Configuring Secure Oracle E-Business Suite 11i Deployment Using Cisco Application Control Engine (ACE)

This document contains information for implementing SSL with Oracle E-Business Suite 11i. It provides guidance for a successful implementation and covers specific steps for configuring Oracle E-Business Suite using the SSL offload feature of the Cisco Application Control Engine (ACE) and various E-Business Suite application components such as Oracle HTTP Server and Forms Server.

The following sections are discussed:

Acronyms

- EBS—(Oracle) E-Business Suite
- OAM—Oracle Applications Manager
- ACE—(Cisco) Application Control Engine
- ANM—(Cisco) Application Networking Manager
- CLI—Command Line Interface
- VIP—Virtual IP

Corporate Headquarters:
Cisco Systems, Inc., 170 West Tasman Drive, San Jose, CA 95134-1706 USA

Prerequisites and Configuration Notes

The following requisites and configurations are available for noteworthy consideration.

- Oracle EBS installation of version 11.5.10 or higher with latest recommended patches
- Oracle EBS environment should be AutoConfig enabled
- Cisco ACE (version 2.1 or greater) must be configured in the network. Refer to detailed configuration procedures outlined in following document:

  http://www.cisco.com/en/US/docs/interfaces_modules/services_modules/ace/v3.00_A2/configuration/slb/guide/slbgd.pdf
- The connection from the client to ACE (SSL termination device) is HTTP over SSLv3/TLS (HTTPS); The ACE is configured to listen on TCP Port 443 for secure connections (HTTPS)
- The backend connection from ACE to Oracle Application Server is clear-text (HTTP); The Oracle application server is listening on TCP port 8000 for non-secure connections
- The load balancer (also acting as SSL accelerator) is doing port translation from TCP 443 on the front-end client side to TCP 8000 on the server-side application host
- Oracle Forms must be configured in servlet mode. For detailed information on configuring Oracle forms in servlet mode, please refer to Oracle Metalink ID 201340.1:

  http://metalink.oracle.com/metalink/plsql/ml2_documents.showFrameDocument?p_database_id=NOT&p_id=201340.1

ACE SSL Offload Feature

The integrated SSL (HTTPS) capabilities of the ACE allow for secure E-Business Suite transactions. The ACE provides hardware-based SSL acceleration, moving this processor-intensive functionality from the CPU or NIC of the server into the network. Centralized SSL services in the network allow secure transactions to be efficiently processed and inspected by other network-based services, such as IDS and IPS.

ACE and Oracle E-Business Suite Application Configuration Details

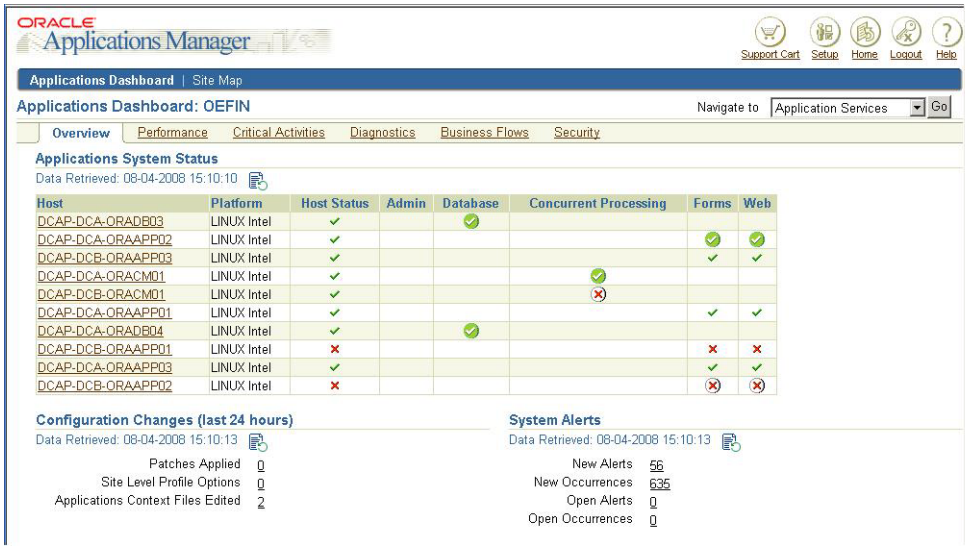
The remainder of this document outlines the application environment, including detailed instructions for implementing load balancing for Oracle EBS environments using Cisco ACE. This section also details modifications required for enabling:

- HTTP/HTTPS load balancing and session persistence
- Forms listener servlet (a pre-requisite required to enable HTTPS)
- SSL accelerator

The Oracle tool AutoConfig manages changes in the Oracle applications systems using an application context file. The context file uses an XML format and represents the application environment on a single node. The context file can be modified by leveraging Oracle Applications Manager (OAM) through a set of configuration wizards. Modifications made through OAM will be reflected in the environment only after running AutoConfig using the **adautocfg** script that is supplied by the software. Figure 1 shows an overview of the application and database nodes along with their status as reported by OAM.

Figure 1 shows the Oracle Applications Manager Dashboard

Figure 1 Oracle Applications Manager Dashboard



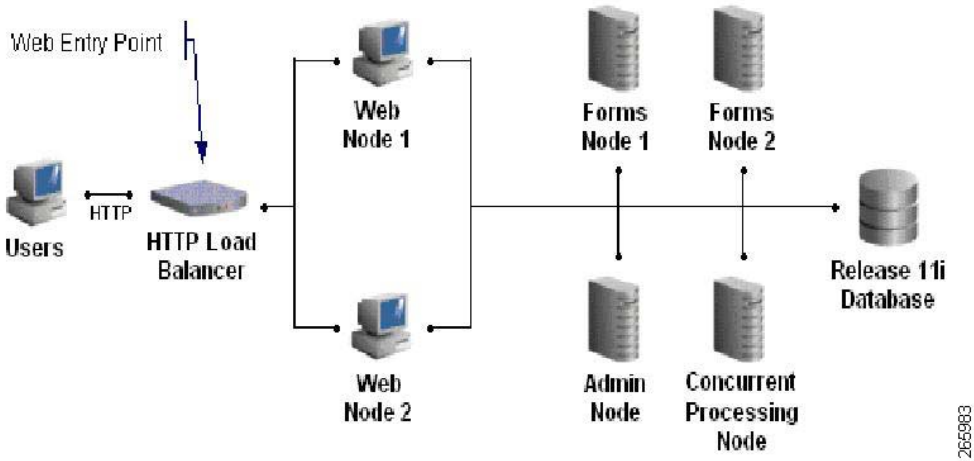
The following configurations are detailed:

- HTTP Load Balancing
- Configuring ACE for SSL

HTTP Load Balancing

Figure 2 shows how the hardware load balancer distributes connections across the multiple web nodes.

Figure 2 Load Balancing Connections



Note Hardware-based HTTP load balancers must be configured to ensure persistent session connections between clients and Web Server Nodes in Oracle E-Business Suite 11i environments. Maintaining session persistence is essential to ensure that a user session remains on the same middle tier throughout the duration of the session. If the session is not maintained, the users may experience a "Lost transaction context" message. For additional details please refer to Oracle Metalink node ID 456906.1: (http://metalink.oracle.com/metalink/plsql/ml2_documents.showFrameDocument?p_database_id=NOT&p_id=201340.1).

The following HTTP load balancing configurations are provided:

Configuring Application Web Nodes to Use Cisco ACE as Load Balancer for EBS 11i

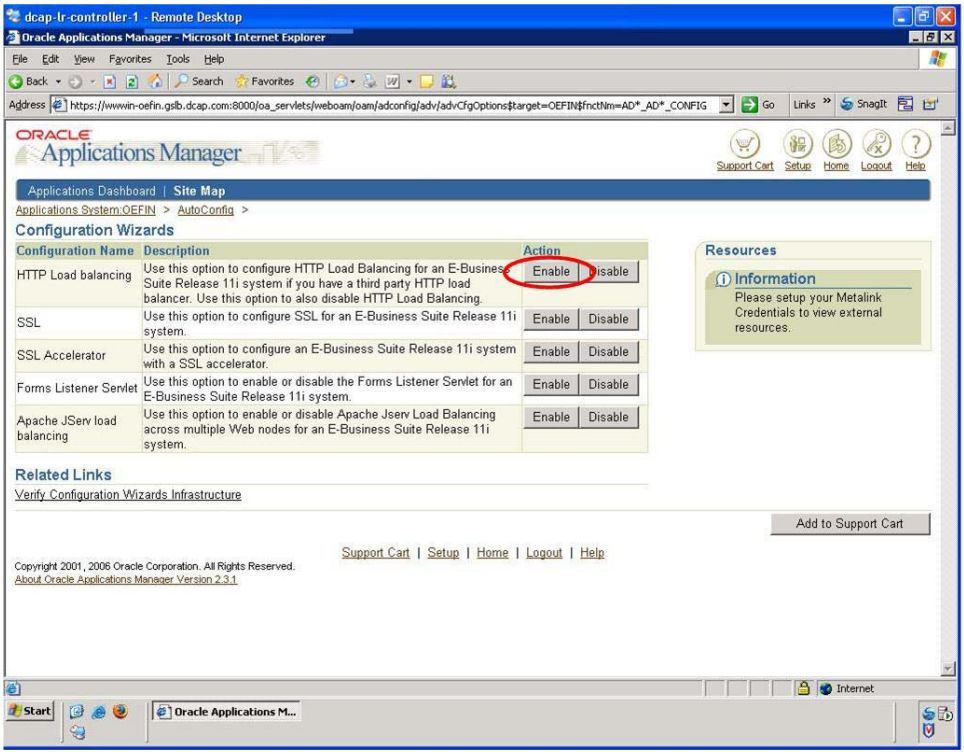
The following procedure is used to configure Application Web Nodes to use Cisco ACE as load balancer for EBS 11i.

Step 1 Log in to the Oracle Applications Manager as shown in Figure 3 and select the **Site Map - System Administration** tab, then select **AutoConfig**.

Step 2 Click **Launch Wizards**.

Step 3 Enable HTTP Load Balancing (Figure 3).

Figure 3 Enabling HTTP Load Balancing in Oracle Application Manager.

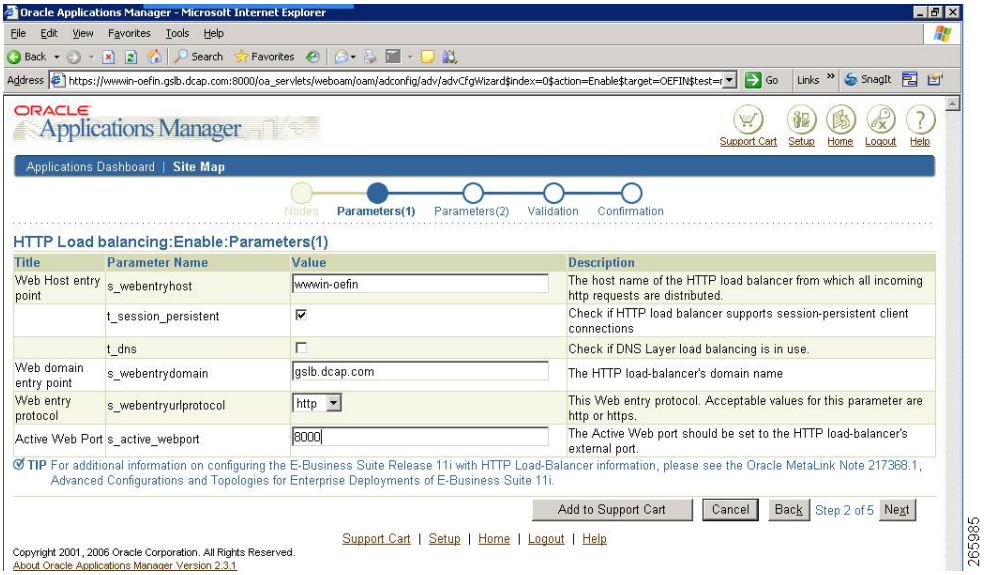


Step 4 Select the nodes to be included as real servers in the ACE server farm configuration.

Step 5 Provide the information for the following variables as shown in Figure 4 so that Oracle apps will create well-formed URLs.

- **s_webentryhost** – DNS name of the ACE VIP (eg: wwwin-oefin)
- **t_session_persistent** – Check this box if you will be using cookie-based stickiness
- **s_webentrydomain** – Domain name associated with VIP
- **s_webentryprotocol** – Select **https**
- **s_active_webport** – default value is 8000; Can be customized to user's choice

Figure 4 Configuring HTTP Load Balancer Variables in Oracle Applications Manager



Step 6 At this point, a series of validation and confirmation panels will be displayed. After successfully completing these steps, basic load balancing configurations will be saved on the context files of each server node selected.

Step 7 Stop the application services (using the **adstpall.sh** script) on each of the application nodes.

Step 8 Run the **autocfg** script on the respective application nodes and verify the log files to ensure there are no errors.

Step 9 Restart the application services using the **adstrtal.sh** script and verify that the load balancer is distributing the connections.

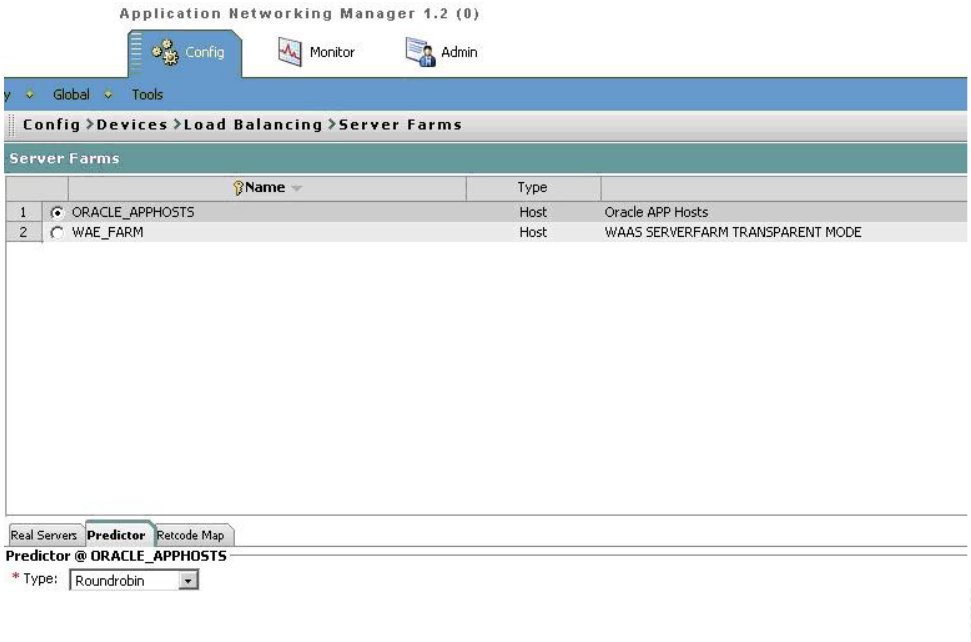
Configuring Server Load Balancing Policy in ACE Using ANM

The following procedure is used to configure server load balancing policy in ACE using ANM.

Step 1 Select the predefined server farm as shown in Figure 5.

Step 2 Select the Predictor tab and choose a predictor Type (the default is Roundrobin).

Figure 5 Configuring Load Balancing in ACE Using ANM



Configuring Server Load Balancing Policy Using ACE CLI

```
serverfarm host ORACLE_APPHOSTS
  predictor roundrobin
  probe ORACLE_DB_CHECK
  rserver ORACLE_APPHOST_1
    inservice
  rserver ORACLE_APPHOST_2
    inservice
  rserver ORACLE_APPHOST_3
    inservice
```

For additional documentation on SLB policies, please refer to the following section of the configuration guide:

http://www.cisco.com/en/US/docs/interfaces_modules/services_modules/ace/v3.00_A2/configuration/slb/guide/rsfarms.html#wpmkr1003481

Configuring Session/Cookie Persistence in ACE Using ANM

The following procedure is used to configure session and cookie persistence in ACE using ANM.

Step 1 Configure the appropriate sticky Group Name as shown in Figure 6.

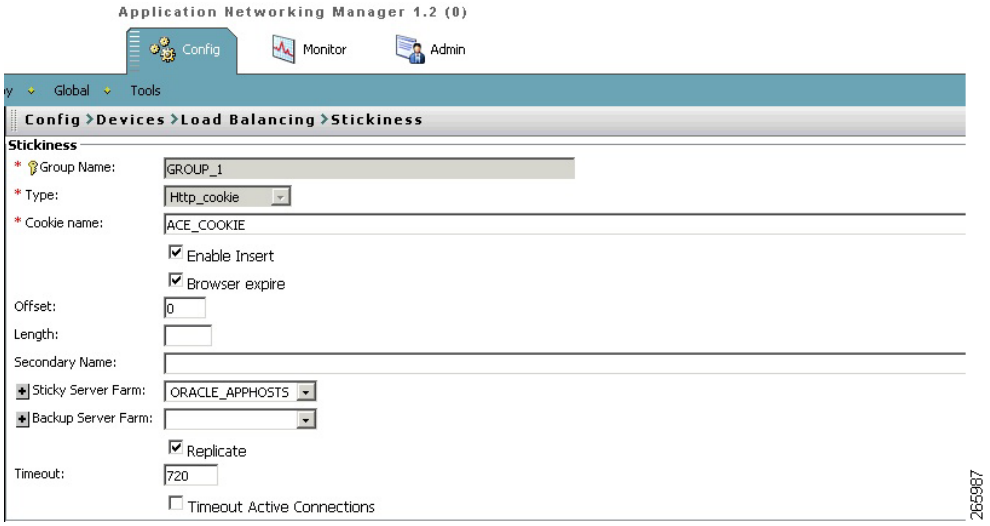
Step 2 Define a name for the HTTP cookie.

Step 3 Enable HTTP cookie insert by checking the box **Enable Insert**.

Step 4 (Optional) Enable the cookie to be a session cookie by checking the box **Browser expire**.

Step 5 Assign a Sticky Server Farm to the sticky group.

Figure 6 Configuring Cookie Sticky Using ANM



Configuring Session Cookie Persistence Using CLI

```
sticky http-cookie ACE_COOKIE_ORACLE GROUP_1
  cookie insert browser-expire
  replicate sticky
  serverfarm ORACLE_APPHOST
```

For additional documentation on how to configure Cisco ACE for session persistence, refer to following section of the documentation listed below.

http://www.cisco.com/en/US/docs/interfaces_modules/services_modules/ace/v3.00_A2/configuration/slb/guide/sticky.html#wpmkr1086901

Creating the HTTP Health Probe Using ANM

The procedure below for setting up health monitors for the Oracle application nodes is optional, but highly recommended. From the ANM config screen shown in Figure 7, follow the steps listed below.

Step 1 Define a name for the health probe. In this example, the name is defined as ORACLE_DB_CHECK.

Step 2 Choose HTTP for the probe Type.

Step 3 (Optional) Provide a description for the health probe.

Step 4 Assign a Probe Interval value (set to 2 seconds here).

Step 5 Assign a probe Pass Detect Interval value (set to 2 seconds here).

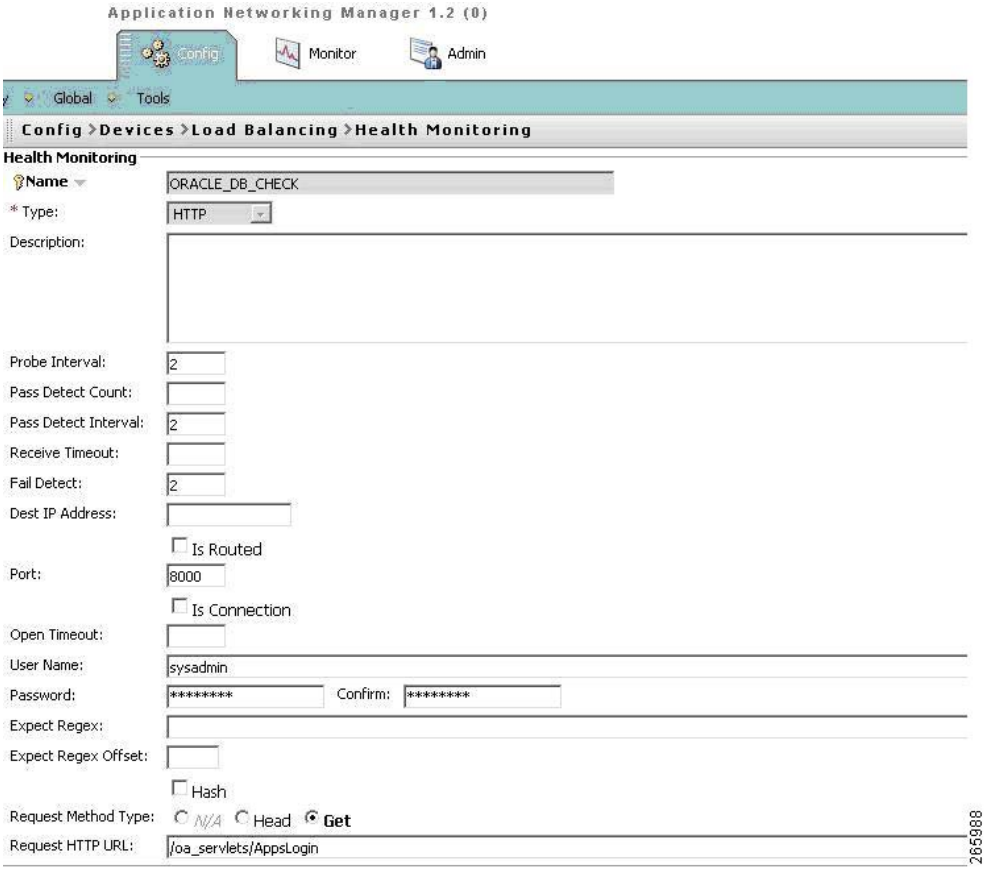
Step 6 Assign a probe Fail Detect value (set to 2 seconds here).

Step 7 Assign a Port number. In this example, the Port is configured to 8000.

Step 8 (Optional, but necessary for this example) Provide a User Name and Password that the probe will pass to the monitored URL (defined in Step 9) for health verification.

Step 9 Define the Request HTTP URL. In this example, the EBS Login URL is supplied to ensure the application is available by logging in using the credentials defined in Step 8, above.

Figure 7 Creating the HTTP Health Probe Using ANM



Creating the HTTP Health Probe Using ACE CLI

```
probe http ORACLE_DB_CHECK
  port 8000
  interval 2
  faildetect 2
  passdetect interval 2
  credentials sysadmin sysadmin
  request method get url /oa_servlets/AppsLogin
```

For additional documentation on creating health probes, please refer to the following section of ACE configuration guide.

http://www.cisco.com/en/US/docs/interfaces_modules/services_modules/ace/v3.00_A2/configuration/slb/guide/probe.html#wp1030892

Enabling Forms Servlet

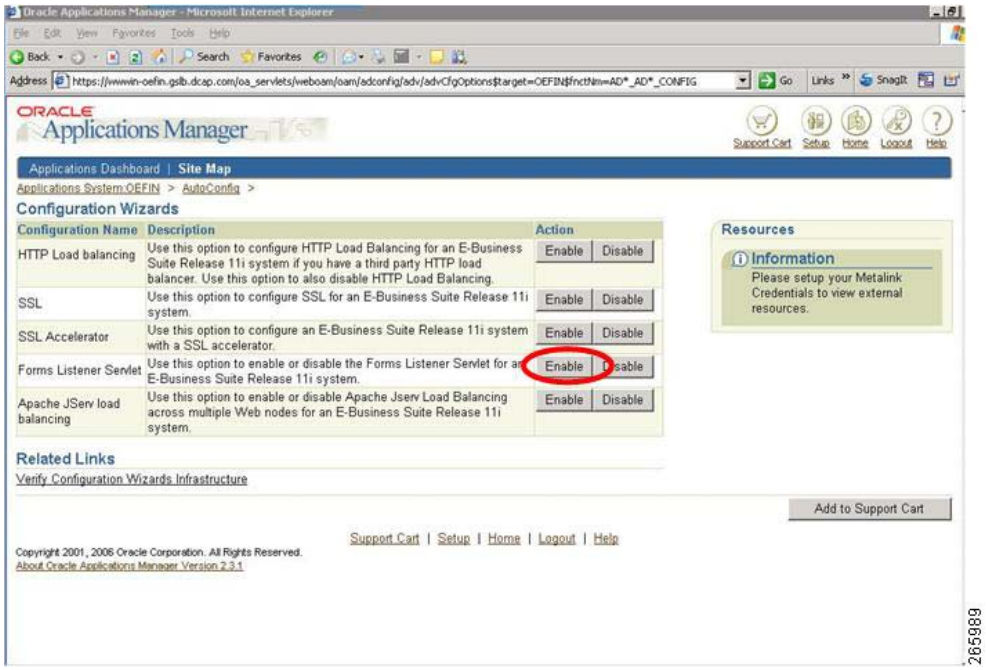
The forms listener servlet is necessary in order to enable Oracle applications to use HTTPS. The servlet mode allows the user to access forms via the web server. The following procedure is required to enable the forms listener servlet.

Step 1 Log in to the Oracle Applications Manager and select the **Site Map - System Administration** tab, then select **AutoConfig**.

Step 2 Click **Launch Wizards**.

Step 3 Click on the **Enable Forms Listener Servlet** button as shown in Figure 8.

Figure 8 Configuring Forms Servlet using OAM



Step 4 Select the application server nodes that will be using the forms listener servlet.

Step 5 Set the Forms Servlet URL variable (**s_forms_servlet_serverurl**) to the value **/forms/formservlet**, as shown in Figure 9.

Step 6 Ensure the Forms Servlet Comment variable (**s_forms_servlet_comment**) is blank.

Figure 9 Enabling Forms Listener Servlet Parameters

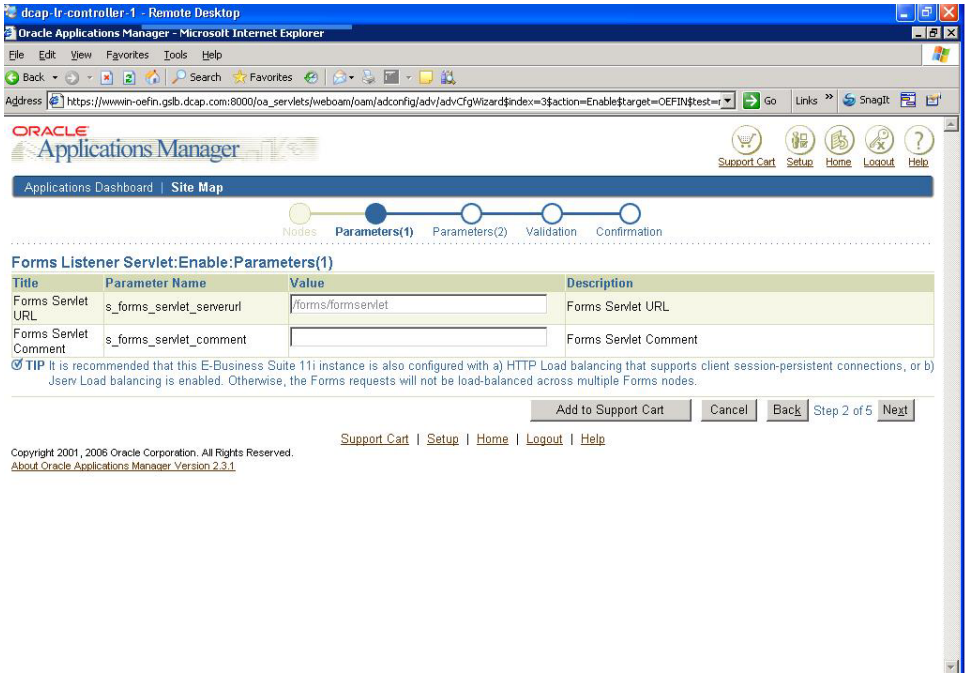
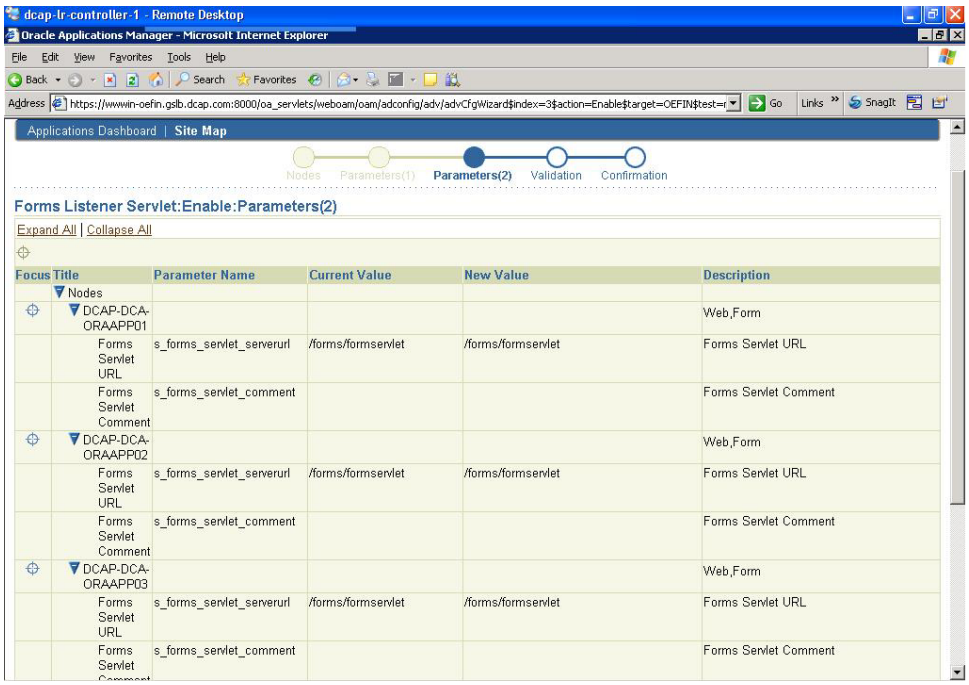


Figure 10 Validating Forms Listener Servlet Parameters



Step 7 At this point, a series of validation and confirmation panels will be displayed (Figure 10). After successful validation, the forms servlet configurations will be saved on the context files of each server node selected.

Step 8 Stop the application services (./adstpall.sh) on each of the application nodes.

Step 9 Run **autocfg** on the respective application nodes and verify the log files to ensure there are no errors.

Step 10 Restart the application services using the **adstrtal.sh** script and verify that the forms are opened in servlet mode.

Enabling SSL Accelerator

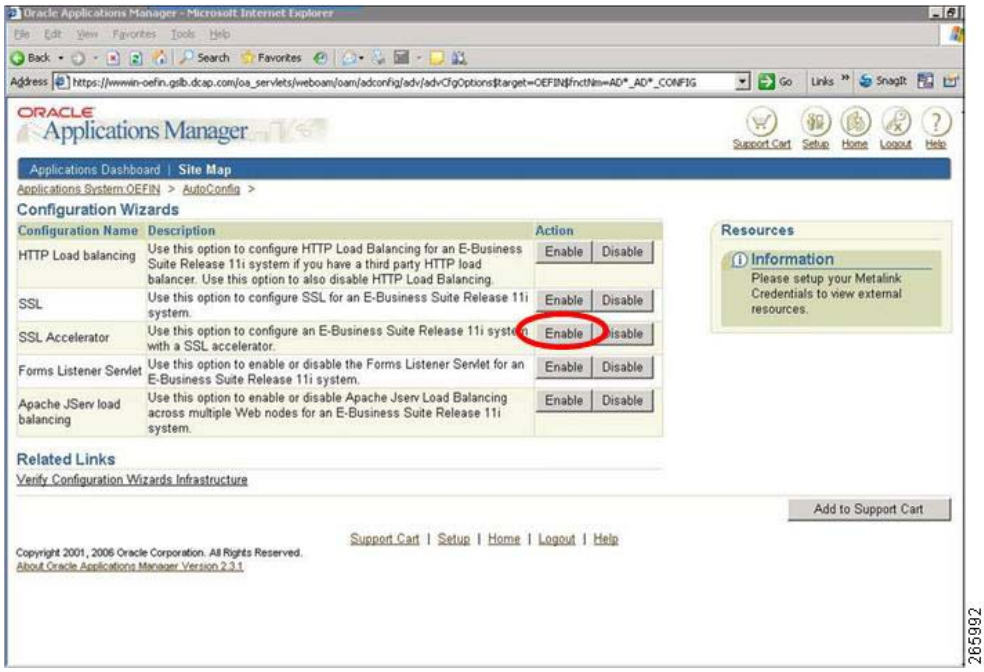
The SSL accelerator wizard configures the Oracle 11i application environment to use the Cisco ACE as the external device for encryption services and server offload. The following procedure summarizes how to enable SSL acceleration using Oracle Applications Manager.

Step 1 Log in to the Oracle Applications Manager and select the **Site Map - System Administration** tab, then select **AutoConfig**.

Step 2 Click **Launch Wizards**.

Step 3 Click on the **Enable SSL Accelerator** button as shown in Figure 11.

Figure 11 Enabling SSL Accelerator Using OAM



Step 4 Click on application nodes using the SSL Accelerator service.

Step 5 Provide the information for the following variables, as shown in Figure 12 and Figure 13, so that the Oracle applications will create well formed URLs.

- **s_webentryhost** – DNS name of the ACE VIP (eg: wwwin-oefin)
- **s_webentrydomain** – Domain associated with the ACE VIP
- **s_webentryurlprotocol** – Set to **https**.
- **s_active_webport** – Change this to a value of your choice, or 443, which is the default well-known HTTPS port value

- **s_webssl_port** – Change this to a value of your choice, or 443, which is the default well-known HTTPS port value

Figure 12 Enable the SSL Accelerator

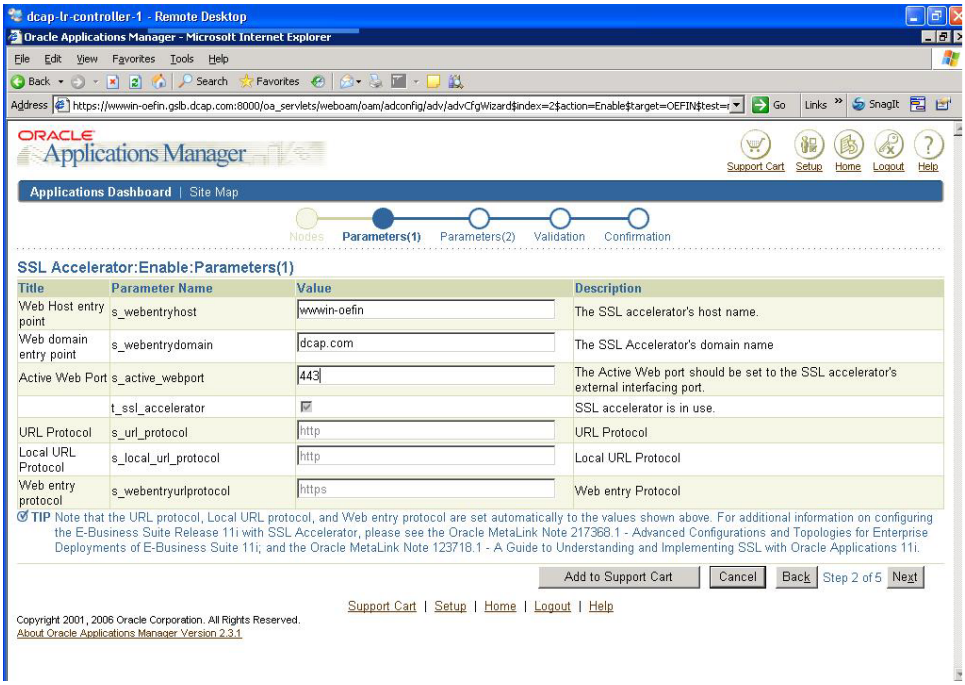
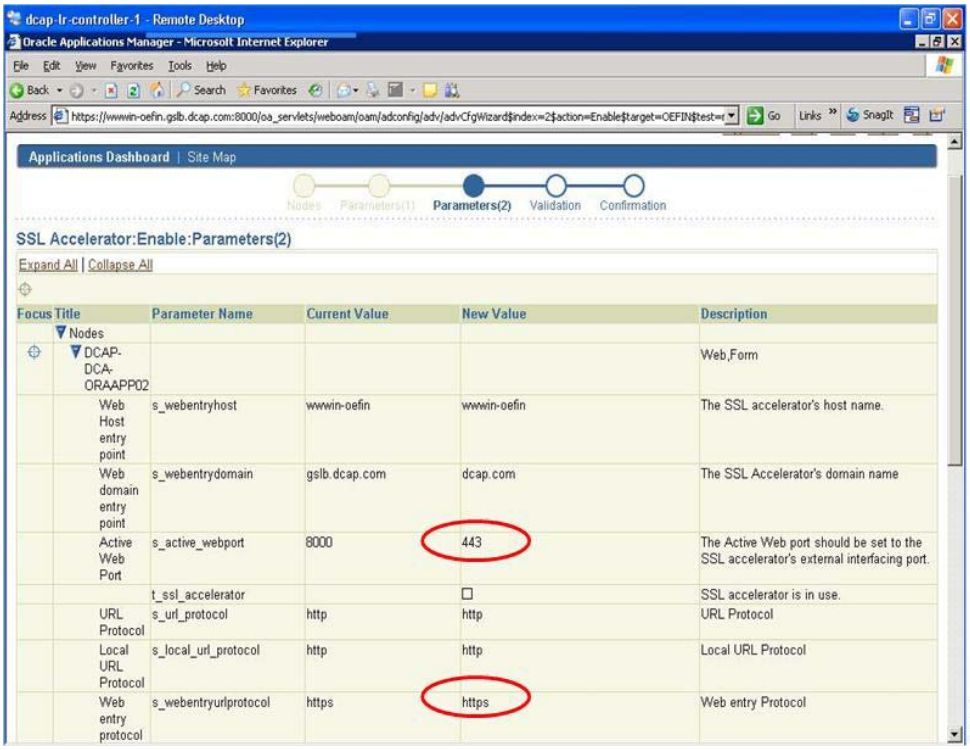


Figure 13 Enable the SSL Accelerator



Step 6 At this point a series of validation and confirmation panels will be displayed. After successfully completing these steps, basic load balancing configurations will be saved on the context files of each server node selected.

Step 7 Stop the application services (using the **adstpall.sh** script) on each of the application nodes.

Step 8 Run the **autocfg** script on the respective application nodes and verify the log files to ensure there are no errors.

Step 9 Restart the application services using **adstrtal.sh** script and verify the load balancer is distributing the connections.

Configuring the Virtual IP (VIP) on ACE using ANM

The following procedure defines the VIP and its properties. A screen capture from the ANM is shown in Figure 14.

Step 1 Define a name for the VIP. In this example name is defined as APPHOST_VIP.

Step 2 Assign an IP address to the VIP.

Step 3 Select the appropriate protocol. In this case, it's **tcp**.

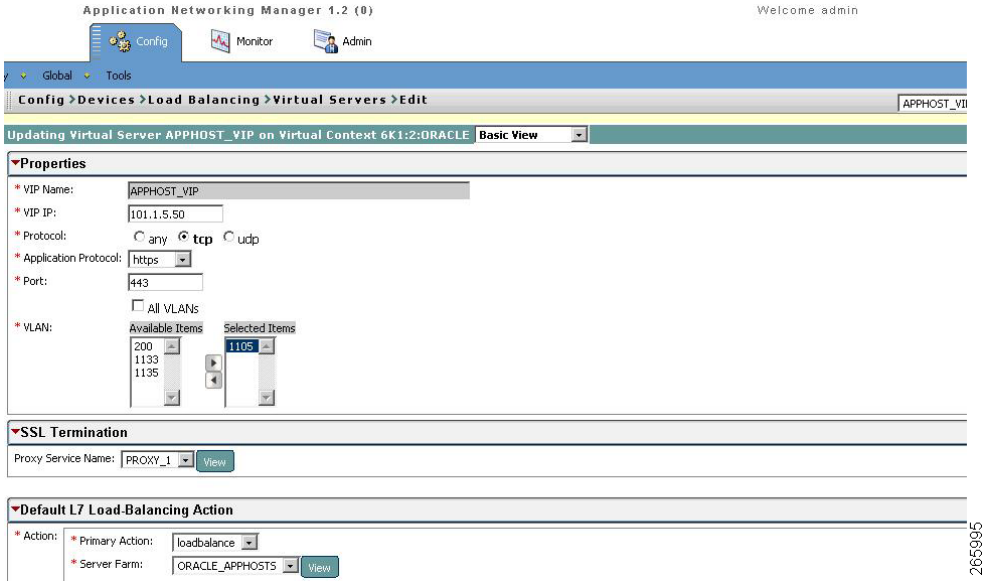
Step 4 Assign the application protocol. In this case it's **https**.

Step 5 Assign the **Port** number. In this case, port **443** has been chosen.

Step 6 Assign the VLAN that the IP address defined in Step 2 belongs to.

Step 7 Assign the SLB and choose the pre-defined server farm.

Figure 14 Configuring the VIP on Cisco ACE



Configuring Virtual IP (VIP) Using ACE CLI

```
class-map match-all APPHOST_VIP
  2 match virtual-address 101.1.5.50 tcp eq 443

policy-map multi-match VIPS
  class APPHOST_VIP
    loadbalance vip inservice
    loadbalance policy APPHOST_POLICY
    loadbalance vip icmp-reply
    appl-parameter http advanced-options RE_USE
    ssl-proxy server PROXY_1
```

Configuring ACE for SSL

This section of the document focuses on configuring the Cisco ACE for SSL. The document details how to configure parameter maps, SSL proxy services and class maps to build policy maps that determine the flow of information between the client, the ACE and the Oracle application hosts.

The following ACE to SSL configurations are provided:

Configuring Parameter Maps Using ANM

The following procedure is used for configuring a parameter map using ANM. Figure 15 shows an ANM screen involved in creating a parameter map.

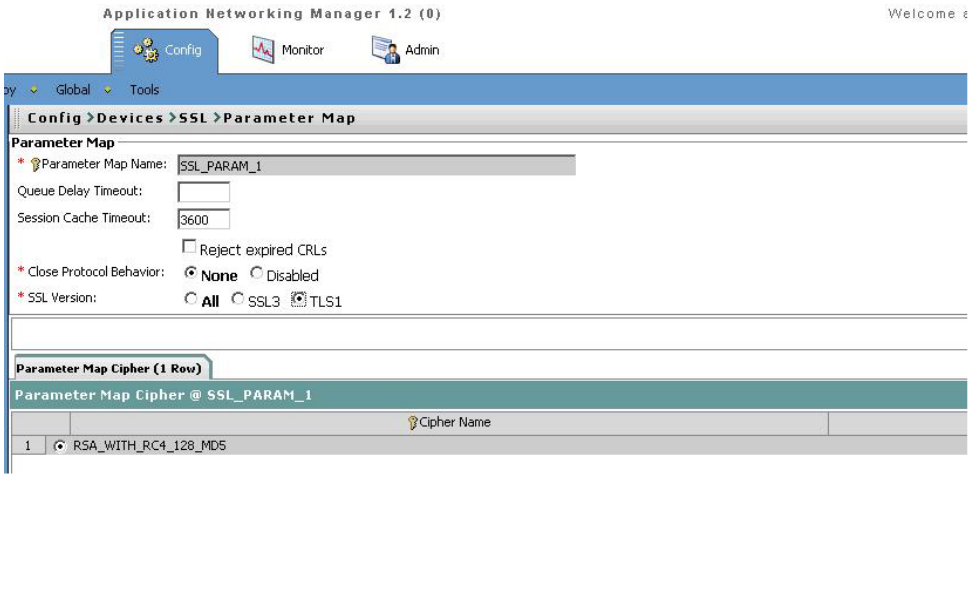
Step 1 Define a name for the Parameter Map.

Step 2 Define the SSL Session Cache Timeout.

Step 3 Select the **Close Protocol Behavior** (default is **None**).

Step 4 Select the SSL Version.

Figure 15 Creating a Parameter Map



265996

Configuring Parameter Map Using ACE CLI

```
parameter-map type ssl SSL_PARAM_1
  cipher RSA_WITH_RC4_128_MD5
```

For additional documentation on configuring parameter map, please refer to the following section of the configuration guide:

http://www.cisco.com/en/US/docs/interfaces_modules/services_modules/ace/v3.00_A2/configuration/ssl/guide/initiate.html#wpmkr1075636

Digital certificates and key pairs are a form of digital identification for user authentication. CA's, such as VeriSign and Thawte, issue certificates that attest to the validity of the public keys they contain. A client or server certificate includes the following identification attributes:

- Name of the CA (the certificate issuer) and CA digital signature
- Serial number
- Name of the client or server (the certificate subject) that the certificate authenticates
- Subject's public key
- Time stamps that indicate the certificate's expiration date

A CA has one or more signing certificates that it uses for creating SSL certificates and certificate revocation lists (CRL). Each signing certificate has a matching private key that is used to create the CA signature. The CA makes the signing certificates (with the public key embedded) available to the public, enabling anyone to access and use the signing certificates to verify that an SSL certificate or CRL was actually signed by a specific CA.

The ACE requires certificates and corresponding key pairs for SSL termination.

The ACE acts as an SSL proxy server and terminates the SSL session between it and the client.

Configuring the SSL Key

The following procedure is used for configuring a SSL Key. Figure 16 shows an ANM screen involved in creating the SSL Private Key.

Step 1 Define a name for the Private Key.

Step 2 Define the Private Key size (default is 1024)

Step 3 Enable the check box for “Exportable Key” if you would like the Private Key to be Exportable via ACE.

Figure 16 Creating a Private Key

265997

Importing the SSL Certificate

You can import certificate and key pair files to the ACE from a remote secure server. To transfer these files, we recommend that you use a secure encrypted transport mechanism between the ACE and the remote server. The ACE supports the Secure Shell protocol (SSHv2), which provides secure encryption

communications between two hosts over an insecure network. The ACE supports file transport between network devices using Secure File Transfer Protocol (SFTP), File Transfer Protocol (FTP), and Trivial File Transfer Protocol (TFTP).

The following procedure is used for importing the SSL certificate. Figure 17 shows an ANM screen involved in importing the SSL Public Certificate.

Step 1 Define the protocol used to import the certificate

Step 2 Define the IP Address of the server where the certificate resides

Step 3 Define the Remote Filename of the certificate

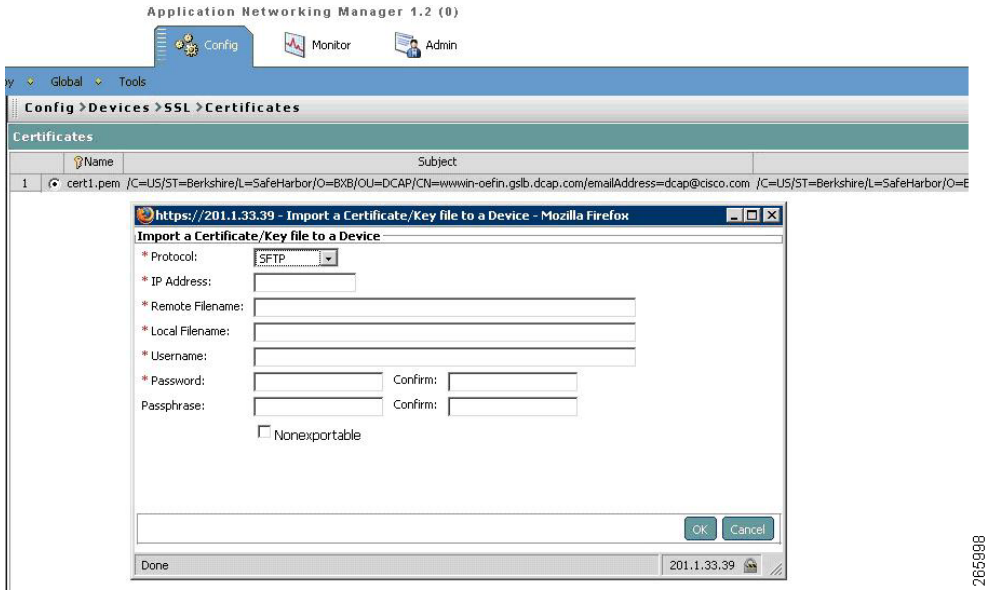
Step 4 Define the Local Filename of the certificate

Step 5 Define the Username and Password

Step 6 Define a Passphrase used to access the certificate (Optional)

Step 7 Enable the check box for "Nonexportable" if you would like the Certificate to be nonexportable via ACE.

Figure 17 Creating a SSL Certificate



Importing the SSL Certificate Using CLI

```
crypto import sftp IP_ADDRESS Passphrase

crypto import sftp 1.1.1.1 Username Password Localfile
```

For additional documentation on configuring Keys and Certificates please refer to the following section of the configuration guide:

http://www.cisco.com/en/US/docs/interfaces_modules/services_modules/ace/v3.00_A1/configuration/ssl/guide/certkeys.html

Configuring SSL Proxy Services Using ANM

The following procedure is used for configuring SSL proxy services using ANM. Figure 18 shows an ANM screen involved in configuring SSL proxy services on the Cisco ACE.

Step 1 Select the pre-defined SSL Proxy Service Name.

Step 2 Select the pre-defined SSL private key.

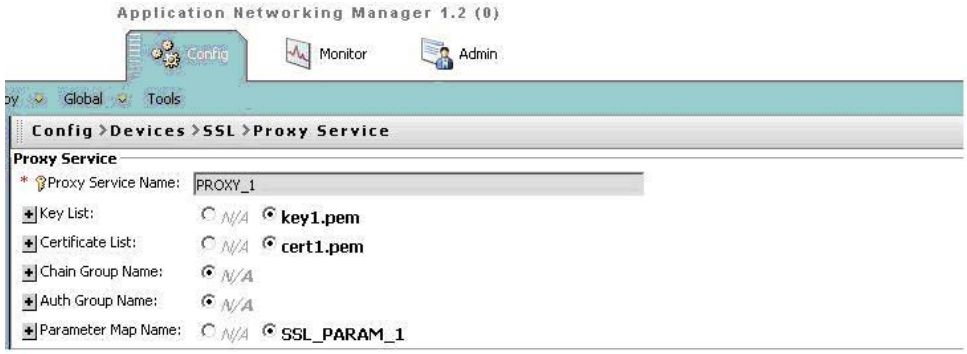
Step 3 (Optional) Select the SSL Chain Group.

Step 4 (Optional) Select the SSL Auth Group.

Step 5 Select the pre-defined SSL public certificate.

Step 6 Select the pre-defined SSL Parameter Map Name.

Figure 18 Configuring SSL Proxy Services



265999

Configure SSL Proxy Services Using CLI

```
parameter-map type ssl SSL_PARAM_1
  cipher RSA_WITH_RC4_128_MD5
!
ssl-proxy service PROXY_1
  key key1.pem
  cert cert1.pem
  ssl advanced-options SSL_PARAM_1
```

For additional documentation on configuring SSL proxy services please refer to the following section of the configuration guide:

http://www.cisco.com/en/US/docs/interfaces_modules/services_modules/ace/v3.00_A2/configuration/slb/guide/classlb.html#wpmkr1076742

References

The following Oracle and Cisco reference material is available.

Oracle Reference Documents

1. Using Autoconfig to Manage system configurations with Oracle Applications 11i (Metalink Noteid 165195.1)

   http://metalink.oracle.com/metalink/plsql/ml2_documents.showFrameDocument?p_database_id=NOT&p_id=165195.1

2. Advanced Configurations and Topologies for Enterprise Deployments of E-business Suite 11i (Metalink Noteid 217368.1)

   http://metalink.oracle.com/metalink/plsql/ml2_documents.showFrameDocument?p_database_id=NOT&p_id=217368.1

Cisco Reference Documents

1. Cisco Application Control Engine Configuration Guide:

   http://www.cisco.com/en/US/docs/interfaces_modules/services_modules/ace/v3.00_A1/configuration/administration/guide/admgd.html