Oracle® Secure Global Desktop

Gateway Administration Guide for Release 5.1

E41497-01
October 2013

Oracle® Secure Global Desktop: Gateway Administration Guide for Release 5.1

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this is software or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, the following notice is applicable:

U.S. GOVERNMENT END USERS: Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are "commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

This software or hardware and documentation may provide access to or information on content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services.

Abstract

This guide describes how to install, configure, and operate the Oracle Secure Global Desktop Gateway.

Document generated on: 2013-10-22 (revision: 2183)

Table of Contents

Preface

The *Oracle Secure Global Desktop Gateway Administration Guide* provides instructions for installing, configuring, and operating the Oracle Secure Global Desktop (SGD Gateway). The document is written for system administrators.

1. Audience

This document is intended for new users of the SGD Gateway. It is assumed that readers are familiar with Web technologies and have a general understanding of Windows and UNIX platforms.

2. Document Organization

The document is organized as follows:

- Chapter 1, *Installing the SGD Gateway* describes how to install the SGD Gateway.
- Chapter 2, *Configuring the SGD Gateway* describes how to configure the SGD Gateway for your network.
- Appendix A, *SGD Gateway Architecture Overview* describes the architecture of the SGD Gateway.
- Appendix B, *Command-Line Reference* describes how to configure and control the SGD Gateway from the command line.
- Appendix C, *Advanced Configuration* covers advanced configuration of the SGD Gateway, including how to configure and use the reflection service of the SGD Gateway.
- Appendix D, *Troubleshooting the SGD Gateway* includes troubleshooting information, to help you to diagnose and fix problems with the SGD Gateway.

3. Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc.

Access to Oracle Support

Oracle customers have access to electronic support through My Oracle Support. For information, visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info or visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs if you are hearing impaired.

4. Related Documents

The documentation for this product is available at:

http://www.oracle.com/technetwork/documentation/sgd-193668.html

For additional information, see the following manuals:

- *Oracle Secure Global Desktop Administration Guide*
- *Oracle Secure Global Desktop Installation Guide*
- *Oracle Secure Global Desktop User Guide*

- *Oracle Secure Global Desktop Platform Support and Release Notes*
- *Oracle Secure Global Desktop Security Guide*

5. Conventions

The following text conventions are used in this document:

Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.
italic	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
`monospace`	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.

Chapter 1. Installing the SGD Gateway

Following a brief introduction to the Oracle Secure Global Desktop Gateway (SGD Gateway), this chapter describes how to install the SGD Gateway software. The chapter also includes details of system requirements for the SGD Gateway.

This chapter includes the following topics:

1.1. About the SGD Gateway

The SGD Gateway is a proxy server designed to be deployed in front of an SGD array in a demilitarized zone (DMZ). This enables the SGD array to be located on the internal network of an organization. Additionally, all connections can be authenticated in the DMZ before any connections are made to the SGD servers in the array.

Using the SGD Gateway is an alternative to running your SGD servers with firewall traversal, also called firewall forwarding.

The SGD Gateway manages load balancing of HTTP connections, so you do not need to use the JavaServer Pages (JSP) technology load balancing page included with SGD.

1.2. System Requirements

The supported installation platforms for the SGD Gateway host are listed in the *Oracle Secure Global Desktop Platform Support and Release Notes*.

The following requirements apply for the SGD servers used with the SGD Gateway:

- **Secure mode.** By default, the SGD Gateway uses secure connections to SGD servers. Your SGD servers must have secure connections enabled. Firewall forwarding must not be enabled.

  In a standard installation, an SGD server is configured automatically to use secure connections and firewall forwarding is disabled. See Secure Connections to SGD Servers if you need information on how to secure an SGD server.

- **SGD version.** It is best to use version 5.1 of SGD with version 5.1 of the Gateway. Use the latest version of the Gateway, where possible.

- **Clock synchronization.** It is important that the system clocks on the SGD servers and the SGD Gateway are in synchronization. Use Network Time Protocol (NTP) software, or the `rdate` command, to ensure that the clocks are synchronized.

For more information on SGD server system requirements, see the *Oracle Secure Global Desktop Platform Support and Release Notes*.

1.2.1. Known Issues

See the *Oracle Secure Global Desktop Platform Support and Release Notes* for details of the known issues with this release of the SGD Gateway.

1.3. Performing the Installation

On Oracle Solaris platforms, install the SGD Gateway with the `pkgadd` command.

On Linux platforms, install the SGD Gateway with the `rpm` command.

By default, the SGD Gateway is installed in the `/opt/SUNWsgdg` directory. You can change the installation directory, as follows:

- **Oracle Solaris platforms** – The installation program asks you for the installation directory when you install the software
- **Linux platforms** – You can choose a different installation directory, by using the `--prefix` option with the `rpm` command when you install the software

1.3.1. How To Install the SGD Gateway

1. Save the SGD Gateway package to a temporary directory on the host.

   If you are installing from the installation media, the package is in the `gateway` directory.

   Alternatively, download the installation program from an SGD web server at `https://server.example.com`, where *`server.example.com`* is the name of an SGD server. When the SGD web server Welcome page is displayed, click Install the Oracle Secure Global Desktop Gateway.

   These are the package files:

   - `SUNWsgdg-`*`version`*`.sol-x86.pkg` for Oracle Solaris on x86 platforms
   - `SUNWsgdg-`*`version`*`.sol-sparc.pkg` for Oracle Solaris on SPARC technology platforms
   - `SUNWsgdg-`*`version`*`.i386.rpm` on Linux platforms

   where *`version`* is the SGD Gateway version number.

2. Log in as superuser (root) on the host.
3. Install the SGD Gateway.

   If the package file is compressed, you must expand it before installing.

   To install on Oracle Solaris on x86 platforms:

   ```
   # pkgadd -d /tempdir/SUNWsgdg-version.sol-x86.pkg
   ```

   To install on Oracle Solaris on SPARC technology platforms:

   ```
   # pkgadd -d /tempdir/SUNWsgdg-version.sol-sparc.pkg
   ```

   Note

   On Oracle Solaris platforms, if the installation fails with a `pwd: cannot determine current directory!` error message, change to the */tempdir* directory and try again.

   To install on Linux platforms:

```
# rpm -Uvh /tempdir/SUNWsgdg-version.i386.rpm
```

4. Verify that the SGD Gateway package is registered in the package database.

   On Oracle Solaris platforms:

   ```
   # pkginfo -x SUNWsgdg
   ```

   On Linux platforms:

   ```
   # rpm -qa | grep -i SUNWsgdg
   ```

5. Run the SGD Gateway setup program.

   ```
   # /opt/SUNWsgdg/bin/gateway setup
   ```

   The SGD Gateway setup program presents the following settings that you can accept or change:

   - **SGD Gateway port settings.** The interface and port used by the SGD Gateway for incoming connections. By default, the SGD Gateway listens on port 443 on all interfaces.
   - **Network entry point.** IP address, or Domain Name System (DNS) name, and the port that client devices use to connect to the SGD Gateway. This is not always the same as the address of the SGD Gateway. Depending on the configuration of your network, this can be the address of a load balancer or other external device.

     For example, if users connect directly to an SGD Gateway at `gateway1.example.com`, type `gateway1.example.com:443` for the network entry point.

     If users connect to the SGD Gateway through a load balancer at `lb.example.com`, type `lb.example.com:443` for the network entry point.
   - **Secure connections.** Whether to secure the connections between the SGD Gateway and the SGD servers in the array. By default, the SGD Gateway uses secure connections. To use secure connections, the SGD servers in the array must be running in secure mode.

     See Section C.5, “Using Unencrypted Connections to the SGD Array” for more information about using unencrypted connections to the SGD servers in the array.

   Note

   These settings can be changed later, by using the `gateway config create` command. See Section 2.2.1.1, “How to Configure the Ports and Connections for the SGD Gateway”.

   After installing the software, you must perform additional configuration of the SGD Gateway. See Chapter 2, *Configuring the SGD Gateway* for details of what you need to do.

1.4. Upgrading the SGD Gateway

This section describes how you upgrade the SGD Gateway.

When you upgrade the SGD Gateway, most of your original configuration such as keystores and routing proxy configuration files are preserved. There is no need to reconfigure the Gateway after upgrading.

An upgrade log is created at `/opt/SUNWsgdg/proxy/var/log/upgrade_`*`oldversion`*`_`*`newversion`*`.log`, where *`oldversion`* is the old version of the SGD Gateway, and *`newversion`* is the upgraded version of the SGD Gateway.

When you upgrade, the SGD Gateway installation program backs up any customized Apache web server files and reflection service configuration files it detects. Such files are listed in the upgrade log and have to be manually upgraded. You can use a utility such as `diff` to compare files and show the changes made.

Changes made to the Gateway logging properties file, `/opt/SUNWsgdg/proxy/etc/logging.properties`, are not preserved on upgrade.

1.4.1. How to Upgrade the SGD Gateway

1. Make sure no user sessions and application sessions are running through the SGD Gateway.
2. Install the new version of the SGD Gateway.

   See Section 1.3.1, “How To Install the SGD Gateway”.

   When you run the SGD Gateway setup program, your current configuration settings are used automatically.

Chapter 2. Configuring the SGD Gateway

This chapter describes how to configure the Oracle Secure Global Desktop Gateway (SGD Gateway) for typical deployment scenarios. How to start and stop the SGD Gateway is also covered in this chapter, along with instructions on how to remove the SGD Gateway software.

This chapter includes the following topics:

2.1. Deploying the SGD Gateway

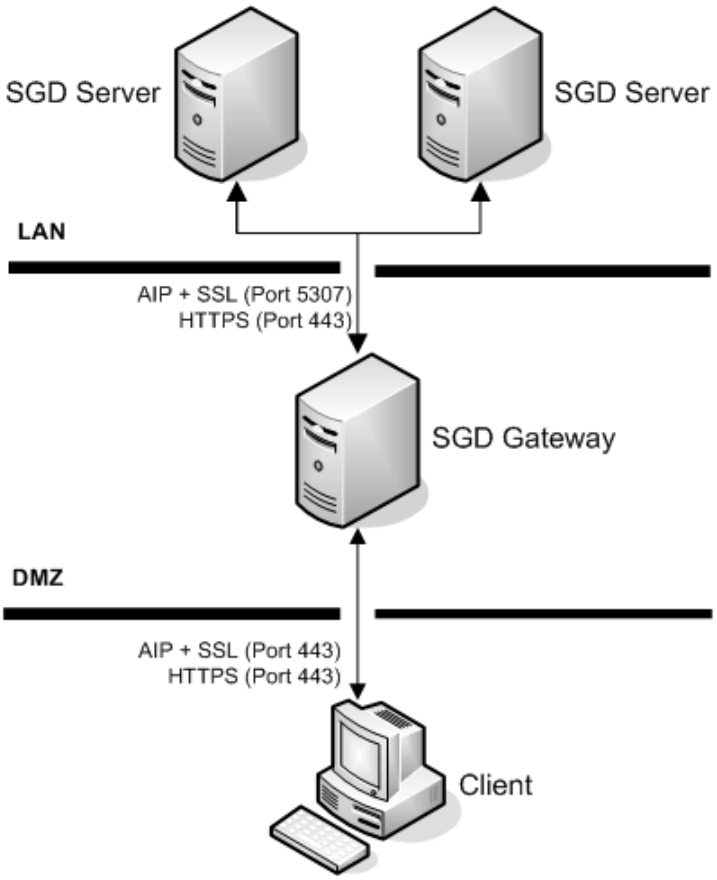
This section describes the following SGD Gateway deployment scenarios:

2.1.1. Basic Deployment

This section describes the configuration tasks for a basic deployment of the SGD Gateway.

A basic deployment uses a single SGD Gateway, as shown in Figure 2.1, “Basic Deployment Using a Single SGD Gateway”.

Figure 2.1. Basic Deployment Using a Single SGD Gateway



Configuring a basic deployment involves configuring the connections shown in Table 2.1, “Connections For a Basic Deployment of the SGD Gateway”.

Table 2.1. Connections For a Basic Deployment of the SGD Gateway

Connection	Configuration Steps
Client device to SGD Gateway	1. Configure the ports and connections used by the SGD Gateway. You configured these settings when you installed the SGD Gateway. See Section 2.2.1.1, “How to Configure the Ports and Connections for the SGD Gateway” if you want to change the configuration of the SGD Gateway. 2. On the SGD Gateway, install a Secure Sockets Layer (SSL) certificate for client connections. See Section 2.2.1.2, “How to Install an SSL Certificate for Client Connections Into the Client Keystore”. 3. (Optional) Configure the SGD Gateway for connections from tablet devices. See Section 2.2.1.3, “How to Configure the SGD Gateway for Connections From Tablet Devices Using Untrusted Certificates”.

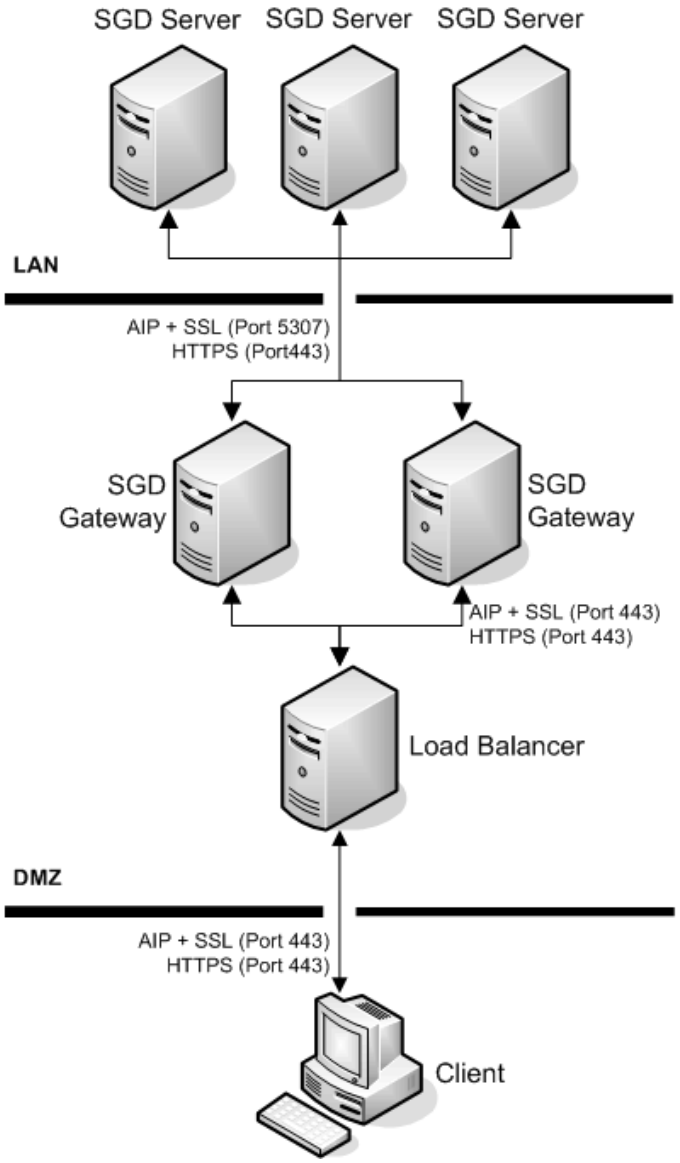
Connection	Configuration Steps
SGD Gateway to SGD servers	1. Enable SGD security services for the array. The SGD servers must be running in secure mode. Firewall forwarding must not be enabled. In a standard installation, an SGD server is configured automatically to use secure connections. See Secure Connections to SGD Servers if you need information on how to secure an SGD server. 2. On the SGD Gateway, install security certificates for the SGD servers. Use the `gateway server` command to import CA certificates and SSL certificates for the SGD servers in the array into the SGD Gateway keystore. See Section 2.2.2.1, “How to Install SGD Server Certificates”. 3. Set up the SGD servers in the array to use the SGD Gateway. Install the SGD Gateway certificate on the SGD array, and use the `tarantella gateway add` command to register the SGD Gateway with the SGD array. See Section 2.2.2.2, “How to Install SGD Gateway Certificates on the SGD Array”. 4. Configure which SGD Client connections can use the SGD Gateway. See Section 2.2.2.3, “How to Configure SGD Client Connections”.

2.1.2. Load-Balanced Deployment

This section describes the configuration tasks for a load-balanced deployment of the SGD Gateway.

A load-balanced deployment uses multiple SGD Gateways and a load balancer as the network entry point, as shown in Figure 2.2, “Network Deployment Using Multiple SGD Gateways and a Load Balancer”.

Figure 2.2. Network Deployment Using Multiple SGD Gateways and a Load Balancer



Configuring a load-balanced deployment involves configuring the connections shown in Table 2.2, "Connections For a Load-Balanced Deployment of the SGD Gateway".

Table 2.2. Connections For a Load-Balanced Deployment of the SGD Gateway

Connection	Configuration tasks
Client device to load balancer	1. Enable incoming connections from client devices. Typically, this uses TCP port 443. See your load balancer documentation for details of how to do this. 2. (Optional) On the load balancer, install the SSL certificate used by the SGD Gateways for client connections. See your load balancer documentation for details of how to do this.

Connection	Configuration tasks
Load balancer to SGD Gateway	1. Configure your load balancer to forward connections to the SGD Gateway. See your load balancer documentation for details of how to do this. 2. Configure the ports and connections used by the SGD Gateway. Set the network entry point to the address of the load balancer. You configured these settings when you installed the SGD Gateway. See Section 2.2.1.1, “How to Configure the Ports and Connections for the SGD Gateway” if you want to change the configuration of the SGD Gateway. 3. On each SGD Gateway, install an SSL certificate for client connections. See Section 2.2.1.2, “How to Install an SSL Certificate for Client Connections Into the Client Keystore”. 4. (Optional) Configure each SGD Gateway for connections from tablet devices. See Section 2.2.1.3, “How to Configure the SGD Gateway for Connections From Tablet Devices Using Untrusted Certificates”.
SGD Gateway to SGD servers	1. Enable SGD security services for the SGD array. The SGD servers must be running in secure mode. Firewall forwarding must not be enabled. In a standard installation, an SGD server is configured automatically to use secure connections. See Secure Connections to SGD Servers if you need information on how to secure an SGD server. 2. On the SGD Gateway, install security certificates for the SGD servers. Use the `gateway server` command to import CA certificates and SSL certificates for the SGD servers in the array into the SGD Gateway keystore. See Section 2.2.2.1, “How to Install SGD Server Certificates”. 3. Set up the SGD servers in the array to use the SGD Gateways. Install SGD Gateway certificates on the SGD array, and use the `tarantella gateway add` command to register the SGD Gateways with the SGD array. See Section 2.2.2.2, “How to Install SGD Gateway Certificates on the SGD Array”. 4. Configure which SGD Client connections can use the SGD Gateways. See Section 2.2.2.3, “How to Configure SGD Client Connections”.

2.2. SGD Gateway Configuration Tasks

This section includes instructions for configuring the connections used by the SGD Gateway.

The following configuration tasks are described:

- Section 2.2.1, “Client Device to SGD Gateway Connections”
- Section 2.2.2, “SGD Gateway to SGD Server Connections”
- Section 2.2.3, “Client Device to Load Balancer Connections”
- Section 2.2.4, “Load Balancer to SGD Gateway Connections”

2.2.1. Client Device to SGD Gateway Connections

Configuring connections between the client device and an SGD Gateway involves the following configuration tasks:

1. (Optional) Configure the ports and connections used by the SGD Gateway.

   You configure these settings when you install the SGD Gateway.

   To change these settings, see Section 2.2.1.1, “How to Configure the Ports and Connections for the SGD Gateway”.

2. (Optional) On the SGD Gateway, install an SSL certificate for client connections.

   See Section 2.2.1.2, “How to Install an SSL Certificate for Client Connections Into the Client Keystore”.

3. (Optional) Configure the SGD Gateway for connections from tablet devices.

   This task is only required if the Gateway is using an untrusted certificate.

   See Section 2.2.1.3, “How to Configure the SGD Gateway for Connections From Tablet Devices Using Untrusted Certificates”.

2.2.1.1. How to Configure the Ports and Connections for the SGD Gateway

You only need to use this procedure if you want to change the settings you made during installation of the SGD Gateway.

1. Log in as superuser (root) on the SGD Gateway host.
2. Run the `gateway config create` command.

   ```
   # /opt/SUNWsgdg/bin/gateway config create
   ```

   Answer the on-screen questions, to configure the following:

   - **SGD Gateway port settings.** The interface and port used by the SGD Gateway for incoming connections.
   - **Network entry point.** The IP address, or DNS name, and port that client devices use to connect to the SGD Gateway. This is not always the same as the address of the SGD Gateway. Depending on the configuration of your network, this can be the address of a load balancer or other external device.

- **Secure connections.** Whether to secure the connections between the SGD Gateway and the SGD servers in the array. To use secure connections, the SGD servers in the array must be running in secure mode.

3. Save the connection and port settings.

   The SGD Gateway is configured using the settings you entered.

2.2.1.2. How to Install an SSL Certificate for Client Connections Into the Client Keystore

The SSL certificate that the SGD Gateway uses for client connections is called the SGD Gateway SSL certificate. The SSL certificate is stored in the client keystore, `/opt/SUNWsgdg/proxy/etc/keystore.client`.

By default, the SGD Gateway uses a *self-signed* SGD Gateway SSL certificate for client connections, but you can replace the self-signed SSL certificate with a certificate signed by a Certificate Authority (CA).

The following procedure assumes you have an SSL certificate signed by a CA.

The private key you install must be in Privacy Enhanced Mail (PEM) format.

1. Log in as superuser (root) on the SGD Gateway host.
2. Copy the SSL certificate and the corresponding private key to the SGD Gateway host.
3. Import the SSL certificate and private key into the client keystore.

   Use the `gateway sslkey import` command, as follows:

```
# /opt/SUNWsgdg/bin/gateway sslkey import \
--keyfile temp.key \
--keyalg RSA \
--certfile example.com.pem
```

   Here, the certificate file *`example.com.pem`* and the corresponding RSA-encoded private key, *`temp.key`*, are imported into the client keystore.

   The existing self-signed SSL certificate in the client keystore is overwritten.

4. (Optional) Restart the SGD Gateway.

Caution

Only use this step if you are not performing initial configuration of the SGD Gateway. Restarting the SGD Gateway at this stage during initial configuration displays an error message, because initial configuration of the SGD Gateway has not been completed.

   Restart the SGD Gateway if you are replacing the SSL certificate on an SGD Gateway that is already configured and running.

Note

Restarting the SGD Gateway disconnects all user sessions and application sessions that are running through the SGD Gateway.

   On the SGD Gateway host, run the following command:

```
# /opt/SUNWsgdg/bin/gateway restart
```

2.2.1.3. How to Configure the SGD Gateway for Connections From Tablet Devices Using Untrusted Certificates

Note

You only need to use this procedure if your users are connecting to the SGD Gateway from a tablet computer and the SGD Gateway is using an untrusted SSL certificate, such as a self-signed certificate.

If the Gateway uses an SSL certificate signed by a custom CA, your users must install the CA certificate on the tablet device. See Using Certificates Signed by a Custom CA in the *Oracle Secure Global Desktop Administration Guide*.

1. Export the SGD Gateway SSL certificate.

   Repeat the following steps for each SGD Gateway.

   a. Log in as superuser (root) on the SGD Gateway host.

   b. Regenerate the SGD Gateway SSL certificate.

      i. Generate a self-signed certificate that is compatible with tablet client devices. Use the script provided for this task:

      ```
      # /opt/SUNWsgdg/bin/scripts/regenerate_sslcert.sh --ca
      ```

      The existing SGD Gateway SSL certificate in the client keystore is replaced.

      ii. Restart the SGD Gateway.

      ```
      # /opt/SUNWsgdg/bin/gateway restart
      ```

   c. Export the SGD Gateway SSL certificate from the client keystore. For example:

   ```
   # /opt/SUNWsgdg/bin/gateway sslcert export --certfile gw1-example-com.pem
   ```

   When you export the certificate, it is best practice to rename the certificate file, so that you can identify the SGD Gateway it came from.

2. Copy the SGD Gateway SSL certificates to the SGD array.

   a. Log in as superuser (root) on the primary SGD host.

   b. Copy the certificate files to the `/opt/tarantella/var/tsp/certs/gateway` directory on the primary SGD host.

   c. Check that the file permissions and ownership are correct. For example:

   ```
   # chmod 600 gw1-example-com.pem
   # chown root:ttaserv gw1-example-com.pem
   ```

3. Update the security configuration used for the SGD Gateway.

   Use the script provided for this task:

   ```
   # /opt/tarantella/bin/scripts/mobile_profile_create.sh
   ```

- A configuration profile at `/opt/tarantella/webserver/apache/apache-version/htdocs/certs/sgdg.mobileconfig` is created and updated with details of the SSL certificate used by the SGD Gateway. A corresponding MD5 checksum file is also generated.
- The Gateway SSL certificates are processed and corresponding `.crt` certificate files are generated in the `/opt/tarantella/webserver/apache/apache-version/htdocs/certs/gateway` directory. The `android_certs.html` file which lists the certificates is updated.

4. Copy the updated security configuration files to the other SGD servers in the array.

   Updated security configuration files are in the `certs/` directory.

   Repeat the following step on every server in the array.

   a. Copy the `/opt/tarantella/webserver/apache/apache-version/htdocs/certs` directory from the primary server to the SGD web server.

      Ensure that file permissions and ownerships are preserved. For example:

      ```
      # cp -pr certs/ /opt/tarantella/webserver/apache/apache-version/htdocs/
      ```

See Secure Connections to Tablet Devices Using Untrusted Certificates in the *Oracle Secure Global Desktop Administration Guide* for more details about the required security configuration when using untrusted certificates.

2.2.2. SGD Gateway to SGD Server Connections

The connections between an SGD Gateway and the SGD servers in the array use certificates for mutual authorization. Configuring these connections involves the following configuration tasks:

1. Install SGD server certificates on the SGD Gateway.

   See Section 2.2.2.1, “How to Install SGD Server Certificates”.

2. Install the SGD Gateway certificate on the SGD array.

   See Section 2.2.2.2, “How to Install SGD Gateway Certificates on the SGD Array”.

3. Configure SGD Client connections for the SGD Gateway.

   See Section 2.2.2.3, “How to Configure SGD Client Connections”.

2.2.2.1. How to Install SGD Server Certificates

To use this procedure, the SGD servers in the array must be running in secure mode.

In a standard installation, an SGD server is configured automatically to use secure connections. See “Secure Connections to SGD Servers” in Chapter 1 of the *Oracle Secure Global Desktop Administration Guide* if you need more information about how to enable security services on an SGD server.

Repeat the following procedure for each SGD server in the array.

1. Log in as superuser (root) on the SGD host.
2. Copy the CA certificate from the SGD server to the SGD Gateway keystore directory.

   The CA certificate for an SGD server is at `/opt/tarantella/var/info/certs/PeerCAcert.pem` on the SGD host.

Note

This is the same CA certificate that the SGD server uses for secure intra-array communication.

The SGD Gateway keystore directory is `/opt/SUNWsgdg/proxy/etc`.

When you copy the CA certificate, it is best practice to rename the certificate file so that you can identify what the file contains and the SGD server it came from.

3. Copy the SSL certificate from the SGD server to the SGD Gateway keystore directory.

   The SSL certificate for an SGD server running in secure mode is at `/opt/tarantella/var/tsp/cert.pem` on the SGD host.

   The SGD Gateway keystore directory is `/opt/SUNWsgdg/proxy/etc`.

   When you copy the SSL certificate, it is best practice to rename the certificate file so that you can identify what the file contains and the SGD server it came from.

4. Log in as superuser (root) on the SGD Gateway host.

5. Import the certificates into the SGD Gateway keystore.

```
# /opt/SUNWsgdg/bin/gateway server add --server sgd-server1 \
--certfile /opt/SUNWsgdg/proxy/etc/PeerCAcert.pem --url https://sgd1.example.com \
--ssl-certfile /opt/SUNWsgdg/proxy/etc/cert.pem
```

   The `--server` option defines the alias names used when storing the certificates in the keystore. In this example, the CA certificate is stored using an alias of *`sgd-server1`*, the SSL certificate is stored using an alias of *`sgd-server1-ssl`*.

   `https://sgd1.example.com` is the URL of the SGD web server.

6. Restart the SGD Gateway.

Note

Restarting the SGD Gateway disconnects all user sessions and application sessions that are running through the SGD Gateway.

   On the SGD Gateway host, run the following command:

```
# /opt/SUNWsgdg/bin/gateway restart
```

2.2.2.2. How to Install SGD Gateway Certificates on the SGD Array

Repeat the following procedure for each SGD Gateway.

1. Export the SGD Gateway certificate.

   a. Log in as superuser (root) on the SGD Gateway host.

   b. Export the SGD Gateway certificate from the SGD Gateway keystore.

      Use the `gateway cert export` command, as follows:

```
# /opt/SUNWsgdg/bin/gateway cert export --certfile gateway1.pem
```

The certificate is exported to the file *gateway1.pem*.

When you export the certificate, it is best practice to name the certificate file so that you can identify the SGD Gateway it came from.

c. Copy the certificate to the `/opt/tarantella/var/tsp` directory on the primary SGD server in the array.

d. Change the file permissions and ownership for the Gateway certificate.

```
# chmod 600 /opt/tarantella/var/tsp/gateway1.pem
# chown ttasys:ttaserv /opt/tarantella/var/tsp/gateway1.pem
```

2. Register the SGD Gateway with the SGD array.

a. On the primary SGD server, log in as superuser (root).

b. Import the SGD Gateway certificate.

```
# tarantella gateway add --name sgd-gateway1 \
--certfile /opt/tarantella/var/tsp/gateway1.pem
```

where *sgd-gateway1* is a name used by SGD to identify the SGD Gateway, and *gateway1.pem* is the SGD Gateway certificate file name.

The SGD Gateway certificate is added to the gateway keystore on the SGD server, at `/opt/tarantella/var/info/gatewaycerts`.

To register multiple SGD Gateways at the same time, use the `--file` option of the `tarantella gateway add` command. See Section B.27, “The tarantella gateway Command” for more details.

Configuration changes made using `tarantella gateway add` are replicated to the other SGD servers in the array.

2.2.2.3. How to Configure SGD Client Connections

1. Configure the SGD Client connections that use the SGD Gateway.

On the primary SGD server, set the `--security-gateway` global attribute to define which SGD Clients can use the SGD Gateway, based on their IP address or DNS name.

To specify that all SGD Client connections are routed through TCP port 443 of a single SGD Gateway `gateway1.example.com`, use the following command:

```
$ tarantella config edit --security-gateway \
"*:sgdg:gateway1.example.com:443"
```

To specify that all SGD Client connections are routed through TCP port 443 of an external load balancer `lb.example.com`, use the following command:

```
$ tarantella config edit --security-gateway \
"*:sgdg:lb.example.com:443"
```

Note

Changes to the `--security-gateway` attribute affect all SGD servers in the array. The changes only apply to new user sessions.

See Section B.31, "The --security-gateway Attribute" for more details about how to use the `--security-gateway` attribute to define multiple SGD Client connection filters.

2.2.3. Client Device to Load Balancer Connections

Configuring connections between the client device and an external load balancer involves the following configuration tasks:

1. Configure the load balancer to accept connections from client devices.

   See your load balancer documentation for details of how to do this.

2. (Optional) Install the SSL certificate for the SGD Gateway on to the load balancer.

   See your load balancer documentation for details of how to do this.

2.2.4. Load Balancer to SGD Gateway Connections

Configuring connections between an external load balancer and the SGD Gateway involves the following configuration tasks:

1. Configure the ports and connections used by the SGD Gateway.

   See Section 2.2.1.1, "How to Configure the Ports and Connections for the SGD Gateway".

2. (Optional) On the SGD Gateway, install an SSL certificate for incoming client connections.

   See Section 2.2.1.2, "How to Install an SSL Certificate for Client Connections Into the Client Keystore".

2.3. Controlling the SGD Gateway

This section describes how to control the SGD Gateway. The following tasks are described:

- Starting the SGD Gateway
- Stopping the SGD Gateway
- Restarting the SGD Gateway

2.3.1. Starting the SGD Gateway

To start the SGD Gateway, use the following command:

```
# /opt/SUNWsgdg/bin/gateway start
```

2.3.2. Stopping the SGD Gateway

Caution

Stopping the SGD Gateway disconnects all user sessions and application sessions that are running through the SGD Gateway. This means that application data can be lost if the SGD Gateway is stopped unexpectedly.

To stop the SGD Gateway, use the following command:

```
# /opt/SUNWsgdg/bin/gateway stop
```

When you use the `gateway stop` command a warning message is displayed, prompting you to confirm that you want to stop the SGD Gateway. Use the `--force` option of the `gateway stop` command if you do not want to display this message.

Note

If the SGD Gateway is stopped, users from outside your network cannot connect to SGD using the SGD Gateway. Client devices that have been enabled using the `--security-gateway` attribute to access SGD directly without going through the SGD Gateway, can still access SGD. See Section B.31, “The --security-gateway Attribute”.

2.3.3. Restarting the SGD Gateway

Caution

Restarting the SGD Gateway disconnects all user sessions and application sessions that are running through the SGD Gateway. This means that application data can be lost if the SGD Gateway is restarted unexpectedly.

To restart the SGD Gateway, use the following command:

```
# /opt/SUNWsgdg/bin/gateway restart
```

When you use the `gateway restart` command a warning message is displayed, prompting you to confirm that you want to stop the SGD Gateway. Use the `--force` option of the `gateway restart` command if you do not want to display this message.

2.4. Removing the SGD Gateway

To remove the SGD Gateway, you remove the software installed on the SGD Gateway host.

2.4.1. How To Remove the SGD Gateway

1. Log in as superuser (root) on the SGD Gateway host.
2. Change the SGD Client routing configuration for the SGD array.
   a. Log in as superuser (root) on the primary SGD server.
   b. Edit the `--security-gateway` attribute for the SGD array.

      For a basic deployment using a single SGD Gateway, run the following command:

      ```
      # tarantella config edit --security-gateway ""
      ```

      Note

      For a load-balanced deployment using multiple SGD Gateways and an external load balancer, you do not need to edit the `--security gateway` attribute.

3. Uninstall the SGD Gateway.

   Run the following command:

   ```
   # /opt/SUNWsgdg/bin/gateway uninstall
   ```

   A warning message is displayed, prompting you to confirm that you want to stop the SGD Gateway.

   Caution

   The `gateway uninstall` command is the only supported method of removing the SGD Gateway. Do not use the `pkgrm` or `rpm` commands directly to remove the SGD Gateway.

4. (Optional) Remove the SGD Gateway from the list of SGD Gateways registered for the SGD array.

   a. Display the SGD Gateways registered for the SGD array.

   ```
   # tarantella gateway list
   Installed gateway: gateway1.example.com
   Issuer: CN=gateway1.example.com, OU=Marketing, O=Example, L=Boston,
   ST=Massachusetts, C=US
   Serial Number: 1208509056
   Subject: CN=gateway2.example.com, OU=Marketing, O=Example, L=Boston,
   ST=Massachusetts, C=US
   Valid from Fri Sep 26 09:57:36 GMT 2008 to Thu Dec 25 09:57:36 GMT 2008
   ```

   b. Remove the SGD Gateway from the list of SGD Gateways registered for the SGD array.

   ```
   # tarantella gateway remove --name gateway1.example.com
   ```

Appendix A. SGD Gateway Architecture Overview

This chapter describes the architecture and the main components of the Oracle Secure Global Desktop Gateway (SGD Gateway).

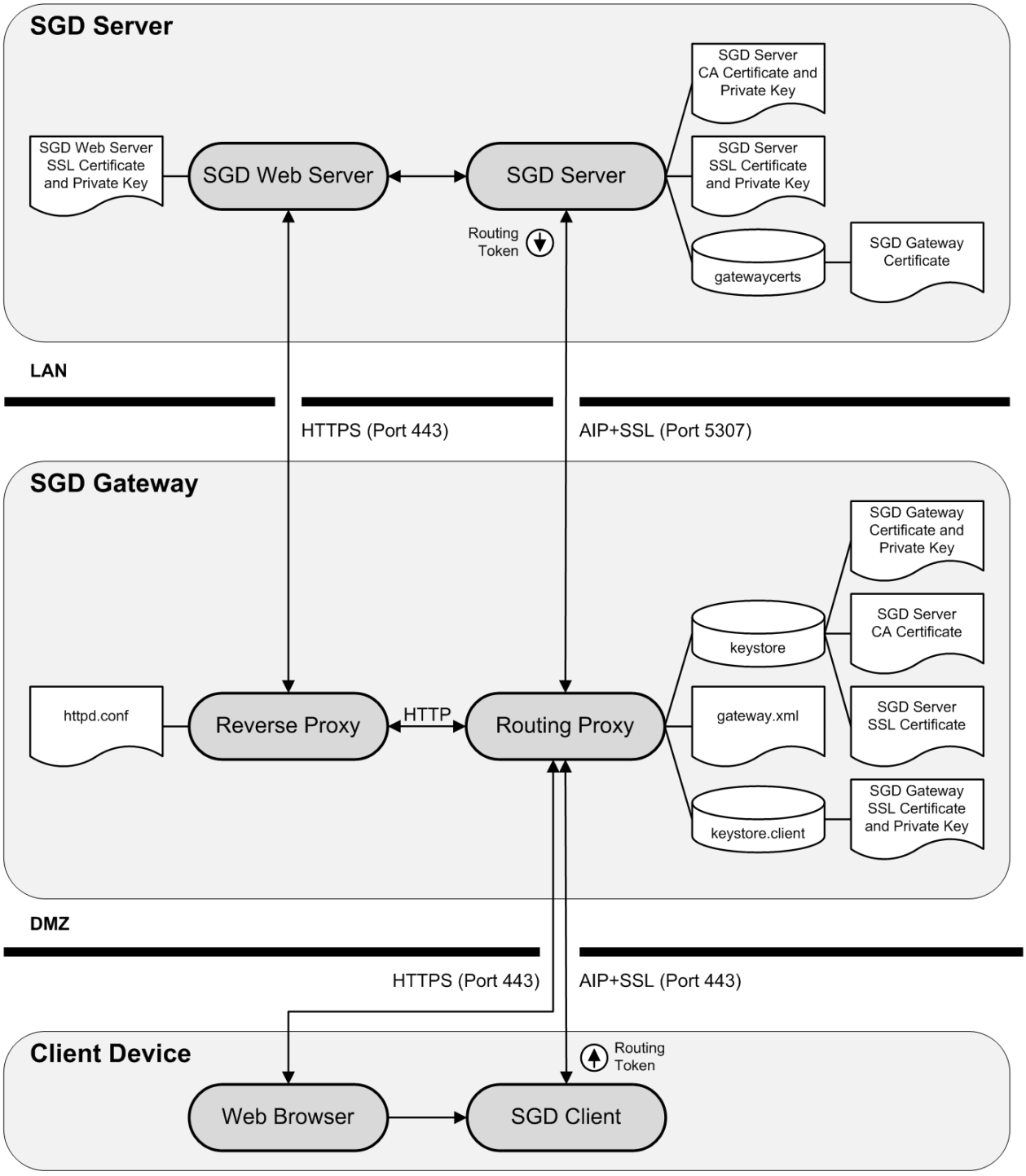
This chapter includes the following topics:

A.1. SGD Gateway Architecture

This section looks at the architecture of the SGD Gateway and includes a description of the connections made when you access SGD through the SGD Gateway.

Figure A.1, “SGD Gateway Architecture” shows the architecture of the SGD Gateway.

Figure A.1. SGD Gateway Architecture



The following steps describe the connections made when you access SGD through the SGD Gateway. The steps cover the initial connection to SGD using a browser, logging on to SGD, through to starting an application.

Note

The steps below describe the connections made when you access the SGD Gateway from a desktop computer.

If you are using the tablet workspace, some minor differences may apply. The SGD Client is not downloaded and installed on the tablet device. Instead, an HTML5 web page is used to manage the connection to the SGD Gateway.

1. A browser on the client device makes an HTTP over Secure Sockets Layer (HTTPS) connection to the SGD Gateway, on TCP port 443.
   - For a basic deployment, users can access SGD by going to the URL of the SGD Gateway.
   - TCP port 443 is the default port for the SGD Gateway. The ports used by the SGD Gateway are defined using the routing proxy configuration file, `gateway.xml`. This file is created automatically during installation of the SGD Gateway, and is updated when the `gateway config` command is used to change the SGD Gateway configuration.
   - The SGD Gateway presents an SSL certificate to the browser on the client device.
   - The location and passwords for the keystores used by the SGD Gateway are defined in the routing proxy configuration file, `gateway.xml`.
2. The routing proxy recognizes an HTTPS connection, decrypts the data stream, and forwards HTTP data to the Apache reverse proxy.
   - HTTP data is sent internally on TCP port 8081.
   - The configuration for the Apache reverse proxy is defined by the `httpd.conf` file. This file and related reverse proxy configuration files are created automatically during installation of the SGD Gateway. The files are updated when the `gateway config` command is used to change the SGD Gateway configuration.
3. The reverse proxy uses HTTP load balancing to select an SGD web server in the array.
   - Connections between the reverse proxy and the SGD web server are secure, using HTTPS on TCP port 443.
   - The Apache reverse proxy sets a load balancing cookie in the browser. All subsequent HTTP requests by the browser use the same SGD web server.
4. HTML from the SGD web server is routed to the browser on the client device.
   - The HTML is sent as HTTPS data on the connection established to TCP port 443 on the SGD Gateway.
   - The SGD Gateway forwards the HTTPS data to the browser.
5. The user logs in to SGD.
   - The SGD server authenticates the user and starts a new user session.
   - The SGD Client is downloaded, installed, and started on the client device.
   - A routing token is included in HTML sent to the browser. The routing token contains the address of the SGD server which manages the user session. This information is used to route Adaptive Internet Protocol (AIP) data to the correct SGD server.

- The routing token is signed using the private key for the SGD server's CA certificate, and then encrypted using the SGD Gateway certificate on the SGD server.
- The routing token is passed to the SGD Client.
- Connections to the client device use HTTPS.

6. The SGD Client connects to the SGD Gateway on TCP port 443.
   - The data connection between the SGD Client and the SGD Gateway uses AIP over Secure Sockets Layer (SSL).
   - The SSL certificate for the SGD Gateway is presented on the connection.
   - The routing proxy recognizes incoming AIP data over SSL.
   - The SSL data stream is decrypted, and the routing token is extracted from the AIP data stream.
   - The routing token is decrypted, using the SGD Gateway private key and then verified, using the CA certificate for the SGD server.
   - The SGD Gateway private key and the CA certificate for the SGD server are stored in the SGD Gateway keystore, `keystore`.
   - The time stamp on the routing token is checked, to ensure the routing token is valid.
   - The AIP data stream is re-encrypted using SSL.
7. AIP over SSL data is routed through the routing proxy to the SGD server indicated by the routing token.
   - The AIP over SSL data connection uses TCP port 5307.
   - The routing token is not included with the AIP data stream.
8. The user starts an application on the SGD workspace.
   - The application launch request is sent to the SGD Gateway using HTTPS.
   - The routing proxy recognises and decrypts HTTPS data, and forwards HTTP traffic to the Apache reverse proxy.
   - The reverse proxy detects the load balancing cookie and uses the SGD web server indicated by the cookie.
   - SGD application session load balancing selects an SGD server to manage the application session.
   - A new routing token is created on the SGD server. The routing token is used to route AIP data to the SGD server selected to manage the application session.
   - The SGD server sends the routing token to the SGD Client. The routing token is included with the existing AIP data stream.
9. The SGD Client connects to the SGD Gateway on TCP port 443.
   - The SSL certificate for the SGD Gateway is presented for the connection.
   - The routing proxy recognizes incoming AIP over SSL data.

- The routing token is decrypted, verified, and validated.
- AIP over SSL data is routed through the routing proxy to the SGD server indicated by the routing token.
- The routing token is not included with the AIP data stream.

10. The SGD server manages the application session.
    - The application runs on an application server located on the local area network (LAN).

A.2. Components of the SGD Gateway

The SGD Gateway consists of the following components:

- **Routing proxy**. A Java technology-based application that routes AIP data connections to an SGD server.

  The main components of the routing proxy are:

  - Routing tokens – See Section A.2.1, “About Routing Tokens”
  - Keystores – See Section A.2.2, “Keystores Used By the SGD Gateway”
  - Routing proxy configuration file – See Section A.2.3, “Routing Proxy Configuration File”

- **Reverse proxy.** An Apache web server, configured to operate in reverse proxy mode. The reverse proxy also performs load balancing of HTTP connections.

  The main components of the reverse proxy are:

  - Configuration files for the Apache web server – See Section A.2.4, “Apache Web Server Configuration Files”
  - Apache modules for reverse proxying and HTTP load balancing – See Section A.2.5, “Apache Modules Used by the SGD Gateway”

A.2.1. About Routing Tokens

The SGD Gateway uses a *routing token* to manage an AIP connection. A routing token is a signed, encrypted message which identifies the origin and destination SGD server for a route. The routing token includes a time stamp, which is used to limit the token lifetime.

Outgoing routing tokens are:

- Signed on the SGD server, using the private key for the SGD server's CA certificate.
- Encrypted on the SGD server, using the SGD Gateway certificate.
- Sent to the SGD Client on the client device.

Incoming routing tokens are:

- Decrypted on the SGD Gateway, using the SGD Gateway private key.
- Verified on the SGD Gateway, using the CA certificate for the origin SGD server.

- Discarded on the SGD Gateway. The connection presenting the routing token is routed to the destination SGD server.

A.2.2. Keystores Used By the SGD Gateway

The SGD Gateway uses private keys and certificates to digitally sign and verify routing tokens, to secure connections to the SGD servers in the array, to secure client connections to the SGD Gateway, and to authorize access to the reflection service.

The certificates and private keys used by the SGD Gateway are stored in keystores in the `/opt/SUNWsgdg/proxy/etc` directory.

This directory contains the following keystores:

- **SGD Gateway keystore.** The SGD Gateway keystore, `keystore`, contains the SGD Gateway certificate and private key, CA certificates for the SGD servers in the array, and SGD server SSL certificates for secure connections to the SGD servers in the array.

  To add, remove, and list entries for the SGD Gateway keystore, use the `gateway` command.

- **Client keystore.** The client keystore, `keystore.client`, contains a single SGD Gateway SSL certificate and private key used for securing connections between the client device and the SGD Gateway. By default, this keystore contains a self-signed certificate. You can replace this certificate with a certificate signed by a Certificate Authority (CA).

- **Reflection service keystore.** The reflection service keystore, `keystore.reflection`, contains a certificate and private key used to authorize access to the reflection service on the SGD Gateway. By default, this keystore contains a self-signed certificate and private key.

The keystores are created automatically when you run the `gateway setup` command after installing the SGD Gateway.

Note

All keystores use the same password, which is defined in the `/opt/SUNWsgdg/etc/password` file. The password is a random password created automatically when the keystores are first created. The password file is only readable by superuser (root).

A.2.3. Routing Proxy Configuration File

The routing proxy configuration file is `/opt/SUNWsgdg/etc/gateway.xml`. This is an XML file that configures routes, depending on the data protocol type. The file also configures the keystore locations and passwords required for routing and SSL protocols.

The routing proxy configuration file is created automatically when you install the SGD Gateway and is updated when you use the `gateway config` commands to change the configuration of the SGD Gateway.

Caution

Use the `gateway config` commands to configure the Gateway. If possible, avoid editing the `gateway.xml` file manually. Incorrect configuration in the `gateway.xml` file might cause the SGD Gateway to stop working.

The default routing proxy configuration file uses the password in the `/opt/SUNWsgdg/etc/password` file to access the keystores used by the SGD Gateway. If you do not want to store this password on disk,

make a note of the entry in the password file. Delete the password file, and delete the `password` entries for all `<keystore>` elements in the `gateway.xml` file. You are then prompted for the keystore password when you next start the SGD Gateway.

To change the password for a keystore used by the SGD Gateway, use the `-storepasswd` option of the `keytool` command. For example, to change the password for the `keystore.client` keystore run the following command:

```
# /opt/SUNWsgdg/java/default/bin/keytool -storepasswd \
-keystore /opt/SUNWsgdg/proxy/etc/keystore.client
```

See the JDK Tools and Utilities documentation for details on how to use the `keytool` application.

Note

The `/opt/SUNWsgdg/etc` directory also contains other `.xml` and `.template` files. These files are used internally by the `gateway config` command to update the `gateway.xml` file. Do not edit these files manually.

A.2.4. Apache Web Server Configuration Files

Configuration files for the Apache web server configured for use with the SGD Gateway are in the `/opt/SUNWsgdg/httpd/apache-version/conf` directory.

The configuration files in this directory are used to configure reverse proxy operation and load balancing for the Apache web server.

A.2.4.1. Configuring Reverse Proxying and Load Balancing

Files for configuring reverse proxy operation and load balancing are in the `extra/gateway` subdirectory. These files are enabled by the following `Include` directive in the main `httpd.conf` file:

```
# SGD Reverse Proxy/Load Balance settings
Include conf/extra/gateway/httpd-gateway.conf
```

The `httpd-gateway.conf` file configures reverse proxying and load balancing for the Apache web server. The members of the load balancing group are defined using an `Include` directive in the `httpd-gateway.conf` file, as follows:

```
<Proxy Balancer://mysgdservers/>
Include conf/extra/gateway/servers/*.conf
</Proxy>
```

The `extra/gateway/servers` directory contains configuration files for each of the SGD web servers in the load balancing group. The configuration files are named *`server-name`*`.conf`, where *`server-name`* is the server name used in the `gateway server add` command. See Section B.12, “gateway server add” for more details about this command.

The SGD Gateway uses *sticky session* HTTP load balancing. This means that the Apache reverse proxy sets a cookie in the client browser, to ensure that the browser always returns to the SGD web server that was selected by load balancing. The cookie expires at the end of the user session.

Sticky session cookies are enabled by the `Header add Set-Cookie` directive in the `httpd-gateway.conf` file, as follows:

```
Header add Set-Cookie "BALANCEID=balanceworker.%{BALANCER_WORKER_ROUTE}e; path=/" \
env=BALANCER_ROUTE_CHANGED
```

where `BALANCEID` is the name of the cookie, and `BALANCER_WORKER_ROUTE` and `BALANCER_ROUTE_CHANGED` are environment variables exported by the Apache `mod_proxy_balancer` module. See the Apache mod_proxy_balancer documentation for more information about these environment variables.

A.2.5. Apache Modules Used by the SGD Gateway

The Apache web server supplied with the SGD Gateway uses the standard Apache modules for reverse proxying and load balancing. The modules are installed as Dynamic Shared Object (DSO) modules.

The modules are enabled by `LoadModule` directives in the `httpd.conf` Apache configuration file, at `/opt/SUNWsgdg/httpd/apache-version/conf/httpd.conf`.

Appendix B. Command-Line Reference

This chapter describes how you can manage, control, and change the configuration for the Oracle Secure Global Desktop Gateway (SGD Gateway) from the command line.

Commands are provided for tasks such as setting up keystores and certificates, configuring the ports used by the SGD Gateway, and configuring load balancing for the SGD servers in the array.

This chapter includes the following topics:

B.1. The gateway Command

Use the `gateway` command to configure and control the SGD Gateway.

Note

The full path of the `gateway` command is `/opt/SUNWsgdg/bin/gateway`.

Syntax

```
gateway start | stop | restart | config | server | status | setup | version | sslcert |
sslkey | cert | key | setup | uninstall
```

Description

The available `gateway` commands are shown in the following table.

Command	Description	More Information
`gateway start`	Starts the SGD Gateway	Section B.22, “gateway start”
`gateway stop`	Stops the SGD Gateway	Section B.24, “gateway stop”
`gateway restart`	Stops and then restarts the SGD Gateway	Section B.10, “gateway restart”
`gateway config`	Configures the SGD Gateway, and updates the Apache reverse proxy configuration files	Section B.3, “gateway config”
`gateway server`	Installs SGD server security certificates and configures load balancing for the SGD array	Section B.11, “gateway server”
`gateway status`	Displays the current status for the SGD Gateway	Section B.23, “gateway status”
`gateway version`	Displays the version number of the SGD Gateway	Section B.26, “gateway version”
`gateway sslcert`	Exports and prints the Secure Sockets Layer (SSL) certificate in the client keystore	Section B.16, “gateway sslcert”
`gateway sslkey`	Manages the private key and certificate in the client keystore	Section B.19, “gateway sslkey”

Command	Description	More Information
`gateway cert export`	Exports the SGD Gateway certificate from the SGD Gateway keystore	Section B.2, “gateway cert export”
`gateway key import`	Imports a private key and certificate into the SGD Gateway keystore	Section B.9, “gateway key import”
`gateway setup`	Runs the SGD Gateway setup program	Section B.15, “gateway setup”
`gateway uninstall`	Uninstalls the SGD Gateway software	Section B.25, “gateway uninstall”

Note

All `gateway` commands include a `--help` option. You can use this option to display help for the command.

Examples

The following example starts the SGD Gateway.

```
# /opt/SUNWsgdg/bin/gateway start
```

The following example means that the SGD server `server.example.com` is not authorized to use the SGD Gateway.

```
# /opt/SUNWsgdg/bin/gateway server remove --server server.example.com
```

B.2. gateway cert export

Exports the SGD Gateway certificate from the SGD Gateway keystore.

Syntax

```
gateway cert export --certfile file-name
```

Description

Exports the SGD Gateway certificate from the SGD Gateway keystore, at `/opt/SUNWsgdg/proxy/etc/keystore`. The certificate is written to the file specified by the `--certfile` option.

To access the SGD Gateway keystore, this command uses the password in `/opt/SUNWsgdg/etc/password`. If this file is not present, the command prompts for a password.

Examples

The following example exports the SGD Gateway certificate from the SGD Gateway keystore to the file, `gateway1.pem`.

```
# /opt/SUNWsgdg/bin/gateway cert export --certfile gateway1.pem
```

B.3. gateway config

Configures the SGD Gateway. The `gateway config` command configures secure connections, ports, and reverse proxy server settings for the SGD Gateway.

Syntax

```
gateway config create | show
```

Description

The following table shows the available subcommands for this command.

Subcommand	Description	More Information
`create`	Creates a new configuration for the SGD Gateway	Section B.4, “gateway config create”
`list`	Lists the current configuration for the SGD Gateway	Section B.8, “gateway config list”
`edit`	Edits the current configuration for the SGD Gateway	Section B.6, “gateway config edit”
`enable`	Enables an SGD Gateway service	Section B.7, “gateway config enable”
`disable`	Disables an SGD Gateway service	Section B.5, “gateway config disable”

Examples

The following example lists the current configuration for the SGD Gateway.

```
# /opt/SUNWsgdg/bin/gateway config list
```

B.4. gateway config create

Creates a new configuration for the SGD Gateway, overwriting the current configuration.

Syntax

```
gateway config create { [ --interface interface:port ]
                        [ --entry-point ip-address:port ]
                        [ --out plaintext | ssl ]
                      } | --file file
```

Description

The following table shows the available options for this command.

Option	Description
`--interface`	Interface and port that the SGD Gateway listens on for incoming proxy connections. The default is TTCP port 443, on all interfaces.
`--entry-point`	Entry point for the network. This is the IP address, and port that clients use to connect to the SGD Gateway. You can specify a Domain Name System (DNS) address instead of an IP address.
`--out`	Format of outgoing traffic from the SGD Gateway to the SGD servers in the array. If you are using secure connections, choose `ssl`.

Option	Description
`--file`	Specifies a file containing configuration settings.

Note

If no options are specified for the `gateway config create` command, a series of online prompts are displayed, enabling you to type in the required settings.

If you use the `--file` option for `gateway config create`, the specified file must be of the same format as the `/opt/SUNWsgdg/etc/gatewayconfig.xml` file. This file is created during initial configuration of the SGD Gateway, as described in Section 2.2.1.1, “How to Configure the Ports and Connections for the SGD Gateway”.

Examples

The following example configures an SGD Gateway to listen on TCP port 443 for connections from the network entry point, at 192.168.0.1. Secure connections are used between the SGD Gateway and the SGD servers in the array.

```
# /opt/SUNWsgdg/bin/gateway config create --interface *:443 \
--entry-point 192.168.0.1:443 --out ssl
```

B.5. gateway config disable

Disables one or more SGD Gateway services.

Syntax

```
gateway config disable [ --services-reflection ]
                       [ --services-reflection-auth ]
                       [ --routes-http-redirect ]
```

Description

Use the command line options to disable specific SGD Gateway services. You must specify at least one command-line option.

Note

After using this command to disable a service, you must restart the SGD Gateway to stop the service.

The following table shows the available options for this command.

Option	Description
`--services-reflection`	Disables unauthenticated access to the SGD Gateway reflection service. By default, this service is disabled. See Section C.10, “The Reflection Service” for more details about the SGD Gateway reflection service.

Option	Description
`--services-reflection-auth`	Disables authenticated access to the SGD Gateway reflection service. By default, this service is disabled. See Section C.10, "The Reflection Service" for more details about the SGD Gateway reflection service.
`--routes-http-redirect`	Disables the HTTP redirection service. By default, this service is disabled.

Examples

The following example disables authenticated access to the SGD Gateway reflection service.

```
# /opt/SUNWsgdg/bin/gateway config disable --services-reflection-auth
```

B.6. gateway config edit

Edits the current SGD Gateway configuration.

Syntax

```
gateway config edit [ --binding int:port ]
                    [ --routes-http-maxcon num ]
                    [ --routes-aip-maxcon num ]
                    [ --routes-ws-maxcon num ]
                    [ --routes-reverseproxy-redirect port ]
                    [ --services-reflection-binding int:port ]
                    [ --services-reflection-auth-binding int:port ]
```

Description

The command-line options enable you to edit specific configuration settings. You must specify at least one command-line option.

The current SGD Gateway configuration is stored in the `/opt/SUNWsgdg/etc/gatewayconfig.xml` file.

You must restart the SGD Gateway to enable any configuration changes you make.

The following table shows the available options for this command.

Option	Description
`--binding`	Interface and port that the SGD Gateway listens on for incoming proxy connections. The default is TCP port 443, on all interfaces.
`--routes-http-maxcon`	Maximum number of HTTP connections. The default value is configured at install time and depends on the memory resources available on the SGD Gateway. See Section C.1, "Tuning the SGD Gateway".

Option	Description
`--routes-aip-maxcon`	Maximum number of AIP connections. The default value is configured at install time and depends on the memory resources available on the SGD Gateway. See Section C.1, “Tuning the SGD Gateway”.
`--routes-ws-maxcon`	Maximum number of websocket connections. Websocket connections are used for data connections between the SGD Gateway and tablet devices. The default value is configured at install time and depends on the memory resources available on the SGD Gateway. See Section C.1, “Tuning the SGD Gateway”.
`--routes-reverseproxy-redirect`	HTTP redirection port. The default is TCP port 8080.
`--services-reflection-binding`	Interface and port used for unauthenticated access to the SGD Gateway reflection service. The default is TCP port 81 on the localhost loopback interface.
`--services-reflection-auth-binding`	Interface and port used for authenticated access to the SGD Gateway reflection service. The default is TCP port 82 on all interfaces.

Examples

The following example changes the maximum number of HTTP and AIP connections for the SGD Gateway.

```
# /opt/SUNWsgdg/bin/gateway config edit --routes-http-maxcon 200
# /opt/SUNWsgdg/bin/gateway config edit --routes-aip-maxcon 3000
```

B.7. gateway config enable

Enables one or more SGD Gateway services.

Syntax

```
gateway config enable [ --services-reflection ]
                      [ --services-reflection-auth ]
                      [ --routes-http-redirect ]
```

Description

Use the command line options to enable specific SGD Gateway services. You must specify at least one command-line option.

Note

After using this command to enable a service, you must restart the SGD Gateway to start the service.

The following table shows the available options for this command.

Option	Description
`--services-reflection`	Enables unauthenticated access to the SGD Gateway reflection service.

Option	Description
	By default, this service is disabled. See Section C.10, “The Reflection Service” for more details about the SGD Gateway reflection service.
`--services-reflection-auth`	Enables authenticated access to the SGD Gateway reflection service. By default, this service is disabled. See Section C.10, “The Reflection Service” for more details about the SGD Gateway reflection service.
`--routes-http-redirect`	Enables the HTTP redirection service. By default, this service is disabled.

Examples

The following example enables authenticated access to the SGD Gateway reflection service.

```
# /opt/SUNWsgdg/bin/gateway config enable --services-reflection-auth
```

B.8. gateway config list

Lists the current SGD Gateway configuration.

Syntax

```
gateway config list [ --binding ]
                    [ --routes-http-maxcon ]
                    [ --routes-aip-maxcon ]
                    [ --routes-ws-maxcon ]
                    [ --routes-reverseproxy-redirect ]
                    [ --services-reflection-binding ]
                    [ --services-reflection-auth-binding ]
```

Description

The command-line options enable you to list specific configuration settings. If no options are specified, the full configuration details for the SGD Gateway are displayed.

The current SGD Gateway configuration is stored in the `/opt/SUNWsgdg/etc/gatewayconfig.xml` file.

The following table shows the available options for this command.

Option	Description
`--binding`	Interface and port that the SGD Gateway listens on for incoming proxy connections
`--routes-http-maxcon`	Maximum number of HTTP connections
`--routes-aip-maxcon`	Maximum number of Adaptive Internet Protocol (AIP) connections

Option	Description
`--routes-ws-maxcon`	Maximum number of websocket connections. Websocket connections are used for data connections between the SGD Gateway and tablet devices.
`--routes-reverseproxy-redirect`	HTTP redirection port
`--services-reflection-binding`	Interface and port used for unauthenticated access to the SGD Gateway reflection service
`--services-reflection-auth-binding`	Interface and port used for authenticated access to the SGD Gateway reflection service

Examples

The following example shows binding configuration and the maximum number of AIP connections for the SGD Gateway.

```
# /opt/SUNWsgdg/bin/gateway config list --binding --routes-aip-maxcon
binding: *:443
routes-aip-maxcon: 2920
```

The following example shows full details for the current SGD Gateway configuration.

```
# /opt/SUNWsgdg/bin/gateway config list
binding: *:443
routes-http-maxcon: 100
routes-aip-maxcon: 2920
routes-ws-maxcon: 512
routes-reverseproxy-redirect: null
services-reflection-binding: localhost:81
services-reflection-auth-binding: *:82
```

B.9. gateway key import

Imports an SGD Gateway key and SGD Gateway certificate into the SGD Gateway keystore.

Syntax

```
gateway key import --keyfile key-file
                [ --keyalg RSA|DSA ]
                { --certfile cert-file |
                  --certfile cert-file.. [ --cacertfile ca-cert-file ] }
                [ --alwaysoverwrite ]
```

Description

Imports a private key, and the corresponding public key certificate, into the SGD Gateway keystore, at `/opt/SUNWsgdg/proxy/etc/keystore`.

If the keystore already has an SGD Gateway key entry, it is overwritten. By default, a confirmation prompt is shown.

To access the SGD Gateway keystore, this command uses the password in `/opt/SUNWsgdg/etc/password`. If this file is not present, the command prompts for a password.

The following table shows the available options for this command.

Option	Description
`--keyfile`	File containing the private key. The key must be in PEM format.
`--keyalg`	Encoding algorithm used by the private key. Options are RSA and DSA. By default, RSA is selected.
`--certfile`	SSL certificate file.
`--cacertfile`	CA or root certificate file.
`--alwaysoverwrite`	Do not prompt before overwriting an entry in the keystore.

To import a certificate chain, use the `--cacertfile` option to specify an Intermediate CA certificate. All certificates in the chain must be in PEM format.

If a certificate chain uses multiple CA certificates, combine all the CA certificates in the chain into a single file. The CA certificate used to sign the server certificate *must appear first*, for example:

```
-----BEGIN CERTIFICATE-----
...Intermediate CA's certificate...
-----END CERTIFICATE-----
-----BEGIN CERTIFICATE-----
...CA root certificate...
-----END CERTIFICATE-----
```

Examples

The following example imports an RSA-encoded private key `gateway1.key`, and the corresponding public key certificate `gateway1.pem`, into the SGD Gateway keystore.

```
# /opt/SUNWsgdg/bin/gateway key import \
--keyfile gateway1.key \
--certfile gateway1.pem
```

The following example imports a private key and a certificate chain into the SGD Gateway keystore. The Intermediate CA certificate is `gateway1-ca.pem`.

```
# /opt/SUNWsgdg/bin/gateway key import \
--keyfile gateway1.key \
--certfile gateway1.pem \
--cacertfile gateway1-ca.pem
```

B.10. gateway restart

Stops and then restarts the SGD Gateway.

Syntax

```
gateway restart [--force]
```

Description

Stops and then restarts the SGD Gateway. Before stopping the SGD Gateway, the user is prompted for confirmation.

The `--force` option stops the SGD Gateway, without asking for confirmation.

Examples

The following example stops and restarts the SGD Gateway, prompting the user for confirmation.

```
# /opt/SUNWsgdg/bin/gateway restart
```

B.11. gateway server

Authorizes SGD servers to use the SGD Gateway.

Syntax

```
gateway server add | remove | list
```

Description

The following table shows the available subcommands for this command.

Subcommand	Description	More Information
`add`	Authorizes an SGD server to use the SGD Gateway	Section B.12, “gateway server add”
`remove`	Removes authorization for an SGD server to use the SGD Gateway	Section B.14, “gateway server remove”
`list`	Lists the SGD servers authorized to use the SGD Gateway	Section B.13, “gateway server list”

Examples

The following example removes authorization to use the SGD Gateway for the SGD server `sgd.example.com`.

```
# /opt/SUNWsgdg/bin/gateway server remove --server sgd.example.com
```

B.12. gateway server add

Authorizes an SGD server to use the SGD Gateway.

Syntax

```
gateway server add --server server-name
                   --certfile cert-file
                   --url server-url
                 [ --ssl-certfile ssl-cert ]
```

Description

The following table shows the available options for this command.

Option	Description
`--server`	DNS name of the SGD server
`--cert-file`	Certificate Authority (CA) certificate for the SGD server

Option	Description
`--url`	URL for the SGD web server
`--ssl-certfile`	SSL certificate for the SGD server

The `gateway server add` command does the following:

- Imports the CA certificate for the SGD server into the SGD Gateway keystore, at `/opt/SUNWsgdg/proxy/etc/keystore`. The CA certificate is stored to the keystore using an alias with the same name as the SGD server specified by the `--server` option.
- Imports the SSL certificate for the SGD server into the SGD Gateway keystore, at `/opt/SUNWsgdg/proxy/etc/keystore`. The SSL certificate is stored to the keystore using an alias constructed by appending `-ssl` to the SGD server name specified by the `--server` option.
- Adds the SGD server to the load balancing group used by the Apache reverse proxy server

Note

After using `gateway server add`, you must restart the SGD Gateway for any changes to take effect.

Examples

The following example adds the CA certificate `PeerCAcert.pem` to the SGD Gateway keystore, using the alias `sgd.example.com`. The SSL certificate `cert.pem` is also added to the keystore, using the alias `sgd.example.com-ssl`.

```
# /opt/SUNWsgdg/bin/gateway server add --server sgd.example.com \
--certfile PeerCAcert.pem \
--url https://sgd.example.com \
--ssl-certfile cert.pem
```

In this example, the URL for the SGD web server, `https://sgd.example.com`, is added to the reverse proxy load balancing group and a configuration file is created at `/opt/SUNWsgdg/httpd/apache-version/conf/extra/gateway/servers/conf/sgd.example.com.conf`.

B.13. gateway server list

Shows details for the SGD servers authorized to use the SGD Gateway.

Syntax

```
gateway server list
```

Description

This command shows certificate details and URLs for the SGD servers that are authorized to use the SGD Gateway.

Examples

The following example lists details of the authorized SGD servers for the SGD Gateway.

```
# /opt/SUNWsgdg/bin/gateway server list
```

B.14. gateway server remove

Removes authorization for an SGD server to use the SGD Gateway.

Syntax

```
gateway server remove --server server-name
```

Description

The CA certificate and SSL certificate for the SGD server are removed from the SGD Gateway keystore.

Note

After using `gateway server remove`, you must restart the SGD Gateway for any changes to take effect.

Examples

The following example removes authorization for the SGD server `sgd.example.com` to use the SGD Gateway.

```
# /opt/SUNWsgdg/bin/gateway server remove --server sgd.example.com
```

B.15. gateway setup

Runs the setup program for the SGD Gateway.

Syntax

```
gateway setup
```

Description

Answer the on-screen questions to configure ports, interfaces, and security settings used by the SGD Gateway.

Examples

The following example runs the SGD Gateway setup program.

```
# /opt/SUNWsgdg/bin/gateway setup
```

B.16. gateway sslcert

Print or exports the SGD Gateway SSL certificate stored in the client keystore.

Syntax

```
gateway sslcert export | print
```

Description

The following table shows the available subcommands for this command.

Subcommand	Description	More Information
`export`	Exports the SGD Gateway SSL certificate from the client keystore	Section B.17, “gateway sslcert export”
`print`	Prints the SGD Gateway SSL certificate stored in the client keystore	Section B.18, “gateway sslcert print”

Examples

The following example prints the SGD Gateway SSL certificate stored in the client keystore.

```
# /opt/SUNWsgdg/bin/gateway sslcert print
```

B.17. gateway sslcert export

Exports the SGD Gateway SSL certificate from the client keystore.

Syntax

```
gateway sslcert export --certfile cert-file
```

Description

Exports the SGD Gateway SSL certificate from the client keystore, at `/opt/SUNWsgdg/proxy/etc/keystore.client`. The certificate is written to the file specified by the `--certfile` option.

To access the client keystore, this command uses the password in `/opt/SUNWsgdg/etc/password`. If this file is not present, the command prompts for a password.

Examples

The following example exports the SGD Gateway SSL certificate from the client keystore to the file, `gateway-ssl.pem`.

```
# /opt/SUNWsgdg/bin/gateway sslcert export --certfile gateway-ssl.pem
```

B.18. gateway sslcert print

Prints the SGD Gateway SSL certificate.

Syntax

```
gateway sslcert print
```

Description

Prints the SGD Gateway SSL certificate stored in the client keystore, at `/opt/SUNWsgdg/proxy/etc/keystore.client`.

The command writes details of the certificate to the terminal window.

To access the client keystore, this command uses the password in `/opt/SUNWsgdg/etc/password`. If this file is not present, the command prompts for a password.

Examples

The following example prints the SGD Gateway SSL certificate stored in the client keystore.

```
# /opt/SUNWsgdg/bin/gateway sslcert print
```

B.19. gateway sslkey

Manages SSL key and certificate entries in the client keystore.

Syntax

```
gateway sslkey import | export
```

Description

The following table shows the available subcommands for this command.

Subcommand	Description	More Information
`import`	Imports a private key and certificate into the client keystore	Section B.21, “gateway sslkey import”
`export`	Exports a private key from the client keystore	Section B.20, “gateway sslkey export”

Examples

The following example exports the SGD Gateway SSL certificate stored in the client keystore.

```
# /opt/SUNWsgdg/bin/gateway sslkey export --keyfile gateway-ssl.key
```

B.20. gateway sslkey export

Exports the SGD Gateway SSL private key from the client keystore.

Syntax

```
gateway sslkey export --keyfile key-file [ --keypass passwd ]
```

Description

Exports the SGD Gateway SSL private key from the client keystore, at `/opt/SUNWsgdg/proxy/etc/keystore.client`. The private key is written to the file specified by the `--keyfile` option.

A password for the private key can be specified using the `--keypass` option. By default, the password from `/opt/SUNWsgdg/etc/password` is used.

Examples

The following example exports the SGD Gateway SSL private key from the client keystore to the file, `gateway-ssl.key`.

```
# /opt/SUNWsgdg/bin/gateway sslkey export --keyfile gateway-ssl.key
```

B.21. gateway sslkey import

Imports an SSL key and certificate into the client keystore.

Syntax

```
gateway sslkey import --keyfile key-file
                    [ --keypass ]
                    [ --keyalg RSA|DSA ]
                    { --certfile cert-file |
                      --certfile cert-file.. [ --cacertfile ca-cert-file ] }
                    [ --alwaysoverwrite ]
```

Description

Imports an SSL private key, and the corresponding SSL certificate, into the client keystore, at `/opt/SUNWsgdg/proxy/etc/keystore.client`. By default, this keystore contains a single self-signed certificate.

If the client keystore already has an entry, this command overwrites it. By default, a confirmation prompt is shown before overwriting the keystore entry.

To access the client keystore, this command uses the password in `/opt/SUNWsgdg/etc/password`. If this file is not present, the command prompts for a password.

Specifying `--keypass` as an argument prompts the user for the password for the private key. This option must be used if the private key is password-protected.

The following table shows the available options for this command.

Option	Description
`--keyfile`	File containing the SSL private key. The key must be in Privacy Enhanced Mail (PEM) format.
`--keypass`	Prompt for the password for the SSL private key. Use this option if the private key is password-protected.
`--keyalg`	Encoding algorithm used by the private key. Options are RSA and Digital Signature Algorithm (DSA). By default, RSA is selected.
`--certfile`	SSL certificate file.
`--cacertfile`	CA certificate or root certificate file.
`--alwaysoverwrite`	Do not prompt before overwriting the entry in the client keystore.

To import a certificate chain, use the `--cacertfile` option to specify the Intermediate CA certificate. All certificates in the chain must be in PEM format.

If a certificate chain uses multiple CA certificates, combine all the CA certificates in the chain into a single file. The CA certificate used to sign the server certificate *must appear first*, for example:

```
-----BEGIN CERTIFICATE-----
...Intermediate CA's certificate...
-----END CERTIFICATE-----
-----BEGIN CERTIFICATE-----
...CA root certificate...
-----END CERTIFICATE-----
```

Examples

The following example imports an RSA-encoded SSL private key `gateway1-ssl.key`, and the corresponding SSL certificate `gateway1-ssl.pem`, into the client keystore. The user is prompted for the password for the private key.

```
# /opt/SUNWsgdg/bin/gateway sslkey import \
--keyfile gateway1-ssl.key \
--certfile gateway1-ssl.pem \
--keypass
```

The following example imports an RSA-encoded SSL private key and an SSL certificate chain into the client keystore. The Intermediate CA certificate is `gateway1-ca.pem`.

```
# /opt/SUNWsgdg/bin/gateway sslkey import \
--keyfile gateway1-ssl.key \
--certfile gateway1-ssl.pem \
--cacertfile gateway1-ca.pem
```

B.22. gateway start

Starts the SGD Gateway.

Syntax

```
gateway start
```

Description

Starts the SGD Gateway.

Examples

The following example starts the SGD Gateway.

```
# /opt/SUNWsgdg/bin/gateway start
SGD Gateway started successfully
```

B.23. gateway status

Displays the current status of the SGD Gateway.

Syntax

```
gateway status
```

Description

This command indicates if the SGD Gateway is started, stopped, or if there is a problem.

Examples

The following example displays status information for the SGD Gateway. In this example, the SGD Gateway is stopped.

```
# /opt/SUNWsgdg/bin/gateway status
SGD Gateway status: STOPPED
```

B.24. gateway stop

Stops the SGD Gateway.

Syntax

```
gateway stop [--force]
```

Description

Stops the SGD Gateway, after prompting the user for confirmation.

The `--force` option stops the SGD Gateway, without asking for confirmation.

Examples

The following example stops the SGD Gateway, prompting the user for confirmation.

```
# /opt/SUNWsgdg/bin/gateway stop
```

B.25. gateway uninstall

Uninstalls the SGD Gateway software.

Syntax

```
gateway uninstall
```

Description

Stops the SGD Gateway and removes the SGD Gateway software, including all configuration information.

Before stopping the SGD Gateway, the command prompts the user for confirmation.

Examples

The following example uninstalls the SGD Gateway software from the host where the command is run.

```
# /opt/SUNWsgdg/bin/gateway uninstall
```

B.26. gateway version

Displays the version number of the SGD Gateway software.

Syntax

```
gateway version
```

Description

Displays the version number of the SGD Gateway.

Examples

The following example displays the SGD Gateway version installed on the host where the command is run.

```
# /opt/SUNWsgdg/bin/gateway version
Oracle Secure Global Desktop Gateway 4.50.301
```

B.27. The tarantella gateway Command

Use the `tarantella gateway` command to configure authorized gateways for an SGD array.

Syntax

```
tarantella gateway add | list | remove
```

Description

Using the `tarantella gateway` command, you can add, remove, and list the gateways for an SGD array.

The `tarantella gateway` command can be used on any SGD server in the array. Any changes you make are automatically replicated on other array members.

When an SGD server joins an array, the set of gateways defined on the primary SGD server is copied to the new array member, overwriting any authorized gateways already present. Registered gateways are not deleted from an SGD server when it is detached from an array.

The available subcommands for the `tarantella gateway` command are shown in the following table.

Subcommand	Description	More Information
`add`	Adds an SGD Gateway for an SGD array	Section B.28, “tarantella gateway add”
`list`	Lists the SGD Gateways for an SGD array	Section B.29, “tarantella gateway list”
`remove`	Removes an SGD Gateway for an SGD array	Section B.30, “tarantella gateway remove”

Note

All `tarantella gateway` subcommands include a `--help` option. You can use this option to display help for the subcommand.

Examples

The following example adds `gateway1.example.com` to the list of registered gateways for the SGD array.

```
$ tarantella gateway add --name gateway1.example.com \
--certfile /opt/gateway1_cert_file.pem
```

B.28. tarantella gateway add

Registers an SGD Gateway with an SGD array.

Syntax

```
tarantella gateway add {
                        --name server-name
                        --certfile cert-file
                        } | --file file
```

Description

The following table shows the available options for this command.

Option	Description
`--name`	Name of the SGD Gateway to register.
`--certfile`	SGD Gateway certificate used by the SGD server. The certificate can be in Definite Encoding Rules (DER) or PEM format.
`--file`	A batch file containing configuration settings for multiple SGD Gateways.

Examples

The following example adds `gateway1.example.com` to the list of registered gateways for the SGD array.

```
$ tarantella gateway add --name gateway1.example.com \
--certfile /opt/gateway1_cert_file.pem
```

The following example uses the `--file` option of `tarantella gateway add` to register multiple gateways at the same time.

```
$ tarantella gateway add --file gateways.list
```

The `--file` option specifies a batch file, `gateways.list`, that contains a line of settings for each gateway, as follows:

```
--name gateway1.example.com --certfile /opt/gateway1_cert_file.pem
--name gateway2.example.com --certfile /opt/gateway2_cert_file.pem
```

B.29. tarantella gateway list

Lists the SGD Gateways registered for an SGD array.

Syntax

```
tarantella gateway list
```

Description

Shows details for the SGD Gateways that have been registered for an SGD array using `tarantella gateway add`.

Examples

The following example lists the registered gateways for the SGD array.

```
$ tarantella gateway list
```

B.30. tarantella gateway remove

Removes an SGD Gateway from the list of registered gateways for an SGD array.

Syntax

```
tarantella gateway remove --name server-name | --file file
```

Description

The following table shows the available options for this command.

Option	Description
`--name`	Name of the SGD Gateway to remove registration details for
`--file`	A batch file containing configuration settings for multiple SGD Gateways

Examples

The following example removes the SGD Gateway `gateway1.example.com` from the list of registered gateways for the SGD array.

```
$ tarantella gateway remove --name gateway1.example.com
```

B.31. The --security-gateway Attribute

Description

You use the `--security-gateway` attribute to enable SGD Gateway usage for the SGD array. The attribute defines the following:

- The SGD Clients that can access a SGD Gateway, based on their IP address or DNS name.

- The address that client devices use to contact the SGD Gateway.

Note

The `--security-gateway` attribute is used for AIP connections only. Routing of HTTP connections is handled by the HTTP load balancing service on the Apache reverse proxy component of the Gateway.

Changes to the `--security-gateway` attribute apply to all SGD servers in the array.

Syntax

The syntax for the `--security-gateway` attribute is as follows:

```
--security-gateway filter-spec...
```

Replace *`filter-spec`* with a filter specification of the type:

```
client-ip-address|*:gateway protocol:gateway-address:gateway-port
```

- The *`client-ip-address`* is the IP address of the SGD Client. For connections through the SGD Gateway, this is the interface that the SGD Gateway uses to connect to the SGD servers in the array.

  A single asterisk, `*`, represents all IP addresses.

  The client IP address string can contain the `*` and `?` wildcards, where `*` matches multiple characters and `?` matches a single character. For example:

  `192.169.10.*` matches all addresses on the `192.169.10` network.

  `192.169.10.12?` matches the address range from `192.169.10.120` to `192.169.10.129`.

Note

If you are using an external load balancer with the SGD Gateway, type the address of the load balancer for the *`client-ip-address`*.

- The *`gateway protocol`* is `sgdg` for connections through the SGD Gateway, or `direct` for SGD Clients that connect directly to an SGD array, without going through the SGD Gateway.
- The *`gateway-address`* is the external address of the SGD Gateway, or an external load balancer, if used. This is the address that client devices use to contact the SGD Gateway.

  For `direct` connections to an SGD array, specify the address of the primary server in the array.
- The *`gateway-port`* is the TCP port that client devices use to connect to the SGD Gateway, or an external load balancer, if used.

  For `direct` connections to an SGD array, specify the port for the primary server in the array.

Separate multiple *`filter-spec`* entries with a comma and enclose the entire string in double quotation marks (`" "`). See Section B.31, “ Using Multiple Filters ”.

Examples

The following example enables all SGD Clients to connect using TCP port 443 of the SGD Gateway `gateway1.example.com`.

```
$ tarantella config edit --security-gateway "*:sgdg:gateway1.example.com:443"
```

The following example enables all SGD Clients to connect using an external load balancer, `lb.example.com`.

```
$ tarantella config edit --security-gateway "*:sgdg:lb.example.com:443"
```

The following example enables all SGD Clients to connect directly to an SGD array, without going through the SGD Gateway. The primary server in the array is `sgd1.example.com`.

```
$ tarantella config edit --security-gateway "*:direct:sgd1.example.com:443"
```

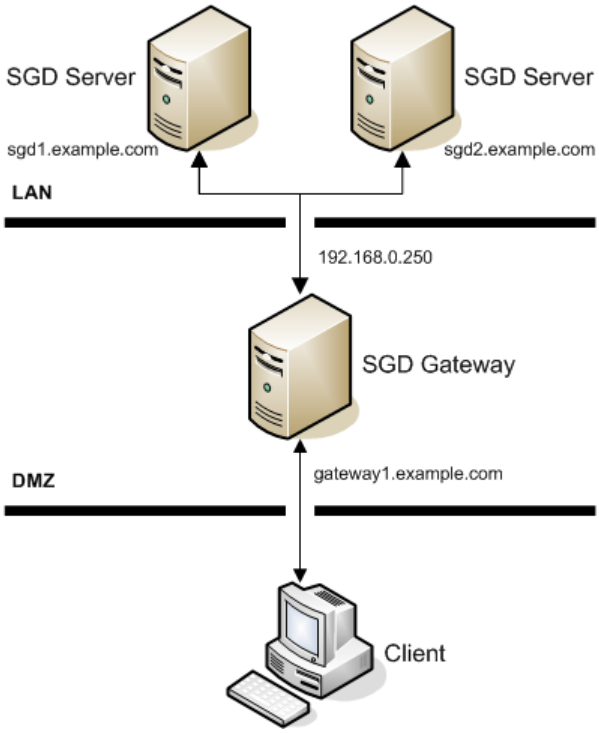
Using Multiple Filters

You can use multiple filter specifications, as shown in the following example.

Consider a basic deployment, as shown in Figure B.1, "Using Multiple Filter Specifications". The deployment uses a single SGD Gateway, `gateway1.example.com`, with an SGD array that contains two SGD servers, `sgd1.example.com` and `sgd2.example.com`. The primary server in the array is `sgd1.example.com`.

The address of the SGD Gateway on the internal network is `192.168.0.250`.

Figure B.1. Using Multiple Filter Specifications



The following filter specification might be used for this example:

```
"192.168.0.250:sgdg:gateway1.example.com:443,*:direct:sgd1.example.com:80"
```

With this configuration, the following applies:

- Connections to the SGD servers in the array are allowed from the SGD Gateway IP address, `192.168.0.250`. SGD Clients outside the organization connect using TCP port 443 of the SGD Gateway, `gateway1.example.com`.
- All other SGD Clients, such as those on the local area network (LAN), connect directly to TCP port 80 on the primary SGD server `sgd1.example.com`. These connections do not use the SGD Gateway.
- The order of the filters is important. If the order of the filters is reversed, all SGD Clients connect directly to the SGD server `sgd1.example.com`.

Appendix C. Advanced Configuration

This chapter includes information about configuring and using the advanced features of the Oracle Secure Global Desktop Gateway (SGD Gateway).

This chapter includes the following topics:

C.1. Tuning the SGD Gateway

When you install the SGD Gateway, default values for the maximum number of simultaneous Adaptive Internet Protocol (AIP), websocket, and HTTP connections are configured automatically, based on the available memory on the SGD Gateway host. The memory size allocated to the SGD Gateway's Java Virtual Machine (JVM) is also optimized for this number of connections.

After installing the SGD Gateway, depending on the expected number of SGD users and the number of applications they will run, you can adjust the default settings. When you do this, you might also need to adjust the JVM memory size. This process is called *tuning* the SGD Gateway.

Caution

If the JVM memory size is too low for the expected number of connections, the SGD Gateway might stop working and refuse all subsequent connections. In this case, you need to tune the SGD Gateway so that sufficient JVM memory is available. A `java.lang.OutOfMemoryError` error message on the SGD Gateway indicates that tuning might be required.

To tune the SGD Gateway, you do the following:

- Change the maximum number of AIP connections. See Section C.1.1, “Changing the Maximum Number of AIP Connections”.
- Change the maximum number of websocket connections. See Section C.1.2, “Changing the Maximum Number of Websocket Connections”.
- Change the maximum number of HTTP connections. See Section C.1.3, “Changing the Maximum Number of HTTP Connections”.
- Change the JVM memory size. See Section C.1.4, “Changing the JVM Memory Size”.

C.1.1. Changing the Maximum Number of AIP Connections

The maximum number of AIP connections is configured at install time. The default setting depends on the memory resources available on the SGD Gateway host.

You can change this setting to a value more appropriate for your deployment. See Section C.1.1.1, “Calculating the Number of AIP Connections” for details of how to calculate the maximum number of AIP connections used by an SGD Gateway.

To change the maximum number of AIP connections, use the `--routes-aip-maxcon` option of the `gateway config edit` command. For example, to change the maximum number of AIP connections to 3000, run the following command:

```
# /opt/SUNWsgdg/bin/gateway config edit --routes-aip-maxcon 3000
```

You must restart the SGD Gateway to enable any changes you make.

C.1.1.1. Calculating the Number of AIP Connections

The number of AIP connections used by an SGD Gateway depends on the number of concurrent SGD users, and the number of applications they run, as follows:

Number of AIP connections = (*number of applications* + 3) x *number of SGD users*

For example, an SGD Gateway with 1000 SGD users, each running four applications requires the following maximum number of simultaneous AIP connections:

(4 + 3) x 1000 = 7000 AIP connections

C.1.2. Changing the Maximum Number of Websocket Connections

Websocket connections are AIP connections to tablet devices.

The maximum number of websocket connections is configured at install time. The default setting depends on the memory resources available on the SGD Gateway host.

You can change this setting to a value more appropriate for your deployment. The maximum number of websocket connections used by an SGD Gateway is calculated in the same way as for AIP connections. See Section C.1.1.1, “Calculating the Number of AIP Connections” for details.

To change the maximum number of websocket connections, use the `--routes-ws-maxcon` option of the `gateway config edit` command. For example, to change the maximum number of websocket connections to 1000, run the following command:

```
# /opt/SUNWsgdg/bin/gateway config edit --routes-ws-maxcon 1000
```

You must restart the SGD Gateway to enable any changes you make.

C.1.3. Changing the Maximum Number of HTTP Connections

The maximum number of HTTP connections is configured at install time. This setting defines the maximum number of concurrent users. The default value is 100.

To change the maximum number of HTTP connections, use the `--routes-http-maxcon` option of the `gateway config edit` command. For example, to change the maximum number of HTTP connections to 200, run the following command:

```
# /opt/SUNWsgdg/bin/gateway config edit --routes-http-maxcon 200
```

You must restart the SGD Gateway to enable any changes you make.

C.1.4. Changing the JVM Memory Size

When you change the maximum number of AIP, websocket, and HTTP connections, you might need to change the memory size allocated to the SGD Gateway's JVM. To do this, edit the following settings in the `/opt/SUNWsgdg/proxy/etc/tuning_parameters` file:

- `-Xms` – Initial memory size for the JVM, in bytes
- `-Xmx` – Maximum memory size for the JVM, in bytes

Tip

You can use the `K` (kilo) and `M` (mega) modifiers with these settings. For example: 960K = 960 kilobytes, and 512M = 512 megabytes.

See Section C.1.4.1, “Calculating the JVM Memory Size” for details of how to calculate JVM memory size values.

Note

Ensure that your system is configured with sufficient memory resources for the JVM settings you make.

You must restart the SGD Gateway to enable any changes you make.

C.1.4.1. Calculating the JVM Memory Size

The amount of JVM memory used by the SGD Gateway depends on the number of simultaneous AIP connections, websocket connections, and HTTP connections.

As each SGD Gateway connection requires approximately 300 kilobytes of JVM memory, the required JVM memory is given by:

(*number of AIP connections* + *number of websocket connections* + *number of HTTP connections*) x 300 kilobytes

For example, an SGD Gateway with 500 SGD users, each running two applications. The maximum number of simultaneous AIP connections is:

(2 + 3) x 500 = 2500 AIP connections

Suppose the SGD Gateway has 100 tablet device users, each running two applications. The maximum number of simultaneous websocket connections is:

(2 + 3) x 100 = 500 websocket connections

The SGD Gateway must also handle sufficient simultaneous HTTP connections to the SGD web server. For this example, the maximum number of HTTP connections is:

300 HTTP connections

So, the required JVM memory is:

(2500 + 500 + 300) x 300 kilobytes = 990 megabytes, approximately.

Note

In the `/opt/SUNWsgdg/proxy/etc/tuning_parameters` file, set the `-Xms` and `-Xmx` parameters to the calculated JVM memory value. `-Xms` and `-Xmx` are normally set to be the same value for performance reasons.

C.2. Configuring Data Compression for Connections to Tablet Devices

By default, data connections between the SGD Gateway and tablet devices are compressed. This is called *websocket compression*.

For some network deployments, you may want to turn off websocket compression. For example, if you need to reduce the processing load on the client device.

To disable websocket compression for the Gateway, edit the `<websocketCompression>` setting in the `/opt/SUNWsgdg/etc/gateway.xml` file, as follows:

```
<websocketCompression>off</websocketCompression>
```

You must restart the SGD Gateway to enable any changes you make.

Note

The websocket compression setting for the Gateway takes precedence over the websocket compression setting used for the SGD array. See the Configuring Data Compression for Connections to Tablet Devices section in the *Oracle Secure Global Desktop Administration Guide*.

C.3. Configuring HTTP Redirection

By default, the SGD Gateway refuses HTTP connections on TCP port 80.

To enable connections on TCP port 80, use the `gateway config enable` command to enable the HTTP redirection service, as follows:

```
# /opt/SUNWsgdg/bin/gateway config enable --routes-http-redirect
```

You must restart the SGD Gateway to enable any changes you make.

C.4. Changing the Binding Port for the SGD Gateway

The interface and port that the SGD Gateway uses for incoming connections is called the *binding port*. By default, the SGD Gateway uses TCP port 443 on all interfaces as the binding port.

To change the binding port, use the `--binding` option of the `gateway config edit` command. For example, to change the binding port to TCP port 4443, run the following command:

```
# /opt/SUNWsgdg/bin/gateway config edit --binding *:4443
```

Alternatively, you can change the binding port by running the `/opt/SUNWsgdg/bin/gateway config create` command on the SGD Gateway host. This command prompts you to specify an interface and port to use for incoming proxy connections.

Note

The `gateway config create` command creates a new configuration and overwrites any configuration settings you have made.

You must restart the SGD Gateway to enable any changes you make.

C.5. Using Unencrypted Connections to the SGD Array

By default, connections between the SGD Gateway and the SGD servers in the array are secured using Secure Sockets Layer (SSL). This means that AIP over SSL data uses TCP port 5307, and HTTPS data uses TCP port 443.

To use unencrypted connections between the SGD Gateway and the SGD servers in the array, see Section C.5.1, “Configuring the Gateway to Use Unencrypted Connections to the SGD Array”.

For unencrypted connections, AIP data uses TCP port 3144, and HTTP data uses TCP port 80.

C.5.1. Configuring the Gateway to Use Unencrypted Connections to the SGD Array

This procedure describes how to reconfigure a Gateway deployment to use unencrypted connections.

1. Modify the Gateway configuration to use unencrypted connections to the SGD array.

```
# gateway config create
```

Note

This command overwrites the current configuration of the Gateway.

When prompted whether to secure the connections between the Gateway and the SGD servers in the array, enter `n`.

2. Remove any previously registered SGD servers for the Gateway.

```
# /opt/SUNWsgdg/bin/gateway server remove --server sgd.example.com
```

where *`sgd.example.com`* is the name of the SGD server.

The CA certificate and SSL certificate for the SGD server are removed from the Gateway keystore.

3. Ensure that the SGD servers in the array are configured to use standard, unencrypted connections.

Run the following command on each SGD server in the array to turn off SGD security services.

```
# tarantella security disable
```

4. Register the SGD servers in the array with the Gateway.

```
# /opt/SUNWsgdg/bin/gateway server add --server sgd.example.com \
--certfile PeerCAcert.pem \
--url http://sgd.example.com
```

This example adds the CA certificate `PeerCAcert.pem` to the SGD Gateway keystore, using the alias *`sgd.example.com`*. The URL of the SGD web server is *`http://sgd.example.com`*.

5. Restart the Gateway.

```
# /opt/SUNWsgdg/bin/gateway restart
```

C.6. Using External SSL Accelerators

By default, the SGD Gateway is configured to work with incoming HTTP and AIP data connections that are secured using SSL. The Gateway also supports the use of external SSL accelerators for handling SSL processing.

To use an external SSL accelerator with the Gateway, do the following:

- Configure the external SSL accelerator to decrypt SSL connections and forward them as unencrypted connections to the Gateway.
- Enable external SSL accelerator support on the Gateway.

  This enables the Gateway to accept unencrypted connections on the secure port. See Section C.6.1, “How to Enable External SSL Accelerator Support”.
- Ensure that client devices use the SSL accelerator as the network entry point.

  Typically the SSL accelerator is also a load balancer. Configure the SGD servers and Gateways for a load-balanced deployment as described in Section 2.1.2, “Load-Balanced Deployment”.

C.6.1. How to Enable External SSL Accelerator Support

Ensure that no users are connected to SGD through the Gateway.

1. Log in as superuser (root) on the SGD Gateway host.
2. Enable support for unencrypted incoming connections.

   Change the symbolic link for the `gateway.xml` file, so that it links to the `gateway-plaintext.xml` file, instead of the default setting of `gateway-ssl.xml`.

   Run the following command:

```
# ln -fs /opt/SUNWsgdg/etc/gateway-plaintext.xml /opt/SUNWsgdg/etc/gateway.xml
```

3. (Optional) Change the binding port for the Gateway.

   Depending on your network configuration, you might also need to change the binding port for the SGD Gateway.

   See Section C.4, “Changing the Binding Port for the SGD Gateway”.
4. Restart the SGD Gateway.

```
# /opt/SUNWsgdg/bin/gateway restart
```

C.7. Configuring Ciphers for the SGD Gateway

The Gateway supports a wide range of cipher suites for SSL connections. See the *Oracle Secure Global Desktop Platform Support and Release Notes* for a list of supported cipher suites.

During installation, the Gateway is configured to use a set of ciphers consisting of only high grade ciphers. This means that SSL connections to the Gateway always use enhanced security. If required, you can configure the Gateway to use a different set of ciphers.

C.7.1. How to Configure Ciphers for the Gateway

1. Stop the Gateway.

```
# /opt/SUNWsgdg/bin/gateway stop
```

2. Configure the required ciphers.

   In the `/opt/SUNWsgdg/etc` directory, edit the `ciphersuites.xml` file.

   By default, the `ciphersuites.xml` file contains the following entries for high grade ciphers.

```
<ciphersuites>
 <cipher>SSL_RSA_WITH_RC4_128_MD5</cipher>
 <cipher>SSL_RSA_WITH_RC4_128_SHA</cipher>
 <cipher>TLS_RSA_WITH_AES_128_CBC_SHA</cipher>
 <cipher>TLS_RSA_WITH_AES_256_CBC_SHA</cipher>
 <cipher>TLS_DHE_RSA_WITH_AES_128_CBC_SHA</cipher>
 <cipher>TLS_DHE_RSA_WITH_AES_256_CBC_SHA</cipher>
 <cipher>TLS_DHE_DSS_WITH_AES_128_CBC_SHA</cipher>
 <cipher>TLS_DHE_DSS_WITH_AES_256_CBC_SHA</cipher>
 <cipher>SSL_RSA_WITH_3DES_EDE_CBC_SHA</cipher>
 <cipher>SSL_DHE_RSA_WITH_3DES_EDE_CBC_SHA</cipher>
 <cipher>SSL_DHE_DSS_WITH_3DES_EDE_CBC_SHA</cipher>
</ciphersuites>
```

3. Check that the following entry is present in the `/opt/SUNWsgdg/etc/gateway.xml` file, so that it includes `ciphersuites.xml`.

```
<service id="sgd-ssl-service" class="SSL">
                ...
  <keystore file="/opt/SUNWsgdg/proxy/etc/keystore.client"
   password="/opt/SUNWsgdg/etc/password"/>
   <xi:include href="ciphersuites.xml" parse="xml"/>
</service>
                ...
<service id="http-ssl-service" class="SSL">
                ...
  <keystore file="/opt/SUNWsgdg/proxy/etc/keystore.client"
   password="/opt/SUNWsgdg/etc/password"/>
  <xi:include href="ciphersuites.xml" parse="xml"/>
</service>
```

4. Restart the Gateway.

```
# /opt/SUNWsgdg/bin/gateway start
```

C.8. Using Client Certificates With the SGD Gateway

You can use *client certificates* to enhance the security of the SGD Gateway, by restricting access to those users who have a valid certificate.

A client certificate is an SSL certificate that is installed in the browser on the client device. See the online documentation of your browser for details of how to install a client certificate.

See Section C.8.2, “How to Generate a CSR for a Client Certificate” if you need to generate a certificate signing request (CSR) for a new client certificate.

The following procedures use the `keytool` application. See the JDK Tools and Utilities documentation for details on how to use the `keytool` application.

C.8.1. How to Configure the SGD Gateway to Use Client Certificates

1. Log in as superuser (root) on the SGD Gateway host.
2. Stop the SGD Gateway.

```
# /opt/SUNWsgdg/bin/gateway stop
```

3. Configure the SGD Gateway to use client certificates for HTTPS client connections.

   Add a `<needClientAuth>` entry to the `/opt/SUNWsgdg/etc/gateway.xml` file, as follows:

```
<service id="http-ssl-service" class="SSL">
    <needClientAuth>true</needClientAuth>
        <!-- Decrypts HTTPS traffic -->
        <subService id="ssl-splitter">
            <binding>*</binding>
        </subService>
```

4. (Optional) Import the client certificate into the SGD Gateway client keystore.

   Note

   You do not need to do this step if the client certificate is signed by a trusted Certificate Authority (CA).

   Use the `keytool` command, as follows:

```
# /opt/SUNWsgdg/java/default/bin/keytool -importcert \
-alias mycert -keystore /opt/SUNWsgdg/proxy/etc/keystore.client \
-file mycert.crt -storepass ‘cat /opt/SUNWsgdg/etc/password‘
```

   In this example, the client certificate `mycert.crt` is imported into the SGD Gateway client keystore. The client certificate is stored using an alias of `mycert`.

5. Start the SGD Gateway.

```
# /opt/SUNWsgdg/bin/gateway start
```

C.8.2. How to Generate a CSR for a Client Certificate

To obtain a client certificate that you can use with the Gateway, you first need to generate a CSR. You then send the CSR to a Certificate Authority (CA) for signing.

Note

This procedure describes how you can use the `keytool` application on the Gateway host to generate a CSR. However, you do not have to use the steps described in this procedure. Instead, you can use your favorite certificate management tool to generate the CSR.

1. Log in as superuser (root) on the SGD Gateway host.

2. Generate a self-signed certificate and a corresponding private key.

   Use the `keytool` command, as follows:

   ```
   # /opt/SUNWsgdg/java/default/bin/keytool -genkeypair -keyalg RSA \
   -alias mycert -keystore keystore.mycert -storepass letmein
   ```

   In this example, a self-signed certificate and private key are created and stored in a keystore called `keystore.mycert`. The key pair is stored using an alias of `mycert`.

3. Generate a CSR for the self-signed certificate.

   Use the `keytool` command, as follows:

   ```
   # /opt/SUNWsgdg/java/default/bin/keytool -certreq  \
   -alias mycert -keystore keystore.mycert -storepass letmein \
   -file /tmp/gateway-name.csr
   ```

   In this example, a CSR is generated and stored in the file `/tmp/gateway-name.csr`, where *`gateway-name`* is the name of the Gateway.

C.9. Enabling the Balancer Manager Application

The Apache reverse proxy includes a web application called Balancer Manager. Balancer Manager enables you to manage the SGD web servers in the load balancing group used by the reverse proxy.

Using Balancer Manager, you can do the following:

- View status information for SGD web servers in the load balancing group
- View and change load balancing routes for SGD web servers
- Remove SGD web servers from the load balancing group

To enable Balancer Manager, remove the comments in the reverse proxy configuration file, `/opt/SUNWsgdg/httpd/apache-version/conf/extra/gateway/httpd-gateway.conf` that disable the application.

```
# Allows the configuration of load balancing parameters
#
#    <Location /balancer-manager>
#        SetHandler balancer-manager
#        Order Deny,Allow
#        Deny from all
#        Allow from all
#    </Location>
```

You must restart the Gateway to enable any changes you make.

```
# /opt/SUNWsgdg/bin/gateway restart
```

To access Balancer Manager, start a browser and go to `https://gateway.example.com/balancer-manager`, where *`gateway.example.com`* is the SGD Gateway host.

For more details about configuring the Balancer Manager, see the Apache mod_proxy_balancer documentation.

C.10. The Reflection Service

The *reflection service* is a collection of RESTful web services used by the routing proxy component of the SGD Gateway. Using the reflection service, an SGD Gateway Administrator can configure routes, services, logging levels, and connections, and show status information for the routing proxy.

This section includes the following reflection service topics:

C.10.1. Enabling the Reflection Service

By default, the reflection service is not enabled for the SGD Gateway.

You enable the reflection service for one or more of the following access methods:

- **Unauthorized access** – Users do not need to authenticate.

  By default, unauthorized access is only available from the SGD Gateway host.

  See Section C.10.1.1, “How to Enable Unauthorized Access to the Reflection Service” for details of how to enable unauthorized access.

- **Authorized access** – Users must authenticate before accessing the reflection service.

  See Section C.10.1.2, “How to Enable Authorized Access to the Reflection Service” for details of how to enable authorized access.

C.10.1.1. How to Enable Unauthorized Access to the Reflection Service

1. Log in as superuser (root) on the SGD Gateway host.
2. Enable unauthorized access to the reflection service.

   ```
   # /opt/SUNWsgdg/bin/gateway config enable --services-reflection
   ```

3. (Optional) Change the interface used by the reflection service.

   Caution

   By default, unauthenticated access to the reflection service is only available from the SGD Gateway host. Enabling unauthenticated access on other interfaces can present a security risk.

   The default interface used for unauthorized access to the reflection service is the `localhost` loopback interface. The following example shows how to enable unauthorized access on all interfaces:

   ```
   # /opt/SUNWsgdg/bin/gateway config edit \
   --services-reflection-binding *:81
   ```

4. (Optional) Change the port used by the reflection service.

   The default port used for unauthorized access to the reflection service is TCP port 81. You can change this to another port that is not in use, as follows:

```
# /opt/SUNWsgdg/bin/gateway config edit \
--services-reflection-binding localhost:portnum
```

where *portnum* is the port number used by the reflection service.

5. Restart the SGD Gateway.

```
# /opt/SUNWsgdg/bin/gateway restart
```

6. Access the reflection service.

   On the SGD Gateway host, you can start a browser and go to `http://localhost:81`.

   The home page for the reflection service is shown.

C.10.1.2. How to Enable Authorized Access to the Reflection Service

1. On the SGD Gateway host, log in as superuser (root).
2. Export the certificate and private key for the reflection service.

   The certificate and private key for the reflection service are stored in the reflection service keystore, at `/opt/SUNWsgdg/proxy/etc/keystore.reflection`. This keystore is created automatically during installation of the SGD Gateway.

   By default, the reflection service keystore contains a single, self-signed certificate and key pair.

   a. Export the certificate for the reflection service.

   ```
   # /opt/SUNWsgdg/java/default/bin/keytool -exportcert \
   -alias server-name -rfc \
   -keystore /opt/SUNWsgdg/proxy/etc/keystore.reflection \
   -storepass "$(cat /opt/SUNWsgdg/etc/password)" \
   -file client.pem
   ```

   where *server-name* is the alias used for the reflection service certificate in the reflection keystore and *client.pem* is the file name of the exported certificate.

   See the JDK Tools and Utilities documentation for details on how to use the `keytool` application.

   b. Export the private key for the reflection service.

   Use the KeyManager application included with the SGD Gateway.

   ```
   # /opt/SUNWsgdg/java/default/bin/java \
   -jar /opt/SUNWsgdg/proxy/KeyManager.jar export \
   --keyfile client.key \
   --keystore /opt/SUNWsgdg/proxy/etc/keystore.reflection \
   --keyalias alias-name \
   --keypass "$(cat /opt/SUNWsgdg/etc/password)" \
   --storepass "$(cat /opt/SUNWsgdg/etc/password)"
   ```

   where *alias-name* is the alias used for the reflection service key in the reflection keystore and *client.key* is the file name of the exported key.

3. Install the certificate and private key on the client device.

The certificate and private key are used by the client device to authorize to the reflection service.

To import the certificate and key into a browser certificate store you must first convert the certificate and key into a PKCS12 format file. For example:

```
# openssl pkcs12 -export -in mycert.crt -inkey mycert_key.pem -out mycert.p12
```

This command converts the certificate file `mycert.crt` and associated private key `mycert_key.pem` into a PKCS12 format certificate file `mycert.p12`.

For more details of how to import a PKCS12 format certificate into your browser, see the online documentation for your browser.

4. Enable authorized access to the reflection service.

   On the SGD Gateway host, run the following command:

   ```
   # /opt/SUNWsgdg/bin/gateway config enable --services-reflection-auth
   ```

5. (Optional) Change the interface and port used by the reflection service.

   The default binding used for authorized access to the reflection service is TCP port 82 on all interfaces. You can change this to another interface and a port that is not in use, as follows:

   ```
   # /opt/SUNWsgdg/bin/gateway config edit \
   --services-reflection-binding int:portnum
   ```

   where *`int`* is the interface, and *`portnum`* is the port number used by the reflection service.

6. Restart the SGD Gateway.

   ```
   # /opt/SUNWsgdg/bin/gateway restart
   ```

7. Connect to the reflection service from the client device, using the certificate and private key.

   - Using the `curl` command:

     ```
     $ curl --cert client.pem --key client.key -k -X GET https://gateway.example.com:82
     ```

     In this example, the `curl` command is used to access the home page of the reflection service at `https://gateway.example.com:82`, where *`gateway.example.com`* is the name of the SGD Gateway. The certificate and private key for the reflection service are *`client.pem`* and *`client.key`*.

   - Using a browser:

     Go to `https://gateway.example.com:82`, where *`gateway.example.com`* is the name of the SGD Gateway.

     The home page for the reflection service is shown.

C.10.2. Using the Reflection Service

Use a client application to access the RESTful web services provided by the reflection service. Examples of suitable client applications include the following:

- **Browser.** Using a browser is the simplest method of accessing the reflection service. However, a browser only supports HTTP `GET` requests and therefore limits access to only those RESTful web services that retrieve information. In practice, using a browser is helpful for tasks such as displaying status information and listing routes and services for the routing proxy.
- **curl.** This is a command-line tool for UNIX and Linux platforms that supports HTTP `GET`, `PUT`, `POST`, and `DELETE` requests. This means that the full range of RESTful web services for the reflection service can be used. Output from this tool can be redirected to a file, or to another program for further processing.

Alternatively, if you have your own client application that supports RESTful web services, you can use it to access the reflection service.

Note

You do not need to restart the SGD Gateway when you use the reflection service to change the configuration of the routing proxy.

Data can be returned from the reflection service in the following output formats:

- **ASCII.** This is the default output format. Data is returned in tab-delimited ASCII format. This output format is useful if the data is to undergo further processing, such as parsing.
- **HTML.** Data is returned in HTML format, suitable for displaying in a browser. To return HTML output, append `/html` to the end of the web service Uniform Resource Identifier (URI).

C.10.2.1. About the RESTful Web Services

Table C.1, "RESTful Web Services for the SGD Gateway Reflection Service" lists the RESTful web services for the SGD Gateway reflection service.

Table C.1. RESTful Web Services for the SGD Gateway Reflection Service

Relative URI	HTTP Request Method	Description
`/`	`GET`	Shows high-level information for the routing proxy, such as the uptime.
`/service`	`GET`	Lists the available services. A service represents an entry point from which the routing proxy creates incoming connections.
`/service/`*`Service-Id`*	`GET`	Lists information for a service, identified by *`Service-Id`*.
`/service/`*`Service-Id`*	`PUT`	Starts a service, identified by *`Service-Id`*.
`/service/`*`Service-Id`*	`DELETE`	Stops a service, identified by *`Service-Id`*.
`/client`	`GET`	Lists the available clients. A client represents an exit point on which the routing proxy constructs outgoing connections.
`/client/`*`Client-Id`*	`GET`	Lists information for a client, identified by *`Client-Id`*.
`/route`	`GET`	Lists the available routes. A route represents a path through the routing proxy, from incoming connections through services to outgoing connections through clients.

Relative URI	HTTP Request Method	Description
`/route/Route-Id`	`GET`	Lists information for a route, identified by *`Route-Id`*.
`/route/Route-Id`	`PUT`	Starts a route, identified by *`Route-Id`*.
`/route/Route-Id`	`DELETE`	Stops a route, identified by *`Route-Id`*.
`/route/Route-Id/connection`	`GET`	Lists the connections for a specific route, identified by *`Route-Id`*.
`/route/Route-Id/connection/Connection-Id`	`DELETE`	Terminates a connection, identified by *`Connection-Id`*.
`/connection`	`GET`	Lists all currently running connections, for all routes.
`/logging/level`	`GET`	Shows the global logging level.
`/logging/level/Log-Level`	`PUT`	Sets the routing proxy's global logging level.
`/logging/Package/level`	`GET`	Shows the logging level for a specific component of the routing proxy.
`/logging/Package/level/Log-Level`	`PUT`	Sets the logging level for a specific component of the routing proxy.

To access a RESTful web service, append the relative URI for the web service to the URL of the reflection service.

For example, to list the available routes for an SGD Gateway, *`gateway.example.com`*, append `/route` to the URL of the reflection service, as follows:

```
$ curl --cert client.pem --key client.key -k -X GET https://gateway.example.com:82/route
```

where *`client.pem`* and *`client.key`* are the certificate and private key for the reflection service. In this example, the client is authorized before accessing the reflection service.

C.10.2.2. Examples of Using the Reflection Service

All of the following examples use the `curl` command as a client application to access the reflection service.

The examples use authenticated access to the reflection service on an SGD Gateway called *`gateway.example.com`*. The client is authorized using a certificate, *`client.pem`*, and a private key, *`client.key`*.

To list the available services for the SGD Gateway:

```
$ curl --cert client.pem --key client.key -k \
-X GET https://gateway.example.com:82/service
```

To stop a route, specify the Route Id that the reflection service uses for the route:

```
$ curl --cert client.pem --key client.key -k \
-X GET https://gateway.example.com:82/route
Route Id  Route Uptime   Service Id  ...
0         21h18m20s743m  ssgd-route-service  ...
1         21h18m20s736m  shttp-ssl-service   ...
$ curl --cert client.pem --key client.key -k \
```

```
-X DELETE https://gateway.example.com:82/route/1
```

To set the global logging level to FINER:

```
$ curl --cert client.pem --key client.key -k \
-X PUT https://gateway.example.com:82/logging/level/FINER
```

Appendix D. Troubleshooting the SGD Gateway

This chapter includes troubleshooting topics, to help you to diagnose and fix problems with the Oracle Secure Global Desktop Gateway (SGD Gateway).

This chapter includes the following topics:

D.1. Logging and Diagnostics

This section describes the logging and diagnostics features of the SGD Gateway.

This section includes the following topics:

D.1.1. About SGD Gateway Logging

SGD Gateway logging uses the Java logging application programming interface (API). For more details about how logging is implemented in Java, see http://download.oracle.com/javase/6/docs/technotes/guides/logging/overview.html.

D.1.1.1. Changing the Logging Level

A logging properties configuration file, `logging.properties`, is supplied with the SGD Gateway. This file is in the `/opt/SUNWsgdg/proxy/etc` directory.

You can edit the `logging.properties` file to change the default logging level, and to configure logging levels for specific SGD Gateway services. Each SGD Gateway service is represented by an `async.channel` entry in the `logging.properties` file.

For example, if you want to increase logging levels for incoming and outgoing TCP connections, set the TCP service logging level to `FINEST`. Uncomment the following line in the `logging.properties` file:

```
# async.channel.tcp.level=FINEST
```

The `FileHandler` class documentation describes the logging level parameters that you can use in a `logging.properties` file.

You must restart the SGD Gateway to enable any changes to logging levels you make by editing the `logging.properties` file.

Note

You can also use the SGD Gateway reflection service to change logging levels. See Section C.10, “The Reflection Service” for information about configuring and using the reflection service.

D.1.1.2. Log File Locations

If you have problems with the SGD Gateway, consult the following log files:

- **Routing proxy log files.** The location and names of these log files are set in the `logging.properties` file. By default, the SGD Gateway creates routing proxy log files in the `/opt/SUNWsgdg/proxy/var/log` directory on the SGD Gateway host.
- **Reverse proxy log files.** Details of load balancing and proxy server activity for HTTP and HTTPS connections are logged to the Apache log files in the `/opt/SUNWsgdg/httpd/apache-version/logs` directory on the SGD Gateway host.
- **SGD server log files.** Each SGD server in the array writes error messages to log files in the `/opt/tarantella/var/log` directory on the SGD server host. See “Monitoring and Logging” in Chapter 6 of the *Oracle Secure Global Desktop Administration Guide* for more details about configuring logging for SGD servers.

D.1.2. Displaying SGD Gateway Process Information

When you start the SGD Gateway, the process ID of the routing proxy is stored to the `/opt/SUNWsgdg/proxy/var/run/proxy.pid` file on the SGD Gateway host.

The process ID of the reverse proxy is stored to the `/opt/SUNWsgdg/httpd/apache-version/logs/httpd.pid` file. This file location can be changed using the `PidFile` directive in the `httpd.conf` Apache configuration file.

To display the running SGD Gateway processes, use the following command on the SGD Gateway host:

```
# ps -ef| grep SUNWsgdg
```

D.1.3. Checking the Configuration From the Command Line

You can use the following commands to check your SGD Gateway configuration.

- `gateway status` – Shows status information for the SGD Gateway.

  Run the following command on the SGD Gateway host:

  ```
  # /opt/SUNWsgdg/bin/gateway status
  ```

  See also Section B.23, “gateway status” for more information about this command.

- `tarantella gateway list` – Displays a list of the SGD Gateways that are authorized for use by the SGD array.

  Run the following command on any SGD server in the array:

  ```
  $ tarantella gateway list
  ```

  See Section B.27, “The tarantella gateway Command” for more details about using the `tarantella gateway` command.

- `tarantella config list` – Displays global settings for the SGD array.

  Run the following command on any SGD server to show the `--security-gateway` attribute setting. This attribute determines which SGD Clients are allowed to use the SGD Gateway.

```
$ tarantella config list --security-gateway
```

See Section B.31, “The --security-gateway Attribute” for more details about this attribute.

D.2. Changing the Peer DNS Name of an SGD Server

The *peer DNS name* is the DNS name that an SGD server uses to identify itself to other SGD servers in the array. For example, `boston.example.com`.

When you change the peer DNS name of an SGD server, the Gateway might no longer be able to connect to the server. This is because the certificates used by the Gateway do not contain the new DNS name.

You might have to reconfigure your Gateway deployment as follows:

1. (Optional) Install the new SGD server SSL certificate. See Section 2.2.2.1, “How to Install SGD Server Certificates”.

   This step is required if the new peer DNS name is not included in the SSL certificate used by the SGD server. You must replace the SSL certificate on the SGD server and install the new SSL certificate on each Gateway.

2. (Optional) Install the new CA certificate for the SGD server. See Section 2.2.2.1, “How to Install SGD Server Certificates”.

   This step is required if you change the peer DNS name of the primary server in the array. You must regenerate the certificates used for secure intra-array communication and install the new CA certificate on each Gateway.

See the "Peer DNS Names" section in Chapter 1 of the *Oracle Secure Global Desktop Administration Guide* for more information about how to change the peer DNS name of an SGD server.

D.3. SGD Gateway Error Messages

SGD Gateway error messages are reported to the routing proxy log files, located in the `/opt/SUNWsgdg/proxy/var/log` directory on the SGD Gateway host.

Some typical SGD Gateway error messages, along with an explanation of the probable cause, are listed in Table D.1, “Error Messages for the SGD Gateway”.

Table D.1. Error Messages for the SGD Gateway

Error Message	Probable Cause

`Failed to validate token:`
 	
`Token time not yet valid`
	The clocks on the SGD Gateway and the SGD servers in the array are not synchronized

`Failed to decode token:`
 	
`No trusted signature found`
	The CA certificate for the SGD server has not been installed on the SGD Gateway

`Failed to validate token:`
 	
`No recipient available to decrypt token`
	The SGD Gateway certificate has not been installed on the SGD array

`SSL error:`	The SSL certificate for the SGD server has not been installed on the SGD Gateway

Error Message	Probable Cause
Check the proxy SSL keystore has valid trusted certificates	