

Novell NetWare® 6.5

www.novell.com

February 28, 2005

TOMCAT ADMINISTRATION GUIDE



Novell®

Legal Notices

Novell, Inc. makes no representations or warranties with respect to the contents or use of this documentation, and specifically disclaims any express or implied warranties of merchantability or fitness for any particular purpose. Further, Novell, Inc. reserves the right to revise this publication and to make changes to its content, at any time, without obligation to notify any person or entity of such revisions or changes.

Further, Novell, Inc. makes no representations or warranties with respect to any software, and specifically disclaims any express or implied warranties of merchantability or fitness for any particular purpose. Further, Novell, Inc. reserves the right to make changes to any and all parts of Novell software, at any time, without any obligation to notify any person or entity of such changes.

You may not use, export, or re-export this product in violation of any applicable laws or regulations including, without limitation, U.S. export regulations or the laws of the country in which you reside.

Copyright © 2001-2005 Novell, Inc. All rights reserved. No part of this publication may be reproduced, photocopied, stored on a retrieval system, or transmitted without the express written consent of the publisher.

U.S. Patent No. 5,157,663; 5,349,642; 5,455,932; 5,553,139; 5,553,143; 5,572,528; 5,594,863; 5,608,903; 5,633,931; 5,652,854; 5,671,414; 5,677,851; 5,692,129; 5,701,459; 5,717,912; 5,758,069; 5,758,344; 5,781,724; 5,781,733; 5,784,560; 5,787,439; 5,818,936; 5,828,882; 5,832,274; 5,832,275; 5,832,483; 5,832,487; 5,859,978; 5,870,561; 5,870,739; 5,873,079; 5,878,415; 5,884,304; 5,893,118; 5,903,650; 5,903,720; 5,905,860; 5,910,803; 5,913,025; 5,913,209; 5,915,253; 5,925,108; 5,933,503; 5,933,826; 5,946,002; 5,946,467; 5,956,718; 5,956,745; 5,964,872; 5,974,474; 5,983,223; 5,983,234; 5,987,471; 5,991,810; 6,002,398; 6,014,667; 6,016,499; 6,023,586; 6,029,247; 6,052,724; 6,061,726; 6,061,740; 6,061,743; 6,065,017; 6,081,774; 6,081,814; 6,094,672; 6,098,090; 6,105,062; 6,105,069; 6,105,132; 6,115,039; 6,119,122; 6,144,959; 6,151,688; 6,157,925; 6,167,393; 6,173,289; 6,216,123; 6,219,652; 6,233,859; 6,247,149; 6,269,391; 6,286,010; 6,308,181; 6,314,520; 6,324,670; 6,338,112; 6,345,266; 6,353,898; 6,424,976; 6,466,944; 6,477,583; 6,477,648; 6,484,186; 6,496,865; 6,510,450; 6,516,325; 6,519,610; 6,532,451; 6,532,491; 6,539,381; RE37,178. Patents Pending.

Novell, Inc.
404 Wyman Street, Suite 500
Waltham, MA 02451
U.S.A.

www.novell.com

Tomcat Administration Guide for NetWare 6.5

February 28, 2005

Online Documentation: To access the online documentation for this and other Novell products, and to get updates, see www.novell.com/documentation.

Novell Trademarks

eDirectory is a trademark of Novell, Inc.

exteNd is a trademark of Novell, Inc.

exteNd Composer is a trademark of Novell, Inc.

exteNd Director is a trademark of Novell, Inc.

exteNd Workbench is a trademark of Novell, Inc.

NetWare is a registered trademark of Novell, Inc. in the United States and other countries.

NetWare Loadable Module and NLM are trademarks of Novell, Inc.

Novell is a registered trademark of Novell, Inc. in the United States and other countries.

Novell Cluster Services is a trademark of Novell, Inc.

Novell iFolder is a registered trademark of Novell, Inc.

Third-Party Trademarks

All third-party trademarks are the property of their respective owners.

Contents

- About This Guide** **7**

- 1 Tomcat Servlet Container Overview** **9**
 - Benefits of Running Tomcat on NetWare 9
 - Administration Instance vs. Public Instance of Tomcat on NetWare 10
 - About the Administration Instance of Tomcat 10
 - About the Public Instance of Tomcat 11
 - Tomcat and the Novell exteNd Application Server 12
 - Managing Tomcat 12
 - Managing Web Applications and Servlets 13
 - What's Next. 13

- 2 Tomcat Installation and Configuration** **15**
 - Deciding How to Install Tomcat on NetWare 15
 - Installing the Administration Instance of Tomcat 15
 - Starting and Stopping Tomcat from the NetWare System Console 16
 - Managing Tomcat with Tomcat Admin 16
 - What's Next. 16

- 3 Deploying Servlets and JSPs** **17**
 - About JSPs 17
 - Deploying Servlets 18
 - About Tomcat Manager. 18
 - What's Next. 20

- 4 Securing Tomcat on NetWare** **21**
 - Authentication and Authorization 21
 - OU Container Rights 21
 - Required Settings 22
 - Relocating the Tomcat-Roles Container 22
 - Using Dynamic Groups 22
 - What's Next. 23

About This Guide

This guide introduces you to the Tomcat Servlet Container 4.1 as it is used on NetWare® 6.5. It introduces you to the two management tools used to manage Tomcat and to deploy servlets, JavaServer* Pages (JSPs), and Web applications.

This guide is intended for administrators whose role is to install, configure, and manage Tomcat for NetWare. Web application developers will also find the information useful.

This guide is divided into the following sections:

- ♦ Chapter 1, “Tomcat Servlet Container Overview,” on page 9
- ♦ Chapter 2, “Tomcat Installation and Configuration,” on page 15
- ♦ Chapter 3, “Deploying Servlets and JSPs,” on page 17
- ♦ Chapter 4, “Securing Tomcat on NetWare,” on page 21

Additional Documentation

For complete Tomcat documentation, visit the [Jakarta Project Web site \(http://jakarta.apache.org/tomcat/tomcat-4.1-doc/index.html\)](http://jakarta.apache.org/tomcat/tomcat-4.1-doc/index.html).

Documentation Conventions

In this documentation, a greater-than symbol (>) is used to separate actions within a step and items within a cross-reference path.

A trademark symbol (®, ™, etc.) denotes a Novell® trademark. An asterisk (*) denotes a third-party trademark.

When a single pathname can be written with a backslash for some platforms or a forward slash for other platforms, the pathname is presented with a backslash. Users of platforms that require a forward slash, such as Linux* or UNIX*, should use forward slashes as required by your software.

User Comments

We want to hear your comments and suggestions about this guide and the other documentation included with NetWare. Please use the User Comment feature at the bottom of each page of the NetWare online documentation.

1

Tomcat Servlet Container Overview

Tomcat is a *servlet container*, which is a runtime shell that manages and invokes servlets when they are requested by a Web browser or by another servlet. Servlets are programs that run on a Web server and automatically generate Web pages as a result of user input. Two or more servlets working together to provide a common set of functions is referred to as a Web application.

Web servers, such as the Apache Web server, also included with NetWare[®] 6.5, depend on a servlet container like Tomcat to process JavaServer Pages (JSPs) and servlets.

Tomcat is developed under a larger project named Jakarta. Jakarta is a collaboration between Apache developers, major corporate sponsors, smaller companies, and individual private developers. Like the Apache project, Jakarta is an open source project. *Open source* means that the source code is freely available.

Tomcat is the reference implementation from Sun* Microsystems*. It implements the servlet specification 2.3 and the JSP specification 1.2.

For more information about the Jakarta Project, visit <http://jakarta.apache.org/tomcat/index.html> (<http://jakarta.apache.org/tomcat/index.html>).

This overview includes the following topics:

- ◆ “Benefits of Running Tomcat on NetWare” on page 9
- ◆ “Administration Instance vs. Public Instance of Tomcat on NetWare” on page 10
- ◆ “Tomcat and the Novell exteNd Application Server” on page 12
- ◆ “Managing Tomcat” on page 12
- ◆ “Managing Web Applications and Servlets” on page 13
- ◆ “What’s Next” on page 13

Benefits of Running Tomcat on NetWare

Tomcat provides many business benefits to your existing network that can ultimately increase productivity, improve communication between departments and employees. When used in conjunction with the Apache Web server, Tomcat can host powerful Web applications.

Here are some of the key uses and benefits of using Tomcat on NetWare:

- ◆ Offers a highly flexible, robust JSP servlet container that is tightly integrated with NetWare.
- ◆ Provides a simple entry point for organizations planning to prototype and deploy Java* based utilities and solutions on a NetWare server.
- ◆ Works with major development tools available through commercial vendors and open source communities.

- ◆ Tomcat can be deployed with Novell® Cluster Services™ (included with NetWare) to provide high availability, load balancing, and fault tolerance for important business processes running in the Tomcat JSP servlet container.

Administration Instance vs. Public Instance of Tomcat on NetWare

As with the Apache Web server, two instances of Tomcat are configured on your server: an administration instance and a public instance. By default, Tomcat is installed to `sys:\tomcat`. Unlike Apache, which also includes administration and public versions, each installed into their own directories, both instances of Tomcat share the same Tomcat directory. Adding additional instances of Tomcat is a matter of modifying the `autoexec.ncf` file.

If during the NetWare installation you didn't choose to install Tomcat, only the administration instance is installed.

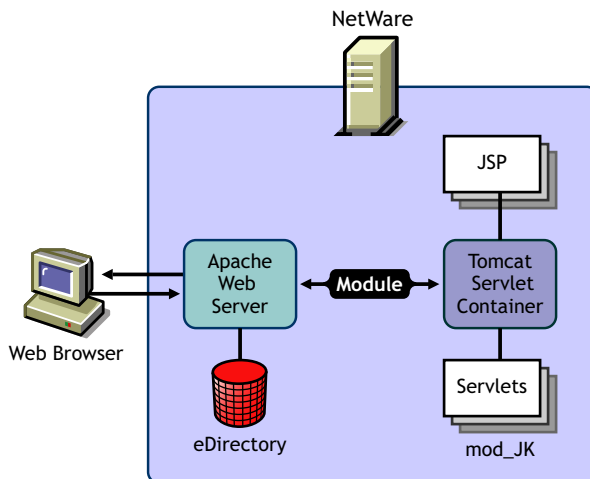
- ◆ “About the Administration Instance of Tomcat” on page 10
- ◆ “About the Public Instance of Tomcat” on page 11

TIP: If you are a developer and you need to add additional instances of Tomcat, see the [Novell Developer Kit \(http://developer.novell.com/ndk\)](http://developer.novell.com/ndk).

About the Administration Instance of Tomcat

The administration instance of Tomcat is used by NetWare to host and manage Novell services such as Apache Manager and the NetWare Welcome Web site. When requests come to the Apache Web server from any of the NetWare products, Apache identifies requests that require the use of Tomcat. These requests are passed through the `mod_jk` module on Apache to Tomcat, as shown in the following diagram.

Figure 1 How Apache and Tomcat communicate to serve up dynamic Web content from Java servlets and JSPs.



The `mod_jk` plug-in opens a socket to an instance of Tomcat and communicates with it using the AJP 13 protocol. Tomcat processes the HTTP request and then returns an HTTP response, which is sent back through the same socket connection to Apache. Apache then serves the response up to the client that requested it.

The following table lists key files used by the administration instance of Tomcat and how they are used.

Key Files	Purpose
<code>sys:/adminsrv/conf/admin_tomcat.xml</code>	Configuration file for the administration instance of Tomcat.
<code>sys:/system/tcadmup.ncf</code>	Use at the system console to start the administration instance of Tomcat.
<code>sys:/system/tcadmdn.ncf</code>	Use at the system console to stop the administration instance of Tomcat.
<code>sys:/adminsrv/conf/mod_jk/workers.properties</code>	Configuration file for the Apache/Tomcat interconnection. NOTE: The ports specified in this file must match the ports specified in the <code>server.xml</code> and <code>admin_tomcat.xml</code> files. Also, worker names must correspond with those used in JkMount statements found in the <code>httpd.conf</code> and <code>adminsrv.conf</code> files.

About the Public Instance of Tomcat

The second, or public, instance of Tomcat functions as your own dedicated servlet container, which you can optimize and use to host your own Web applications.

The following table lists key files used by the public instance of Tomcat and how they are used.

Key Files	Purpose
<code>sys:/tomcat/4/bin/tomcat4.ncf</code>	Use at the system console to start the public instance of Tomcat by entering <code>tomcat4</code> . You can also use <code>tomcat4 stop</code> to stop the server.
<code>sys:/tomcat/4/bin/tc4stop.ncf</code>	Use at the system console to stop the public instance of Tomcat by entering <code>tc4stop</code> .
<code>sys:/tomcat/4/bin/catalina.pl</code>	Primarily used as the start and stop script for Tomcat. It is also used for other purposes, such as starting JPDA mode.
<code>sys:/tomcat/4/bin/startup.ncf</code>	Used during startup to manage NetWare-specific command line parameters. It is used by <code>tomcat4.ncf</code> and <code>tcadmup</code> .
<code>sys:/tomcat/4/conf/server.xml</code>	Configuration file that contains port number specifications. Ports here must match those found in the <code>workers.properties</code> file.
<code>sys:/adminsrv/conf/mod_jk/workers.properties</code>	Same file that is used by the administrative instance of Tomcat. See sys:/adminsrv/conf/mod_jk/workers.properties above.

After reviewing this guide, see the [Tomcat documentation \(http://jakarta.apache.org/tomcat/tomcat-4.1-doc/index.html\)](http://jakarta.apache.org/tomcat/tomcat-4.1-doc/index.html) on the Jakarta Project Web site.

Tomcat and the Novell exteNd Application Server

When you need greater processing power beyond what basic HTML and scripting has to offer, NetWare offers two choices: Tomcat and Novell exteNd™ Application Server. The one you choose depends on what you need.

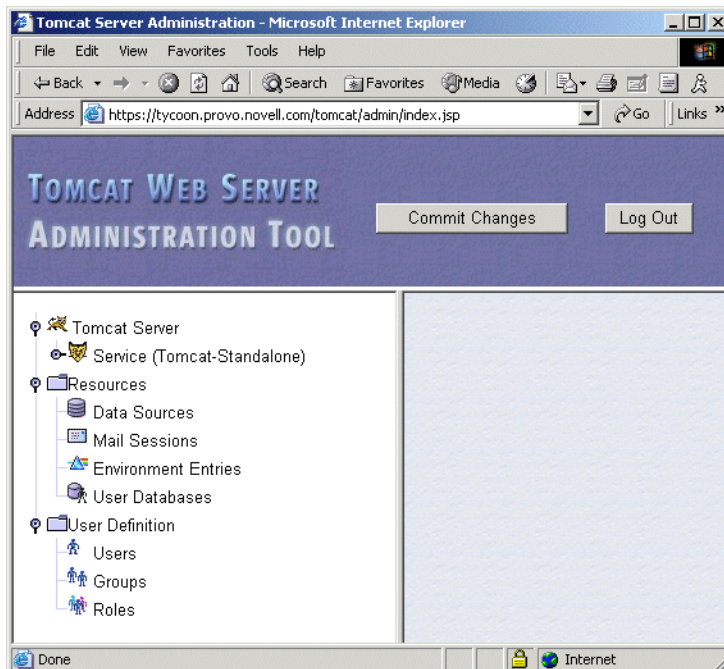
For example, if you need Java API support beyond servlets, JSPs, tag libraries, or basic JavaBeans* components, either immediately or in the future, you should select the Novell exteNd Application Server. Also, if you are going to use any of the high-end development tools included with NetWare, such as exteNd Director™ and exteNd Composer™, exteNd is the better choice because Tomcat is not supported by these tools. (For more information about Novell exteNd, see the [exteNd documentation \(http://www.novell.com/documentation-index/index.jsp?category=exteNd\)](http://www.novell.com/documentation-index/index.jsp?category=exteNd).)

However, if you need only very basic Java servlets and JSPs and you do not plan to migrate to a more robust solution, you should select Tomcat. Tomcat is also the better choice if you are not experienced with Java programming.

Managing Tomcat

The Tomcat Administration Tool, referred to as Tomcat Admin, lets you configure Tomcat from a Web browser. Changes made to Tomcat using Tomcat Admin can be saved persistently so that the changes remain when Tomcat is restarted, or they can be applied to the current session only.

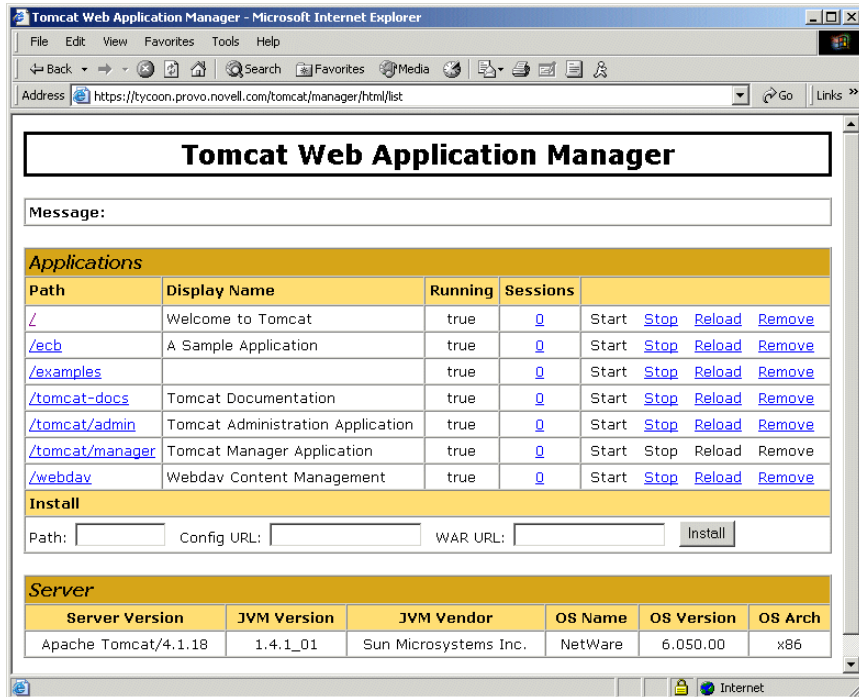
Tomcat Admin is a Web-based administration tool created by the Apache Foundation and is included and fully configured to run on NetWare. It can be accessed from any Web browser that has access to your server.



For more information, see [“Managing Tomcat with Tomcat Admin”](#) on page 16.

Managing Web Applications and Servlets

For deploying and managing Web servlets and applications, you can use Tomcat Manager, also a Web-based administration utility.



When you install Tomcat with NetWare, Tomcat Manager is automatically installed and configured.

For more information about Tomcat Manager on NetWare, see [“About Tomcat Manager” on page 18](#).

What’s Next

For information about installing and using Tomcat, see [Chapter 2, “Tomcat Installation and Configuration,” on page 15](#).

2

Tomcat Installation and Configuration

The administration instance of Tomcat is installed automatically during the NetWare® 6.5 installation process. It is a required component for several key NetWare products. However, if you want to host your own servlets and JavaServer Pages (JSPs), you can install a public instance of Tomcat during the NetWare installation process, or after NetWare is already installed and running.

Once it is installed, you can use Tomcat Admin, a browser-based administration utility, to manage Tomcat.

This section includes the following topics:

- ◆ [“Deciding How to Install Tomcat on NetWare” on page 15](#)
- ◆ [“Starting and Stopping Tomcat from the NetWare System Console” on page 16](#)
- ◆ [“Managing Tomcat with Tomcat Admin” on page 16](#)
- ◆ [“What’s Next” on page 16](#)

Deciding How to Install Tomcat on NetWare

You can install the public instance of Tomcat during or after the NetWare installation. (The administration instance of Tomcat is installed by default.) During the installation process, you can include Tomcat in your NetWare installation in two different configurations parallel to the way Apache is installed: a preconfigured server (the Apache/Tomcat Server pattern) and an independent component (the Tomcat 4 Servlet Container component). For an overview of the installation process, see [“Deciding How to Install Apache on NetWare”](#) in the *Apache Web Server Administration Guide for NetWare 6.5*.

After the NetWare installation has been completed, you can run the NetWare post-install program from the NetWare GUI and select the Tomcat 4 Servlet Container component.

After you have decided which installation option to use, see the *NetWare 6.5 Overview and Installation Guide* for detailed installation instructions.

Installing the Administration Instance of Tomcat

As mentioned above, the administration instance of the Tomcat servlet container is installed by default during the NetWare installation. However, if for any reason you need to install it after you have installed NetWare, you can do so by running the NetWare post-installation program from the NetWare GUI or the NetWare Deployment Manager and then selecting Tomcat 4 Admin Instance.

For detailed information about installing products after installing NetWare, see *Installing Products and Updates* in the *NetWare 6.5 Overview and Installation Guide*.

Starting and Stopping Tomcat from the NetWare System Console

To start Tomcat, enter `tomcat4` at the system console. To stop Tomcat, enter `tcadmdn` at the system console.

Managing Tomcat with Tomcat Admin

After Tomcat is running, you can access the Tomcat Admin utility from a Web browser to make configuration changes to Tomcat.

- 1 Make sure tomcat is running by entering `tomcat4` at the NetWare system console.
- 2 Start a Web browser.
- 3 In the Web browser, point to the following URL:

```
https://domain_name/tomcat/admin/
```

Replace *domain_name* with either your server's domain name or IP address.

IMPORTANT: You must include the trailing slash mark (/) at the end of the URL. If you do not include it, the server returns a 404 Not Found error.

- 4 Enter the administrator username and password created when you installed NetWare.

After Tomcat Admin starts, refer to the [Tomcat Administration Tool \(http://java.sun.com/webservices/docs/1.0/tutorial/doc/Admintool.html\)](http://java.sun.com/webservices/docs/1.0/tutorial/doc/Admintool.html) on Sun's Web site.

What's Next

After Tomcat is running and you can access both Tomcat Admin and Tomcat Manager, you're ready to deploy Web applications. See [Chapter 3, "Deploying Servlets and JSPs,"](#) on page 17.

For information about making Tomcat secure, see [Chapter 4, "Securing Tomcat on NetWare,"](#) on page 21.

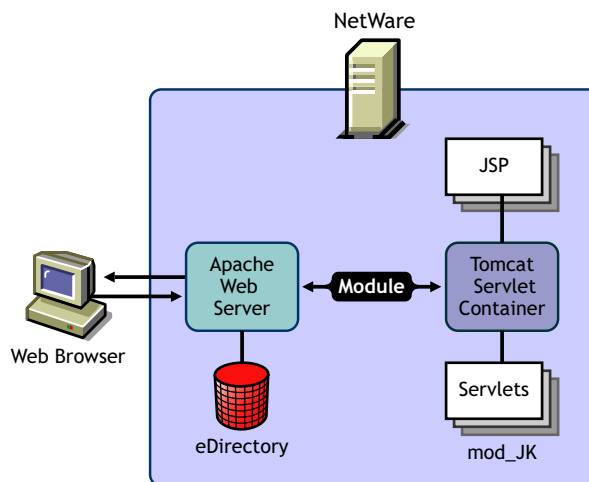
3

Deploying Servlets and JSPs

Servlets are programs that run on a Web server and automatically generate Web pages as a result of user input.

This section describes the methods and tools you can use to deploy servlets and JavaServer Pages (JSPs) on Tomcat Servlet Container 4.1 on NetWare[®] 6.5. It also introduces you to Tomcat Manager, a browser-based tool you can use to deploy your servlets.

Figure 2 Basic architecture of Apache working together with Tomcat to host JSPs And servlets.



This section contains the following topics:

- ◆ [“About JSPs” on page 17](#)
- ◆ [“Deploying Servlets” on page 18](#)
- ◆ [“About Tomcat Manager” on page 18](#)
- ◆ [“What’s Next” on page 20](#)

About JSPs

JSP technology offers a simple way to create dynamic Web pages that are both platform-independent and server-independent, giving you more freedom because it separates content generation from presentation and takes advantage of reusable tags and objects, simplifying the maintenance of your Web applications. JSP technology provides the scripting ability you need to create simple interactive Web pages, or it scales to support complex Web sites that are fully integrated with enterprise class applications.

Deploying Servlets

On Tomcat 4.1, Web applications are deployed in one of the following ways:

- ◆ By placing a WAR file in the `sys:/tomcat/4/webapps` directory (Recommended)
- ◆ By placing an exploded WAR in the `sys:/tomcat/4/webapps` directory
- ◆ By placing an exploded WAR anywhere on disk and then creating an XML file in `sys:/tomcat/4/webapps`
- ◆ By using the first and second methods above with an XML file that specifies the settings for your application

These are standard methods across all platforms for deploying applications using Tomcat 4.x.

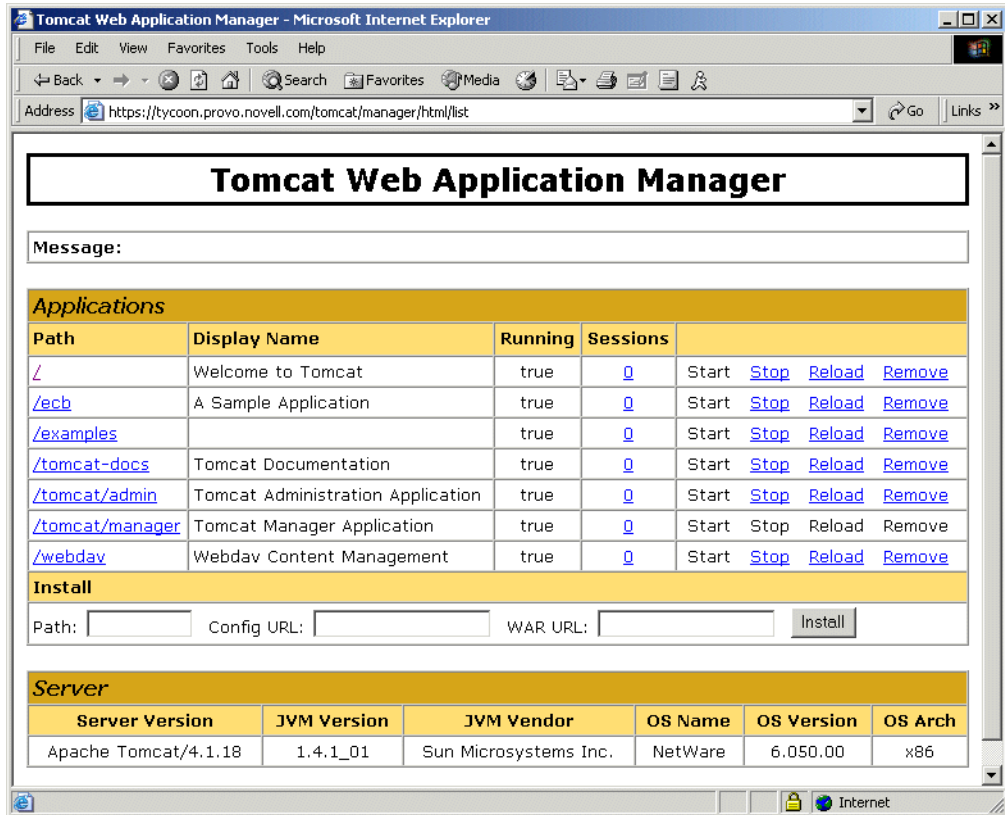
To deploy servlets and JSPs, use Tomcat Manager and see the [Tomcat Web Application Manager How To \(http://jakarta.apache.org/tomcat/tomcat-4.1-doc/html-manager-howto.html\)](http://jakarta.apache.org/tomcat/tomcat-4.1-doc/html-manager-howto.html) documentation on the Apache Web site.

NOTE: If you are deploying administration applications on the Admin version of Tomcat (`sys:/adminsrv/webapps`), use the same methods outlined above, except that you should place XML in the `sys:/adminsrv/webapps` directory rather than in the `sys:/tomcat/4/webapps` directory.

About Tomcat Manager

Tomcat Manager is a browser-based administration tool you can use for installing and deploying Web applications.

TIP: *Web application* is a term used to describe two or more servlets and JSPs, or a combination of both, working together to provide a set of end-user functions or services. However, Web application is also often used synonymously with servlet.



Tomcat Manager lets you do the following things:

- ◆ Install a new Web application, which can be anywhere on the server's disks
- ◆ Deploy a new Web application on a specified context path from the uploaded contents of a WAR file
- ◆ List the currently deployed Web applications, as well as the sessions that are currently active for them
- ◆ Reload an existing Web application to reflect changes in the contents of /WEB-INF/classes or /WEB-INF/lib
- ◆ List operating system and Java Virtual Machine (JVM*) property values
- ◆ List available global JNDI resources for use in deployment tools that are preparing <ResourceLink> elements nested in a <Context> deployment description
- ◆ List available security roles defined in the user database
- ◆ Remove an installed Web application
- ◆ Restart a Web application that has, for any reason, stopped running
- ◆ Stop an existing Web application, making it unavailable to users, without having to undeploy it
- ◆ Undeploy a deployed Web application and delete its document base directory

You can access Tomcat Manager, the browser-based servlet and application management tool, from any Web browser that has access to your NetWare server.

To access Tomcat Manager:

- 1 Open a Web browser and enter the URL to Tomcat Manager, using the following syntax:

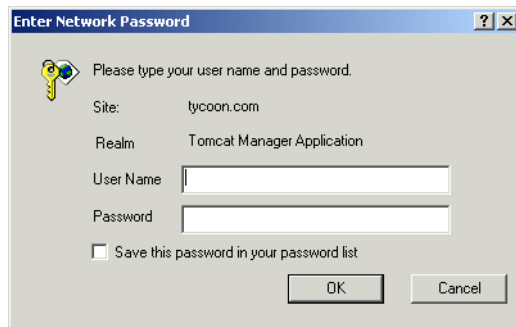
```
https://domain_name/tomcat/manager/html
```

For example:

```
https://www.digitalairlines.com/tomcat/manager/html
```

You can replace the domain name with your server's IP address. This is helpful if you do not have a DNS server.

- 2 When your browser's user authentication dialog box appears, enter your NetWare administrator username and password.



- 3 Click OK.

When you access Tomcat Manager for the first time, it might take a minute before it appears in your Web browser.

For complete Tomcat Manager documentation, see the [Tomcat Web Application Manager How To](http://jakarta.apache.org/tomcat/tomcat-4.1-doc/html-manager-howto.html) (<http://jakarta.apache.org/tomcat/tomcat-4.1-doc/html-manager-howto.html>) documentation on the Apache Web site.

What's Next

After Tomcat is installed and running on NetWare, refer to the [Tomcat 4 Servlet/JSP Container documentation](http://jakarta.apache.org/tomcat/tomcat-4.1-doc/index.html) (<http://jakarta.apache.org/tomcat/tomcat-4.1-doc/index.html>) on the Apache Web site.

4

Securing Tomcat on NetWare

The Tomcat Servlet Container has many built-in security mechanisms. On NetWare[®] 6.5, LDAP is used as the default security realm.

This section contains the following topics:

- ◆ [“Authentication and Authorization” on page 21](#)
- ◆ [“OU Container Rights” on page 21](#)
- ◆ [“Using Dynamic Groups” on page 22](#)
- ◆ [“What’s Next” on page 23](#)

Authentication and Authorization

There are two parts to Tomcat security: authentication and authorization. Authentication occurs when a valid username is entered during login. A valid username is one that matches a Novell[®] eDirectory[™] user ID. Authorization occurs if the valid user has been assigned the correct roles that allow the user to have access to a secured Web application.

To facilitate authorization, a collection of role names is created in the Tomcat-Roles Organizational Unit (OU) in eDirectory. By default, this OU is created in the same container where the NetWare server’s context is created during installation of NetWare.

Roles are represented as eDirectory groups contained in the Tomcat-Roles OU. A role can then be used for authentication purposes when a client requests access to a specific Web application. For example, if you had an inventory tracking application that you wanted to secure, you would add the name of the role to the application’s XML file. When users request access to the application, they are prompted for a username and password. If the user object does not exist in the Tomcat-Roles OU, they do not have access to the application. Therefore, for users to be authorized in the admin role, they must be a member of the admin Group.

OU Container Rights

During installation, Public browse rights are granted on the Tomcat-Roles Organizational Unit. Tomcat first authenticates with the user-supplied username and password. If successful, it does an anonymous LDAP bind to discover if the user is authorized in the role. If there are no Public browse rights, the authorization fails.

- ◆ [“Required Settings” on page 22](#)
- ◆ [“Relocating the Tomcat-Roles Container” on page 22](#)

Required Settings

For secure access to work, the following settings are required:

- ◆ Public must be a trustee of Tomcat-Roles
- ◆ On the Public trustee, the [All Attributes Rights] property must have Compare and Read rights. This must be inheritable.
- ◆ On the Public trustee, the [Entry Rights] property must have Browse rights. This must also be inheritable.

Relocating the Tomcat-Roles Container

If needed, you can relocate the Tomcat-Roles container so that it is available across a directory tree, or you can move it to a lower-level container.

To move the Tomcat-Roles container, move it as you do any other container, or you can re-create it as long as you maintain the correct rights. Then modify the `server.xml` and `admin-tomcat.xml` files to reflect the new location.

In `sys:/tomcat/4/conf/server.xml`, find the `<Realm className="JNDIRealm">` tag. The tag includes an LDAP search string to match the users for authentication, and also for lookup of roles for authorization. `{0}` represents the user name supplied at login time. You can simply use `{0}` as the authentication string, requiring a fully-specified LDAP context to log in (for example, `cn=admin,ou=myorg,o=mycompany`), or you can specify the context of the users. This gives you a contextless login, though it requires all of the users to be in the same container. Because there is only one Tomcat-Roles container involved in the authorization process, this approach is more straightforward. Simply supply the path of the Tomcat-Roles container in the authorization search string, whether this container is tree-wide or in a container that is more specific to the individual server.

You can add additional roles to your NetWare server by simply adding additional groups with role names into the Tomcat-Roles OU. If you have existing Web applications that use role-based security, they will automatically work once you create the required role group and add members to it.

Using Dynamic Groups

Using groups to represent roles lets you set roles up as dynamic groups. Dynamic groups can be created using iManager. If you are using dynamic groups, the administrator does not need to constantly manage the Tomcat roles. In this case, eDirectory can manage the Tomcat roles for you. For example, if you have a payroll application that requires the Payroll role, you can set up a dynamic Payroll group in the Tomcat-Roles container. It could contain a rule to include all members of the `ou=payroll.o=yourcompany`, and when users are created in that container, they are automatically granted access to the payroll Web application.

The same rule applies when a user moves out of the Payroll container; they immediately cease to have rights to the payroll Web application without any action from the system administrator or Web application administrator.

What's Next

After Tomcat is installed and running on NetWare, see the [Tomcat 4 Servlet/JSP Container documentation \(http://jakarta.apache.org/tomcat/tomcat-4.1-doc/index.html\)](http://jakarta.apache.org/tomcat/tomcat-4.1-doc/index.html) on the Apache Web site.

