

Oracle® AutoVue Client/Server Deployment
Security Guide
Release 21.0.0

November 2015

Copyright © 1999, 2015, Oracle and/or its affiliates. All rights reserved.

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this is software or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, the following notice is applicable:

U.S. GOVERNMENT RIGHTS Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, duplication, disclosure, modification, and adaptation shall be subject to the restrictions and license terms set forth in the applicable Government contract, and, to the extent applicable by the terms of the Government contract, the additional rights set forth in FAR 52.227-19, Commercial Computer Software License (December 2007). Oracle America, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

This software or hardware and documentation may provide access to or information on content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services.

Portions of this software Copyright 1996-2007 Glyph & Cog, LLC.

Contents

Preface	v
1 Overview of Oracle AutoVue Security	
1.1 Oracle AutoVue, Client/Server Deployment Overview	1-1
1.2 Security Objectives of Oracle AutoVue	1-1
1.2.1 Providing Basic Security Services.....	1-1
1.2.2 Ensuring Deployment and Configuration Flexibility	1-2
1.2.3 Ensuring Scalability and Predictability	1-2
1.3 General Security Principles	1-2
1.3.1 Keep Software Up to Date	1-2
1.3.2 Restrict Network Access to Oracle AutoVue.....	1-2
1.3.3 Keep Up to Date on Latest Security Information	1-2
1.3.4 Authentication.....	1-2
2 Determining Your Security Needs	
2.1 Security Architecture of Oracle AutoVue.....	2-1
2.1.1 User Authentication	2-1
2.1.1.1 DMS Authentication	2-2
2.1.1.2 Authentication Plug-In	2-2
2.1.1.3 Configuring Server to use JAAS Authentication Plug-In.....	2-2
2.1.2 Enabling SSL Communication	2-3
2.1.2.1 SSL Between the AutoVue Client and the VueServlet	2-3
2.1.2.2 SSL Between the VueServlet and the AutoVue Server	2-4
2.1.2.3 Exporting AutoVue Certificate.....	2-5
2.1.3 NTLM Authentication Protocol.....	2-5
2.2 Recommended Deployment Topologies	2-5
2.3 Clustered Deployments	2-7
2.4 VueServlet	2-7
2.5 Integrations with AutoVue.....	2-7
3 Secure Installation and Configuration	
3.1 Installation Overview	3-1
3.2 Installing the AutoVue Server.....	3-1
3.3 Deploying the VueServlet.....	3-2
3.4 Running the AutoVue Server as a Service.....	3-2

A Feedback

A.1	General AutoVue Information	A-1
A.2	Oracle Customer Support	A-1
A.3	My Oracle Support AutoVue Community	A-1
A.4	Sales Inquiries	A-1

Preface

This document provides guidelines on how to securely install and configure the AutoVue server and its associated components.

Audience

This document is intended for Oracle partners and third-party developers (such as integrators, and system administrators) whose task is to ensure the secure installation and configuration of the AutoVue server.

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc>.

Access to Oracle Support

Oracle customers have access to electronic support through My Oracle Support. For information, visit <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info> or visit <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs> if you are hearing impaired.

Related Documents

For more information, see the following documents:

- *Oracle AutoVue, Client/Server Deployment Installation and Configuration Guide*
- *Oracle AutoVue, Client/Server Deployment Planning Guide*

Conventions

The following text conventions are used in this document:

Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.
<i>italic</i>	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
monospace	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.

Overview of Oracle AutoVue Security

AutoVue is Oracle's suite of Enterprise Visualization solutions, which are designed to view, digitally annotate and collaborate on any digital information in an organization. AutoVue delivers visualization capabilities for many document types, including business documents such as Office and Graphics, as well as technical document types such as 2-D/3-D Computer Aided Design (CAD) and Electronic Design Automation (EDA).

This section provides an overview of Oracle AutoVue, Client/Server Deployment and discusses the security objectives and security architecture of AutoVue.

1.1 Oracle AutoVue, Client/Server Deployment Overview

The Client/Server deployment has AutoVue installed on a server, to which client machines connect to access and view documents. The Client/Server deployment provides a complete, open and standards-based set of integration tools that allows customers to tie AutoVue to any enterprise applications. This deployment provides users with a consistent view of data and business objects, and expands workflow automation to document-based processes.

1.2 Security Objectives of Oracle AutoVue

The security objectives of Oracle AutoVue are based on the operational environments and risk scenarios in which AutoVue may be deployed. The security objectives are:

- Providing Basic Security Services
- Ensuring Deployment and Configuration Flexibility
- Ensuring Scalability and Predictability

1.2.1 Providing Basic Security Services

Oracle AutoVue integrates with the following security services required in a multi-user, networked environment:

- Authentication

This service enables a system to verify the identity of users who request access to the AutoVue server.

- Authorization

Authorization ensures a system grants access to resources in compliance with the security policies defined for those resources. Access decisions are based on the authenticated identity and the privileges given to the requesting user.

- **Accountability**
Accountability ensures that users who access the system can be held accountable for their usage of the system and system resources. This enables you to monitor system usage to identify unauthorized users.
- **Data Protection**
This service prevents unauthorized users from accessing sensitive data. Use encryption to protect the confidentiality of data sent through a public network. Encryption can also be used to protect highly sensitive data from users who bypass access control mechanisms of a system.

1.2.2 Ensuring Deployment and Configuration Flexibility

Oracle AutoVue security services are designed to support the full range of AutoVue deployment scenarios. Security mechanisms in AutoVue are aimed at ensuring that practical, real-world constraints on deployment can be met. The constraints include the need to deploy certain components of AutoVue in the Demilitarized Zone (DMZ), to deploy it in the corporate intranet, and enable those components to communicate across a firewall.

1.2.3 Ensuring Scalability and Predictability

As systems grow in size, there will be a breaking point where a new server is required because of the overall workload or deployment requirements. You can scale your AutoVue deployment to meet your needs while ensuring a secure environment.

1.3 General Security Principles

This section describes fundamental principles to using Oracle AutoVue securely.

1.3.1 Keep Software Up to Date

It is good security practice to keep all software versions and patches up-to-date. Throughout this document, an AutoVue maintenance level of 21.0 is assumed. For updates on critical patches and other security alerts, refer to the [Oracle Critical Patch Updates, Security Alerts and Third Party Bulletin](#).

1.3.2 Restrict Network Access to Oracle AutoVue

Keep both the AutoVue server and any document repository behind a firewall. In addition, you may want to place a firewall between AutoVue servers when deployed as a server farm. Firewalls provide an assurance that access to these systems is restricted to a known network route which can be monitored and, if necessary, restricted.

1.3.3 Keep Up to Date on Latest Security Information

Oracle continually improves its software and documentation. Make sure to check the Oracle AutoVue Documentation Library on the [Oracle Technology Network \(OTN\)](#) for updates to this document.

1.3.4 Authentication

Whenever possible, use the authentication facilities of the install environment to verify the identity of a user that requests access to AutoVue.

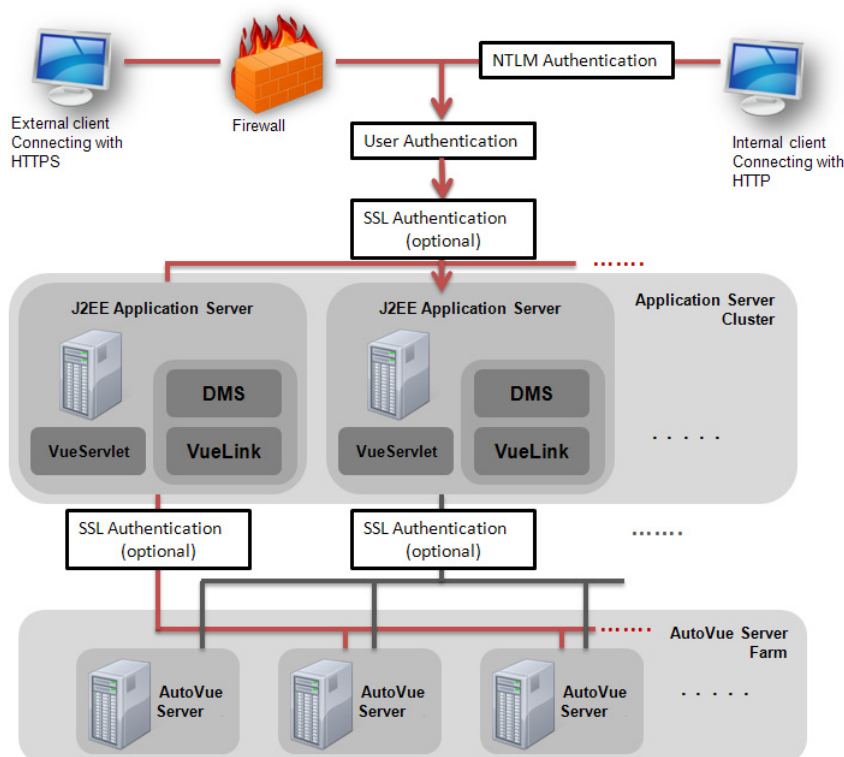
Determining Your Security Needs

Before deploying AutoVue, you must determine your security needs and make sure that you take the appropriate security measures.

2.1 Security Architecture of Oracle AutoVue

Figure 2–1, "Security Architecture of Oracle AutoVue" illustrates the elements of the Oracle AutoVue security architecture.

Figure 2–1 Security Architecture of Oracle AutoVue



2.1.1 User Authentication

A user authentication facility has been added between the client and the AutoVue server to allow integrators to connect AutoVue to Identity Management Systems.

There are two valid authentication mechanisms: a DMS authentication and an authentication plug-in (for example, Kerberos)

Note: If AutoVue is installed as a stand-alone server (that is, without an authentication plug-in or DMS authentication), then users cannot connect to the server. In this scenario, in `javueserver.properties`, the system administrator must set `javueserver.authentication.enable` to `FALSE` so that users can connect. Oracle recommends to prohibit all unauthenticated connections to the AutoVue server.

2.1.1.1 DMS Authentication

DMS authentication is implemented when AutoVue is integrated with a DMS backend system. A VueLink is expected to authenticate through a cookie session or through a session cookie or through a username/password prompt. Once the VueLink has authenticated the user, it returns the username to AutoVue in the `GetProperty Action` for `CSI_UserName`.

2.1.1.2 Authentication Plug-In

The implementation of the authentication mechanism (for example, Kerberos) makes use of a plug-in on the AutoVue client and another plug-in on the AutoVue server. The client uses its plug-in to obtain user credentials as part of the process of connecting to the server. The client encrypts the credentials and sends them to the server which uses its plug-in to authenticate the user who is trying to connect. If the server does not recognize the credentials, it refuses the connection.

A pair of these plug-ins are supplied with AutoVue: the `UsernamePasswordObtainer` class and the `JAASAuthenticator` class. The `UsernamePasswordObtainer` class is supplied so that the client can prompt the user for login information (username and password). The `JAASAuthenticator` class is supplied so that the server can use the Java Authentication and Authorization Service to authenticate using the authentication mechanisms specified in the configuration text file, `jaas_authen.conf`. The default version of this file is configured to authenticate using the Kerberos protocol which is supported by Windows Active Directory and many other standard identity repository solutions.

Note: If you do not select the Default installation option, user authentication between the AutoVue client and server can be configured by following the procedure provided in [Section 2.1.1.3, "Configuring Server to use JAAS Authentication Plug-In"](#).

Perform the procedure provided in [Section 2.1.1.3, "Configuring Server to use JAAS Authentication Plug-In"](#) only if you choose the **Configure Later** option when specifying the authentication mechanism between the AutoVue server and the client during AutoVue installation.

2.1.1.3 Configuring Server to use JAAS Authentication Plug-In

To configure the server to use the JAAS authentication plug-in supplied with AutoVue, perform the following:

1. Edit `javueserver.properties` to specify the plug-in by removing the comment in the following line:

```
javueserver.authenticator=com.cimmetry.javueserver.JAASAuthenticator
```

2. Create a text file called `jaas_authen.conf` in the `<AutoVue install root>\bin` directory. Add the following text in the file:

```
/**
** Example JAAS Login Configuration for the AutoVue server
** /
AVServer
{
com.sun.security.auth.module.Krb5LoginModule required storeKey=true;
};
```

3. Edit `javueserver.properties` and add the following highlighted lines after the `-Djava.security.policy` parameter of `javueserver.cmdline`:

```
javueserver.cmdline=-Xmx128M -
Djava.security.policy="C:\Oracle\AutoVue\bin\policy"
-Djava.security.krb5.realm=<realm> -Djava.security.krb5.kdc=<kdc>
-Djava.security.auth.login.config=<full path to jaas_authen.conf>
```

Replace `<realm>` with your security realm.

Replace `<kdc>` with your key distribution center.

4. Startup the AutoVue server.
5. Launch the AutoVue client.

An authentication dialog appears and prompts for login information. On logging in successfully, the AutoVue client launches.

2.1.2 Enabling SSL Communication

Secure Socket Layer (SSL) is an industry-standard protocol for securing network connections. It is recommended that all communication between the AutoVue client and the VueServlet and also between the VueServlet and the AutoVue server is encrypted.

2.1.2.1 SSL Between the AutoVue Client and the VueServlet

The VueServlet component is implemented as a standard Java servlet which executes within the context of an application or servlet engine. The application engine handles the communications configuration for all servlets and applications, including the provision of secure socket layer services. Secure sockets are implemented through the use of signed digital certificates and a secure handshaking procedure. Although the details of importing a digital certificate into an application server is implementation-specific, the basic process is described in the following steps.

In order to enable SSL between AutoVue client and the VueServlet, you must ensure that SSL is enabled for the application server and that you have a CA-issued certificate installed with your application server/Web server. For example, for the WebLogic application server. The certificate is configured with two keystores:

- *DemoIdentity.jks*: Contains a demonstration private key for WebLogic Server. This keystore contains the identity for WebLogic Server.
- *DemoTrust.jks*: Contains the trusted certificate authorities from the `WL_HOME\server\lib\DemoTrust.jks` and the JDK cacerts keystores. This keystore establishes trust for WebLogic Server.

In addition to enabling SSL and setting up the keystore/truststore for the application server, you must perform the following steps so that the AutoVue server can trust the application server's certificate:

Note: If you already have a copy of the .CER file from your certificate authority, and it is a Base64-encoded format, you can skip steps 1 through 3.

1. Connect to the application sever via HTTPS protocol in order get the application server's certificate.

For example: `https://<ApplicationServerHostName>`

2. Import the certificate into an AutoVue-certified Web browser.
3. Export the certificate from the Web browser as a base-64 encoded format and save the certificate onto the local disk. For example, `C:\certs.cer`
4. Import the certificate into the AutoVue server's JRE using Java's keytool command:

```
<Java Install Directory>\bin>keytool -import -alias <AutoVue server name> -file
c:\certs.cer -trustcacerts -v -keystore
C:\Oracle\AutoVue\jre\lib\security\cacerts
```

5. Restart the AutoVue server.
6. Configure the Web page that embeds the AutoVue applet to point to the `https://` URL for the VueServlet.

2.1.2.2 SSL Between the VueServlet and the AutoVue Server

Perform the following steps to enable SSL between the VueServlet and the AutoVue server.

1. In the web.xml descriptor file for the VueServlet, add the following init-param:

```
<init-param>
<param-name>EnableSSL</param-name>
<param-value>true</param-value>
</init-param>
```

2. Make the following modification to the AutoVue server's `javueserver.properties` file:

```
javueserver.ssl.enable=true
```

3. Set the following in `javueserver.cmdline` entry in `javueserver.properties`:

```
-Djavax.net.ssl.keyStore=<path to the keystore>
-Djavax.net.ssl.keyStorePassword=<password for the keystore>
```

4. Set the following in the application server startup file.

```
-Djavax.net.ssl.keyStore=<path to the keystore>
-Djavax.net.ssl.keyStorePassword=<password for the keystore>
```

Note: If you are using the application server's default/demo certificate, then you can skip Step 4 and enter the following in the startup file:

```
-Djavax.net.ssl.trustStore=<path to the truststore>
-Djavax.net.ssl.trustStorePassword=<password for the truststore>
```

5. Import the SSL certificate into the AutoVue server's JRE.

```
<Java Install Directory>\bin>keytool -import -alias <AutoVue server name> -file
```

```
c:\certs.cer -trustcacerts -v -keystore
C:\Oracle\AutoVue\jre\lib\security\cacerts
```

Note: In the event you are using the same certificate as the application server from section [Section 2.1.2.1, "SSL Between the AutoVue Client and the VueServlet,"](#) then you do not have to import the certificate. If you are using a certificate from Internet Explorer, you must export it as a base-64 encoded format and save the certificate onto the local disk. For example, C:\certs.cer

6. Import the SSL certificate into the application server's JRE.

```
<Java Install Directory>\bin>keytool -import -alias <AutoVue server name> -file
c:\certs.cer -trustcacerts -v -keystore <application
server>\jre\lib\security\cacerts
```

7. Restart the AutoVue server and the application server hosting the VueServlet.

SSL is now configured between the VueServlet and the AutoVue server.

2.1.2.3 Exporting AutoVue Certificate

When installing AutoVue with a self-signed certificate, use the following JAVA keytool command to export the certificate from the AutoVue server's JRE:

```
<Java Install Directory>\bin> keytool -exportcert -alias autovue_ssl -file
C:\AVCSert.cer -keystore C:\Oracle\AutoVue\jre\lib\security\cacerts
```

The AutoVue certificate can now be imported to other trust stores.

2.1.3 NTLM Authentication Protocol

If you are loading files that require you to enter authentication, AutoVue prompts for authentication. When using NT LAN Manager (NTLM) authentication, you must set the `javueserver.ntlm.enable` parameter to TRUE in `javueserver.properties`.

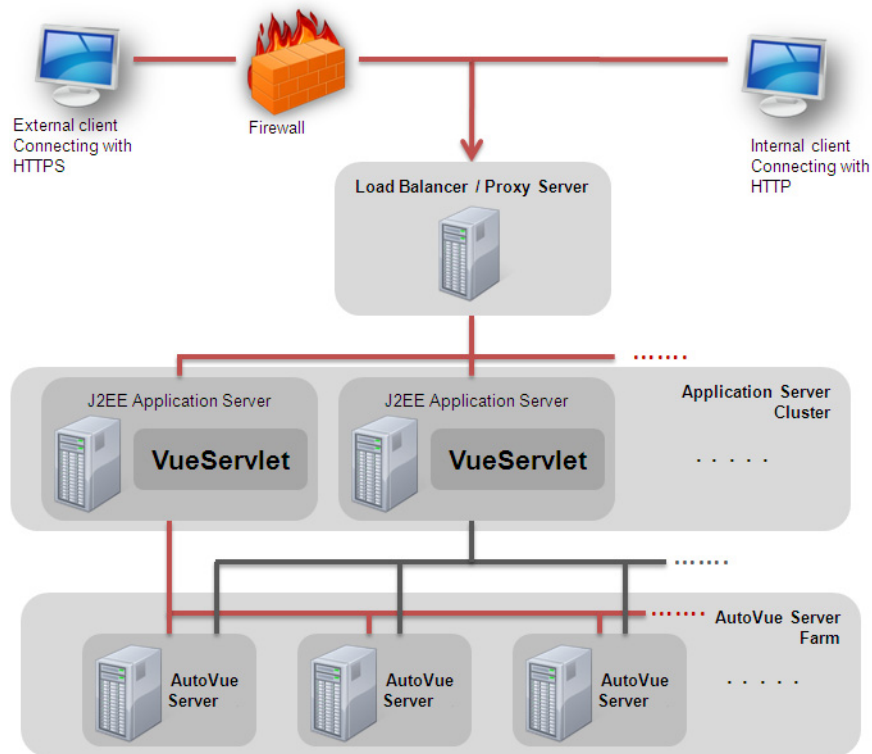
This change requires you to restart the AutoVue server for the change to take effect.

2.2 Recommended Deployment Topologies

The section describes standard architectures for deploying Oracle AutoVue to secure internet access. For a more complete discussion of AutoVue deployment architectures, refer to the *Oracle AutoVue, Client/Server Deployment Planning Guide*.

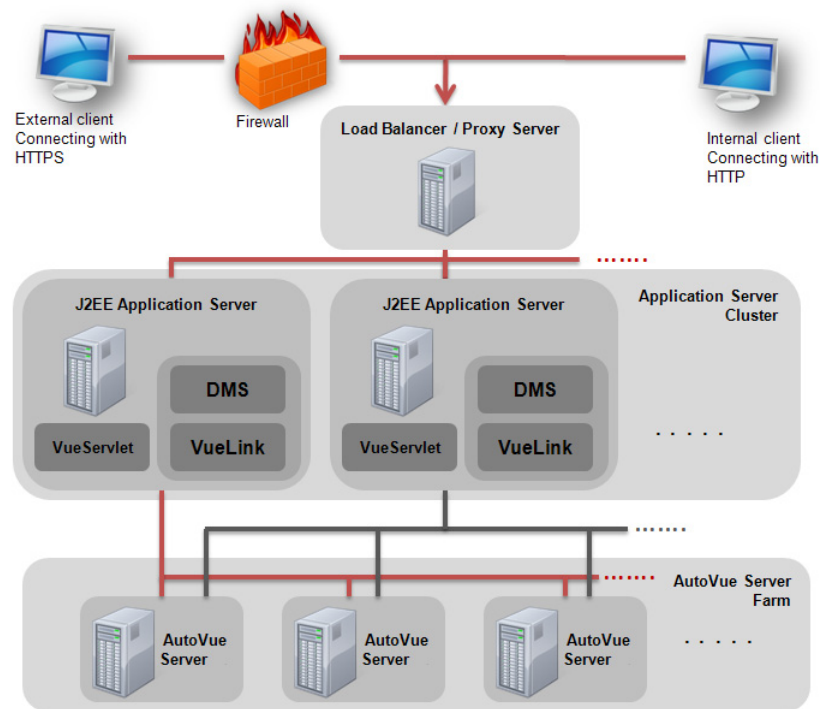
[Figure 2-2, "Deployment Architecture - Standalone"](#) illustrates the most basic deployment architecture. This standalone/non-integrated deployment provides failover as the AutoVue servers are setup in a server farm.

Figure 2–2 Deployment Architecture - Standalone



Note: All servers in the server farm must set the same RMI host (jvueserver.rmi.host.*) for each server in the farm. For example, if a new server is added to an existing server farm without updating the jvueserver.properties file for each server in the farm, then the new server will not be able to connect to the farm. If this happens, a security warning is logged in the server logs.

Another standard deployment architecture is AutoVue with a content repository integration as shown in the [Figure 2–3, "Deployment Architecture - Content Repository Integration"](#). This deployment architecture also provides failover for the AutoVue servers.

Figure 2-3 Deployment Architecture - Content Repository Integration

2.3 Clustered Deployments

The `DMS_PRESERVE_COOKIES` applet parameter is used when AutoVue is deployed in a clustered environment or when an integration with AutoVue relies on setting cookies and having the client pass them back as part of the request.

In AutoVue version 20.0, the `DMS_PRESERVE_COOKIES` applet parameter was updated to allow integrators to specify the exact list of cookies that the AutoVue client should pass on to the AutoVue server and/or the integration/VueLink servlet. If an AutoVue installation is being used in an integrated environment, Oracle recommends that you make use of this enhancement to restrict the use of cookies that are necessary for your deployment/integration to work.

2.4 VueServlet

Oracle recommends that the `ServerInfo` parameter of the `VueServlet` is set to `FALSE`. When set to `TRUE`, the server IP address is displayed if a user accesses the `VueServlet` page. The default value for this parameter is `FALSE`.

You can configure secure socket connection between the `VueServlet` and the AutoVue server. Refer to section [Section 2.1.2, "Enabling SSL Communication"](#) for information on how to set the `EnableSSL` parameter for the `VueServlet`.

2.5 Integrations with AutoVue

If you are developing your own integration between the AutoVue server and a document repository (typically, through the use of the ISDK that ships with AutoVue), the following points should be considered to enhance the system's overall security:

- Ensure that the original URL to a file does not contain sensitive information such as user or server information. Setting sensitive information in URLs is a potential security risk since URLs may be accessed by other users.
- The DMSARGS parameter should not contain a Session ID or other session or user-sensitive information. It is recommended to use cookies for Session ID. Additionally, integrations should use the CSI_UserName to query the user name instead of passing the username through DMSARGS.
- Pass user information through the CSI_UserName property that is queried by the AutoVue server at the beginning of a session. Ensure that you follow a consistent approach to passing user information.

Note: For AutoVue integrations, save the username in the HTTP session on the application server and return it using the GetProperty Action for CSI_UserName.

- Always ensure that you pass valid user information to AutoVue. Handle incorrect authentication properly and enforce encryption of sensitive information.
- Security enhancements are added to AutoVue on an ongoing basis. The ISDK for AutoVue 20.x also leverages these security enhancements. Integrators are encouraged to upgrade their integrations to use the latest AutoVue and ISDK to benefit from these enhancements.

Secure Installation and Configuration

This chapter describes the secure installation and configuration steps of the AutoVue server.

3.1 Installation Overview

Oracle recommends that the AutoVue server is run as an unprivileged named user to ensure that direct access to the server and files on the server is restricted. Users connecting to the AutoVue server through the client can still view files and generate streaming files.

All the components included in the AutoVue deployment should be installed in a secure manner. The following sections cover the following steps:

- Installing the AutoVue Server
- Deploying the VueServlet
- Running the AutoVue Server as a Service

When AutoVue is installed as a component to an existing application, the security requirements of the application must be applied to AutoVue. Take note of the following recommendations when security measures should be applied:

- Use the HTTPS protocol when accessing the VueServlet from the AutoVue client.
For Example: `https://<AutoVue server hostname>:8443/servlet/VueServlet`
- Use SSL communication between the VueServlet and AutoVue Server. For more information, refer to [Section 2.1.2, "Enabling SSL Communication."](#)
- Use the HTTPS protocol between the AutoVue server and Oracle VueLink.

This section discusses security considerations when installing the AutoVue server and its components. For complete installation instructions, refer to the *Oracle AutoVue Client/Server Deployment Installation and Configuration Guide*.

3.2 Installing the AutoVue Server

By default, the AutoVue installer provides a secure installation of the AutoVue server. That is, only essential AutoVue features are installed. For complete instructions on installing AutoVue, refer to the "Installing AutoVue" section of the *Oracle AutoVue Client/Server Deployment Installation and Configuration Guide*.

3.3 Deploying the VueServlet

The VueServlet should be deployed on a secure installation of WebLogic. For information on deploying VueServlet, refer to the *Oracle AutoVue Client/Server Deployment Installation and Configuration Guide*. Refer to WebLogic documentation for more information on secure installations. If you are using an application server other than WebLogic, please refer to its respective security guidelines.

3.4 Running the AutoVue Server as a Service

When running the AutoVue server as a service on either Windows or Linux operating systems, it is recommended that you run it as a named user and not as Local System Account as the local system account has more privileges than a named account. For more information on running the AutoVue server as service, refer to the "Running the AutoVue Server as a Service" section of the *Oracle AutoVue Client/Server Deployment Installation and Configuration Guide*.

If you have any questions or require support for AutoVue, please contact your system administrator. If the administrator is unable to resolve your issue, please contact us using the links below.

A.1 General AutoVue Information

Web Site <http://www.oracle.com/us/products/applications/autovue/index.html>

Blog <http://blogs.oracle.com/enterprisevisualization/>

A.2 Oracle Customer Support

Web Site <http://www.oracle.com/support/index.html>

A.3 My Oracle Support AutoVue Community

Web Site <https://communities.oracle.com/portal/server.pt>

A.4 Sales Inquiries

E-mail autovuesales_ww@oracle.com
