (You can force the service controller to ignore initialization errors on a rule-by-rule basis by using the **OPTIONAL** parameter.)

You can define up to 32 access lists.

Activating access lists

To activate Access lists, add the following Colubris-AVPair value string to the RADIUS profile for a service controller or a user.

```
use-access-list=value
```

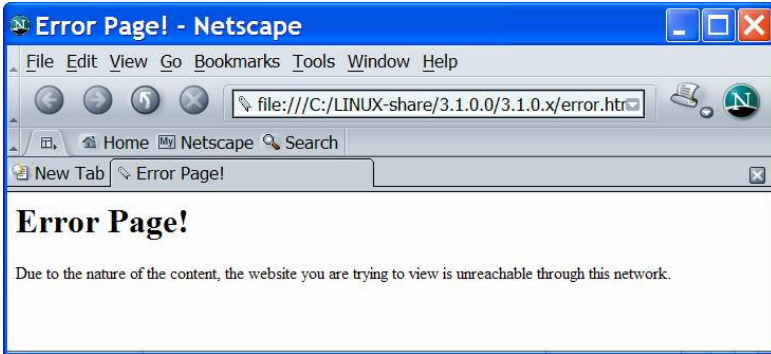
Only one access list can be active per profile. This list must have been initialized without an error.

Use the following Colubris-AVPair value string:

```
access-list=
listname[,OPTIONAL],action,protocol,address,port[,account[,interval]]
```

```
use-access-list=uselistname
```

Parameter	Description
listname	Specify a name (up to 32 characters long) to identify the access list this rule applies to. If a list with this name does not exist, a new list is created. If a list with this name exists, the rule is added to it.
uselistname	Specify the name of an existing access list. This list is activated for the current profile. Lists are checked in the order they are activated.
OPTIONAL	Allows the access list to be activated even if this rule fails to initialize. For example, if you specify a rule that contains an <i>address</i> which cannot be resolved for some reason, the other rules that make up the access list are still initialized. If you do not specify optional, a failed rule causes the entire list to fail. Important: Critical access list definitions (such as for a remote login page, certificates) should not use the OPTIONAL setting because if these definitions fail to initialize there is no indication in the log.

Parameter	Description
action	<p>Specify what action the rule takes when it matches incoming traffic. Two options are available:</p> <ul style="list-style-type: none"> • ACCEPT: Allow traffic matching this rule. • DENY: Reject traffic matching this rule. • WARN: Reject traffic matching this rule and return an HTTP error message (which is not customizable) indicating that access to the site is not allowed by the network. For example:  <ul style="list-style-type: none"> • REDIRECT: Reject traffic matching this rule and redirect the user's web browser to the page specified by redirect-url, or login-url if redirect-url is not defined. For example, one use for this feature could be to block access to a popular protocol, then prompt the user for additional fees to activate support. • DNAT-SERVER: Traffic matching this rule is forwarded to the destination defined by the attribute dnat-server. See “Traffic forwarding (dnat-server)” on page 40 for more information.
protocol	Specify the protocol to check: tcp , udp , icmp , all
address	<p>Specify one of the following:</p> <ul style="list-style-type: none"> • IP address or domain name (up to 107 characters in length). • Subnet address. Include the network mask as follows: address/subnet mask For example: 192.168.30.0/24 • Use the keyword all to match any address. • Use the keyword none if the protocol does not take an address range (ICMP for example).
port	<p>Specify a specific port to check or a port range as follows:</p> <ul style="list-style-type: none"> • none: Used with ICMP (since it has no ports). • all: Check all ports. • 1-65535[:1-65535]: Specify a specific port or port range.

Parameter	Description
account	Specify the name of the user account that the service controller sends billing information to for this rule. Account names must be unique and can be up to 32 characters in length.
interval	Specify time between interim accounting updates. If you do not enable this option, accounting information is only sent when a user connection is terminated. Range: 5-99999 seconds in 15 second increments.

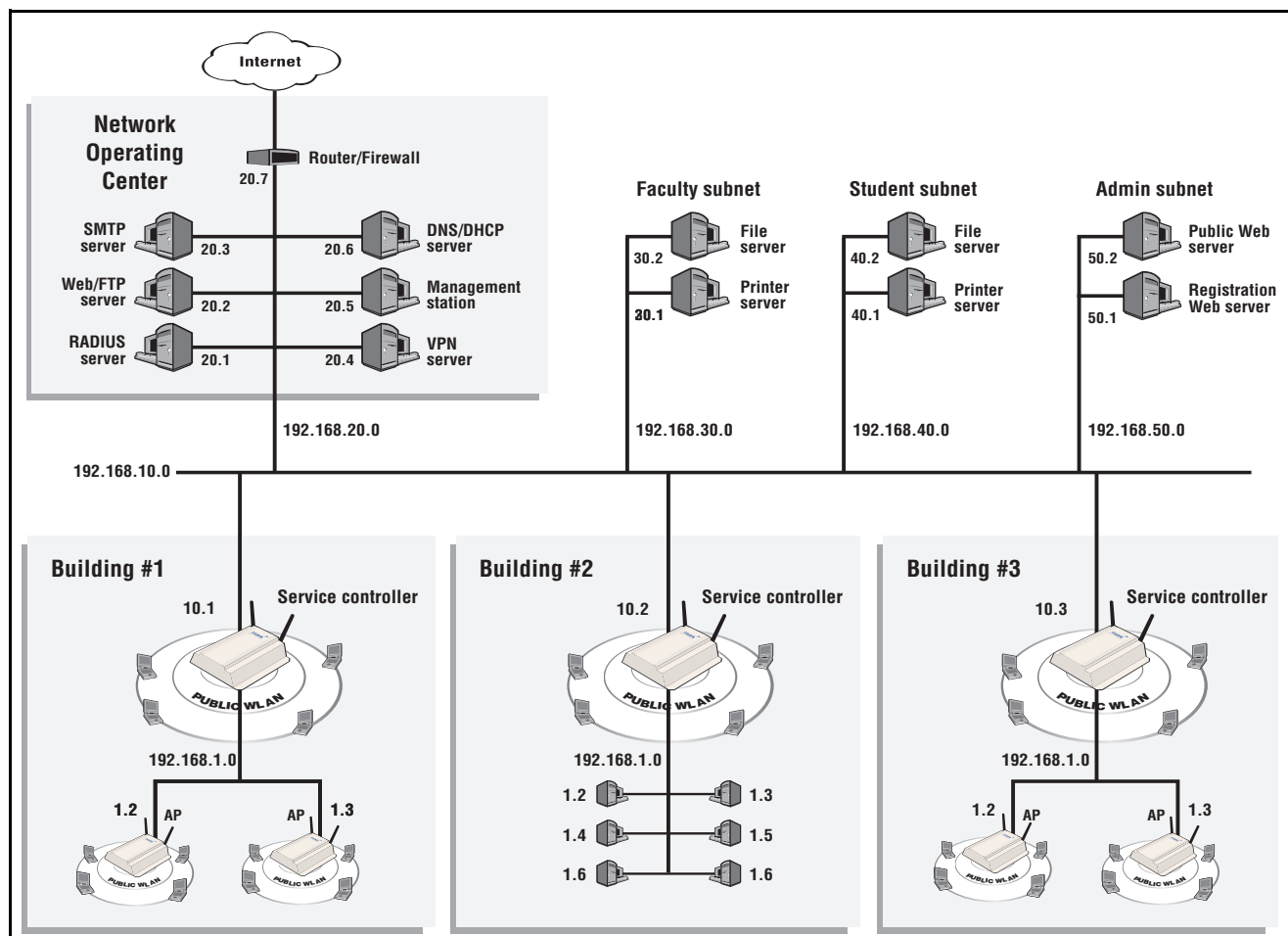
Note: You can use spaces as separators instead of commas.

Access list example

This example illustrates how access lists can be used to control access to network resources for different groups of users at a fictitious university campus.

Topology

The following topology show potential wireless deployments for the campus using HP equipment. A RADIUS server is used to store configuration attributes for the public access network.



Access list definitions

The RADIUS profile for the service controller contains the following:

```
access-list=everyone, ACCEPT, tcp, 192.168.50.2, 80
```

```
access-list=students,ACCEPT,tcp,192.168.50.1,80,students_reg,500
access-list=students,ACCEPT,all,192.168.40.0/24,all
access-list=students,DENY,all,192.168.20.0/24,all
access-list=students,DENY,all,192.168.30.0/24,all
access-list=students,ACCEPT,all,all.all,student_internet_use,5000

access-list=faculty,ACCEPT,tcp,192.168.50.1,80,faculty_reg,500
access-list=faculty,ACCEPT,all,192.168.30.0/24,all
access-list=faculty,DENY,all,192.168.20.0/24,all
access-list=faculty,DENY,all,192.168.40.0/24,all
access-list=faculty,ACCEPT,all,all.all,faculty_internet_use,5000

use-access-list=everyone
```

The RADIUS profile for every student contains the following:

```
use-access-list=students
```

The RADIUS profile for every faculty member contains the following:

```
use-access-list=faculty
```

This definitions create three access lists: everyone, students, and faculty.

Everyone

This list applies to all users (students, teachers, guests), whether they are authenticated or not. This is because the list is active on the service controller, which is accomplished with the entry:

```
use-access-list=everyone
```

It enables everyone to access the public web server.

Students

This list applies to authenticated students only. It is composed of the following entries:

```
access-list=students,ACCEPT,tcp,192.168.50.1,80,students_reg,500
```

Enables web traffic to the registration web server. Accounting data is recorded in the account students_reg.

```
access-list=students,ACCEPT,all,192.168.40.0/24,all
```

Enables traffic to reach the student segment.

```
access-list=students,DENY,all,192.168.20.0/24,all
access-list=students,DENY,all,192.168.30.0/24,all
```

These two entries deny access to the faculty subnet and the NOC.

```
access-list=students,ACCEPT,all,all.all,student_internet_use,5000
```

Enables all other traffic to reach the Internet (via routers on the backbone LAN and the router in the NOC). If this last rule did not exist, this traffic would be dropped.

Faculty

This list applies to authenticated faculty members only. It is composed of the following entries:

```
access-list=faculty,ACCEPT,tcp,192.168.50.1,80,faculty_reg,500
```

Enables web traffic to the registration web server. Accounting data is recorded in the account `faculty_reg`.

```
access-list=faculty,ACCEPT,all,192.168.30.0/24,all
```

Enables traffic to reach the faculty segment.

```
access-list=faculty,DENY,all,192.168.20.0/24,all  
access-list=faculty,DENY,all,192.168.40.0/24,all
```

These two entries deny access to the student subnet and the NOC.

```
access-list=faculty,ACCEPT,all,all.all,faculty_internet_use,5000
```

Enables all other traffic to reach the Internet (via routers on the backbone LAN and the router in the NOC). If this last rule did not exist, this traffic would be dropped.

Redirect URL

The `redirect-url` attribute is used to specify the target URL for redirection in service controller and user profiles when using an access list with the REDIRECT action. Only one `redirect-url` attribute can be specified in each service controller and user RADIUS profile.

When a rule with the REDIRECT action is processed, the user's browser is redirected to a different HTTP address in this order:

- `redirect-url` attribute in the user profile, if present
- `redirect-url` attribute in the service controller profile, if present
- `login-url` attribute, only available in the service controller profile

Use the following Colubris-AVPair value string:

```
redirect-url=URL_of_the_page [placeholder]
```

Where:

Parameter	Colubris-AVPair value string
URL_of_the_page	URL of the redirect page. Access to the web server hosting this page must be granted to all unauthenticated users. Do this with an appropriate access list definition.

The following placeholders can be added to the login-url string.

Placeholder	Description
%c	Returns the IP address of the user's computer.
%d	Returns the WISPr location-ID. Supported for login-url only.
%e	Returns the WISPr location-Name. Supported for login-url only.
%l	Returns the URL on the service controller where user login information should be posted for authentication. By default, this value is URL encoded. (To enable/disable URL encoding, set the value of url-encode in the <ACCESS-CONTROLLER> section in the configuration file.)
%n	Returns the NAS ID assigned to the service controller. By default, this is the unit's serial number. Not supported in local mode.
%s	Returns the RADIUS login name assigned to the service controller. By default, this is the unit's serial number. Not supported in local mode.
%o	Returns the original URL requested by the user. By default, this value is URL encoded. (To enable/disable URL encoding, set the value of url-encode in the <ACCESS-CONTROLLER> section in the configuration file.)
%i	Returns the domain name assigned to the service controller's Internet port.
%p	Returns the port number on the service controller where user login information should be posted to for authentication.
%a	Returns the IP address of the service controller's interface that is sending the authentication request.
%E	When the location-aware feature is enabled, returns the ESSID of the wireless access point the user is associated with.
%P	When the location-aware feature is enabled, returns the wireless mode ("ieee802.11a", "ieee802.11b", "ieee802.11g") the user is using to communicate with the access point.
%G	When the location-aware feature is enabled, returns the group name of the wireless access point the user is associated with.
%C	When the location-aware feature is enabled, returns the Called-station-id content for the wireless access point the user is associated with.
%r	Returns the string sent by the RADIUS server when an authentication request fails. The RADIUS server must be configured to support this feature. The information contained in the returned string depends on the configuration of the RADIUS server.
%m	Returns the MAC address of the wireless/wired client station that is being authenticated.
%v	Returns the VLAN assigned to the client station at the service controller's ingress (LAN port).

Note: The maximum length of the remote login page URL is 512 characters. If this is exceeded (when using placeholders for example), the URL is truncated. HP therefore recommends that you specify the most-important placeholders first.

Example

One way to use this feature is to offer a premium service for a given (or all) sites. For example, in the service controller profile, define two lists, one for normal usage and one for premium usage:

```
access-list=normal,REDIRECT,tcp,www.mypremiumservice.com,80
access-list=normal,ACCEPT,all,all,all
access-list=premium,ACCEPT,all,all,all
redirect-url=http://www.mysite.com/getpremium/
```

In the RADIUS profile for normal users, map them to the “normal” access list:

```
use-access-list=normal
```

In the RADIUS profile for premium users, map them to the “premium” access list:

```
use-access-list=premium
```

The access list only takes effect on an authentication, so a change of service as shown in this example takes effect only at the user’s next authentication (login).

Custom SSL certificate

The service controller can retrieve a custom SSL security certificate to replace the HP certificate that is included by default.

Use the following Colubris-AVPair value string:

```
ssl-certificate=URL[placeholder]
```

Where:

Parameter	Description
URL	Specify the URL that points to the new certificate.

By using the following placeholders, you can customize the **URL** parameter.

Placeholder	Description
%n	Returns the NAS ID assigned to the service controller. By default, this is the unit’s serial number.
%s	Returns the RADIUS login name assigned to the service controller on the Public access > Attributes page. By default, this is the unit’s serial number.
%i	Returns the domain name assigned to the service controller’s Internet port.
%a	Returns the IP address of the service controller’s interface that is sending the authentication request.

The certificate is encoded using PKCS#12 format, and contains the following:

- the private key of the web server
- the certificate of the web server

The file is locked using a password.

Note: The password that the certificate was locked with must be the same as the password specified on the **Public access > Attributes** page. This is the password that the service controller uses to log in to the RADIUS server.

Example

```
ssl-certificate=http://www.hp.com/%s_certificate
```

Configuration file

The service controller can retrieve and load a new configuration file automatically, based on a URL you specify.

Use the following Colubris-AVPair value string:

```
configuration-file=URL[placeholder]
```

Where:

Parameter	Description
URL	Specify the URL that points to the new configuration file.

By using the following placeholders, you can customize the **URL** parameter. This is useful when you need to update multiple service controllers.

Placeholder	Description
%n	Returns the NAS ID assigned to the service controller. By default, this is the unit's serial number.
%s	Returns the RADIUS login name assigned to the service controller on the Public access > Attributes page. By default, this is the unit's serial number.
%i	Returns the domain name assigned to the service controller's Internet port.
%a	Returns the IP address of the service controller's interface that is sending the authentication request.

Example

```
configuration-file=http://www.hp.com/%s_configfile
```

MAC authentication

The service controller can authenticate a device based on its MAC address. This authentication method is useful for authenticating devices that do not have a web browser—for example, cash registers. The method can also be used to authenticate an MSM AP operating in autonomous mode.

To use this feature you must define a local user account or a RADIUS user account for each device as follows:

- **username:** Set this to the username you specified in the mac-address value string. If no username is specified, set the account name to the MAC address of the device. Use dashes to separate characters in the address. For example: 00-20-E0-6B-4B-44.
- **password:** Set this to the password you specified in the mac-address value string. If no password is specified, set this to the same password that is used for the user account you defined for the service controller on the **Security > Authentication** page.

Caution: The username and password are not encrypted for transmission; therefore, it is important that the link with the RADIUS server is secure.

Note: MAC authentication only applies to VSCs that have HTML-based authentication enabled.

Use the following Colubris-AVPair value string:

```
mac-address=address[,username[,password]]
```

Where:

Parameter	Description
address	Specify the MAC address of the device to authenticate. Use dashes to separate characters in the address. Do not use colons (:). For example: 00-20-E0-6B-4B-44.
username	Specify the username to associate with this MAC address. Maximum 32 alphanumeric characters. The username field cannot contain a comma.
password	Specify the password to associate with this MAC address. Maximum 32 alphanumeric characters. The password field cannot contain a comma.

Example

Consider the scenario where several APs operating in autonomous mode are installed with a service controller. For the APs to perform firmware upgrades from a remote web or FTP server, they must log in to the public access network. By using MAC-based authentication, this can easily be accomplished.

Default user idle timeout

Use this to set the default idle timeout for all users whose RADIUS profile does not contain a value for the RADIUS attribute *idle-timeout*.

Use the following Colubris-AVPair value string:

```
default-user-idle-timeout=seconds
```

Where:

Parameter	Description
seconds	Specify the maximum amount of time a user session can be idle. Once this time expires, the session is automatically terminated. A value of 0 means no timeout.

Default user session timeout

Use this to set the default session timeout for all users whose RADIUS profile does not contain a value for the RADIUS attribute *session-timeout*. This value also applies to users authenticated locally via the **Public access > Users** page.

Use the following Colubris-AVPair value string:

```
default-user-session-timeout=seconds
```

Where:

Parameter	Description
seconds	Specify the maximum amount of time a user session can be connected. Once this time expires, the session is automatically terminated. A value of 0 means no timeout.

Default user SMTP server

Use this to set the default SMTP server address for all user sessions. This attribute is used if a specific server is not set for a particular user (See [“SMTP redirection” on page 53](#)).

Use the following Colubris-AVPair value string:

```
default-user-smtp-redirect=hostname[:port] [,username,password]
```

Where:

Parameter	Description
hostname	Specify the IP address or domain name of the e-mail server. Maximum length is 253 characters.
port	Specify the port on the e-mail server to relay to. Range: 1 to 65535. Default: 25
username	Specify the username required to log on to the SMTP server. Maximum 32 characters. Only used if the Support authentication on SMTP proxy server option is enabled on the Public access > Access control page. Works with SMTP servers that support plain or CRAM-MD5 authentication.
password	Specify the password required to log on to the SMTP server. Maximum 32 characters. Only used if the Support authentication on SMTP proxy server option is enabled on the Public access > Access control page. Works with SMTP servers that support PLAIN or CRAM-MD5 authentication.

Default user interim accounting update interval

This attribute lets you define the interim accounting update interval for all users that do not have a specific interval set in their RADIUS profile.

Use the following Colubris-AVPair value string:

```
default-user-acct-interim-update=value
```

Where:

Parameter	Description
value	Number of seconds between interim updates.

Default user one-to-one NAT

Note: This feature only applies to traffic using PPTP on the Internet port.

This attribute lets you define the default setting for one-to-one NAT support for all users that do not have a this setting specified in their RADIUS profile. For more information see [“One-to-one NAT” on page 52](#).

Use the following Colubris-AVPair value string:

```
default-user-one-to-one-nat=value
```

Where:

Parameter	Description
value	Set this to 1 to activate one-to-one NAT support.

Default user quotas

These attributes let you define upload and download limits for all users that do not have a specific limit set in their RADIUS profile. Limits can be defined in terms of packets or octets (bytes).

Use the following Colubris-AVPair value string:

```
default-user-max-input-packets=value  
default-user-max-output-packets=value  
default-user-max-input-octets=value  
default-user-max-output-octets=value  
default-user-max-total-octets=value  
default-user-max-total-packets=value
```

Where:

Parameter	Description
value	For packets: 32-bit unsigned integer value. For octets: 64-bit unsigned integer value.

When a user session is terminated based on a quota, a new non-standard termination cause is used. The value for this termination cause is 0x8744. You can customize this by modifying the value of “radius-quota-exceeded-cause” in the “ACCESS-CONTROLLER” section of the configuration file.

The text value of for the termination cause is defined in the **message.txt** file under the token “stat-quota-exceeded”. The default value for this token is “Logged out. (Quota Exceeded.)”. This value can be displayed with the ASP function GetAuthenticationErrorMessage(). See [“GetAuthenticationErrorMessage\(\)” on page 78](#) for details.

The file **message.txt** can be found in the **Internal_Pages.zip** file. See [“Sample files” on page 65](#).

A series of ASP functions are available that enable you to view quota information on the session page. For details, see [“Session quotas” on page 83](#).

Multiple login servers

This feature lets you dynamically set the URL used for retrieving custom external pages or a remote login page based on the status of a primary or secondary web server.

Syntax

primary-web-server-status-url=*URL_of_page*

secondary-web-server-status-url=*URL_of_page*

Where:

Parameter	Description
URL_of_page	<p>Specify the URL that points to the web server status file. Use HTTP or HTTPS with a port number if required.</p> <p>The status file must contain the following code:</p> <pre><?xml version="1.0" encoding="UTF-8"?> <SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/" xmlns:SOAP-ENC="http://schemas.xmlsoap.org/soap/encoding/" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:MYCOMPANY="http://www.mycompany.com/SOAP/NOCAPI/1.0/" "> <SOAP-ENV:Body> <MYCOMPANY:WebServerStatus> <MYCOMPANY:result>UP</MYCOMPANY:result> </MYCOMPANY:WebServerStatus> </SOAP-ENV:Body> </SOAP-ENV:Envelope></pre> <p>Change the <MYCOMPANY:result> line to indicate the status of the server as follows:</p> <p>Server is UP</p> <pre><MYCOMPANY:result>UP</MYCOMPANY:result></pre> <p>Server is DOWN</p> <pre><MYCOMPANY:result>DOWN</MYCOMPANY:result></pre> <p>Note: Do not change any other lines in the file.</p>

Polling

The service controller attempts to retrieve the server status file from the primary server first. If no response is received before the polling timeout expires (30 seconds by default), the service controller attempts to retrieve the server status file from the secondary server. If no response is received before the polling timeout expires, unauthenticated users attempting to login will see the Fail page with the message: "Login server is unavailable".

After initialization, the service controller continuously polls the servers to determine their status. As long as the primary server is available, it is used. If the primary server fails to respond or returns status DOWN, then the secondary server will be used, but only until the primary server comes back up.

The polling interval and polling timeout are configured by editing the following entries in the configuration file: **web-server-polling-interval** and **web-server-polling-timeout**. For information on editing the configuration file, refer to the *MSM313/MSM323 IS AP Management and Configuration Guide* for this product.

To change the error message, edit the entry **err-msg-login-server-down** in **messages.txt**.

Setting the URLs of other attributes

This feature will redefine the URLs in the following attributes if they have the same hostname as is specified for the **primary-web-server-status-url**:

- login-url
- welcome-url
- goodbye-url
- logout-url
- login-err-url
- ipass-login-url

For example, if the following attributes are defined:

```
primary-web-server-status-url=https://srv1.abc.com/status.html  
secondary-web-server-status-url=https://srv2.abc.com/status.html
```

```
login-url=https://srv1.abc.com/loginpage.html  
welcome-url=http://srv1.abc.com/mywelcome.html  
login-err-url=http://srv3.xyx.com/mywelcome.html
```

- If the primary server is up, then the URLs are not changed.
- If the primary server is down and the secondary server is up, then login-url and welcome-url are changed as follows.

```
login-url=https://srv2.abc.com/loginpage.html  
welcome-url=http://srv2.abc.com/mywelcome.html
```

If both servers are down, then the URLs are not changed.

Traffic forwarding (dnat-server)

This attribute defines the external server to which the service controller will forward traffic when an access list rule with the DNAT-SERVER action matches incoming traffic.

Two external servers can be defined with this attribute. A status polling mechanism is available that enables the service controller to determine the status of the external servers and forward traffic to the one this is operational. To activate the polling mechanism see [“Multiple DNAT servers” on page 41](#).

This attribute can be defined directly on the service controller or in the service controller’s RADIUS profile. See [“Retrieve attributes using RADIUS” on page 13](#) for configuration instructions.

Syntax

```
dnat-server=listname,hostname,port[,hostname2,port2]
```

Where:

Parameter	Description
<i>listname</i>	Specify the name of an access list definition that has its action set to DNAT-SERVER.
<i>hostname</i>	Specify the IP address or domain name of the primary server to which traffic will be redirected. Maximum length is 253 characters. If polling is not enabled, traffic is always sent to this server, even if it is down.
<i>port</i>	Specify the port on the primary server to which traffic will be redirected. Range: 1 to 65535.
<i>hostname2</i>	Specify the IP address or domain name of the secondary server to which traffic will be redirected. Maximum length is 253 characters. Traffic will only be sent to the secondary server if polling is enabled and the primary server is down. See “Multiple DNAT servers” on page 41 for details.
<i>port2</i>	Specify the port on the secondary server to which traffic will be redirected. Range: 1 to 65535.

Example

The following creates an access list called **redirect** which is used to redirect HTTP traffic for authenticated users to **server1.MYCOMPANY.com** on port **8080**.

The following entry is added to the local profile for the service controller:

```
access-list=redirect,DNAT-SERVER,tcp,all,80
```

The following entry is added to the RADIUS profile for each user:

```
dnat-server=redirect,srv1.MYCOMPANY.com,8080
```

Multiple DNAT servers

The **dnat-server** attribute supports the definition of two external servers. To make use to these servers a polling mechanism is provided. Two attributes are available to activate and configure the polling mechanism.

Syntax

```
primary-dnat-server-status-url=listname,URL_of_page  
secondary-dnat-server-status-url=listname,URL_of_page
```

Where:

Parameter	Description
<i>listname</i>	Specify the name of an access list definition that has its action set to DNAT-SERVER.

URL_of_page	<p>Specify the URL that points to a status file on the web server. Use HTTP or HTTPS with a port number if required.</p> <p>The status file must contain the following code:</p> <pre><?xml version="1.0" encoding="UTF-8"?> <SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/" xmlns:SOAP-ENC="http://schemas.xmlsoap.org/soap/encoding/" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:MYCOMPANY="http://www.mycompany.com/SOAP/NOCAPI/1.0/" "> <SOAP-ENV:Body> <MYCOMPANY:WebServerStatus> <MYCOMPANY:result>UP</MYCOMPANY:result> <!--Change this between UP and DOWN to determine the state of your server !> </MYCOMPANY:WebServerStatus> </SOAP-ENV:Body> </SOAP-ENV:Envelope></pre> <p>Change the <code><HP:result></code> line to indicate the status of the server as follows:</p> <p>Server is UP <code><MYCOMPANY:result>UP</MYCOMPANY:result></code></p> <p>Server is DOWN <code><MYCOMPANY:result>DOWN</MYCOMPANY:result></code></p> <p>Note: Do not change any other lines in the file.</p> <p>Note: If the service controller fails to receive an answer to a poll, or receives an incorrect answer (bad format, wrong result setting) it is interpreted as the server being down.</p>
-------------	--

Polling

Initially, the service controller polls the primary server at an interval of 10 minutes. As long as the primary is active, it is used. If it is not available, then the secondary server is used, but only until the primary server becomes available again.

If both servers are not available, both are polled in turn with no delay (other than the poll timeout) until one becomes available. When both servers are unavailable the access list DNAT-SERVER definition is skipped with no action taken, and processing moves to the next rule in the access list. This next rule can then be used to define the action taken when both DNAT-SERVERS are down.

The following table shows possible results when polling is active for both the primary and secondary servers.

Server 1	Server 2	Description
UP	UP	Traffic matching the DNAT-SERVER rule is forwarded to server 1.
UP	DOWN	Traffic matching the DNAT-SERVER rule is forwarded to server 1.
DOWN	UP	Traffic matching the DNAT-SERVER rule is forwarded to server 2.
DOWN	DOWN	No action is performed for the DNAT-SERVER rule. Processing moves to the next rule in the list. To accept all traffic if both servers are down, define this rule as: ACCEPT,all,all,all

Example

The following creates an access list called **redirect** which is used to redirect HTTP traffic for authenticated users to either **srv1.mycompany.com** or **srv2.mycompany.com** depending on which one is active. Port **8080** is used to forward traffic. If neither the primary or secondary DNAT-SERVER is available, all traffic is accepted.

The following entry is added to the local profile for the service controller:

```
access-list=redirect,DNAT-SERVER,tcp,all,80
access-list=redirect,ACCEPT,all,all,all
```

The following entry is added to the RADIUS profile for each user:

```
dnat-server=redirect,srv1.mycompany.com,8080,srv2.mycompany.com,8080
```

User configuration attributes

This section provides complete descriptions for all supported user attributes.

Attributes must be individually defined in the RADIUS account for each user. When a user is successfully authenticated, the attributes are retrieved by the service controller and activated.

Standard RADIUS attributes

The following standard RADIUS attributes are supported.

Note: Strings are defined as 1 to 253 characters long.

Access request

- **Acct-Session-Id** (32-bit unsigned integer): Random value generated by the service controller.
- **NAS-Identifier** (string): The NAS ID set on the **Security > RADIUS > Add New Profile** page for the profile being used.
- **NAS-Ip-Address** (32-bit unsigned integer): The IP address of the port the service controller is using to communicate with the RADIUS server.

- **NAS-Port** (32-bit unsigned integer): A virtual port number starting at 1. Assigned by the service controller.
- **NAS-Port-Type** (32-bit unsigned integer): Always set to 19, which represents WIRELESS_802_11.
- **Calling-Station-Id** (string): MAC address of the user's station in IEEE format. For example: 00-02-03-5E-32-1A.
- **Called-Station-Id** (string): This is set to the MAC address of the service controller's wireless/LAN port in IEEE format. For example: 00-02-03-5E-32-1A. To use the MAC address of the Internet port, you must edit the config file and change the setting of **radius-called-station-id-port** to **WAN** in the <ACCESS-CONTROLLER> section.
- **State** (string): As defined in RFC 2865.
- **Framed-IP-Address** (32-bit unsigned integer): IP Address as configured on the client station (if known) to the service controller.
- **Framed-MTU** (32-bit unsigned integer): Hard-coded value of 1496. The value is always four bytes lower than the wireless MTU maximum which is 1500 bytes in order to support IEEE802dot1x authentication.
- **Connect-Info** (string): The string "HTTPS" or "IEEE802.1X".
- **Service-Type** (32-bit unsigned integer): RADIUS service type.
- **Message-Authenticator** (string): As defined in RFC 2869. Always present even when not doing an EAP authentication. length = 16 bytes.
- **User-Name** (string): The username assigned to the user or a device when using MAC authentication.
- **User-Password** (string): The password supplied by a user or device when logging in. Encoded as defined in RFC 2865. Only present when the authentication method for the RADIUS profile is set to PAP.
- **Colubris-AVPair**: See ["HP vendor-specific attributes" on page 48](#).

The following attributes are mutually exclusive depending on the RADIUS authentication method.

- **CHAP-Password** (string): The password assigned to the service controller on the **Public access > Attributes** page. Encoded as defined in RFC 2865. Only present when the authentication method for the RADIUS profile is set to CHAP.
- **CHAP-Challenge** (string): Randomly generated by the product. As defined in RFC 2865. Only present when the authentication method for the RADIUS profile is set to CHAP. Length = 19 bytes.
- **MSCHAP-Challenge** (string): As defined in RFC 2433. Only present when the authentication method for the RADIUS profile is set to MSCHAPv1 or MSCHAPv2. Length = 8 bytes.
- **MSCHAP-Response** (string): As defined in RFC 2433. Only present when the authentication method for the RADIUS profile is set to MSCHAPv1. Length = 49 bytes.
- **MSCHAPv2-Response** (string): As defined in RFC 2759. Only present when the authentication method for the RADIUS profile is set to MSCHAPv2. Length = 49 bytes.
- **EAP-Message** (string): As defined in RFC 2869. Only present when the authentication method for the RADIUS profile is set to EAP-MD5.

- **Chargeable User Identity** (CUI) (string): As defined in RFC-4372. The CUI is used to associate a unique identifier with a user so that the user can be identified (for billing, authentication or other purposes) when roaming outside of their home network.

Access accept

These values override the settings for their corresponding Colubris AVPair attributes.

- **Acct-Interim-Interval** (32-bit unsigned integer): When present, it enables the transmission of RADIUS accounting requests of the Interim Update type. Specify the number of seconds between each transmission.
- **Session-Timeout** (32-bit unsigned integer): Maximum time a session can be active. The user must re-authenticate when this timer expires. Omitting this attribute or specifying 0 disables the feature.
- **Idle-Timeout** (32-bit unsigned integer): Maximum idle time in seconds allowed for the user. Once reached, the user session is terminated with termination-cause IDLE-TIMEOUT. Omitting the attribute or specifying 0 disables the feature.
- **Class** (string): As defined in RFC 2865. Multiple instances are supported.
- **EAP-Message** (string): Supported only when authentication is 802.1X or EAP-MD5. Note that the content will not be read as the RADIUS Access Accept is overriding whatever the indication is contained inside this packet.
- **MS-MPPE-Recv-Key**: As defined by RFC 3078.
- **MS-MPPE-Send-Key**: As defined by RFC 3078.
- **Termination-Action**: As defined by RFC 2865. If set to 1, a new Access Request is sent. If an Access Accept is returned, the service controller then extends the user's session timeout, and if applicable, session quotas according the value returned by the RADIUS server.
- **Colubris-AVPair**: See [“HP vendor-specific attributes” on page 48](#).
- **Tunnel-Type**: Only used when assigning a specific VLAN number to a user. In this case it must be set to VLAN.
- **Tunnel-Medium-Type**: Only used when assigning a specific VLAN number to a user. In this case it must be set to 802.
- **Tunnel-Private-Group-ID**: Only used when assigning a specific VLAN number to a user. In this case it must be set to the VLAN ID.
- **Chargeable User Identity** (CUI) (string): As defined in RFC-4372. The CUI is used to associate a unique identifier with a user so that the user can be identified (for billing, authentication or other purposes) when roaming outside of their home network.

Access reject

- **MSCHAP-Error** (string): A MSCHAP specific error as defined by RFC 2433.
- **Reply-Message** (string): This string (as defined in RFC 2865) is recorded and passed as is to the GetRadiusReplyMessage() asp function. Multiple string are supported to a maximum length of 252 bytes.

- **EAP-Message** (string): Only supported when authentication is EAP-MD5 or with IEEE802dot1x. Note that the content will not be read as the RADIUS Access Reject is overriding whatever indication contained inside this packet. As defined in RFC 2869.

Access challenge

- **EAP-Message** (string): One or more occurrences of this attribute is supported inside the same packet. All occurrence are concatenate and transmitted to the IEEE802dot1x client as is. As defined in RFC 2869.
- **State** (string): As defined in RFC 2865.

Accounting request

Accounting start/stop/interim-update

- **Acct-Session-Id** (32-bit unsigned integer): Random value generated by the service controller.
- **NAS-Identifier** (string): The NAS ID set on the **Security > RADIUS > Add New Profile** page for the profile being used.
- **NAS-Ip-Address** (32-bit unsigned integer): The IP address of the port the service controller is using to communicate with the RADIUS server.
- **NAS-Port** (32-bit unsigned integer): A virtual port number starting at 1. Assigned by the service controller.
- **NAS-Port-Type** (32-bit unsigned integer): Always set to 19, which represents WIRELESS_802_11.
- **Calling-Station-Id** (string): The MAC address of the user's station in IEEE format. For example: 00-02-03-5E-32-1A.
- **Called-Station-Id** (string): The MAC address of the service controller's LAN port or the APs downstream port if location-aware authentication is enabled.
- **Class** (string): As defined in RFC 2865. Multiple instances are supported.
- **User-Name** (string): The username assigned to the user or to a device when using MAC authentication.
- **Framed-IP-Address** (32-bit unsigned integer): IP Address of the user's station.
- **Acct-Status-Type** (32-bit unsigned integer): Supported value are Start (1), Interim Update (3), and Stop (2).
- **Acct-Event-Timestamp** (32-bit unsigned integer): As defined in RFC 2869.
- **Acct-Delay-Time** (32-bit unsigned integer): As defined in RFC 2865.
- **Acct-Authentic** (32-bit unsigned integer): Always set to 1 which means RADIUS.
- **Chargeable User Identity** (CUI) (string): As defined in RFC-4372. The CUI is used to associate a unique identifier with a user so that the user can be identified (for billing, authentication or other purposes) when roaming outside of their home network.

Accounting stop/interim-update

- **Acct-Session-Time** (32-bit unsigned integer): Number of seconds this session since this session was authenticated.

- **Acct-Input-Octets** (32-bit unsigned integer): Low 32-bit value of the number of octets/bytes received by the user.
- **Acct-Input-Gigawords** (32-bit unsigned integer): High 32-bit value of the number of octets/bytes received by the user.
- **Acct-Input-Packets** (32-bit unsigned integer): Number of packets received by the user.
- **Acct-Output-Octets** (32-bit unsigned integer): Low 32-bit value of the number of octets/bytes sent by the user.
- **Acct-Output-Gigawords** (32-bit unsigned integer): High 32-bit value of the number of octets/bytes sent by the user. As defined in 2869.
- **Acct-Output-Packets** (32-bit unsigned integer): Number of packets sent by the user.

Accounting stop only

- **Acct-Terminate-Cause** (32-bit unsigned integer): Termination cause for the session See RFC 2866 for possible values.

Accounting response

None.

Attribute settings after reauthentication

When the Location change notification option is enabled on the **Public access > Access control** page, client stations are reauthenticated when they switch to:

- a wireless cell with a different SSID
- a VSC with different VLAN ID
- an access point with a different MAC address
- an access point with a different group name
- different wireless mode (802.11a/b/g)

After reauthentication, the values for the following attributes are added to the user's current settings:

- Session-Timeout (standard RADIUS attribute)
- max-input-packets (Colubris vendor-specific attribute)
- max-output-packets (Colubris vendor-specific attribute)
- max-total-packets (Colubris vendor-specific attribute)
- max-input-octets (Colubris vendor-specific attribute)
- max-output-octets (Colubris vendor-specific attribute)
- max-total-octets (Colubris vendor-specific attribute)

Wi-Fi Alliance vendor-specific attribute

For each user profile you can receive the following Wi-Fi-AVPair attributes. The values for these attributes are set globally on the **Public access > Access control** page.

Access request and Accounting Request

Location-Name

- SMI network management private enterprise code = 14122
- Vendor-specific attribute type number = 2
- Attribute type = string

Location-ID

- SMI network management private enterprise code = 14122
- Vendor-specific attribute type number = 1
- Attribute type = string

HP vendor-specific attributes

For each user profile, the following Colubris-AVPair attributes are sent when requesting authentication (RADIUS Request) or returned upon successful authentication (RADIUS Accept). Possible values for all instances are grouped into the following categories:

Access request

Feature	Description	See
Group name	Sends the group name of the wireless access point the user is associated with.	Page 49
NAT port range	Returns the port range that user traffic is being sent on.	Page 49
SSID	Sends the SSID of the wireless access point the user is associated with.	Page 50
Incoming VLAN ID	Sends the VLAN ID that the user traffic is being received on by the service controller.	Page 50

Access accept

Feature	Description	See
Access list	Activates support for an access list.	Page 50
Bandwidth level	Sets the bandwidth control level for the user's session.	Page 51
Colubris-Intercept	Redirects user traffic into a GRE tunnel.	Page 51
Data rate	Sets the transmit and receive data rate limitation for a user's session.	Page 51
One-to-one NAT	Activates support for one-to-one NAT. This feature only applies to traffic using PPTP on the Internet port.	Page 52
Quotas	Enables upload and download limits to be set individually for each user.	Page 52
SMTP redirection	Activates support for the service controller e-mail redirection feature.	Page 37
Station polling	Configures support for client station polling on a per-user basis.	Page 54
URLs for custom HTML pages	Enables you to customize the public access interface for a particular user.	Chapter 3
Redirect URL	Enables you to redirect users with an access list definition.	Page 54

Group name

Note: This feature applies only when location-aware authentication is configured on the VSC to which a user is associated.

This attribute is set to the group name of the access point the user is associated with in the access request packet.

Use the following Colubris-AVPair value string:

```
group=value
```

Where:

Parameter	Description
value	Name of the access point group the user is associated with.

NAT port range

Note: This feature applies only when the **Limit NAT port range** option is enabled on the **Network > Ports > Internet port** page.

The service controller returns the port range that the user's traffic is being sent/received on. This applies to TCP and UDP traffic only.

Use the following Colubris-AVPair value string:

```
nat-port-range=startport-endport
```

Where:

Parameter	Description
startport	Starting port number for the user's traffic.
endport	Ending port number for the user's traffic.

SSID

Note: This feature applies only when location-aware authentication is configured on the VSC to which a user is associated.

This attribute is set to the SSID of the access point to which the user is associated. It is returned in the access request packet.

Use the following Colubris-AVPair value string:

```
ssid=value
```

Where:

Parameter	Description
value	SSID of the access point to which a user is associated.

Incoming VLAN ID

This attribute is set to the VLAN that user traffic is received on.

Use the following Colubris-AVPair value string:

```
incoming-vlan-id=value
```

Where:

Parameter	Description
value	The VLAN ID that traffic for this user is received on by the service controller. This does not necessarily imply that the user traffic is on this VLAN.

Access list

An access list is a set of rules that govern how the service controller controls user access to protected network resources (those attached to the service controller's Internet port). Access lists are defined in the profile for the service controller (see ["Access lists" on page 25](#)) and are activated in the user profiles as needed.

Only one access list can be activated per profile.

Use the following Colubris-AVPair value string:

```
use-access-list=uselistname
```

Where:

Parameter	Description
uselistname	Specify the name of an existing access list. This list is activated for the current profile.

Bandwidth level

This attribute sets bandwidth level for a user's session. The actual data rate associated with a bandwidth level is defined on the **Network > Bandwidth control** page.

Use the following Colubris-AVPair value string:

```
bandwidth-level=level
```

Where:

Parameter	Description
level	Specify one of the following the bandwidth levels for the user's session: VERY-HIGH HIGH NORMAL LOW

Colubris-Intercept

For each user profile, you can specify the Colubris-Intercept attribute to enable traffic from the user to be redirected by the egress mapping in a VSC.

The following attribute values are valid:

- 0: Do not intercept user traffic.
- 1: Intercept user traffic and redirect according to VSC egress mapping.

Data rate

This attribute sets the transmit and receive rates for a user's session. These rates are applied on a per-user basis providing direct control of a user's throughput in Kbps.

Note: The settings for bandwidth level always take precedence over user data rates. This means if you set a data rate which exceeds the configured bandwidth level, the rate is capped at the bandwidth level.

Use the following Colubris-AVPair value string:

```
max-output-rate=rate  
max-input-rate=rate
```

Where:

Parameter	Description
rate	Maximum transmit or receive speed in Kbps.

One-to-one NAT

Note: This feature only applies to traffic using PPTP on the Internet port.

Add this attribute if the user requires a unique IP address when NAT is enabled on the service controller.

Use the following Colubris-AVPair value string:

```
one-to-one-nat=value
```

Where:

Parameter	Description
value	Set this to 1 to activate one-to-one NAT support.

Quotas

These attributes let you define upload and download limits for each user. Limits can be defined in terms of packets or octets (bytes).

Use the following Colubris-AVPair value string:

```
max-input-packets=value  
max-output-packets=value  
max-input-octets=value  
max-output-octets=value  
max-total-octets=value  
max-total-packets=value
```

Where:

Parameter	Description
value	For packets: 32-bit unsigned integer value. For octets: 64-bit unsigned integer value.

When a user session is terminated based on a quota, a new non-standard termination cause is used. The value for this termination cause is 0x8744. You can customize this by modifying the value of “radius-quota-exceeded-cause” in the “ACCESS-CONTROLLER” section of the configuration file.

The text value of for the termination cause is defined in the message.txt file under the token “stat-quota-exceeded”. The default value for this token is “Logged out. (Quota Exceeded.)”. This value can be displayed with the ASP function GetAuthenticationErrorMessage(). See [“GetAuthenticationErrorMessage\(\)” on page 78](#) for details.

A series of ASP functions are available that enable you to display quota information on the session page. For details, see [“Session quotas” on page 83](#).

SMTP redirection

The service controller is able to provide SMTP email service on a per-user basis. This enables users to send e-mail while on the road without the restrictions imposed by most ISPs regarding the source address of outgoing mail. It works by intercepting the call to a user's e-mail server and redirecting it to an SMTP server that you configure. This setting overrides the setting of [“Default user SMTP server” on page 37](#).

Important: For mail redirection to work, the user's email server name must be publicly known. If the e-mail server name cannot be resolved, mail redirection fails.

Important: If an access list definition is active in the service controller profile that enables unauthenticated users to access their SMTP servers, the SMTP redirect feature will not work for these users.

Use the following Colubris-AVPair value string:

```
smtp-redirect=address[:port][,username,password]
```

Where:

Parameter	Description
address	Specify the IP address or domain name of the e-mail server which is used to send outgoing redirected mail.
port	Specify the port on the e-mail server to relay to. Range: 1 to 65535. Default: 25
username	<p>Specify the username required to log on to the SMTP server. Maximum 32 characters.</p> <p>Only supported if the Support authentication on SMTP proxy server option is enabled on the Public access > Access control page. Works with SMTP servers that support PLAIN, CRAM-MD5, and no authentication.</p>
password	<p>Specify the password required to log on to the SMTP server. Maximum 32 characters.</p> <p>Only supported if the Support authentication on SMTP proxy server option is enabled on the Public access > Access control page. Works with SMTP servers that support PLAIN, CRAM-MD5, and no authentication.</p>

Example 1: Proxy support on

```
smtp-redirect=smtp.mycompany.com,jimmy,letMEin  
smtp-redirect=smtp.mycompany.com:8025,jimmy,letMEin
```

Example 2: Proxy support off

```
smtp-redirect=smtp.mycompany.com  
smtp-redirect=smtp.mycompany.com:8025
```

Station polling

The service controller continually polls authenticated client stations to ensure they are active. This feature is configured using the **Query if active** parameter on the **Public access > Access control** page. If no response is received and the number of retries is reached, the client station is disconnected.

These attributes let you override the **Query if active** setting on a per-user basis.

Use the following Colubris-AVPair value string:

```
polling-arp-interval=interval  
polling-max-arp-count=count
```

Where:

Parameter	Description
<i>interval</i>	Specify how (in seconds) long to wait between polls.
<i>count</i>	Specify how many polls a client station can fail to reply to before it is disconnected.

To disable polling, set both *interval* and *count* to 0.

The initial query is always done after the client station has been idle for 60 seconds. If there is no answer to this query, the settings for *polling-arp-interval* and *polling-max-arp-count* are used to control additional retries.

Redirect URL

The *redirect-url* attribute is used to specify the target URL for redirection when using an access list with the REDIRECT action. Only one *redirect-url* attribute can be specified in each user RADIUS profile.

For more information, see [“Redirect URL” on page 31](#).

Administrator configuration attributes

If you want to support multiple administrator names and passwords, you must use a RADIUS server to manage them. The service controller only supports a single admin name and password internally (defined on the **Management > Management tool** page).

Note: Improper configuration of the administrator profile could expose the service controller to access by any user with a valid account. The only thing that distinguishes an administrative account from that of a standard user account is the setting of the service type. Make sure that a user is not granted access if service type is not Administrative,

This is the reason why it may be prudent to use a different RADIUS server to handle administrator logins. This practice reduces the risk of a bad configuration on the RADIUS server side creating a security hole.

Supported RADIUS attributes

This section presents all RADIUS and HP attributes that are supported by an administrator RADIUS account.

Note: In the following definitions, strings are defined as 1 to 253 characters in length.

Admin Access Request

- User-Name (string): The name assigned to the administrator.
- NAS-Identifier (string): The NAS ID set on the **Security > RADIUS > Add New Profile** page for the profile being used.
- Service-Type (32-bit unsigned integer): As defined in RFC 2865. Set to a value of 6, which indicates SERVICE_TYPE_ADMINISTRATIVE.
- Framed-MTU (32-bit unsigned integer): Hard-coded value of 1496. The value is always four bytes lower than the wireless MTU maximum which is 1500 bytes in order to support IEEE802dot1x authentication.
- MSCHAP-Challenge (string): As defined in RFC 2433. Only present when the authentication scheme on the **Security > RADIUS > Add New Profile** page is set to MSCHAPv1 or MSCHAPv2. Length = 8 bytes.
- MSCHAP-Response (string): As defined in RFC 2433. Only present when the authentication scheme on the **Security > RADIUS > Add New Profile** page is set to MSCHAPv1. Length = 49 bytes.

No attributes are supported for the following attribute types:

- **Admin Access Accept**
- **Admin Access Reject**
- **Admin Access Challenge**
- **Admin Accounting Request**
- **Admin Accounting Response**

3

Customizing the public access interface

Contents

Overview - - - - -	58
Site map - - - - -	59
Customizing the internal pages - - - - -	63
Customizing the external pages- - - - -	66
Using a remote login page - - - - -	70
WISPr support - - - - -	73
Location-aware authentication - - - - -	75
iPass support- - - - -	78
ASP functions - - - - -	78

Overview

The public access interface is the sequence of web pages that users use to log in, log out, and view the status of their wireless connections to the public access network.

The service controller enables you to tailor the public access interface web pages to provide a customized look-and-feel for your site. Web pages can be automatically updated using a RADIUS server, enabling you to manage multiple units effortlessly.

Note: Users using PDAs that support a single browser window will have difficulty using the public access interface in its standard configuration. For information on how to correct this problem, see [“Supporting PDAs” on page 69](#).

Common configuration tasks

The following table lists some common configuration tasks and indicates where to find more information.

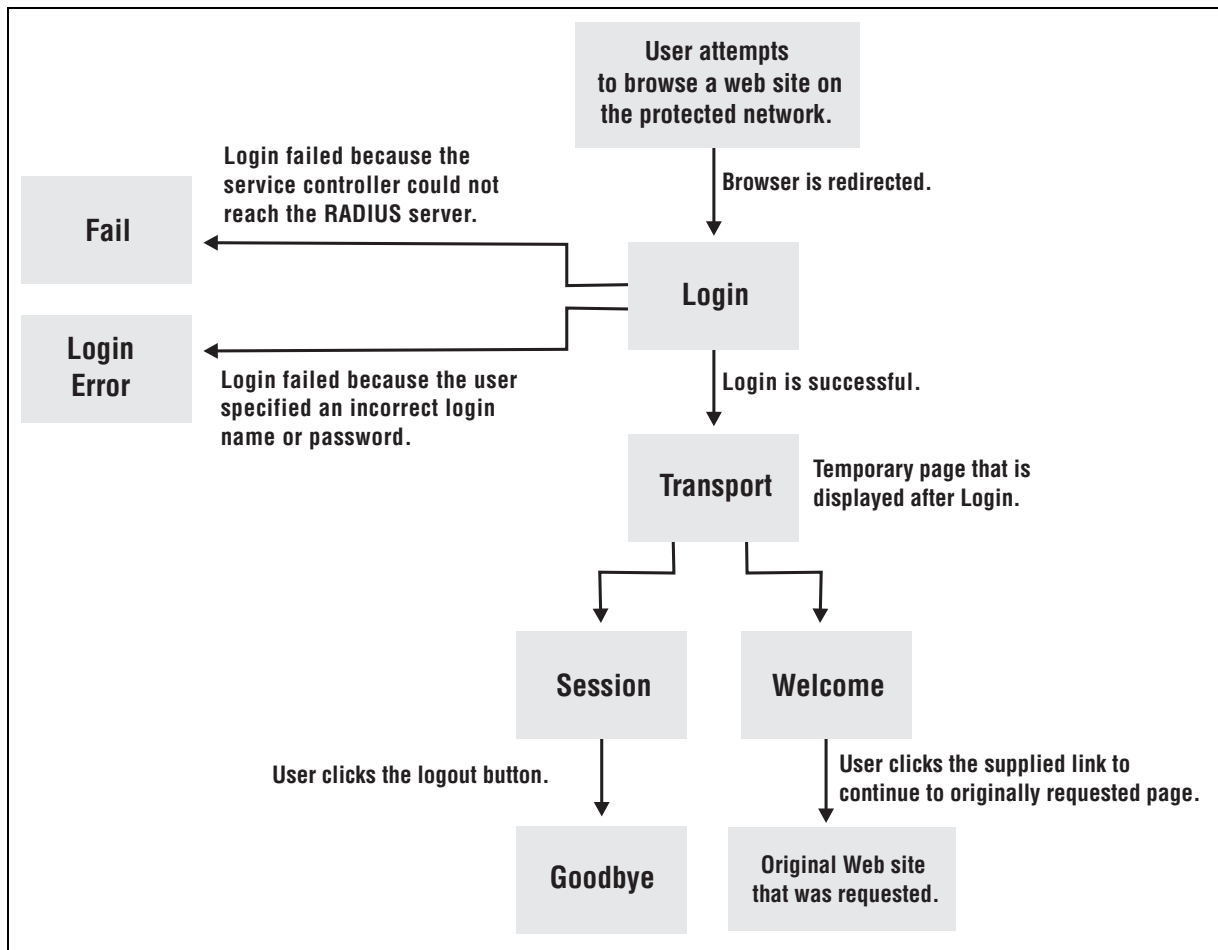
Task	For instructions
Changing the Login page and logo	See page 63
Hosting the login page on your own web server	See page 70
Displaying custom Welcome or Goodbye pages	See page 68
Delivering custom content based on a user's location in the network	See page 69
Supporting PDAs	See page 69
Restricting user logins based on their location in the network	See page 75

Site map

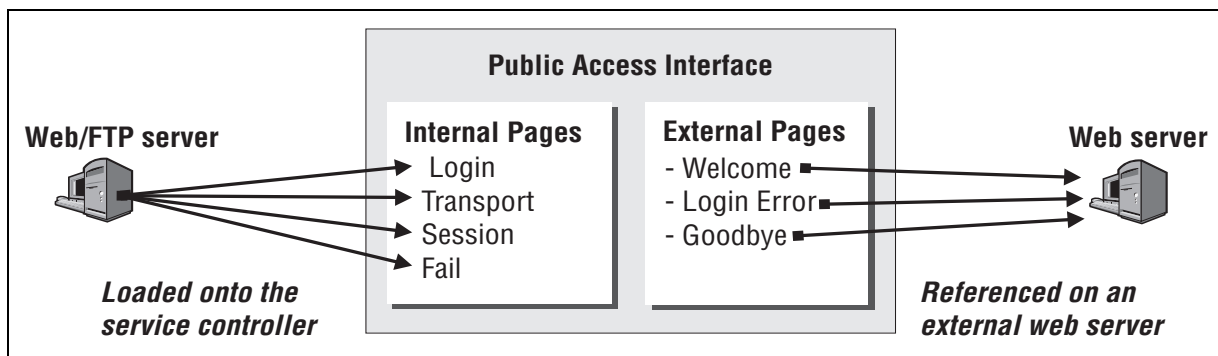
This section describes how the public access interface is structured and provides an overview of each component.

Structure

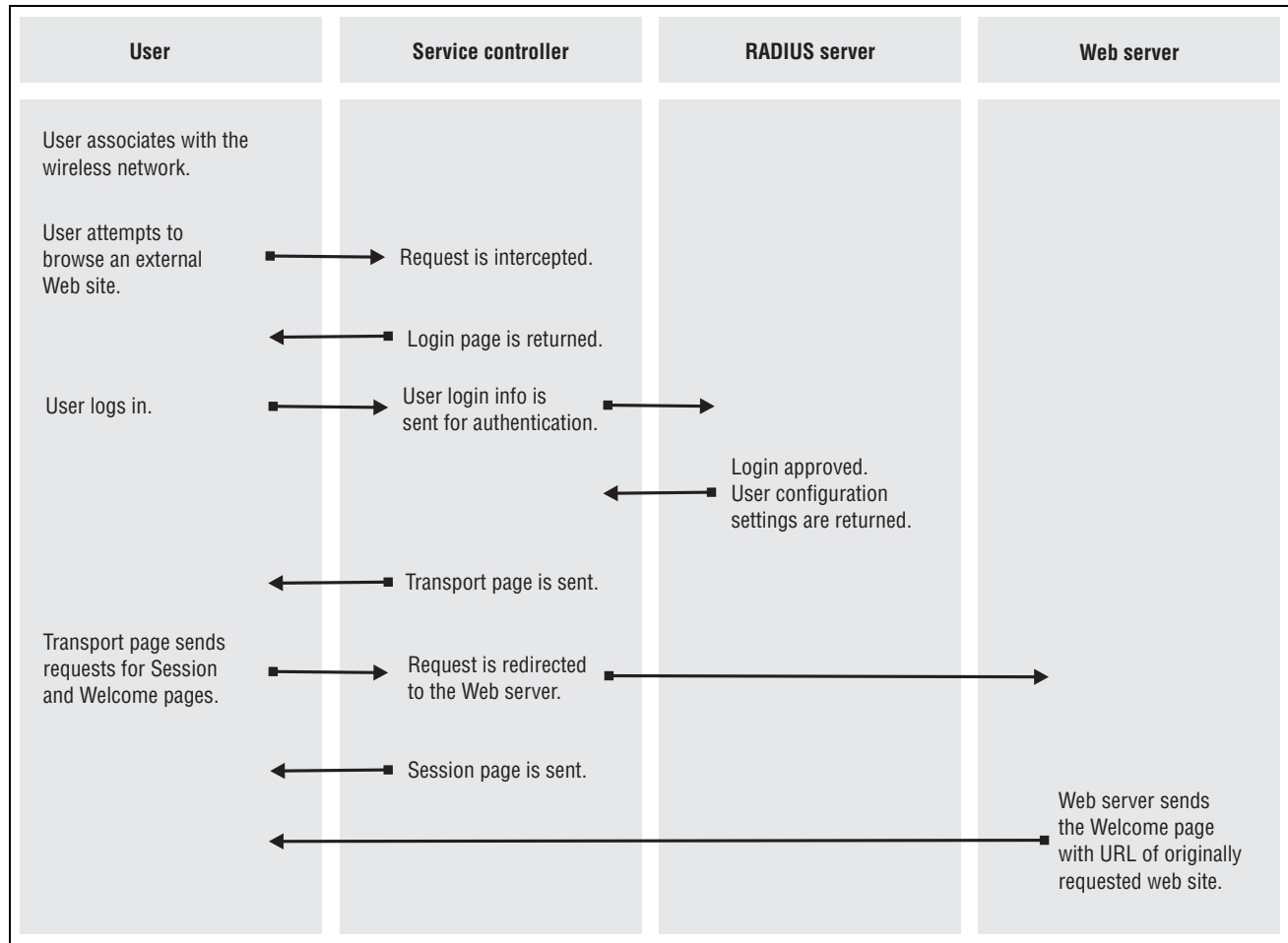
The default public access interface comprises seven pages and is structured as follows:



Pages are categorized into two groups: internal pages and external pages.



The following diagram shows the sequence of events that occur when a user attempts to browse an external web site. This example assumes that the default public access setup is being used with a RADIUS server.



Internal pages

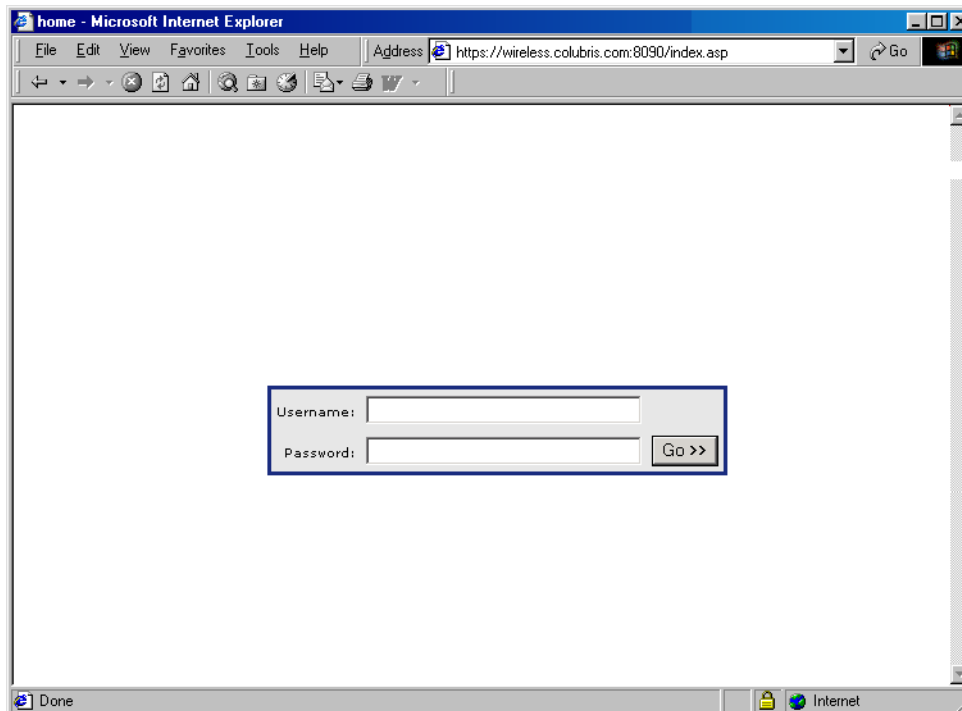
The internal pages reside on the service controller. You can use the default pages supplied with the service controller, or you can replace them with customized pages of your own design.

Login page

The Login page contains a single graphic element suitable for a logo or other identifying element and two fields: username and password.

Note: Users that are authenticated via 802.1X/WPA or MAC address are automatically logged in and in most cases will not see the Login page.

The default Login page:



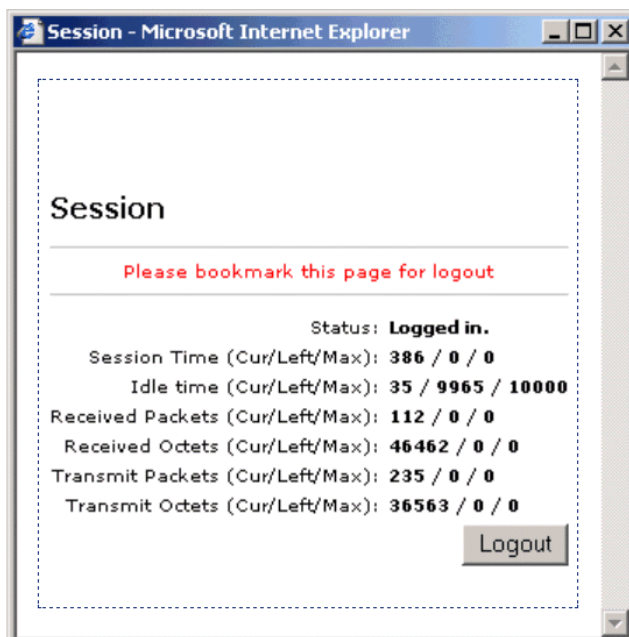
You can also create a remote login page that resides on an external web server and is not downloaded to the service controller. For details see [“Using a remote login page” on page 70](#).

Transport page

The **Transport** page appears briefly and spawns the Session and Welcome pages.

Session page

The Session page shows usage statistics for the session, as well as the logout button that the user clicks to terminate the session. This is the default Session page:



Note: A popup blocker will prevent the session page from being displayed.

Note: Users that are authenticated via MAC address are automatically logged in and do not see the Login page.

Managing the Session page

The Session page opens automatically after the user logs in. By default it contains the logout button. Without the Session page the user cannot log out. The following URL can be used to reopen the session page if a user accidentally closes it.

```
http://service_controller_hostname:port/session.asp.
```

For example:

```
http://wireless.colubris.com:8080/session.asp
```

Launching the Session page from the Welcome page

You can embed the following URL on the Welcome page to dynamically link to the Session page:

```
<a href="http://service_controller_hostname:port/session.asp">Session page</a>
```

Forcing a logout

You can force a logout with this URL:

```
http://service_controller_hostname:port/goform/HtmlLogout
```

For example:

```
http://wireless.colubris.com:8080/goform/HtmlLogout
```

Users with PDAs

Users using PDAs that support only a single browser window never see the session page. This makes it impossible for them to log out. For information on how to correct this problem, see [“Supporting PDAs” on page 69](#).

Fail page

The Fail page appears if the service controller cannot contact the RADIUS server to authenticate a user.

Note: Users using 802.1X/WPA may not see the **Fail** page.

External pages

The external pages are hosted by an external web server. The service controller redirects users to these pages as required.

The service controller can be configured to use a different external web page for each user if required.

Welcome page

The Welcome page appears after the user has successfully logged in and can be used to provide information about the public access network and its options. The Welcome page also includes a link to the page that was originally requested by the user. If the service controller cannot reach the custom URL specified for the Welcome page, or if a custom URL is not defined, it jumps directly to the page originally requested by the user.

Goodbye page

The **Goodbye** page acknowledges a user logout.

Login error page

The Login error page appears if the user cannot be authenticated. The reason is shown on the page. You can customize the messages on this page by editing the file **messages.txt**. See [“Customizing error messages” on page 66](#).

Remote Login page

Instead of using the internal Login page you can create a remote login page that resides on an external web server. For details see [“Using a remote login page” on page 70](#).

Customizing the internal pages

This section explains how to create new internal pages, as well as how to edit the shared image file (logo) and the message file.

Creating new internal pages

If you do not already have them, you can get the sample files as described in [“Sample files” on page 65](#).

Note: Do not create new pages by saving an internal page while viewing it within your web browser. If you do so, the server-side code is removed, and the resulting pages will not work.

The internal pages use a number of ASP functions to display status information. You can also use these functions to enhance your custom pages. For descriptions of these functions, see [“ASP functions” on page 78](#).

Important restrictions

Because the internal pages must be loaded onto the service controller, the following restrictions apply to their construction.

- **You must specify a URL for ALL internal pages, even if you want to change only one page. Simply use copies of the standard internal pages for the pages you do not want to change.**
- Do not alter the ID tags “<!-- HP -->” & “<!-- Custom -->” located at the top of the page.
- Do not alter any JavaScript code, except for the **Session** window parameters *width* and *height*.

- Only one image can be included on these pages. It must be a *.gif* file, and HP recommends that the file size be less than 20K. This same image file is shared by all pages and must be resident on the service controller. For instructions on how to change the image, see [“Sample files” on page 65](#).
- Do not alter any occurrences of “Get...();” or “GetWelcomeURL();”
- Do not alter any form elements. Leave intact all names and values.
- Do not change the filename extensions of the internal pages.

Loading new internal pages

To load new internal pages, you must define the URLs where the service controller can download them using a service controller attribute. The attribute can be defined in the RADIUS account for the service controller (if you use a RADIUS server) or they can be locally configured.

See the following topics for more information:

- [“Configuring the public access network” on page 12](#).
- [“Service controller attributes” on page 21](#).

Colubris-AVPair value strings

The following table presents the Colubris-AVPair value strings used for customizing the internal pages.

Internal page	Colubris-AVPair value string	Notes
Login	<code>login-page=URL_of_page</code>	Required. (Unless a remote login page is being used as explained in “Using a remote login page” on page 70).
Transport	<code>transport-page=URL_of_page</code>	Required.
Session	<code>session-page=URL_of_page</code>	Required.
Fail	<code>fail-page=URL_of_page</code>	Required.
Re-usable image	<code>logo=URL_of_gif_file</code>	Required. This image is shared by all pages.
Error messages	<code>messages=URL_of_text_file</code>	Optional. These messages appear when various error conditions occur.

Note: The maximum length of any internal page URL is 512 characters. If this is exceeded (when using placeholders for example), the URL is truncated. HP therefore recommends that you specify the most-important placeholders first.

Note: The internal pages can only be changed as a group. You cannot, for example, just use the login-page string in a RADIUS profile. You must use all required items. This means that the minimum set you can specify is as follows:

```
login-page= URL_of_page
transport-page= URL_of_page
session-page= URL_of_page
fail-page= URL_of_page
logo= URL_of_gif_file
```

Placeholders

The following optional placeholders can be appended to the Colubris-AVPair value strings for the internal pages. These placeholders are not available in local mode.

Placeholder	Description
%n	Returns the NAS ID assigned to the service controller. By default, this is the unit's serial number.
%s	Returns the RADIUS login name assigned to the service controller. By default, this is the unit's serial number.
%i	Returns the domain name assigned to the service controller's Internet port.
%a	Returns the IP address of the service controller's interface that is sending the authentication request.

Example

Sample files

Sample public access files are referenced in this chapter. To get these files, go to the HP support Website at: www.hp.com/networking/support and select the option needed to get to the MSM product documentation page (ProCurve).

Select the documentation page for *MSM313 and MSM323 Integrated Services Access Points*. You will find the Public Access Examples zip near the other MSM313/MSM323 documentation. Download the zip file and extract its content to a folder on your computer. The sample files include **Internal_Pages.zip** and **External_Pages.zip**.

Changing the login page and logo

1. Create a folder called **newpages** on your web sever.
2. Create a file called **logo.gif** that contains your logo and place it in the **newpages** folder.
3. Copy the following files from **Internal_Pages.zip** and place them in the **newpages** folder.
 - login.html
 - transport.html
 - session.html
 - fail.html

4. Edit **login.html** to customize it for your site.
5. Add the following entries to the **Configured attributes** table on the **Public access > Attributes** page. (You can also define these attributes in the RADIUS profile for the service controller if you are using a RADIUS server.)

```
login-page=web_server_URL/newpages/login.html  
transport-page=web_server_URL/newpages/transport.html  
session-page=web_server_URL/newpages/session.html  
fail-page=web_server_URL/newpages/fail.html  
logo=web_server_URL/newpages/logo.gif
```

Customizing error messages

Several of the internal pages use the functions `GetAuthenticationErrorMessage()` and `GetSessionStateMessage()` to return a string from the file **message.txt**. You can customize the messages in this file for your installation as follows:

1. Create a folder called **newpages** on your web sever.
2. Copy the file **messages.txt** from **Internal_Pages.zip** and place it in the **newpages** folder.
3. Edit **messages.txt** with an ASCII editor. Customize the messages to suit your installation.
4. Add the following entry to the **Configured attributes** table on the **Public access > Attributes** page. (You can also define this attribute in the RADIUS profile for the service controller if you are using a RADIUS server.)

```
messages=web_server_URL/newpages/messages.txt
```

Customizing the external pages

This section explains how to customize the three external pages: Welcome, Login error, and Goodbye.

Creating new external pages

Unlike the internal pages, the external pages do not have any restrictions on their construction since they reside on a third-party server. See [“Customization examples” on page 68](#).

Activating new external pages

To activate new external pages, you must define their URLs using the Colubris-AVPair value string when you create a RADIUS profile for the service controller or a user. See [“Configuring the public access network” on page 12](#).

When the service controller authenticates itself, or a user, it retrieves the URLs for the custom pages, then automatically redirects users to them when required.

Note: The service controller maintains a separate copy of the URLs for external pages for each user. This means it is possible to provide different pages for each user. See [“Displaying custom welcome and goodbye pages” on page 68](#).

The following table presents the Colubris-AVPair value strings used for customizing the external pages.

Attribute	Notes
login-err-url= <i>URL_of_page[placeholder]</i>	Access to the web server hosting this page must be granted to all unauthenticated users. Do this with an appropriate access list definition. (Users can see this page <i>before</i> they are logged in.)
welcome-url= <i>URL_of_page[placeholder]</i>	The user is authenticated, so the welcome page can be located on any URL reachable by the user.
goodbye-url= <i>URL_of_page[placeholder]</i>	Access to the web server hosting this page must be granted to all unauthenticated users. Do this with an appropriate access list definition. (Users see this page <i>after</i> they are logged out.)

Note: The maximum length of any external page URL is 512 characters. If this is exceeded (when using placeholders for example), the URL is truncated. HP therefore recommends that you specify the most-important placeholders first.

An important feature of the external pages is that they make it easy to deliver a unique experience for each user. By appending the following optional placeholders to the Colubris-AVPair value strings for the external pages, you can pass important information to the web server. Server-side code can process this information to generate custom pages on-the-fly.

Placeholder	Description
%c	Returns the IP address of the user's computer.
%d	Returns the WISPr location-ID. Supported for login-url only.
%e	Returns the WISPr location-Name. Supported for login-url only.
%l	Returns the URL on the service controller where user login information should be posted for authentication. This option is used with the remote login page feature. By default, this value is URL encoded. (To enable/disable URL encoding, set the value of url-encode in the <ACCESS-CONTROLLER> section in the configuration file.)
%n	Returns the NAS ID assigned to the service controller. By default, this is the unit's serial number. Not supported in local mode.
%s	Returns the RADIUS login name assigned to the service controller. By default, this is the unit's serial number.
%u	Returns the login name of the user.
%o	Returns the original URL requested by the user. By default, this value is URL encoded. (To enable/disable URL encoding, set the value of url-encode in the <ACCESS-CONTROLLER> section in the configuration file.)
%i	Returns the domain name assigned to the service controller's Internet port.
%p	Returns the IP port number on the service controller where user login information should be posted for authentication.

Placeholder	Description
%a	Returns the IP address of the service controller's interface that is sending the authentication request.
%E	When the location-aware feature is enabled, returns the ESSID of the wireless access point the user is associated with.
%P	When the location-aware feature is enabled, returns the wireless mode ("ieee802.11a", "ieee802.11b", "ieee802.11g") the user is using to communicate with the access point.
%G	When the location-aware feature is enabled, returns the group name of the wireless access point the user is associated with.
%C	When the location-aware feature is enabled, returns the Called-station-id content for the wireless access point the user is associated with.
%r	Returns the string sent by the RADIUS server when an authentication request fails. The RADIUS server must be configured to support this feature. The information contained in the returned string depends on the configuration of the RADIUS server.
%m	Returns the MAC address of the client station that is being authenticated.
%v	Returns the VLAN assigned to the client station at the service controller's ingress (LAN port).

Customization examples

If you do not already have them, you can get the sample files as described in ["Sample files" on page 65](#).

Displaying custom welcome and goodbye pages

This example shows how to display unique welcome and goodbye pages for specific users or groups of users.

For this example, assume you have two sets of users: basic and premium. To distinguish the two groups, you have set up the user accounts on the RADIUS server accordingly. (Perhaps you are using access lists to restrict each group to a different section of the public network as described in ["Access list example" on page 29](#)).

1. Create the following two folders on your web sever: **basic** and **premium**.
2. Copy **welcome.html** and **goodbye.html** from **External_Pages.zip** into each folder.
3. Customize **welcome.html** and **goodbye.html** in each folder for each set of users.
4. Add the following entry to the RADIUS profile for the basic users.

```
welcome-url=web_server_URL/basic/welcome.html  
goodbye-url=web_server_URL/basic/goodbye.html
```

5. Add the following entry to the RADIUS profile for the premium users.

```
welcome-url=web_server_URL/premium/welcome.html  
goodbye-url=web_server_URL/premium/goodbye.html
```

6. Add the following entry to the RADIUS profile for the service controller. This gives all unauthenticated users access to the web server hosting the goodbye page.

```
access-list=loginserver,ACCEPT,tcp,web_server_IP_address,port_number
```

Delivering dynamically generated content

Another way to generate custom pages is to add placeholders in the URLs for the custom external pages and then use server-side scripting to dynamically create the pages. This method provides a powerful mechanism to automatically generate completely customized pages on a per-user basis. Rather than designing one or more static pages, as in the previous example, the custom pages in this example can be built on-the-fly based on user preferences stored in a central database, or based on a user's location within the network.

For example, if you want to generate a custom welcome page for each user:

1. Add the following entry to the RADIUS profile for the service controller.

```
welcome-url=web_server_URL/premium/welcome.html ?loginname=%u&IPAddress=%i
```

2. Create a server-side script to retrieve the user's login name (%u) and the service controller's IP address or domain name (%u). The script can use this information to then display a custom page based on user's preferences (stored in the server's database) and the user's location within the wireless network.

Supporting PDAs

Users using PDAs that only support a single browser window will have difficulty using the public access interface in its standard configuration.

Once a user logs in to the public access interface, two web pages are sent to their browser: the Welcome page and the Session page.

The Session page contains a logout button. Users who are unable to view this page will not be able to log out.

To solve the problem, modify the Welcome page to include a logout button.

1. Create a folder called **PDAusers** on your web sever.
2. Copy **welcome.html** and **goodbye.html** from **External_Pages.zip** into this folder.
3. Edit **welcome.html** to include a logout link with the target:

```
http://service_controller_name:port/goform/HtmlLogout.
```

For example:

```
http://wireless.colubris.com:8080/goform/HtmlLogout.
```

Add a warning to this page that tells PDA users to bookmark the Welcome page so that they can logout.

4. Add the following entry to the RADIUS profile for all PDA users.

```
welcome-url=web_server_URL/PDAusers/welcome.html
```

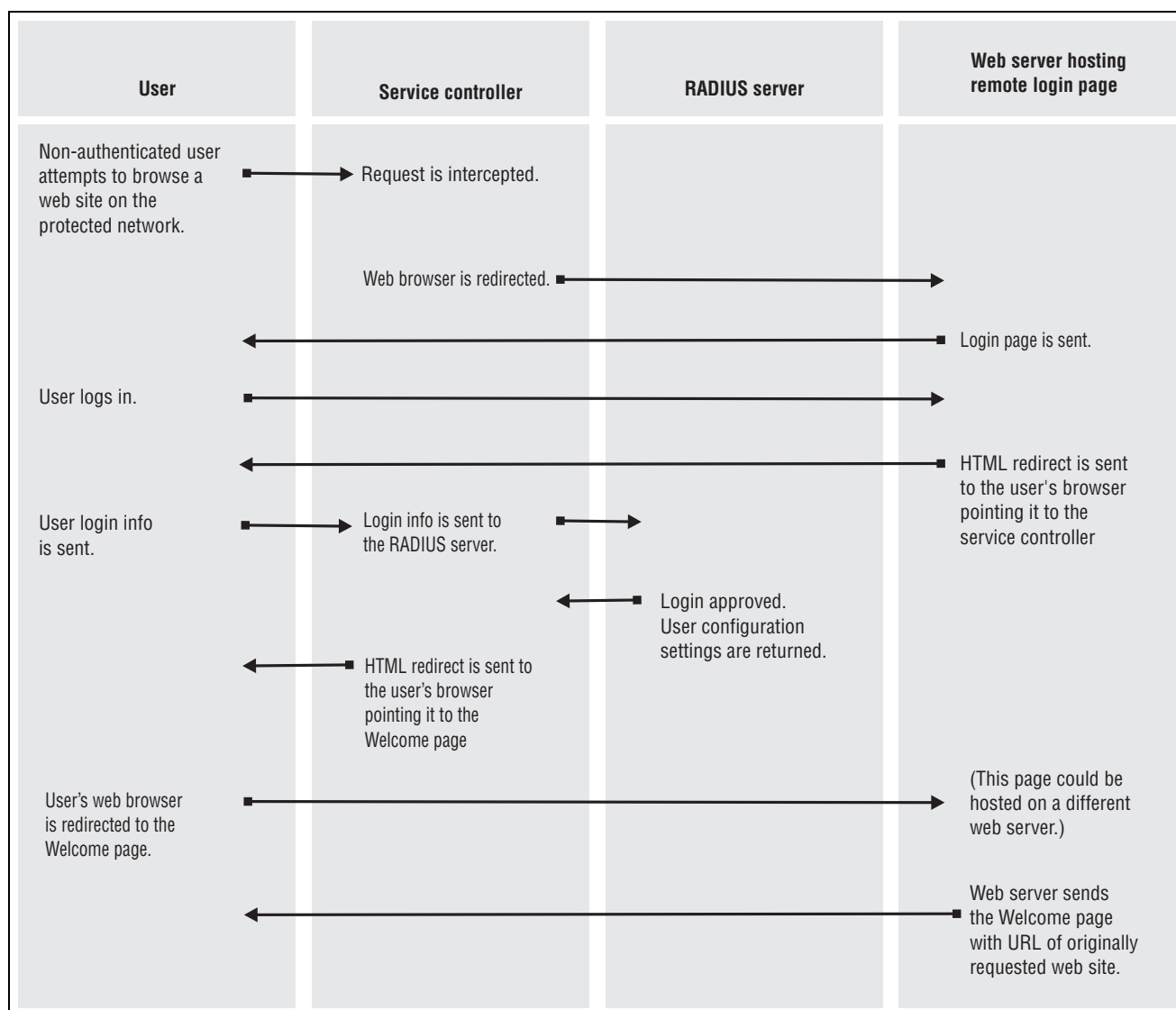
Using a remote login page

The service controller provides an option that enables you to redirect users to a remote server to log in to the public access interface instead of using the internal login page. Hosting the login page on a remote server means that the login page is completely customizable. You are not bound by the limits imposed by loading a login page onto the service controller.

How it works

Although the remote login page feature enables you to host the public access login page on a remote web server, authentication of users is still performed by the service controller through a RADIUS server or using the local user list. To accomplish this, the remote web server must send user login information back to the service controller. There are two ways this can be done: basic remote login (as described in this section), or by using the NOC-based authentication feature (described in [“Chapter 4: NOC authentication” on page 87](#)).

The following diagram shows the sequence of events for a typical user session when using a remote login page and a RADIUS server for authentication.



Activating a remote login page

To activate a remote login page, you must define the URL where the service controller can redirect login requests using a service controller attribute. The attribute can be defined in the RADIUS account for the service controller (if you are using a RADIUS server) or it can be locally configured.

Use the following Colubris-AVPair value string:

```
login-url=URL_of_the_page [placeholder]
```

Where:

Parameter	Colubris-AVPair value string
URL_of_the_page	URL of the remote login page. Access to the web server hosting this page must be granted to all unauthenticated users. Do this with an appropriate access list definition.

The following placeholders can be added to the login-url string.

Placeholder	Description
%c	Returns the IP address of the user's computer.
%d	Returns the WISPr location-ID. Supported for login-url only.
%e	Returns the WISPr location-Name. Supported for login-url only.
%l	Returns the URL on the service controller where user login information should be posted for authentication. By default, this value is URL encoded. (To enable/disable URL encoding, set the value of url-encode in the <ACCESS-CONTROLLER> section in the configuration file.)
%n	Returns the NAS ID assigned to the service controller. By default, this is the unit's serial number. Not supported in local mode.
%s	Returns the RADIUS login name assigned to the service controller. By default, this is the unit's serial number. Not supported in local mode.
%o	Returns the original URL requested by the user. By default, this value is URL encoded. (To enable/disable URL encoding, set the value of url-encode in the <ACCESS-CONTROLLER> section in the configuration file.)
%i	Returns the domain name assigned to the service controller's Internet port.
%p	Returns the port number on the service controller where user login information should be posted to for authentication.
%a	Returns the IP address of the service controller's interface that is sending the authentication request.
%E	When the location-aware feature is enabled, returns the ESSID of the wireless access point the user is associated with.
%P	When the location-aware feature is enabled, returns the wireless mode ("ieee802.11a", "ieee802.11b", "ieee802.11g") the user is using to communicate with the access point.

Placeholder	Description
%G	When the location-aware feature is enabled, returns the group name of the wireless access point the user is associated with.
%C	When the location-aware feature is enabled, returns the Called-station-id content for the wireless access point the user is associated with.
%r	Returns the string sent by the RADIUS server when an authentication request fails. The RADIUS server must be configured to support this feature. The information contained in the returned string depends on the configuration of the RADIUS server.
%m	Returns the MAC address of the wireless/wired client station that is being authenticated.
%v	Returns the VLAN assigned to the client station at the service controller's ingress (LAN port).

Note: The maximum length of the remote login page URL is 512 characters. If this is exceeded (when using placeholders for example), the URL is truncated. HP therefore recommends that you specify the most-important placeholders first.

Security issues

- HP recommends that the web server hosting the remote login page be secured with SSL (requires an SSL certificate from a well-known certificate authority), to ensure that user logins are secure. Without SSL security, logins are exposed and may be compromised, enabling fraudulent use of the network.
- Communications between the user's browser and the service controller is always SSL-based. The default certificate on the service controller generates a warning on the user's browser unless replaced with a certificate signed by a well-known certificate authority.

Example

The file **message.txt** can be found in the **Internal_Pages.zip** file. See [“Sample files” on page 65](#).

To enable a basic remote login page, do the following:

1. Create the following folder on your web sever: **newlogin**
2. Copy the following files from **Internal_Pages.zip** and place them in the **newlogin** folder.
 - login.html
 - transport.html
 - session.html
 - fail.html
 - logo.gif
3. Add the following entries to the **Configured attributes** table on the **Public access > Attributes** page. (You can also define these attributes in the RADIUS profile for the service controller if you are using a RADIUS server.)

```
login-url=web_server_URL/newlogin/login.html?loginurl=%l
transport-page=web_server_URL/newlogin/transport.html
session-page=web_server_URL/newlogin/session.html
fail-page=web_server_URL/newlogin/fail.html
logo=web_server_URL/newlogin/logo.gif
access-list=loginserver,ACCEPT,tcp,web_server_IP_address
use-access-list=loginserver
```

4. Customize **login.html** to accept username and password information from users and then send it to the service controller. You can use code similar to the following example to redirect the user's web browser to the login URL on the service controller for authentication:

```
<form action="https://wireless.colubris.com:8090/goform/HtmlLoginRequest"
method="POST">
```

For more flexibility, the remote login page should be written using a server-side scripting language such as ASP, PHP, or PERL. This enables the remote login page to take advantage of the placeholders that may have been defined in the login-url section of the RADIUS profile.

WISPr support

The public access interface provides support for WISPr (Wireless Internet Service Project Roaming) using WISPr and HP vendor-specific attributes.

WISPr vendor-specific attributes

HP supports three Wi-Fi Alliance vendor-specific attributes for Access Request and Accounting Request. These attributes are:

```
wispr-location-name=location_name
wispr-location-id=location_id
wispr-logout-url=URL
```

WISPr-Location-Name

- SMI network management private enterprise code = 14122
- Vendor-specific attribute type number = 2
- Attribute type = string

WISPr-Location-ID

- SMI network management private enterprise code = 14122
- Vendor-specific attribute type number = 1
- Attribute type = string

WISPr-Logout-url

- SMI network management private enterprise code = 14122
- Vendor-specific attribute type number = 3
- Attribute type = string

HP vendor-specific attributes

WISPr login URL

This attribute lets you define the location of the WISPr login page. The service controller automatically redirects users with WISPr-compatible wireless client software to this page. To customize the redirection use the WISPr redirect page attribute.

Use the following Colubris-AVPair value string:

```
wispr-login-url=URL_of_page
```

Where:

Parameter	Description
URL_of_page	URL of the WISPr login page.

WISPr abort login URL

This attribute lets you define the destination where the WISPr abort login will be POSTed.

Use the following Colubris-AVPair value string:

```
wispr-abort-login-url=URL_of_page
```

Where:

Parameter	Description
URL_of_page	URL where to POST the WISPr abort login.

WISPr redirect page

This attribute lets you define the location of the WISPr redirect page. Use this page to customize the code that the service controller includes in the HTTP redirect sent to a user's browser.

```
redirect-page=URL_of_page
```

Where:

Parameter	Description
URL_of_page	URL of the page containing code to use for WISPr redirect.

If this attribute is not defined the following code is used by default:

```
<!-- Colubris -->
<!-- Default -->
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">
<!--iPass
<WISPAccessGatewayParam>
  <Redirect>
    <MessageType>100</MessageType>
    <ResponseCode><% iPassGetRedirectResponseCode(); %></ResponseCode>
    <AccessProcedure><% iPassGetAccessProcedure(); %></AccessProcedure>
    <LocationName><% iPassGetLocationName(); %></LocationName>
    <AccessLocation><% iPassGetAccessLocation(); %></AccessLocation>
    <LoginURL><% iPassGetLoginUrl(); %></LoginURL>
    <AbortLoginURL><% iPassGetAbortLoginUrl(); %></AbortLoginURL>
  </Redirect>
</WISPAccessGatewayParam>
-->
<!-- Boingo
<smartClient>
  <page>
    <login>
      <login_url><% BoingoGetLoginUrl(); %></login_url>
    </login>
  </page>
</smartClient>
-->
</html>
```

Location-aware authentication

This feature enables you to control logins to the public access network based on the wireless access point a user is associated with. Once authenticated, this feature is also used to monitor and control roaming to other access points in the network.

How it works

Location-aware is automatically enabled when a VSC is set to **provide access control**. When enabled, the location-aware feature causes the service controller to return location-specific information for RADIUS-authenticated users. This information is returned:

- when the user logs in
- each time the user roams to a new access point or switches SSIDs on the same access point (which causes the user to be re-authenticated)

Note: Due to security constraints in 802.1X client software, users cannot automatically be re-authenticated when roaming to a new access point. Therefore, location-aware information cannot be returned when these user's roam.

Returned information

The service controller can return the following attributes in the RADIUS access request for all user authentications (whether initial login or re-authentication due to roaming).

- Called-station-ID (Standard RADIUS attribute)
- Colubris-specific attribute: SSID
- Colubris-specific attribute: GROUP

Note: When re-authenticating users, the returned RADIUS attribute Service-Type is set to 8744 (decimal).

Called-Station-ID value

By default, this is the MAC address of the wireless port (radio) the user is associated with. This is the MAC address of the **wvlan0** or **wvlan1** interface in IEEE format as displayed by **Tools > System Tools > Interface info**.

If required, the service controller can return other values for this attribute by setting the **Called-Station-Id content** on a per-VSC basis. The other available options are:

- SSID: SSID of the access point the user is associated with.
- GROUP: Group name of the access point the user is associated with.

Note: If the user is connected via a wired connection, the value returned is the MAC address of the service controller's wireless/LAN port. To use the MAC address of the Internet port, you must edit the config file and change the setting of **radius-called-station-id-port** to **WAN** in the <ACCESS-CONTROLLER> section.

Colubris-specific attribute: SSID

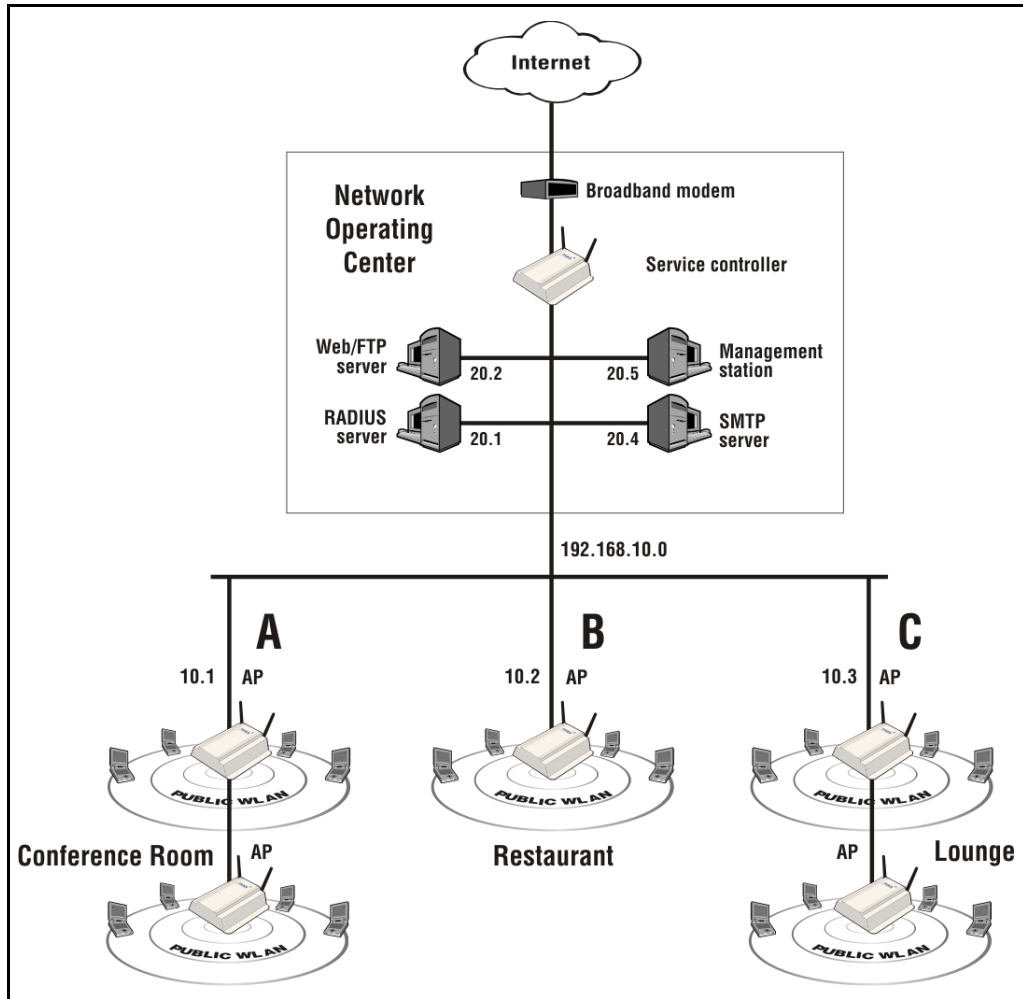
The SSID of the access point the user is associated with (wireless only).

Colubris-specific attribute: GROUP

The GROUP of the access point the user is associated with (wireless only).

Example

Consider the following topology for a fictional small hotel. The restaurant and lounge are available to all hotel users who subscribe to the wireless service. However, the conference room is available only to a specific group of guests who book it in advance.



In this example, the access points in each area are assigned the following unique group names:

- conference_room
- restaurant
- lounge

When a user logs in, server-side code can be used to determine the access point they are associated with by inspecting the Called-Station-ID. Then, using user's account information, access can either be granted or denied.

Security

The service controller accepts location-aware information only from HP APs that have a matching shared secret to its own.

iPass support

The service controller provides support for the Generic Interface Specification from iPass which enables you to create an iPass-compatible hotspot.

To set up the service controller as an iPass hotspot, you must define the iPass authentication server on the **Security > RADIUS** page.

Note: The RADIUS Reply-Message can be retrieved when using NOC authentication by using the %r placeholder in login-err-url and welcome-url, and extracting it from the answer sent by the service controller upon a NOC authentication request.

iPass login URL

This attribute has been replaced by the WISPr login URL attribute. It is still supported for backward compatibility. However, new development should use the WISPr login URL attribute.

This attribute lets you define the location of the iPass login page. The service controller automatically redirects users with iPass client software to this page.

Use the following Colubris-AVPair value string:

```
ipass-login-url=URL_of_page
```

Where:

Parameter	Description
URL_of_page	Address of the iPass login page.

ASP functions

The following ASP functions can be called from the internal pages only.

Errors

GetAuthenticationErrorMessage()

Returns a message (from message.txt) indicating the status of the last authentication request. This function is used on the default Login and Fail pages to update the user on the status of the login or logout.

RADIUS

GetMsChapV2Failed()

Returns the MS CHAP V2 error string. This function is only supported if you select MSCHAP V2 as the authentication scheme on the service controller (**Security > RADIUS** page). The RADIUS server must also support this feature. For a list of possible return values see RFC 2759.

GetRadiusNasId()

Returns the NAS ID configured for RADIUS Profile on the service controller. (See the service controller's *MSM313/MSM323 IS AP Management and Configuration Guide* for details on setting the NAS ID.) This can be used to identify the service controller that authenticated a user. For an example of how this function is used, see `GetNasAddress()`.

GetRadiusReplyMessage()

Returns the string sent by the RADIUS server when an authentication request fails. The RADIUS server must be configured to support this feature. The information contained in the returned string depends on the configuration of the RADIUS server.

GetNasAddress()

Returns the fully-qualified domain name of the service controller as is specified in the currently loaded SSL certificate.

For example, in certain instances you may want users to register for an account before they log in. To accomplish this you could modify the Login page by adding a register button. This redirects the user's browser to a registration web server where they can set up their account. (This page must be made accessible to non-authenticated users using the appropriate access list rule.)

To avoid having the user login once registration is complete, the registration web server can send the user back to the service controller using a special URL that automatically logs the user into the public access interface.

Assuming the registration server is 192.169.30.1, the register button code on the Login page might look something like this:

```
<FORM><INPUT
onclick="javascript:window.location='https://192.168.30.1/demo-php/
register.php?
NASip=<%GetNasAddress();%&NASid=<%GetRadiusNasId();%>';"
type=button value="Click Here to Register">
</FORM>
```

The NAS ID and NAS address are required when the user is redirected back to the service controller after registration. The code on the registration web page would look something like this:

```
// Registering user information in the backend database
RegisterUser($username,
$firstname,
$lastname,
$company,
$title,
$phone,
$email,
$NASid,    // identifies the service controller the user is connected to
$NASip
);

// set URL to redirect browser to
$targetURL = "location: https://"
" . $NASip . ":8090/goform/HtmlLoginRequest?
username=" . $username . "&password=" . $password;

// When done
header($targetURL);
```

The target URL is built using the NAS IP and username and password. The form name is hard-coded.

Page URLs

GetFailRetryUrl()

Returns the URL of the next internal page to display as follows:

- Returns the Fail page URL if a login or logout request is currently pending.
- Returns the Transport page URL if the user is already logged in.

This function is designed to be used in conjunction with `IsRequestPending()`.

GetLoginUrl()

Returns the URL of the Login page.

GetOriginalUrl()

Returns the URL the user tried to access before being redirected to the Login page.

GetSessionUrl()

Returns the URL of the Session page.

GetWelcomeUrl()

Returns the URL of the Welcome page.

Session status and properties

Session time

GetSessionTime()

Returns session duration for the current user in minutes and seconds in the format: mm:ss.

ConvertSessionTime(unit)

Returns session duration for the current user in the specified unit. See `ConvertMaxSession` time for details.

TruncateSessionTime(unit)

Returns session duration for the current user truncated to the specified unit. See `TruncateMaxSession` time for details.

GetSessionTimeHMS()

Returns session duration for the current user in hours, minutes and seconds in the format: hh:mm:ss.

GetSessionRemainingTime()

Returns the amount of connection time remaining for the current user session in minutes and seconds in the format: mm:ss.

GetSessionRemainingTimeHMS()

Returns the amount of connection time remaining for the current user session in hours, minutes and seconds in the format: hh:mm:ss.

ConvertSessionRemainingTime(unit)

Returns the total amount of connection time remaining for the current user in the specified unit. See ConvertMaxSession time for details.

TruncateSessionRemainingTime(unit)

Returns the total amount of connection time remaining for the current user truncated to the specified unit. See TruncateMaxSession time for details.

GetMaxSessionTime()

Returns the total amount of connection time configured for the current user session in minutes and seconds in the format: mm:ss.

GetMaxSessionTimeHMS()

Returns the total amount of connection time configured for the current user session in hours, minutes and seconds in the format: hh:mm:ss.

ConvertMaxSessionTime(unit)

Returns the total amount of connection time configured for the current user in the specified unit.

y	Years
d	Days
h	Hours
m	Minutes
s	Seconds

For example if the user account is configured for 5000 seconds, then:

- ConvertSessionTime("y") returns 0, calculated as $(5000 / (365 * 24 * 60 * 60))$.
- ConvertSessionTime("d") returns 0, calculated as $(5000 / (24 * 60 * 60))$.
- ConvertSessionTime("h") returns 1, calculated as $(5000 / (60 * 60))$.
- ConvertSessionTime("m") returns 83, calculated as $(5000 / 60)$.
- ConvertSessionTime("s") returns 5000, calculated as $(5000 / 1)$.

TruncateMaxSessionTime(unit)

Returns the total amount of connection time configured for the current user truncated to the specified unit.

y	Years
d	Days
h	Hours
m	Minutes
s	Seconds

For example if the user account is configured for 5000 seconds, then:

- TruncateSessionTime("y") returns 0.

- `TruncateSessionTime("d")` returns 0.
- `TruncateSessionTime("h")` returns 1.
- `TruncateSessionTime("m")` returns 23.
- `TruncateSessionTime("s")` returns 20.

Session input/output/totals

If you specify a value for the optional parameter **div**, then the return value is divided by **div**.

GetSessionInputPackets()

GetSessionInputOctets(div)

Returns the number of packets/octets received by the current user session.

GetSessionOutputPackets()

GetSessionOutputOctets(div)

Returns the number of packets/octets sent by the current user session.

GetSessionTotalPackets()

GetSessionTotalOctets(div)

Returns the number of packets/octets sent and received by the current user session.

GetSessionMaxTotalPackets()

GetSessionMaxTotalOctets(div)

Returns the maximum number of packets/octets that can be sent and received by the current user session.

GetSessionRemainingInputPackets()

GetSessionRemainingInputOctets(div)

Returns the remaining number of packets/octets that can be received by the current user session.

GetSessionRemainingOutputPackets()

GetSessionRemainingOutputOctets(div)

Returns the remaining number of packets/octets that can be sent by the current user session.

GetSessionRemainingTotalPackets()

GetSessionRemainingTotalOctets(div)

Returns the remaining number of packets/octets that can be sent or received by the current user session.

GetSessionMaxInputPackets()

GetSessionMaxInputOctets(div)

Returns the maximum number of packets/octets that can be received by the current user session.

GetSessionMaxOutputPackets()

GetSessionMaxOutputOctets(div)

Returns the maximum number of packets/octets that can be sent by the current user session.

Other

GetSessionStateMessage()

Returns a message (from message.txt) indicating the status of the user session.

GetUserName()

Returns the username for the current user.

IsLoggedIn()

Returns "yes" if the user is logged in. See `IsRequestPending()` for an example that shows how to use this function.

IsRequestPending()

Returns 'yes' if a login or logout request is already pending for the current user. This function is useful when a RADIUS server is slow to respond and a user repeatedly clicks the login or logout buttons. For example, consider the following code which could be used to modify the Fail page to address this problem.

```
function loading() //called when the fail page is first loaded
{
    if ("<% IsLoggedIn(); %>" == "yes") //logout is pending, so refresh page
        refresh();
    else
    {
        // user is already logged out or a login is currently pending
        // (i.e., user clicked login button twice)
        if ("<% IsRequestPending(); %>" == "yes")
            setTimeout('refresh()',3000);
        else //no login or logout is pending and user is logged out
            document.form1.close.value = "Close window"; //change button label
    }
}

function refresh() // refresh the Fail page
{document.location="<%GetFailRetryUrl();%>"; }
```

SetSessionRefreshInterval(sec)

Specifies the refresh interval for the Session page in seconds.

Session quotas

These functions let you retrieve the quota limits that are set for the current user session. If any of these limits are reached, the user is logged out. For details see [“Quotas” on page 52](#).

If you specify a value for the optional parameter **div**, then the return value is the number of octets divided by **div**.

- Packets values are returned as a decimal string (10 characters) representing a 32-bit unsigned integer.
- Octet values are returned as a decimal string (20 characters) representing a 64-bit unsigned integer.

GetSessionRemainingInputPackets()**GetSessionRemainingInputOctets(div)**

Returns the number of incoming packets/octets the current user session can still receive.

GetSessionRemainingOutputPackets()**GetSessionRemainingOutputOctets(div)**

Returns the maximum number of outgoing packets/octets the current user session can still send.

GetMaxSessionInputPackets()**GetMaxSessionInputOctets(div)**

Returns the maximum number of incoming packets/octets the current user session can receive.

Returns the maximum number of incoming octets the current user session can receive.

GetMaxSessionOutputPackets()

GetMaxSessionOutputOctets(div)

Returns the maximum number of outgoing packets/octets the current user session can send.

iPass support

iPassGetLoginUrl()

Returns the iPass Login URL.

iPassGetAbortLoginUrl()

Returns the iPass Abort Login URL.

iPassGetLogoffUrl()

Returns the iPass Logout URL.

iPassGetRedirectResponseCode()

Checks if the iPass authentication server is reachable and enabled. Returns one of the following values:

0	Authentication server is reachable and enabled.
105	The authentication server could not be reached or is unavailable.
255	The authentication server could not be reached due to an error on the service controller (Internet port not up, for example).

iPassGetAccessProcedure()

Returns the access procedure supported by the service controller. The service controller supports procedure version 1.0.

iPassGetLocationName()

Returns the location name defined on the **Public access > Access control** page.

iPassGetAccessLocation()

Returns a value which can be used to determine the access point a user is connected to. This is useful when you are using one or more APs in addition to the service controller.

- If a user logs into an AP, this function returns the MAC address of the APs downstream port.
- If a user logs into the service controller, this function returns the MAC address of the service controller's LAN port.

iPassGetLoginResponseCode()

Returns one of the following values when a user attempts to login to iPass:

50	Login was successful.
100	Login failed. Access was rejected.
102	Login failed. Authentication server error or timeout.
201	Authentication is pending.
255	The authentication server could not be reached due to an error on the service controller (Internet port not up, for example).

iPassGetLoginResponseCode()

Returns one of the following values when a user attempts to logout from iPass:

150	Logout was successful.
255	The authentication server could not be reached due to an error on the service controller (Internet port not up, for example).

4

NOC authentication

Contents

Main benefits - - - - -	88
How it works - - - - -	88
Activating a remote login page with NOC authentication - - - - -	89
Addressing security concerns - - - - -	91
Setting up the certificates - - - - -	92
Authenticating users - - - - -	93
Simple NOC authentication example - - - - -	97
Forcing user logouts - - - - -	99

Main benefits

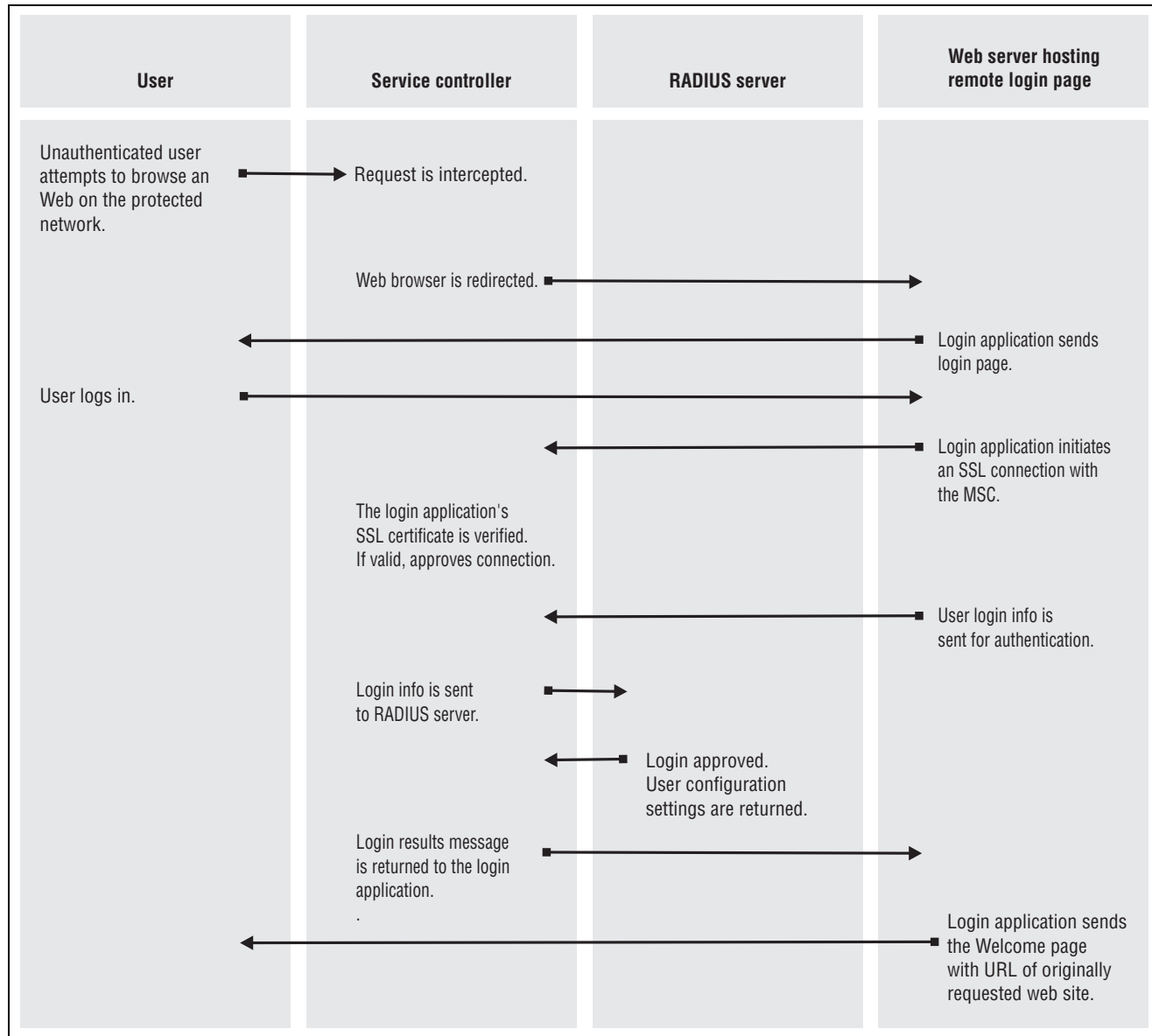
Using a remote login page with NOC (network operations center) authentication provides you with the following benefits:

- The login page is completely customizable. You are not bound by the limits imposed by loading a login page onto the service controller.
- Users can login to the public access interface without exposing their web browsers to the SSL certificate on the service controller. This eliminates warning messages caused by having an SSL certificate on the service controller that is not signed by a well-known certificate authority.
- If you want to support secure login with SSL, but have multiple service controllers, using a remote login page means you only need to purchase a single SSL certificate signed by a well-known certificate authority, instead of one for each access point.

How it works

The NOC authentication feature provides a secure way of authenticating public access users, with strong mutual authentication between the login application on the web server hosting the remote login page and the service controller used for authenticating user logins. This occurs via the two Colubris-AVPair value strings (**ssl-noc-certificate** and **ssl-noc-ca-certificate**), which define the locations of two certificates. These certificates enable the service controller to validate that the user login information does indeed come from a trusted application. For example, from a login application on the web server.

The following diagram shows the sequence of events for a typical user session when using the NOC-based authentication feature.



Activating a remote login page with NOC authentication

To activate a remote login page, you must define several service controller attributes. These attributes can be defined in the RADIUS account for the service controller (if you are using a RADIUS server) or they can be locally configured.

The following table summarizes the Colubris-AVPair value strings for the remote login page with NOC authentication.

Item	Colubris-AVPair value string
External login	login-url= <i>URL_of_the_page</i> [<i>placeholder</i>] URL of the remote login page. Access to the web server hosting this page must be granted to all unauthenticated users. Do this with an appropriate access list definition.
NOC certificate	ssl-noc-certificate= <i>URL_of_the_Certificate</i> Certificate issued to the application on the web server that sends user info to the service controller for authentication.
NOC CA certificate	ssl-noc-ca-certificate= <i>URL_of_the_certificate</i> Certificate of the certificate authority (CA) that issued the NOC certificate.
Custom SSL certificate	ssl-certificate= <i>URL</i> Custom certificate installed on the service controller.

The following placeholders can be added to the login-url string.

Placeholder	Description
%c	Returns the IP address of the user's computer.
%d	Returns the WISPr location-ID. Supported for login-url only.
%e	Returns the WISPr location-Name. Supported for login-url only.
%l	Returns the URL on the service controller where user login information should be posted for authentication. By default, this value is URL encoded. (To enable/disable URL encoding, set the value of url-encode in the <ACCESS-CONTROLLER> section in the configuration file.)
%n	Returns the NAS ID assigned to the service controller. By default, this is the unit's serial number. Not supported in local mode.
%s	Returns the RADIUS login name assigned to the service controller. By default, this is the unit's serial number. Not supported in local mode.
%o	Returns the original URL requested by the user. By default, this value is URL encoded. By default, this value is URL encoded. (To enable/disable URL encoding, set the value of url-encode in the <ACCESS-CONTROLLER> section in the configuration file.)
%i	Returns the domain name assigned to the service controller's Internet port.
%p	Returns the port number on the service controller where user login information should be posted to for authentication.
%a	Returns the IP address of the service controller's interface that is sending the authentication request.

%E	When the location-aware feature is enabled, returns the ESSID of the wireless access point the user is associated with.
%P	When the location-aware feature is enabled, returns the wireless mode ("ieee802.11a", "ieee802.11b", "ieee802.11g") the user is using to communicate with the access point.
%G	When the location-aware feature is enabled, returns the group name of the wireless access point the user is associated with.
%C	When the location-aware feature is enabled, returns the Called-station-id content for the wireless access point the user is associated with.
%r	Returns the string sent by the RADIUS server when an authentication request fails. The RADIUS server must be configured to support this feature. The information contained in the returned string depends on the configuration of the RADIUS server.
%m	Returns the MAC address of the wireless/wired client station that is being authenticated.
%v	Returns the VLAN assigned to the client station at the service controller's ingress (LAN port).

Addressing security concerns

It is important that the connection between the login application and the service controller be secure to protect the exchange of user authentication traffic. The following strategy provides for complete connection security.

Securing the remote login page

HTTPS can be used on the web server to secure the login page. To avoid warning messages on the user's browser, the SSL certificate installed on the web server should be signed by a well-known CA.

Authenticating with the login application

The connection between the login application and the service controller is secured using SSL. When establishing the SSL connection with the service controller, the login application must supply its SSL certificate. In a standard SSL setup, the service controller uses the CA for this certificate to validate the certificate's identity and authenticate the login application.

However, the service controller does not want to accept SSL connections from *just any* remote entity with a valid certificate. Rather, it only wants to accept connections from a specific entity: the login application.

To uniquely identify the login application, the *ssl-noc-certificate* attribute is defined in the RADIUS profile for the service controller. This attribute contains the URL of the login application's SSL certificate. When the login application presents its SSL certificate, the service controller retrieves *ssl-noc-certificate* and checks to make sure that they match.

For further authentication, a second attribute, *ssl-noc-ca-certificate*, is defined in the RADIUS profile for the service controller. This attribute contains the URL of the public key of the certificate authority (CA) that signed the login application's SSL certificate. The service controller uses the public key to determine if the login application's SSL certificate can be trusted.

Authenticating the service controller

To identify itself, the service controller uses the SSL certificate configured on the **Security > Certificates** page or via the *ssl-certificate* attribute.

For added security, the login application could also check that this SSL certificate has been signed by the certificate authority for which the login application has the public key certificate. The default certificate installed on the service controller, is not signed by a well-known CA and cannot be used for this purpose. Instead, a new certificate must be installed on the service controller. This certificate could be signed by a well-known certificate authority or your own CA.

NOC authentication list

Additional security is provided via the NOC authentication list on the **Public access > Access control** page. You use this list to define the set of remote IP addresses that the service controller accepts authentication requests from. If a request is received from an address not in this list, it is discarded.

Setting up the certificates

This section presents an overview of the certificates you need to install to secure communication between the remote login page and the service controller. For detailed discussion of the issues, see [“Addressing security concerns” on page 91](#).

Install certificates on the web server

Install an SSL certificate and its matching CA certificate into a folder on the web server hosting the remote login page. The login application and the service controller access the certificates from this location.

The SSL certificate is used by the login application to secure communications with the service controller.

Define attributes

Add the following attributes to the **Configured attributes** table on the **Public access > Attributes** page. (You can also define these attributes in the RADIUS profile for the service controller if you are using a RADIUS server.) This enables it to retrieve the SSL and CA certificates from the web server:

<code>ssl-noc-certificate=URL_of_the_Certificate</code>
Certificate issued to the application on the web server that sends user info to the service controller for authentication.
<code>ssl-noc-ca-certificate=URL_of_the_certificate</code>
Certificate of the certificate authority (CA) that issued the NOC certificate.
<code>ssl-certificate=URL</code>
Custom certificate installed on the service controller.

Install a certificate on service controller

Note: This step is optional, but recommended.

Install an SSL certificate on the service controller to replace its default SSL certificate. This certificate is used to secure communications between the service controller and the login application on the web server.

If you do not change the default certificate on the service controller, the login application may not be able to validate the service controller's certificate when establishing the SSL connection. The reason for this is because the default certificate is self-signed and is not trusted by any well-known CA.

This can be done by adding an additional attribute to the **Configured attributes** table on the **Public access > Attributes** page. (You can also define this attribute in the RADIUS profile for the service controller if you are using a RADIUS server.).

`ssl-certificate=URL`

Authenticating users

After a user has supplied login information on the remote login page, the login application must submit an authentication request containing the user's login name, password, and IP address to the service controller by establishing an SSL session to the following URL:

`https://service_controller_ip:8090/goform/HtmlNocLoginRequest
?username=username&password=password&ipaddr=user_ip`

Where:

Parameter	Description
<code>service_controller_ip</code>	<p>Defines the IP address of the service controller or you could use a domain name if you have defined one using the hosts file on the web server. (By default, the secure web server on the service controller operates on port 8090. This can be changed on the Management > Management Tool page if required.)</p> <p>The service controller requires that the contents of the Host HTTP header match the actual domain name/IP address and port the service controller is operating on:</p> <p>Host: service controller_domain_name:secure_web_server_port_number or Host: service controller_IP_address:secure_web_server_port_number</p> <p>This is usually the case unless the service controller is behind a device that provides network address translation (NAT). In this situation, the login application must manually forge the Host HTTP header. The easiest way to do this is to define <code>login-url</code> with the <code>%i</code> and <code>%p</code> placeholders. This returns the domain name of the service controller and the port number of its secure web server. The login application can then construct the appropriate Host HTTP header.</p>
<code>username</code>	Username supplied by the user.
<code>password</code>	Password supplied by the user.
<code>user_ip</code>	IP address of the user's computer.

Example 1

Assume that the service controller is not behind a NATing device, and that its IP address is 192.168.4.2. The subject DN in its SSL certificates is `www.noc-cn3.com`.

The Host HTTP header should be set to one of:

- Host: `www.noc-cn3.com:8090`
- Host: `192.168.4.2:8090`

Example 2

Assume that the service controller is behind a NATting device. The device has the address 192.168.30.173, and the service controller has the address 192.168.4.2. A NAT mapping is defined on the NATting device that redirects traffic received on port 8090 to 192.168.4.2:8090.

The login application must send its requests to 192.168.30.173, which results in a HTTP Host header that contains one of the following:

- Host: `natting.device.com:8090`
- Host: `192.168.30.173:8090`

When this request is forwarded to the service controller, it is rejected. To solve the problem, the login application must forge the host HTTP header. This is easily done by plugging in the values returned by the %i, %a, and %p placeholders. For example:

Host: %i:%p

or

Host: %a,%p

The service controller sends the username and password to the RADIUS server to authenticate the user. If authentication is successful, the user's IP address is used to grant wireless network access to the user's computer.

The service controller returns a positive or negative answer for the user login, along with the relevant URLs that may be needed by the login application in order to redirect the user to either a Welcome page or a Login error page located on the web server. This information is returned as standard HTML. The login application must parse this information to retrieve the response. All possible responses are described in the following section.

Returned values

The following examples show the information returned for various authentication conditions.

NOC authentication mode is not enabled

```
<HTML>
NOC_INFO_STATUS=NOC_STATUS_DISABLED
</HTML>
```

The service controller did not receive the login application's SSL certificate

The login application did not send its certificate. Therefore, the request was rejected.

```
<HTML>
NOC_INFO_STATUS=NOC_STATUS_FAILURE
NOC_INFO_INT_ERR_MESSAGE=NOC_CANNOT_GET_PEER_CERT
</HTML>
```

Certificate mismatch

The login application sent an SSL certificate that does not match the one defined by ssl-noc-certificate in the RADIUS profile for the service controller.

```
<HTML>
NOC_INFO_STATUS=NOC_STATUS_FAILURE
NOC_INFO_INT_ERR_MESSAGE=NOC_CANNOT_GET_PEER_CERT
</HTML>
```

Certificate not valid yet

The login application sent an SSL certificate that matches the one defined by ssl-noc-certificate in the RADIUS profile for the service controller. However, the certificate that was sent is not yet valid.

```
<HTML>
NOC_INFO_STATUS=NOC_STATUS_FAILURE
NOC_INFO_INT_ERR_MESSAGE=NOC_CERT_NOT_YET_VALID
</HTML>
```

Certificate not valid anymore

The login application sent an SSL certificate that matches the one defined by ssl-noc-certificate in the RADIUS profile for the service controller. However, the certificate that was sent is not valid anymore.

```
<HTML>
NOC_INFO_STATUS=NOC_STATUS_FAILURE
NOC_INFO_INT_ERR_MESSAGE=NOC_CERT_EXPIRED
</HTML>
```

Certificate not signed by proper CA

The login application sent a valid SSL certificate that matches the one defined by ssl-noc-certificate in the RADIUS profile for the service controller. However, the certificate is not signed by the CA defined by noc-ca-certificate in the RADIUS profile for the service controller.

```
<HTML>
NOC_INFO_STATUS=NOC_STATUS_FAILURE
NOC_INFO_INT_ERR_MESSAGE=NOC_CERT_NOT_SIGNED_BY_AUTHORIZED_CA
</HTML>
```

Missing username and/or password

The user's username or password was not supplied.

```
<HTML>
NOC_INFO_STATUS=NOC_STATUS_FAILURE
NOC_INFO_INT_ERR_MESSAGE=NOC_MISSING_USERNAME_OR_PASSWORD
</HTML>
```

The specified IP address is already logged in

```
<HTML>
NOC_INFO_STATUS=NOC_STATUS_LOGGED_IN
</HTML>
```

Authentication was successful

```
<HTML>
NOC_INFO_STATUS=NOC_STATUS_SUCCESS
NOC_INFO_WELCOME_URL=<welcome url>
NOC_INFO_SESSION_URL=<session url>
</HTML>
```

Authentication failed

```
<HTML>
NOC_INFO_STATUS=NOC_STATUS_FAILURE
NOC_INFO_ERR_MESSAGE=<error message>
NOC_INFO_LOGIN_ERR_URL =<login error url>
</HTML>
```

Logout succeeded

```
<HTML>
NOC_INFO_STATUS=NOC_STATUS_SUCCESS
</HTML>
```

Logout failed

```
<HTML>
NOC_INFO_STATUS=NOC_STATUS_FAILURE
NOC_INFO_INT_ERR_MESSAGE=<error message>
</HTML>
```

Examples of returned HTML code

The following examples show the actual HTML code returned file for various authentication conditions.

User was successfully authenticated by the RADIUS server

```
<HTML>
status=success
welcome-url=https://206.162.167.226:8888/cebit-php/welcome.php?site=www.noc-
service controller.com&user=user00&wantedurl=&nasipaddress=&nasid=L003-00069
session-url=http://192.168.1.1:8080/session.asp
</HTML>
```

User's IP address is already in use by an active session

```
<HTML>
status=already-logged-in
</HTML>
```

User authentication was refused by the RADIUS server

This could be due to an unknown username, or invalid username or password.

```
<HTML>
status=failure
external-err-msg=Your login was refused.
login-err-url=https://206.162.167.226:8888/cebit-php/login-
error.php?site=eperie-cn3000&user=user12&nasipaddress=
</HTML>
```

User could not be authenticated

The service controller could not contact a RADIUS server.

```
<HTML>
status=failure
external-err-msg=You cannot be logged in at this time. Please try again
later.
login-err-url=https://206.162.167.226:8888/cebit-php/login-
error.php?site=eperie
-cn3000&user=user12&nasipaddress=
</HTML>
```

Simple NOC authentication example

See [“Sample files” on page 65](#) for information about how to get the `Internal_Pages.zip` file.

This is a simple example showing how to use the NOC authentication feature.

1. Create the following folder on your web sever: **newlogin**.

2. Copy the following files from **Internal_Pages.zip** place them in the **newlogin** folder.

- login.html
- transport.html
- session.html
- fail.html
- logo.gif

3. Customize **login.html** to accept username and password information from users and then send it to the service controller. You could use code similar to the following PHP example to send login information back to the service controller for authentication:

```
https://ipaddress of CNx:8090/goform/HtmlNocLoginRequest  
?username=username&password=password&ipaddress=user_ip
```

The variable `loginurl` contains the URL on the service controller where user information is sent for authentication.

4. Start the management tool.

5. Click **Public access > Access control**.

6. Enable the **NOC authentication** feature.

7. Add the IP address of the web server to the **Allowed Addresses** box.

8. Under **Active interfaces** make sure that the interface on which the request will be received is enabled.

9. Click **Save**.

10. Add the following entries to the **Configured attributes** table on the **Public access > Attributes** page. (You can also define these attributes in the RADIUS profile for the service controller if you are using a RADIUS server.)

```
login-url= URL_of_page_on_remote_server  
access-list=loginserver,ACCEPT,tcp,web_server_IP_address,443  
ssl-noc-certificate= URL_of_the_certificate  
ssl-noc-ca-certificate= URL_of_the_certificate  
transport-page=web_server_URL/newlogin/transport.html  
session-page=web_server_URL/newlogin/session.html  
fail-page=web_server_URL/newlogin/fail.html  
logo=web server URL/newlogin/logo.gif  
use-access-list=loginserver
```

Forcing user logouts

Users can be logged out by calling the following URL:

```
https://service_controller_ip:8090/goform/HtmlNocLogoutRequest  
?ipaddress=user_ip
```

Note: This request must come from the login application (or another other application that is using the same SSL certificate).

The service controller returns a positive or negative answer for the user logout as standard HTML. The login application must parse this information to retrieve the response.

Logout success

```
<HTML>  
NOC_INFO_STATUS=NOC_STATUS_SUCCESS  
</HTML>
```

Logout failure

```
<HTML>  
NOC_INFO_STATUS=NOC_STATUS_FAILURE  
NOC_INFO_INT_ERR_MESSAGE=<error message>  
</HTML>
```

Note: These definitions are contained in noc.h.

Technology for better business outcomes

To learn more, visit www.hp.com/networking/

© Copyright 2010 Hewlett-Packard Development Company, L.P. The information contained herein is subject to change without notice. The only warranties for HP products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. HP will not be liable for technical or editorial errors or omissions contained herein.



July 2010

Manual Part Number
5998-0449