HEALTHEVET WEB SERVICES CLIENT (HWSC)

SYSTEM MANAGEMENT GUIDE

Version 1.0

February 2011

Department of Veterans Affairs
Office of Information and Technology
Product Development

Revision History

Table i. Revision History

Date	Description	Author(s)
02/2011	HWSC Version 1.0 Initial release	Product Development Services Security Program HWSC development team. Albany, NY OIFO: • Developer—Mike Kilmade • Developer—Liz Defibaugh Bay Pines, FL OIFO: • Development Manager—Charles Swartz Oakland, CA OIFO: • Developer—Kyle Clarke • Developer—Jose Garcia • SQA—Gurbir Singh • Tester—Padma Subbaraman • Technical Writer—Susan Strack

Contents

Tables

1 Introduction

1.1 Document Overview

This guide provides information for:

- M administrators managing the Health*e*Vet Web Services Client (HWSC) application, and managing applications using HWSC to call external web services
- M developers that need to manage HWSC during the development of web service client code
- Anyone needing detailed information on security issues related to HWSC and Caché Web Services

It assumes readers are familiar with:

- VistA/M
- Caché

1.1.1 Additional Resources

The complete HWSC 1.0 end-user documentation package consists of:

- *HWSC 1.0 Installation Guide*
- *HWSC 1.0 System Management Guide*
- *HWSC 1.0 Developers Guide*

They are available from the Product Support anonymous directories and the VHA Software Documentation Library (VDL) Web site (http://www4.va.gov/vdl/application.asp?appid=180).

Health*e*Vet-VistA end-user documentation and software can be downloaded from the Product Support anonymous directories:

Preferred Method	download.vista.med.va.gov This method transmits the files from the first available FTP server.
Albany OIFO	ftp://ftp.fo-albany.med.va.gov/
Hines OIFO	ftp://ftp.fo-hines.med.va.gov/
Salt Lake City OIFO	ftp://ftp.fo-slc.med.va.gov/

HealtheVet-VistA end-user documentation is made available online in Microsoft Word format and Adobe Acrobat Portable Document Format (PDF). The PDF documents *must* be read using the Adobe Acrobat Reader (i.e., ACROREAD.EXE), which is freely distributed by Adobe Systems Incorporated at the following Web address:

http://www.adobe.com/

REF: For more information on the use of the Adobe Acrobat Reader, please refer to the *Adobe Acrobat Quick Guide* at the following Web address:

http://vista.med.va.gov/iss/acrobat/index.asp

DISCLAIMER: The appearance of any external hyperlink references in this manual does not constitute endorsement by the Department of Veterans Affairs (VA) of this Web site or the information, products, or services contained therein. The VA does not exercise any editorial control over the information you may find at these locations. Such links are provided and are consistent with the stated purpose of this VA Intranet Service.

1.2 HWSC Overview

As VistA is migrated to the new HealtheVet-VistA system, some legacy VistA applications will need synchronous access to HealtheVet application services and data. HWSC uses Caché's Web Services Client to invoke web service methods on external servers and retrieve results. It provides helper methods and classes to improve the use of the Web Service Client in a HealtheVet-VistA environment.

1.2.1 HWSC Features

HWSC acts as an adjunct to the web services client functionality provided in Caché, by:

- Leveraging Caché's platform-provided Web services client capabilities.
- Adding a file and UI to manage the set of external web server endpoints (IP, port, etc.)
- Adding a file and UI to register and manage the set of external web services.
- Providing runtime API to invoke a specific web service on a specific web server.
- Providing a runtime API to facilitate error processing in a VistA environment.
- Providing a deployment API to install/register a web service proxy from a WSDL file.
- Providing a management UI including the ability to 'ping' (test) a given web service/server combination from VistA/M.
- Supporting both SOAP- and REST-style web services
- Fostering consistent implementation of VistA/M web service consumers.

2 HWSC Management Functions

HWSC provides several management screens allowing you to create and manage the web server and web service information needed by VistA applications to consume external web services. The management screens are:

- Web Server Manager
- Web Service Manager
- Lookup Key Manager

2.1 Using the Web Server Manager

You can use the XOBW WEB SERVER MANAGER option to call up the HWSC Web Server Manager. You should use this tool to enter web server information.

NOTE: Programmer access (DUZ(0)=”@”) is required to use this option.

```
Web Server Manager            Apr 18, 2007@16:07:54          Page:    1 of 1
                       HWSC Web Server Manager
                      Version: 1.0     Build: xx

 ID    Web Server Name           IP Address or Domain Name:Port
 1    *Oakland Test Server1      vhaisxsysa.vha.med.va.gov:7111
 2    *Oakland Test Server2      vhaisxsysb.vha.med.va.gov:7112
 3    *Oakland Test Server3      vhaisxsysc.vha.med.va.gov:7113
 4    *Oakland Test Server4      vhaisxsysd.vha.med.va.gov:7114


          Legend:  *Enabled
AS   Add Server                          TS   Test Server
ES   Edit Server                         WS   Web Service Manager
DS   Delete Server                       CK   Check Web Service Availability
EP   Expand Entry                        LK   Lookup Key Manager
Select Action:Quit//
```

The table below summarizes the actions available in the Web Server Manager:

Table 1. Web Server Manager Actions

Action	Description
AS (Add Server)	Add a new WEB SERVER (18.12) entry.
DS (Delete Server)	Delete a WEB SERVER (18.12) entry.
ES (Edit Server)	Edit a WEB SERVER (18.12) entry.

Action	Description
EP (Expand Entry)	View all information about a particular WEB SERVER (18.12) entry.
TS (Test Server)	If the XOBT sample application is installed, it runs some of its tags to call sample external web services. Disabled if the sample application (XOBT) is not installed. For more information on the XOBT sample, see the HWSC Developer Guide.
WS (Web Service Manager)	Invoke the Web Service Manager screen.
CK (Check Web Service Availability)	Check availability for each web service authorized/assigned to the web server.
LK (Lookup Key Manager)	Invoke the Lookup Key Manager screen.

When you add or edit a web server, you are prompted for the following information:

Table 2. Web Server Fields

Field	Description
Name	Name to identify the web server entry. Must be 3-30 characters in length.
Server	The full domain name (for DNS) or IP address of the web service server
Port	The TCP/IP port of web service server
Default Http Timeout	A default http timeout to use for outgoing requests made to this server. The default value is 30.
Status	Select either ENABLED or DISABLED
	Security Credentials
Login Required?	If a login is required, enter YES (allows editing of username and password)
Username	Name of the authorized user in the security realm on the web server.
Edit Password?	Enter 'Y' if you wish to change the password; otherwise 'N'.
	SSL Setup
SSL Enabled	Determines whether SSL/TLS is enabled for the web server.
SSL Configuration	Name of Caché SSL configuration to use for this web server.
SSL Port	SSL port number to use for this web server
	Authorize Web Services
Select Web Service	Select one of the web services listed, or enter a new one. A web service must be 'authorized' by entering here, to be used with this web server.
Status	Select ENABLED or DISABLED.

NOTE: The editing of SSL Setup fields is disabled on OpenVMS systems.

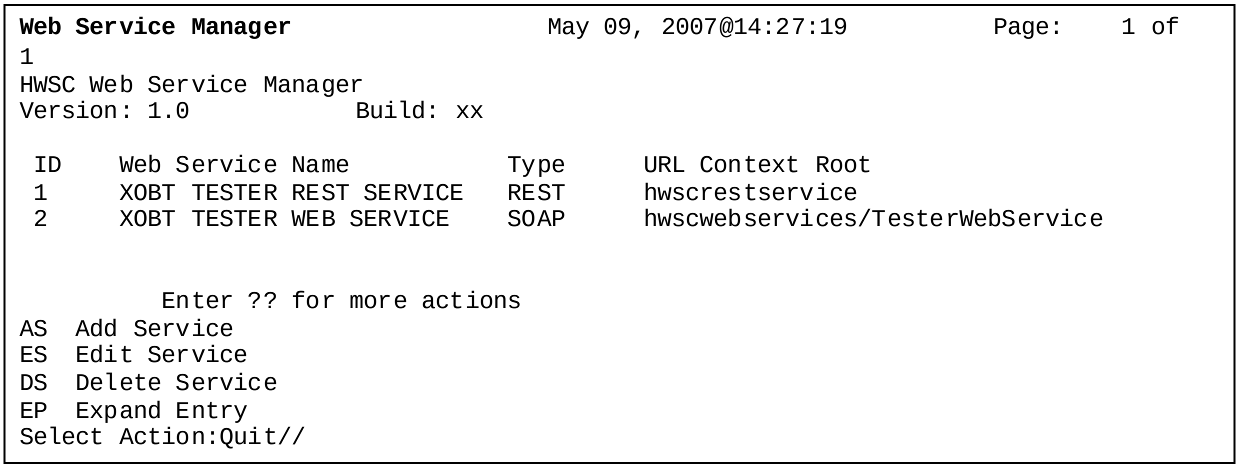
2.2 Using the Web Service Manager

Use the Web Service Manager to enter or modify information for web services that M applications will access.

NOTE: VistA applications that install web service clients will probably automatically create web service entries for the external web services they will be accessing.

NOTE: Programmer access (DUZ(0)=”@”) is required to use this option.

To display the Web Service Manager, select the WS action in the Web Server Manager (see “Using the Web Server Manager”). In addition to adding a new service, you can edit or delete a previous entry, or display complete information previously entered for a particular service.



```
Web Service Manager                     May 09, 2007@14:27:19          Page:    1 of
1
HWSC Web Service Manager
Version: 1.0            Build: xx

 ID    Web Service Name            Type      URL Context Root
 1     XOBT TESTER REST SERVICE    REST      hwscrestservice
 2     XOBT TESTER WEB SERVICE     SOAP      hwscwebservices/TesterWebService


          Enter ?? for more actions
AS  Add Service
ES  Edit Service
DS  Delete Service
EP  Expand Entry
Select Action:Quit//
```

The table below summarizes the actions available in the Web Service Manager:

Table 3. Web Service Manager Actions

Action	Description
AS (Add Service)	Add a new WEB SERVICE (18.02) entry.
DS (Delete Service)	Delete a WEB SERVICE (18.02) entry.
ES (Edit Service)	Edit a WEB SERVICE (18.02) entry.
EP (Expand Entry)	View all information about a particular WEB SERVICE (18.02) entry.

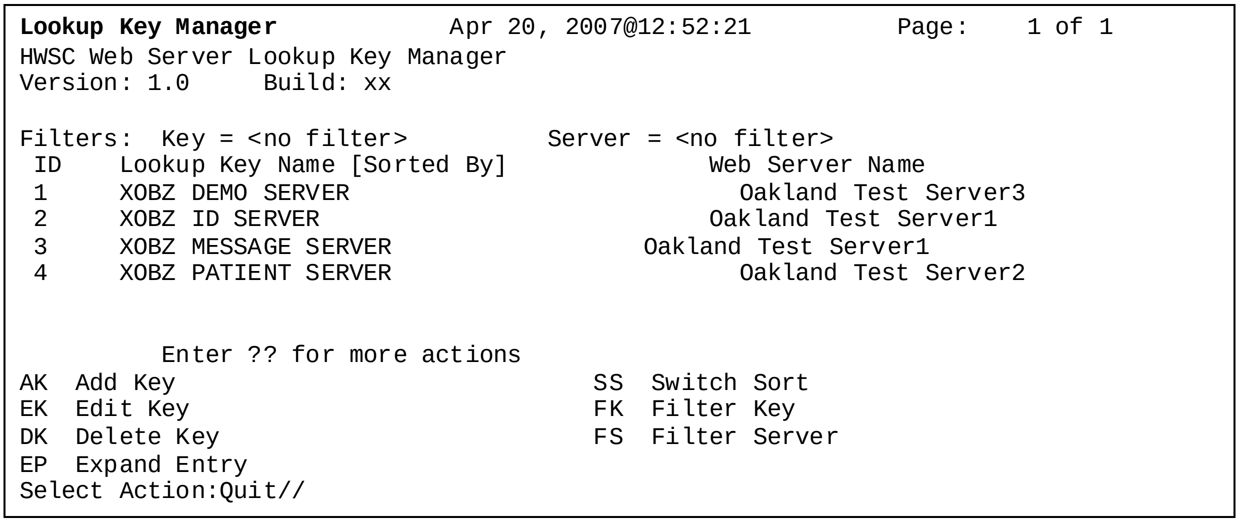
There are two types of web services supported by HWSC: REST and SOAP. When you register a new web service, you will be prompted for slightly different information depending on the type:

Table 4. Web Service Fields

Service Type	Field	Description
REST	Name	Name to identify the web service entry. Must be non-numeric, 3-30 characters long, and starting without punctuation.
	Date Registered	Date the entry was registered (created).
	Service Type	Choose **REST**.
	Context Root	The context root of the web service
	Availability Resource	A 'resource' to append to the context root to create a URL that can be used to check if the web service is available.
SOAP	Name	Name to identify the web service entry. Must be non-numeric, 3-30 characters long, and starting without punctuation.
	Date Registered	Date the entry was registered (created).
	Service Type	Choose **SOAP**.
	Proxy Class Name	Name of the Caché Object class that is the web service client proxy, as created by the Caché WSDL compiler.
	Context Root	The context root of the web service
	Availability Resource	A 'resource' to append to the context root to create a URL that can be used to check if the web service is available.

2.3 Using the Lookup Key Manager

The Lookup Key Manager is a tool for sites to use to associate a mnemonic name with a particular server. Applications use the mnemonic at runtime to retrieve the web server.



```
Lookup Key Manager           Apr 20, 2007@12:52:21          Page:    1 of 1
HWSC Web Server Lookup Key Manager
Version: 1.0     Build: xx

Filters:  Key = <no filter>         Server = <no filter>
 ID    Lookup Key Name [Sorted By]             Web Server Name
 1     XOBZ DEMO SERVER                          Oakland Test Server3
 2     XOBZ ID SERVER                          Oakland Test Server1
 3     XOBZ MESSAGE SERVER                 Oakland Test Server1
 4     XOBZ PATIENT SERVER                       Oakland Test Server2


          Enter ?? for more actions
AK  Add Key                            SS  Switch Sort
EK  Edit Key                           FK  Filter Key
DK  Delete Key                         FS  Filter Server
EP  Expand Entry
Select Action:Quit//
```

The table below summarizes the actions available in the Lookup Key Manager:

Table 5. Lookup Key Manager Actions

Action	Description
AK (Add Key)	Add a new WEB SERVER LOOKUP KEY (18.13) entry.
DK (Delete Key)	Delete a WEB SERVER LOOKUP KEY (18.13) entry.
EK (Edit Key)	Edit a WEB SERVER LOOKUP KEY (18.13) entry.
EP (Expand Entry)	View all information about a particular WEB SERVER LOOKUP KEY (18.13) entry.
FK (Filter Key)	Limit the list of lookup keys displayed by the key manager. The user specifies text to be used as a filter against the beginning characters of the key values. Only matching keys will be listed. This protocol is also used to clear a key filter if one is currently being applied.
FS (Filter Server)	Limit the list of web server entries displayed by the key manager. The user specifies text to be used as a filter against the beginning characters of the key values. Only matching keys will be listed. This protocol is also used to clear a server filter if one is currently being applied.
SS (Switch Sort)	Switch sorting between 'key' and 'server' in the list of web server lookup keys.

When you enter a new lookup key, you will be prompted for the following information:

Table 6. Lookup Key Manager Fields

Field	Description
Key Name	The name for the lookup key must be 3-30 characters in length.
Brief Description	The description must be 2-50 characters in length.
Associated Web Server Name	The WEB SERVER NAME to associate this key with. Lookups made on the key will return this web server.

3 Security

3.1 Using SSL/TLS and Certificate-Based Authentication with HWSC

HWSC http connections can be secured with SSL/TLS. Doing so makes those connections much more secure, by encrypting the authentication handshake as well as the message contents. SSL/TLS is not currently supported on OpenVMS systems due to a memory leak issue that has been diagnosed as an issue with a VMS-level library.

REF: For more information about using SSL/TLS (on Windows or Linux systems), contact the Help Desk and file a Remedy ticket with the HWSC Product Support team.

In addition, HWSC supports two authentication mechanisms:

- HTTP Basic authentication
- Certificate-based authentication

Certificate-Based authentication is not currently supported on OpenVMS systems due to a memory leak issue that has been diagnosed as an issue with a VMS-level library.

REF: For more information about using Certificate-Based Authentication (on Windows or Linux systems), contact the Help Desk and file a Remedy ticket with the HWSC Product Support team.

3.2 Securing a Web Service Using HTTP Basic Authentication

To use HTTP Basic Authentication (with or without SSL/TLS), some setup is required. The J2EE and M server administrators need to perform the following steps for HTTP Basic Authentication:

J2EE Administrator

1. Based on the **web.xml** and **weblogic.xml** service descriptors for the web service application, create a group that has the role to invoke the web service.
2. Identify or create a user that can be assigned to this group.
3. Provide the user’s credentials (username and password) to the M server administrator.

M server Administrator

1. Identify or create a WEB SERVER (#18.12) file entry for the J2EE server in question, enter YES for "Login Required?", and enter the username and password provided by the J2EE administrator.

When using HTTP Basic Authentication without SSL/TLS, values for username and password are passed 'in the clear' in the HTTP message header. Because these values are transmitted in clear text, they are not secure on "open" networks (e.g., not behind a computer room firewall).

4 Troubleshooting

4.1 HWSC Availability Checking

The Web Server Manager ‘Check Availability’ action, is probably the best tool to troubleshoot web services problems. It is one way to test the validity of the web server and web service configuration; it will list all authorized/assigned web services and test each one individually. (See the section “Supporting HWSC Availability Checking” in the *HWSC Developer Guide* for more information.)

Assuming that the web service entry is set up with a valid 'Availability Resource', this option connects to the URL composed of the server, port, context root and availability resource, and reports any errors encountered. Some examples are provided below for a few error types.

Unsuccessful availability check (listener down):

```
Web Service Availability     May 18, 2007@11:46:56       Page:    1 of    1
Web Server:
 4    *VHAISXSYSA                vhaisxsysa:7111

  1  Unable to retrieve '?wsdl' for XOBT TESTER WEB SERVICE
    o  ERROR #6059: Unable to open TCP/IP socket to server vhaisxsysa:7111
```

Unsuccessful availability check (authorization failure – HTTP error code 401):

```
Web Service Availability     May 18, 2007@11:54:12       Page:    1 of    1
Web Server:
 13   * VHAISXSYSB                vhaisxsysb:7111

  1  Unable to retrieve '?wsdl' for XOBT TESTER WEB SERVICE
    o  HTTP Response Status Code: 401
  2  Unable to retrieve '/available' for XOBT TESTER REST SERVICE
    o  HTTP Response Status Code: 401
```

Successful availability check:

```
Web Service Availability     May 18, 2007@11:49:08       Page:    1 of    1
Web Server:
  5   *VHAISXSYSC                vhaisxsysc:7111

  1  XOBT TESTER WEB SERVICE is available
  2  XOBT TESTER REST SERVICE is available
```

4.2 Some Runtime Errors Due to Configuration Issues

4.2.1 Caché Error #5005: Cannot Open File

On VMS systems, the VMS accounts used by end-users must have RWED access to the directory used by Caché for creating temporary files. Otherwise, calls to external web services made in a given end-user's process may fail. Runtime errors may be of the form:

```
Cannot open file 'WREK2XEPAXY.stream'
```

or:

```
ERROR #5005: Cannot open file 'TRMUAJZVAQT.stream'
```

(where 'WREK2XEPAXY.stream' and 'TRMUAJZVAQT.stream' are examples of randomly assigned temporary file names used by Caché).

For detailed information on how to address this issue, see the "Pre-Installation Preparation" section of the HWSC Installation Guide.

4.2.2 zDelete Errors

On VMS systems VMS accounts used for end-user processes need to have adequate VMS process parameters (quotas). If VMS process parameters are not set to at least the minimum values recommended by InterSystems, calls to external web services made in a given end-user's process may fail. Errors messages may include the phrase:

```
"Error: <FUNCTION>zDelete^%ooLibrary.File.1"
```

This error is usually caused (at a lower level) by a VMS process quota exceeded error.

For detailed information on how to address this issue, see the "Pre-Installation Preparation" section of the HWSC Installation Guide.

Appendix A: HWSC Error Codes

Error code entries are contained in the DIALOG file (#.84). The following dialog entries are used in the HWSC package:

Table 7. HWSC Error Codes

Dialog Number	Short Description
186001	(reserved for future use)
186002	Web Server Disabled
186003	Web Service not registered to server
186004	Web Service disabled for web server
186005	Web Server not defined
186006	Web Service not defined
186007	Web Service is wrong type.
186008	Invalid Server Lookup Key
186009	Server Lookup Key Missing Association

Glossary

AA

Authentication and Authorization

Business Delegate

A business delegate acts as a representative of the client components and is responsible for hiding the underlying implementation details of the business service. It knows how to look up and access the business services.

Certificate Authority (CA)

"A certificate authority (CA) is an entity that creates and then 'signs' a document or file containing the name of a user and his public key. Anyone can verify that the file was signed by no one other than the CA by using the public key of the CA. By trusting the CA, one can develop trust in a user's public key.

The trust in the certification authority's public key can be obtained recursively. One can have a certificate containing the certification authority's public key signed by a superior certification authority *(Root CA)* that he already trusts. Ultimately, one need only trust the public keys of a small number of top-level certification authorities. Through a chain of certificates *(Sub CAs)*, trust in a large number of users' signatures can be established.

A broader application of digital certification includes not only name and public key but also other information. Such a combination, together with a signature, forms an extended certificate. The other information may include, for example, electronic-mail address, authorization to sign documents of a given value, or authorization to sign other certificates."[1]

Currently, the Department of Veterans Affairs (VA) uses VeriSign, Inc. as the Certificate Authority (CA).

Cryptography

The system or method used to write or decipher messages in code (see "Encryption" and "Decryption").

CSR

Certificate Signing Request.

Decryption

Using a secret key to unscramble data or messages previously encrypted with a cipher or code so that they are readable. In some cases, encryption algorithms are one directional (i.e., they only encode and the resulting data *cannot* be unscrambled).

[1] DEA Web site (http://www.deadiversion.usdoj.gov/ecomm/e_rx/con_ops/index.html): "Public Key Infrastructure Analysis Concept of Operations," Section 3.4.3 "Public Key - The I in PKI"

Encryption

Scrambling data or messages with a cipher or code so that they are unreadable without a secret key. In some cases, encryption algorithms are one directional (i.e., they only encode and the resulting data *cannot* be unscrambled).

HTTP Protocol

Hyper Text Transfer Protocol is the underlying protocol used by the World Wide Web. HTTP defines how messages are formatted and transmitted, and what actions web servers and browsers should take in response to various commands.

HWSC

Health*e*Vet Web Services Client is a support framework that offers VistA/M applications real-time, synchronous client access to n-tier (J2EE) web services through the supplied M-based and Caché APIs.

Intermediate CA

Intermediate Certificate Authority. Currently, the Department of Veterans Affairs (VA) uses VeriSign, Inc. as the Certificate Authority (CA). VeriSign requires the use of an CA Intermediate Certificate. The CA Intermediate Certificate is used to sign the peer's (server) certificate. This provides another level of validation-managed PKI for SSL.

J2EE

The Java 2 Platform, Enterprise Edition (J2EE) defines the standard for developing multi-tier enterprise applications. J2EE defines components that function independently, that can be deployed on servers, and that can be invoked by remote clients. The J2EE platform is a set of standard technologies and is not itself a language. The current J2EE platform is version 1.4.

PKI

Public Key Infrastructure technology adds the following security services to an electronic ordering system:

- Confidentiality — only authorized persons have access to data.
- Authentication — establishes who is sending/receiving data.
- Integrity — the data has not been altered in transmission.
- Non-repudiation — parties to a transaction cannot convincingly deny having participated in the transaction."[2]

[2] DEA Web site (http://www.deadiversion.usdoj.gov/ecomm/e_rx/con_ops/index.html): "Public Key Infrastructure Analysis Concept of Operations," Section 3.3 "Security"

Private Certificate

This is the certificate that contains both the user's public and private keys. This certificate will reside on a smart card.

Public Certificate

This is the certificate that contains the user's public key. This certificate will reside in a file or database.

REST

Representational State Transfer (REST) is an architectural style for simplified web services, based on accessing resources via HTTP.

Root CA

Root Certificate Authority. In cryptography and computer security, a root certificate is an unsigned public key certificate, or a self-signed certificate, and is part of a public key infrastructure scheme. The most common commercial variety is based on the ITU-T X.509 standard. Normally an X.509 certificate includes a digital signature from a Certificate Authority (CA), which vouches for correctness of the data contained in a certificate. Root certificates are implicitly trusted.

Currently, the Department of Veterans Affairs (VA) uses VeriSign, Inc. as the Certificate Authority (CA).

Service Facade

The Service Façade acts as the server-side bridge between the Business Delegate and the capability. The Service Façade is responsible for taking a request from the delegate and doing any translation necessary to invoke the capability and provide the response to the delegate.

Servlet Container

A servlet is managed by a servlet container (formerly referred to as *servlet engine*.) The servlet container is responsible for loading and instantiating the servlets and then calling `init()`. When a request is received by the servlet container, it decides what servlet to call in accordance with a configuration file. A famous example of a servlet container is Tomcat.

The servlet Container calls the servlet's service() method and passes an instance of ServletRequest and ServletResponse. Depending on the request's method (mostly GET and POST), service calls `doGet()` or `doPost()`. These passed instances can be used by the servlet to find out who the remote user is, if and what HTTP POST parameters have been set and other characteristics.

Together with the web server (or *application server*) the servlet container provides the HTTP interface to the world.
It is also possible for a servlet container to run standalone (without web server), or to even run on a host other than the web server. [3]

[3] From the *ADP –Analyze, Design & Programming GmbH* website:

Signed Certificate

The Signed Certificate (a.k.a. self-signed certificate) is the peer's (server) digital certificate. Currently, the Department of Veterans Affairs (VA) uses VeriSign, Inc. as the Certificate Authority (CA) to sign (validate) digital certificates. VeriSign, Inc. requires the use of CA Root and Intermediate Certificates. The Subject and Issuer will have the same content when signed by VeriSign; the issuer will have VeriSign's content.

SOAP

Simple Object Access Protocol (SOAP) is a protocol for exchanging structured information over a network, often via HTTP.

SSL

Secure Socket Layer. A low-level protocol that enables secure communications between a server and a browser. It provides communication privacy.

TLS

Transport Layer Security. Transport Layer Security (TLS) and its predecessor, Secure Sockets Layer (SSL), are cryptographic protocols which provide secure communications on the Internet for such things as web browsing, e-mail, Internet faxing, instant messaging and other data transfers. There are slight differences between SSL 3.0 and TLS 1.0, but the protocol remains substantially the same.

Web Service

A web resource meant to be consumed over a network via HTTP, by an autonomous program.

WebLogic

A BEA product, WebLogic Server 8.1 is a J2EE- v1.3-certified application server for developing and deploying J2EE enterprise applications.

WSDL

Web Services Definition Language. "WSDL is an XML-based service description on how to communicate using web services. The WSDL defines services as collections of network endpoints, or ports. WSDL specification provides an XML format for documents for this purpose.

The abstract definition of ports and messages is separated from their concrete use or instance, allowing the reuse of these definitions. A port is defined by associating a network address with a reusable binding, and a collection of ports define a service. Messages are abstract descriptions of the data being exchanged, and port types are abstract collections of supported operations. The concrete protocol and data format specifications for a particular port type constitutes a reusable binding, where the messages and operations are then bound

http://www.adp-gmbh.ch/java/servlets/container.html to a concrete network protocol and message format. In this way, WSDL describes the public interface to the web service.

WSDL is often used in combination with SOAP and XML Schema to provide web services over the Internet. A client program connecting to a web service can read the WSDL to determine what functions are available on the server. Any special datatypes used are embedded in the WSDL file in the form of XML Schema. The client can then use SOAP to actually call one of the functions listed in the WSDL."[4]

ℹ For a comprehensive list of commonly used infrastructure- and security-related terms and definitions, please visit the Security and Other Common Services Glossary Web page at the following Web address:

http://vista.med.va.gov/iss/glossary.asp

For a comprehensive list of acronyms, please visit the Security and Other Common Services Acronyms Web site at the following Web address:

http://vista/med/va/gov/iss/acronyms/index.asp

[4] http://en.wikipedia.org/wiki/Web_Services_Description_Language