Oracle® Secure Global Desktop

Security Guide for Release 4.7

E36389-01
August 2012

Oracle® Secure Global Desktop: Security Guide for Release 4.7

Abstract

This guide explains how to install, configure, and manage Oracle Secure Global Desktop securely.

Document generated on: 2012-08-06 (revision: 1020)

Table of Contents

Preface

The *Oracle Secure Global Desktop Security Guide for Release 4.7* provide information about how to install, configure, and deploy Oracle Secure Global Desktop (SGD) securely. This document is written for system administrators.

1. Audience

This document is intended for new users of SGD. 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, *Overview of Security for SGD* gives an overview of security for SGD.
- Chapter 2, *Secure Installation and Configuration of SGD* describes how to install SGD securely.
- Chapter 3, *Network Security for SGD* describes how to secure the network connections used by SGD.
- Chapter 4, *Security for Users, Applications, and Clients* describes the supported authentication mechanisms for SGD. Security features when configuring applications displayed through SGD are covered. This chapter also describes how you use organizational hierarchies to manage SGD users and objects.
- Chapter 5, *Security for SGD Servers and Arrays* includes security topics about SGD servers and arrays.
- Chapter 6, *Troubleshooting an SGD Deployment* describes how to troubleshoot potential security issues with your SGD deployment.

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 for Release 4.7*
- *Oracle Secure Global Desktop Installation Guide for Release 4.7*
- *Oracle Secure Global Desktop Gateway Administration Guide for Release 4.7*

- *Oracle Secure Global Desktop User Guide for Release 4.7*
- *Oracle Secure Global Desktop Platform Support and Release Notes for Release 4.7*

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. Overview of Security for SGD

This chapter provides an overview of security for Oracle Secure Global Desktop (SGD).

The chapter describes the architecture of SGD, the security mechanisms used by SGD, and how some general principles of information security apply to SGD.

1.1. SGD Network Architecture

SGD is built around a three-tier network architecture model, consisting of the following tiers:

- Client devices
- SGD servers
- Application servers

See the following links for more information about the SGD network architecture model:

- SGD Network Architecture describes the SGD network architecture.
- Section 3.1, “Network Connections for SGD” gives an overview of the main network connections involved when using SGD.

1.2. SGD Server Security

By default, an SGD server is installed and configured to use secure connections. See Chapter 2, *Secure Installation and Configuration of SGD*.

Connections in an SGD array are secured as follows:

- Connections to the SGD server are secured using SSL
- Connections between SGD servers in the array are secured using SSL
- Connections between SGD servers and application servers are encrypted

See Chapter 3, *Network Security for SGD* for more details about secure connections in an SGD deployment.

1.3. The SGD Gateway

The SGD Gateway can be used to provide an increased level of security between client devices and SGD servers. When you use the Gateway, client devices do not connect directly to SGD.

See Section 3.7, “The SGD Gateway” for more information about the Gateway.

1.4. SGD Administrators

An SGD Administrator is a user who has special privileges to create user accounts and manage an SGD array. See Section 4.2.2, “SGD Administrators”.

SGD has the following administration tools:

- **Administration Console** – Enables user and user session management, SGD server configuration, and the configuration of applications for SGD users.

- **Profile Editor** – Enables definition of settings for the SGD Client for the users in your organization.
- **`tarantella` command** – Enables control and configuration of SGD from the command line.

The Administration Console and the Profile Editor are available on the webtop of SGD Administrators.

1.5. Authenticating Users

SGD is designed to integrate with your existing authentication infrastructure and has the following methods for authenticating users:

- **System authentication.** SGD checks the user's credentials against one or more external authentication services, for example a Lightweight Directory Access Protocol (LDAP) directory.
- **Third-party authentication.** An external mechanism authenticates the user and SGD trusts that the authentication is correct. The most common use of third-party authentication is web server authentication.

See Section 4.1, "Authenticating Users" for more details about the authentication mechanisms supported by SGD.

1.6. Access Control

Access to data and resources in SGD is controlled using an object hierarchy.

Users, applications, and application servers are represented by *objects* in a directory. The objects are arranged into an *organizational hierarchy* representing your organization. See Section 4.2.1, "Organizations and Objects" for more details.

In SGD, administration privileges are managed using the Global Administrators role object in the System Objects organization. See Section 4.2.2, "SGD Administrators" for more details.

Privileged users can create objects, configure settings, and use commands to control and manage SGD.

1.7. Security Auditing and Logging

SGD supports auditing of system events, monitoring of user activity, and logging of system activity.

Log filters can be configured, to obtain information for specific system events or SGD components.

See Section 5.4, "Monitoring and Logging" for more details.

1.8. General Security Principles

Note

It is good practice to establish a thorough security policy and make sure that the policy is enforced. SGD must be configured to comply with your security policy.

SGD supports the following general principles of information security:

- **Minimize the attack surface**. All SGD network traffic can be directed through a single port, usually port 443. This is achieved by using the SGD Gateway or by running SGD servers in firewall traversal mode.
- **Least privilege permissions**. Application files and generated files such as password, audit, and log files are given the most restrictive permissions possible.

- **Secure installation**. Following a default installation, SGD is configured automatically to use secure connections.
- **Secure connections**. Transport Layer Security (TLS) or Secure Sockets Layer (SSL) are used to provide secure connections to an SGD server.
- **Monitor system activity**. SGD includes support for logging and auditing of system activity.

1.9. Security Fixes for Oracle Products

The Security page on Oracle Technology Network (OTN) contains details of the latest security fixes and critical patch updates for Oracle products.

Chapter 2. Secure Installation and Configuration of SGD

This chapter gives an overview of the installation process for Oracle Secure Global Desktop (SGD).

Installation of the main SGD component, as well as optional components such as the SGD Enhancement Module and SGD Gateway are described. Some post-installation topics are also covered.

2.1. Overview of Installing SGD

This section gives an overview of how to install SGD in a secure manner.

By default, SGD is installed to use secure connections between client devices and the SGD server, and to use secure connections between the SGD servers in the array.

Connections between the client device and the SGD server are secured during installation as follows:

- AIP connections are secured by installing an SSL certificate on the SGD server and enabling SGD security services.
- HTTP connections are secured by enabling HTTPS connections on the SGD web server. The SGD web server is preconfigured to use the same SSL certificate as the SGD server.

Users are able to install and use their own security certificates for securing connections to SGD.

Connections between the SGD servers in the array are secured using an SSL certificate that has been signed by the primary SGD server in the array. The primary server acts as the trusted certificate authority (CA).

The following table includes some security topics for installing SGD.

Table 2.1. SGD Installation Topics

Topic	More Information
Installing SGD. The SGD software is supplied as a package file. When you install SGD, connections to the SGD server and between the SGD servers in the array are secured using SSL.	• Installing the Main SGD Component
Installing the SGD Gateway. The Gateway is a secure proxy server for SGD.	• Installing the SGD Gateway
Installing the SGD Client manually. The SGD Client is usually installed automatically when a user connects to an SGD server using a browser. With manual installation, you have full control over where the SGD Client is installed.	• Manual Installation of the SGD Client
Removing SGD. To remove SGD, you remove the components installed on hosts, on application servers, and on client devices.	• Removing SGD

2.2. Post Installation Configuration

Following a default installation, SGD is configured to use secure connections. However, there are some optional post-installation tasks that you might want to do to improve security.

The following table includes some topics for post-installation tasks.

Table 2.2. Optional Post-Installation Tasks

Topic	More Information
Secure communications. During installation, a Secure Sockets Layer (SSL) certificate is installed and the SGD server is configured to use SSL for secure communications. Following installation, you might want to use a different SSL certificate. For example, if you do not specify certificate details during installation, a self-signed SSL certificate is created and installed automatically. If you are using SGD in a production deployment you must replace any self-signed certificates with certificates signed by a recognized Certificate Authority (CA). **Caution** Only use self-signed SSL certificates in a test environment. For a production deployment, always use certificates signed by a recognized CA.	• Section 3.3, “Secure Connections to SGD Servers” • Section 3.6, “Tuning Secure Connections”
Cryptographic algorithms. You can configure SGD to use cryptographic algorithms that meet your security needs.	• Section 3.6, “Tuning Secure Connections”
Protecting sensitive data. Files that contain sensitive information have restrictive permissions by default. You may want to review the file permissions to suit your security policy.	• Section 5.6, “SGD Installations”
SGD Gateway. The SGD Gateway is a proxy server specially designed to improve the security of an SGD installation.	• Section 1.3, “The SGD Gateway”

Chapter 3. Network Security for SGD

This chapter includes security topics for integrating Oracle Secure Global Desktop (SGD) into your network infrastructure and securing the network connections used by SGD.

3.1. Network Connections for SGD

When using SGD, client devices never connect directly to application servers. Instead they connect to SGD using HTTP or HTTPS and the SGD Adaptive Internet Protocol (AIP). SGD then connects to the application servers on the user's behalf.

The following table describes the main network connections involved when using SGD.

Table 3.1. Main Network Connections for SGD

Connection	More Information
Connections between client devices and SGD servers	• Connections Between Client Devices and SGD Servers
Connections between SGD servers and application servers	• Connections Between SGD Servers and Application Servers
Connections between SGD servers in an array	• Connections Between SGD Servers in an Array

3.2. Firewalls and Ports

Firewalls can be used to protect various parts of a network and must be configured to allow the connections required by SGD.

The following table describes the firewall configurations you need to consider when deploying SGD. The table provides links to information about the ports you might need to open to allow connections when using SGD.

Table 3.2. Firewall Configurations for SGD

Connection	More Information
Firewalls between client devices and SGD servers. Client devices must be able to make HTTPS connections and AIP connections to any SGD server in the array.	• Firewalls Between Client Devices and SGD Servers
Firewalls between SGD servers in an array. The SGD servers in an array must be able to connect to any other member of the array.	• Firewalls Between SGD Servers
Firewalls between SGD servers and application servers. An SGD server must be able to connect to an application server in order to run applications.	• Firewalls Between SGD Servers and Application Servers
Other firewalls. SGD needs to make connections to any authentication services and directory services you might be using.	• Other Firewalls

3.2.1. Using a Port Scanner

When you deploy an SGD server in a production environment, it is a good idea to use a port scanner such as `nmap`, to ensure there are no unnecessary services or ports in use on the SGD host.

3.3. Secure Connections to SGD Servers

You secure connections to an SGD server by installing an SSL certificate on the SGD server and enabling SGD security services. This is done automatically when you install SGD.

The following table describes some features of secure connections for SGD.

Table 3.3. Secure Connections to SGD Servers

Feature	More Information
SSL certificates. When secure connections are enabled, an SGD server requires an SSL certificate.	• SSL Certificates
Firewall traversal. AIP connections to an SGD server are made on TCP ports 3144 and 5307. If it is not possible to open the required ports in your firewalls, you can direct all SGD traffic through a single port, usually port 443.	• Firewall Traversal
Enabling secure connections (automatic configuration). The `tarantella security enable` command enables you to quickly configure and enable secure connections.	• Enabling Secure Connections (Automatic Configuration)
Enabling secure connections (manual configuration). In some cases you need to secure connections using manual configuration.	• Enabling Secure Connections (Manual Configuration)
Security warnings. When using secure connections to SGD, users see certificate security warnings.	• Secure Connections and Security Warnings

3.4. Secure Connections Between SGD Servers

Connections between the SGD servers in an array are secured using SSL. This is called *secure intra-array communications*.

See Secure Intra-Array Communication for more details.

3.5. Secure Connections to Application Servers

By default, SGD uses SSH to provide secure connections between SGD servers and *UNIX or Linux system application* servers.

SSH provides the following benefits:

- All communication between application servers and SGD servers using SSH is encrypted, including the X protocol if you are running X applications
- User names and passwords are always encrypted before being transmitted over the network

See Using SSH for more details about using SSH with SGD.

By default, *Windows applications servers* use enhanced network security, such as Network Level Authentication (NLA) or Transport Layer Security (TLA). If enhanced network security is not configured or supported on the application server, Windows applications use the Microsoft Remote Desktop (RDP)

protocol. This means that all communication is encrypted, and connections to Microsoft Windows application servers are secure. See Section 4.2.3, “Windows Applications”.

3.6. Tuning Secure Connections

The following table describes the tuning that can be applied to secure connections to SGD servers.

Table 3.4. Tuning Secure Connections

Feature	More Information
SSL Daemon. The SSL Daemon is the SGD component that handles secure connections to SGD servers.	• Tuning the SSL Daemon
SSL accelerators. Performance can be improved by off-loading the processor-intensive transactions required for SSL connections to an external SSL accelerator.	• Using External SSL Accelerators
Cipher suites. You can select the ciphers that are used for secure connections to SGD servers.	• Section 3.6.1, “Configuring Ciphers”
Connection definitions. Connection definitions can be used to control whether a secure or a standard connection is used when connecting to an SGD server.	• Using Connection Definitions

3.6.1. Configuring Ciphers

SGD servers and the SGD Gateway are configured by default to use strong ciphers for secure communication.

You may want to disable the use of unneeded or weak ciphers as follows:

- **SGD servers.** Configure the cipher suites that can be used for secure connections to SGD servers. See Selecting a Cipher Suite for Secure Connections.
- **SGD Gateway.** Configure the cipher suites that can be used for secure connections to the SGD Gateway. See Configuring Ciphers for the SGD Gateway.

3.7. 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.

Using the SGD Gateway is the preferred alternative to running your SGD servers with firewall traversal, also called firewall forwarding. See Section 3.8, “Firewall Traversal”.

Using the Gateway has the following advantages, compared to using firewall traversal:

- The Gateway can be deployed away from the SGD servers on the internal network.
- All connections can be authenticated in the DMZ before any connections are made to the SGD servers in the array.
- The Gateway manages load balancing of HTTP connections, so you do not need to use the JSP technology load balancing page included with SGD.

See About the SGD Gateway for more information about how to install and configure the SGD Gateway.

3.8. Firewall Traversal

If you are not using the SGD Gateway, you can use *firewall traversal* to give users access to SGD using a single port. With firewall traversal, you configure the SGD server to listen on port 443. The SGD server then forwards all traffic that is not AIP traffic to the SGD web server.

See Firewall Traversal for more information.

Chapter 4. Security for Users, Applications, and Clients

This chapter describes the mechanisms for authenticating to an Oracle Secure Global Desktop (SGD) server to log in to SGD. This is known as *Secure Global Desktop authentication*.

Security aspects of publishing and configuring SGD applications are also covered. Finally, using peripherals and other client devices with SGD is described.

4.1. Authenticating Users

The following table describes the supported mechanisms for authenticating users to an SGD server.

Table 4.1. User Authentication Mechanisms

Authentication Mechanism	More Information
System authentication. SGD checks the user's credentials against one or more external authentication services, for example a Lightweight Directory Access Protocol (LDAP) directory.	• Anonymous User Authentication • UNIX System Authentication • LDAP Authentication • Active Directory Authentication • SecurID Authentication
Third-party authentication. An external mechanism authenticates the user and SGD trusts that the authentication is correct. The most common use of third-party authentication is web server authentication.	• Third-Party Authentication and Web Authentication

4.1.1. Password Security

When logging in to SGD, passwords are only encrypted if there is an HTTPS connection. By default, the SGD server is configured for HTTPS connections.

SGD uses external mechanisms for authenticating users. The security of passwords when authenticating users is as follows:

- Active Directory authentication uses the Kerberos protocol for authentication, which is secure
- LDAP authentication can be configured to use a secure connection
- Web server authentication is only secure if the user has an HTTPS connection
- All other authentication mechanisms use the native protocols for authenticating users

4.1.2. Two-Factor Authentication

For enhanced authentication security, you can use the RSA SecurID two-factor authentication system to authenticate SGD users. In SGD this is called SecurID authentication.

RSA SecurID uses two-factor authentication based on something you *know*, a PIN, and something you *have*, a tokencode supplied by a separate token such as a PIN pad.

SecurID authentication enables users with RSA SecurID tokens to log in to SGD. SGD authenticates users against an RSA Authentication Manager.

See SecurID Authentication for details of how to configure SGD to use SecurID authentication.

See Chapter 6, *Troubleshooting an SGD Deployment* for more examples of using two-factor authentication with SGD.

4.2. Objects and Applications

SGD uses organizational hierarchies to manage users and give them access to applications.

4.2.1. Organizations and Objects

SGD is built on the principles of directory services. Users, applications, and application servers are represented by *objects* in a directory. The objects are arranged into an *organizational hierarchy* representing your organization.

See Designing the Organizational Hierarchy for details of how the authentication mechanisms you use can affect your organizational hierarchy.

4.2.2. SGD Administrators

An SGD Administrator is a user who has special privileges to create users and manage an SGD array.

In SGD, administration privileges are managed using the Global Administrators role object in the System Objects organization.

See SGD Administrators for details of how to add and remove SGD Administrators.

4.2.3. Windows Applications

Windows applications in SGD can use the following security features of Microsoft Windows Remote Desktop Services.

- Authentication settings
- Encryption level
- Transport Layer Security (TLS)
- Network Level Authentication (NLA)

See Configuring Microsoft Windows Remote Desktop Services for Use With SGD for more details of Remote Desktop Services security features supported by SGD.

4.2.4. X Applications

By default, SGD secures X displays using X authorization. This prevents users from accessing X displays that they are not authorized to access.

4.2.5. Integrating With Oracle VDI

SGD provides the following methods of integrating with Oracle Virtual Desktop Infrastructure (Oracle VDI).

- **Using a broker.** SGD includes virtual server brokers (VSBs) that enable users to access desktops provided by an Oracle VDI server.

  Connections to Oracle VDI desktops are secured using Remote Desktop Protocol (RDP).

The VDI broker for Oracle VDI 3.3 and later uses the VDI web services API to authenticate users, obtain a list of desktops, and start and stop a desktop. Web services connections are secured using HTTPS.

- **Using a Windows application object.** This method can be used if you are unable to use either of the brokers supplied with SGD.

  Connections to Oracle VDI desktops are secured using RDP.

See Integrating SGD With Oracle VDI for details of integrating SGD with Oracle VDI.

See the Oracle VDI Documentation for more information about securing Oracle VDI.

4.2.6. Application Authentication

When a user clicks a link to start an application, SGD connects to the application server, handles the authentication process, and starts the application.

By default, SGD stores the user names and passwords used to run applications in its application server password cache. SGD also stores the user names and passwords used to log in to SGD.

Entries in the application server password cache are encrypted with an encryption key. When starting applications, the passwords are decrypted as they are needed. See The Application Server Password Cache for more details.

For Windows applications, the Remote Desktop Session Host handles the authentication process.

See Application Authentication for more details about how application authentication works in SGD.

4.3. Client Device Security

The following table includes security topics for the client device services supported by SGD.

Table 4.2. Security Topics for Client Device Services

Service	More Information
Printing. Users can print to PDF or to a printer attached to the client device.	• Configuring an SGD Server for Printing • Configuring Printing to Microsoft Windows Client Devices • Configuring Printing to UNIX, Linux, and Mac OS X Platform Client Devices
Client drive mapping. Client drive mapping (CDM) enables users to access the drives on their client device from applications running on application servers.	• Enabling CDM Services in SGD • Configuring the Client Drives Available to Users
Copy and paste. Copy and paste is supported for applications displayed through SGD.	• Using Copy and Paste • Controlling Copy and Paste in Applications • An Example of Using Clipboard Security Levels
Smart cards. Users can access a smart card reader attached to their client device from applications running on a Windows application server.	• Using Smart Cards With Windows Applications • Setting Up Access to Smart Cards

Service	More Information
Serial ports. Users can access serial ports on the client device from Windows applications displayed through SGD.	• Enabling Serial Port Access in SGD • Configuring the Client Device

4.3.1. Using the SGD Client

The SGD Client is the part of SGD that is installed on client devices. The SGD Client is required to run applications.

The following table includes security topics for the SGD Client.

Table 4.3. Security Topics for the SGD Client

Topic	More Information
Installing the SGD Client. The SGD Client can be installed automatically or manually. If your organization prefers not to use Java technology, the SGD Client must be manually downloaded and installed.	• Automatic Installation of the SGD Client • Manual Installation of the SGD Client
Running the SGD Client. The SGD Client normally runs automatically, but it can be run from the command line.	• Running the SGD Client From the Command Line • Using SGD Without Java Technology
Client profiles. A *client profile* is a group of configuration settings that control the SGD Client.	• Client Profiles and the SGD Client • Managing Client Profiles • How to Configure Client Profile Editing for Users

Chapter 5. Security for SGD Servers and Arrays

This chapter includes security topics for Oracle Secure Global Desktop (SGD) servers and arrays. Supported logging mechanisms and security-related commands are described.

5.1. SGD Arrays

In SGD, an *array* is a collection of SGD servers that share configuration information.

In the array, each SGD server has a peer DNS name and one or more external DNS names. SGD servers always use peer DNS names to communicate with each other. See DNS Names.

By default, SGD is installed with *secure intra-array communication* enabled. This means that connections between the SGD servers in an array are encrypted. See Secure Intra-Array Communication.

5.2. SGD Web Server

When you install SGD, the SGD web server is also installed. See Introducing the SGD Web Server.

Because the SGD web server is an Apache web server preconfigured for use with SGD, you can use Apache directives to configure it.

If you require enhanced security, a more secure version of the `httpd.conf` Apache configuration file used by the SGD web server is supplied. See Securing the SGD Web Server.

5.3. Administration Console

The Administration Console is an administration tool for SGD. The Administration Console enables an SGD Administrator to manage users and user sessions. SGD server configuration, and the configuration of applications for SGD users can be done using the Administration Console. See Using the Administration Console.

You can secure access to the Administration Console. See Securing Access to the Administration Console.

5.4. Monitoring and Logging

The following table includes logging and monitoring topics for SGD.

Table 5.1. Logging and Monitoring Topics for SGD

Topic	More Information
Log filters. When you first install SGD, the default log filters log all errors on the SGD server. If you want to obtain more detailed information, for example to troubleshoot a problem, you can set additional log filters.	• Using Log Filters to Troubleshoot Problems With an SGD Server • Using Log Filters for Auditing • Using Log Filters to Troubleshoot Problems With Protocol Engines
Web server logging. SGD web server messages are recorded in the following logs: • Tomcat JSP technology container logs • Apache web server logs	• SGD Web Server Logging

Topic	More Information
SGD Client Logging. By default, log messages for the SGD Client are stored to a file on the client device.	• SGD Client Logging

5.5. SGD Server Certificate Stores

Each SGD server has two certificate stores, a CA certificate truststore and a client certificate store.

The following table describes the SGD server certificate stores.

Table 5.2. Certificate Stores on an SGD Server

Certificate Store	More Information
The CA certificate truststore. The CA certificate truststore contains the CA certificates that the SGD server trusts.	• The CA Certificate Truststore
The Client Certificate Store. The client certificate store contains the client certificates that an SGD server uses to identify itself when connecting to another server.	• The Client Certificate Store

5.6. SGD Installations

SGD can be installed in a standard location or a user-specified directory.

The standard installation directory for SGD is `/opt/tarantella`.

During SGD installation, you have the option of specifying a different installation directory.

About Your SGD Installation includes details of the files and directories included in an SGD installation.

You may want to review these files in the context of your security policy.

5.7. SGD Commands

SGD includes a built-in command set for controlling and configuring SGD.

The following table describes some security-related commands for SGD.

Table 5.3. Security-Related Commands for SGD

Command	More Information
`tarantella`. Administrators can control SGD from the command line using the `/opt/tarantella/bin/tarantella` command.	• The tarantella Command
`tarantella passcache`. Manipulates the application server password cache. SGD Administrators can create, modify, delete, and examine entries.	• tarantella passcache
`tarantella query`. Examines the SGD server's log files.	• tarantella query
`tarantella restart`. Stops and then restarts services on the SGD server, prompting if users are currently connected.	• tarantella restart

Command	More Information
`tarantella security`. Controls SGD security services and manages server certificates.	• tarantella security
`tarantella start`. Starts SGD services on the host.	• tarantella start
`tarantella status`. Reports SGD server information.	• tarantella status
`tarantella stop`. Stops SGD services on the SGD host, prompting if users are currently connected.	• tarantella stop
`tarantella webserver`. Configures trusted users for the third-party authentication mechanism.	• tarantella webserver
`tarantella webtopsession`. Enables SGD Administrators to list and end user sessions.	• tarantella webtopsession

Chapter 6. Troubleshooting an SGD Deployment

This chapter provides a checklist of common techniques and strategies that administrators can use in support of secure SGD deployments. Note that some of the strategies might not be appropriate for your environment.

6.1. Operating System Environment

This section includes some suggestions for securing the operating system (OS) on the SGD host.

- **Install the operating system off-network.** Install, patch, and configure your server OS while you are disconnected from the network. This prevents your system from being detected, attacked, and compromised before you have finished OS installation.
- **Use disk partitions**. Use separate partitions for directory structures that may fill up your root file system, which can be a form of a Denial Of Service (DOS) attack.

  For example, if you use the default SGD installation directory, `/opt/tarantella`, you might want to do the following:

  - Create a separate `/opt` partition to store the SGD binaries and log files.
  - Relocate the SGD server, Apache, and Tomcat log file locations to separate partitions.
  - Move the `/opt/tarantella/var` directory to a separate partition.
- **Minimize OS installation.** Only install the software and services that you require.

  Do not install tools that an attacker can use to further their attacks. Such tools include `C` compilers, which can be used to compile root kits, and network utilities such as `ping`, `nslookup`, and `telnet`.
- **Minimize the network services footprint.** Eliminate unnecessary network services, to reduce the number of attack points that an attacker may try to exploit.

  Oracle Solaris 10 11/06 (update 3) and later provides a Secure By Default option at installation time, which has a reduced network services footprint. This option can also be enabled after installation by using the following command:

  ```
  # netservices limited
  ```
- **Use Oracle Solaris zones.** Create a non-global Oracle Solaris zone to install and run SGD in. Even if an attacker manages to compromise the SGD zone, forensic evidence of the attack should still be available in the global zone.
- **Use time source synchronization.** Using a synchronized time source makes it easier to correlate security event logs. Synchronization of system clocks is also a requirement for SGD arrays, the SGD Gateway, and Kerberos authentication.

  If possible, use Network Time Protocol (NTP) software to synchronize clocks. Alternatively, use the `rdate` command.
- **Disable Routing and Forwarding.** On a multi-homed system, disable all routing functions.

6.2. SGD Configuration

The following topics describe how to make your SGD configuration more secure.

6.2.1. Install SGD in Secure Mode

By default, SGD is installed in secure mode. This means that the SGD server is configured automatically to use secure connections between client devices and the SGD server, and to use secure connections between the SGD servers in the array.

During installation, you can specify your own security certificate for securing connections to SGD.

See Chapter 2, *Secure Installation and Configuration of SGD* for more details.

6.2.2. Use a Non-Root Administrator Account

Always use a non-root account to administer the SGD array. See Section 4.2.2, “SGD Administrators” for details of how to create SGD Administrators.

Disable root user logins for the SGD host. Administrators who need root access should log in using a non-root account and use the `su` command or equivalent.

Note that some `tarantella` commands, such as those that control the SGD server and SGD web server can be run only by the root user. See The tarantella Command.

6.2.3. Use Firewall Traversal

Firewall traversal is the multiplexing of both the HTTPS and Adaptive Internet Protocol (AIP) protocols onto a single TCP port (port 443). Using this technique means that you do not have to reconfigure your firewalls to allow SGD to operate across them.

SGD supports the following methods of firewall traversal:

- **The SGD Gateway.** The SGD Gateway is a reverse proxy server that can be used to provide an increased level of security between client devices and SGD servers. All client connections to the Gateway are made on port 443. The Gateway is included in the SGD distribution.
- **Firewall forwarding.** For firewall forwarding, you configure the SGD server to listen on port 443. The SGD server then forwards all traffic that is not AIP traffic to the SGD web server.

6.2.4. Do Not Use Self-Signed Certificates

Self-signed server certificates are for testing purposes only. Because client computers do not recognize the Certificate Authority (CA) used to sign the presented server certificate, the browser shows a security warning and gives the user the option to override the warning. Instruct users that they should never override such security warnings.

SGD does include a host verification sequence for the client device. When the X.509 server certificate installed on the SGD server changes, users are warned that they may be victims of a host spoofing attack. This is because a fingerprint of each SGD server's X.509 certificate is stored on the client device whenever a user allows a connection to an SGD host. On subsequent connections, the stored fingerprint is compared to the certificate fingerprint presented by the server. If they are not the same, a dialog box warning the user of a possible spoofing attack is displayed.

Instruct your users to never proceed if such a message is displayed. When you are going to replace a server certificate, be sure to warn your users in advance.

6.2.5. Use SSL and TLS

Always secure connections between client devices and SGD servers using Secure Sockets Layer (SSL) or Transport Layer Security (TLS). Both HTTP and AIP connections should be secured.

For a default installation, an SGD server is configured automatically to use SSL or TLS for all connections with the client device.

6.2.6. Use Secure Session Cookies

By default, if secure connections are being used, SGD marks all user session cookies as secure. This prevents transmission of the cookie over a non-secure connection.

Wherever possible, you should always use secure connections for your SGD deployment. However, in some circumstances you might want cookies to be marked as secure, even though the connection is not secure. For example:

- In a SGD Gateway deployment, if you are using unencrypted connections to the SGD array
- If you are using an SSL accelerator that connects to SGD using HTTP

To mark all cookies as secure, edit the Tomcat configuration file `/opt/tarantella/webserver/tomcat/tomcat-version/conf/server.xml` on the SGD host.

Add the attribute `secure="true"` to the AJP and HTTP Connector elements. For example:

```
<!-- Define an AJP 1.3 Connector on port 8009 -->
  <Connector port="8009" protocol="AJP/1.3" redirectPort="8443"
   secure="true" allowTrace="false" />
```

Following this change, all requests made using these Connectors will appear to be secure, meaning that Tomcat's `JSessionID` cookie and all SGD user session cookies will be marked as secure.

See the Apache Tomcat documentation for more details about the `secure` attribute.

6.2.7. Restrict the Use of Weak SSL Ciphers

By default, the SGD web server is configured to use only strong cipher suites.

The supported OpenSSL cipher suites are configured by the `SSLCipherSuite` directive in the `httpd.conf` file for the SGD web server.

```
SSLCipherSuite HIGH:!SSLv2:!ADH:!aNULL:!eNULL:!NULL
```

This setting means that browsers must support higher-grade encryption.

By default, the SGD Gateway is configured to use only high grade ciphers for SSL connections.

To configure the cipher suites used by the Gateway, you must edit the configuration file called `ciphersuites.xml` in the `/opt/SUNWsgdg/etc/` directory on the Gateway. See Configuring Ciphers for the SGD Gateway for more details.

6.2.8. Disable Unencrypted AIP Communications

In a default installation, the SGD Client uses encrypted connections when connecting to the SGD server. To make certain that only encrypted AIP connections are used, you can disable unencrypted connections. Unencrypted connections are also called standard connections.

In the SGD Administration Console, click on the SGD server name in the Secure Global Desktop Servers tab and deselect the Standard check box in the Security tab.

Alternatively, use the following command:

```
# tarantella config edit --security-connectiontypes ssl
```

6.2.9. Enable Secure Intra-Array Communication

When you create an SGD array, the traffic replicated between SGD servers might not be encrypted. To encrypt this traffic, enable secure intra-array communication for all SGD servers in the array.

For a default installation, secure intra-array communication is enabled automatically for an SGD server.

6.2.10. Securing the SGD Web Server

The SGD web server is an Apache web server preconfigured for use with SGD. The following topics describe how to enhance security for the SGD web server.

6.2.10.1. Use the Secure Apache Configuration File

The `httpd.conf.secure` file is an Apache server configuration file that configures the SGD web server for enhanced security. The file is included with the SGD software, at `/opt/tarantella/webserver/apache/`*`apache-version`*`/conf/httpd.conf.secure` on the SGD host.

The `httpd.conf.secure` file provides the following additional security features, compared to the standard `httpd.conf` file used by the SGD web server:

- Apache modules that are not used by SGD are disabled
- Access to the `/cgi-bin` directory on the SGD web server is not allowed

6.2.10.2. Redirect Connections to a Secure Port

When a user connects to the unencrypted HTTP port (TCP port 80), the connection should be redirected automatically to the HTTPS port (TCP port 443).

You might want to disable connections on port 80. But this can cause problems, as any user that forgets to specify `https://` in the URL will see an error message.

The following are some mechanisms that you can use to redirect users to a secure port.

- **Install in secure mode.** For a default installation, SGD is configured automatically to redirect HTTP connections to port 443.

  The SGD web server has a preconfigured rule that uses the Apache `mod_rewrite` module to redirect users. The following rule is enabled automatically when you install in secure mode.

```
# SGD BEGIN AUTO-FORWARD TO HTTPS (don't delete this line!)
   <IfModule rewrite_module>
     RewriteEngine On
     RewriteCond %{SERVER_PORT} !^443$
     RewriteRule (/.*) https://%{SERVER_NAME}$1
   </IfModule>
# SGD END AUTO-FORWARD TO HTTPS (don't delete this line!)
```

- **Use a `VirtualHost` directive.** Connections can be redirected using an `httpd.conf` entry such as:

```
<VirtualHost *:80>
     Redirect / https://sgd-server.example.com/
</VirtualHost>
```

- **Use the Load Balancing JSP.** If you are using the Load Balancing JSP to distribute user sessions, specify `https` URLs for your target servers, for example:

```
hosts[0] = "https://www1.example.com"
```

```
hosts[1] = "https://www2.example.com"
```

- **Use a redirection page.** Replace the default SGD web server Welcome Page `index.html`, as follows:

```
<html>
    <head>
      <title> SGD Redirect Page</title>
      <meta HTTP-EQUIV="Refresh" content="0; url=https://server.example.com/sgd">
    </head>
</html>
```

6.2.10.3. Hide Links That Are Not Required

The SGD web server Welcome Page contains a number of links for logging in to the Administration Console, downloading software, and viewing user documentation. You might want to hide these links from users.

The technique for specifying a redirection page described in Section 6.2.10.2, “Redirect Connections to a Secure Port” redirects the connection to the SGD login page `https://server.example.com/sgd`. This prevents the user from seeing the SGD web server Welcome Page.

By redirecting users directly to the SGD login page, you remove the temptation for users to try other links. This helps to avoid problems such as users changing their locale to one which their application is not configured for.

6.2.10.4. Display Legal Warnings

Some people may interpret the presence of the Log In link on the SGD web server Welcome Page as an *invitation* to log in, even if they have no business in doing so. The reasoning is that, because they were invited to log in, the owner of the site has no basis for taking action against an unwanted visitor.

If you are publishing applications on the Internet, you might want to alter the default SGD web server Welcome Page to include an appropriate legal warning.

6.2.10.5. Eliminate Unnecessary Web Server Content

The Apache Axis distribution included with SGD includes some example code and scripts meant to assist web services developers. To restrict access to these files, move unnecessary files from the `/axis` web directory, as follows:

```
# cd /opt/tarantella/webserver/tomcat/tomcat-version/webapps/axis
# mv *.jsp WEB-INF/
# mv *.html WEB-INF/
```

6.2.10.6. Reduce Web Server Response Information

By default, the Apache web server used by SGD returns information such as the version number in the server response header. This information could be used by scanning scripts to fingerprint the web server.

You can reduce the amount of information returned in the response headers by using the following directives in the `httpd.conf` file.

```
ServerTokens Prod
ServerSignature Off
```

Note that `ServerTokens Prod` still returns the web server name (`Server: Apache`). If you wish to further obscure your web server identity, install the `mod_security` Apache module, and set the `SecServerSignature` string to an arbitrary identifier.

`mod_security` contains a number of other useful logging and intrusion prevention features. See Section 6.3.3, “Intrusion Detection and Prevention Systems”.

6.2.10.7. Disable Unused Apache Modules

The version of Apache web server supplied with SGD includes a number of Dynamic Shared Objects (DSO) to provide optional extensions to the web server.

Some of these modules are not needed by SGD. For example, `mod_dav`, `mod_dav_fs`, and `mod_userdir` can be removed by commenting out the relevant `LoadModule` directive in the `httpd.conf` file. Note that when removing such modules, any directives in the `httpd.conf` file that the module handles will no longer be recognized.

Where possible, use the secure Apache configuration file `httpd.conf.secure`. Apache modules that are not used by SGD are disabled in this file. See Section 6.2.10.1, “Use the Secure Apache Configuration File”.

6.2.10.8. Disable Autocomplete for Web Pages

Autocompletion of user input can be disabled for the SGD login page and the Administration Console login page. Disabling autocomplete prevents browser caching of sensitive data, such as user names and password.

To disable autocomplete, edit the `/opt/tarantella/webserver/tomcat/`*`tomcat-version`*`/conf/web.xml` file and change the value of the `disableloginautocomplete` parameter to `true`. This parameter is `false` by default. Restart the SGD web server after making changes.

6.2.10.9. Control Web Server Log File Sizes

The Apache and Tomcat log files can, over time, become quite large. To avoid consuming excessive amounts of disk space, consider putting these logs under the control of `logadm` (Oracle Solaris platforms) or `logrotate` (Linux platforms).

Be aware that attackers can use log files to compromise systems. For example, deliberately generating repeated log file entries can fill up a file system, and can act as a Denial Of Service (DOS) attack. Similarly, attackers might generate seemingly innocuous log file entries to overload an administrator with data, and thus obscure log file entries that might point to more nefarious activities.

6.2.11. Disable "Show Details" for Application Launches

Displaying launch details can be useful when diagnosing launch failures. However, it is sometimes possible for the user password sent to the application server to be displayed in the Launch Details window.

In practice this issue rarely occurs, as only UNIX application servers might be affected, the details shown are transient in nature, and the information is not logged. However, if you have pre-populated user password caches and have not revealed the credentials to your users, a user may see details when starting an application .

To prevent launch details from being displayed, go to the Global Settings → Application Authentication tab in the Administration Console. Deselect the following settings for the "Launch Details Pane" attribute:

- Showed by Default
- Showed When Launch Fails
- Show/Hide By User Enabled

6.2.12. Restrict Access to the Administration Console

The Administration Console is a web application running on the SGD web server which is used by SGD Administrators to configure and control the SGD servers in an array.

The following methods can be used to restrict access to the Administration Console:

- **Remove the Administration Console link from the SGD web server Welcome Page.** Edit the following HTML file for the SGD web server Welcome Page, at:

  `/opt/tarantella/webserver/apache/apache-version/htdocs/webtop_locale.html`, where *`locale`* is the language for the Welcome Page.

  Comment out the following code that creates the link to the Administration Console at `/sgdadmin`.

```
<tr>
  <td>
   <p>
    <a href="/sgdadmin"
    alt="Launch the Oracle Secure Global Desktop Administration Console"
    title="Launch the Oracle Secure Global Desktop Administration Console">Launch the
    Oracle Secure Global Desktop Administration Console</a>
     ...
</tr>
```

  Note that after making this change, users can still access the Administration Console by going to `http://server.example.com/sgdadmin`, where *`server.example.com`* is the name of the SGD server.

- **Restrict access by IP address.** Use the Apache `<Location>` directive to only allow access to the `/sgdadmin` URL from specific client IP addresses. This is described in Securing Access to the Administration Console.

- **Use client certificates.** A client certificate is an SSL certificate that is installed in the browser on the client device. You can restrict access to only those browsers which have a valid client certificate.

  To configure the SGD server for client certificates, do the following:

  - Copy the CA certificate file for the client certificate to a location on the SGD server.
  - Add the following to the SSL Virtual Host Context section of the `httpd.conf` file:

```
<Location /sgdadmin/*>
 SSLVerifyClient require
 SSLVerifyDepth 1
</Location>
```

  - Set the `SSLCACertificateFile` directive to the location of the CA certificate file. For example:

```
SSLCACertificateFile /opt/tarantella/var/tsp/client/CA.crt
```

6.2.13. Restrict Access to Client Device Features

You might want to limit the ability of users to transfer information from an application displayed using SGD to their client devices.

Mechanisms for transmitting data include printing, copy and paste, and client drive mapping. You might want to consider disabling or otherwise restricting the ability of users to use these client device features. See Section 4.3, “Client Device Security”.

6.2.14. Create an Audit Trail

To record what users are doing, and attempting to do, you can enable audit logging for an SGD server. Audit logging records system events such as starting and stopping the SGD server, configuration changes, user logins, and running applications.

Audit logging is configured using a `*/*/*auditinfo` log filter. See Using Log Filters for Auditing.

You use the `tarantella query audit` command to view audit log files.

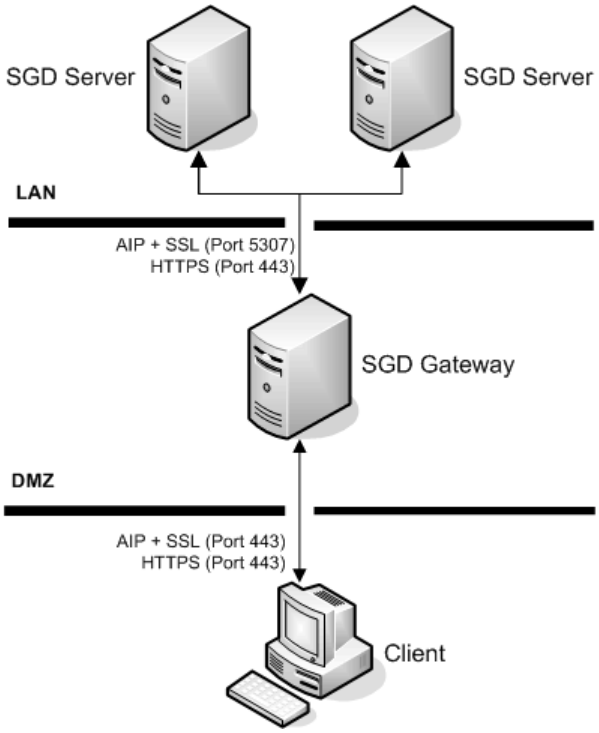
6.3. Supporting Services

This section describes how you can secure supporting services used by SGD, such as authentication and application hosting. Information on using intrusion detection systems and penetration testing of servers is also included.

6.3.1. Firewall Policies

For a basic SGD deployment using a single Gateway, you generally need to use two firewalls, as shown in Figure 6.1, “Firewalls for a Gateway Deployment”.

Figure 6.1. Firewalls for a Gateway Deployment



The external firewall need only be configured to pass encrypted traffic on port 443 to the Gateway. The LAN firewall then passes encrypted traffic to the SGD servers in the array, on port 443 and port 5307.

Additional internal firewalls may also need to be configured to pass traffic on other ports for services used by SGD. For example you might need to allow RDP traffic for connections to Windows application servers, and LDAP traffic for user authentication.

The source IP address for connections to other services should only be from SGD servers. You should limit the allowed destination IP addresses to be the addresses of specific servers.

Your firewalls should also provide basic protection against Denial Of Service Attacks, (DOS) and Distributed Denial of Service (DDOS) such as "Ping of Death", SYN Flood, Ping Flood, `smurfd`, and similar attack types. Note that most so called "application firewall" technologies are not useful until the traffic is decrypted.

See Section 3.2, “Firewalls and Ports” for more information about the ports used by SGD.

6.3.2. Use Two-Factor Authentication for Internet Deployments

If you are going to publish applications to users on the Internet, it is strongly advised that you use a two-factor authentication system, such as RSA SecurID.

For example, if a user uses an uncontrolled client device, such as a computer at an Internet cafe, it is possible for software or hardware on the client device to capture the user's keystrokes. A user may reveal their username and password when using such a compromised system.

Use of a two-factor authentication with a one-time password component will prevent such stolen credentials from being used successfully in an attack.

SGD supports the following methods of implementing two-factor authentication:

- **RSA SecurID authentication.** In SGD this is called SecurID authentication. See Section 4.1.2, “Two-Factor Authentication”.
- **Client certificates.** These can be used to enhance the security of the SGD Gateway, by restricting access to those users who have a valid certificate. See Using Client Certificates With the SGD Gateway.
- **External authentication mechanisms**. Configure SGD to use an external authentication mechanism such as RADIUS that supports two-factor authentication. See Third-Party Authentication and Web Authentication.

6.3.3. Intrusion Detection and Prevention Systems

Intrusion detection systems (IDS) and intrusion prevention systems (IPS) are designed to monitor network traffic and look for patterns and behaviors that may indicate an unwanted intrusion is being attempted. The IPS system differs from the IDS system in that the IPS system takes active measures to immediately counter the perceived attack, by blocking the offending source IP address, resetting the connection, or through other mechanisms.

IDS and IPS systems can generally be broken down into host-based solutions, and network-based solutions. Host-based solutions are installed on the systems they are meant to protect, while network-based solutions will usually monitor traffic at one or more points "in front of" the host or hosts being protected. These devices may act in-line, so that all traffic must pass through them, or may simply act as passive sensors, often with no assigned IP address, which monitor traffic by setting its network interface in promiscuous mode.

Some IDS and IPS systems perform protocol decoding, while others do not. Protocol decoding for SGD would primarily examine HTTP traffic. In order to do this effectively, the IDS or IPS system must be able to view the traffic in unencrypted mode.

One mechanism to do this is to use an SSL accelerator, which decrypts the HTTP traffic, and forwards it onto the SGD server. The network sensor is placed on the network between the accelerator and the SGD server.

For more information on configuring SGD for an SSL accelerator, see Using External SSL Accelerators.

Two IDS and IPS tools that can prove useful for SGD deployments are:

- **`mod_security`**. This is an example of a host-based IPS. More specifically, it is a web application firewall, because it specifically monitors and evaluates web traffic.

  `mod_security` is a plug-in to the Apache web server, and comes with a set of "core" rules that detect protocol violations and known attack signatures to prevent web-based attacks. For more info, visit the `mod_security` home page.

- **SNORT.** This program has a variety of modes, including packet decoding, but is most commonly deployed as a network-based IDS and IPS tool. SNORT is claimed to be the most widely deployed IDS in the world, and is an open-source project.

6.3.4. Perform Penetration Testing

When deploying SGD on the Internet, it is recommended that you perform active penetration testing to ensure you have covered the most obvious issues.

Penetration testing is often performed by outside consultants, while some organizations have their own internal test teams. If you wish to perform some basic tests on your own, consider some of the following tools:

- **`nmap`.** A utility that is commonly used to scan for hosts, determine the OS, and determine what services a host may be offering. An intruder would use a tool like this to perform network reconnaissance, the first step in attempting to break-in. Be sure to provide as little information about your installation to would-be intruders as possible.

- **nessus.** An automated vulnerability scanner which uses plug-ins to test for specific types of vulnerabilities, such as cross-site scripting vulnerabilities, known web server bugs, and so on. This is free for home use only.

- **nikto.** An automated web server scanner, which also tests for a variety of web server problems.

- **firewalk.** A firewall reconnaissance tool, which tries to determine what access control lists you have on your firewall.