Sun Secure Global Desktop and OpenSSO Integration

Author: Joachim.Andres@sun.com

Document Date: October 2008

Table of Contents

1. Summary

Sun Secure Global Desktop Software provides access to centralized Windows, UNIX/Linux, Mainframe and Midrange applications from a wide range of popular clients. Essentially it allows users to get full desktop functionality through a web browser.

OpenSSO provides the foundation for integrating diverse web applications that might typically operate against a disparate set of identity repositories and are hosted on a variety of platforms such as web and application servers.

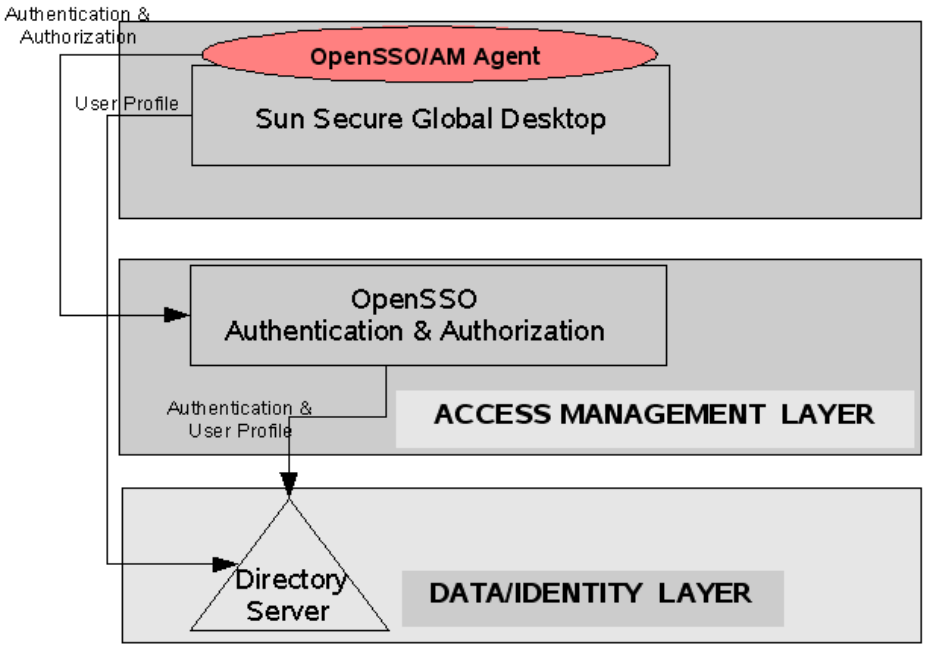
This document outlines how Sun Secure Global Desktop can be integrated in a web access management infrastructure based on OpenSSO enabling single sign on and policy enforcement.

Credits to Andy Hall and Paul Walker from Sun Microsystems for their contribution on architecture and configuration.

2. Overview

2.1. Architecture

The architecure consists of a directory server for authentication and user profiles, OpenSSO providing the access management infrastructure and Sun Secure Global Dektop deployed with an OpenSSO/Access Manager agent.



2.2. Software

All software can be downloaded free of charge from either www.sun.com/download or www.opensso.org.

Software	*Product File*
Sun Application Server 9.1 U2	`sjsas-9_1_02-solaris-sparc-ml.bin`
Sun Directory Server Enterprise Edition 6.3	`DSEE.6.3.Solaris-Sparc-full.tar.gz`
OpenSSO V1 Build5	`opensso.zip`
Sun Secure Global Desktop 4.41	`tta-4.41-907.sol-sparc.pkg`
Sun Access Manager Agent 2.2 for Apache 2.2	`apache_v22_SunOS_agent.zip`

2.3. Licenses

Sun Secure Global Desktop Software is available as a free download with a 30 day trial period. During the trial period all features of the software are enabled for a maximum of 5 concurrent users. To continue using the product after the trial period expires, a product license must be purchased.
All other products used in this architecture do not require license keys for evaluation beyond 30 days.

3. Access Management Infrastructure Installation

3.1. Application Server

1. Install Application Server 9.1 U2
   ```
   /space2/software/sjsas-9_1_02-solaris-sparc-ml.bin -console
   -savestate /space2/tools/as91install_state.txt
   ```
   Use the following installation options:

Parameter	*Value*
Installation Directory	/opt/SUNWappserver
Path to a Java 2 SDK 5.0	/usr/jdk/instances/jdk1.5.0
Admin User	admin
Admin User's Password	XXXXXXXX
Store admin user name and password in .asadminpass file ?	Yes.
Admin Port	4848
HTTP Port	8080
HTTPS Port	8181
Update center client	No.
Upgrade from previous versions	No.

2. (Optional, depending on system resources) Update JVM settings by editing the `/opt/SUNWappserver/domains/domain1/config/domain.xml` file and set `<jvm-options>-Xmx1G</jvm-options>`
3. (Optional) Disable the unencrypted port by editing the `/opt/SUNWappserver/domains/domain1/config/domain.xml` file and set enabled to false for http-listener-1
   ```
   <http-listener acceptor-threads="1" address="0.0.0.0" blocking-
   enabled="false" default-virtual-server="server" enabled="false"
   family="inet" id="http-listener-1" port="8080" security-
   enabled="false" server-name="" xpowered-by="true">
   <property name="proxiedProtocols" value="ws/tcp"/>
   </http-listener>
   ```
4. Start the server
   ```
   /opt/SUNWappserver/bin/asadmin start-domain domain1
   ```
5. Verify Installation
   Point a browser to https://myhost.france.sun.com:8181
6. Create a password file
   ```
   echo "AS_ADMIN_ADMINPASSWORD=XXXXXXXX" > /space2/tools/.password.as
   ```

3.2. Directory Server

1. Install Directory Server 6.3 Enterprise Edition
   ```
   /space2/software/dsee63/DSEE_ZIP_Distribution/dsee_deploy install --
   install-path /opt/dsee63
   ```
2. Create a password file
   ```
   echo "XXXXXXXX" > /space2/tools/.password.ds
   ```
3. Create Directory Server instance
   ```
   mkdir /opt/dsee63/instances/
   /opt/dsee63/ds6/bin/dsadm create --hostname myhost.france.sun.com --
   ```

```
port 389 --rootDN "cn=Directory Manager" --pwd-file
/space2/tools/.password.ds /opt/dsee63/instances/ds389
```

4. Start the directory server instance
   ```
   /opt/dsee63/ds6/bin/dsadm start /opt/dsee63/instances/ds389
   ```
5. Create a naming suffix
   ```
   /opt/dsee63/ds6/bin/dsconf create-suffix --user-dn "cn=Directory
   Manager" --pwd-file /space2/tools/.password.ds --no-inter --accept-
   cert dc=france,dc=sun,dc=com
   ```
6. Verify naming suffix creation
   ```
   /opt/dsee63/dsrk6/bin/ldapsearch -b "" -s base "objectclass=*"
   namingContexts
   version: 1
   dn:
   namingContexts: dc=france,dc=sun,dc=com
   ```

3.2.1. Directory Server Control Center (DSCC)

1. Initialize the DSCC registry. Use “XXXXXXXX” for the admin password
   ```
   /opt/dsee63/dscc6/bin/dsccsetup initialize
   ```
2. Register the directory instance with DSCC
   ```
   /opt/dsee63/dscc6/bin/dsccreg add-server
   /opt/dsee63/instances/ds389
   ```
3. Deploy the administration console within application server (as a web application)
   ```
   /opt/SUNWappserver/bin/asadmin deploy --user admin --
   passwordfile /space2/tools/.password.as --contextroot /dscc
   /opt/dsee63/var/dscc6/dscc.war
   ```
4. Verify.
   Point a browser to https://myhost.france.sun.com:8181/dscc and login as admin/password. The control center should appear.

3.2.2. Directory Data

The directory tree is held simpel and systems access using the “cn=Directory Manager” account. For a more structured and secure user and DIT design, see Appendix A.
At a minimum, some example user should be created:

1. Create example users
   ```
   /opt/dsee63/dsrk6/bin/ldapmodify -D "cn=Directory Manager" -w
   XXXXXXXX -c << EOF
      dn: uid=jwheeler,dc=france,dc=sun,dc=com
      changetype: add
      objectclass: person
      objectclass: organizationalperson
      objectclass:  inetorgperson
      uid: jwheeler
      sn: Wheeler
      cn: Jango Wheeler
      userpassword: XXXXXXXX

      dn: uid=pflower,dc=france,dc=sun,dc=com
      changetype: add
      objectclass: person
      objectclass: organizationalperson
      objectclass:  inetorgperson
      uid: pflower
      sn: Flower
      cn: Peter Flower
      userpassword: XXXXXXXX

      EOF
   ```

3.3. OpenSSO

1. Deploy the OpenSSO web application
   `/opt/SUNWappserver/bin/asadmin deploy --user admin --passwordfile /space2/tools/.password.as --contextroot /opensso /space2/software/opensso_build5/opensso/deployable-war/opensso.war`
2. Make the configuration directory
   `mkdir /opt/opensso`
3. Point a browser to `https://myhost.france.sun.com:8181/opensso` which should display the OpenSSO configuration page.
4. Choose “Custom Configuration”and create a new configuration
   Use the following configuration options:

Parameter	*Value*
Default User (the admin user)	amadmin
Password	XXXXXXXX
Server URL	https://myhost.france.sun.com:8181
Cookie Domain	.sun.com
Platform Locale	en_US
Configuration Directory	/opt/opensso
Configuration Store Type	External (Sun Java System DS)
Configuration Directory Host	myhost.france.sun.com
Configuration Directory Port	389
Encryption Key	EncryptionKey
Configuration Directory Root Suffix	dc=france,dc=sun,dc=com
Configuration Directory Login ID	cn=Directory Manager
Configuration Directory Password	XXXXXXXX
User Store Settings	Remote Directory
User Store Directory Host	myhost.france.sun.com
User Store Directory Port	389
User Store Root Suffix	dc=france,dc=sun,dc=com
User Store Directory Login ID	cn=Directory Manager
User Store Directory Password	XXXXXXXX
Store Type	LDAP with FAM schema.
Site Configuration	No.
Default agent user	amldapuser
Default agent user password	XXXXXXXX1(needs to be different from amadmin)

5. Verify deployment by pointing a browser to
   https://myhost.france.sun.com:8181/opensso and login as amadmin
   Note: A restart is normally not required before this step. If you get an error message in this step, restart the application server.

3.3.1. OpenSSO Tools

1. Generate the tools for the existing deployment
```
mkdir /opt/opensso/tools
cp -p
/space2/software/opensso_build5/opensso/tools/famAdminTools.zip
/opt/opensso/tools
cd /opt/opensso/tools
unzip famAdminTools.zip
export JAVA_HOME=/usr/jdk/instances/jdk1.5.0
/opt/opensso/tools/setup --path /opt/opensso
```
2. Create symbolic links for convenience as the path to the tools is not obvious
```
ln -s /opt/opensso/tools/opensso/bin/famadm
/opt/opensso/tools/famadm
ln -s /opt/opensso/tools/opensso/bin/ampassword
/opt/opensso/tools/ampassword
ln -s /opt/opensso/tools/opensso/bin/amtune
/opt/opensso/tools/amtune
```
3. Verify the tools
```
/opt/opensso/tools/famadm --version

Sun Federated Access Manager 8.0 (2008-July-21 07:32)
```
4. Create a password file to be used with famadm
```
echo "XXXXXXXX" > /space2/tools/.password.opensso
chmod 400 /space2/tools/.password.opensso
```

3.3.2. (Optional) OpenSSO Client SDK

1. Make a debug directory
```
mkdir -p /opt/opensso/sdk/debug
```
2. Extract the bits
```
cp -p /space2/software/opensso_build5/opensso/samples/fam-client.zip
/opt/opensso
cd /opt/opensso
unzip fam-client.zip
```
This archive contains a couple of war files which are deployed inside the client container.
3. Generate the SDK
```
chmod +x /opt/opensso/sdk/scripts/setup.sh
cd /opt/opensso/sdk
scripts/setup.sh
```

Parameter	*Value*
Debug directory	`/opt/opensso/sdk/debug`
Password of the server application	`XXXXXXXX`
Protocol of the server	`https`
Host name of the server	`myhost.france.sun.com`
Port of the server	`8181`
Server's deployment URI	`opensso`
Naming service	`https://myhost.france.sun.com:8181/opensso/namingservice`

The client SDK now consists of the following 2 files:
```
/opt/opensso/sdk/resources/AMConfig.properties
/opt/opensso/sdk/lib/openssoclientsdk.jar
```

3.3.3. OpenSSO LDAP Authentication

3.3.3.1. Local LDAP Authentication Module

1. Access the OpenSSO console at https://myhost.france.sun.com:8181/opensso/console and login as amadmin
2. Under Access Control -> Realm france -> Authentication, edit the LDAP authentication module and set the following parameters

Parameter	*Value*
DN for root user bind	`cn=Directory Manager`
Password for root user bind	`XXXXXXXX`

3.3.3.2. Profile Settings

This is optional if the deployment is only linked with the local LDAP directory.

1. Configure not to require user profiles. In the console, under Access Control -> Realm france -> Authentication, click “Advanced Properties”.
2. Set “User Profile” to “Ignored”

3.3.3.3. Authentication Chain

1. In the console, under Access Control -> Realm france -> Authentication, set the default authentication chain to `LocalDSChain.`
2. ddd

Parameter	*Value*
Name 3.	`LocalDSChain`
Item 1	`Instance: LDAP`

4. In the console, under Access Control -> Realm france -> Authentication, set the default authentication chain to `LocalDSChain.`
5. Note: The administration console now needs to be accessed through https://myhost.france.sun.com:8181/opensso**/console**

6. Verify authentication by accessing https://myhost.france.sun.com:8181/opensso and login with a valid LDAP username and password. After logout, access the same URL and login as amadmin.

4. Secure Global Desktop (SGD) Installation

4.1. Add SGD system group and users

1. Add group ttaserv by adding the following line to `/etc/group`
   ```
   ttaserv::100:
   ```
2. Add the users ttaserv and ttasys by adding the following lines to `/etc/passwd`
   ```
   ttasys:x:100:100::/export/home/ttasys:/bin/sh
   ttaserv:x:101:100::/export/home/ttaserv:/bin/sh
   ```
3. Create the home directories
   ```
   mkdir -p /export/home/ttasys
   mkdir -p /export/home/ttaserv
   ```
4. Set the ownership of the home directories
   ```
   chown ttasys:ttaserv /export/home/ttasys
   chown  ttaserv:ttaserv  /export/home/ttaserv
   ```
5. Verify home directories
   ```
   ls -l /export/home/
   total 4
   drwxr-xr-x   2 ttaserv  ttaserv       512 Sep 11 14:44 ttaserv
   drwxr-xr-x   2 ttasys   ttaserv       512 Sep 11 14:44 ttasys
   ```

4.2. Install Secure Global Desktop

1. Add the SGD packages
   ```
   pkgadd -d tta-4.41-907.sol-sparc.pkg
   ```

Parameter	*Value*
Installation type	`install 4.41.907`
Installation directory	`/opt/tarantella`

2. Start SGD
   ```
   /opt/tarantella/bin/tarantella start
   ```
   At first startup, SGD allows to configure certain parameters.

Parameter	*Value*
Installation type	`install 4.41.907`
Peer DNS name	`myhost`
HTTP Port	`80`
Archive logs every week?	`yes (Sunday 03:00 hours)`

3. Add SGD licence key
   ```
   /opt/tarantella/bin/tarantella license add <Your-License-Key>
   ```
4. Verify that SGD is running
   Point a browser to http://myhost.france.sun.com:80/sgd
   Login as root/XXXXXXXX. The webtop should appear.

4.2.1. Create SGD Administrator Account

By default, the administration console can only be accessed by the superuser, which the system's root user.

1. Create a new local system user by
   a.) Editing the /etc/passwd file and adding the line
   ```
   sgdadmin:x:102:100::/export/home/sgdadmin:/bin/sh
   ```
   b.) Create the home directory and set permissions
   ```
   mkdir -p /home/export/sgdadmin
   chown sgdadmin:ttaserv /home/export/sgdadmin
   ```
   c.) Editing the /etc/shadow file and adding the line
   ```
   sgdadmin:NP:6445::::::
   ```
   d.) Setting the password for sgdadmin
   ```
   passwd sgdadmin (set it to XXXXXXXX)
   ```
2. Create a new user
   ```
   /opt/tarantella/bin/tarantella object new_person --name
   "o=Tarantella System Objects/cn=sgdadmin"  --surname
   Administrator --user sgdadmin --inherit true
   ```
3. Add new user to administrators role
   ```
   /opt/tarantella/bin/tarantella role add_member --role
   ".../_ens/o=Tarantella System Objects/cn=Global
   Administrators" --member ".../_ens/o=Tarantella System
   Objects/cn=sgdadmin"
   ```
4. Access the SGD console at http://myhost.france.sun.com:80/sgdadmin and login as sgdadmin.

4.3. (Optional) Configure LDAP authentication

1. Set the LDAP login URL
   ```
   /opt/tarantella/bin/tarantella config edit --login-ldap-url
   "ldap://myhost.france.sun.com:389/dc=france,dc=sun,dc=com"
   ```
2. Set the username and password for the LDAP connection
   ```
   /opt/tarantella/bin/tarantella passcache new --ldap --resuser
   "cn=Directory Manager" --respass "XXXXXXXX"
   ```
3. Set LDAP as the first authentication module
   ```
   /opt/tarantella/bin/tarantella config edit --login-ldap 1
   ```
4. Verify. Point a browser to http://myhost.france.sun.com:80/sgd and login with a valid ldap username and password.

4.4. Enable SSL for SGD

1. Get an X.509 certificate
   a.) Create request
   ```
   /opt/tarantella/bin/tarantella security certrequest --country US --
   state CA --orgname "Test Inc."
   ```
   b.) Send request to a CA or generate a self signed certificate (ignore warnings)
   ```
   /opt/tarantella/bin/tarantella security selfsign
   ```
   Note: The cert has been place in /opt/tarantella/var/tsp
2. Restart the web server
   ```
   /opt/tarantella/bin/tarantella restart webserver
   ```

5. SGD and OpenSSO Integration (SSO and Authorization)

5.1. Configure SGD for Third Party Authentication

1. Enable third party authentication
   `/opt/tarantella/bin/tarantella config edit --login-thirdparty 1`
2. Configure SGD where to retrieve user profile for users with third party authentication
   `/opt/tarantella/bin/tarantella config edit --login-ldap-thirdparty-profile 1`
3. Configure SGD where to find more information about those users
   `/opt/tarantella/bin/tarantella config edit --login-ldap-url "ldap://myhost.france.sun.com:389/dc=france,dc=sun,dc=com"`
   (For SGD versions prior to 4.41) `/opt/tarantella/bin/tarantella config edit --login-web-ldap-profile 1`
4. If not done before as part of an optional step, set the LDAP access user and password:
   `/opt/tarantella/bin/tarantella passcache new --ldap --resuser "cn=Directory Manager" --respass "XXXXXXXX"`

5.2. Configure SGD Tomcat for SSO

1. Edit the `/opt/tarantella/webserver/tomcat/5.0.28_axis1.2/conf/server.xml` file and set `tomcatAuthentication=false`

```
<!-- Define a Coyote/JK2 AJP 1.3 Connector on port 8009 -->
<Connector port="8009" minProcessors="5" maxProcessors="75"
      enableLookups="true" redirectPort="8443"
      acceptCount="10" debug="0" connectionTimeout="0"
      useURIValidationHack="false"
      tomcatAuthentication="false"
      protocolHandlerClassName="org.apache.jk.server.JkCoyoteHandler
      "/>
```

5.3. Create agent profile in OpenSSO

1. Create the agent account which is used by the Apache agent to communicate with OpenSSO
   `/opt/opensso/tools/famadm create-agent --adminid amadmin --password-file /space2/tools/.password.opensso --realm / --agenttype WebAgent --agentname sgdagent --attributevalues userpassword=XXXXXXXX`

5.4. Create access policy in OpenSSO

1. Access the OpenSSO console at https://myhost.france.sun.com:8181/opensso/console and login as amadmin
2. Under Access Control -> Realm france -> Policies, create a new policy:

Parameter	*Value*
Policy Name	SGD Access
Rule 1	Type: URL Policy Agent Name: SGD Access Resource: https://myhost.france.sun.com:443/sgd* Actions: GET, allow; POST, allow
Rule 2	Type: URL Policy Agent Name: SGD Access Resource: https://myhost.france.sun.com:443/

Parameter	*Value*
	`Actions: GET, allow; POST, allow`
Subject	`Type: Authenticated Users` `Name: All`

Note that with this configuration, access to the un encrypted port (80) for /sgd is not allowed by the access management infrastructure regardless if it is up or not.

5.5. OpenSSO/Access Manager Agent Installation

1. Verify the version of the embedded Apache web server
```
/opt/tarantella/webserver/apache/2.2.8_openssl-
0.9.8g_jk1.2.25/bin/httpd -v
Server version: Apache/2.2.8 (Unix)
Server built:   Apr 16 2008 10:57:20
```
Note that prior versions of SGD might have different versions of Apache and thus require a different Access Manager agent.
2. Create the installation directory and copy files
```
mkdir /opt/agents
cp -p /space2/software/apache_v22_SunOS_agent.zip /opt
cd /opt
unzip  apache_v22_SunOS_agent.zip
```
3. Make the agentadmin command executable
```
chmod +x /opt/web_agents/apache22_agent/bin/agentadmin
```
4. Create a password file
```
chmod +x /opt/web_agents/apache22_agent/bin/crypt_util
/opt/web_agents/apache22_agent/bin/crypt_util XXXXXXXX >
/space2/tools/.password.agent
cat /space2/tools/.password.agent
LynKyOIgdWt404ivWY6HPQ==
```
5. Install the agent
```
/opt/web_agents/apache22_agent/bin/agentadmin --install --
saveResponse /space2/tools/apache22agent22Install_state.txt
```
Enter the following parameters:

Parameter	*Value*
Apache Server Config Directory Path	`/opt/tarantella/webserver/apache/2.2.8_openssl-0.9.8g_jk1.2.25/conf`
Access Manager Services Host	`myhost.france.sun.com`
Access Manager Services port	`8181`
Access Manager Services Protocol	`https`
Access Manager Services Deployment URI	`/opensso`
Agent Host name	`myhost.france.sun.com`
Port number for Web Server instance	`443`
Preferred Protocol for Web Server	`https`
Agent Profile name	`UrlAccessAgent`
Path to password file	`/space2/tools/.password.agent`

6. Restart the SGD web server
```
/opt/tarantella/bin/tarantella restart webserver
```

5.6. Agent Configuration

1. Edit the /opt/web_agents/apache22_agent/Agent_001/config/AMAgent.properties file and set:
```
com.sun.am.log.level = all:5
com.sun.am.policy.am.userid.param=UserToken
com.sun.am.policy.agents.config.profile.attribute.fetch.mode=HTTP_HEADER
com.sun.am.policy.agents.config.notenforced_list =
https://myhost.france.sun.com:443/
https://myhost.france.sun.com:443/index*
https://myhost.france.sun.com:443/axis* http://localhost:443/axis*
https://myhost.france.sun.com:443/sgdadmin*
com.sun.am.policy.am.username = sgdagent
com.sun.am.policy.am.password = LynKyOIgdWt404ivWY6HPQ==
```
(The encrypted password as generated by `crypt_util` for value `XXXXXXXX`)
2. Restart the SGD web server
```
/opt/tarantella/bin/tarantella restart webserver
```

5.7. Logout

1. Configure SGD to re-direct after logout to the OpenSSO logout page by editing the `/opt/tarantella/webserver/tomcat/5.0.28_axis1.2/webapps/sgd/webtops/standard/webtop/logged-out.jsp` file.
2. In the following section, add the `response.sendRedirect` statement
```
    {
%>
          <td><img src="<%= getWebtopGif("loggedout_login.gif") %>"
name="login"
 width="560" height="21" border="0"></td>
<%
    }
response.sendRedirect(response.encodeRedirectURL("https://myhost.france.sun.com:8181/opensso/UI/Logout?goto=https://myhost.france.sun.com/sgd"));

%>
        </tr>
        <tr>
          <td colspan="3"><img src="<%=
getWebtopGif("loggedout_bottom.gif") %>"
 width="560" height="47"></td>
          </tr>
      </table>
    </td>
 </tr>
</table>
```

6. Uninstallation

1. Application Server
   ```
   /opt/SUNWappserver/bin/asadmin stop-domain domain1
   /opt/SUNWappserver/uninstall
   ```
2. Directory Server
   ```
   /opt/dsee63/instances/ds389/stop-slapd
   /space2/software/dsee63/DSEE_ZIP_Distribution/dsee_deploy uninstall --
   install-path /opt/dsee63/
   ```
3. OpenSSO/Access Manager Agent
   ```
   /opt/web_agents/apache22_agent/bin/agentadmin --uninstallAll
   ```
4. Secure Global Desktop
   ```
   /opt/tarantella/bin/tarantella uninstall
   ```

7. Appendix A: Directory Information Tree

So far the directory has been used in a simple form with a flat tree and all system access done by the “cn=Directory Manager” account. A more meaningful DIT containing accounts for systems and users can be built as follows:

1. Create the container entries for people, system accounts and the OpenSSO configuration

```
/opt/dsee63/dsrk6/bin/ldapmodify -D "cn=Directory Manager" -w XXXXXXXX
-c << EOF
   dn: ou=people,dc=france,dc=sun,dc=com
   changetype: add
   objectclass: organizationalUnit
   ou: people

   dn: ou=openssconfig,dc=france,dc=sun,dc=com
   changetype: add
   objectclass: organizationalUnit
   ou: openssoconfig

   dn:ou=systemaccounts,dc=france,dc=sun,dc=com
   changetype: add
   objectclass: organizationalUnit
   ou:  systemaccounts

   EOF
```

2. Create system accounts.

```
/opt/dsee63/dsrk6/bin/ldapmodify -D "cn=Directory Manager" -w XXXXXXXX
-c << EOF
   dn: uid=sgdaccess,ou=systemaccounts,dc=france,dc=sun,dc=com
   changetype: add
   objectclass: person
   objectclass: organizationalperson
   objectclass:  inetorgperson
   uid: sgdaccess
   sn: agdaccess
   cn: sgdaccess
   userpassword: XXXXXXXX

   dn: uid=openssoaccess,ou=systemaccounts,dc=france,dc=sun,dc=com
   changetype: add
   objectclass: person
   objectclass: organizationalperson
   objectclass:  inetorgperson
   uid: openssoaccess
   sn:  openssoaccess
   cn:  openssoaccess
   userpassword: XXXXXXXX

   dn:
   uid=openssoconfigaccess,ou=systemaccounts,dc=france,dc=sun,dc=com
   changetype: add
   objectclass: person
   objectclass: organizationalperson
   objectclass:  inetorgperson
   uid: openssoconfigaccess
   sn:   openssoconfigaccess
   cn:   openssoconfigaccess
   userpassword: XXXXXXXX

   EOF
```

3. Set directory server access control for the system accounts

Account	*Access Rights*
sgdaccess	Read: `ou=people,dc=france,dc=sun,dc=com`
openssoaccess	Read: `ou=people,dc=france,dc=sun,dc=com`
openssoconfigaccess	Read/Write: `ou=opensssconfig,dc=france,dc=sun,dc=com`

```
/opt/dsee63/dsrk6/bin/ldapmodify -D "cn=Directory Manager" -w XXXXXXXX
-c << EOF
   dn: ou=people,dc=france,dc=sun,dc=com
   changetype: modify
   add: aci
   aci: (target = ldap:///ou=people,dc=france,dc=sun,dc=com)
   (targetscope = subtree) (targetattr="*") (version 3.0; acl "People
   Read  Access"; allow (read,compare, search) (userdn =
   "ldap:///uid=sgdaccess,ou=systemaccounts,dc=france,dc=sun,dc=com" or
   userdn =
   "ldap:///uid=openssoaccess,ou=systemaccounts,dc=france,dc=sun,dc=com
   ") ;)

   dn: ou=opensssconfig,dc=france,dc=sun,dc=com
   changetype: modify
   add: aci
   aci: (target = ldap:///ou=opensssconfig,dc=france,dc=sun,dc=com)
   (targetscope =  subtree) (targetattr="*") (version 3.0; acl "OpenSSO
   Config Access"; allow (all) (userdn =
   "ldap:///uid=openssoconfigaccess,ou=systemaccounts,dc=france,dc=sun,
   dc=com") ;)

   EOF
```

4. Create example users

```
/opt/dsee63/dsrk6/bin/ldapmodify -D "cn=Directory Manager" -w XXXXXXXX
-c << EOF
   dn: uid=jwheeler,ou=people,dc=france,dc=sun,dc=com
   changetype: add
   objectclass: person
   objectclass: organizationalperson
   objectclass:  inetorgperson
   uid: jwheeler
   sn: Wheeler
   cn: Jango Wheeler
   userpassword: XXXXXXXX

   dn: uid=pflower,ou=people,dc=france,dc=sun,dc=com
   changetype: add
   objectclass: person
   objectclass: organizationalperson
   objectclass:  inetorgperson
   uid: pflower
   sn: Flower
   cn: Peter Flower
   userpassword: XXXXXXXX

   EOF
```