RSA Authentication Agent 5.3 for Web for Apache Web Server on Red Hat Linux 4.0 Installation and Configuration Guide

The Security Division of EMC

Contact Information

See the RSA corporate web site for regional Customer Support telephone and fax numbers: **www.rsa.com**

Trademarks

RSA and the RSA logo are registered trademarks of RSA Security Inc. in the United States and/or other countries. For the most up-to-date listing of RSA trademarks, see **www.rsasecurity.com/legal/trademarks_list.pdf**. EMC is a registered trademark of EMC Corporation. All other goods and/or services mentioned are trademarks of their respective companies.

License agreement

This software and the associated documentation are proprietary and confidential to RSA, are furnished under license, and may be used and copied only in accordance with the terms of such license and with the inclusion of the copyright notice below. This software and the documentation, and any copies thereof, may not be provided or otherwise made available to any other person.

No title to or ownership of the software or documentation or any intellectual property rights thereto is hereby transferred. Any unauthorized use or reproduction of this software and the documentation may be subject to civil and/or criminal liability.

This software is subject to change without notice and should not be construed as a commitment by RSA.

Note on encryption technologies

This product may contain encryption technology. Many countries prohibit or restrict the use, import, or export of encryption technologies, and current use, import, and export regulations should be followed when using, importing or exporting this product.

Distribution

Limit distribution of this document to trusted personnel.

First printing: February 2007

Contents

Preface

About This Guide

This guide describes how to install and configure RSA Authentication Agent 5.3 for Web for Apache Web Server on Red Hat Linux 4.0. It is intended for administrators and other trusted personnel. Do not make this guide available to the general user population.

RSA Authentication Agent 5.3 for Web for Apache Web Server on Red Hat Linux 4.0 Documentation

For more information about the Web Agent, see the following documentation:

Readme. Provides workarounds for known issues. The latest version of the *Readme* is available from RSA SecurCare Online: **https://knowledge.rsasecurity.com**.

Installation and Configuration Guide. Describes detailed procedures on how to install and configure the Web Agent.

Developer's Guide. Provides information about developing custom programs using the Web Agent application programming interfaces (APIs). Includes an overview of the APIs and Javadoc for Java APIs.

Related Documentation

RSA Secured Partner Solutions directory. RSA has worked with a number of manufacturers to qualify products that work with RSA products. Qualified third-party products include virtual private network (VPN) and remote access servers (RAS), routers, web servers, and many more. To access the directory, including implementation guides and other information, go to **http://www.rsasecured.com**.

Getting Support and Service

RSA SecurCare Online	https://knowledge.rsasecurity.com
Customer Support Information	www.rsa.com/support
RSA Secured Partner Solutions Directory	www.rsasecured.com

RSA SecurCare Online offers a Knowledgebase that contains answers to common questions and solutions to known problems. It also offers information on new releases, important technical news, and software downloads.

The RSA Secured Partner Solutions Directory provides information about third-party hardware and software products that have been certified to work with RSA products. The directory includes Implementation Guides with step-by-step instructions and other information about interoperation of RSA products with these third-party products.

Before You Call Customer Support

Make sure you have direct access to the computer running the Web Agent software.

Please have the following information available when you call:

- ❑ Your RSA Customer/License ID.
- ❑ RSA Authentication Agent 5.3 for Web software version number.
- ❑ The make and model of the machine on which the problem occurs.
- ❑ The name and version of the operating system under which the problem occurs.

1 Overview

RSA Authentication Agent 5.3 for Web for Apache on Red Hat Linux enables you to protect selected web pages with RSA SecurID.

The Web Agent software, residing on a web server, intercepts all user requests for protected web pages. When a user attempts to access a protected URL, the Web Agent requests a user name and passcode and submits them to the RSA Authentication Manager for authentication. If the authentication is successful, the Web Agent stores the information in a cookie in the user's browser. As long as the cookie remains valid, the user is granted access to protected web pages.

Note: Web access authentication protects **http** and **https** URLs. Due to security risks associated with **ftp** file transfers across the Internet, web access authentication does **not** protect files on an **ftp** server. In addition, it does not support gopher, news, ftp, wais, or telnet protocols.

Security Features

The following table describes some of the security features that the Web Agent provides.

Security Feature	Description
Two-factor authentication	To gain access to a protected web page, users enter their user name and a valid RSA SecurID passcode, which consists of • A secret, memorized personal identification number (PIN). • The tokencode currently displayed on an RSA SecurID token.
Support for SSL	This feature establishes a private communication channel between the user and the web server, which prevents third parties from eavesdropping.
Tamper-evident cookies	The cookie that the Web Agent distributes to a user's browser contains • Information indicating the user has successfully authenticated. • An encrypted data string. The encrypted string is used to detect whether someone has altered the cookie contents. Any tampering is logged in the system Web Agent audit files. In addition, cookies have set expiration times to help protect the URL if users walk away from their computers.

Security Feature	Description
Name locking	This feature prevents an intruder from detecting a legitimate logon attempt, intercepting the user's passcode, and using the passcode to log on.
Auditing	The Web Agent records • Access attempts • Status of connections • Any instances of cookie tampering in audit logs on the Agent Host

Types of User Access

You can configure the Web Agent to

- Protect URLs on the local server on which the Web Agent is installed
- Allow users access to URLs on other servers that the Agent protects in the same domain or in multiple domains

For each access type, the Web Agent distributes a cookie to the user's browser so that the user does not have to reauthenticate to each protected resource during a browser session.

The following table describes the different types of user access.

Access Type	Cookies Distributed to User's Browser Upon Successful Authentication	Protected URLs the User Can Access	Configuration Instructions
Local	Local Cookie	Protected URLs on the local web server	"Using the Setup Menu" on page 16
Domain	Domain Cookie	Protected URLs on all web servers in the domain	"Using the Domain and Multi-Domain Menu" on page 18
Multi-Domain	Domain cookies from each domain	Protected URLs on web servers in multiple domains	"Using the Domain and Multi-Domain Menu" on page 18

Getting Started

Task	Chapter
Install or migrate the Web Agent	"Installing RSA Authentication Agent 5.3 for Web for Apache Web Server"
• Configure Web access authentication cookies • Protect resources • Configure advanced settings • Set up multi-server and multi-domain authentication	"Configuring Web Access Authentication Settings"
Customize templates	"Customizing Templates and Message Strings"
Troubleshoot issues	"Troubleshooting"

2 Installing RSA Authentication Agent 5.3 for Web for Apache Web Server

Installation Requirements

The Web Agent is supported on Apache Web Server 2.0.59 on Red Hat Enterprise Linux 4.0 AS/ES.

Note: Make sure the web server computer is located in a secure area so that only trusted personnel have access.

Client System Requirements

Users accessing protected web pages must have one of the following web browsers installed on their computers:

- Microsoft Internet Explorer 6.0 with SP2 or Microsoft Internet Explorer 7.0
- Mozilla Firefox 2.0

RSA SecurID Web authentication through wireless access protocol (WAP) requires the following WAP 1.1 and 1.2.1 specifications:

- Caching of cookies
- Wireless Markup language (WML) Document Type Definition (DTD) version 1.1

RSA SecurID users must enable the cookie acceptance feature in their browsers. They must also use web browsers that support FORMs and Persistent Client State HTTP Cookies.

Additional Requirements

The Web Agent works in conjunction with RSA Authentication Manager 5.2 or later. The Web Agent administrator must be familiar with the Authentication Manager system and its features.

In addition, make sure the Authentication Manager administrator has registered the RSA SecurID users in the Authentication Manager database and distributed tokens to these users.

Pre-Installation Tasks

Before you install the Web Agent, configure your web server and RSA Authentication Manager environments to work with the Web Agent.

Enabling the Apache Web Server to Work with the Web Agent

The Apache server binaries must have module **mod_so** and either **worker** or **prefork** enabled.

If your Apache web server is already installed and configured, use the following procedure to verify whether the modules are enabled.

1. Change to the Apache web server installation directory. For example:

   ```
   cd /usr/local/apache/bin
   ```

2. Type

   ```
   ./httpd -l
   ```

3. Look for **mod_so.c** and either **worker.c** or **prefork.c** in the output.

   If the correct modules are not listed, you must recompile the Apache web server binaries with the modules enabled. For instructions, see your Apache web server documentation.

Proceed to the following section, "Adding the Web Server to the RSA Authentication Manager Environment."

Adding the Web Server to the RSA Authentication Manager Environment

To add the web server to the RSA Authentication Manager environment:

1. Add the web server to the Authentication Manager as an Open Agent Host. For assistance, contact your Authentication Manager Administrator.
2. Get the **sdconf.rec** file from your Authentication Manager administrator. The Web Agent software uses the **sdconf.rec** file to locate the Authentication Manager on the network.
3. Put the **sdconf.rec** file in a directory, such as /var/ace, that is accessible to the web server. The user who owns the web server must have write permissions to the directory. By default, this user is called "nobody."

   Note: If you install multiple Agents on the server, create different directories in which to store their respective **sdconf.rec** files.

4. Add a VAR_ACE environment variable to your web server configuration file so it is set whenever the web server runs. This environment variable identifies the location of the **sdconf.rec** file. For example:

   ```
   setenv VAR_ACE /var/ace
   ```

Installing the Web Agent Software

The Web Agent software requires approximately 10 MB of free disk space.

Note: RSA recommends that you stop the web server before installing the Web Agent.

To install the RSA Authentication Agent 5.3 for Web software:

1. Log on to an account that has write permissions to the web server root directory.
2. Change to the directory you created when you downloaded the software, and extract the software files.
3. To run the installation script, type:

   ```
   ./install
   ```

   When prompted to specify where you obtained your Web Agent product, if you obtained it from somewhere other than the countries listed, type **n**. Otherwise, press ENTER.
4. Accept the License Terms and Conditions by typing **A**.
5. If the **sdconf.rec** path is correct, press ENTER.

   The pathname entered for the VAR_ACE environment variable is displayed. If the pathname is not correct, it may not be correctly defined in the variable. For information about this setting, see the preceding section, "Adding the Web Server to the RSA Authentication Manager Environment."
6. When you are prompted for the path to the Apache servername directory, specify the complete path to the web server, and press ENTER.

The configuration script starts automatically. For directions, see "Configuring the Software" on page 15.

If you have an RSA ACE/Agent 5.2 for Web installation on an Apache 1.3 web server, and you want to use the same configuration settings when you install the RSA Authentication Agent 5.3 for Web on an Apache 2.0.59 web server, perform the following procedure.

To migrate from RSA ACE/Agent 5.2 for Web:

1. Install the Apache 2.0.59 server according to your Apache documentation. Make sure the server meets all of the requirements described in "Installation Requirements" on page 11.
2. On the Apache 2.0.59 server, install the RSA Authentication Agent 5.3 for Web.
3. From the RSA ACE/Agent 5.2 for Web installation on the Apache 1.3 server, copy the **rsawebagent/RSAWebAgent.INI** file, and migrate it to the **rsawebagent** directory on the RSA Authentication Agent 5.3 for Web installation on the Apache 2.0.59 server.

4. If you do not have customized templates, go to step 5. Otherwise, do one of the following:
   - If your RSA ACE/Agent 5.2 for Web HTML templates have customized text, graphics, or buttons, copy the templates to a directory outside of the RSA Authentication Agent 5.3 for Web **rsawebagent/templates** directory.
   - If your RSA ACE/Agent 5.2 for Web HTML templates are customized for another language, migrate the entire **rsawebagent/templates/nls/*language code*** directory to the RSA Authentication Agent 5.3 for Web installation.

   Note: RSA has updated the appearance of the templates for the RSA Authentication Agent 5.3 for Web release.

5. On the RSA Authentication Agent 5.3 for Web installation, run the Web Agent configuration script, and update the **Template** path in the Setup menu.

   Note: This step is required even if you do not have customized templates.

   - If you do not have customized templates or you are using foreign language templates, point to the RSA Authentication Agent 5.3 for Web **rsawebagent/templates** directory.
   - If you have customized templates, point to the directory in which you stored your templates.

   For directions, see "Changing Configuration Settings" on page 20 and "Using the Setup Menu" on page 16.

Uninstalling the Web Agent

Note: RSA recommends that you stop your web server before uninstalling the Web Agent.

To uninstall the Web Agent:

1. Change to the web server directory.
2. Run the uninstallation script from the web server directory by typing

```
./rsawebagent/uninstall
```

Next Steps

- For administration tasks such as managing URLs and changing configuration settings, see Chapter 3, "Configuring Web Access Authentication Settings."
- To customize the HTML and WML pages included with this Web Agent product, see Chapter 4, "Customizing Templates and Message Strings."

3 Configuring Web Access Authentication Settings

You administer the web access authentication settings of your web servers through a utility. You can quickly add, remove, and view URLs from the protected resource list without having to directly access all of the configuration settings.

With the utility, you can

- Configure web access authentication cookies
- Protect entire sites, individual directories, or individual files
- Configure advanced settings
- Set up multi-server and multi-domain authentication

When you make changes to the web access authentication properties of a virtual server, a directory, or a file, you must restart the web server.

Important: By default, the Web Agent sets the ownership and permission to all the files and directories it uses. Changing these permissions or ownership properties could create a security hole in the system.

Configuring the Software

The initial configuration sets default attribute values in the Web Agent configuration file. Once this configuration is complete, run the configuration script again if you want to make changes to individual virtual servers set up on this web server. For more information, see "Changing Configuration Settings" on page 20.

The configuration program is grouped in the following menus:

Setup Menu. Used to configure how the Agent interacts with the browser. Includes:

- Adjusting cookie validity time
- Changing the SSL port number
- Changing the WebID URL
- Changing the location of the templates

Configuration Menu. Used to configure access to protected URLs. Includes:

- Redirecting URLs to secure ports
- Using separate pages for user name and passcode
- Using the name locking feature

Domain and Multi-Domain Menu. Used to configure which domain an authentication cookie is valid for and generate a new domain secret for use on other Web Agents.

Using the Setup Menu

The Setup menu displays automatically following a successful installation.

To accept the defaults, press ENTER. Otherwise, type the line number of the option you want to change.

The following table describes the options.

Line Option	Description
1 Idle Cookie Expiration	Set the amount of time, in minutes, that an idle cookie is valid. When the cookie expires, the user must reauthenticate. Setting a value that is greater than the Cookie Expiration value deactivates this feature.
2 Cookie Expiration	Set the length of time, in minutes, that an active cookie is valid. When the cookie expires, the user must reauthenticate to get a new cookie.
3 SSL Port Number	Type in the SSL port number to be used for secure data transfer.
4 WebID URL	Accept the default name, unless you have an existing URL with the same name.
5 HTML/WML Templates	Accept the default. After the initial installation and configuration, you may customize the templates. Once you do so, run the configuration script again to designate the new location of your customized templates.

Using the Configuration Menu

The Configuration menu displays automatically after you complete the Setup menu.

To accept the defaults, press ENTER. Otherwise, type the line number of the option you want to change.

The following table describes the options.

Line Option	Description
1 Agent Protection	Accept the default. **Important**: Disable the Web Agent only when it is absolutely necessary to temporarily halt protection of all URLs on this web server for troubleshooting purposes. When the Web Agent is disabled, your data is unprotected.
2 Name Locking	The Web Agent locks the user's name while waiting for the passcode so the name cannot be used elsewhere during the authentication process.

Line Option	Description
3 Separate Pages	The Web Agent uses separate HTML or WML pages to request the user's name and passcode. If you disable this feature, the user name and passcode are sent across the Internet together.
4 Require SSL Connection	The Web Agent connects to protected URLs through an SSL port. If you disable this feature, data transmitted over the Internet is unprotected, meaning cookies can be seen in plain text. **Note**: If you do not have an SSL connection, you must disable this feature.
5 Redirect	When a user attempts to access a protected URL through an unprotected page, the Web Agent redirects the user to an authentication page. If you disable this feature, the user receives an RSA message and a link to a secure connection. **Note:** This option does not appear if you disable option 4 (Require SSL Connection).
6 Caching Pages	The Web Agent prevents the browser from caching protected URLs on the local PC. If you disable this feature, protected URLs may be cached on the local hard drive.
7 Auto Submit	After the user enters authentication information on the Web page, the Agent automatically presents the next window so the user does not have to click **CONTINUE**.
8 JavaScript Pop-up	The Web Agent allows the use of JavaScript Pop-up Windows for web pages that use frames. By default, this feature is disabled.
9 Ignore Browser Address	By default, this feature is disabled so that the Web Agent uses the browser IP address to sign the cookie. However if there is a proxy or a firewall between the browser and the Agent, the IP address used may be the same. If you have web sites that are accessed through load balanced proxy servers, meaning that the browser IP addresses may change, you may want to enable this feature. Otherwise, the user may have to authenticate quite frequently.
10 Current Domain Access	Once a user is authenticated, the user can access URLs on any of the web servers in the current protected domain. If you disable this feature, the user is asked to authenticate each time a protected URL is accessed on a different web server.

Line Option	Description
11 Multi-Domain Access	Once a user is authenticated, the user can access URLs on any web server in the multi-domain list. If you disable this feature, the user is asked to authenticate each time a protected URL is accessed on a web server that is outside the current domain.

Using the Domain and Multi-Domain Menu

If you enabled option number 10 (Current Domain Access) or 11 (Multi-Domain Access) in the Configuration menu, the Domain and Multi-Domain Configuration menu displays automatically.

The following table describes the Domain and Multi-Domain Configuration menu options.

Line Option	Description
1 Generate Domain Secret	A domain secret was automatically generated when you installed the Web Agent. Use this option to generate a new domain secret.
2 Generate and Export Domain Secret	If you have multiple web servers on which users will be able to access protected URLs, each web server within the domain must have the same domain secret. Use this option to generate and export the domain secret to a file so you can import it to all other web servers at your site that will issue and accept domain cookies. You must name and create a password for the export file. The file is then stored in the Web Agent directory (the default directory is **rsawebagent**).
3 Import Domain Secret	If you are configuring protected URL access in a domain environment, use this option to import the domain secret from other Agent-protected web servers. You are asked for the filename and file password that you set up in option 2 (Generate and Export Domain Secret).

Line Option	Description
Current Domain Options	**The following options appear only if you chose number 10 (Current Domain Access) in the Configuration menu.**
4 Domain Name	Use this option to create subdomains. For example, suppose you have http://server1.domain1.domain.com http://server2.domain1.domain.com http://server3.domain2.domain.com http://server4.domain2.domain.com and you want to protect URLs on all of these servers. By entering **domain.com** as the Domain Name, you create a subdomain which includes all of the preceding web servers . You must enter a domain name.
5 Cookie Name	Use this option to change the default cookie name (rsacookie). Maximum name length is 30 characters.
Multi-Domain Options	**The following options appear only if you chose number 11 (Multi-Domain Access) in the Configuration menu.**
6 Add to Multi-Domain List	Enter the Agent-protected web servers on which you want all users to access protected URLs once they have authenticated. Use the format http://*server1.domain1.com*. You must enter a domain name.
7 Remove from Multi-Domain List	The Multi-Domain List of Agent-protected web servers displays. Choose the number of the web server you want to remove from the list. (This option does not appear if there are no hosts in the Multi-Domain List.)
8 View Multi-Domain List	View the list of Agent-protected web servers you entered with option 6 for the Multi-Domain List. (This option does not appear if there are no web servers in the Multi-Domain List.)

CAUTION: If you have separate web servers that authenticate users to separate Authentication Manager databases, specify different domain secrets for the different domain cookies. Otherwise, users might gain unauthorized access to protected URLs.

After you have configured the Agent for the first time following installation, the product registration web page opens. If you choose not to register now, you can access the page at your convenience, or you can run the registration script (**./registerWA**) from the Web Agent installation directory.

Important: RSA recommends that you register the software to ensure that you receive security patches as they become available.

Changing Configuration Settings

You may need to make adjustments to the default configurations for the Web Agent. For example, you may find that you need a longer cookie expiration time.

To change configuration settings:

1. Run the configuration script in the Web Agent installation directory. Type

   ```
   ./Config
   ```

   A list of the current web server and any virtual servers you have set up in the web server configuration file displays.

2. Choose the server you want to configure. You can make changes to the default settings applied to all servers, or you can make changes to an individual server.

For details about the different configuration menus, see the preceding section, "Configuring the Software."

Managing URLs

By default, the Web Agent protects all URLs on the web server on which the Agent is installed. The **protectURL** utility is an interactive menu from which you can add, remove, or unprotect individual URLs. The **protectURL** utility is located in the default Web Agent directory. Type

```
./protectURL
```

You can also manage the protected resource list by importing a list of URLs from a file. To add URLs to the protected resource list, type

```
./protectURL -a -f listURL
```

where *listURL* is a text file that contains a list of URLs, with one URL per line, that you want to add to the resource list.

To remove protected URLs from the resource list, type

```
./protectURL -d -f listURL
```

All of the URLs listed in the file are removed from the protected resource list.

Important: When you unprotect a URL, all URLs under it are also unprotected.

Advanced UNIX administrators can manage the protected resource list using command line-only operations. For a list of options and syntax, type

```
./protectURL -h
```

Adding and Removing Virtual Web Servers

To add additional virtual servers to the Web Agent configuration:

1. Run the configuration script with the name of the virtual web server. Type

   ```
   ./config server.domain.com
   ```

2. Verify that you want to create the new server.
   The Setup menu displays.

For details about the different configuration menus, see "Configuring the Software" on page 15.

You can add as many virtual servers as you want. However, if you want access to protected URLs to function the same way on all virtual web servers, you need to make changes to your default web server rather than individual virtual servers.

To remove a virtual server from the Web Agent configuration file:

Use the -d option. Type

```
./config -d server.domain.com
```

Note: Removing a virtual server from the configuration file does not remove or disable the web server or the Web Agent.

Using the Log Off URL to Invalidate Web Access Authentication Cookies

Using the Log Off URL, you can set up a link on a web page that automatically invalidates users' Web access authentication cookies and prompts users to authenticate.

To set up the Log Off URL, add the following URL to a link on your web pages:

http://www.*server.domain*.com/webauthentication?logoff?referrer=/*sample*.html

where

- ***server*** is the name of your server
- ***domain*** is the name of your domain
- ***sample*.html** is the web page

Important: If you do not provide an argument to **referrer=**, users are sent to the root directory on the virtual Web server.

Using Auto-Redirect Scripts to Enforce RSA SecurID Authentication

The Web Agent includes an auto-redirect script that enables you to require users to authenticate before accessing a URL that is not formally protected by RSA SecurID. The URL does not have to be hosted on the same server or be within the same domain as the server on which the Web Agent is installed.

You use the customized redirect URL from the script as the hyperlink to the unprotected site. When a user clicks the HTML link to the URL that you want to protect, the script is invoked, and the user is forced to authenticate before gaining access to the site.

The Perl script included with the Web Agent is a sample script only. To use it, you must first customize it with your own code.

To customize an auto-redirect script:

1. Copy the Perl sample script (**PerlScriptRedirect.pl**) from the **/cgi_scripts** directory of your Web Agent installation, and store it in the web server's **/cgi-bin** directory.
2. Customize the script with your own code.

   Important: RSA strongly recommends that your script contain a list of URLs that users are allowed to access using the redirect URL. The script's input argument should be compared to the list of allowed URLs before any redirect takes place. Any user who attempts to access the redirect hyperlink can see the link definition and could potentially use the redirect script to access the authentication cookie. By implementing a URL comparison list, you minimize the security risk.

3. Use the customized redirect URL from the script as the hyperlink to the unprotected site.

An example redirect URL looks like this:

http://*protectedHostname*/webauthentication?referrer=/cgi-bin/PerlScriptRedirect.pl?target=http://*unprotectedHostname*/new_application.jsp

- **/webauthentication/** is the virtual Web Agent reference. It ensures that a user attempting to access the unprotected URL is prompted to authenticate.
- **/cgi-bin/PerlScriptRedirect.asp** is the script that performs the redirect to the input argument.
- **http://*unprotectedHostname*/new_application.jsp** is the input argument, or unprotected URL.

For more information about customizing auto-redirect scripts, see the directions included in each script.

Configuring the Agent for Proxy Servers

To authenticate through a proxy server, change the value of WebID_URL on the remote Agent-protected web server from the default value of **/webauthentication** to

https://*proxyserver.domain.com/xxx*/webauthentication

where **https://*proxyserver.domain.com*/xxx/** is the path to the root directory of the remote Agent-protected web server.

To make the change, run the Web Agent configuration script on the remote Agent-protected web server. The WebID URL option is in the Setup menu of the configuration program.

4 Customizing Templates and Message Strings

When users authenticate using a web browser or a wireless device microbrowser, the Web Agent software prompts users for their user name and RSA SecurID passcode and informs them about the success of the authentication attempt. For standard browsers, the system returns these messages as HTML pages. For wireless device microbrowsers, the system returns messages in WML format.

The Web Agent software provides default versions of HTML and WML templates and messages. However, you can customize the templates and messages to reflect your company's image and administrative needs. You can

- Add a custom greeting message
- Add your own custom graphics
- Change standard buttons to custom graphics
- Display Web access authentication prompts in a language other than English
- Customize the Web access authentication messages

Important Guidelines

To ensure that the templates will function properly after you have made changes, follow these rules:

- Copy the templates into a new directory before making changes to them. If any templates are missing from this new directory, the Web Agent automatically defaults to the original templates.
- Use a text editor to make changes. Programs such as FrontPage and HomeSite tend to add unnecessary additional HTML/WML tags to templates. In addition, these programs may alter the substitution strings that are necessary in the templates.
- After you have completed your changes, test the templates to make sure they are functioning properly. For information on utilities you can use to troubleshoot problems, see "Troubleshooting" on page 33.
- For security purposes, do not change the administrative privileges when customizing templates. In addition, the web server may not be able to read the templates if you change the privileges.

Important: Do not alter any of the substitution strings in the templates or message text files (**webagent.msg** and **strings.txt**). These strings begin with two "at" signs (@@). The substitution strings are used to include error messages and text from the RSA Authentication Manager and provide placeholders for graphics and message strings.

Customizing Templates

During the Agent installation, the default templates are copied into the **/templates** directory of your Web Agent installation. If you decide to use customized templates, you must store them in a different directory.

To access the templates and text strings, log on as a web server user as defined in the web server configuration file. To specify the location of a virtual server's customized templates, run the Web Agent Setup configuration script. For directions, see "Using the Setup Menu" on page 16.

Modifying Static Text

You can change the static text that appears in Web access authentication templates, or you can add your own static text.

To modify the text in a Web access authentication template:

1. Using a text editor, open one of the templates in the directory. The templates are listed in "List of Templates" on page 30.

   Important: When editing templates, avoid altering the contents of substitution strings. These strings begin with two "at" signs (@@).

2. Delete the static text you want to change, and add the new text.

   For example, the tag **<H1>Welcome to Widgets, Inc.</H1>**, when placed in the **passcode.htm** or **passcode.wml** file, changes the text of the first heading in that page from "RSA SecurID Passcode Request" to "Welcome to Widgets, Inc."

3. Save and close the file.

Adding Custom Graphics

You can add one or more custom graphics to the Web access authentication templates.

Note: WAP/WML devices usually have limited display space for graphics. Be sure the use of graphics is appropriate for your WAP devices before using them.

To add a custom graphic to a Web access authentication template:

1. Using a text editor, open one of the templates in the directory. The templates are listed in "List of Templates" on page 30.
2. Decide where you want the image to be placed on the page, then insert the appropriate tag in the HTML or WML markup pointing to the image file. Use one of the following methods for naming graphic files:
   - A substitution macro (@@URL?GetPic?image=) works with HTML and WML. With WML, the images must be WBMP. With HTML, the images must be JPG. Substitution macros cannot have absolute paths. The images must be in the same directory as the templates, and you must omit the filename extension from the file specification, as in the following example:

     ```
     <IMG src="@@URL?GetPic?image=logo" ALIGN="left">
     ```

- You can use HTTP URLs instead of substitutions if the image files reside in an area of the server that is unprotected by RSA SecurID authentication, or on a separate server hosting the URL. HTTP URLs are always absolute; relative URLs cannot be used in templates. The image types for HTTP URLs can be **.jpg**, **.gif**, or **.wbmp**, as in the following example:

  ```
  <IMG src="http://server.domain.com/img/logo.jpg"
  ALIGN="left">
  ```

  Note: When using HTTP URLs, ensure the image file you point to in the **src** path is in a directory that is not protected by RSA SecurID and that you always specify a fully qualified path to the image file.

3. Save and close the file.
4. Stop and restart the web server for the changes to take effect.

   The web authentication prompt displays the new graphic.

Changing the Buttons (HTML Only)

Changing the Send, Reset, and Cancel Buttons

You can replace the standard **Send**, **Reset**, and **Cancel** buttons that are displayed in the HTML templates with custom graphics.

Note: Make sure the image file you point to in the **src** path is in a directory that is not protected by RSA SecurID and that you always specify a fully qualified path to the image file.

To change the buttons in a Web access authentication template:

1. Using a text editor, open one of the HTML templates in the directory. The templates are listed in "List of Templates" on page 30.
2. Do one or all of the following:
   - To replace the **Send** button, replace the line that reads

     ```
     <INPUT TYPE=SUBMIT VALUE="Send">
     ```

     with

     ```
     <A HREF="JavaScript:document.forms[0].submit()"><IMG
     SRC="path to your image" BORDER="0"></A>
     ```

     where ***path to your image*** is a fully qualified path to an image file.
   - To replace the **Reset** button, replace the line

     ```
     <INPUT TYPE=RESET VALUE="Reset">
     ```

     with

     ```
     <A HREF="JavaScript:document.forms[0].reset()"><IMG
     SRC="path to your image" BORDER="0"></A>
     ```

     where ***path to your image*** is a fully qualified path to an image file.

- To replace the **Cancel** button, replace the line

  ```
  <INPUT TYPE=CANCEL VALUE="Cancel">
  ```

  with

  ```
  <A HREF="JavaScript:document.forms[0].cancel()"><IMG
  SRC="path to your image" BORDER="0"></A>
  ```

  where ***path to your image*** is a fully qualified path to an image file.

3. Save and close the file.
4. Stop and restart the web server for the changes to take effect.

Customizing Templates for Another Language

If you need to customize the templates for a language other than English, you must store them in a language-specific directory under the Web Agent templates directory.

The default directory for language specific templates is ***/Web_Agent_installation_directory*/templates/nls/*language_code*** where *language_code* is the language preference code used by web browsers.

To find the correct language code, refer to the language preferences code list in the Internet Explorer or Netscape Navigator web browsers.

Important Guidelines

- Your end users must have their browser language preference set to use the appropriate language code.
- The code must correspond to your language-customized template directory name. The new language preference must appear at the top of end users' web browser's list of language preferences.
- If the preference settings are incorrect, language-customized templates do not exist, or the Agent cannot find the specified templates for a virtual web server, the browser displays the default English version of the templates.

To translate HTML and WML forms for a non-English language:

1. Create a language-specific subdirectory in the templates directory of the Web Agent.

   For example:

   ```
   ./web_server_directory/rsawebagent/Templates/nls/fr
   ```

   where **fr** is the language preference code for French.
2. Copy the templates to the language-specific subdirectory that you have just created.
3. Customize the text strings within the templates.

Note: Do not remove the substitution macros. These macros begin with a double at sign (@@) in the text. The macros are replaced with actual values when the text is displayed.

4. Run the Web Agent configuration script, and update the **Template** path in the **Setup** menu to point to the language specific templates.

Customizing Message Strings

You can customize certain messages that display while users interact with the Web access authentication prompt pages that are produced from the templates. The message strings are contained in a file named **strings.txt** located in the */**Web_Agent_installation_directory**/**templates** directory.

For example, **strings.txt** contains passcode page errors like:

```
[Messages]
; PASSCODE page errors and messages.
1="100: Access denied. The RSA Authentication Manager
rejected the PASSCODE you supplied. Please try again."
2="101: Access denied. Unexpected RSA Authentication
AgentError %d. Please try again."
3="102: You must enter a valid PASSCODE. Please try again."
```

Important: If you modify the message strings, make certain that you do not remove or alter the position of the variable strings (**@@SUB1**, **@@SUB2**, and so on) contained in the message text. The strings are replaced by actual values when the messages are displayed.

To customize the text displayed by the **multidom.htm** or **multidom.wml** template, search for the following section in the **strings.txt** file:

```
; multiple domain authentication string
; This is HTML only
22="<strong>Requesting authentication from server
@@SUB1</strong> <img alt=failed src="@@SUB2"><br>"
; This is for WML with image tag support
23="<strong>Server @@SUB1 <img alt="failed"
src="@@SUB2"/></strong><br/>"
```

Note: If you translate the text messages in **strings.txt** into a language other than English, you must store the translated file in the same language-specific directory where other translated templates are stored. For more information, see "Customizing Templates for Another Language" on page 28.

List of Templates

The following table summarizes the purpose of each template.

Note: If you are using RSA SecurID PINPads instead of tokens, you need to change the **passcode** and/or **useridandpasscode** templates to display the correct message to your users. The correct message to display is included in the templates in a comment section.

Template	Purpose
Errors	
error.htm **error.wml**	The page that RSA SecurID users see when a fatal error occurs during authentication. The @@sub macro in the template substitutes the error message passed from the system or from the **strings.txt** file.
forbidden.htm **forbidden.wml**	The page that RSA SecurID users see in response to requesting a forbidden URL.
Authentication Templates	
newpin.htm **newpin.wml**	The New PIN page displayed when users are authenticating with their token for the first time. From this page, users create their own PINs.
newpin1.htm **newpin1.wml**	The New PIN page displayed to a user that will receive a system-generated PIN. This functionality is determined in the RSA Authentication Manager.
newpin2.htm **newpin2.wml**	The New PIN page displayed when a user is given the choice of whether to create their own PIN or receive a system-generated PIN. This functionality is determined in the RSA Authentication Manager.
nextprn.htm **nextprn.wml**	The page displayed when a token is in Next Tokencode mode. This happens when a user enters a series of incorrect passcodes during authentication. After the user enters a correct tokencode, the user is prompted for another correct tokencode before being allowed access.
sslredir.htm **sslredir.wml**	The page users might see momentarily with some browsers when they must use a secure channel to access protected pages. In some cases, users must click a link on the **sslredir** page to continue.

Template	Purpose
redirect.htm/redirect-get.htm **redirect.wml**	The page displayed when users complete the authorization process or when they log out. **Note**: If you customize **redirect.htm**, you must customize **redirect-get.htm** to look the same.
redirectmanual.wml	This page is displayed to cell phone users when the cell phone does not support automatic redirection to a protected URL. The user is provided with a list of secure URLs and must manually choose one.
cancel.htm/cancel-get.htm **cancel.wml**	The page displayed to users when they cancel the authorization process. **Note**: If you customize **cancel.htm**, you must customize **cancel-get.htm** to look the same.
showsys.htm **showsys.wml**	The page displayed to users for ten seconds while the system generates an RSA SecurID PIN for them.
multidom.htm/multidom-get.htm **multidom.wml**	The page displayed when users are authenticating across multiple domains. **Note**: If you customize **multidom.htm**, you must customize **multidom-get.htm** to look the same.
userid.htm **userid.wml**	If you chose to present separate web pages to users to input the user name and passcode, this template is used for the user name. If you did not choose to present separate pages, the **useridandpasscode** template is used.
passcode.htm **passcode.wml**	If you chose to present separate web pages to users to input the user name and passcode, this template is used for the passcode. If you did not choose to present separate pages, the **useridandpasscode** template is used.
useridandpasscode.htm **useridandpasscode.wml**	If you chose to present one web page to users to input both the user name and passcode, this template is used. If you chose to present separate web pages to input the user name and passcode, the **userid** and **passcode** templates are used.

The HTML and WML forms are supported by the following files, which are also installed into the Templates directory:

Template	Purpose
Bitmaps	
denied.jpg **denied.wbmp**	If you have configured the Web Agent to allow multiple domain authentications, the word “Denied” is displayed if a user’s authentication request to a virtual web server does not succeed.
ok.jpg **ok.wbmp**	If you have configured the Web Agent to allow multiple domain authentications, the word “OK” is displayed if a user’s authentication request to a virtual web server succeeds.
rsalogo.jpg	This is the background graphic used on the authentication pages.
securid_banner.jpg	This graphic displays the RSA SecurID banner on the authentication pages.
Other Files	
strings.txt	This file contains text strings that are used to display various messages while users interact with the Web access authentication prompt pages.
style.css	The cascading style sheet used for the web pages.

5 Troubleshooting

RSA Authentication Manager Utilities

Use these utilities to determine communication between the Web Agent and the Authentication Manager.

These utilities reside in the Web Agent directory (***/web_server_directory*/rsawebagent** is the default).

acestatus

This utility provides information about the Authentication Manager such as the configuration version, the server name and address, the number of client retries, and the client time-out period.

acetest

This utility enables you to authenticate to the Authentication Manager from the command line rather than going through authentication web pages in your browser. This will help you determine whether a problem lies with the templates or with the authentication process itself.

Important: Make sure you run **acetest** as the user who owns the web server. Otherwise, ownership for the files under the $VAR_ACE environment variable may change and cause RSA SecurID authentication to fail.

Logging Authentication Attempts

Authentication attempts are logged in ***/web_server_directory*/logs/error_log**.

Note: The different types of error messages logged can be found in the **webagent.msg** file located in the Web Agent directory (***/web_server_directory*/rsawebagent** is the default).

The following table provides a list of possible error messages and their cause:

Error Message	Possible Cause and Solution
File /usr/local/web_server_directory/conf/*file*.conf isn't writable.	The user account with which you logged on does not have write permissions. Log on with a web server user account that has write permissions to the web server root directory.

Error Message	Possible Cause and Solution
100:Access denied. The RSA Authentication Manager rejected the passcode you supplied. Please try again.	The first time an authentication occurs after the Web Agent has been installed on the web server, the Authentication Manager generates a node secret and sends it to the web server. If the node secret file is missing, or the node secret on the Authentication Manager and web server do not match, users are denied access. Contact your Authentication Manager administrator. If the problem persists, verify that the Agent's hostname resolves to the same IP address throughout the network. Contact your network administrator for assistance.
Unexpected RSA Authentication Agent error 103. Please try again.	This error is received when there are network problems. Contact your Authentication Manager administrator.
AceInitialize Failed during acetest authentication.	• The **sdconf.rec** file is missing. Obtain an **sdconf.rec** file from your Authentication Manager administrator. Place the file in a directory that is accessible to the web server and Web Agent software. Restart the web server. • Verify that 5500 UDP traffic is not blocked. If it is blocked, the Web Agent does not have a valid route to the RSA Authentication Agent. • Verify that the Authentication Manager is running.
The page cannot be found.	The requested page may not be present.
RSA Securid Error. 106: Web server too busy. Please try again later.	This error may occur when communication to the Authentication Manager is down or the **sdconf.rec** file is missing. Contact your Authentication Manager administrator.
Unexpected authentication error.	This error may occur when authenticating using the **acetest** utility. Communication to the Authentication Manager is down. Contact your Authentication Manager administrator.
The Page cannot be displayed.	There are two possible causes for this error message: • Communication to the web server is down. • The web server was started without SSL. Therefore, the Redirect Secure feature in the Web Agent is disabled. The best solution is to restart the web server with SSL. You could also have users access the page with an **https** request.

Error Message	Possible Cause and Solution
RSA Web Access Authentication Extension Error. RSA Web Access Authentication: Internal server configuration error.	The path to the templates is invalid. Verify the correct path in the Web Agent configuration.
For Multi-Domain Authentication: Requesting authentication from server http://server Denied.	Make sure that the same domain secret exists on each web server within the multi-domain area.

ErrorMessages

The Web Agent logs events in the web server error log file. This section lists all error and event messages alphabetically.

ACECheck processing error for userid *username*

If the **ACECheck** function returns an error, an Authentication Manager time-out or some other communications error has occurred.

ACEClose processing error *errornumber*

If the **ACEClose** function returns an error, an RSA Authentication Manager time-out or some other communications error has occurred.

ACENext processing error for userid *username*

If the **ACENext** function returns an error, an Authentication Manager time-out or some other communications error has occurred.

ACEPin processing error for userid *username*

If the **ACEPin** function returns an error, an Authentication Manager time-out or some other communications error has occurred.

Authentication Manager: Access Denied.

The user did not enter a valid RSA SecurID passcode.

Authentication Manager: Invalid RSA Authentication Manager configuration. User *username*.

The **sdconf.rec** file is not valid. The file is either corrupted, has been moved to another directory, or has been deleted from the system.

To correct the problem, get a new copy of **sdconf.rec** from your Authentication Manager administrator.

Authentication Manager: New PIN Accepted. User *username*.

The user successfully associated a new PIN with his or her token.

Authentication Manager: New PIN Rejected. User *username*.

The user did not successfully associate a new PIN with his or her token. If the user is attempting to create his or her own PIN, make sure the user understands the PIN length and syntax parameter settings for your Authentication Manager.

Authentication Manager: Next Tokencode Accepted. User *username*.

After entering a series of bad passcodes, the user was prompted to enter the next tokencode from his or her token. The next tokencode was valid and the user was authenticated successfully.

Authentication Manager: User Canceled New PIN Mode. User *username*.

The user was prompted to associate a new PIN with his or her token, but the user did not complete the new PIN procedure. Make sure the user understands how to use his or her token in New PIN mode.

Authentication Manager: User Canceled Transaction. User *username*.

The user was prompted to authenticate, but then canceled out of the Enter passcode dialog box. This is a purely informational message.

Authentication Manager: User I/O Timeout. User *username*.

The user waited too long at the **Enter passcode** prompt, so the RSA Authentication Agent canceled the transaction.

Cookie rejected. Cached client info does not match.

If a user is using more than one workstation, this message appears each time the user switches from one workstation to another.

Cookie rejected. Cookie failed MD5 test.

An unauthorized user has attempted to access the web server with a bogus Web access authentication cookie.

Cookie rejected. Expired cookie. Username *username*

A Web access authentication cookie has expired in response to the time-out values defined in the web properties sheet.

Could not initialize RSA Authentication Agent

Will be preceded by a number of RSA Authentication Agent error messages, such as **Cannot find sdconf.rec**. Try reinstalling the **sdconf.rec** file.

Could not initialize Cookie Cache

A memory error has occurred within an internal function. Your web server may be overloaded; you may need more physical memory.

Could not open HTML template *filename*

The HTML template file is missing.

Also check the security settings for the file. Make sure the account that the web server is running has Full Access privileges to the HTML file.

Could not query value *valuename*

If you have enabled the Domain Cookies feature without setting a domain secret, you might get a **valuename DomainData** message, followed by a **Domain cookies are disabled** message.

Could not read HTML template *filename*

The HTML template file is missing.

Could not resolve hostname *hostname*

The DNS function of the web server is configured incorrectly. Domain cookies cannot be used until the configuration is corrected.

Failed authentication for userid *username*.

The Authentication Manager did not grant the user access; the most common causes for this are wrong user name or an invalid passcode.

Failed to create service thread, aborting.

There were too many other processes running, so the service did not start.

File incorrect size: sdconf.rec.

It is likely that the **sdconf.rec** file was not copied in binary or ftp mode. Ask the Authentication Manager administrator for a new copy of **sdconf.rec**.

File not found: sdconf.rec.

The **sdconf.rec** file was either removed or never copied from the Authentication Manager. Ask the Authentication Manager administrator for a new copy of **sdconf.rec**.

New PIN accepted for userid *username*.

The Authentication Manager verified the RSA SecurID user's new PIN.

New PIN rejected for userid *username*.

The PIN was rejected by the Authentication Manager. The user must reauthenticate to set the PIN. Check the Activity Log on the Authentication Manager.

New PIN requested from userid *username*.

The Authentication Manager has prompted the RSA SecurID user to create his or her own PIN or receive a system-generated PIN.

Next code accepted for userid *username*.

The Next Tokencode was accepted by the Authentication Manager and access was granted.

Next code rejected for userid *username*.

The user must reauthenticate.

Next code requested from userid *username*.

The user's token was in Next Tokencode mode and the Authentication Manager asked for the second tokencode.

No cookie or corrupted information.

This message will appear each time a new user logs in to the web server.

Out of memory in *functionname*.

A memory error has occurred within an internal function. Your web server may be overloaded; you may need more physical memory.

Remote authentication denied for userid *username*.

Another web sever within the DNS domain has requested authentication of user *username* with a domain cookie and was not given access.

Check the security settings for the file. Make sure the account that the web server is running has Full Access privileges to the HTML file.

Remote authentication given for userid *username*.

Another web server within the DNS domain has requested authentication of user *username* with a domain cookie and was given access.

Remote authentication received deny for userid *username*.

A web server requesting authentication of a domain cookie was rejected.

Remote cookie rejected. Cookie failed MD5 test.

An unauthorized user has attempted to access the web server with a bogus Web access authentication domain cookie.

RSA Authentication Agent initialization failed

The Agent cannot make the connection to the Authentication Manager. Make sure the Authentication Manager and network are operational and that all network interface cards and cables are properly installed and in good condition.

RSA Authentication Manager is not responding

There is a network communications problem between the Authentication Manager and the RSA Authentication Agent, the server cannot be found (because the IP address is wrong, for example), or the Authentication Manager daemon is not running.

RSA Authentication Manager is not responding. Run CLNTCHK to verify port and IP address of RSA Authentication Manager.

There is a network communications problem between the Authentication Manager and the RSA Authentication Agent, the Authentication Manager cannot be found (because the IP address is wrong, for example), or the Authentication Manager daemon is not running.

Session Manager: Failed to Create Server Thread.

There are too many server threads running (too many users connecting at once). Try widening the intervals at which users attempt to log on.

Session Manager: Failed to Resolve Hostname.

Most likely a configuration error. The machine that is connecting has no DNS or NetBIOS name, or has an invalid IP address. Make sure your network is configured properly and that your host file entries are correct.

Session Manager: Not Enough Memory.

The system does not have enough physical RAM, or there were too many other processes running in memory. If you receive this message often, add more physical memory to the computer.

The security descriptor could not be found. The file may not exist: *filename*.

A user requested a URL that does not resolve to a file on the machine. Make sure the user is entering the URL correctly.

The user *server/username* disconnected from port *portnumber*.

The user closed the connection on the specified port.

The user *server/username* connected on port *portnumber* on date at time and disconnected on date at time. . .

A normal RSA Authentication Agent disconnection has occurred.

The user *username* has connected and been authenticated on port *portnumber*.

A normal (authenticated) RSA Authentication Agent-Server connection occurred.

Unexpected error from RSA Authentication Agent.

The value returned by the Authentication Manager is not valid.

User <blank> canceled out of RSA SecurID Authentication routine.

The user canceled without entering a user name.

User I/O Timeout-User took too long to respond.

The system timed out after waiting for a response from the user.

User *username* canceled out of New PIN routine.

The user canceled the authentication attempt.

User *username*: ACCESS DENIED. ATTEMPT 1.

The user was denied access. Check the Authentication Manager Activity Log for the specific reason.

User *username*: Access denied. Attempt to use invalid handle. Closing connection.

An internal error occurred. If the message recurs, call the RSA Customer Support Center.

User *username*: ACCESS DENIED. Next Tokencode failed.

The user must reauthenticate.

User *username*: ACCESS DENIED. Server signature invalid.

This message indicates that the identity of the Authentication Manager could not be verified by the client. If you see this message, call the RSA Customer Support Center.

User *username*: ACE Check Error: Invalid group SID. Passcode required.

The user's group SID did not contain a valid group name. The user was challenged for an RSA SecurID passcode.

User *username*: canceled out of Next Tokencode routine.

The user canceled out of the Next Tokencode process.

User *username*: canceled out of RSA SecurID Authentication routine.

The user canceled after entering a user name.

User *username*: Domain not found. User challenged for passcode.

The user may have entered the domain name incorrectly and will be challenged for a passcode.

User *username*: New PIN accepted.

The user's New PIN was verified.

User *username*: New PIN rejected.

The PIN was rejected by the Authentication Manager. The user needs to reauthenticate to set the PIN. Check the Authentication Manager Activity Log.

User *username*: Not found. User challenged for passcode.

The user is unknown to the system, but the system still challenges the user for a passcode.

User *username*: Successfully logged on with Next Tokencode.

The Next Tokencode was accepted by Authentication Manager and access was granted to the user.

Known Problems of Third-Party Software

Netscape 7.2 Browser Issues

Unlike Internet Explorer, Netscape maintains a single browser session across multiple instances of the browser. If a user has successfully authenticated to a protected resource in one instance of the browser, as long as that instance remains open, all other instances of the browser share the same authentication cookie. Therefore, the user does not have to reauthenticate in any other instances of the Netscape browser to access protected resources.

To exit the browser session, users must close all instances of the browser.

Wireless Devices

A user could experience the following scenarios when using a cellular phone equipped with a microbrowser to access protected URLs:

- If your environment includes a GSM network, your WAP connection needs to be in connection mode. Multiple domain environments require that handset devices and gateways support the receipt of cookies from multiple domains.
- Requiring an SSL connection to protected URLs creates a more secure environment. For ease of use, you can configure the Web Agent to automatically redirect the URL request to a secure connection.

  However, if your microbrowser does not support automatic redirection, you must disable the redirect option. Instead of automatic redirection, a web page opens that contains a link to the secure connection.
- When the Web Agent is configured to use a single web page for entering the user name and passcode, the LCD on certain devices may appear to be using separate pages, one for entering the user name and a second page for entering the passcode. However, the microbrowser on the device is sending the data all at once, unless you have specifically enabled the **Use Separate username and Passcode Pages** option in the Web Agent.
- When **Name Locking** and **Use Separate Username and Passcode Pages** are enabled in the Web Agent, and the carrier signal is lost after transmitting the user name, the user name is locked in the Web Agent database until the Name Lock time-out expires. Instruct the user to authenticate again after the Name Lock expiration time.
- It can be difficult for users to enter the PIN and tokencode within the designated time limit (typically 60 seconds) before the tokencode changes again. Most WAP devices by default are set up for alphanumeric entries. That means the user must scroll through the letters assigned to a button before reaching the numbers. Since tokencodes are always numeric, instruct users to switch their phone to numeric entry, if their phone allows this, only after entering the PIN.
- Some gateways have very specific size limitations for WML templates. You may need to reduce the amount of information provided in the templates.

- To enable the **Redirect HTTP Connections to Secure Server** option, the cellular device and its gateway must allow for SSL redirection. RSA recommends that you instruct the user to refer to the documentation provided with his or her cellular device.
- Devices that allow for an image display may, during the course of an authentication, display the status "Failed" for several seconds (depending on the speed of the micro browser) until an image is shown on the LCD that indicates success. In these instances, the user should wait for several seconds until the success image is shown. If, however, the "Failed" status message is displayed for a substantial amount of time, it is most likely valid, and the user should reauthenticate.
- For increased security on WAP browsers, RSA recommends setting the cookie expiration times to less than the defaults of 15 minutes for idle cookies and 60 minutes for all cookies.

Multi-Domain Issues

When connecting to multiple domains, a web page is displayed showing the domain URL and the success or failure of the connection. In some environments, the appropriate GIFs do not appear in the web page. This problem occurs only when there is no valid certificate on the web server. If this occurs, use **http** instead of **https** when you input domains in your multi-domain list.

The following issues may occur when using multi-domain access on wireless devices:

- When Multi-Domain Access is enabled in the Web Agent, a list of URLs for the domains is displayed. WAP devices that allow for an image display may, during the course of an authentication, display the "Failed" status for several seconds until an image is shown on the LCD that indicates success. In these instances, the user should wait for several seconds until the success image is shown. However, if the "Failed" status message remains for a substantial amount of time, it is most likely valid, and the user should reauthenticate.
- When multi-domain is enabled, the Web Agent attempts to get an image from each of the domains to verify the connection. With some cell phones, the image is displayed even though the connection was never actually made. The user is forced to reauthenticate each time he or she attempts to access a URL in another domain.

  To work around this issue, set the variable **UseTextWML=1** in the **RSAWebAgent.ini** file located in the Web Agent installation directory (the default is **rsawebagent**). This forces the user to manually click a text link for each domain instead of attempting to automatically make the connection using images.

Index