Oracle® Communications Network Integrity

Security Guide

Release 7.3.2

E66033-01

May 2016

Oracle Communications Network Integrity Security Guide, Release 7.3.2

E66033-01

Contents

3 Implementing Network Integrity Security

4 Security Considerations for Developers

A Network Integrity Secure Deployment Checklist

Preface

This guide provides guidelines and recommendations for setting up Oracle Communications Network Integrity in a secure configuration.

Audience

This guide is intended for system administrators, database administrators, developers, and integrators, who work with Oracle Communications Network Integrity.

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 that have purchased support 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.

Document Revision History

The following table lists the revision history for this guide:

Version	Date	Description
E66033-01	May 2016	Initial release.

1

Network Integrity Security Overview

This chapter provides an overview of Oracle Communications Network Integrity security.

Basic Security Considerations

The following principles are fundamental to using any application securely:

- **Keep software up to date.** This includes the latest product release and any patches that apply to it.
- **Limit privileges as much as possible.** Users should be given only the access necessary to perform their work. User privileges should be reviewed periodically to determine relevance to current work requirements.
- **Monitor system activity.** Establish who should access which system components, how often they should be accessed, and who should monitor those components.
- **Install software securely.** For example, use firewalls, secure protocols (such as SSL), and secure passwords. See "Performing a Secure Network Integrity Installation" for more information.
- **Learn about and use Network Integrity security features.** See "Implementing Network Integrity Security" for more information.
- **Use secure development practices.** For example, take advantage of existing database security functionality instead of creating your own application security. See "Security Considerations for Developers" for more information.
- **Keep up to date on security information.** Oracle regularly issues security-related patch updates and security alerts. You must install all security patches as soon as possible from the Oracle Technology Network web site:

  `http://www.oracle.com/technetwork/topics/security/alerts-086861.html`

Understanding the Network Integrity Environment

When planning your Network Integrity implementation, consider the following:

- **Which resources must be protected?**

  For example:

  - You must protect customer data.
  - You must protect internal data, such as proprietary source code.

- You must protect system components from being disabled by external attacks or intentional system overloads.

- **Who are you protecting data from?**

  For example, if your business has service subscribers, you must protect their data from other subscribers, but someone in your organization might have to access that data to manage it. You can analyze your workflows to determine who needs access to the data; for example, a system administrator could manage your system components without needing to access the system data.

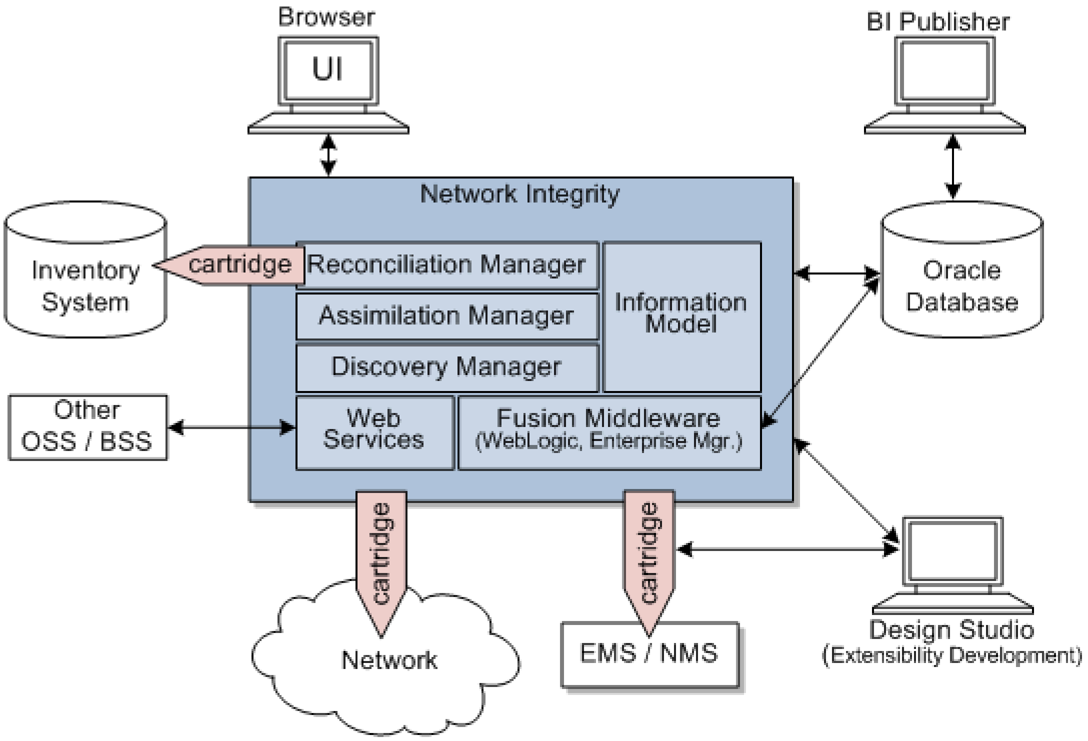
- **What happens if protections on strategic resources fail?**

  In some cases, a fault in your security scheme is nothing more than an inconvenience. In other cases, a fault might cause great damage to you or your customers. Understanding the security ramifications of each resource helps you protect it properly.

Overview of Network Integrity Security

Figure 1–1 shows all the various components that can comprise Network Integrity, including the components to which it connects. Each installed or integrated component requires special steps and configurations to ensure system security.

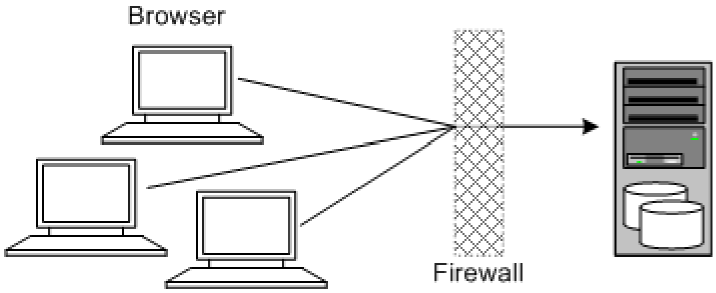
Figure 1–1 Network Integrity Components



Recommended Deployment Topologies

Figure 1–2 shows a single-computer installation topology: the simplest Network Integrity deployment architecture.

Figure 1–2 Single-Computer Deployment



In this topology, all the application components and data are kept on a single system, protected from external attacks by a firewall. The firewall can be configured to block known illegal traffic types. There are fewer resources to secure because all the components are on a single system and all the communication is local. Fewer ports have to be opened through the firewall.

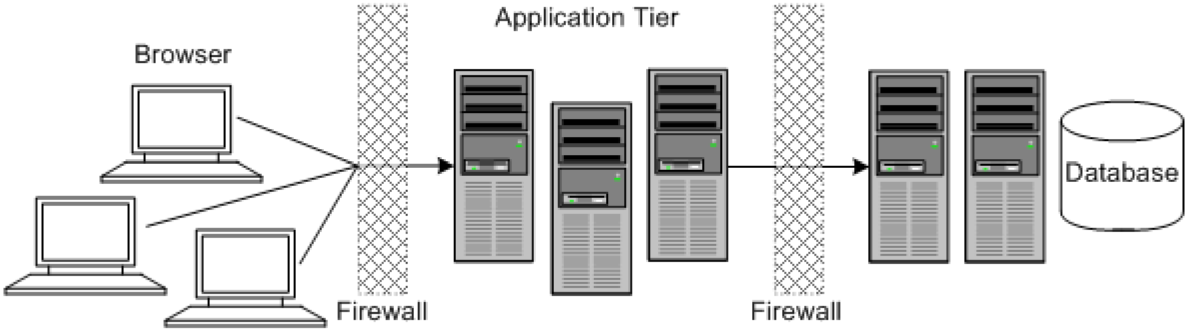
Conversely, there are fewer points of attack, and if security is compromised, an attacker would have access to the entire system and data.

A single-computer installation topology is best suited for test and lab environments.

A single-computer deployment is cost effective for small organizations but does not provide high availability because all components are stored on a single system.

Figure 1–3 shows a tiered installation deployment: a scalable Network Integrity deployment offering greater security and high availability.

Figure 1–3 Tiered Deployment



In this topology, the application tier is isolated by firewalls from both the Internet and the intranet. The database and servers are protected from potential attacks by two layers of firewall. Both firewalls can be configured to block known illegal traffic types. The two layers of firewall provide intrusion containment. Although there are a greater number of components to secure, and more ports have to be opened to allow secure communication between the tiers, the attack surface is spread out.

Operating System Security

This section lists Network Integrity-specific OS security configurations. This section applies to all supported OSs.

Firewall Port Configuration

Network Integrity communicates through the firewall with various components on specific ports. Ensure that the OS IPtables for the firewalls are configured to manage traffic on the following ports:

- Port 22 (optional, both directions): Used by the File Transfer and Parsing cartridge for SSH communication. Close this port if you are not using the File Transfer and Parsing cartridge.
- WebLogic Server SSL listen ports (both directions): Used by Administration and Managed servers for listening for traffic.
- CORBA Name Server port or SSL port (optional, both directions): Used by the CORBA cartridge and Optical TMF814 CORBA cartridge to retrieve device data from element and network management systems. Close this port if you are not using the CORBA and Optical TMF814 CORBA cartridges.
- Port 992 (optional, both directions): Used by the TL1 cartridge for SSH communication. Close this port if you are not using the TL1 cartridge.
- SNMP port (optional, both directions): Used for scanning SNMP devices. By default, the SNMP port is port 161. Close this port if you are not using an SNMP cartridge.
- Oracle Database listener ports: Used by the Oracle Database for listening for traffic.

Close all unused ports, especially non-SSL ports. Opt for SSL-enabled ports, instead of non-SSL ports, for all communications (for example: HTTPS, IIOPS, t3s).

For more information about securing your OS, see your OS documentation.

Oracle Database Security

This section lists the Network Integrity-specific security configurations for the Oracle Database:

- Data Encryption
- Secure Database Connections

For more information about securing Oracle Database, see *Oracle Database Security Guide* and *Oracle Database Advanced Security Administrator's Guide*.

Data Encryption

Network Integrity encrypts sensitive information and stores the encrypted data in the database. The Network Integrity encryption mechanism uses the advanced encryption standard (AES) algorithm.

The following data is encrypted:

- Scan parameters
- Inventory system configuration passwords

It is also possible (but not recommended) to encrypt the Network Integrity tablespace and schema, at the expense of system performance. Encrypting the schema and tablespace is not necessary, because the database is sufficiently secure without the encryption.

If you must encrypt the tablespace and schema, use Oracle Database Transparent Data Encryption (TDE), because it supports AES.

You must configure TDE on the tablespace before creating the Network Integrity schema. See *Oracle Database Advanced Security Administrator's Guide* for more information.

Secure Database Connections

Encrypting network data is a critical security measure that ensures that data travelling over the network is difficult to intercept and access.

Secure network connections to the Oracle Database using the Oracle Advanced Security feature. You can configure the Oracle Database with either Network Data Encryption and SSL authentication, as both ensure that the data is secure while travelling over the network.

The Oracle Advanced Security feature also provides security against the following types of attacks:

- Data modification attack, where an unauthorized party intercepts data in transit over the network, alters it, and transmits the altered data to the database.
- Replay attack, where an unauthorized party repeatedly transmits entire sets of valid data.

SSL Authentication

Use the Oracle Advanced Security feature to enable SSL authentication, using a digital certificate, on data that travels over the network to the database. See *Oracle Database Advanced Security Administrator's Guide* for more information.

Using SSL authentication allows Network Integrity to communicate with servers over an encrypted connection and to communicate with the database over an encrypted connection.

SSL authentication supports the following authentication modes:

- Only the server authenticates itself to the client.
- Both client and server authenticate themselves to each other.
- Neither the client nor the server authenticate with each other (SSL encryption feature by itself).

WebLogic Server Security

For information about securing WebLogic Server, see *Oracle Fusion Middleware Securing a Production Environment for Oracle WebLogic Server*.

WebLogic Server Authorization

Authorization is the process where the interactions between users and WebLogic Server resources are controlled, based on user identity or other information. In WebLogic Server, an Authorization provider is used to limit the interactions between users and WebLogic resources to ensure integrity, confidentiality, and availability.

For more information about changing WebLogic server passwords, see *Network Integrity System Administrator's Guide*.

WebLogic Resources

A WebLogic Server resource is a structured object used to represent an underlying WebLogic Server entity, which can be protected from unauthorized access using security roles and security policies.

WebLogic resources are hierarchical. Therefore, the level at which you define these security roles and security policies is up to you. For example, you can define security roles and security policies on entire enterprise applications; an Enterprise Java Bean JAR containing multiple EJBs; a particular Enterprise Java Bean (EJB) within that JAR; or a single method within that EJB.

Security Policies

Security policies replace access control lists and answer the question “Who has access to a WebLogic server resource?” A security policy is created when you define an association between a WebLogic resource and one or more users, groups, or security roles. You can optionally define date and time constraints for a security policy. A WebLogic resource has no protection until you assign it a security policy.

Security policies are stored in an authorization provider's database. By default, the XACML Authorization provider is configured in a domain, and security policies are stored in the embedded LDAP server.

To use a user or group to create a security policy, the user or group must be defined in the security provider database for the authentication provider that is configured in the default security realm. To use a security role to create a security policy, the security role must be defined in the security provider database for the Role Mapping provider that is configured in the default security realm. By default, the authentication and XACML Role Mapping providers are configured in the database in the embedded LDAP server. Also by default, security policies are defined in WebLogic Server resources. These security policies are based on security roles and default global groups. You also have the option of basing a security policy on a user.

Secure Sockets Layer (SSL)

SSL enables secure communication between applications connected through the web. WebLogic Server fully supports SSL communication. By default, WebLogic Server is configured for one-way SSL authentication. Using the Administration Console, you can configure WebLogic Server for two-way SSL authentication.

- To use one-way SSL from a client to a server, enable the SSL port on the server, configure identity for the server and trust for the client.
- To use two-way SSL between a client and a server, enable two-way SSL on the server, configure trust for the server, and identity for the server.

  In either case, the trusted CA certificates must include the trusted CA certificate that issued the peer's identity certificate. This certificate does not necessarily have to be the root CA certificate.

To acquire a digital certificate for your server, generate a public key, private key, and a Certificate Signature Request (CSR), which contains your public key. Send the CSR request to a certificate authority and follow their procedures for obtaining a signed digital certificate.

After you have your private keys, digital certificates, and any additional trusted CA certificates that you may need, you must store them so that WebLogic Server can use them to verify identity. Store private keys and certificates in keystores.

For more information on security fundamentals, see the Oracle Fusion Middleware documentation:

http://www.oracle.com/technology

Network Integrity uses HTTPS to connect to the UI, by default. You must enable both SSL and TLS for your browser to connect to the Network Integrity UI.

See *Network Integrity Installation Guide* and *Network Integrity System Administrator's Guide* for information about the CA certificate used by the WebLogic server.

LDAP Security

Oracle recommends that you use Oracle Internet Directory for identity management (for example, users, roles, certificates). You can also use an external LDAP, which you must integrate with Network Integrity through the WebLogic Application Server.

For information about setting up Oracle Internet Directory, see *Oracle Fusion Middleware Administrator's Guide for Oracle Internet Directory*.

For information about setting up an external LDAP, see the LDAP application documentation. For information about Security Realms and setting up Network Integrity with the external LDAP, see *Network Integrity System Administrator's Guide*.

Logging Security

Oracle recommends that you not use FINE or lower logging levels for Network Integrity. By default, WebLogic Server uses a higher log level for Network Integrity. An explicit administrative action is required to change the log level. See *Network Integrity System Administrator's Guide* for more information.

When the log levels are set to FINE or lower, the logs can contain raw exceptions and stack traces that could be exploited to compromise the security of your Network Integrity system.

Oracle Security Documentation

Network Integrity uses other Oracle products, such as Oracle Database and Oracle WebLogic Server. See the following documents, as they apply to Network Integrity:

- *Oracle Database Security Guide*
- *Oracle Fusion Middleware Securing a Production Environment for Oracle WebLogic Server*
- *Oracle Application Server Security Guide*
- *Oracle Application Server Administrator's Guide*

2

Performing a Secure Network Integrity Installation

This chapter presents planning information for your Oracle Communications Network Integrity system and describes recommended deployment topologies that enhance security.

For more information about installing Network Integrity, see *Network Integrity Installation Guide*.

Installing Network Integrity Securely

You can perform a custom installation or a typical installation. Perform a custom installation to avoid installing options and products you do not need. If you perform a typical installation, however, you can always remove or disable features you do not need after the installation is complete.

When installing Network Integrity, do the following:

- When creating the WebLogic Server domain for Network Integrity:
  - Make sure that SSL ports are being used on the Administration Server and all Managed servers.
  - If installing Network Integrity on a cluster of servers, configure the cluster addresses to use SSL ports.
  - After you have created the WebLogic Server domain for Network Integrity, start the Administration Server. Then, use t3s to start the Managed servers:

    ```
    startManagerServer.sh ManagedServer_1 t3s://host_name
    ```

    where *ManagedServer_1* is the name of the first Managed server, and *host_name* is the host name of the Administration server.
- Using the WebLogic Administration Console, configure Certificate Identity and trust store to use SSL. Do not use the default, demonstration certificate that comes with WebLogic Server. See the WebLogic administrator's documentation for more information.
- During the installation of Network Integrity, on the Disable unsecured Listen Port screen of the Oracle Universal Installer, select the **Disable all the non-SSL ports** check box to secure all communication between components, and JCA and JMS collection, over SSL ports.

Secure File System Access

Consider the following when planning your Network Integrity installation:

- Access to files created during the installation is limited. To have access to the created files, the installer must have root or admin access.
- Data Source passwords are encrypted using the Oracle AES algorithm. The encrypted passwords are stored in WebLogic Server configuration files.

Configure the following directories with the following permission settings:

- *WL_Home* and all its subdirectories: **750** permissions, but all files you create should be set with **640**.
- *Domain_Home* and all its subdirectories: **750** permissions, but all files you create should be set with **640**.
- *NI_Home* and all its subdirectories (a temporary directory used during installation): **750** permissions, but all files you create should be set with **640**.

Set secure file system access permissions for the Oracle database and Oracle Internet Directory.

Note: The Network Integrity Installer never writes or records its schema or base user account information to any file.

About Password Policies

Oracle recommends having strong password policies for Network Integrity and database schema users. Consider enforcing the following password policies:

- Minimum length of password is eight characters.
- Password must contain at least one digit, one capital letter, and one special character. For example: `WebLogic@123`.
- The user name must not be part of the password.

As a minimum, passwords must be at least eight characters long and contain at least one non-alphabetic character.

Stricter rules can be set for the authentication provider using the WebLogic Administration console. For details on authentication providers and their configuration, refer to WebLogic administrator documentation.

See *Network Integrity System Administrator's Guide* for information about changing and setting Network Integrity passwords.

Securely Installing Cartridges

Oracle recommends installing Network Integrity cartridges over SSL. For details on installing or deploying the cartridges over SSL, see the Oracle Cartridge Deployer documentation.

For the File Transfer and Parsing cartridge, enable secure file transfer. See *Network Integrity File Transfer and Parsing Cartridge Guide* for more information.

Securely Integrating BI Publisher with Network Integrity

Oracle Business Intelligence Publisher (BI Publisher) is installed into a WebLogic Server domain. When installing BI Publisher, configure it to communicate with the

Oracle Database over an SSL-enabled channel, and disable all unused ports, especially unsecured ports. See the BI Publisher documentation for more information.

Post-Installation Configuration

This section explains security configurations to complete after Network Integrity is installed.

Setting Up User Accounts to Lock and Expire

Create Network Integrity user accounts to lock after a certain number of failed log in attempts, and to expire after a certain amount of idle time.

See *Network Integrity System Administrator's Guide* for information about changing and setting Network Integrity passwords.

Enabling SSL for Network Integrity Data Sources

When the Oracle Database communicates with Network Integrity through an SSL-enabled port, the following data source connections must also be configured to enable SSL communication:

- CMWSPersistentDS
- JobDispatcherDS
- JobDispatcherPersistentDS
- mds-commsNIRepository
- NIDataSource
- NIPersistentDS
- NIPomsPersistentDS

For information about configuring data sources, see *Oracle Database Security Guide*.

Enabling SSL for LDAP Authentication Provider

For secure communication between WebLogic Server and an external LDAP, enable SSL on both the external LDAP and the corresponding WebLogic Security Provider. SSL on the WebLogic Security Provider is enabled from the WebLogic Administration console.

For secure communication between WebLogic Server and Oracle Internet Directory, see *Oracle Fusion Middleware Securing Oracle WebLogic Server*.

3

Implementing Network Integrity Security

This chapter explains the security features of Oracle Communications Network Integrity.

Configuring and Using Authentication

Authentication is the mechanism by which users provide specific information as a proof of having access to a system. Authentication answers the question "Who are you?" using credentials such as user name and password.

In Oracle WebLogic Server, authentication providers are used to prove the identity of users or system processes. Authentication providers also remember, transport, and make identity information available to various components of a system when needed. During the authentication process, a principal validation provider provides additional security protection for the principals (users and groups) contained within the subject by signing and verifying the authenticity of those principals.

Network Integrity supports the following authentication providers:

- WebLogic-embedded lightweight directory access protocol (LDAP)
- External LDAP, such as Oracle Internet Directory
- Relational database management system (RDBMS)
- Security Assertion Markup Language (SAML)

Other security providers are supported using WebLogic Server Application server.

Oracle recommends Oracle Internet Directory as your authentication provider.

Network Integrity uses user name and password authentication. See *Network Integrity System Administrator's Guide* for more information.

Whether Network Integrity is configured to communicate with WebLogic Server over HTTP or HTTPS, login authentication is always sent over a secured HTTPS channel.

If you are using a web services interface, authentication details are supplied with each request using the User name token header. See *Network Integrity Developer's Guide* for more information.

Java Authentication and Authorization Service

WebLogic Server uses the Java Authentication and Authorization Service (JAAS) classes to authenticate to the client, whether the client is an application, applet, Enterprise JavaBean, or servlet that requires authentication.

JAAS implements a Java version of the Pluggable Authentication Module (PAM) framework, which permits applications to remain independent from underlying authentication technologies. Therefore, the PAM framework allows the use of new or updated authentication technologies without requiring modifications to the application.

About Callback Handlers

A callback handler is a flexible JAAS standard that allows a variable number of arguments to be passed as complex objects to a method.

There are three types of callback handlers: **NameCallback**, **PasswordCallback**, and **TextInputCallback**, all of which are part of the **javax.security.auth.callback** package. **NameCallback** and **PasswordCallback** return the user name and password, respectively. You can use **TextInputCallback** to access the data users enter into any additional fields on a login form (that is, fields other than those for obtaining the user name and password). When used, there should be one **TextInputCallback** per additional form field, and the prompt string of each **TextInputCallback** must match the field name in the form. WebLogic Server uses only the **TextInputCallback** for form-based web application login.

An application implements a callback handler and passes it to underlying security services so that they may interact with the application to retrieve specific authentication data, such as user names and passwords, or to display certain information, such as error and warning messages.

Callback handlers are implemented in an application-dependent fashion. For example, implementations for an application with a UI may pop up windows to prompt for requested information or to display error messages. An implementation may also choose to obtain requested information from an alternative source without asking the user.

Underlying security services make requests for different types of information by passing individual call backs to the callback handler. The callback handler implementation decides how to retrieve and display information depending on the call backs passed to it.

Configuring and Using Access Control

Authorization is used to control access by:

- Permitting only certain users to access a resource or action.
- Applying varying limitations on user access or actions.

Network Integrity has a single user role: **NetworkIntegrityRole**. Every user must have this role to access and perform any action in Network Integrity.

The **NetworkIntegrityRole** role grants full access to the Network Integrity UI, allowing users to manage all scans, view results, and correct discrepancies.

Users without the **NetworkIntegrityRole** role are prevented from logging in and are shown the following message:

```
You do not have any permission.
```

Configuring and Using Security Audit

Network Integrity, Oracle database, and Oracle Internet Directory allow you to record logs of key user actions to the WebLogic Server logs. Audit logs can be viewed using the Enterprise Manager or with a text editor.

Auditing provides an electronic trail of computer activity. In the WebLogic Server security architecture, an auditing provider is used to provide auditing services.

If WebLogic Security Framework configured, it calls through to an auditing provider before and after security operations (such as authentication or authorization) have been performed, when changes to the domain configuration are made, or when management operations on any resources in the domain are run. The decision to audit a particular event is made by the Auditing provider itself and can be based on specific audit criteria or severity levels. The records containing the audit information may be written to output repositories such as an LDAP server, database, and a file.

See the WebLogic Server documentation and the Oracle Database documentation for information about enabling audit logs.

The following examples show sample audit logs.

Example 3–1 Sample Audit Log for Logging Onto Network Integrity

```
[2011-09-14T10:41:47.684+00:00] [AdminServer] [NOTIFICATION] []
[oracle.communications.integrity.auditlog] [tid: [ACTIVE].ExecuteThread: '20' for
queue: 'weblogic.kernel.Default (self-tuning)'] [userId: user_ID] [ecid:
0000J9bXnYZ17iJLMmCCye1ES8Co000016,0] [APP: NetworkIntegrity] User Name: user_name
NI-Action: Login
```

Example 3–2 Sample Audit Log for Creating a Scan in Network Integrity

```
[2011-09-14T10:49:53.311+00:00] [AdminServer] [NOTIFICATION] []
[oracle.communications.integrity.auditlog] [tid: [ACTIVE].ExecuteThread: '19' for
queue: 'weblogic.kernel.Default (self-tuning)'] [userId: user_ID] [ecid:
0000J9bZe5617iJLMmCCye1ES8Co00002U,0] [APP: NetworkIntegrity] User Name: user_name
NI-Action: Create Scan 'UIM CISCO' starts
...
[2011-09-14T10:49:53.400+00:00] [AdminServer] [NOTIFICATION] []
[oracle.communications.integrity.auditlog] [tid: [ACTIVE].ExecuteThread: '19' for
queue: 'weblogic.kernel.Default (self-tuning)'] [userId: user_ID] [ecid:
0000J9bZe5617iJLMmCCye1ES8Co00002U,0] [APP: NetworkIntegrity] User Name: user_name
NI-Action: Create Scan 'UIM CISCO' ends
```

Example 3–3 Sample Audit Log for Running a Scan in Network Integrity

```
[2011-09-14T10:50:35.804+00:00] [AdminServer] [NOTIFICATION] []
[oracle.communications.integrity.auditlog] [tid: [ACTIVE].ExecuteThread: '20' for
queue: 'weblogic.kernel.Default (self-tuning)'] [userId: user_ID] [ecid:
0000J9bZneR17iJLMmCCye1ES8Co00002a,0] [APP: NetworkIntegrity] User Name: user_name
NI-Action: Scan for 'UIM CISCO'  with scanrun ID '627023' Starts
...
[2011-09-14T10:50:40.588+00:00] [AdminServer] [NOTIFICATION] []
[oracle.communications.integrity.auditlog] [tid: [ACTIVE].ExecuteThread: '10' for
queue: 'weblogic.kernel.Default (self-tuning)'] [userId: user_ID] [ecid:
0000J9bZpbz17iJLMmCCye1ES8Co00002e,0] [APP: NetworkIntegrity] NI-Action: Scan for
'UIM CISCO' with scanrun ID '627023' Completed
```

Example 3–4 Sample Audit Log for Logging Off of Network Integrity

```
[2011-09-14T10:41:37.970+00:00] [AdminServer] [NOTIFICATION] []
[oracle.communications.integrity.auditlog] [tid: [ACTIVE].ExecuteThread: '20' for
queue: 'weblogic.kernel.Default (self-tuning)'] [userId: user_ID] [ecid:
0000J9bXlAk17iJLMmCCye1ES8Co000012,0] [APP: NetworkIntegrity] User Name: user_name
NI-Action: Logout
```

Scan Parameter Security

All scan parameters are encrypted with the advanced encryption standard (AES) algorithm and are stored in the Oracle database.

Secure Access to Network Integrity Web Services

The web services API is standards based using JAX-WS over HTTPS. The Network Integrity web services API uses the same security access level as the Network Integrity UI. So any user able to log in to Network Integrity can also use the web Service API. This is assigned using the **NetworkIntegrityRole**.

Managing Network Integrity Security

Network Integrity System Administrator's Guide contains information on the following security management topics:

- Oracle Platform Security Services
- Security Realms
- Network Integrity User passwords
- Managing Users
- Encrypting Properties

4

Security Considerations for Developers

This chapter provides information for developers about how to create secure applications for Oracle Communications Network Integrity and how to extend Network Integrity without compromising its security.

About Network Integrity Security Policies

Network Integrity uses ADF security for its UI resources (JSDD or JSPX), and protects them with the **NetworkIntegrityRole** role. Users having this role can run, create, read, update, and delete operations on these pages. These policies can be customized in Oracle Fusion Middleware Enterprise Manager.

Secure Web Services Development

Web Services Security (WSS) is an extension to SOAP that is used to apply security to web services. WSS describes enhancements to SOAP messaging to provide quality of protection using features like message integrity, message confidentiality, and single message authentication. You can use these mechanisms to accommodate a wide variety of security models and encryption technologies.

Network Integrity uses policies- and roles-based WSS which means that only users who belong to the **NetworkIntegrityRole** role are allowed to access the Network Integrity web services.

Show here is an example of the Network Integrity policy and role allowed.

Policy

```
uri="policy:Wssp1.2-2007-Https-UsernameToken-Plain.xml", attachToWsdl=true
```

Roles Allowed

```
SecurityRole(role="NetworkIntegrityRole")
```

Note: SSL port should be enabled for all Network Integrity servers for Network Integrity WSS to work.

For secure communication between Network Integrity and Network Integrity web services, developers must connect to web services using SSL. The following example shows how to obtain all plug-in names using web services:

```
URL url = new URL("https", ipAddress, SSLPort,
"/NetworkIntegrityApp-NetworkIntegrityControlWebService-context-root/NetworkIntegr
ityControlServicePortType?wsdl");

NetworkIntegrityControlService sr = new NetworkIntegrityControlService(url,
"ipdpna21", "ipdp_Na21");

NetworkIntegrityControlServicePortType port =
sr.getNetworkIntegrityControlServicePortType();

ObjectFactory of = new ObjectFactory();

GetAllDisNetworkDiscoveryPluginRequestType req =
of.createGetAllDisNetworkDiscoveryPluginRequestType();

GetAllDisNetworkDiscoveryPluginResponseType res =
port.getAllDisNetworkDiscoveryPlugin(req);

  for(DisNetworkDiscoveryPluginType p : res.getDisNetworkDiscoveryPlugin()){
    System.out.println("Plugin Id:" + p.getEntityId());
    System.out.println("Plugin Name:" + p.getPluginName());
    }
  }
```

If the user is invoking web services from the Java client, the classpath must include the following parameter:

```
-Djavax.net.ssl.trustStore
```

Cartridge Development

Network Integrity provides an extensibility mechanism for receiving scan completion notification using a JMS-MDB client. Cartridge developers must follow the following example to securely retrieve the notification messages:

```
@MessageDestinationConfiguration(connectionFactoryJNDIName =
"oracle/communications/integrity/NIXATCF")
@RunAs("NetworkIntegrityRole")
public class ScanNotificationBean implements MessageListener
  {
    ...
  }
```

See *Network Integrity Developer's Guide* for more information about developing a JMS or MDB client to listens to event notification messages.

A

Network Integrity Secure Deployment Checklist

The following security checklist lists guidelines to help you secure Oracle Communications Network Integrity and its components.

Secure Deployment Checklist

- Install only the components you require.
- Lock and expire default user accounts.
- Enforce strong password management.
- Enable data dictionary protection on the Oracle Database for Network Integrity.
- Restrict, control, and revisit user privileges:
  - Grant only the necessary privileges to each user.
  - Revoke unnecessary privileges from the PUBLIC user group.
  - Restrict permissions on run-time facilities.
- Enforce the use of access controls.
- Require clients to authenticate.
- Restrict network access by doing the following:
  - Use firewalls.
  - Never leave an unnecessary hole in a firewall.
  - Password-protect the Oracle listener against remote access.
  - Monitor listener activity.
  - Monitor who accesses your systems.
  - Restrict system access by IP addresses.
  - Encrypt network traffic.
  - Harden the operating system by installing it in a secure location where it would be difficult for a hacker to access, by ensuring that all null passwords have been changed, and by disabling remote root login.
- Apply all security patches and workarounds.
- Encrypt sensitive information.

- Contact Oracle Security Products if you discover a vulnerability in any Oracle product.