Administering WAY4™ Application Server

Table of Contents

Introduction

This document gives an overview of WAY4 Application Server architecture, principles of operation, and setup and administration rules.

When working with this document, it is recommended to refer to the following resources from the OpenWay documentation series:

- Management Web Console Installation and Configuration Manual (m2_web_console).
- Management Web Console (m2_web_console) Operation Manual.

The following notation is used in this document:

- Key combinations are shown in angular brackets such as <Ctrl>+<F3>.
- Button labels used in screen forms are placed in square brackets, such as [Approve].
- The names of directories and/or files that vary for each local instance of the program are also displayed in angular brackets, like <AppServer_HOME>.

Warnings and information messages are marked as follows:

> ❗ Warnings about potentially hazardous situations or actions.

> ⓘ Information about important features, additional options, or the best use of certain system functions.

Chapter 1. WAY4 Application Server Overview

WAY4 Application Server is the platform for running WAY4 applications: J2EE Web applications and back-end applications that require high availability:

- WAY4 web application – an application developed according to a J2EE specification, distributed using a WAR archive.
- WAY4 application – an application distributed using a ZIP archive.

WAY4 Application Server has been developed to make possible the highly available execution of WAY4 applications, their monitoring and troubleshooting errors.

To run WAY4 applications on the WAY4 Application Server, perform these actions:

- Install and configure the WAY4 Application Server instance.
- Start WAY4 Application Server.
- Create an instance of the WAY4 application on the running WAY4 Application Server.
- Configure the WAY4 application.
- Start the configured WAY4 application and confirm its availability.

> ⓘ It is strictly forbidden to use third-party applications under the management of WAY4 Application Server without the WAY4 vendor's prior approval.

Chapter 2. WAY4 Application Server Architecture

The WAY4 Application Server architecture is shown in Fig. 1.

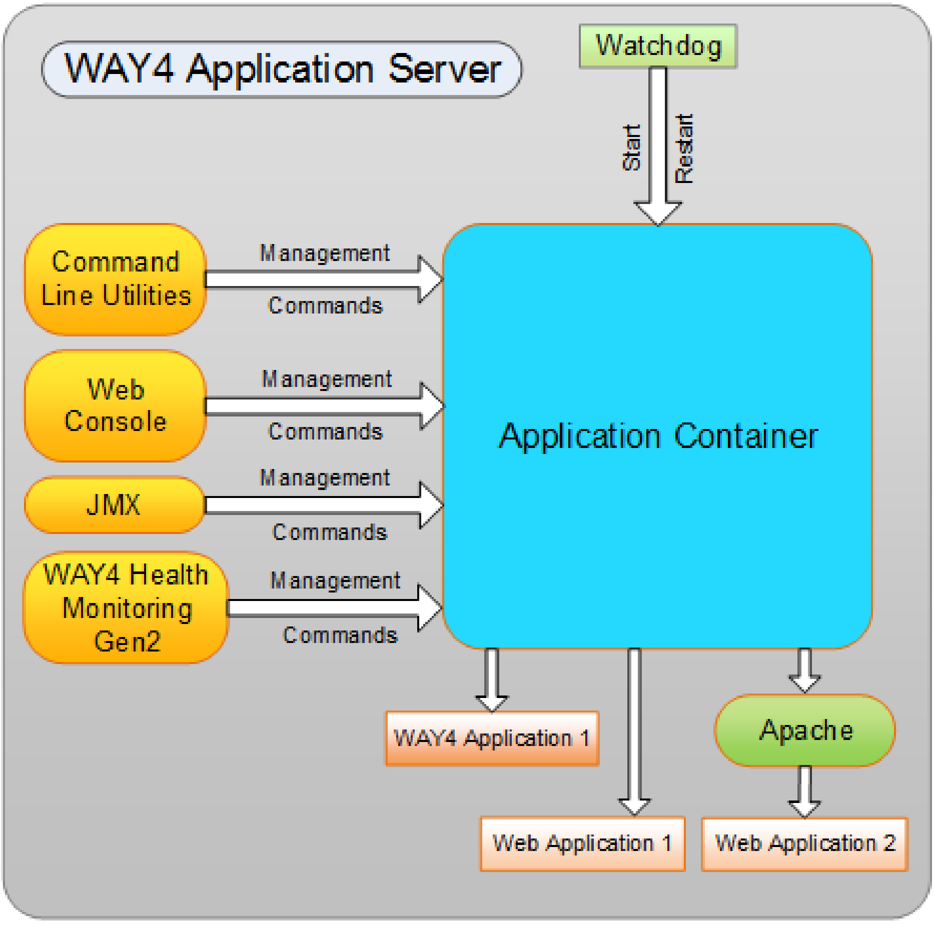


Fig. 1. WAY4 Application Server architecture

WAY4 Application Server includes:

- Application Container – the main process delivering application management commands. Also, this process monitors application availability and will restart the application when necessary.
- Command Line Utilities – console utilities through which the administrator sends management commands for applications and main process.

> ⓘ Console utilities may only be installed on the WAY4 Application Server computer.

- Web Console – web console for managing WAY4 applications.
- JMX – Java Management Extensions, technology for managing WAY4 applications.

- WAY4 Health Monitoring Gen2 – system application used to gather and process information about the functioning of system applications and components and for sending this information to external monitoring systems.

ⓘ The WAY4 Health Monitoring Gen2 solution is supplied according to an additional agreement with the WAY4™ vendor.

- Apache – "apache_24" application (or the web server component when a WAY4 Application Server version earlier than 1.7.1402 is used), see "Apache Web Server").
- Watchdog – service process that ensures the availability of the main process.

ⓘ For Microsoft Windows, the "WAY4 Application Server" service is used instead of the auxiliary process "Watchdog".

ⓘ WAY4 applications are not part of the WAY4 Application Server architecture and only appear in the figure above for demonstration purposes.

Chapter 3. Installing WAY4 Application Server

This section describes the rules for WAY4 Application Server installation under Unix operating systems (Red Hat Enterprise Linux, CentOS (Community ENTerprise Operating System), OEL (Oracle Enterprise Linux), IBM AIX and Oracle Solaris) and Microsoft Windows. Before WAY4 Application Server installation, it is necessary to perform preliminary setup of the operating system and install additional software. It is also necessary to install Oracle Java on all operating systems except IBM AIX (s ee "Preliminary Setup of the Operating System").

Note that only one instance of WAY4 Application Server can be installed on one operating system in one server.

WAY4 Application Server Distribution Kit

The WAY4 Application Server distribution includes the files shown in Table 1. <AppServer_HOME> – "<WAY4ApplicationServer>/appserver/" directory (<WAY4ApplicationServer> – the directory in which WAY4 Application Server is installed).

Table 1. WAY4 Application Service distribution kit

Group	Component	Description
Apache	appserver-apache-<platform>-<2.4_apache.version>.zip	Web server application.
Core	appserver-ant-<core_version>.zip	Apache Ant component.
	appserver-certs-<core_version>.zip	Certificate generation utilities.
	appserver-docs-<core_version>.zip	WAY4 Application Server documentation.
	appserver-startup-<core_version>.zip appserver-startup-common-<core_version>.zip	Scripts for starting, stopping, and registering WAY4 Application Server services.
	appserver-system-conf-<core_version>.zip	WAY4 Application Server system (read-only) configuration files.
	appserver-system-lib-<core_version>.zip	WAY4 Application Server system libraries (.jars).

Group	Component	Description
	appserver-tomcat-<core_version>.zip	Apache Tomcat component.
	appserver-sysapps-standard-<core_version>.zip	System applications, for example, WAR archive of the web console component and zip archive of the "monitoring" application.
	appserver-user-conf-<core_version>.zip	User configurable WAY4 Application Server configuration files.
	appserver-legacy-<platform>-<legacy_version>.zip	Component used for starting, stopping, and registering WAY4 Application Server services. The archive also contains the "nscipher.exe" utility used to encrypt passwords; after installation, the utility will be located in the directory "<AppServer_HOME>/bin/tools".
Installer	appserver-installer-<installer_version>-<platform>.zip	WAY4 Application Server installer, 'syscheck' utility.
Version	appserver-version-<appserver_version>.zip	Component for supporting WAY4 Application Server versions.
Checksum	Checksum.dat	File containing checksums of each distribution file.

Installing WAY4 Application Server under Unix

WAY4 Application Server must work in a session of a specific user (by default, the user "way4", group "way4" is specified during installation). If necessary, create this user account before installation.

WAY4 Application Server can be installed in one of the following modes:

- GUI.
- Command line interface.
- Silent mode (installation without user participation).

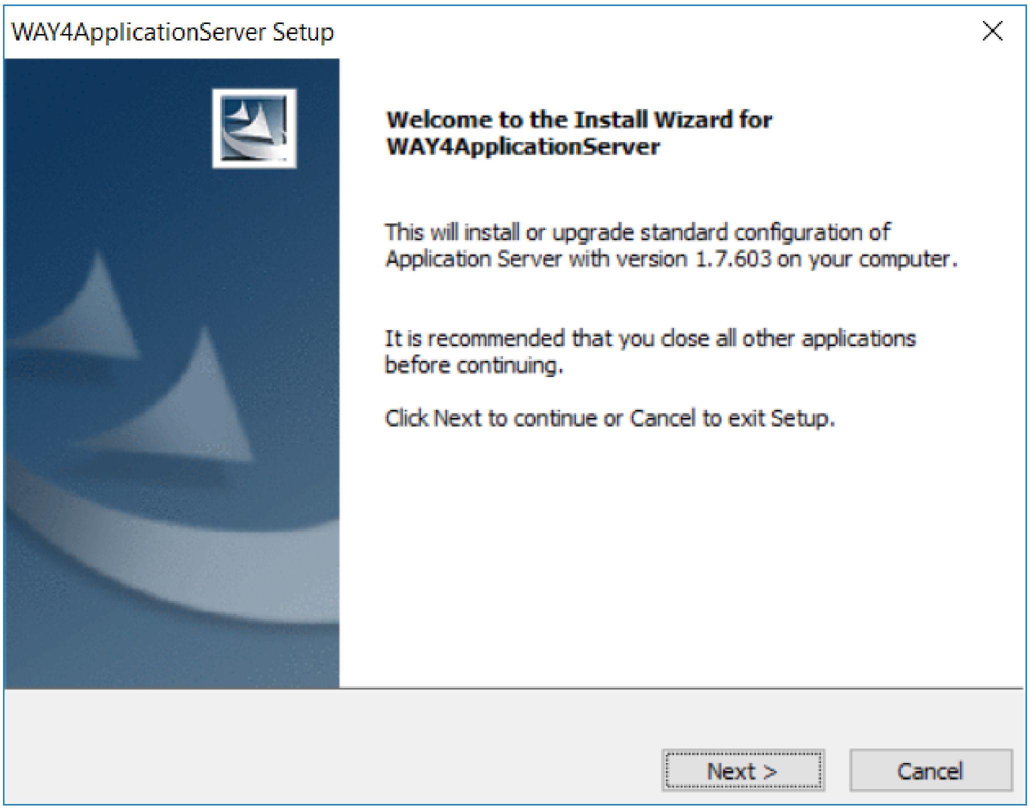
To install WAY4 Application Server, perform these actions:

- Copy the archive with the distribution kit of the WAY4 Application Server (the file "appserver-<version>-<platform>.zip") to the computer where it will be installed (see the section "WAY4 Application Server Distribution Kit" for file names and functions).
- Unpack the "appserver-<version>-<platform>.zip" file containing the installer "WAY4ApplicationServer-<installer_version>-<platform>-Install".
- Run the installer "WAY4ApplicationServer-<installer_version>-<platform>-Install" as a superuser (root).

> ⓘ This installer may also be run to upgrade WAY4 Application Server

After starting, the installation program determines which interface is in use at the present time (GUI or command line interface), and WAY4 Application Server is installed in the corresponding mode.

If installation is being performed in GUI mode, the following windows will be displayed on the screen.



WAY4ApplicationServer Setup

Welcome to the Install Wizard for WAY4ApplicationServer

This will install or upgrade standard configuration of Application Server with version 1.7.603 on your computer.

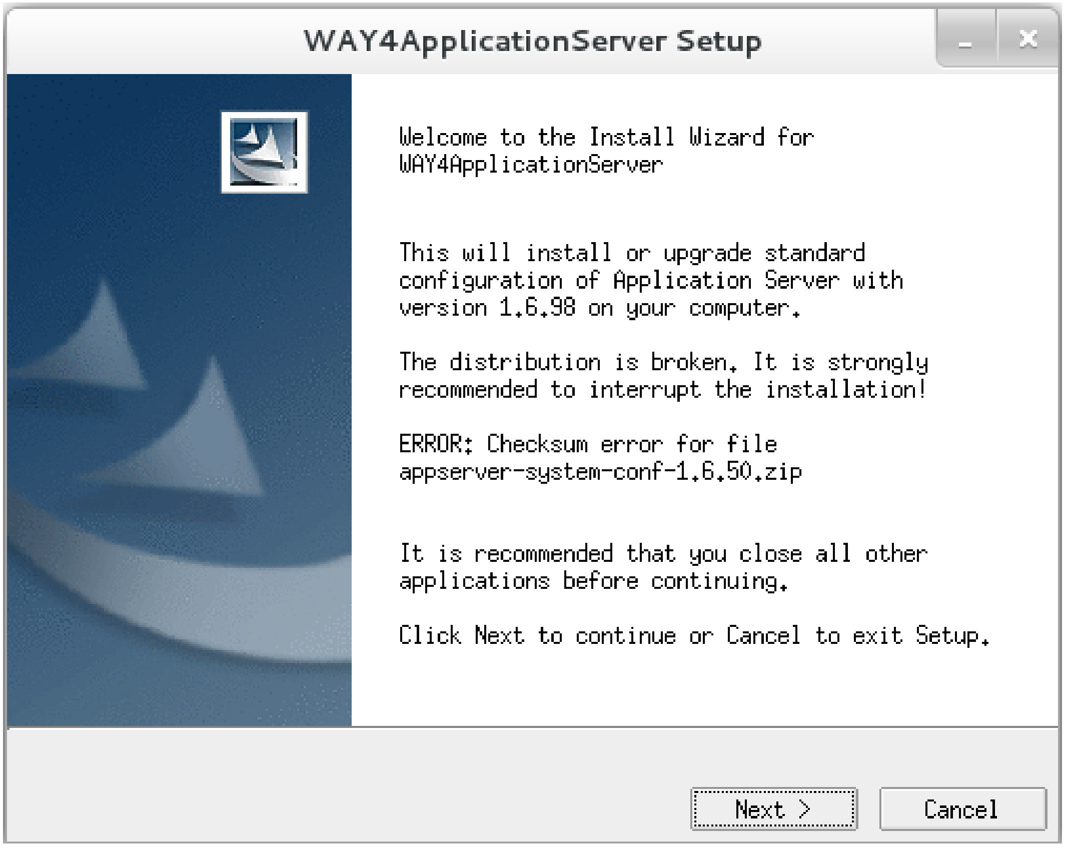
It is recommended that you close all other applications before continuing.

Click Next to continue or Cancel to exit Setup.

Fig. 2. "WAY4ApplicationServer Setup" window

At the start of installation, a process will be started during which distribution file checksums and the checksums specified in the "checksum.dat" file are checked for correspondence. If this process is completed without errors, the initial installation window will be displayed (see Fig. 2).

If the checksums do not correspond, the following window with an error message will be displayed.


WAY4ApplicationServer Setup

Welcome to the Install Wizard for WAY4ApplicationServer

This will install or upgrade standard configuration of Application Server with version 1.6.98 on your computer.

The distribution is broken. It is strongly recommended to interrupt the installation!

ERROR: Checksum error for file appserver-system-conf-1.6.50.zip

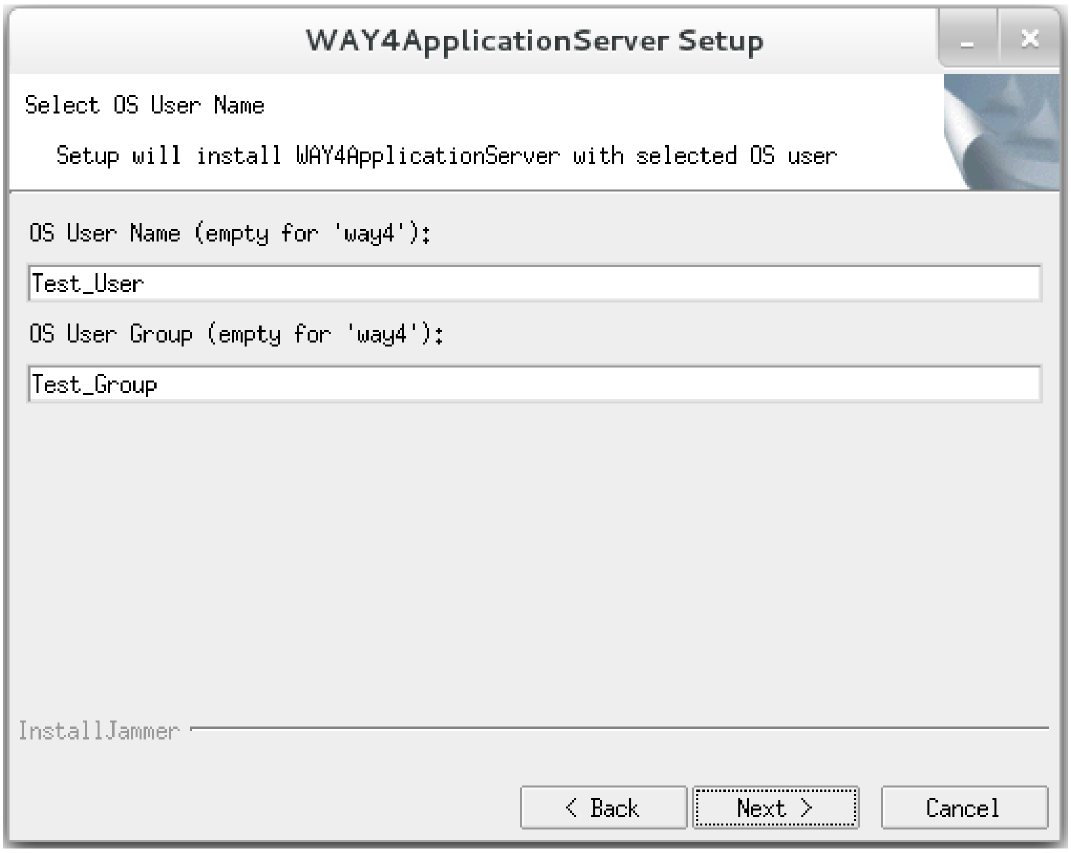
It is recommended that you close all other applications before continuing.

Click Next to continue or Cancel to exit Setup.

Fig. 3. Error when checking file checksums

In this case, it is necessary to fix the error and restart the installation process.

To continue installation, click [Next]; to cancel installation, click [Cancel].


WAY4ApplicationServer Setup

Select OS User Name

Setup will install WAY4ApplicationServer with selected OS user

OS User Name (empty for 'way4'): Test_User

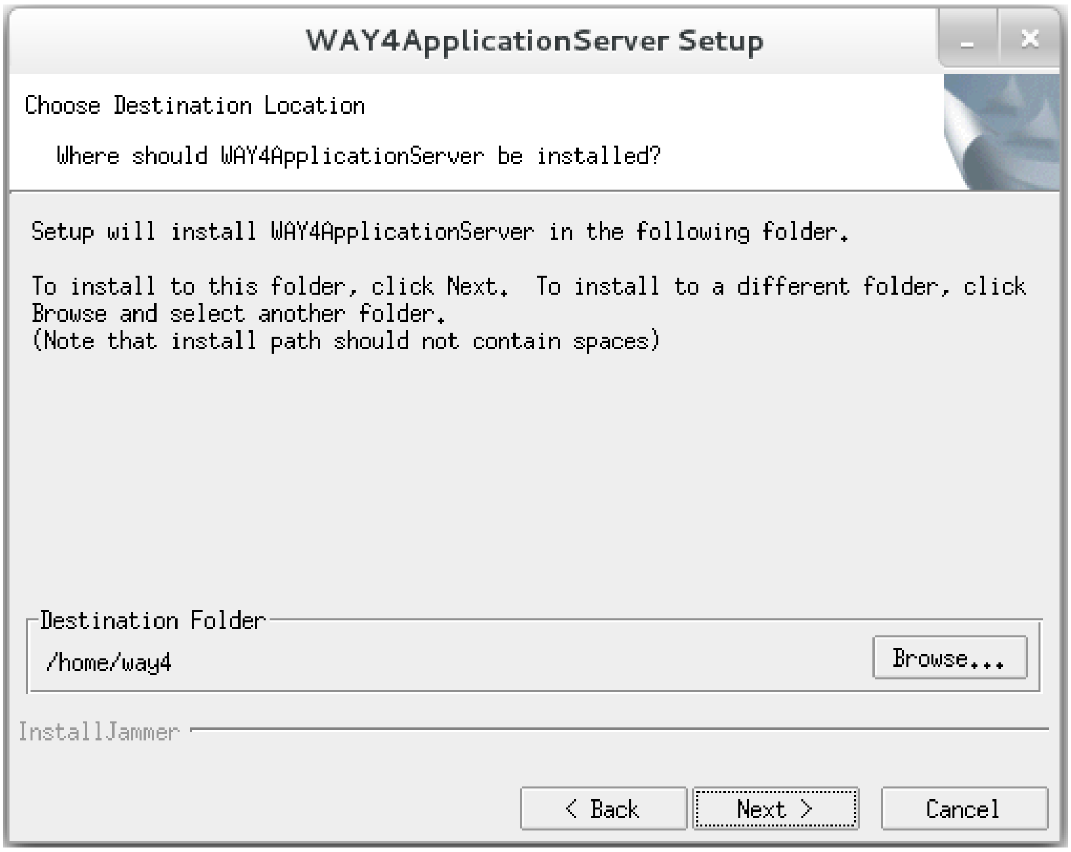
OS User Group (empty for 'way4'): Test_Group

Fig. 4. Setting the user name and group

In the *OS User Name (empty for 'way4')* field of this form, specify the name of the operating system user in whose session WAY4 Application Server will be used. In the *OS User Group (empty for 'way4') field*, specify the name of the user group to which the user whose name is specified in the *OS User Name (empty for 'way4')* field belongs.

ⓘ If the *OS User Name (empty for 'way4')* and *OS User Group (empty for 'way4')* fields are not filled in, "way4" user will be specified as the user and "way4" group as the group.

To continue installation, click [Next].



WAY4ApplicationServer Setup

Choose Destination Location

Where should WAY4ApplicationServer be installed?

Setup will install WAY4ApplicationServer in the following folder.

To install to this folder, click Next. To install to a different folder, click Browse and select another folder.
(Note that install path should not contain spaces)

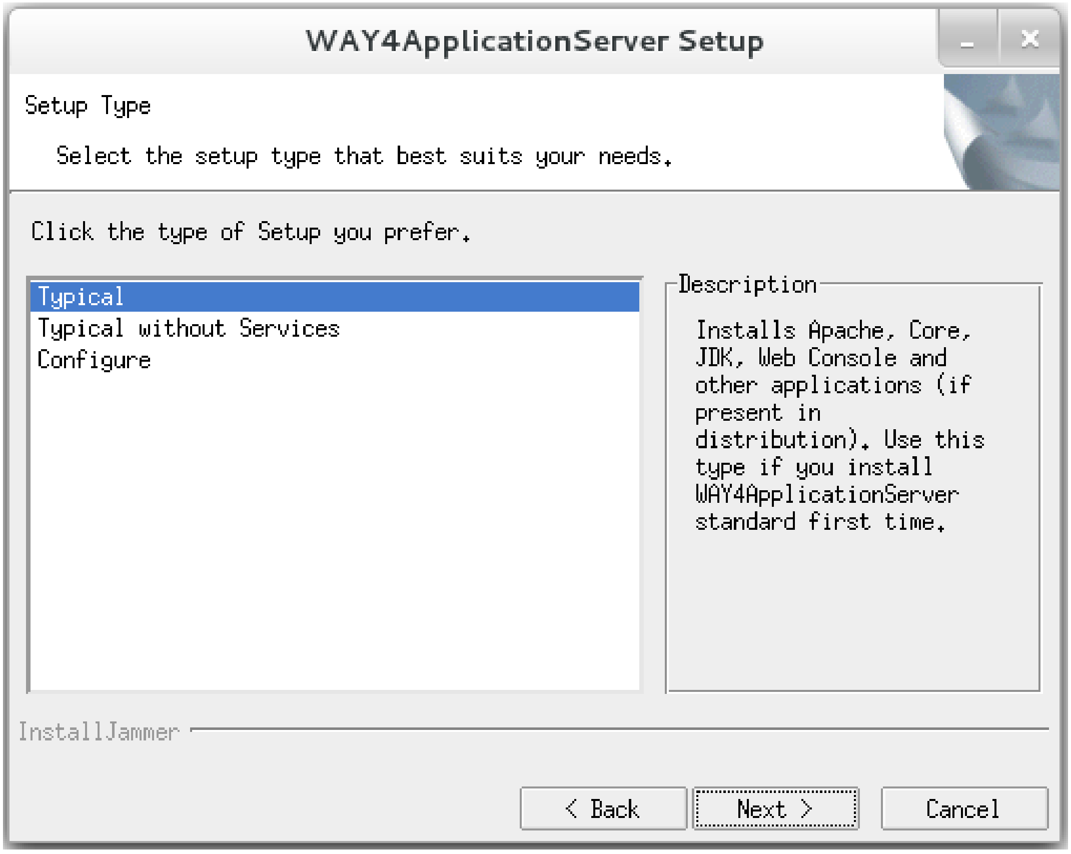
Destination Folder: /home/way4

Fig. 5. Specify the directory where WAY4 Application Server will be installed

In this window, use the [Browse] button to specify the directory where WAY4 Application Server will be installed. By default, WAY4 Application Server will be installed in the directory "/home /way4".

ⓘ Note that directory names may not contain spaces.

To continue installation, click [Next]; to return to the previous window, click [Back]; to cancel installation, click [Cancel].



WAY4ApplicationServer Setup

Setup Type

Select the setup type that best suits your needs.

Click the type of Setup you prefer.

Typical
Typical without Services
Configure

Description

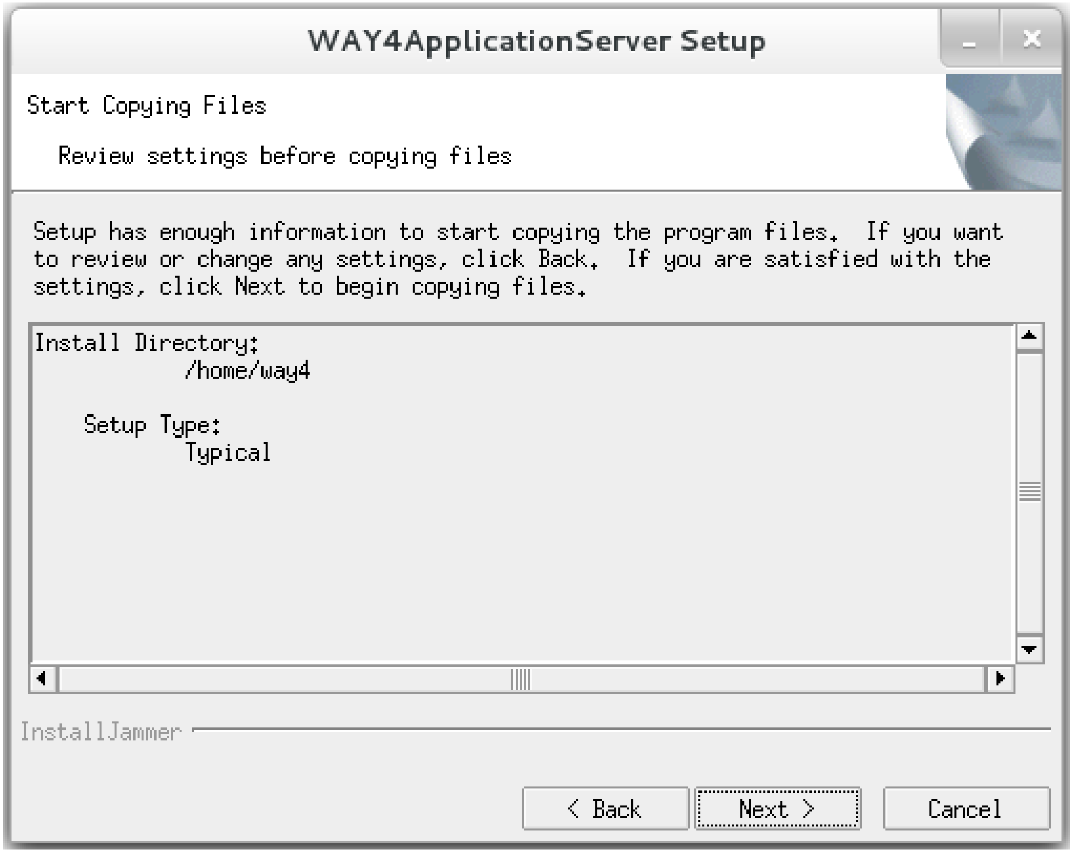
Installs Apache, Core, JDK, Web Console and other applications (if present in distribution). Use this type if you install WAY4ApplicationServer standard first time.

Fig. 6. Select installation type

Select the installation type in this form:

- "Typical" – full installation of WAY4 Application Server. It is recommended to select this installation type when installing WAY4 Application Server for the first time.
- "Typical without Services" – installation of WAY4 Application Server without creating a service. It is recommended to choose this type of installation in test mode only.
- "Configure" – repeat registration of a service, change in operating system user, and installation of some components. It is recommended to select this installation type when changing the operating system user, when it is necessary to rename a service or when errors arise in the operation of services.

To continue installation, click [Next].



Start Copying Files

Review settings before copying files

Setup has enough information to start copying the program files. If you want to review or change any settings, click Back. If you are satisfied with the settings, click Next to begin copying files.

```
Install Directory:
           /home/way4

    Setup Type:
           Typical
```

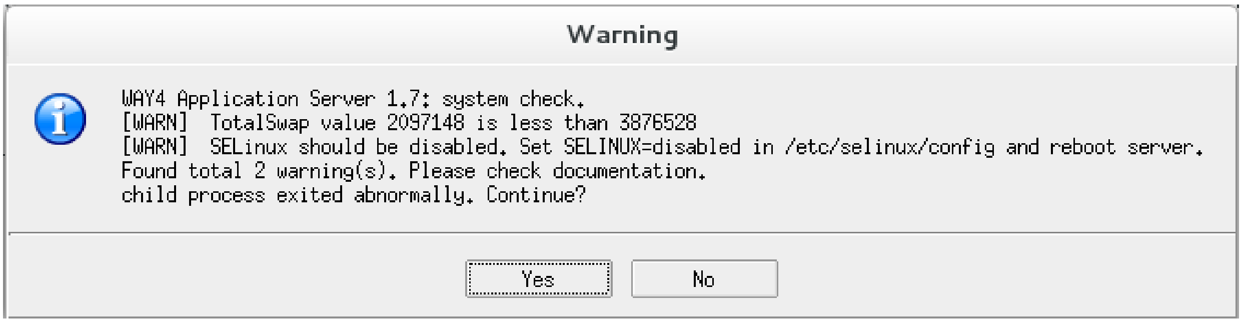
Fig. 7. Installation parameters

Information about installation parameters is shown in this window.

To continue installation, click [Next]. As a result, the installation process will be started.

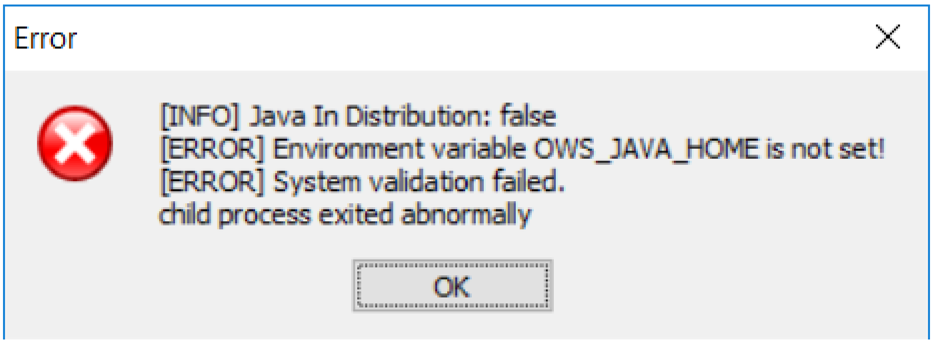
As a result, the process will be started to check that the parameters of the server (hardware-software suite) on which WAY4 Application Server is being installed meet technical requirements and instructions given in the section "Preliminary Setup of the Operating System".

If inconsistencies were found during the check, a window will be displayed with a warning message (Warning, see Fig. 8) and/or error message (Error, see Fig. 9), as well as an informational message (Info).



```
WAY4 Application Server 1.7: system check.
[WARN]  TotalSwap value 2097148 is less than 3876528
[WARN]  SELinux should be disabled. Set SELINUX=disabled in /etc/selinux/config and reboot server.
Found total 2 warning(s). Please check documentation.
child process exited abnormally. Continue?
```

Fig. 8. Warning during system check



```
[INFO] Java In Distribution: false
[ERROR] Environment variable OWS_JAVA_HOME is not set!
[ERROR] System validation failed.
child process exited abnormally
```

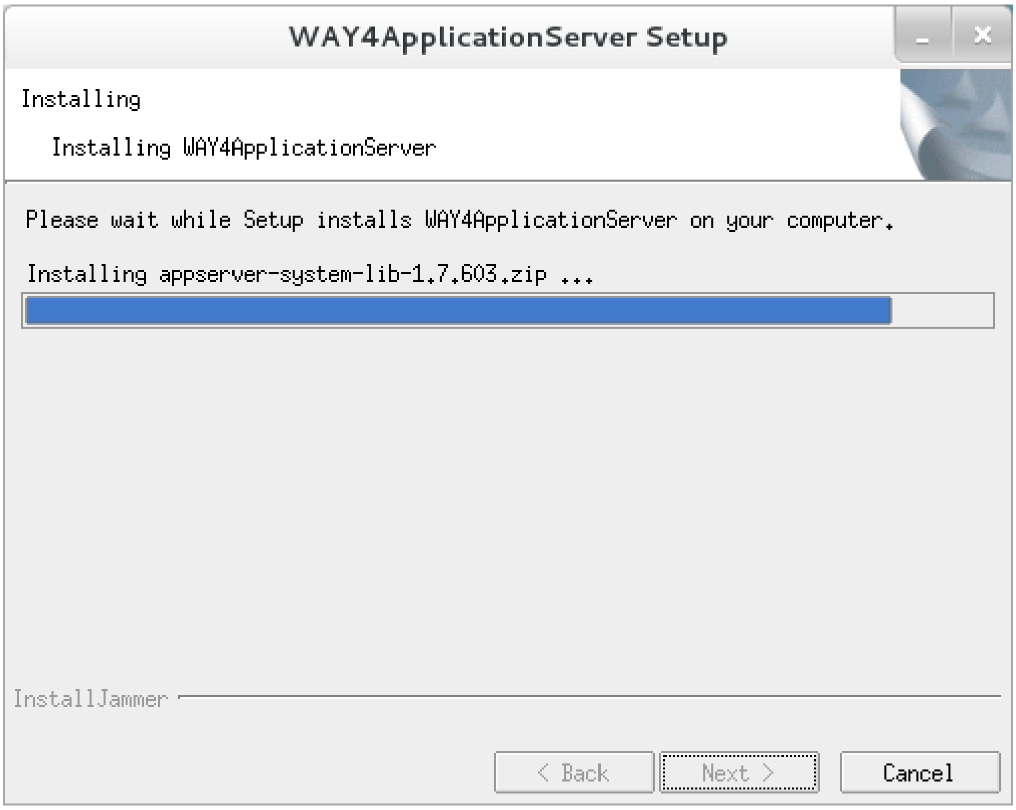
Fig. 9. Error during system check

If only Warnings occurred during the system check for compliance with requirements, the installation process may be continued. To do so, click [Yes] in the "Warning" form (see Fig. 8). In this case, after installation is completed, WAY Application Server may operate incorrectly. For correct operation, it is recommended to click [No] in the "Warning" form, fix all warnings and restart the installer.

If errors occur during the system check (for example, insufficient space for installing WAY4 Application Server), an "Error" window will be displayed (see Fig. 9). Clicking [OK] terminates the installation process. In this case, fix the errors and restart the installer.

To fix errors and warnings follow the instructions given in the section "Preliminary Setup of the Operating System" and ensure that hardware parameters comply with the technical requirements described in the document "WAY4 Application Server Main Technical Requirements".

If there are no errors and warnings during the system parameter check, the installation process will be continued.



WAY4ApplicationServer Setup

Installing

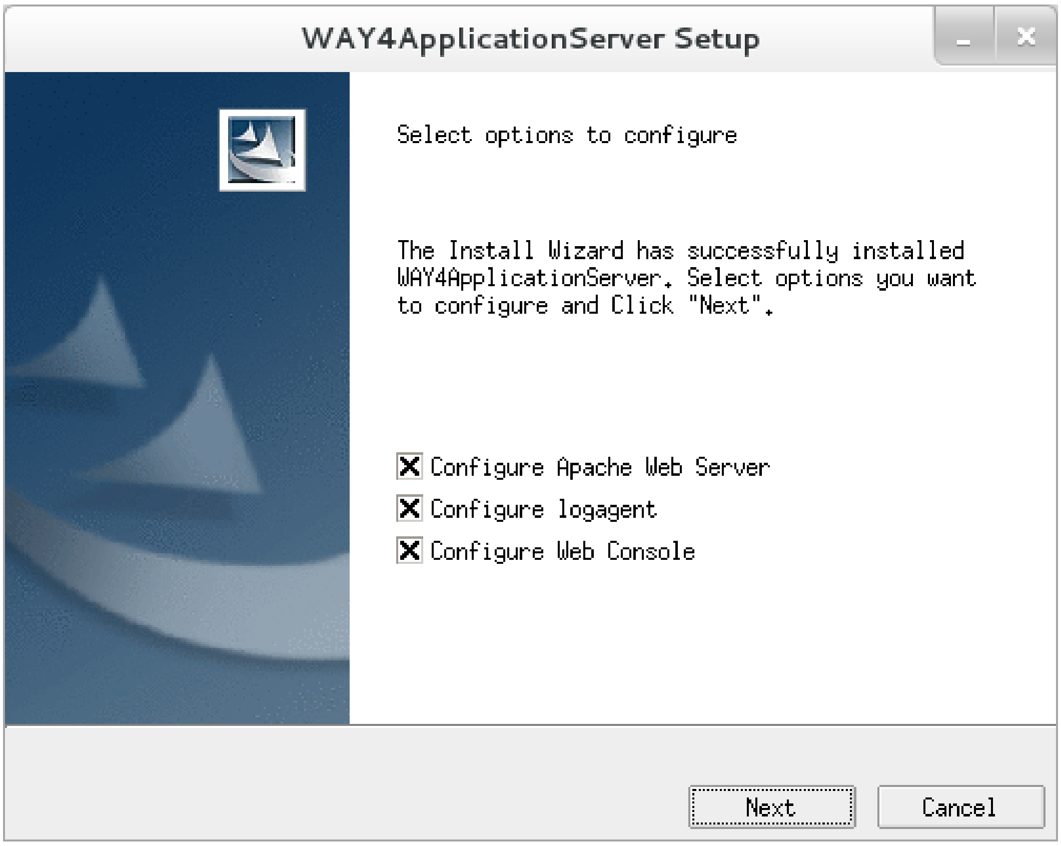
Installing WAY4ApplicationServer

Please wait while Setup installs WAY4ApplicationServer on your computer.

Installing appserver-system-lib-1.7.603.zip ...

Fig. 10. Installing

If installation is successfully completed, the following window will be displayed on the screen:



WAY4ApplicationServer Setup

Select options to configure

The Install Wizard has successfully installed WAY4ApplicationServer. Select options you want to configure and Click "Next".

☒ Configure Apache Web Server

☒ Configure logagent

☒ Configure Web Console

Fig. 11. Successful completion of installation

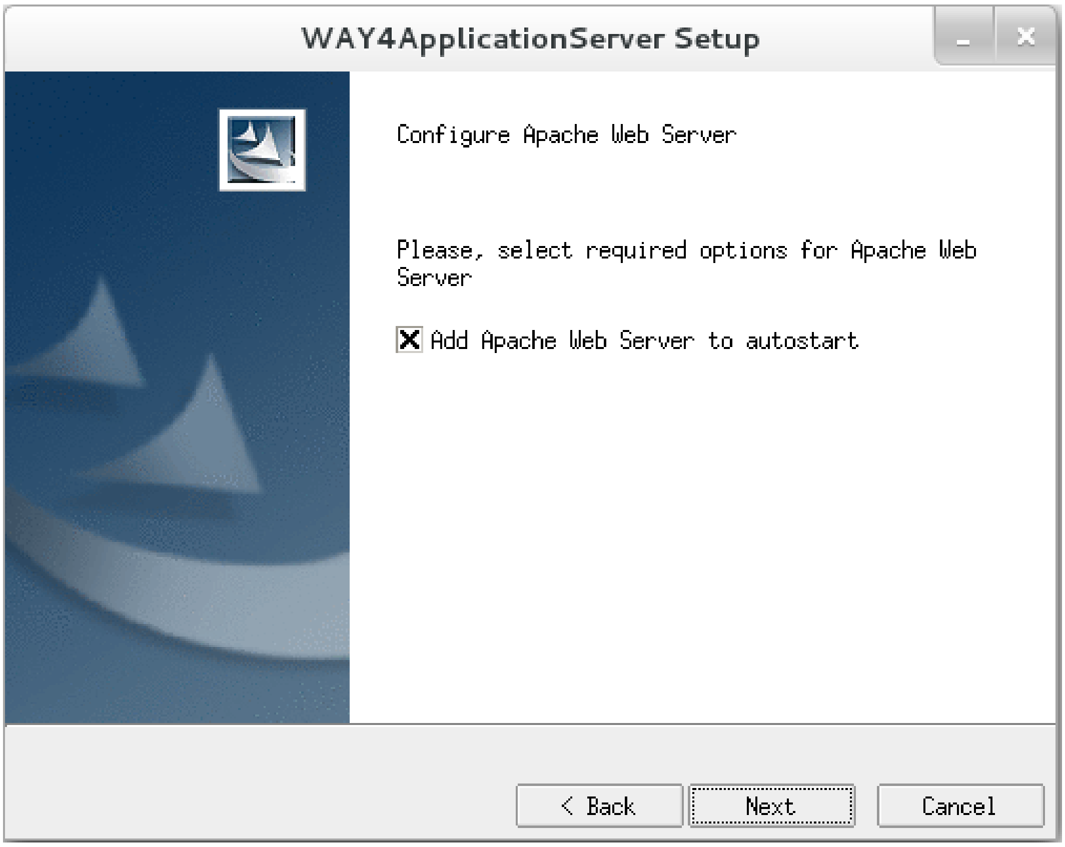
To complete installation of WAY4 Application Server, click the [Next] button.

The following checkboxes are displayed in this window:

- *Configure Apache Web Server* – configure the Apache web server application ("apache_24").
- *Configure Logagent* – configure Log agent.
- *Configure Web Console* – configure the WAY4 application web console will be started.

After [Next] is clicked, the following windows will be displayed (in the order specified and depending on the checkboxes that were selected):

- If the Configure Apache Web Server checkbox is selected, a window for configuring the Apache web server application ("apache_24") will be displayed (see Fig. 12).



Configure Apache Web Server

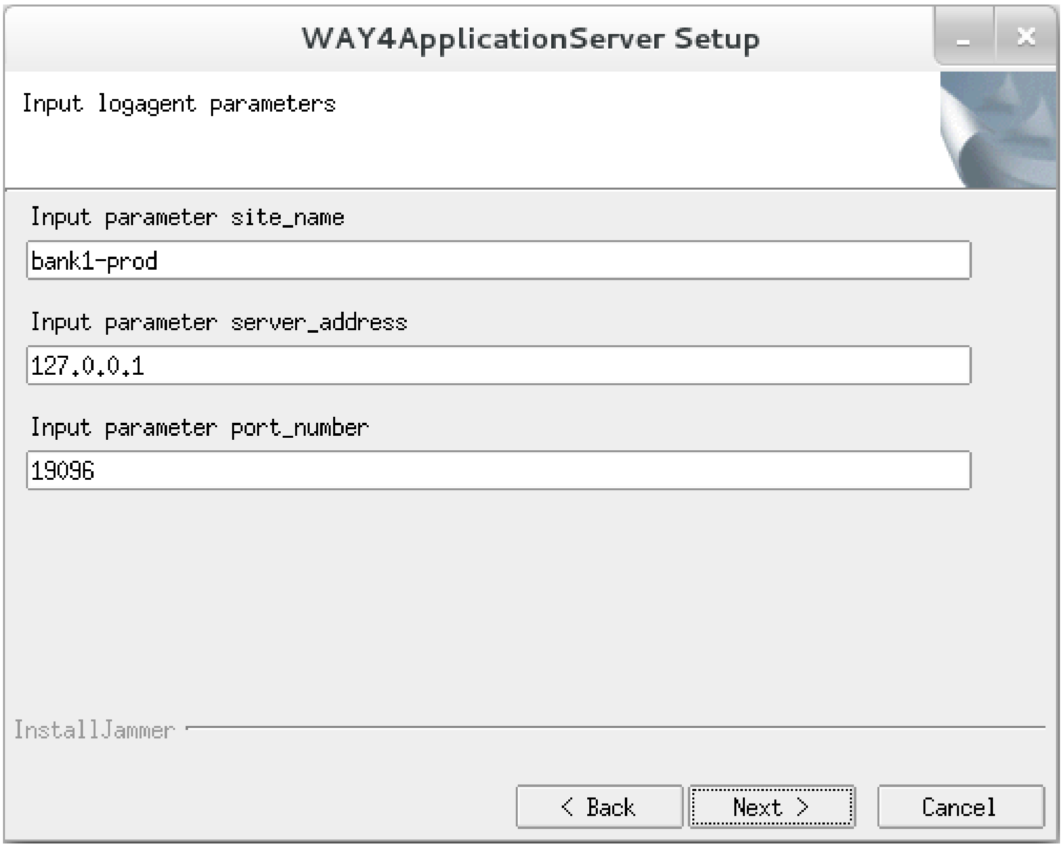
Please, select required options for Apache Web Server

☒ Add Apache Web Server to autostart

Fig. 12. Configuring the Apache web server application ("apache_24")

When the *Add Apache Web Server to autostart* checkbox is selected, the Apache web server application ("apache_24") will start automatically when WAY4 Application server is started.

- If the *Configure Logagent* checkbox is selected, a window for configuring Log agent will be displayed (see Fig. 13).



WAY4ApplicationServer Setup

Input logagent parameters

Input parameter site_name

bank1-prod

Input parameter server_address

127.0.0.1

Input parameter port_number

19096

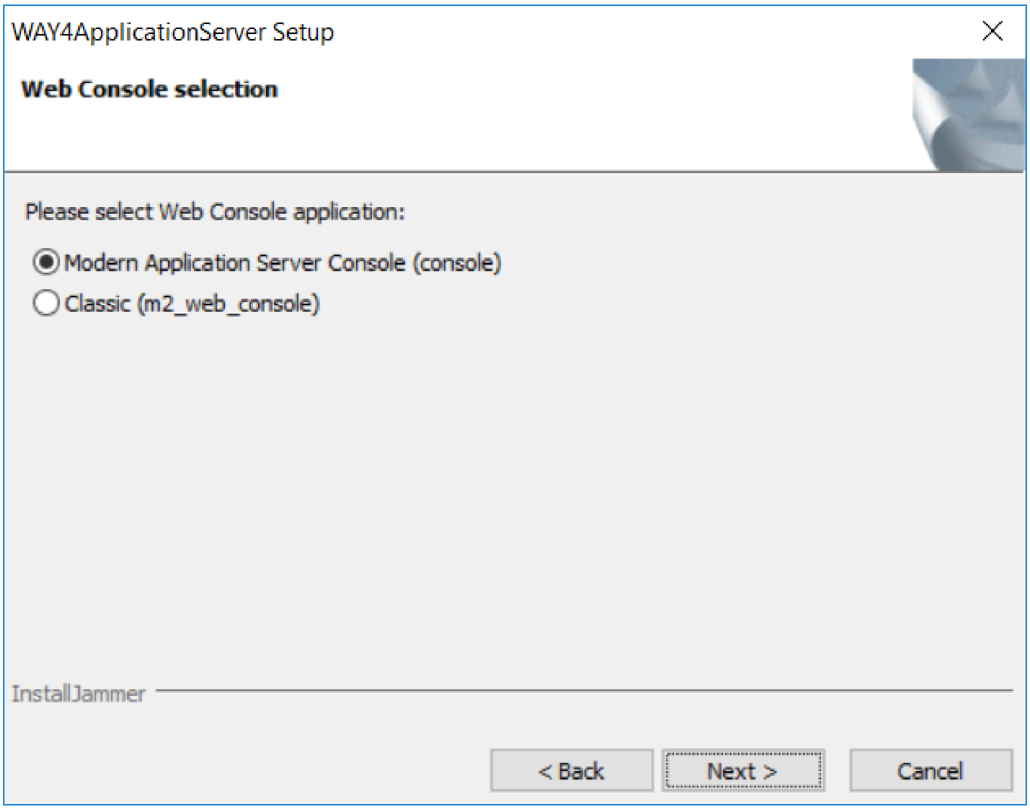
Fig. 13. Configuring Log agent

The following parameters are specified in this window:

- "site_name" – Client identifier written to the header of each log file. When analyzing application log files, makes it possible to identify the client from which a file was received.
- "server_address" – IP address of the server on which the system application WAY4 Health Monitoring Gen2 is installed.
- "port_number" – port number of the server on which the system application WAY4 Health Monitoring Gen2 is installed.

> ⓘ The WAY4 Health Monitoring Gen2 solution is supplied according to an additional agreement with the WAY4™ vendor.

- If the *Configure Web Console* checkbox is selected, a window will be displayed to select the WAY4 web console for which access parameters must be configured.



WAY4ApplicationServer Setup

Web Console selection

Please select Web Console application:

● Modern Application Server Console (console)

○ Classic (m2_web_console)

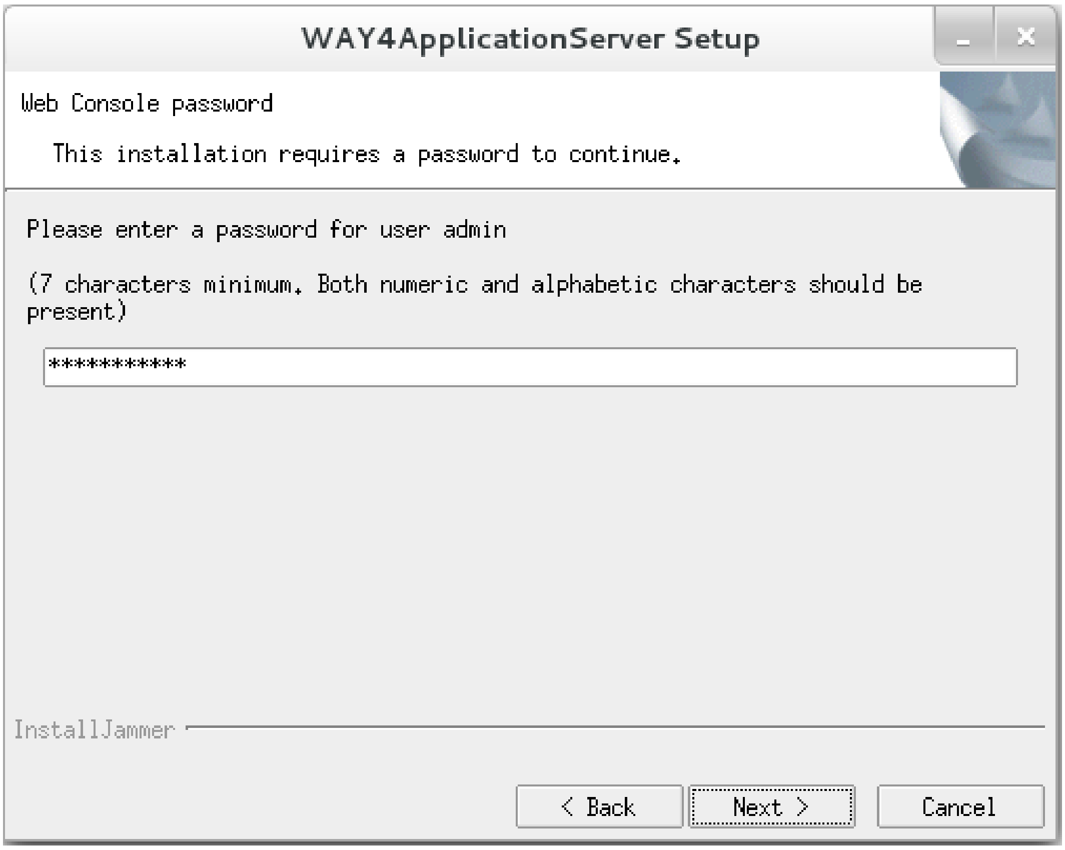
Fig. 14. Window for selecting the WAY4 web console

One of the following values can be selected in this window (the name of the system application is shown in brackets).

- Modern Application Server Console (console) – "updated" management web console.
- Classic (m2_web_console) – Management Web console (for more information, see the documents "Management Web Console Installation and Configuration Manual (m2_web_console)" and "Management Web Console (m2_web_console) Operation Manual").

Note that after installation of WAY Application Server is completed, the system application corresponding to the selected web console option will be started automatically, and the other application will not be started.

After choosing a web console and clicking [Next], a window for configuring the WAY4 application web console will be displayed (see Fig. 15).



WAY4ApplicationServer Setup

Web Console password

This installation requires a password to continue.

Please enter a password for user admin

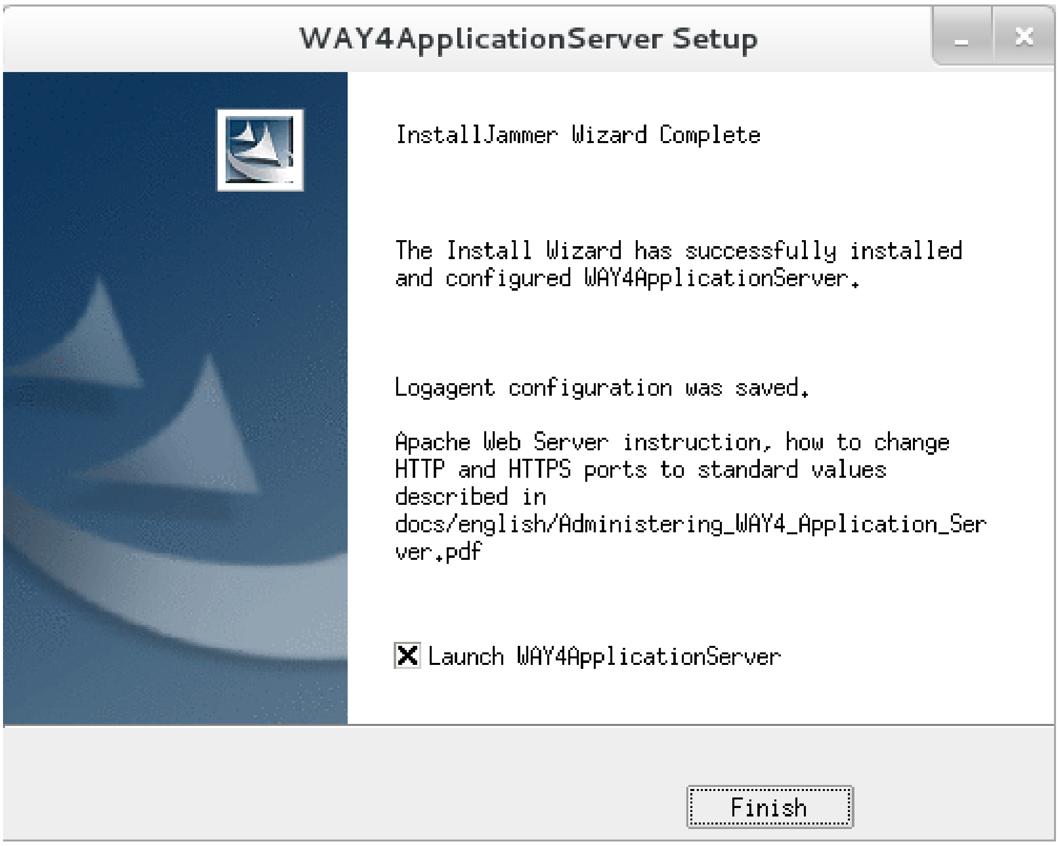
(7 characters minimum. Both numeric and alphabetic characters should be present)

```
***********
```

Fig. 15. Configuring the WAY4 Web console

In this window's field, specify the "admin" user password that will be used for access to the web console selected in Fig. 14. Password requirements pursuant to PCI DSS are described in the section "Registering Users and Setting Passwords" of the document "WAY4™ PCI DSS Implementation Guide". After clicking the [Next] button the web console will be configured (assignment of HTTP/HTTPS ports, creation of an "admin" user, certificate generation and import). A window will be displayed with the message "Web Console is being configured now…".

At the end of the configuration process, the following window will be shown:



InstallJammer Wizard Complete

The Install Wizard has successfully installed and configured WAY4ApplicationServer.

Logagent configuration was saved.

Apache Web Server instruction, how to change HTTP and HTTPS ports to standard values described in docs/english/Administering_WAY4_Application_Server.pdf

☒ Launch WAY4ApplicationServer

Fig. 16. Installation window

To finish installation, click the [Finish] button. If the *Launch WAYApplication Server* checkbox is selected, WAY4 Application Server will be started.

> ⓘ If no flags are set (Fig. 11), the installation process will be completed.

If WAY4 Application Server is being installed in the command line interface mode, it is necessary to answer the following questions and fill in the fields as requested by the installer:

- "This will install or upgrade standard configuration of ApplicationServer with version <number> on your computer. Continue? [y/N]". To continue installation, answer "y" or press <Enter>.

  As a result, a process will be started during which the distribution file checksums and the checksums specified in the "checksum.dat" file are checked for correspondence. If the checksums do not correspond, a window with an error message will be displayed: "The distribution is broken. It is strongly recommended to interrupt the installation! ERROR: Checksum error for file <filename>.zip".

  In this case, it is necessary to fix the error and restart the installation process.

  If there are no errors when checking file checksums, the following message will be displayed:

- "Enter OS user [way4]" – specify the operating system user name in whose session WAY4 Application Server will be used. By default, (by pressing the <Enter> key), this will be the user with the name "way4".

- "Enter OS group [way4]" – specify the name of the user group to which the user specified in the previous paragraph belongs. By default (by clicking <Enter>) the "way4" group will be used.

- "Where do you want to install WAY4ApplicationServer? [/home/way4]". Specify the directory where WAY4 Application Server will be installed. By default (by clicking <Enter>) WAY4 Application Server will be installed in the directory "/home/way4".

- "WAY4 Application Server: system check" – starts the process for checking that the parameters of the server (hardware-software suite) on which WAY4 Application Server is being installed comply with technical requirements and the instructions given in the section " Preliminary Setup of the Operating System". If errors and warnings occur, the corresponding messages will be displayed. If even one error occurred, the installation process will be terminated; when there are no errors, but warnings exist, the message "Found total <number> warning(s). Please check documentation. Child processes exited abnormally. Continue? [y/N]" will be displayed. Clicking "y" continues the installation process. If warnings are not fixed, WAY4 Application Server may operate incorrectly after installation is completed, therefore is it recommended to terminate installation (click "N"), fix all warnings and restart the installer.

  To fix errors and warnings follow the instructions given in the section "Preliminary Setup of the Operating System" and ensure that hardware parameters comply with the technical requirements described in the document "WAY4™ Application Server Main Technical Requirements".

- "Would you like to configure Apache Web Server? [y/N]" – when "y" is pressed, the following question about configuring the web server component will be shown.

- "Would you like to start Apache Web Server automatically? [n/Y]" – when "Y" is pressed, the web server component will be started automatically when WAY4 Application Server is started.

- "Would you like to configure logagent? [y/N]" – when "y" is pressed, questions about configuring the Log agent component will be shown.

- "Input site_name (for logagent) [<Value>]" – Client identifier written to the header of each log file. When analyzing application log files, makes it possible to identify the client from which a file was received. By default (by pressing <Enter>) the "<Value>" value will be used.

- "Input server_address (for logagent) [<IP address>]"– IP address of the server on which the system application WAY4 Health Monitoring Gen2 is installed. By default (by pressing <Enter>) the "<IP address>" value will be used.

- "Input port_number (for logagent) [<Port>]" – port number of the server on which the system application WAY4 Health Monitoring Gen2 is installed. By default (by pressing <Enter>) the "<Port>" value will be used

- "Would you like to configure Modern Application Server Console (Console)? [y/N]" – to configure the web console, click "y" or <Enter>; to cancel, click "N".

- "Do you want to use Classic (m2_web_console) [Y] instead of Modern Application Server Console [n]? [n/Y]" – when "Y" is pressed, Management Web Console (m2_web_console) will be configured. For more information about Management Web Console, see the documents "Management Web Console Installation and Configuration Manual (m2_web_console)" and "Management Web Console (m2_web_console) Operation Manual"). When "n" is pressed, Modern Application Server Console will be configured.

- "Input password for user admin in Web Console (7 characters minimum. Both numeric and alphabetic characters should be present)" – a password to access the web console must be specified for the "admin" user; this password must be no less than 7 characters and include Latin letters and digits.
- "Web Console was configured." – completion of web console configuration.
- "Would you like to launch WAY4ApplicationServer? [n/Y]" – to start WAY4 Application Server immediately after installation has been completed, click "Y".
- "Installation complete" – the installation process has been completed.

To install WAY4 Application Server in silent mode (for example, for automatic installation), execute the following command:

```
WAY4ApplicationServer-<installer_version>-<platform>-Install --mode silent --
prefix <InstallDir> --user <user name> --group <group name>
```

The following parameters are used:

- <InstallDir> - the directory to which WAY4 Application Server will be installed. By default (if the parameter is not set) WAY4 Application server will be installed to the directory "/home /way4".
- <user name> - name of the operating system user in whose session WAY4 Application Server will be used. By default (if the parameter is not set), this user will be the "way4" user.
- <group name> - the name of the user group to which the user whose name is specified with the <user name> parameter belongs. By default (if the parameter is not set), the "way4" group will be used.

Installing WAY4 Application Server under MS Windows

To install WAY4 Application Server, proceed as follows:

- Copy the archive with the distribution kit of the WAY4 Application Server (the file "appserver-<version>-windows-<architecture>.zip") to the computer where it will be installed (see the section "WAY4 Application Server Distribution Kit" for file names and functions).
- Unpack the "appserver-<version>-windows-<architecture>.zip" file containing the installer "WAY4ApplicationServer-<installer_version>-Setup.exe".
- Run the installer "WAY4ApplicationServer-<installer_version>-Setup.exe".

The installation process is the same as for installing WAY4 Application Server under Unix (see " Installing WAY4 Application Server under Unix"). The only difference from installation under Unix is that for MS Windows, the process for checking the correspondence of distributive file checksums and the checksums specified in the "checksum.dat" file will not be run. To check checksums, download the "md5" utility (for example, from the site http://www.winmd5.com) and manually check the checksums.

When installing WAY4 Application Server on the MS Windows platform, the web server component (Apache Web Server) is not installed and configured.

During installation, the services "WAY4 Application Server" and "WAY4 Application Web Server" ensuring WAY4 Application Server and web server availability will also be installed.

> ⓘ The services are started and stopped automatically when the operating system starts and stops.

Chapter 4. Updating WAY4 Application Server

Upgrade of WAY4 Application Server components under Unix and MS Windows is described in the sections "Updating WAY4 Application Server under Unix" and "Updating WAY4 Application Server under MS Windows", respectively.

Updating WAY4 Application Server under Unix

> ⓘ It is necessary to stop the current version of WAY4 Application Server before updating. If WAY4 Application Server is not stopped, during installation, the following error message will be displayed "WAY4 Application Server is running. Stop Server manually and start installation again. Would you like to retry?". Stop WAY4 Application Server and click [Yes] in the error message window.

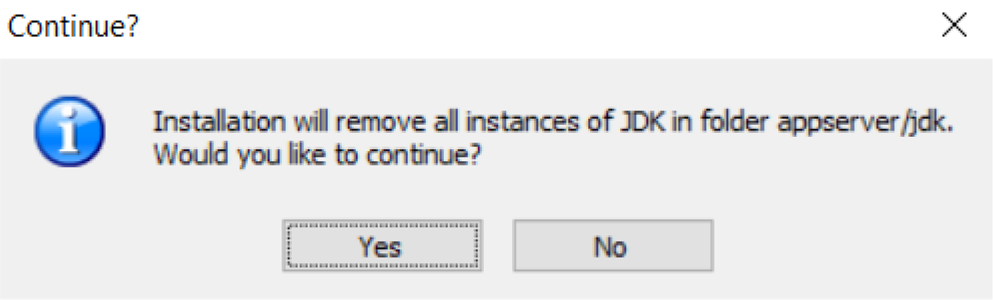
> ⓘ If the distribution of WAY4 Application Server being updated includes the "<AppServer_HOME>/appserver/jdk" directory, before starting update, delete the "WAY4ApplicationServer" service (for more information, see the section "Deleting WAY4 Application Server").

To update WAY4 Application Server, do as follows:

- Copy the archive with the distribution kit of a new WAY4 Application Server version (the file "appserver-<version>-<platform>.zip") to the computer where it will be deployed (for more details on file names and functions, see the section "WAY4 Application Server Distribution Kit ").

  > ⓘ Note that when WAY4 Application Server is updated, it is not necessary to unpack configuration files "appserver-user-conf-<core_version>.zip" and "appserver-apache-user-conf-<apache_version>.zip".

- Unpack the "appserver-<version>-<platform>.zip" file containing the installer "WAY4ApplicationServer-<installer_version>-<platform>-Install".
- Run the installer "WAY4ApplicationServer-<installer_version>-<platform>-Install" as a superuser (root).
- If the distribution of WAY4 Application Server being updated includes the "<AppServer_HOME>/appserver/jdk" directory, a warning that this directory will be deleted will be displayed. Click on the [Yes] button.



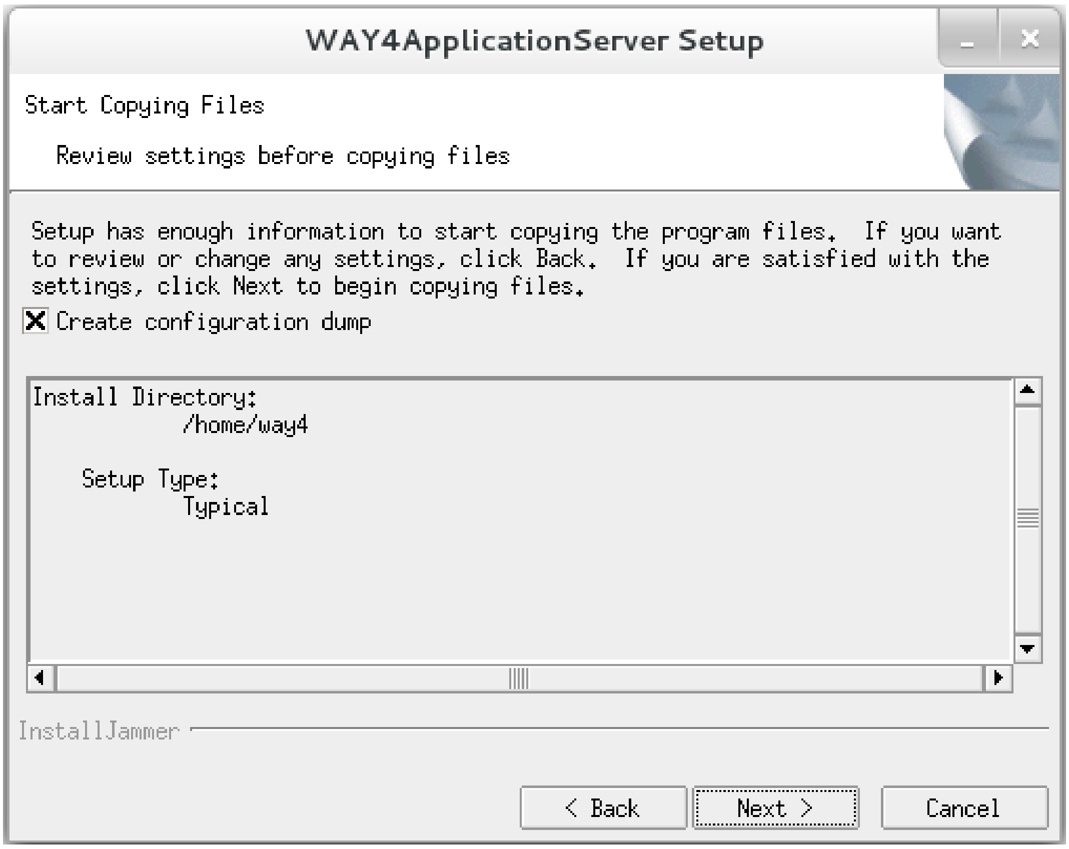
Continue?

Installation will remove all instances of JDK in folder appserver/jdk. Would you like to continue?

Warning about deleting the "<AppServer_HOME>/appserver/jdk" directory

The update process is similar to the installation of WAY4 Application Server under Unix (see " Installing WAY4 Application Server under Unix"); before installation it is possible to save the configuration of all installed applications and WAY4 Application Server:

- In GUI mode in the screen with installation parameters (see Fig. 17) the *Create configuration dump* checkbox will be shown. When this box is checked, the configuration of all installed applications and WAY4 Application Server will be saved in the "<AppServer_HOME>/dumps" directory. This checkbox is similar to execution of the "dump level=conf" command (see " dump").



WAY4ApplicationServer Setup

Start Copying Files

Review settings before copying files

Setup has enough information to start copying the program files. If you want to review or change any settings, click Back. If you are satisfied with the settings, click Next to begin copying files.

☒ Create configuration dump

```
Install Directory:
            /home/way4

    Setup Type:
            Typical
```

Fig. 17. Installation parameters and checkbox to save the configuration

- In the command line interface, the screen will display the messages:
  - "Would you like to create appserver configuration dump? [y/N]". Pressing "y" saves the configuration of all installed applications and WAY4 Application Server.
  - " Installation will remove all instances of JDK in folder appserver/jdk. Would you like to continue? [y/N] ", if the distribution of WAY4 Application Server being updated includes the "<AppServer_HOME>/appserver/jdk" directory . C lick on the [y] button .

- In silent mode set the parameter "--createdump yes" to save the configuration, the command will appear as follows:

```
WAY4ApplicationServer-<installer_version>-<platform>-Install --mode silent --
prefix <InstallDir> --user <user name> --group <group name> --createdump yes
```

When updating WAY4 Application Server from a version that is earlier than 1.7.1402, the "apache_24" will be additionally installed on the target system. The entire web server configuration that was set up in the web server component in the previous version of WAY4 Application Server will be duplicated in the configuration of the "apache_24" configuration. When the Add Apache Web Server to autostart checkbox is selected, the updated web server component will be started automatically.

To start the "apache_24" application, use the "wsstop" command to stop the web server component and use the "start" command to start the "apache_24" application from the directory "<AppServer_HOME>/appserver/applications /apache_24".

Over the next few days, ensure that the "apache_24" application runs without errors, and then delete the directories "<AppServer_HOME>/appserver/apache-2.4" and "<AppServer_HOME>/appserver/conf/webserver". In this case, the next time WAY4 Application Server is updated, the Apache web server component will not be updated; instead, the "apache_24" application will be updated, and when the *Add Apache Web Server to autostart* checkbox is selected, the updated "apache_24" application will be started automatically.

Updating WAY4 Application Server under MS Windows

ⓘ It is necessary to stop the current version of WAY4 Application Server before updating. If WAY4 Application Server is not stopped, during installation, the following error message will be displayed "WAY4 Application Server is running. Stop Server manually and start installation again. Would you like to retry?". Stop WAY4 Application Server and click [Yes] in the error message window.

To update WAY4 Application Server, perform these actions:

- Copy the archive with the distribution kit of a new WAY4 Application Server version (the file "appserver-<version>-windows-<architecture>.zip") to the computer where it will be installed (for more details on file names and functions, see the section "WAY4 Application Server Distribution Kit").
- Unpack the "appserver-<version>-windows-<architecture>.zip" file containing the installer "WAY4ApplicationServer-<installer_version>-Setup.exe".
- Run the installer "WAY4ApplicationServer-<installer_version>-Setup.exe".

The update process is similar to the installation of WAY4 Application Server under MS Windows (see "Installing WAY4 Application Server under MS Windows"); before installation it is possible to save the configuration of all installed applications and WAY4 Application Server. To do so, check the *Create configuration dump* checkbox in the installation parameter window (see Fig. 17 in the section "Updating WAY4 Application Server under Unix").

ⓘ

> When updating WAY4 Application Server, the web server component (Apache Web Server) is not updated (if it was installed). When updating WAY4 Application Server, in the web server component configuration window, use the Add Apache Web Server to autostart checkbox to indicate whether the web server component must be started automatically when WAY4 Application Server is started.

As a result of updating, WAY4 services will also be reinstalled.

Chapter 5. Deleting WAY4 Application Server

To delete WAY4 Application Server under Unix, superuser (root) privileges are required.

To delete WAY4 Application Server under MS Windows and Unix, do as follows:

- Stop WAY4 Application Server by executing the following command:

```
<USER_HOME>/bin/appsd stop
```

- Delete the service by executing the following command:

```
<USER_HOME>/bin/appsd uninstall
```

As a result, the "WAY4 ApplicationServer" service will be deleted. The WAY4 Application Server directory structure, installed applications, etc. will not be deleted. To restore the remote service, execute the following command:

```
<USER_HOME>/bin/appsd install
```

> ⓘ If it is necessary to completely delete WAY4 Application Server, manually delete all directories containing WAY4 Application Server files. After full deletion of WAY4 Application Server, it is not possible to restore services with the "install" command.

Chapter 6. Configuring WAY4 Application Server

Configuration of WAY4 Application Server and WAY4 applications is split into two categories:

- Configuration that is automatically updated when an upgrade of WAY4 Application Server or WAY4 applications is installed (see "Updating WAY4 Application Server").
- Static configuration that is configured once during installation of WAY4 Application Server or WAY4 applications and is updated manually (for example, WAY4 application port numbers).

Table 2 shows the directories where configuration files are located (for the "<AppServer_HOME>" directory).

Table 2. Location of configuration files

Automatically updated configuration	Static configuration
applications/<Application_Name>/conf /templates	applications/<Application_Name> /conf
conf/templates	conf
conf/webserver/templates	conf/webserver/

WAY4 Application Server is configured by setting the values of parameters in the "<AppServer_HOME>/conf/AppContainer.properties" file. The table Table 3 shows WAY4 Application Server configuration parameters.

Table 3. WAY4 Application Server configuration parameters

Name	Default value	Description
tcp_management_port	32371	The number of the port for receiving management commands from Command Line Utilities.
tcp_internal_port	32372	The number of the port for interacting with the Application Container component.
log_level_container_system	30	Log level. The larger the value, the higher the level and each level includes the information from all previous levels. The following values are possible: "30" – for production systems. "40" – for systems at the acceptance testing stage.
log_file_size	1000000	The maximum size of one WAY4 Application Server log file ("<AppServer_HOME>/logs/container_<file creation date and time>", for more information, see the section "WAY4 Application Server Log Files"). The following values can be specified as measurement units "KB" (kilobytes), "MB" (megabytes), "GB" (gigabytes). Bytes will be used if measurement units are absent.

Name	Default value	Description
log_max_files	1000	Maximum number of WAY4 Application Server log files stored. When a file is created causing the value set by the parameter to be exceeded, the oldest stored file will be deleted.
before_start_cmd		Command that must be executed before starting WAY4 Application Server. For example, to start an executable file or mount a file system. Example of the parameter's use: "before_start_cmd=net use q: ////network_drive//vol3 /user:<user> <password>"
encr_before_start_cmd		Encrypted command that must be executed before starting WAY4 Application Server. Can be used, for example, for password encryption. Data are encrypted using the program "nscipher.exe", located in the directory "<AppServer_HOME>/bin /tools". For encryption, this program must be run, specifying the "ApplicationServer-E55X74D" product code as the parameter. During execution of this program, the user will be asked in dialogue mode to specify and confirm data. Then the encrypted data will be shown on the screen. An example of the parameter's use: "encr_before_start_cmd=B70F05804476663AA5BB0D".
after_stop_cmd		Command that must be executed after stopping WAY4 Application Server.
encr_after_stop_cmd		Encrypted command that must be executed after stopping WAY4 Application Server. Encryption is performed in the same way as described for the parameter "encr_before_start_cmd".
dumpOnDelete	true	When the parameter value is "false", a snapshot will not be made for the application when executing the "delinst" command (see Table 7 of the section " Managing WAY4 Applications").
dumpOnUpdate	true	When the parameter value is "false", a snapshot will not be made for the application during execution of the "updinst" command (see Table 7 of the section " Managing WAY4 Applications").
apps_start_stop_thread_count	1	The number of applications simultaneously started /stopped when starting/stopping WAY4 Application Server, when using the commands "startall" and "stopall" or when using the web console. The default value is equal to half of the number of CPU cores (in multi-core systems); if one processor is used, the value of the parameter will be "1".
		The following values are possible:

Name	Default value	Description
log_clf_enabled	No	"Yes" – log files will be created in the "clf" (Common Log Format) format. For example, this format can be used for WAY4 Health Monitoring module. "No" (default value) – compact format for log files will be used.
start_stop_application_server_timeout	300s	Time interval within which it is necessary to stop WAY4 Application Server. Countdown begins immediately after running the "appsd stop" command (see " Managing WAY4 Application Server"). WAY4 Application Server is stopped in the following order: applications are stopped, the web server component is stopped (if it was installed), Application Container is stopped. If WAY4 Application Server did not stop within this interval, operating system tools will be used to force quit WAY4 Application Server. The parameter value can be changed, for example, if it is known that if the server on which WAY4 Application Server is installed will be down for a certain amount of time in the event of emergency shutdown (limited, for example, by the time an interrupted power source operates, UPS). The value of this parameter must be at least 10 seconds more that "stop_applications_timeout" parameter.
stop_applications_timeout	290s	Time interval within which it is necessary to stop all applications installed on WAY4 Application server. The value of this parameter must be at least 10 seconds less than the value of the " start_stop_application_server_timeout" parameter. If applications were not stopped within this interval, operating system tools will be used to force quit all applications. In this case, resources may be incorrectly freed up, database connections not closed, etc. As the parameter value, it is recommended to specify a time no less than the total value of the " shutdown_timeout" parameter for all applications.
max_installed_applications	100	Maximum number of installed WAY4 applications. When a different value is set, configure the min_managed_port and max_managed_port parameters accordingly.
min_managed_port	15000	Minimum value of the service port number for a WAY4 application.
max_managed_port	15100	Maximum value of the service port number for a WAY4 application.

Apache Web Server

Apache web server version 2.4 is used as the web server component included in WAY4 Application Server.

ⓘ Note that in WAY4 Application Server version 1.7.1402 and later, it is recommended to use the application "<AppServer_HOME>/appserver/applications/apache_24" as the web server, instead of the web server component.

The main function of this application is to optimise work with server content:

- Displaying static content (images, styles, static pages) without using application server tools
- Exchanging dynamic requests with applications running on the application server

This application can also be set up to work as a proxy server (see "Web Server Operating in Proxy Server Mode").

Supporting Additional Apache Modules

The "apache_24" application (or the Apache web server component in WAY4 Application Server versions earlier than 1.7.1402) in addition to standard modules includes additional Apache modules, listed in Table 4.

Table 4. Support of additional Apache modules

Module	Operating system	Description
mod_ssl	IBM AIX, RHEL, Oracle Solaris, MS Windows	A description of the module can be found on the site http://httpd.apache.org/docs/2.4/mod/mod_ssl.html.
mod_fcgid	RHEL, Oracle Solaris	A description of the module can be found on the site http://httpd.apache.org/docs/2.4/mod/mod_cgid.html.
mod_security2	RHEL	A description of the module can be found on the site https://www.modsecurity.org. For more information about configuration, see the section " mod_security Component"
mod_geoip	RHEL, Oracle Solaris	A description of the module can be found on the site http://dev.maxmind.com/geoip/legacy/mod_geoip2.
mod_qos	RHEL	A description of the module can be found on the site: http://opensource.adnovum.ch/mod_qos

Configuring the web server

It is recommended to use the "apache_24" application (Apache web server component in WAY4 Application Server versions earlier than 1.7.1402) for processing requests coming in from the red zone (for example, the internet) on the HTTP/HTTPS protocol.

The "apache_24" application is configured using the file "<AppServer_HOME>/appserver /applications/apache_24/conf/config.properties" with the following parameters.

- "auto_start" – enable automatic start of the "apache_24" application when starting WAY4 Application Server. The following values can be used for the "auto_start" parameter:
  - "yes" or "true" – the "apache_24" application will be started automatically when WAY4 Application Server is started.
  - "no" or "false" – the "apache_24" application will not be started automatically when WAY4 Application Server is started.
  - "recovery" – if WAY4 Application Server crashed and the "apache_24" application was running, the next time WAY4 Application Server is started, the "apache_24" application will also be started.
- "ws_access_max_files" – maximum number of web server "httpd-access.<sequence number>.log" files that are stored (for more information about file purpose, see the section "Web Server Log Files"). When a file is created causing the value set by the parameter to be exceeded, the oldest stored file will be deleted.
- "ws_ssl_access_max_files" – the parameter is similar to "ws_access_max_files", but determines the maximum number of "httpd-ssl-access.<sequence number>.log" files.
- "ws_error_max_files" – the parameter is similar to "ws_access_max_files", but determines the number of "httpd-error.<sequence number>.log" files.
- "ws_ssl_error_max_files" – the parameter is similar to "ws_access_max_files", but determines the number of "httpd-ssl-error.<sequence number>.log" files.
- "modsec_audit_max_files" – maximum number of mod_security log files that can be stored on the disk. If, for example, 50 files have already been created, when the next file is created, the first (by sequence number) file will be deleted. mod_security log files named "modsec_audit.<sequence number>.log" are located in the "<AppServer_HOME>/logs" directory.
- "launcher_log_files_max_file_size" – maximum number of stored httpd-launcher.log files after which the oldest file will be deleted and logging will continue to a new file. When the application is started, information is logged to the httpd-launcher.log until the size indicated in the parameter's value is reached. When this size is reached, a new file is created with a creation date and sequence number starting from "0". The default value is 10 MB.
- "launcher_log_files_max_history" – maximum number of stored httpd-launcher.log files. When a file is created causing the value set by the parameter to be exceeded, the oldest stored file will be deleted. The mechanism for rotating httpd-launcher.log files is implemented in this way. The default value is 60.
- "launcher_log_level" – logging level in httpd-launcher.log files:
  - "FATAL" – only messages about fatal and system errors are saved, whose occurrence prevents the application from working and require it to be restarted.
  - "ERROR" – only information about errors is saved.
  - "LOG" – standard logging level, recommended for high-performance work environments.
  - "INFO" – additional information is stored; for example, messages generated at each step of a cluster's operation.

- "DEBUG" – detailed debugging information is stored. This information may be useful when installing and setting up the application and if errors occur.

The default value is "INFO".

When the Apache web server component is used, parameters are set in the "<AppServer_HOME>/conf/webserver/webserver.properties" configuration file .

Table. 5. Parameters of the webserver.properties configuration file

Parameter	Description	Default value
service_name	Service name.	WAY4 Application Web Server
apache_version	Apache version.	2.4
ws_auto_start	When this flag is set, the web server component will be started automatically when WAY4 Application Server is started. Can be set when the checkbox Add Apache Web Server to autostart is selected during installation of WAY4 Application Server, see Fig. 12 in the section "Installing WAY4 Application Server under Unix". The following values may be used for the "ws_auto_start" parameter: • "yes" or "true" – the web server component will start automatically when WAY4 Application Server is started. • "no" or "false" – the web server component will not start automatically when WAY4 Application Server is started. • "recovery" – if WAY4 Application Server aborted due to failure and the web server component was running, the next time WAY4 Application Server is started, the web server component will also be started.	false
ws_alive_ping_timeout	Interval with which the web server component's status is checked (check of whether the web server component is active).	15s
ws_access_max_files	Maximum number of stored web server component "httpd-access.<sequence number>.log" files (for more information about these files, see the section "Web Server Log Files"). When a file is created	500

Parameter	Description	Default value
	causing the value set by the parameter to be exceeded, the oldest stored file will be deleted.	
ws_ssl_access_max_files	Similar to the "ws_access_max_files" parameter, but defines the maximum number of "httpd-ssl-access.<sequence number>.log" files.	500
ws_ssl_request_max_files	Similar to the "ws_access_max_files" parameter, but defines the maximum number of "httpd-ssl-request.<sequence number>.log" files (by default, these files are not created and are custom files which may contain information on use of "OpenSSL" libraries).	500
ws_error_max_files	Similar to the "ws_access_max_files" parameter, but defines the maximum number of "httpd-error.<sequence number>.log" files.	50
ws_ssl_error_max_files	Similar to the "ws_access_max_files" parameter, but defines the maximum number of "httpd-ssl-error.<sequence number>.log" files.	50
modsec_audit_max_files	Maximum number of mod_security component log files that can be stored on the disk. The default value is "50". If, for example, 50 files have been created, when the next file is created, the first one (according to date and time of creation) will be deleted. "modsec_audit.log.<timestamp>" mod_security component log files are stored in the "<AppServer_HOME>/logs" directory.	50
stop_web_server_timeout	Time interval within which it is necessary to stop the web server component. The web server begins to be stopped either after all applications have been stopped (if before stopping WAY4 Application Server there is enough time exceeding the parameter value), or begins at the time equal to the value of the " stop_application_server_timeout" minus the value of the " stop_web_server_timeout"	20s

Parameter	Description	Default value
	parameter. If the web server was not stopped within this interval, operating system tools will be used to force quit the web server.	
exec_timeout	Timeout to run Apache commands (in milliseconds).	60 000
mime_types_file	Specifies the location of the "mime. types" file. This parameter may be needed if "mime.types" supplied with Apache doesn't contain one of the required extensions. In this case, it is necessary to create a custom "mime.types" file in a non-updatable configuration and use the "mime_types_file" parameter to specify the absolute path to it.	default mime.types

When upgrading Application Server, files from a custom configuration must be merged with template files from the directory "<AppServer_HOME>/conf/webserver/templates" (for more information, see the section " Upgrading Apache 2.2 to Apache 2.4 ").

When upgrading Application Server from a version earlier than 1.6.2162, it is not permitted to switch Apache versions without merging configurations.

Note that when upgrading WAY4 Application Server from a version earlier than 1.6.2566 or earlier than 1.7.1244, the values of all web server component configuration parameters are set in the file "<AppServer_HOME>/conf/webserver/webserver. properties" and have a higher priority than values set in the file "<AppServer_HOME>/conf /AppContainer.properties".

When using the web server under UNIX, if WAY4 Application Server is installed in a directory that differs from "/home/way4", it is necessary to do the following:

```
mkdir -p /home/way4/appserver/apache-2.4 cd /home/way4/appserver/apache-2.4 ln -
s <AppServer_HOME>/apache-2.4/lib /home/way4/appserver/apache-2.4/lib
```

The "apache_24" application (Apache web server component) is configured by editing standard Apache web server parameters (a description of the parameters can be found on the site http://httpd.apache.org/docs). They are located in standard files found in the directory <AppServer_HOME>/appserver/applications/apache_24/conf/webserver" for the "apache_24" application or in the directory "<AppServer_HOME>/conf/webserver" for the Apache web server component:

- "httpd-default.conf" – basic transport level configurations for the web server
- "httpd-main.conf" – main parameters of the web server.

> ⓘ Configuration of the web server component is organised using Name-based Virtual Hosts and is contained in the file "httpd-main.conf". Use of this technology is linked with PCI DSS requirements for the web server component's secure operation (to eliminate vulnerability "CVE-2011-3192"). Additional information about configuring Virtual Hosts for Apache web server can be found on the site http://httpd.apache.org/docs.
>
> For the web server component to operate, the following parameters must be defined in the "httpd-main.conf" file: "NameVirtualHost", "Listen", "ServerName", "ServerAlias", "DocumentRoot", and "Directory". A description of the parameters and examples of their use can be found on the site http://httpd.apache.org/docs/2.2/mod/core.html#documentroot.

- "httpd-ssl.conf" – configurations of secure connection under HTTPS (SSL). The TLS 1.2 protocol is used by default.
- "httpd-vhosts.conf" – configurations for support and maintenance of several domain names. "OpenSSL" is a set of libraries for creating a secure connection under HTTPS (SSL).

```
# Uncomment for enabling OpenSSL
#Include ../conf/webserver/httpd-openssl.conf
```

To set up the parameters for working with standard or custom software modules in PHP, Perl, Python, etc., use configuration files "<config_file_name>.conf" that must be created manually. The configuration files must contain the configurations used by external software modules.

To use configurations from custom configuration files, add the following string in the "httpd-main.conf" file:

```
Include <config_file_name>.conf
```

Configuration files found in the "<AppServer_HOME>/appserver/applications/apache_24/conf/webserver/error" subdirectory for the "apache_24" application or in the "<AppServer_HOME>/conf/webserver/error" subdirectory for the Apache web server component are used to redefine error messages displayed during web server operation.

Configuration files found in "<AppServer_HOME>/appserver/applications/apache_24/conf/webserver/ssl" subdirectories for the "apache_24" application or in "<AppServer_HOME>/conf/webserver/ssl" subdirectories for the Apache web server component are used to set up the storage of certificates necessary to work under SSL/HTTPS protocol.

To change the web page icon that is shown in the browser's address line and also appears next to a bookmark, in tabs and other interface elements:

- Create a file containing the required icon. This file must be named "favicon.ico".

- Place the created "favicon.ico" file in the <AppServer_HOME>/appserver/applications /apache_24/conf/webserver/htdocs" directory for the "apache_24" application or in the "<AppServer_HOME>/conf/webserver/htdocs" directory for the Apache web server component (if necessary, replacing the old file with the new one).
- Restart the browser.

mod_security component

> ⓘ Note that the Apache mod_security component is only installed and configured for Linux.

The mod_security component is a software firewall that protects from attacks directed at web applications, allows monitoring of HTTP traffic and analysis of requests in real time.

> ⓘ For mod_security to operate under RHEL 5, the library "xz-libs" must be installed. This library's archive can be found on the installation disk with the operating system or in the repository containing packages for installing and upgrading the operating system.

To install mod_security, do as follows:

- Install or upgrade WAY4 Application Server.
- Run the script "updateRules.sh" located in the <AppServer_HOME>/appserver/applications /apache_24/app/mod_security" directory for the "apache_24" application or in the "<AppServer_HOME>/ bin/mod_security" directory for the Apache web server component. This script unpacks and copies a set of rules (Core Rule Set, CRS) to the <AppServer_HOME> /appserver/applications/apache_24/conf/webserver/mod_security/crs.original" directory for the "apache_24" application or to the "<AppServer_HOME>/conf/webserver/mod_security /crs.original" directory for the Apache web server component. This rule set can be used for defense from attacks.

  > ⓘ Note that the "updateRules.sh" script can only be run under the user in whose session WAY4 Application Server will be used. If an attempt is made to run the script under the superuser (root), an error message will be displayed.

- An archive containing files with the set of rules (CRS) that are used by mod_security for defence from attacks is located in the <AppServer_HOME>/appserver/applications /apache_24/conf/webserver/mod_security/crs/rules/*.conf" directory for the "apache_24" application. For the Apache web server component, the subdirectory with this archive is located in "<AppServer_HOME>/conf/webserver/mod_security/crs/rules/*.conf". Administrators configure CRS rules according to bank requirements and tasks. If required, data from a directory with an updated set of rules ("<AppServer_HOME>/appserver /applications/apache_24/conf/webserver/mod_security/crs.original" for the "apache_24" application or "<AppServer_HOME>/conf/webserver/mod_security/crs.original" for the

Apache web server component) should be transferred to the working directory used by mod_security ("<AppServer_HOME>/appserver/applications/apache_24/conf/webserver/mod_security/crs/rules" for the "apache_24" application or "<AppServer_HOME>/conf/webserver/mod_security/crs/rules" for the Apache web server component). More information about mod_security and CRS rules is provided on the site https://www.modsecurity.org. If WAY4 Application Server is rarely upgraded, it is recommended to download the latest version of CRS rules from the site https://github.com/SpiderLabs/owasp-modsecurity-crs. If necessary, configure the current set of rules (<AppServer_HOME>/appserver/applications/apache_24/conf/webserver/mod_security/crs/rules" for the "apache_24" application or "<AppServer_HOME>/conf/webserver/mod_security/crs/rules" for the Apache web server component) according to the downloaded version of CRS rules.

Changing the content of HTTP error web pages

To change the external appearance of HTTP error pages generated by the web server component, changes must be made in files located in the <AppServer_HOME>/appserver/applications/apache_24/conf/webserver/mod_security/crs/error" directory for the "apache_24" application or "<AppServer_HOME>/conf/webserver/error" for the Apache web server component and their subdirectories. The standard error page is generated from "HTTP_<ERROR_NAME>.html.var" files, which support several languages and include:

- "include/top.html" – HTML file of the header header
- "include/spacer.html" - HTML file of the spacer
- "include/bottom.html" – HTML file for the footer

The header, spacer and footer are common to all errors.

For example, to change the error message "404 (page not found)", it is necessary to change the file "HTTP_NOT_FOUND.html.var".

The format of the HTTP response file determines the error message for the set of languages. Additional information about file format can be found at: http://httpd.apache.org/docs/2.2/custom-error.html

> ⓘ Pursuant to PCI DSS requirements, WAY4 Application Server / Web Server component versions and other sensitive information may not be published on error pages.

Forwarding Requests from HTTP to HTTPS Protocol

It is possible to forward requests from the HTTP protocol to the HTTPS protocol. Moreover, all HTTP requests can be forwarded or a portion of requests initiated by certain applications.

> ⓘ Note that the "<AppServer_HOME>/conf/webserver/httpd-main.conf" and "<AppServer_HOME>/conf/webserver/httpd-secure-urls.add.conf" file strings shown in the instructions are commented. Therefore to set up request forwarding it is sufficient to delete the comment characters ("#"). It is only necessary to add new strings if they are missing from the specified files. The <ssl_conf_file> is understood to be the "httpd-openssl.conf" file.

- To forward all HTTP requests to the HTTPS protocol, add the following strings to the "<AppServer_HOME>/conf/webserver/httpd-main.conf" file:

```
RewriteEngine On RewriteMap SSLPortMap txt:../conf/webserver/<ssl_conf_file>
RewriteCond %{HTTPS} off RewriteRule (.*) https://%{SERVER_NAME}:${SSLPortMap:
Listen}%{REQUEST_URI}
```

- To forward a portion of HTTP requests from certain web applications, add the following strings to the "<AppServer_HOME>/conf/webserver/httpd-main.conf" file:

```
RewriteEngine On RewriteMap SSLPortMap txt:../conf/webserver/<ssl_conf_file>
Include ../conf/webserver/httpd-secure-urls.add.conf RewriteCond %{REQUEST_URI} ^
/$ RewriteCond %{HTTPS} off RewriteRule (.*) https://%{SERVER_NAME}:${SSLPortMap:
Listen}%{REQUEST_URI}
```

Moreover, the following strings should be added to the "<AppServer_HOME>/conf/webserver/httpd-secure-urls.add.conf" file:

```
RewriteCond %{REQUEST_URI} ^/application1/ [OR] RewriteCond %{REQUEST_URI} ^
/application2/ [OR]
```

In this case, HTTP requests from web applications available at the addresses "<server_name:port>/application1/" and "<server_name:port>/application2/" will be forwarded.

Apache web server in proxy server mode

The "apache_24" application (Apache web server component in WAY4 Application Server versions earlier than 1.7.1402) included in WAY4 Application Server is used in proxy server mode to ensure reliability and load distribution for WAY4U-based solutions.

Fig. 18 shows the load distribution mechanism.

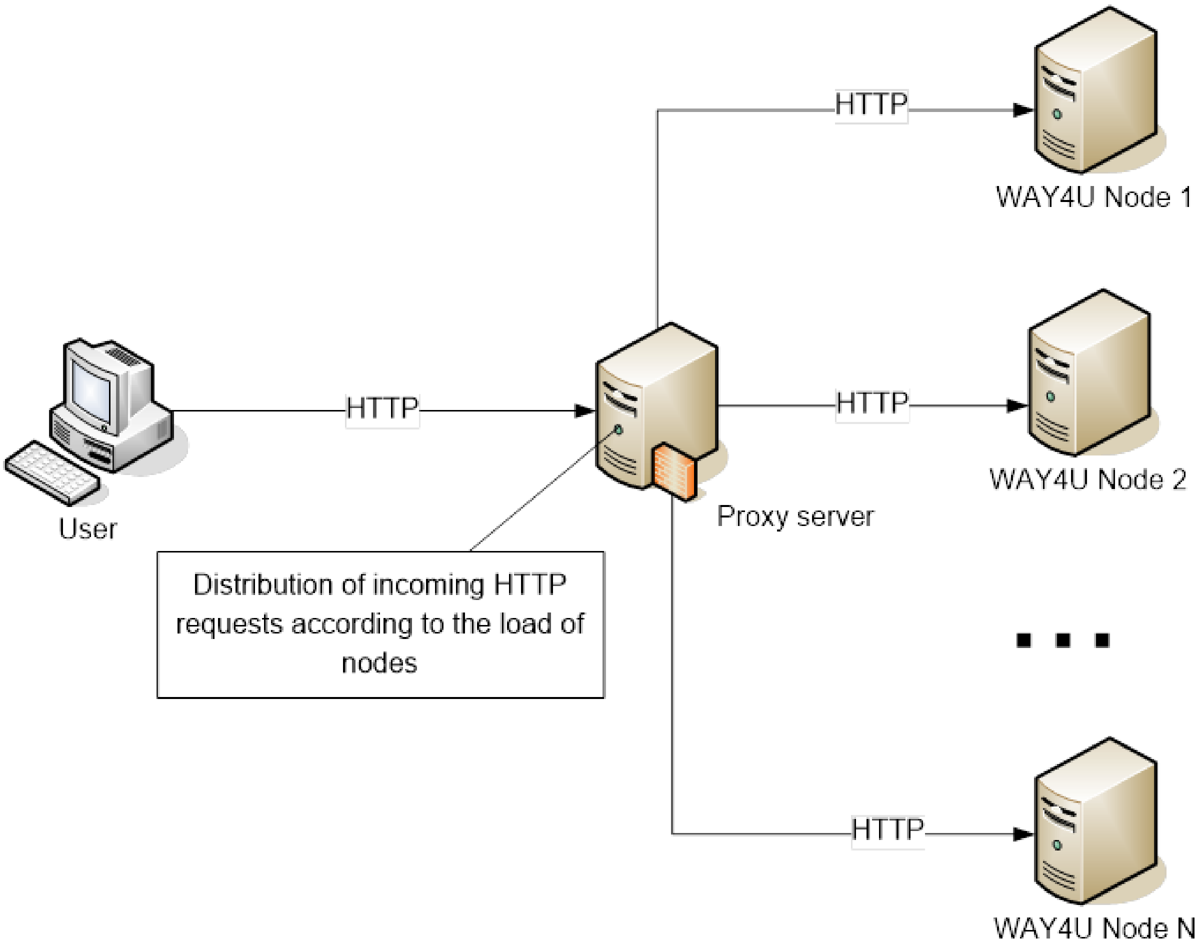


Fig. 18. Load distribution mechanism of the web server operating in proxy server mode

The system implements the mechanism by distributing incoming HTTP requests to several proxy server nodes.

The main features of the load distribution mechanism are as follows:

- The Web server uses the pending request counting algorithm and sends new incoming HTTP requests to the least busy node.
- When one of the nodes is switched off or breaks down and when a node is added or switched on, requests are automatically redistributed among the operating nodes.
- The mechanism supports HTTP and HTTPS.

To use the load distribution mechanism when working with the "apache_24" application, do as follows:

- Install WAY4 Application Server.
- Edit the "httpd-cluster.conf" file found in the directory "<AppServer_HOME>/appserver /applications/apache_24/conf/webserver" (see "Apache web server in proxy server mode").
- Restart the application using the "restart" console utility (see "Managing the Apache Web Server")

To use the load distribution mechanism when working with the web server component, do as follows:

- Install WAY4 Application Server.

- Edit the "httpd-cluster.conf" file found in the "<AppServer_HOME>/conf/webserver" directory (see "Configuring the web server component in proxy server mode").
- Restart the Web server using the "wsrestart" console utility (see "Managing the Web Server").

Note that the web server must be restarted every time the configuration file is modified.

Configuring the web server in proxy server mode

To configure the "apache_24" application in proxy server mode, use the "httpd-cluster.conf" file found in the directory "<AppServer_HOME>/appserver/applications/apache_24/conf /webserver".

To configure the web server component in proxy server mode, use the "httpd-cluster.conf" file found in the directory "<AppServer_HOME>/conf/webserver".

Example of the file:

```
#
# This configuration file reflects Proxy Balancer for Apache HTTP Server.
#

<IfVersion >= 2.4>
  DefaultRuntimeDir ${apache_root}/temp/
</IfVersion>

ProxyPass /balancer-manager !

<Location /balancer-manager>
  SetHandler balancer-manager
  <IfVersion < 2.4>
    Order Deny,Allow
    Deny from all
    Allow from 127.0.0.1
  </IfVersion>
  <IfVersion >= 2.4>
    Require ip 127.0.0.1
  </IfVersion>
</Location>

<Proxy balancer://w4cluster>
  # Add balancer members here
  BalancerMember http://localhost:10100
  BalancerMember http://localhost:10200
  BalancerMember http://localhost:10300
  ProxySet lbmethod=bybusyness
</Proxy>

# This line is required for monitoring WebServer status
ProxyPass /server-status !
ProxyPass /cluster balancer://w4cluster/ nofailover=On
```

To add/remove a node, add/remove the following string to/from the file:

```
BalancerMember http://hostN:portN
```

> After adding/removing a string to/from the configuration file, restart the "apache_24" application or the web server component.

The following string is used to set up access to proxy server resources:

```
ProxyPass /WebAppName balancer://w4cluster/ nofailover=On
```

The default URL address of the resource is "http://proxy_server:proxy_port/WebAppName". Values "proxy_server" and "proxy_port" are specified in web server configurations (see "Web Server"). When it is necessary to change the URL address of the resource, e.g. to "http://proxy_server:proxy_port/", the string will be as follows:

```
ProxyPass / balancer://w4cluster/ nofailover=On
```

> Note that when the root context is used, all requests addressing proxy server resources will be distributed among operating nodes.

Proxy Server's Web Console

The proxy server's web console is used to monitor requests received by the proxy server and to view and edit information on operating proxy server nodes. To access the web console, use the resource "http://proxy_server:proxy_port/balancer-manager".

Recommendations for Firewall Configuration

A firewall must be configured according to functional requirements for the WAY4 Application Server. A firewall is configured for WAY4 applications and the web server.

Table 5 shows the list of firewalls that should be used for various operating systems.

Table 5. Location of configuration files

Operating System	Firewall
AIX	ipfilter
Solaris	ipfilter
Linux	iptables
MS Windows	Windows Firewall

When configuring a firewall, it is recommended to:

- Permit all outgoing connections.
- Permit incoming connections for ports used to provide services:
  - If a web server is used, it is recommended to leave the HTTP port and/or HTTPS port open. The web server and WAY4 applications interact within the WAY4 Application Server (see " Configuring WAY4 Applications for Execution on WAY4 Application Server").
  - If direct access to the application is used (on the TCP/IP protocol), it is recommended to leave the HTTP port and/or HTTPS port (for web applications) open, as well as ports for providing services (for example, TCPServerAdapter ports).
- It is recommended to leave ports used for administration (for example, SSH or FTP) open only for a limited number of machines, from which WAY4 Application Server is administered.

To work with Apache Tomcat, it is necessary to do the following (setup must be made with superuser (root) privileges):

- On Red Hat Enterprise Linux 6 and CentOS 6 operating systems, add the following rule to the " /etc/sysconfig/iptables" file:

```
-A INPUT -m pkttype -j ACCEPT --pkt-type multicast
```

and execute the "service iptables restart" command

- on Red Hat Enterprise Linux 7 and CentOS 7 operating systems, execute the command

```
# firewall-cmd --permanent --direct --add-rule ipv4 filter INPUT 0 -m pkttype --
pkt-type multicast -j ACCEPT
success
# firewall-cmd --reload
```

JMX Interface

WAY4 Application Server supports the Java Management Extensions (JMX) used to manage WAY4 applications.

The JMX interface is used for the following:

- Operation of the system application WAY4 Health Monitoring Gen2 used to gather and process information about the functioning of system applications and components and for sending this information to external monitoring systems.
- Centralised management of applications installed on several instances of WAY4 Application Server.
- Management Web Console ("m2_web_console" system application). For more information, see the document "Management Web Console Installation and Configuration Manual (m2_web_console)".

JMX interface communication parameters are specified in the file "<AppServer_HOME>/conf /AppContainer.properties":

```
jmx_enabled=yes
jmx_port=9999
```

where "jmx_enabled" is a flag that enables (when the value is "yes") support of the JMX interface; "jmx_port" is the number of the port for communication with JMX.

Chapter 7. Configuring WAY4 Applications for Execution on WAY4 Application Server

WAY4 web applications are provided as WAR archives. Back-end applications are supplied as zip archives. To configure WAY4 applications, unpack the corresponding WAR archive.

It is recommended that users create instances of WAY4 applications on the WAY4 Application Server in order to unpack WAR archives (see section "Managing WAY4 Applications"). This will allow archive data to be unpacked automatically to the "<AppServer_HOME>/applications" folder.

During installation, each instance of the back-end application will be assigned a specific port with a default number between "15000" and "15100" (see the description of the min_managed_port and max_managed_port parameters). Each WAY4 application also uses the following ports:

- "32371" – port for receiving commands from command line utilities
- "32372" – port for interaction with Application Container

For correct operation of WAY4 Application Server, it is recommended that these ports not be used. The numbers of the ports can be redefined in the configuration file "<AppServer_HOME>/conf/AppContainer.properties" (see "Configuring WAY4 Application Server").

Fig. 19 shows the use of server WAY4 application ports.

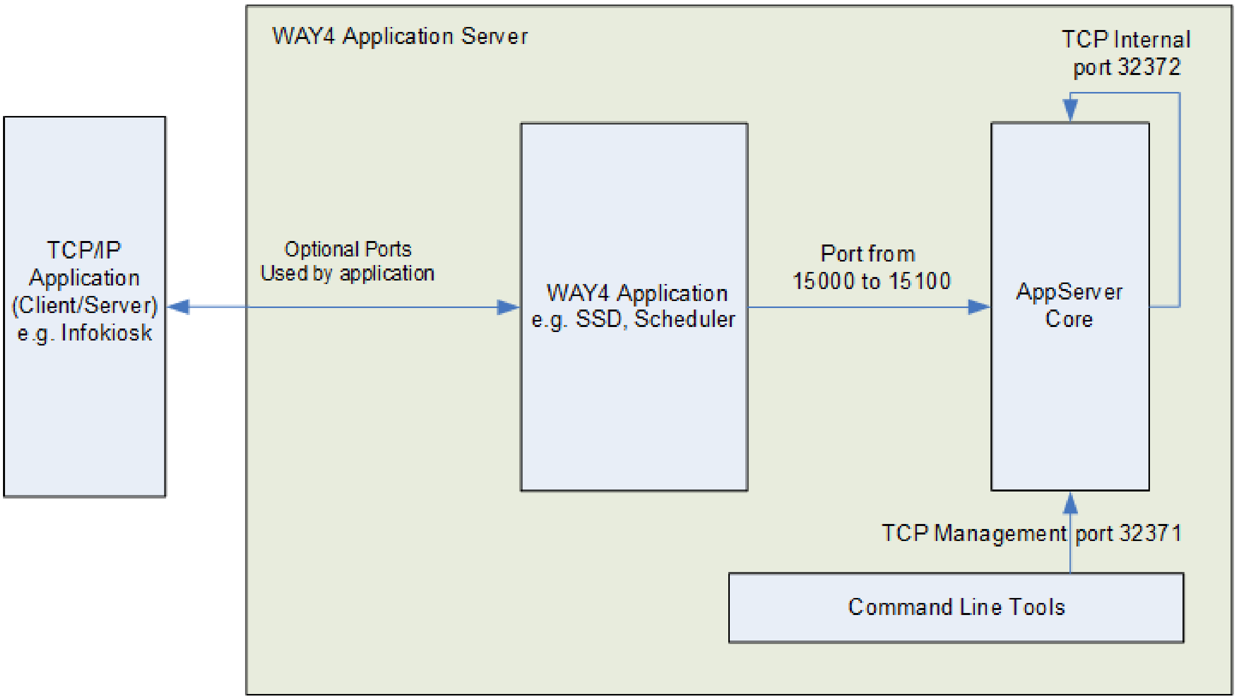

WAY4 Application Server
TCP Internal port 32372
TCP/IP Application (Client/Server) e.g. Infokiosk
Optional Ports Used by application
WAY4 Application e.g. SSD, Scheduler
Port from 15000 to 15100
AppServer Core
TCP Management port 32371
Command Line Tools

Fig. 19. Use of WAY4 application ports

In addition to standard WAY4 application ports, during installation a web application is also assigned an HTTP port with the number <http_port> (the port is specified by users or assigned automatically during installation). Additionally, the web application is also assigned three ports with numbers "<http_port>+1", "<http_port>+2" and "<http_port>+3"; the ports will be used by Apache Tomcat.

Fig. 20 shows the use of web application ports.

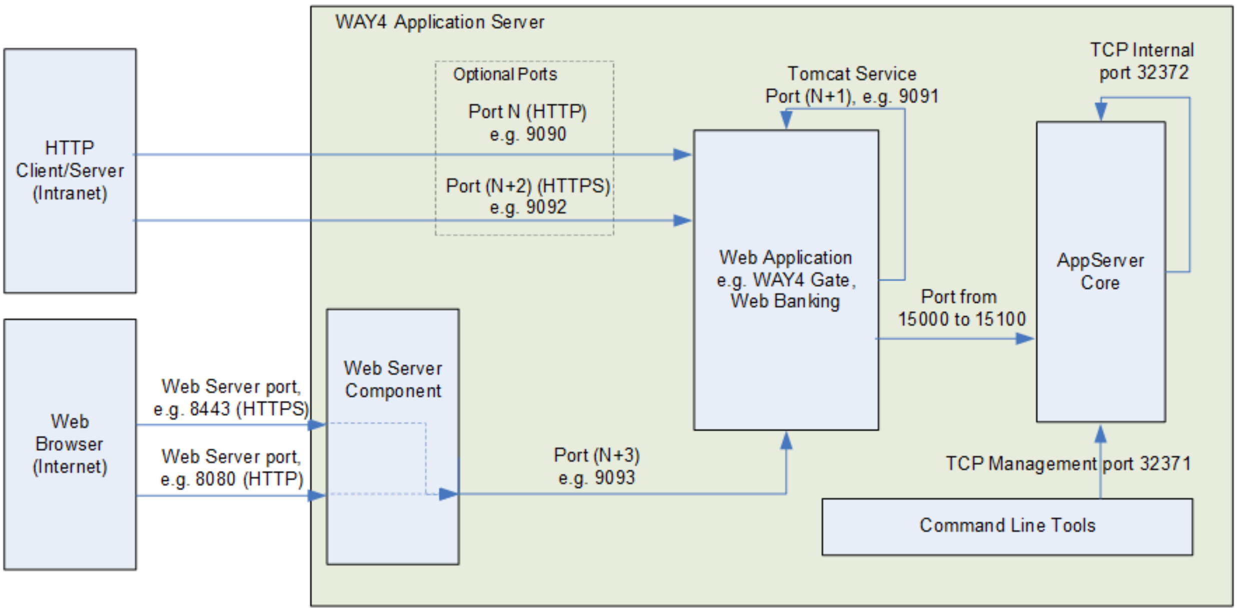

WAY4 Application Server
HTTP Client/Server (Intranet)
Web Browser (Internet)
Optional Ports
Port N (HTTP) e.g. 9090
Port (N+2) (HTTPS) e.g. 9092
Tomcat Service Port (N+1), e.g. 9091
TCP Internal port 32372
Web Application e.g. WAY4 Gate, Web Banking
AppServer Core
Port from 15000 to 15100
Web Server Component
Web Server port, e.g. 8443 (HTTPS)
Web Server port, e.g. 8080 (HTTP)
Port (N+3) e.g. 9093
TCP Management port 32371
Command Line Tools

Fig. 20. Use of web application ports

ⓘ Note that it is necessary to restart the web server after installing a web application.

WAY4 applications are configured by indicating parameter values in files located in the "<AppServer_HOME>/applications/<Application_Name>/conf" directory.

Configuring Application Common Parameters

The main WAY4 application configuration file is the "<AppServer_HOME>/applications /<Application_Name>/conf/config.properties" file. Through this file, it is possible to set values for these parameters:

- "auto_start" – parameter indicating whether the WAY4 application will automatically start when the WAY4 Application Server is started. This following values may be used for this parameter:
  - "yes" or "true" – the WAY4 application will automatically start when the WAY4 Application Server starts.
  - "no" or "false" – the WAY4 application will not automatically start when the WAY4 Application Server starts.
  - "recovery" – if the WAY4 Application Server shut down unexpectedly while the WAY4 application was running, the WAY4 application will start when the WAY4 Application Server restarts.

- "startup_timeout" – parameter indicating the timeout interval for WAY4 application start phase (from the start of the OS run process until the application is ready to accept requests). If the application does not start before timeout (process freezes on startup), the operation system process will be forcibly stopped and the WAY4 Application will attempt to restart it. If this is repeated, the system will make attempts to restart the WAY4 application only after a series of increasing time intervals (100ms, 10s, 1m, 3m, etc.). The default value is "120s".

- "shutdown_timeout" – parameter defining the time interval during which the WAY4 Application Server will attempt to normally shut down a running WAY4 application. The default value is "60s".
- "startup_order" - number that determines the order in which an application will be started. The default value is "0".
- "exec_on_ping_failed" – troubleshooting command used for diagnostics when an application has hung. Example of the command for MS Windows:

```
exec_on_ping_failed=cmd /c call C://WAY4ApplicationServer//appserver//bin//diag.
bat type=java_app app_name=app1
```

- "exec_on_ping_failed_timeout" – parameter determining the time interval during which WAY4 Application Server will wait for execution of the "exec_on_ping_failed" command. The default value is 60 seconds. For example:

```
exec_on_ping_failed_remoteServiceAPItimeout=50s
```

- "jvm_params" – parameter defining properties for a Java machine in this format: "jvm_params=-Xmx<integer>m -Xms<integer>m ", for example, "jvm_params=-Xmx512m -Xms128m" where:
  - -"Xmx" – maximum memory allowed for the WAY4 application
  - -"Xms" – maximum memory allocated for the WAY4 application during startup
- "jvm_initial_heap_size" – parameter redefining the memory (-Xms) allocated for a WAY4 application when it is started, set in another configuration file (the order for redefining and merging the values of the parameter that determines Java machine properties is described below).
- "jvm_maximum_heap_size" – parameter redefining the maximum memory (-Xmx) that a WAY4 application can use, set in another configuration file (the order for redefining and merging the values of the parameter that determines Java machine properties is described below).
- "jvm_heap_dump=<...>" – parameter responsible for saving JVM heap dumps when "OutOfMemoryError" errors occur (the default value is "false"):

  - If no value is set for the parameter, the mode for saving JVM heap dumps is enabled, and default parameters are used (the set of parameters depends on the platform on which WAY4 Application Server is installed). Dumps will be saved in the directory "<AppServer_HOME>/applications/<Application_Name>/logs". The maximum number of dumps is determined by the "max_heap_dumps" parameter.
  - Parameters for saving JVM heap dumps can be specified as the value. For example, to redefine the directory to which a dump should be saved, "jvm_heap_dump=-XX:+HeapDumpOnOutOfMemoryError -XX:HeapDumpPath=<AppServer_HOME>/applications/<app_name>/logs2" can be specified. In this case, dumps will be saved to the "log2" directory. Note that if the directory is redefined, the "max_heaps_dump" parameter is not considered.
  - If "no" or "false" is specified as the parameter value, JVM heap dups are not saved.

> To comply with PA DSS requirements, automatic creation of JVM Heap Dump must be disabled. Ensure that the value of the "jvm_heap_dump" parameter is "false" or "no":

```
jvm_heap_dump = false
```

- "jvm_statistics_period" - parameter defining the period of time between checks of jvm statistics. If errors occur, they are written to a log file. The default value is 60 seconds.
- "stat_count" - parameter defining the number of periods (jvm_statistics_period) after which jvm statistics are written to a log file, even if there were no errors. The default value is "10".
- "webserver_enabled" – specifies whether this application will be available through the web server. By default the parameter's value is "yes".
- "webserver_context" – URL address part (context) used to identify this application through the Apache web server (by default, this is the same as the application context in Apache Tomcat, and, in turn, by default, corresponds to the application name).

> To identify the application, use the "web_context" parameter (for more information, see Managing WAY4 Applications"), or use the same values for the "webserver_context" and "web_context" values. The value of the "webserver_context" parameter should only be set if access to the web application through the Apache web server at an URL address that differs from the value of the application's "web_context" parameter is required.
>
> When the "webserver_context" parameter is used, careful attention should be paid to the following: the Apache web server forwards requests to applications running in the Apache Tomcat servlet container. The "webserver_context" parameter determines which URL address will be used for access to the application through the Apache web server, "web_context" sets the context of the application in the Apache Tomcat servlet container. The value of these parameters may differ.
>
> An application deployed in the Apache Tomcat servlet container has a context and by default it corresponds to the name of the application.

If the values of the "webserver_context" and "web_context" parameters do not match, this may lead to system failure.

- "webserver_secure" – when this flag is set, the application is only available using the HTTPS protocol.
- "webserver_secure_port" – the number of the HTTPS port to which client requests are redirected if the "webserver_secure" flag is set and the application is accessed on the HTTP protocol. By default, the port value corresponds to the application's HTTPS port.
- os_env=<name1>\=<value1>,<name2>\=<value2>,..." – list of environment variables and their values defined for this application.

If the values of system variables are used that contain paths to files or directories (for example, "PATH", "LD_LIBRARY_PATH"), delimiters must be used:

- ";" – for Windows (for example, "os_env=PATH\=C:\drivers\win32;c:\windows, WORK_DIR\=work);
- ":" – for Unix platforms (AIX/Linux/Solaris, for example, "os_env=PATH\=/opt/freeware:/usr/bin, WORK_DIR\=work").

Note that if the "/" character is used in a variable value, this character should be escaped with the "/" character. For example, for the environment variable "PATH=C:/Program Files/Folder", the parameter value should be specified as "os_env=PATH=C://Program Files//Folder". Moreover, characters whose encoding differs from ASCII may be used as variable values. In this case, specify the value in the format "/u<four-digit code> (Unicode), for example "/u0020" for a space character.

The "=" character in assignment of variable values should be escaped with the "\" character, (for example: key1\=value1:key2\=value2).

- "copy_managed_jars" – when this flag is set, WAY4 Application Server system libraries will be copied to the directory "<AppServer_HOME>/applications/<Application_Name>/WEB-INF /lib" during installation of the application.
- "default_snapshot_logs_max_age" – time interval for which log files will be included in an application snapshot (the interval is calculated from the date and time a snapshot is made). The following values can be used as measurement units: "s" (seconds, used by default), "m" (minutes), "h" (hours), "d" (days), and "w" (weeks).

For example, when "default_snapshot_logs_max_age=3600" is specified, an application snapshot will include files created for the last 3600 seconds, and when "default_snapshot_logs_max_age=28h" – for the last 28 hours.

- "default_snapshot_logs_max_size" – total size of log files that are included in an application snapshot. The following values can be used as measurement units: "k" (kilobytes), "M" (megabytes), "G" (gigabytes). When no measurement unit or "b" is specified, bytes will be used. Application log files are selected in lexographic order, if when a file is added, the total value exceeds the parameter value, the file will not be included in the snapshot, while all previous files will be included.

  For example, if "default_snapshot_logs_max_size=100000" is specified, the total size of log files included in a snapshot will not exceed 10000 bytes, and when "default_snapshot_logs_max_size=100M" – 100 megabytes.

- "ajp_connection_settings" – parameter that makes it possible to change settings for the Apache web server AJP connection with the application. Example:

```
ajp_connection_settings=keepalive=Off retry=10 timeout=4 # will disable
keepalive for AJP connections # response timeout will be equal to 4 seconds #
Apache Web Server will not forward any requests to application for 10 seconds
after failed responce (http code 503 or responce timeout)
```

System parameters (JVM Args) can be set for specific java applications or for all installed java applications at the same time. Parameters are specified in the configuration file "<AppServer_HOME>/conf/AppGlobalConfig.properties" in the format "<app_name>.<parameter>=<value>", where <app_name> is the application name (the "*" character can be used to specify the parameter for all applications), <parameter> is the parameter name, and <value> is the parameter value. For example, the parameter "*.javax.net.debug=all" enables SSL debugging for all applications.

In the "<AppServer_HOME>/conf/AppGlobalConfig.properties", the "jvm_mon_enabled" parameter can be used to disable logging jvm statistics (by default, logging is enabled). It is also possible to disable logging jvm statistics separately for each application in the "<AppServer_HOME>/applications/<Application_Name>/conf/config.properties" configuration file.

In the "<AppServer_HOME>/conf/AppGlobalConfig.properties" configuration file, the "rest_api_enabled" parameter can be used to disable the ability to use the management web console or the REST API interface (by default "rest_api_enabled=true").

WAY4 Application Server uses Apache Tomcat as a web application servlet container. Its specially parameterised distribution package is provided together with WAY4 Application Server.

The value of the parameter defining Java machine properties ("jvm_params") is merged from values specified in the files "<AppServer_HOME>/applications/<Application_Name>/system/config.properties" and "<AppServer_HOME>/applications/<Application_Name>/conf/config.properties". "-D" and "-XX" parameters, and some "-X" can be redefined in the file "<AppServer_HOME>/applications/<Application_Name>/conf/config.properties" without copying the entire value from the file "<AppServer_HOME>/applications/<Application_Name>/system/config.properties".

The order for redefining parameters that specify Java machine properties ("jvm_params" and "system_jvm_params"):

- "<AppServer_HOME>/conf/AppGlobalConfig.properties".
- "<AppServer_HOME>/applications/<Application_Name>/system/config.properties", "system_jvm_params" parameter.
- "<AppServer_HOME>/applications/<Application_Name>/system/config.properties", "jvm_params" parameter.
- "<AppServer_HOME>/applications/<Application_Name>/conf/config.properties", "system_jvm_params" parameter.

- "<AppServer_HOME>/applications/<Application_Name>/conf/config.properties", "jvm_params" parameter.

Accordingly, the "jvm_params" parameter set in the file "<AppServer_HOME>/applications /<Application_Name>/conf/config.properties" has the highest priority.

ⓘ WAY4 Application Server caches application parameters the first time the application is started and the next time it is started merges the cached values with the values from config.properties. For an application that is running, commenting parameters in config. properties (that are not set in other places) doesn't clear the last value that was set before restarting WAY4 Application Server.

Configuring Web Applications

WAY4 web applications are provided as WAR archives.

The recommended method for unpacking WAR archives is to create instances of WAY4 applications on WAY4 Application Server (see the section "Managing WAY4 Applications"), which results in these archives being automatically unpacked to the "<AppServer_HOME> /applications" directory.

The main configuration file for a WAY4 web application is the file "<AppServer_HOME> /applications/<Application_Name>/conf/config-web.properties", which is referred to each time before the web application is started; it can be used to set the values of the following parameters:

- "container_type" – container used for servlets. Possible values:
  - TOMCAT_V8
  - TOMCAT_V7

The default value is "TOMCAT_V8".

- "web_context" – part of the URL address (context), used to identify this application through the Apache web server (by default it matches the context of the application in Apache Tomcat, which, in turn, by default matches the application's name).
- "http_port" – HTTP port of the Apache Tomcat web server. This value is set when the application is created.
- "server_port" – TCP/IP port for managing the Apache Tomcat web server. The default value is "server_port=<http_port+1>".
- "http_disable" – this parameter can be used to disable getting requests to the application on the HTTP port. The default value is "no".
- "ciphers" – ciphers allowing the use of OpenSSL syntax.
- "https_port" – HTTPS port of the Apache Tomcat web server. The default value is "https_port=<http_port+2>".

- "key_store_file" – path to the keystore file that is used by Java applications for encryption, authentication and installation of HTTPS connections. The default value is "conf/security /appkeystore.jks".
- "key_store_password" – password for reading the contents of the keystore file.
- "key_alias" – alias for the key that is used from the keystore file.
- "trust_store_file" – path to the certificate store file.
- "trust_store_password" – password for the certificate store file.
- "https_disable" – this parameter can be used to disable getting requests to the application on the HTTPS port. The default value is "yes".
- "user_database_file_path" – path to the file with user credentials that is used to authenticate a user. The default value is "conf/security/tomcat-users.xml".
- "user_database_realm_type" – type of DB Realm user. Possible values:
  - "STANDARD" – it is not recommended to set this type, used only for backward compatibility. PCI DSS compliance will be lost if this type is specified.
  - "EXPIRING" – type of user, with the time the password is effective specified.

The default value is "EXPIRING".

- "user_database_password_expiration_timeout" – time (number of days) the user's password is effective. Used if "user_database_realm_type=EXPIRING". After the timeout, if the password was not changed during the effective period, authorisation will be prohibited. The default value is "90".
- "user_database_password_expiration_warning" – number of days after which the user will get a warning about the password's expiration. When the threshold is reached for the warning about expiration of the user's password, the warning is logged in the Apache Tomcat container log each time the user is authenticated. The default value is "15".
- "cluster_enabling" – enable an Apache Tomcat cluster. The default value is "no".
- "cluster_membership_address" – broadcast address together with the port number (cluster_membership_port) determines the group of servers or network used to build a cluster group consisting of instances of web applications that are running (Apache Tomcat container servlet).
- "cluster_membership_port" – port for broadcast transmission of data.
- "cluster_receiver_port" – port for listening for Apache Tomcat cluster incoming data.
- "ajp_port" – Apache Tomcat AJP port. The default value is "ajp_port=<http_port+3>".
- "container_access_log" – enable or disable the container access log. The default value is "false".
- "xml_attribute" – XPath expression for changing, adding, or deleting an attribute in the Apache Tomcat "server.xml" file (for more information, see "Configuring Apache Tomcat").
- "web_xml_content_file_location" – path to the file which content will be added to the generated "web.xml" file. Should be used for customising generated "web.xml file" (for more information, see "Configuring Apache Tomcat" section "Changes in the content of HTTP error web pages").

In the file "<AppServer_HOME>/applications/<Application_Name>/conf/config-web.properties", it is also possible to redefine the values of the "sslProtocol" and "sslEnabledProtocols" parameters from the "server.xml" file (for a description of the parameters see https://tomcat.apache.org /tomcat-7.0-doc/config/http.html). To do so, add these parameters to the file "<AppServer_HOME>/applications/<Application_Name>/conf/config-web.properties" and change their values. By default, in the "server.xml" file, the value TLSv1.2 is specified for both parameters. When redefining parameter values, several values can be specified, separated by commas. If a version is not specified for TLS, TLS version 1.0 (TLSv1.0) will be used by default.

Example:

```
sslProtocol=TLSv1.2 sslEnabledProtocols=TLSv1.1,1.2,1.3
```

To apply the new values of the "sslProtocol" and "sslEnabledProtocols" parameters, restart the WAY4 web application.

The file "<AppServer_HOME>/applications/<Application_Name>/conf/config.properties" is used to initialise an application; it can be used to specify the values of the following parameters:

- "auto_start" – parameter specifying whether a WAY4 application will automatically be started when WAY4 Application Server is started. The following values can be used as the parameter's value:
  - "yes" or "true" – the WAY4 application will automatically be started when WAY4 Application Server is started.
  - "no" or "false" – the WAY4 application will not automatically be started when WAY4 Application Server is started.
  - "recovery" – if WAY4 Application Server shut down due to failure and the WAY4 application was running, the next time WAY4 Application Server is started, the WAY4 application will also be started.
- "jvm_params" – parameter defining properties for a Java machine in this format: "jvm_params=-Xmx<integer>m -Xms<integer>m ", for example, "jvm_params=-Xmx512m -Xms128m" where:
  - "-Xmx" – maximum memory allowed for the WAY4 application.
  - "-Xms" – maximum memory allocated for the WAY4 application during startup.
- "jmx_port" – number of the port for communication with JMX. The default value is "9992".
- "auth_type" – the parameter is used to configure two-factor authentication, it sets the priority for determining a user's role. The parameter can have one of the following values:
  - "cert" – the user's role is determined by the selected certificate on authentication in the web console.
  - "password" – the user's role is determined by values entered in the "user" and "password" fields in the web console authentication window.

- "appcontainer_extension" – if the "true" value is set for the parameter, the list of available application services (for example, WAY4 Transaction Switch services or WAY4 NetServer channels) will be shown in the web console ("console" or "m2_web_console" application) or will be available when working in the command line ("m_tools" application).

- "log_level" – logging level in "m_agent" application log files:
- "10" (FATAL) – only messages about fatal and system errors making it impossible for the application to continue operating and requiring restart of the application are saved.
- "20" (ERROR) – only information about errors is saved.
- "25" (WARNING) – warnings and error messages are saved.
- "30" (LOG) – standard logging level, recommended for high performance operating environments.
- "35" (LOG_MORE) – standard logging level, recommended for operating environments, this is the default value.
- "40" (INFO) – additional information is saved, for example, messages generated at each stage of the cluster's operation. Performance may be significantly decreased when this value is set.
- "50" (DEBUG) – detailed debugging information is saved. This information may be useful when installing and configuring applications and if errors occur. Performance may be significantly decreased when this value is set.
- "60" (DEBUG_MORE) – all detailed information is saved. This information may be useful when analysing reasons for errors. Performance may be significantly decreased when this value is set.

The default value is "35".

- "log_max_files" – maximum number of log files after which the oldest file will be deleted and a new file will be used for logging. The default value is "100".

Configuring Apache Tomcat

The Apache Tomcat version that is used and its port are specified in the file "<AppServer_HOME>/applications/<Application_Name>/conf/config-web.properties" on the web application level (for more information, see the section "Configuring WAY4 Applications for Execution on WAY4 Application Server").

Configuring "server.xml"

When a web application is started, the "server.xml" file is automatically generated in the directory "<AppServer_HOME>/applications/<Application_Name>/temp/container/conf".

If necessary, an attribute of an XML element that was generated for the "server.xml" file can be added, changed, or deleted using the "xml_attribute" parameter of the file "<AppServer_HOME>/applications/<Application_Name>/conf/config-web.properties".

Syntax for the "xml_attribute" parameter:

```
xml_attribute.<label>.<attribute name>=<new attribute value>|<XPath expression fo
r attribute element>
```

where:

- "<label>" – label to distinguish the operation for the XML attribute in the "config-web.properties" file.
- "<attribute name>" – name of the attribute whose value must be changed.
- "<new attribute value>" – new value. A prefix must be used:
  - "+" – add the attribute
  - "-" – delete the attribute
  - "" – change the attribute
- "<XPath expression for attribute element>" – XPath expression for the element that must be processed.

Examples:

- Change the value of the "maxThreads" attribute to "1000" for the "Connector" element configured for port "24200":

```
xml_attribute.max-threads.maxThreads=1000|/Server/Service/Connector[@port='24200'
]
```

- Delete the "redirectPort" attribute from the "Connector" element configured for port "24200":

```
xml_attribute.redirectPort.redirectPort=-|/Server/Service/Connector[@port='24200'
]
```

- Add the "redirectPort" attribute with the value "8888" for the "Connector" element configured for port "24200":

```
xml_attribute.redirectPort.redirectPort=+8888|/Server/Service/Connector[@port='24
200']
```

Changes in the content of HTTP error web pages

When a web application is started, the "web.xml" file is automatically generated in the directory "<AppServer_HOME>/applications/<Application_Name>/temp/container/conf".

To change the external appearance of HTTP error pages generated by Apache Tomcat, the following settings must be made, for example, for the error "404 (page not found)":

- Create an XML file with the following information:

```
<error-page>
```

```
        <error-code>404</error-code>
        <location>/errorpages/404.html</location>
    </error-page>
```

- Using the "web_xml_content_file_location" parameter of the file "<AppServer_HOME>/applications/<Application_Name>/conf/config-web.properties" specify the path to the file which content will be added to the generated "web.xml" file Apache Tomcat.
- Create a "404.html" file with the required HTML content in directories:
  - "<AppServer_HOME>/applications/<Application_Name>/webapps/ <Application_Name>/errorpages"
  - "<AppServer_HOME>/applications/<Application_Name>/webapps/ ROOT/errorpages"
- Restart the application.

Dynamic pages (for example ".jsp" or servlets) may be used instead of static html pages specified in these files.

> Pursuant to PCI DSS requirements, WAY4 Application Server / Web Server component versions and other sensitive information may not be published on error pages.

Configuring web application log files

Log file configuration parameters are shown in the file "<AS_HOME>/appserver/applications/<app>/temp/container/conf/logging.properties" that is created when an application is started. This file cannot be changed. To change these parameters, create a copy of the "logging.properties" file and in the file "C:\WAY4ApplicationServer\appserver\applications\<app>\conf\config-web.properties", specify the full path to this file in the "_custom_configuration_files" parameter.

```
_custom_configuration_files=</path_to_logging.properties>
```

Name	Default Value	Description
java.util.logging. FileHandler.limit	10485760	Maximum size of one web application log file. The following values can be specified as a measurement unit: "KB" (kilobytes), "MB" (megabytes), "GB" (gigabytes). If no measurement unit is specified, bytes will be used.
java.util.logging. FileHandler.count	25	Maximum number of web application log files that are stored. When a file is created whose number exceeds the value set by the parameter, the oldest file in the log is deleted.

Name	Default Value	Description
handlers	java.util.logging.FileHandler, java.util.logging.ConsoleHandler	List of all handlers that are used.
.level	INFO	The parameter determines the maximum logging level for all handlers. For possible values, see https://docs.oracle.com/javase/8/docs/api/java/util/logging/Level.html
java.util.logging.FileHandler.*		Output file handler parameters. For more information, see https://docs.oracle.com/javase/8/docs/api/java/util/logging/FileHandler.html
java.util.logging.ConsoleHandler.*		Error handler parameters. For more information, see https://docs.oracle.com/javase/8/docs/api/java/util/logging/ConsoleHandler.html

To increase the debug level for a web application, change the values of the following parameters:

- ".level"
- "java.util.logging.FileHandler.level"

Chapter 8. Managing WAY4 Application Server

Managing WAY4 Application Server under Unix

WAY4 Application Server is managed through the "appsd" script, located in the standard setup in the "<USER_HOME>/bin" directory.

Through this script, it is possible to run commands by setting them as parameters as follows:

```
<USER_HOME>/bin/appsd <command>
```

The following commands are used:

- "start"– start WAY4 Application Server.
- "stop" – stop WAY4 Application Server. The time required to stop WAY4 Application Server, applications, and the web server component are defined by the parameters " stop_application_server_timeout", "stop_applications_timeout", and " stop_web_server_timeout".
- "forcestop" – force WAY4 Application Server to stop.
- "restart" – restart WAY4 Application Server.
- "console" – start WAY4 Application Server in the current console.
- "status" – receive a list of installed applications, application and WAY4 Application Server statistics. The format "status [format=<values>]" can also be used for this command. If the value "format=<value>" is not specified (i.e., the "appsd status" command is not used), the following information will be shown for each application: application name, status, version number, number of restarts, and the number of the HTTP port used by the application. Values of the "format=<values>" parameter are shown in Table 6.

Table 6. Values of the "status" command

Column name and description	Parameter abbreviated value	Parameter full value
"Name" – application name.	None (information is always provided)	None (information is always provided).
"Status" – application status. A WAY4 application may be in one the following statuses: "ON" – the application is running. "off" – the application has been stopped.		

Column name and description	Parameter abbreviated value	Parameter full value
"starting" – the application is starting. "stopping" – the application is stopping. "restarting" – the application is running, but WAY4 Application Server it trying to restart it. "CRASH" – the application is not running due to an error in the application's behaviour (for example, the application needs a connection with the database, but the database is unavailable). "corrupted" – a directory with the name of the application has been created, but files required to start the application are missing.	None (information is always provided)	None (information is always provided)
"Version" – application version number.	v	version
"Restarts" – number of application restarts.	R	restarts
"Start Request" – application start request date and time.	r	start_request
"Started" – application start date and time	s	started
"Heap Mem" – RAM used by the application.	m	memory
"HTTP" – number of the HTTP port used by the application (only specified for web applications).	h	http
"HTTPS" – number of the HTTPS port used by the application (only specified for web applications).	t	https
"Web Context" – application context. The value is set by the the "web_context" parameter on installation or upgrade of the application (for more information, see the section "Managing WAY4 Applications").	w	web_context
Output full information (all columns)		all

Note that when the full values of parameters are specified, commas must be used as delimiters.

Example. The command

```
/appsd status format=vRr
```

is the same as the command

```
/appsd status format=version,restarts,start_request
```

When the command is executed, the application's name and status will be shown, as well as the version, number of restarts, and start request date and time. Command result:

```
Appserver Service is installed
WAY4 Application Server (version 1.6.98) status:
+---------------------------------------------------------------------------+
| Name                  | Status      | Version        | Restarts | Start Request |
+-------------------+------------+--------------+----------+----------------+
|apache_24               |          ON | 1.7.815       | 0         | 15-04-06 15:15|
|container            |          ON |               | 0          | 15-04-06 15:15|
|logagent             |          ON | 2.0.593       | 0          | 15-04-06 15:15|
|monitoring           |          ON | 1.0.679       | 0          | 15-04-06 16:02|
|scheduler_WS         | off         | 1.1.31-706    | 0          |                |
|way4gate1            |          ON | 2.6.295       | 0          | 15-04-06 15:16|
|console              |          ON | 1.6.50        | 0          | 15-04-06 15:16|
|wsruntime_WS         |          ON | 1.2.48        | 0          | 15-04-06 15:16|
+---------------------------------------------------------------------------+
```

- "install" – create a service with the name "WAY4ApplicationServer" that will start and stop WAY4 Application Server when the operating system is started and stopped.
- "uninstall" – delete the service with the name "WAY4ApplicationServer".
- "version" – receive a list of WAY4 Application Server installed component current versions.

"systemd" is used to manage RHEL 7 operating system services. Therefore, WAY4 Application Server is started, restarted and stopped under the superuser (root) with one of the following commands:

```
service WAY4ApplicationServer <command>
```

or

```
systemctl <command> WAY4ApplicationServer
```

Here "WAY4ApplicationServer" is the name of the service, "<command>" is one of the following commands: "start", "restart", and "stop".

In test mode (should not be used in production) for RHEL 7 it is possible to start, restart, and stop WAY4 Application Server under users included in the group "way4". To do so, use the utility "appsd" described earlier in this section, Note that use of the utility may lead to an incorrect state of the "WAY4ApplicationServer" service.

Managing WAY4 Application Server under MS Windows

WAY4 Application Server is managed in one of the following ways:

- Using the operating system control panel ("Start => Control Panel => Administrative Tools => Services"). To start or stop WAY4 Application Server, use the "WAY4 Application Server" service.

> ⓘ When the "WAY4 Application Server" service is stopped (including when restarting /switching off the server), the total time required to shut down the service does not exceed 2 minutes (operating system limitation). Therefore, for applications to be shut down properly, ensure that the total value of "shutdown_timeout" parameters for all applications does not exceed 2 minutes. Instead of stopping the service from the operating system's console, WAY4 Application Server can be stopped using the "appsd.bat" console utility, that does not have a two-minute limit for stopping during execution.

> ⓘ The "WAY4 Application Web Server" service is managed by WAY4 Application Server main component. Therefore, it is not recommended that the service be started and stopped manually.

- Using the "appsd.bat" utility available in the "<USER_HOME>/bin" directory. This utility and commands are similar to the "appsd" script (see "Managing WAY4 Application Server under Unix").To start WAY4 Application Server under a user who does not have administrator privileges, do as follows:
- Install WAY4 Application Server (see "Installing WAY4 Application Server under MS Windows") under the name of a user who has administrator privileges.
- Register the user who does not have administrator privileges (for example, "way4").
- Grant the user "way4" full access privileges (reading, writing, etc.) to the directory where WAY4 Application Server is installed ("<AppServer_HOME>").
- Grant the user "way4" privileges to manage "WAY4 Application Server" and "WAY4 Application Web Server" services (start, restart, stop). To do so, follow the instructions given on the site http://support.microsoft.com/kb/325349.
- In "WAY4 Application Server" and "WAY4 Application Web Server" service properties, specify the user "way4" in the "Log On" tab.
- Start WAY4 Application Server under the user name "way4".

Chapter 9. Managing the Apache Web Server

In WAY4 Application Server versions earlier than 1.7.1402, the Apache web server component is used, in WAY4 Application Server version 1.7.1402 and later, it is recommended to use the "apache_24" application.

Managing the "apache_24" application

The "apache_24" application is managed with console utilities found in the "<AppServer_HOME> /appserver/applications/apache_24" directory:

- "dump" – save the application's logs, configuration, snapshot (for more information, see " Managing WAY4 Applications").
- "restart"– restart the application.
- "security_conf" – enable/disable the mode for forwarding requests from HTTP to HTTPS (for more information, see "Managing WAY4 Applications").
- "start" – start the application.
- "status" – show information about the application's operation.
- "stop" – stop the application

> ⓘ Note that when Apache web server module settings are changed, use the "stop" and then the "start" command instead of the "restart" command to restart the "apache_24" application. The "restart" command should be used if a new application was added and it must be made available through "apache_24", and this application affects the availability of other applications.

Managing the web server component

The web server is managed by console utilities found in the "<AppServer_HOME>/bin" directory:

- "wsstart" – start the web server
- "wsstop" – stop the web server
- "wsrestart" – restart the web server
- "wsstatus" – show information about the web server's operation

> ⓘ Note that when Apache web server module settings are changed, to restart the web server, sequentially execute the commands "wsstop" and "wsstart" insead of the command "wsrestart". The "wsrestart" command should only be used if a new application was added and it is necessary to make it available through the web server component, and this application affects the availability of other applications.

When the web server component is started using the "wsstart" utility, the following error may appear:

```
0509-022 Cannot load module /home/way4/appserver/apache-2.4/lib/libaprutil-1.so.
0509-150 Dependent module /usr/lib/libiconv.a(libiconv.so.2) could not be
loaded.
```

In this case, it is necessary to do as follows:

- In the "<USER_HOME>/.profile" file, specify the path to libraries having defined the "LD_LIBRARY_PATH" variable (if the AIX platform is used, define the "LIBPATH" library):

```
if test "x$LIBPATH" != "x" ; then
 LIBPATH="/home/way4/appserver/apache-2.4/lib:/opt/freeware/lib:$LD_LIBRARY_PATH"

else
 LIBPATH="/home/way4/appserver/apache-2.4/lib:/opt/freeware/lib"
fi
```

- Restart the session (for example, repeat login via SSH or restart cmd.exe).

Specifying Standard Ports for HTTP and HTTPS

If it is necessary to access the "apache_24" app or the web server through standard ports (HTTP port 80 and HTTPS port 443), use the operating system tools. For example, the "iptables" utility can be used for Linux (see "Accessing the Web Server through a Port up to 1024").

Obtaining a Global CA Certificate for the Web Server

WAY4 Application Server can be used as a front-end server. In this case, it is necessary to ensure a secure connection for the "apache_24" application or for the web server using the SSL/HTTPS protocol.

To obtain and use a certificate for the "apache_24" application or for the web server, proceed as follows:

- Select a global certification authority (CA).

  This may be either a certification authority or an agent assisting in obtaining a certificate. The official sites of the most well-known certification authorities are: http://www.verisign.com, http://www.comodo.com, http://www.thawte.com, http://www.digicert.com, and http://www.geotrust.com.

- Generate a private key and a request to certify a public key (certificate signing request, CSR).

  A detailed manual for private key and CSR generation may be found on the CA site.

- Fill in a certificate request on the CA site.

When prompted for a CSR ("Enter Certificate Signing Request"), enter the CSR generated before and select either "SSL Web Server Certificate" or "Apache" as a platform.

- Receive the certificate and import it into the web server.

> If the format of the certificate file is "PKCS#12" (the file name is "<certificate name>.p12"), it must be converted into the PEM format. For this, execute the following command:

```
openssl pkcs12 -in <path_to_file>/<certificate name>.p12 -out <path_to_file>
/<certificate name>.pem -nodes
```

To import a certificate to the "apache_24" application when OpenSSL is used, do as follows:

- Replace the contents of the "<AppServer_HOME>/appserver/applications/apache_24/conf/webserver/ssl/crt/server.crt" file with the certificate which must be copied from the "<certificate name>.pem" file.
- Replace the contents of the "<AppServer_HOME>/appserver/applications/apache_24/conf/webserver/ssl/key/server.key" file with the private key which must be copied from the private key file that was created earlier.

  To import a certificate into the web server when OpenSSL is used, do as follows:

  - Replace the contents of the "<AppServer_HOME>/conf/webserver/ssl/crt/server.crt" file with the certificate which must be copied from the "<certificate name>.pem" file.
  - Replace the contents of the "<AppServer_HOME>/conf/webserver/ssl/key/server.key" file with the private key which must be copied from the private key file that was created earlier.

> For the "apache_24" application or the web server component to be started automatically when WAY4 Application Server is started, the private key file must be stored in open form. To ensure information security, it is recommended that access to the file containing the private key in open form be restricted.

- After importing the private key and certificate files, restart the "apache_24" application or the web server.

Chapter 10. Managing WAY4 Applications

WAY4 applications are managed through console utilities located in the "<AppServer_HOME>/bin" directory, listed in Table 7.

ⓘ Command parameters are given in the table in square brackets.

Table 7. Console utilities for managing WAY4 applications

Command	Function
creinst [app_name=<application name>] [app_type=<application type>] [file=<path to application archive file (WAR or zip>] [http_port=<port number>]	Creates an instance of the application, deploys the WAY4 application (if the path to the WAY4 application WAR archive is indicated), also indicates the HTTP port for interaction with the server. Note that when the application is created, the default value of the parameter auto_start=true. Also, when the application is created, its default configuration is saved in the archive file "<AppServer_HOME>/applications/<Application_Name>/conf/original_configuration.zip". The "app_type" parameter is only used when an application template is created (when the "file" parameter is not specified). If the "file" parameter is specified, the "app_type" parameter is ignored. As a value of the "app_type" parameter, specify "w4webapp" for web applications (e.g. WAY4U, WAY4 Web Banking, and WAY4 Remote Access) or "w4app" for back-end applications (e.g. WAY4 Health Monitoring). An HTTP port number must only be specified for web applications. Note that this command can only be executed under the user in whose session WAY4 Application Server will be used. If an attempt is made to execute this command under the superuser (root), an error message will be displayed.

Command	Function
web_context=<context name>] [configure_https=<configure or do not configure HTTPS connection. The default value is "y">] [https_port=<HTTPS port number>] [p12_file=<path to the file "<certificate name>. p12" in "PKCS#12" format>] [p12_store_password=<password for the certificate file "<certificate name>.p12" in "PKCS#12" format>] [p12_private_key_password=<password for a private key in the keystore. If the keystore does not contain private keys, this parameter is not used>]	When installing an application, the application's name that is used to identify this application can be redefined in the URL address by the "web_context" parameter (by default, corresponds to the name of the application). the same value of the "web_context" parameter can be set for different instances of applications. In this case, the URL address string will only differ by the HTTP port number. Note that use of the same "web_context" parameter value for different instances of applications in distributed load mode (see "Web Server Operating in Proxy Server Mode") will lead to inability of the application through the Apache web server. If the "n" value is selected for the "configure_https" parameter when executing the "creinst" command, the warning message "The HTTPS configuration is skipped" will be displayed, since the absence of an HTTPS connection leads to a loss of compliance with PCI DSS requirements (for more information, see the document "WAY4™ PA-DSS Implementation Guide"). If the "y" value (default) is selected for the "configure_https" parameter when executing the "creinst" command, set values for the "p12_file", "p12_store_password", and "p12_private_key_password" parameters. Connection on the HTTP protocol will be disabled. For more information about generation of certificates for secure access to applications, see the section " Secure Web Application Access Management". The connection type can be changed in the "<AppServer_HOME>/appserver /applications/<Application_Name>/conf /config-web.properties" configuration file.
	Deletes the WAY4 application. During execution of this command, a snapshot of the application is made if the "dumpOnDelete" parameter value is "true" (see Table 3 in the section "Configuring WAY4 Application Server"). It is possible to limit the total size of log files and the date

Command	Function
delinst [app_name=<application name>]	from which log files will be included in a snapshot (see the description of the parameters "default_snapshot_logs_max_age" and "default_snapshot_logs_max_size" in the section "Configuring Application Common Parameters".
updinst [app_name=<application name>] [file=<path to file>] [web_context=<context name>]	Updates the WAY4 application. During execution of this command, a snapshot of the application is made if the "dumpOnUpdate" parameter value is "true" (see Table 3 in the section " Configuring WAY4 Application Server"). It is possible to limit the total size of log files and the date from which log files will be included in a snapshot (see the description of the parameters "default_snapshot_logs_max_age" and "default_snapshot_logs_max_size" in the section "Configuring Application Common Parameters". Note that this command can only be executed under the user in whose session WAY4 Application Server will be used. If an attempt is made to execute this command under the superuser (root), an error message will be displayed. When updating an application, the "web_context" parameter can be used to redefine the name of the context that was set on installation. If the parameter's value is not set, the context name will not change.
security_conf [operation={enable | disable}] [app_name=<application name>]	Enable/disable the mode for redirecting requests from HTTP to HTTPS. Only used if an application is accessed using Apache Web server. If the parameter "app_name=<application name>" is not specified, redirecting will be enabled for all web applications. Note that this command can only be executed under the user in whose session WAY4 Application Server will be used. If an attempt is made to execute this command under the superuser (root), an error message will be displayed.

Command	Function
status	Receives a list of deployed applications, application and WAY4 Application Server statistics. "status" command format is described in the section "Managing WAY4 Application Server under Unix".
gc <application name>	Runs garbage collection; this function is automatically started after a set time period, and may be manually started, for example when it is necessary to know exactly how much memory is taken up by the WAY4 application.
start <application name>	Starts the WAY4 application.
startall	Start all WAY4 applications
reninst <app_name=application name> <new_name=new application name>	Rename a WAY4 application. Before changing the name, the application must be stopped. When the name is changed, a snapshot of the application is saved. The names of all applications can be changed, except system applications. When an attempt is made to rename a system application, the error message "Application type for <system_app_name> is system" will be displayed. Note that this command can only be executed under the user in whose session WAY4 Application Server is used. If an attempt is made to execute this command under the superuser (root), an error message will be displayed.
restart <application name>	Restarts the WAY4 application.
stop <application name>	Stops the WAY4 application.
stopall	Stop all WAY4 applications
check <application name>	Check whether the application is available and what memory volume it occupies.
	Save logs, configuration, snapshot, as well as information about the configuration and operation of the operating system on which WAY4 Application Server is installed. The archive with the necessary files will be located in the "<AppServer_HOME> /dumps" directory. The "level" parameter can have one of the following values:

Command	Function
dump [level={logs | conf | all | snapshot | diag}] [app_name=<application name or comma-separated list>] [from=<date_from>] [to=<date_to>]	• "logs", "log", "l" (default value) – save only log files of all applications and WAY4 Application Server. • "conf", "cfg", "c" – save only the configuration of all applications and WAY4 Application Server, create the file "<AppServer_HOME>/conf/generated /osinfo.dat" with minimal information about the operating system and save this file in the archive with the configuration. • "all", "a" – save log files and the configuration of all applications and WAY4 Application Server. • "snapshot", "snap", "s" – saves the snapshot of the application whose name is set by the "app_name=<name>" parameter. It is possible to limit the total size of log files and the date from which log files will be included in a snapshot (see the description of the parameters "default_snapshot_logs_max_age" and "default_snapshot_logs_max_size" in the section "Configuring Application Common Parameters"). • "diag", "d" – starts scripts that gather information about the configuration and operation of the operating system, as well as about execution of processes. This information is used to diagnose WAY4 Application Server operation. Script results will be located in the file "<AppServer_HOME>/logs/diag-<date and time>.log" and the archive in the "<AppServer_HOME>/dumps" directory. In addition, the "diag" utility parameter can be used for the "diag" value. An application name or comma-separated list of application names can be specified in the "app_name" parameter. The "from" and "to" parameters determine the time interval for the creation date and modification date of log files that must go into the archive. These parameters make it possible to reduce the number of log files in the archive. Parameters can be set for the "logs", "all" and "snapshot" levels. A time interval only applies to log files (for example, for the "all" level, application configuration files will always be saved,

Command	Function
	regardless of whether their modification date falls in a specific time interval). "from" /"to" dates can be set in one of the following formats: • "yyyyMMdd" – starting/ending from this date, where "yyyy" – is the year, MM is the number of the month, "dd" – is the day of the month; • "yyyyMMdd_HHmm" – same as "yyyyMMdd", but time is additionally defined (hours "HH" and minutes "mm"); • "[-Kw] -Nd [-Mh [-Lm]]" – files created "K" weeks, "N" days, "M" hours and "L" minutes before the current date and time. The "Kw", "-Mh" и "-Lm" parameters are not mandatory. Examples: • "dump level=all from=20160123 to=20160223" – configuration and log files from 23 January 2016 to 23 February 2016 will be dumped for all applications and WAY4 Application Server. • "dump level=all from=20160123_1540" – configuration and log files in the period starting from 15:40 on 23 January 2016 will be dumped for all applications and WAY4 Application Server. • "dump level=logs from=-3d to=-1d" – log files in the period from 3 days ago to 1 day ago will be dumped for all applications and WAY4 Application Server. • "dump level=logs from=-6h" – log files created in the last 6 hours will be dumped for all applications and WAY4 Application Server. Note that this command can only be executed under the user in whose session WAY4 Application Server will be used. If an attempt is made to execute this command under the superuser (root), an error message will be displayed.
	Gather information about the configuration and operation of the operating system, about the execution of processes, stack traces and JVM heap dump for an application. The "type" parameter can have one of the following values:

Command	Function
diag [type={OS|java_app }] [app_name=<application name>]	• "OS" (default value) – gather information about the configuration and operation of the operating system. The result will be saved in the file "<AppServer_HOME>/logs /diag-<date and time>.log". • "java_app" – save stack traces for java applications. The "app_name=<name>" (application name) parameter must be set.
restore file=<path to snapshot of application>	Restore the application from the snapshot made earlier.
importcert [app_name=<application name>] [p12_file=path to PKCS#12 keystore] [p12_keystore_password=password for source keystore] [private_key_password=password for private key] [p12_file=<path to the file "<certificate name>.p12" in "PKCS#12" format>] [p12_store_password=<password for the certificate file "<certificate name>.p12" in "PKCS#12" format>] [p12_private_key_password=<password for a private key in the keystore. If the keystore does not contain private keys, this parameter is not used>]	Imports the key into the web application keystore (see "Utility "importcert"). Note that this command can only be executed under the user in whose session WAY4 Application Server will be used. If an attempt is made to execute this command under the superuser (root), an error message will be displayed.
usrmgmt [app_name=<application name>] [auth_method={FORM|BASIC}] [oper={add|del|pwd}] [user_name=<username>] [password=<password>] [role_name=<user role>]	Adds and deletes users, changes user passwords and switches on mandatory authorisation mode for the web application (see "Utility "usrmgmt"). Note that before executing the "usrmgmt" command it is first necessary to import a certificate for the web application by executing the "importcert" command. Note that this command can only be executed under the user in whose session WAY4 Application Server will be used. If an attempt is made to execute this command under the superuser (root), an error message will be displayed
digestpassword [user_password=<password to digest>]	Transforms the given password to a hash amount using the SHA-1 algorithm.
logagent_conf	Utility used to configure the "logagent" system application (for more information, see the section "logagent.xml File" of the document "Administering WAY4 Health Monitoring Gen2").
version	Shows the current version of WAY4 Application Server installed components.

ⓘ If the console utilities are started without any parameters, the parameters will be entered in interactive mode.

Checking Application Distribution Consistency

When the creinst or updinst commands are executed, the application distribution's consistency is checked"

- Using checksums.
- Using a digital signature.

If the distribution contains a file with the ".sha256" extension, SHA-256 checksums are matched. If the checksums don't match, operation of the creinst/updinst utilities is terminated. If the distribution does not contain a file with the "sha256" extension, a message that there is no checksum file: "Warning: The application distribution <file> does not have checksum file" is output to the log file "<AppServer_HOME>/logs/core-ant<file creation date and time>".

If the distribution contains a file with the ".sign" extension, the digital signature is checked. If it does not match, operation of the creinst/updinst utilities is terminated. If the distribution does not contain a file with the ".sign" extension, a message that there is no file with a digital signature: "Warning: The application distribution <file> does not have signature file. The distribution consistency verification has not been performed" is output to the log file "<AppServer_HOME>/logs/core-ant<file creation date and time>".

Universal Utility "appcmd"

The "appcmd" utility is used to call any management command.

Management commands are invoked as follows:

```
./appcmd <command_name> [parameters]
```

The list of available commands and information on their execution is accessed by this command:

```
./appcmd help
```

The following management commands are available through the "appcmd" utility:

- check – display WAY4 application statistics.
- start – start the WAY4 application.

ⓘ Note that if the java version in which the application was started is incompatible with the java version supported by WAY4 Application Server, when an attempt is made to start, an error message will be displayed, the application will not be started, and the message "[error] ... <app_name>: The process cannot be started due to the

> application java <version_number_1> specification incompatibility with required java <version_number_2> specification" will be added to the log. In this case, contact WAY4 customer service.

- startall – start all WAY4 applications.
- restart – restart the WAY4 application.
- stop – stop the WAY4 application.
- stopall – stop all WAY4 applications.
- status – display a list of installed applications, application and WAY4 Application Server statistics. "status" command format is described in the section "Managing WAY4 Application Server under Unix".

Universal Utility "appsrvctl"

The "appsrvctl" utility is used to call the following management commands:

- "creinst" – create an instance of an application, install a WAY4 application.
- "sm" – start the WAY4 Solution Mounter component to install a WAY4 package component according to a scenario and response file that contains parameters specified by the user (see "Deploying and configuring WAY4 components with WAY4 Solution Mounter ").
- "upgrade" – upgrade WAY4 Application Server.
- "updinst" – update a WAY4 application.

A management command is called as follows:

```
./appsrvctl <command_name> [parameters]
```

The "creinst" and "updinst" parameters are the same as the respective console utilities located in the "<AppServer_HOME>/bin" directory (see the section "Managing WAY4 Applications").

For the "sm" command's parameters, see the section "Deploying and configuring WAY4 components with WAY4 Solution Mounter " .

Deploying and configuring WAY4 components with WAY4 Solution Mounter

WAY4 Solution Mounter is a product that is used to deploy and configure regular WAY4 components and WAY4 package components.

Solution Mounter components interact as follows:

- Save a WAY4 package component distribution to the computer on which WAY4 Application Server is installed.

- Edit the response file - the file contains definitions and parameter values that are required to deploy the component. These parameters are, for example, application names, port numbers, paths for application servers (including SSH access parameters) on which applications must be installed, etc. After parameters have been set, save the response file. This file can be used later to update the package component.

- Using the command line interface, start WAY4 Solution Mounter. WAY4 Solution Mounter is started with the command " ./appsrvctl sm <parameters> ". Specify the following parameters:
- "--scenario-location <scenario_path>" – path to the installation scenario that can specify a decompressed or archived (zip) installation scenario directory (mandatory parameter). An installation scenario is a file that contains a sequence of steps that WAY4 Solution Mounter must execute to deploy and configure a WAY4 package component. An installation scenario is part of a WAY4 package component's distribution.
- "--action <action>" – action (mandatory parameter). possible values:
  - "deploy" – install.
  - "configure" – this parameter is not currently available.
  - "undeploy" – this parameter is not currently available.
- "--response-file <response_file_path>" – path to the response file. Can specify an absolute or relative path. A response file is filled in before installing a package component.
- "-v" – debugging mode.
- "-P<parameter>=<value>" – parameter for redefining response file parameter values.

For example:

```
appsrvctl sm --scenario-location C:\temp\zookeeper-sm-3.4.12.152\zookeeper-sm-1.0
-SNAPSHOT --action deploy --response-file C:\temp\zookeeper-sm-3.4.12.152\zookeep
er-sm-1.0-SNAPSHOT\work\zk.local.response.properties -Pzk-applications.zk1.
server-ports=5555
```

- The executive process for deploying and configuring starts installation and/or update of components (creinst/updinst) according to the installation scenario.
- If deployment of a component was terminated, the next time WAY4 Solution Mounter is started, deployment will continue from the step at which it was terminated, since files with information about the deployment process are saved in the store of components that the package component consists of, as well as installation log files and files for states of the WAY4 package component's deployment and configuration.
- After installation and configuration of a package component has been completed, a response file and installation log files will be saved.

Secure Web Application Access Management

Secure access can be set up for WAY4 Web applications, and for WAY4 applications if the WAY4 application supports this functionality.

There are two types of web applications: applications that service Intranet requests and applications that service Internet requests.

It is recommended that secure access for web applications servicing Internet requests be set up through the "apache_24" application (see "Managing the Apache Web Server ").

This section describes the main configurations for applications servicing Intranet requests. Setup of secure connection over SSL/HTTPS consists of the following steps:

- Generating certification authority (CA) certificates and application keys and certificates
- Importing keys into the application keystore
- Working with names and passwords of users having access to the application

Pursuant to PCI DSS (http://www.pcisecuritystandards.org), the encryption algorithms (ciphers) provided in the following configuration files can be used to ensure security when using a secure connection over HTTPS.

- "<AppServer_HOME>/conf/webserver/templates/httpd-openssl.conf" – template of configuration file. The file is updated each time WAY4 Application Server is updated. In this file encryption methods are listed, respectively, in the "SSLCipherSuite" parameter. Example:

```
SSLCipherSuite EECDH+ECDSA+AESGCM:EECDH+aRSA+AESGCM:EECDH+ECDSA+SHA384:
EECDH+ECDSA+SHA256:EECDH+aRSA+SHA384:EECDH+aRSA+SHA256:EECDH+aRSA+RC4:
EECDH:EDH+aRSA:RC4:!aNULL:!eNULL:!LOW:!3DES:!MD5:!EXP:!PSK:!SRP:
```

- "<AppServer_HOME>/appserver/applications/<Application_Name>/conf/templates /config-web.properties" – template for the application "<Application_Name>"; encryption methods are listed in the "ciphers" parameter, for example:

```
ciphers="TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256,
TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA,TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384,
TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA,TLS_RSA_WITH_AES_128_CBC_SHA256,
TLS_RSA_WITH_AES_128_CBC_SHA,TLS_RSA_WITH_AES_256_CBC_SHA256,
TLS_RSA_WITH_AES_256_CBC_SHA"
```

Therefore, after updating WAY4 Application Server, manually transfer information on encryption methods from template files to the corresponding configuration files (not required in initial installation of WAY4 Application Server), specifically:

- Transfer the parameter "SSLCipherSuite" from the file "<AppServer_HOME>/appserver/conf/webserver/templates/httpd-openssl.conf" to the file "<AppServer_HOME>/conf/webserver/httpd-openssl.conf".
- Transfer the "ciphers" parameter from the file "<AppServer_HOME>/appserver/applications/<Application_Name>/conf/templates/config-web.properties" to the file "<AppServer_HOME>/appserver/applications/<Application_Name>/conf/config-web.properties".

Certificate Generation

WAY4 Application Server certificates are generated by utilities supplied together with the WAY4 Application Server distribution kit.

Certificates are generated in two steps:

- Certification authority certificate generation
- Generation of application key and certificates

> ! Pursuant to PCI DSS, it is only allowed to use certificates issued by global certification authorities when configuring WAY4 Application Server to process user requests from external systems (e.g. from the web). Certificates generated by utilities supplied together with the WAY4 Application Server distribution kit are used to set up a secure connection in the internal network of the bank or processing centre. The certificates are also used when all users of a WAY4 application have received it over a secure data exchange channel, e.g. on a removable physical medium or encrypted by PGP.

CA Certificate Generation

To generate a CA certificate, start the command file "new_ca.cmd" (for MS Windows) or "new_ca.sh" (for Unix). These files are found in the directory containing certificate generation utilities ("<AppServer_HOME>/certs"). The following fields displayed by the command file are filled in during generation:

- *Country Name* – two-letter country code. The default value is "Unknown"
- State or Province Name – full name of the country. The default value is "Unknown"
- Locality Name – city name. The default value is "Unknown"
- Organisation Name – company name. The default value is "Unknown"
- *Organisational Unit Name* – department name. The default value is "Unknown"
- *Common Name (Application CA name)* – certification authority certificate name. The default value is "Unknown"
- *Email Address* – e-mail address, for instance, an administrator's one. The default value is "Unknown"

To use default data, press <Enter>.

These parameters can be also specified in the command line. For example:

```
new_ca.cmd C=EU ST=Belgium L=Brussels O=OpenWay OU=Online Card Division
CN=Application CA EMAILADDRESS=admin@openwaygroup.com
```

where:

- C – *Country Name*
- ST – *State or Province Name*
- L – *Locality Name*
- O – *Organization Name*
- OU – *Organizational Unit Name*
- CN – *Common Name (Application CA name)*
- EMAILADDRESS – *Email Address*

In addition, the following parameters can be specified:

- DAYS – the certificate duration time in days. The default value is 1095
- SAN – *Subject Alternative Name*

For example:

```
new_ca.cmd DAYS=730 SAN="ip.1=127.0.0.1,dns.1=localhost;dns.2=test.domain.
com"
```

ⓘ It is necessary to remember the entered password (PEM passphrase), which will further be used to generate other certificates.

After all the fields are filled in, the "CERTS" directory is automatically created in the directory containing certificate generation utilities. The CA certificate files "cacert.pem" and "cacert.p12" are will be created in the "CA" subdirectory.

The CA certificate file must be imported into application users' browsers. If the CA certificate is not imported into the browser, the corresponding warning will be displayed when the user addresses the web application.

! If an error occurs during CA certificate generation, delete the "CERTS" directory and generate a CA certificate again.

Web Application Certificate Generation

To generate an application certificate, start the command file "new_cert.cmd" (for MS Windows) or "new_cert.sh" (for Unix). These files are found in the directory containing certificate generation utilities ("<AppServer_HOME>/certs"). Specify a certificate name in the command line:

```
new_cert.cmd <certificate name> (for Windows)
```

```
./new_cert.sh <certificate name> (for Unix)
```

The created certificate must be signed by the CA certificate. For this, the CA certificate password (PEM passphrase) is prompted for in a dialogue with the command file: "Enter password for CA key storage". Then, enter a user certificate access password: "Enter password for "cert_name" key storage". After the passwords has been entered, fill in the following fields:

- *Country Name* – two-letter country code. The default value is "Unknown"
- *State or Province Name* – full name of the country. The default value is "Unknown"
- Locality Name – city name. The default value is "Unknown"
- *Organisation Name* – company name. The default value is "Unknown"
- *Unit Name* – value "OpenWay Application" is specified in the field by default and must not be changed
- *Certificate common name* – certificate name; it is recommended that the domain name for accessing WAY4 Application Server be specified in this field

> If a certificate name is not the same as the domain name for accessing WAY4 Application Server, the corresponding warning will be displayed when the user addresses the application.

- *Email Address* – e-mail address, for instance, that of an administrator. The default value is "Unknown"

To use default data, press <Enter>.

After the certificate has been created successfully, the following message will be displayed:

```
New application certificate and key are in
<AppServer_HOME>\appserver\certs\certs\<cert_name>.p12 file
Application certificate is in
<AppServer_HOME>>\appserver\certs\certs\<cert_name>.pem file
```

These parameters can be also specified in the command line. For example:

```
new_cert.cmd C=EU ST=Belgium L=Brussels O=OpenWay OU=OpenWay Application
CN=<domain_name> EMAILADDRESS=admin@openwaygroup.com
```

where:

- C – *Country Name*
- ST – *State or Province Name*
- L – *Locality Name*
- O – *Organization Name*
- OU – *Organizational Unit Name*
- CN – *Certificate common name*

- EMAILADDRESS – *Email Address*

In addition, the following parameters can be specified:

- DAYS – the certificate duration time in days. The default value is 365
- SAN – Subject Alternative Name

For example:

```
new_cert.cmd <certificate name> DAYS=730 SAN="ip.1=127.0.0.1,dns.
1=localhost;dns.2=test.domain.com"
```

As a result, the certificate files "<certificate name>.pem" and "<certificate name>.p12" in the "PKCS#12" format will be created in the "CERTS" directory. The certificate data will be inputted in the certificate database. The entire chain of trusted certificates is also added to the "<certificate name>.p12" certificate.

Utilities for Secure Access Setup

The "importcert" command is used to set up secure access to applications over SSL/HTTPS. The "usrmgmt" command is used to manage user access to applications.

"importcert" Utility

The "importcert" utility is used to import a key, certificate, and entire chain of trusted certificates from the "<certificate name>.p12" file in the "PKCS#12" format into a web application keystore. The keystore is located in the "<Application_Name>/conf/security/appkeystore.jks" file. Also, the utility is used to switch on HTTPS connection support.

> Note that the "importcert" utility can only be run under the user in whose session WAY4 Application Server will be used. If an attempt is made to run this utility under the superuser (root), an error message will be displayed.If the "importcert" utility is used but the "usrmgmt" utility is not used, web applications can be accessed both under HTTPS and HTTP.

The "importcert" utility has the following parameters:

- "app_name=<application instance name>" – application name
- "p12_file=<path to PKCS#12 keystore>" – full path to the "<certificate name>.p12" file in the "PKCS#12" format
- "p12_keystore_password=<password for source keystore>" – access password of the certificate file "<certificate name>.p12" in the PKCS#12 format
- "private_key_password=<password for private key>" – password of the private key found in the keystore. If there are no private keys in the keystore, the parameter is not used.

> If a PKCS#12 file was generated by "new_ca.cmd" ("new_ca.sh") and "new_cert.cmd" ("new_cert.sh") utilities included in the WAY4 Application Server distribution kit, the private key password must be the same as the certificate file access password.

> After the key expires, it is necessary to import a new effective key into the application keystore. It is recommended to import a certificate with the same name ("<certificate name>") as the one in use.

It is not recommended to import several certificates with different names to the keystore. If a certificate has not expired and an attempt is made to import a certificate with another name to the keystore, the following message will be displayed "The certificate file already exists. Importing new data to it can cause key alias collisions. Are you sure to proceed? [y/N]". If "y" is specified, a second certificate will be imported to the keystore, i.e., the keystore will contain two valid certificates with different names. In this case, the Apache Tomcat component will select one certificate for use, according to its algorithm. In WAY4, it is possible to explicitly specify the certificate to use. To do so, in the the "<AppServer_HOME>/appserver/applications/<Application_Name>/conf/config-web.properties" file, set the value of the "key_alias=" <certificate alias>" parameter. For example:

```
key_store_file=conf/security/appkeystore.jks
key_store_password=<encrypted password>
key_alias=<certificate alias>
ciphers=TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256,TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA,
TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384,TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA,
TLS_RSA_WITH_AES_128_CBC_SHA256,TLS_RSA_WITH_AES_128_CBC_SHA,
TLS_RSA_WITH_AES_256_CBC_SHA256,TLS_RSA_WITH_AES_256_CBC_SHA
```

"usrmgmt" Utility

The "usrmgmt" utility is used to add and delete users and change user passwords.

Also, the utility switches on mandatory authorisation for accessing an application. In this case, the application can only be accessed through a secure HTTPS connection.

> Note that the "usrmgmt" utility can only be run under the user in whose session WAY4 Application Server will be used. If an attempt is made to run this utility under the superuser (root), an error message will be displayed.
>
> Before executing this command, import an application certificate using the "importcert" command.
>
> After executing this command, the application must be restarted.

The "usrmgmt" utility has the following parameters:

- "app_name=<application instance name>" – application name.
- "auth_method=<value>" – authentication method; the field takes on the following values:
  - "BASIC" (default value) – browser authorisation with authorisation data transfer is used.

  - "FORM" – authorisation is performed using user forms found in files "<Application_Name>/webapps/<Application_Name>/jsp/login.jsp" (the form for entering a username and a password) and "<Application_Name>/webapps/<Application_Name>/jsp/error.jsp" (the form used in case of authorisation errors).

- "oper=<value>" – operation that needs to be performed:
  - "add" – add a new user
  - "del" – delete a registered user
  - "pwd" – change a registered user's password
- "user_name=<username>" – name of the user for whom the operation is performed.
- "password=<password>" – user password. Password requirements pursuant to PCI DSS are described in the section "Registering Users and Setting Passwords" of the document "WAY4™ PCI DSS Implementation Guide".
- "role_name"=<user role>" – user role; the list of roles is determined by the specific application. By default, the "as_administrator" role is used.

Authorising Applications using Client Certificates

To authorise applications using client certificates, the following settings must be made:

- Using the "keytool" utility, create a certificate store and import to it the received Certification Authority (CA) certificate that will be used to sign client certificates. To do so, execute the command:

```
<OWS_JAVA_HOME>/jdk/bin/keytool -import -keystore <AppServer_HOME>/applications
/<Application_Name>/conf/security/tcstore.jks -storepass <password> -file <pem
file>
```

Here <Application_Name> is the directory where the application is installed; "tcstore.iks" is the certificate store in the directory <AppServer_HOME>/applications/<Application_Name>/conf/security (if this directory doesn't exist, it must be created manually); <password> is the password for access to the store; <pem file> is the path to the CA certificate file in PEM format.

In the configuration file "<AppServer_HOME>/appserver/applications/<Application_Name>/conf/config-web.properties", do as follows:

  - Ensure that the "false" value is not set for the "https_disable" parameter.
  - Set a value for the "trust_store_file" parameter, specifying the relative path to the certificate store "conf/security/tcstore.jks" as the value.
  - Set a value for the "trust_store_password" parameter, specifying the password from the certificate store as the value.

ⓘ If a password is set in clear text, insert the "plain:" variable before the password value, for example: "trust_store_password=plain:PAssword123".

> If passwords "trust_store_password" and "key_store_password" must be encrypted, use the "nscipher" utility that is located in the "<AppServer_HOME>/bin/tools" directory. The "nscipher" utility must be started with the "ows_application" key. An encrypted password should be specified without the "plain:" variable.

Example:

```
key_store_file=conf/security/appkeystore.jks
key_store_password=oTB09wZW5XYXkxHDAaBgNV
trust_store_file=conf/security/tcstore.jks
trust_store_password=plain:PAssword123
```

- After the settings have been made, restart the application.
- For access to the application from a browser, the client certificate received earlier and signed with the CA certificate must be imported to the browser.

Enabling Two-Factor Authentication

An application may use two-factor authentication for secure access: based on a certificate as the first factor, and a user name and password as the second factor.

To enable two-factor authentication, do as follows:

- If necessary, generate a CA certificate (see "CA Certificate Generation").
- Generate a certificate for the application (see "Application Certificate Generation").
- Import the key and certificate from the "<certificate name>.p12" certificate that was generated to the application's keystore (see "Utility "importcert"").
- Add a new user or change the password of an existing one (see "Utility "usrmgmt"").
- Generate client certificates in the same way as application certificates (see "Application Certificate Generation").
- In the configuration file "<AppServer_HOME>/applications/<Application_Name>/conf/config.properties" specify "auth_type=password".
- To access the application from the browser, import the CA certificate and client certificate to the browser; enter the password and name of the user created with the "usrmgmt" utility.

Registering Versions of WAY4 Applications in the WAY4 Cards Database

In WAY4, it is possible to register versions of WAY4 applications in the WAY4 Cards database. The version number of each WAY4 application is taken from metadata files supplied together with the application, after which they are automatically registered in the SY_APPL_VERSION table. Note that a version is registered when the application is started.

> If there are no metadata files, the application version is not registered. Information about the possibility to register the versions of a specific application is provided by WAY4 customer support.

To register WAY4 application versions in the WAY4 Cards database, it is necessary to do as follows:

- Delete comment characters "#" or ensure they have been deleted in the "<AppServer_HOME>/conf/AppContainer.properties" file for the following strings:

```
jmx_enabled=yes
jmx_port=9999
```

- In the "<AppServer_HOME>/applications/monitoring/conf/config.properties" file, specify the following parameters for connection to the database:
  - "db_url" – string for connection with the database in the format "jdbc:oracle:thin:@<Host>:<Port>:<SID>".
  - "db_user" – Oracle user under whom connection is made with the WAY4 Cards database.

  Note that this user must have privileges to connect with the database, establish sessions and execute the package SOFT.REGISTER_APPL_VERSION".

  - "db_password" – user password specified with the "db_user" parameter. Password should be encrypted using the program "nscipher.exe", located in the directory "<AppServer_HOME>/bin/tools".
  - "db_owner" – name of the WAY4 Cards database schema owner.

  An example of parameters for connection with the database:

```
db_url=jdbc:oracle:thin:@SERVER:1521:SID
db_user=OWS_MON_USER
db_password=PASSWORD
db_owner=OWS
```

- Restart WAY4 Application Server.
- Start WAY4 applications.

Application versions registered in the database are viewed using the client application DB Manager / WAY4 Manager; the "Application Versions" form (Full → DB Administrator Utilities → Users & Grants → Application Versions) contains the application name, version and the time the application was first and most recently run.

Chapter 11. Monitoring WAY4 Application Server

This section describes the function and location of log files used to monitor the availability of WAY4 Applications Server, WAY4 applications based on it, and the web server.

> ⓘ It is possible to save the log files of all WAY4 applications. To do so, use the "dump" console utility (see Table 7 of the section "Managing WAY4 Applications").

WAY4 Application Server Log Files

If WAY4 Application Server malfunctions, it is recommended that users analyse the reasons for failure contained in the WAY4 Application Container log files: "<AppServer_HOME>/logs /container_<date and time of file creation>".

After WAY4 Application Server is started, WAY4 Application Container's total and free memory volume will be logged in the component's log file every hour. After WAY4 Application Server is stopped, the maximum memory volume used by WAY4 Application Container will be logged in the component's log file.

Also, it is recommended that users analyse the data contained in the servlet container log file "<AppServer_HOME>/applications/<Application_Name>/logs/tomcat.log".

Audit Log

The WAY4 Application Server audit log is a set of text CSV files (Comma-Separated Values) containing information about actions by users and internal processes with WAY4 Application Server and WAY4 applications. CSV files can be analysed with Microsoft Excel, for example.

The audit log contains information about starting, stopping and restarting applications (their channels and internal processes); about adding, deleting and upgrading applications, etc.

WAY4 Application Server has the following types of audit log file:

- "<AppServer_HOME>/logs/audit_<creation date and time>.log" – log of WAY4 Application Server runtime operations; for example, starting and stopping an application.
- "<AppServer_HOME>/logs/audit_offline.log" – log of operations not linked to WAY4 Application Server runtime; for example, creating an application instance ("creinst" command).

CLI Utility Log Files

Log files with a detailed log of Command line interface utility operation: "<AppServer_HOME> /logs/core-ant<file creation date and time>".

WAY4 Application Log Files

WAY4 application log files are located in directories whose paths vary depending on the application. For example, "<AppServer_HOME>/applications/<Application_Name>/webapps/<Application_Name>/WEB-INF/runtime/log" or "<AppServer_HOME>/applications/<Application_Name>/logs".

Log Files with JVM Metrics

Log files with JVM metrics: : "<AppServer_HOME>/logs/mon_<file creation date and time>".

Web Server Log Files

Log files for the "apache_24" application are found in the "<AppServer_HOME>/appserver/applications/apache_24/logs" directory. They include:

- "httpd-access.log.<file creation date and time>" – files for registering incoming requests
- "httpd-ssl-access.log.<file creation date and time>" – files for registering incoming SSL/HTTPS requests when "OpenSSL" libraries are used.
- "httpd-error.log.<file creation date and time>" – files for registering any errors that occur during web server operation
- "httpd-ssl-error.log.<file creation date and time>" – files for registering transport level errors while working under SSL/HTTPS protocol when "OpenSSL" libraries are used.

> ⓘ When the web server component is used, log files are located in the "<AppServer_HOME>/logs" directory.

OS Service Logs

Log files for OS services can be found in the following locations:

- for Solaris 11 in the file "/var/svc/log/application-WAY4ApplicationServer:default.log"
- for RHEL6 in the file "/var/log/boot.log"
- for AIX in the file "/var/adm/messages"
- for Windows in a file in Event Viewer snap management.

For RHEL7, log files for OS services are output with the following command:

```
journalctl -u WAY4ApplicationServer
```

Chapter 12. WAY4 Application Server Access and Management Requirements

Pursuant to PCI DSS (http://www.pcisecuritystandards.org), the following guidelines for WAY4 Application Server access and management must be observed:

- Configure a firewall on the computer with WAY4 Application Server (see "Recommendations for Firewall Configuration"). Only those ports directly used for work must remain open for connection (for example, "22" for SSH, "80" for HTTP, and "433" for HTTPS).
- Remote connection to the computer with WAY4 Application Server is possible using the SSH protocol or RDP (only RDP over TLS).
- The web console is used for user administration. Each user with access to WAY4 Application Server and applications must have his own user account (for information about setting up work with web console users, see the section "Secure Web Application Access Management "). The system administrator must perform regular actions: delete inactive users, monitor regular password changes, etc. Password requirements pursuant to PCI DSS are described in the section "Registering Users and Setting Passwords" of the document "WAY4™ PCI DSS Implementation Guide". In particular, a password must be at least 7 characters long, contain letters and digits and be changed at least once every 90 days.
- It is recommended to prohibit the user used by default in installation ("way4" user) from connecting to the computer with WAY4 Application Server installed. All operations using the command line interface must be performed either under user accounts of users created with the web console or included in the same group as the "way4" user.

Chapter 13. Preliminary Setup of the Operating System

This section contains a description of preliminary Linux, Oracle Solaris 10/11 SPARC and IBM AIX setup.
Requirements for installing Oracle Java (for all operating systems except IBM AIX) are also covered in this section.

The "bash" and "unzip" libraries for all operating systems must be installed on the server, as well as the following libraries:

- for Linux – "expat" and "e2fsprogs-libs".
- for Oracle Solaris 10 SPARC – "libgcc", "expat" and "libiconv".
- for Oracle Solaris 11 SPARC – " gcc-45-runtime".
- for IBM AIX – the RPM utility, and the libraries "expat", "libgcc", "zlib", "gettext" and "glib2".

Preliminary configurations as well as the installation of libraries and the necessary packages are required before installing the WAY4 Application Server.

In addition, this section describes the "syscheck" utility used to check the correspondence of server parameters (hardware-software suite) on which WAY4 Application Service is being installed to technical requirements and instructions provided in this section.

Installing Oracle Java

Before installing WAY4 Application Server, it is necessary to install Oracle Java on all operating systems except IBM AIX. Ensure the Oracle Java version that is compatible with the WAY4 Application Server according to platform and bits is being installed (for the number of the required version and a description of installation, see the document "Oracle Java Commercial Updates for WAY4™ "). When Oracle Java is installed according to instructions, the following information will be shown in WAY 4 Application Server component and java application log files:

```
Java: Java Version: 8.0.192; Vendor: Oracle Corporation; JVM Type: Java HostSpot
(TM) [1.8.0_192-b26]; Compatibility: compatible [1.0.418]; JCE Unlimited
Strength Jurisdiction Policy Status: installed; Path: <OWS_JAVA_HOME>\jre (Java
HotSpot(TM) 64-Bit Server VM) in <OWS_JAVA_HOME>\jre
```

If the user changed the "OWS_JAVA_HOME" variable's value on the Windows platform, recreate the service by sequentially executing the "appsd.bat uninstall" and "appsd.bat install" commands.

"syscheck" Utility

The "syscheck" utility is located in the archive with the installer of the WAY4 Application Server distribution (see "WAY4 Application Server Distribution Kit"). This utility is used to check the correspondence of server parameters (hardware-software suite) on which WAY4 Application Server is being installed to technical requirements, and also to check for the presence of the required libraries and system settings.

It is recommended to run the "syscheck" utility manually before installing or upgrading WAY4 Application Server for diagnosis of the server state. Moreover, this utility is run automatically during WAY4 Application Server installation and upgrade (see "Installing WAY4 Application Server").

The utility is run as follows:

- Under Unix:

```
./syscheck.sh <install_dir> <user_name>
```

- Under MS Windows:

```
syscheck.bat <install_dir> <user_name>
```

Here <install_dir> is the directory in which WAY4 Application Server is installed (will be installed), <user_name> is the user in whose session work with WAY4 Application Server will take place.

The results of the utility's execution will be shown on the screen; the following types of message may be displayed:

- "INFO" – informational messages.
- "WARN" – warning. If a message of this type occurred as a result of the check, WAY4 Application Server can be installed (upgraded). However, it is not recommended to put WAY4 Application Server into production until all warnings have been eliminated.
- "ERROR" – error. If errors occur during the system check (for example, insufficient space to install WAY4 Application Server), they must be eliminated; WAY4 Application Server can only be installed (upgraded) after errors have been eliminated.

To eliminate errors and warnings, it is necessary to execute the instructions given in the section " Preliminary Setup of the Operating System" and to ensure that machine parameters conform to technical requirements given in the document "WAY4™ Application Server Main Technical Requirements".

Preliminary Setup of Linux

Before WAY4 Application Server installation, it is necessary to perform preliminary setup of the operating system. All changes must be made under the superuser (root).

For Linux, proceed as follows:

- Add the following data in the "/etc/sysctl.conf" file:

```
kernel.msgmni = 1024
kernel.sem = 250 256000 32 1024
fs.file-max = 65536
net.ipv4.ip_local_port_range = 32768 65000
```

For Red Hat Enterprise Linux 7, create a new configuration file in the directory "/etc/sysctl.d" and specify the aforementioned parameters in it. For more information, see https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/7/html/kernel_administration_guide/working_with_sysctl_and_kernel_tunables).

Note that if parameters were not applied, it is necessary to consider the information from the article https://access.redhat.com/solutions/3913331.

- Depending on the operating system, add or change the following strings in the "/etc/security/limits.conf" file (for Red Hat Enterprise Linux 5) or in "*.conf" files (for Red Hat Enterprise Linux 6 / 7) located in the /etc/security/limits.d directory; (<OS User Name> is the name of the operating system user in whose session WAY4 Application Server will be used). For more information about file structure and parameters, see the handbook accessed by executing the console command "man limits.conf".

```
<OS User Name> soft nproc 32767
<OS User Name> hard nproc 32768
<OS User Name> soft nofile 65535
<OS User Name> hard nofile 65536
```

- Restart the operating system.

Preliminary Setup of Oracle Solaris 10/11 SPARC

Before installing WAY4 Application Server, preliminary configuration of the operating system is required. All changes must be made with superuser (root) privileges.

For the operating system and Oracle Solaris 10/11 SPARC, do as follows:

- Add the following information to the "/etc/system" file

```
set rlim_fd_max=65535
set rlim_fd_cur=65536
set pidmax=32767
set max_nprocs=32766
set maxuprc=65536
```

- Restart the operating system.

Preliminary Setup of IBM AIX Operating System

Before installing WAY4 Application Server, prepare the operating system. All changes must be made with superuser (root) privileges.

For the IBM AIX operating system, the following is required:

- Add or change the following strings in the "/etc/security/limits" file (<OS_User_Name> – name of the operating system user in whose session WAY4 Application Server will be used). For more information about "/etc/security/limits" file structure and parameters, see the site https://www.ibm.com/support/knowledgecenter/ssw_aix_71/com.ibm.aix.files/limits.htm (for IBM AIX 7.1) or the site https://www.ibm.com/support/knowledgecenter/ssw_aix_72/com.ibm.aix.files/limits.htm (for IBM AIX 7.2).

```
<OS_User_Name>:
nofiles = 65535
nofiles_hard = 65536
```

- Execute the following command:

```
# chdev -l sys0 -a maxuproc=32768
```

- Restart the operating system.

Installing "expat", "e2fsprogs-libs", and "unzip" libraries under Linux

The "expat", "e2fsprogs-libs", and "unzip" libraries are necessary for web server operation. These libraries must be installed in a session of a superuser (root).

To install the "expat", "e2fsprogs-libs", and "unzip" libraries, do as follows:

- Copy the "expat", "e2fsprogs-libs", and "unzip" distribution packages (the files "expat-<version>-<release>.<architecture>.rpm", "e2fsprogs-libs-<version>-<release>.<architecture>.rpm", and "unzip-<version>-<release>.<architecture>.rpm") to a temporary directory on the server.

> The archive with the distribution package of the "expat", "e2fsprogs-libs", and "unzip" libraries can be found on the installation disk with the operating system.

- In the directory containing the distribution packages, execute the following command:

```
# rpm -i *.rpm
```

"libgcc", "expat" and "libiconv" Library Installation under Oracle Solaris 10 SPARC

The "libgcc", "expat" and "libiconv" libraries must be installed in a superuser (root) session.

To install these libraries, do as follows:

- Copy the archive with the "libgcc" (the file "libgcc-3.4.6-sol10-sparc-local.gz"), expat" (the file "expat-2.0.1-sol10-sparc-local.gz") and "libiconv" (the file "libiconv-1.14-sol10-sparc-local.gz") libraries to a temporary directory on the server.

> ⓘ The archive with the "libgcc", "expat", and "libiconv" library distribution kit is supplied together with the WAY4 Application Server distribution kit. Information about these libraries can also be found on the site of the system vendor.

- In the directory containing the archive with the libraries, execute the following commands:

```
# gunzip libgcc-3.4.6-sol10-sparc-local.gz
# pkgadd -d libgcc-3.4.6-sol10-sparc-local
# gunzip expat-2.0.1-sol10-sparc-local.gz
# pkgadd -d expat-2.0.1-sol10-sparc-local.gz
# gunzip libiconv-1.14-sol10-sparc-local.gz
```

- # pkgadd –d libiconv-1.14-sol10-sparc-local.gzExecute the following command:

```
# crle -u -l /usr/local/lib
```

Installing "gcc-45-runtime" and "system/accounting/legacy" libraries under Oracle Solaris 11 SPARC

The "gcc-45-runtime" and "system/accounting/legacy" libraries must be installed with superuser (root) privileges:

To install the "gcc-45-runtime" and "system/accounting/legacy" libraries, do as follows:

```
# pkg install gcc-45-runtime
# pkg install system/accounting/legacy
```

Additional information about installing libraries can be found on the site http://docs.oracle.com/cd/E23824_01/html/E21802/gihhp.html

Installing the RPM Utility under IBM AIX

The RPM utility must be installed in a superuser (root) session.

To install the RPM utility, do as follows:

Copy the RPM distribution (the files "rpm-3.0.5-37.aix5.1.ppc.rpm" and "rpm.rte") to a temporary directory on the server.

> ⓘ The archive with the RPM distribution can be found on the site ftp://ftp.software.ibm.com /aix/freeSoftware/aixtoolbox/INSTALLP/ppc/rpm.rte.

- Install the RPM utility using the following command:

```
# installp -qacXgd rpm.rte rpm.rte
```

- Upgrade the RPM utility using the command:

```
# rpm --upgrade rpm-3.0.5-37.aix5.1.ppc.rpm
```

- Verify the RPM utility installation by executing the following command:

```
# rpm --version
```

> ⓘ After executing these commands, ensure that the message "RPM version 3.0.5" is displayed on the screen.

Installing additional libraries under IBM AIX

Additional libraries must be installed in a superuser (root) session.

For the AIX 7.2 and AIX 7.1 platforms, the following library versions must be downloaded and installed :

- "bash" version 4.+;
- "expat" version 2.2.+;
- "gettext" version 0.+;
- "glib2" version 2.+;
- "unzip" version 6.+;
- "zlib" version 1.2.+;
- "libiconv" version 1.+;
- "libgcc" version 8.+ for AIX 7.2, "libgcc" version 6.+ for AIX 7.1.

All libraries can be downloaded from the vendor's site https://www.ibm.com/developerworks /aix/library/aix-toolbox/date.html.

To install these libraries, do as follows:

- Copy the library archives to a temporary directory on the server. Library archive file names have the following format: "<library name>-<version>-<release>.<architecture>.ppc.rpm"; for example, "expat-2.2.0-1.aix5.1.ppc.rpm".
- In the directory containing the "expat", "libgcc", "zlib", "bash", "unzip" and "libiconv" library distributions, execute the following commands:

```
# rpm –ivh expat-<version>-<release>.<architecture>.ppc.rpm
# rpm –ivh libgcc-<version>-<release>.<architecture>.ppc.rpm
# rpm –ivh libiconv-<version>-<release>.<architecture>.ppc.rpm
# rpm –ivh zlib-<version>-<release>.<architecture>.ppc.rpm
# rpm –ivh bash-<version>-<release>.<architecture>.ppc.rpm
# rpm –ivh unzip-<version>-<release>.<architecture>.ppc.rpm
```

- In the directory with the "gettext" and "glib2" library distributions, execute the following commands:

```
# rpm -Uvh gettext-<version>-<release>.<architecture>.ppc.rpm --nodeps
# rpm -U glib2-<version>-<release>.<architecture>.ppc.rpm --nodeps
```

Chapter 14. Accessing the Web Server through a Port up to 1024

To set up access to the "apache_24" application (when WAY4 Application Server version 1.7.1402 and later is used) or to the web server component (when a version of WAY4 Application Server earlier than 1.7.1402 is used) through ports with numbers up to 1024, the standard port redirection algorithm can be used in Linux and CentOS.

Linux 7 / CentOS 7

To set up web server access through HTTP port 80 and HTTPS port 443, proceed as follows (superuser (root) privileges are required to perform setup):

```
[root@server ~]# firewall-cmd --zone=public --add-masquerade --permanent
success
[root@server ~]# firewall-cmd --zone=public --add-forward-port=port=80:proto=tcp:
toport=8080 --permanent
success
[root@server ~]# firewall-cmd --zone=public --add-forward-port=port=443:
proto=tcp:toport=8443 --permanent
success
[root@server ~]# service firewalld restart
```

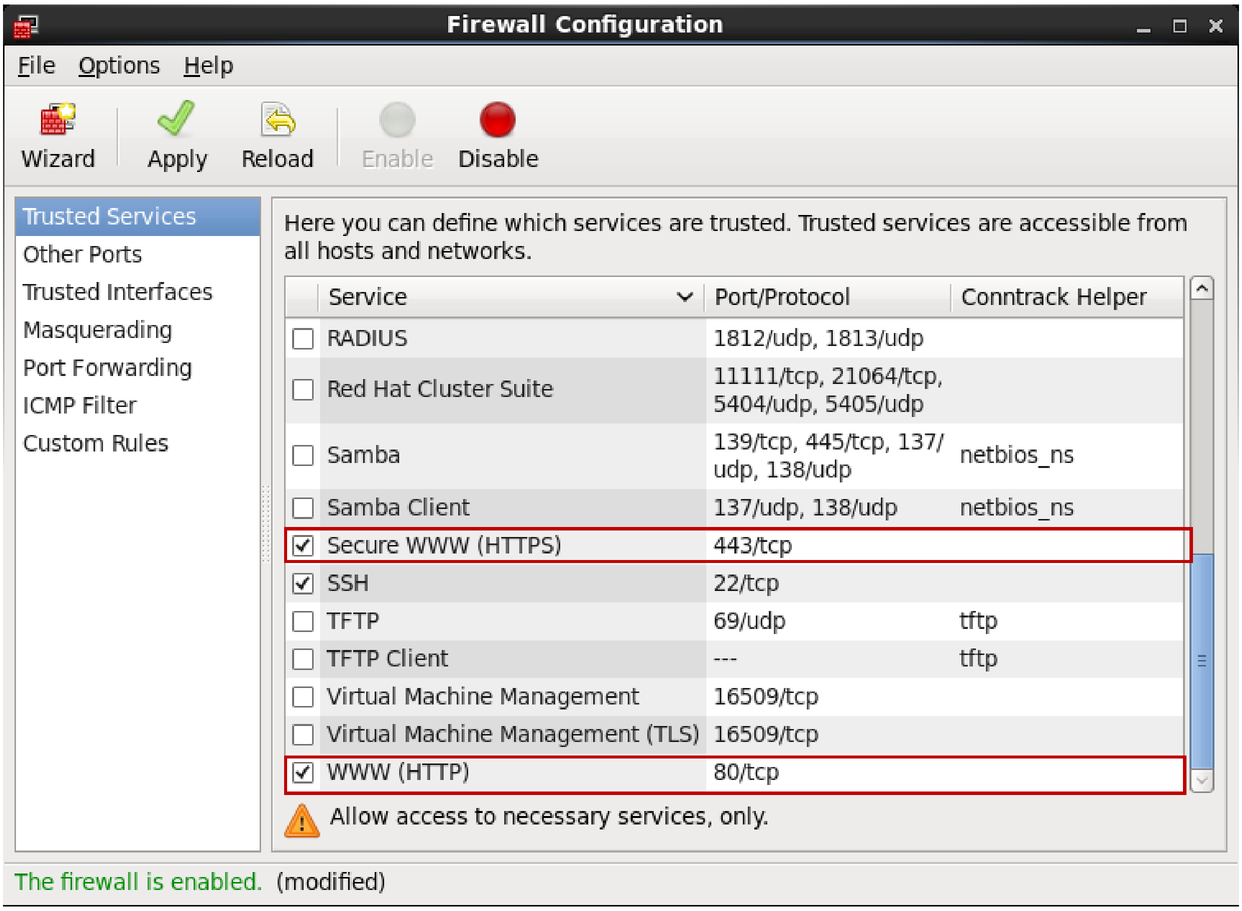
Linux 6

To set up web server access through HTTP port 80 and HTTPS port 443, proceed as follows (superuser (root) privileges are required to perform configuration):

- Execute the following command:

```
# system-config-firewall
```

- As a result, the "Firewall Configuration" window will be displayed. On the "Trusted Services" tab (in the left-hand part of the screen), set flags for the "Secure WWW (HTTPS)" and "WWW (HTTP)" services (see Fig. 27).



Here you can define which services are trusted. Trusted services are accessible from all hosts and networks.

	Service	Port/Protocol	Conntrack Helper
□	RADIUS	1812/udp, 1813/udp	
□	Red Hat Cluster Suite	11111/tcp, 21064/tcp, 5404/udp, 5405/udp	
□	Samba	139/tcp, 445/tcp, 137/udp, 138/udp	netbios_ns
□	Samba Client	137/udp, 138/udp	netbios_ns
☑	Secure WWW (HTTPS)	443/tcp	
☑	SSH	22/tcp	
□	TFTP	69/udp	tftp
□	TFTP Client	---	tftp
□	Virtual Machine Management	16509/tcp	
□	Virtual Machine Management (TLS)	16509/tcp	
☑	WWW (HTTP)	80/tcp	

Allow access to necessary services, only.

The firewall is enabled. (modified)

Fig. 27. Including "Secure WWW (HTTPS)" and "WWW (HTTP)" in the list of permitted services

In the left-hand part of the screen, select the "Port Forwarding" tab and in the window that appears to the right, click the [Add] button.

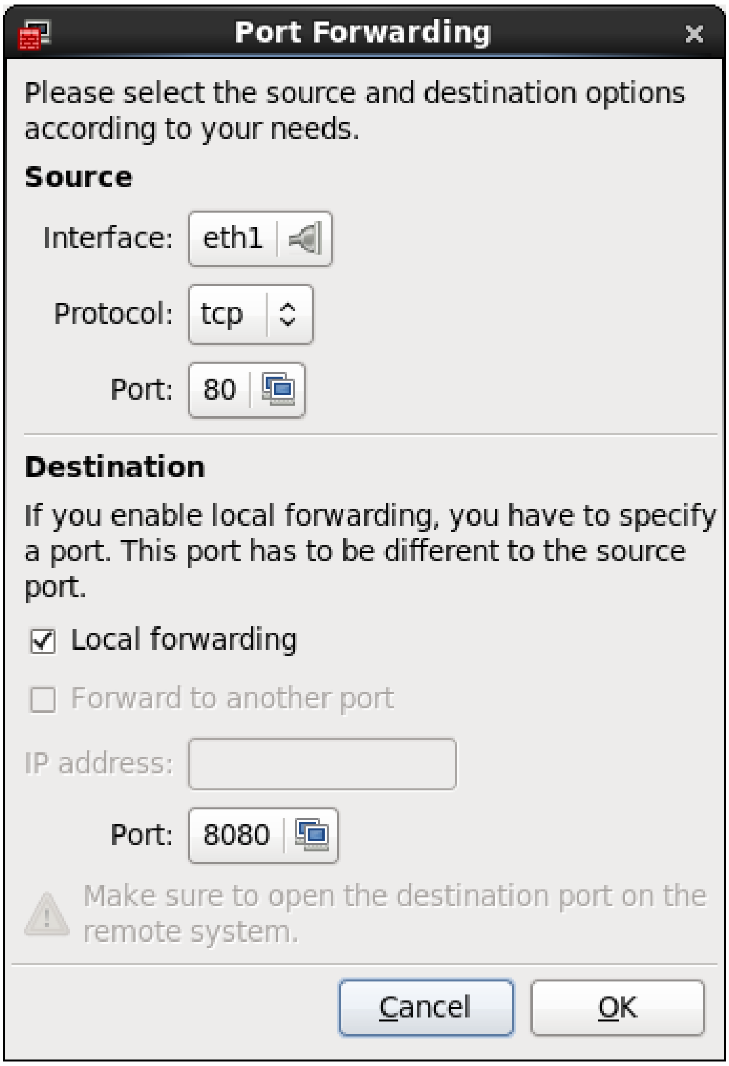
In the "Port Forwarding" form that opens, fill in the following fields to configure access to the web server component through HTTP port 80.

"Source" group:

- *Interface* – select the interface name from the list.
- *Protocol* – select the protocol name from the list.
- *Port* – select the value "80"from the list.

Set the "Local Forwarding" flag in the "Destination" group, and in the *Port* field, specify the port for redirecting requests, for example "8080". After filling in the fields, click [OK].

An example of the form is shown in Fig. 28.



Port Forwarding

Please select the source and destination options according to your needs.

Source

Interface: eth1

Protocol: tcp

Port: 80

Destination

If you enable local forwarding, you have to specify a port. This port has to be different to the source port.

☑ Local forwarding

□ Forward to another port

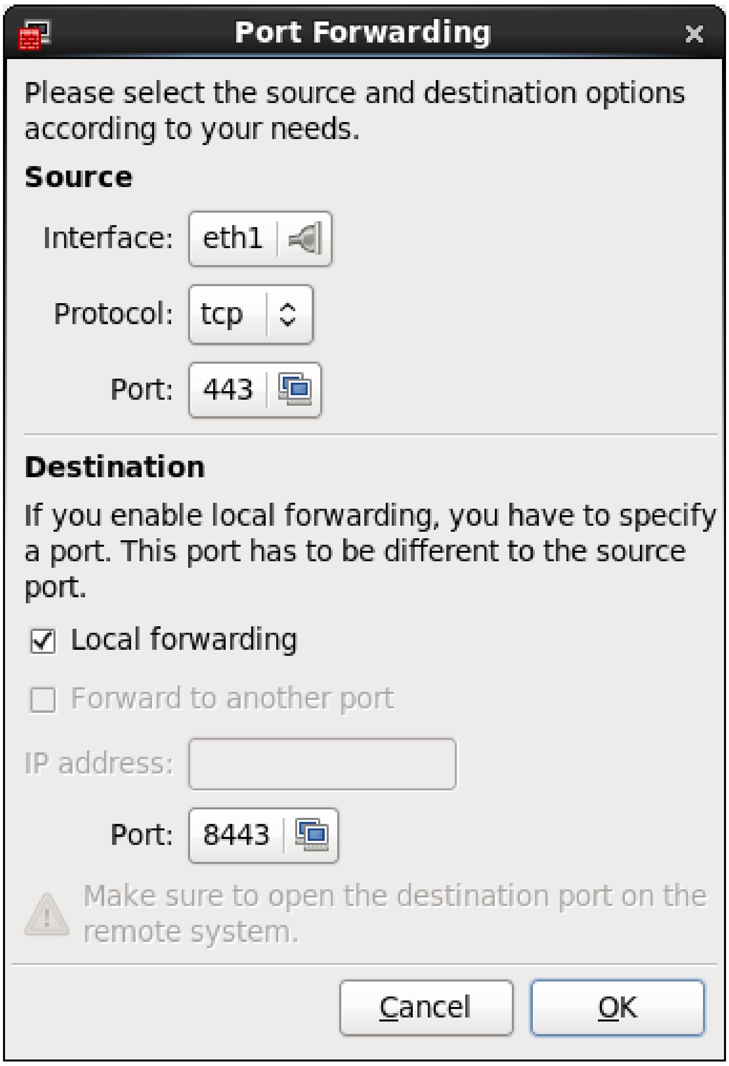
IP address:

Port: 8080

Make sure to open the destination port on the remote system.

Fig. 28. Access to the web server component through HTTP port 80

In the left-hand part of the "Port Forwarding" tab, click the [Add] button again and then fill in the fields to configure access to the web server component through HTTPS port 443. These fields are filled in as in the preceding item, with "443" and "8443" specified in the *Port* fields of the "Source" and "Destination" groups, respectively (see Fig. 29). After filling in the form, click [OK].



Port Forwarding

Please select the source and destination options according to your needs.

Source

Interface: eth1

Protocol: tcp

Port: 443

Destination

If you enable local forwarding, you have to specify a port. This port has to be different to the source port.

☑ Local forwarding

□ Forward to another port

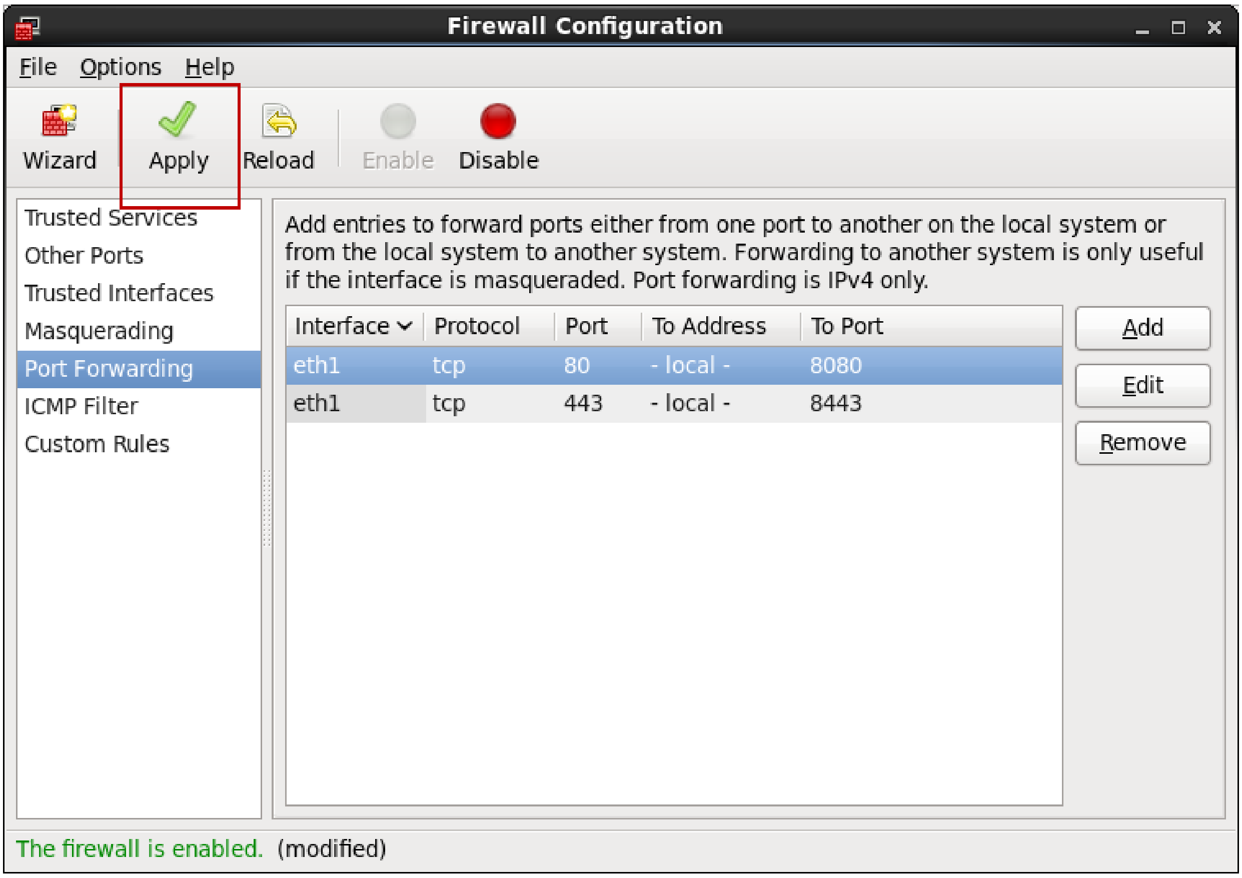
IP address:

Port: 8443

Make sure to open the destination port on the remote system.

Fig. 29. Access to the web server component through HTTPS port 443

In the "Firewall Configuration" window, click the [Apply] button (see Fig. 30) and then the [Yes] button in the "system-config-firewall" form (see Fig. 31).

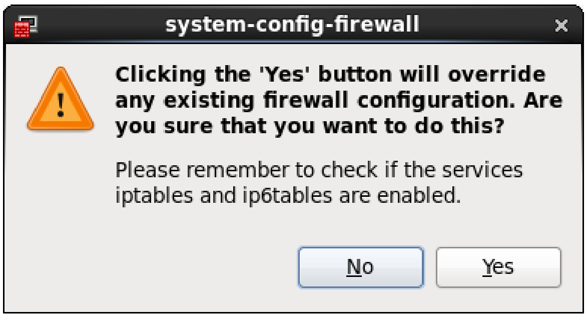


Add entries to forward ports either from one port to another on the local system or from the local system to another system. Forwarding to another system is only useful if the interface is masqueraded. Port forwarding is IPv4 only.

Interface	Protocol	Port	To Address	To Port
eth1	tcp	80	- local -	8080
eth1	tcp	443	- local -	8443

The firewall is enabled. (modified)

Fig. 30. Applying changes



Clicking the 'Yes' button will override any existing firewall configuration. Are you sure that you want to do this?

Please remember to check if the services iptables and ip6tables are enabled.

Fig. 31. Confirming changes in firewall configuration

Ensure that the local machine has access to the server through an HTTP port and HTTPS port.

Chapter 15. Port Use

Table 8 shows reference information about port use in WAY4 Application Server.

Table 8. Port use in WAY4 Application Server

Parameter name	Port number /range (default)	Description
WAY4 Application Server parameters		
tcp_management_port	32371	Number of the port for getting management commands from Command Line Utilities. File <AppServer_HOME>/conf/AppContainer. properties.
tcp_internal_port	32372	Number of the port for interaction with the Application Container component. File <AppServer_HOME>/conf/AppContainer. properties.
min_managed_port max_managed_port	15000-15100	Range of service ports for WAY4 applications. File <AppServer_HOME>/appserver/appcontainer /system/AppContainer.xml.
jmx_port	9999	Number of the port for interaction with JMX. File <AppServer_HOME>/conf/AppContainer. properties.
jolokia_http_port	32373	Jolokia (REST) HTTP port. Only works if the parameter "jmx_enabled=yes". File <AppServer_HOME>/conf/AppContainer. properties.
Application parameters		
port_number	19096	Port number for the server on which WAY4 Health Monitoring Gen2 is installed. File <AppServer_HOME>/applications/logagent/conf /config.properties.
core_jmx_server_port	6099	Number of the JMX port for the "monitoring" console utility to communicate with the "monitoring" application. The parameter value should only be changed if several instances of the "monitoring" application are installed on one WAY4 Application Server, or several instances of WAY4 Application Server are installed on one server (not recommended in a live installation). File <AppServer_HOME>/applications/monitoring/conf /config.properties.
		Port used to establish a connection between the "monitoring" and "monitoring_ui" system applications.

Parameter name	Port number /range (default)	Description
rmi_api_port	1099	File <AppServer_HOME>/applications/monitoring/conf /config.properties.
rmi_service_port	1098	Port used to transmit messages between the "monitoring" and "monitoring_ui" system applications via the SSL secure protocol. The port number must differ from the value of the "rmi_api_port". File <AppServer_HOME>/applications/monitoring/conf /config.properties.
snmp_nms_trap_ port	1162	Number of the port (or ports, separated by commas) for the server on which an external monitoring system is installed; the port is used to transmit trap messages to the external system. File <AppServer_HOME>/applications/monitoring/conf /config.properties.
snmp_agent_por t	1161	Number of the port for the server on which the "monitoring" application is installed; the port is used for getting get requests from an external monitoring system and for requests to confirm trap message delivery (when the "snmp_v2_confirm" parameter is set). File <AppServer_HOME>/applications/monitoring/conf /config.properties.
log_server_port_ number	19096	Number of the port for the server (with the "monitoring" application") to which the "logagent" application sends information about log files. This parameter's value must match the value of the "port_number" parameter specified in the configuration file of the "logagent" application. File <AppServer_HOME>/applications/monitoring/conf /config.properties.
log_server_mqs_ port	9999	Number of the port for the server on which WebSphere MQ queue manager is installed, used for connecting the "monitoring" application. File <AppServer_HOME>/applications/monitoring/conf /config.properties.
log_agent_mqs_ port	9999	Same as the "log_server_mqs_port" parameter. File <AppServer_HOME>/applications/logagent/conf /config.properties.
server_port http_port https_port ajp_port	9091 9090 9092 9093	Ports assigned to the "m2_web_console" and "monitoring_ui" applications. File <AppServer_HOME>/applications /<Application_Name>/conf/web-config.properties.
		Port assigned to the "console" application.

Parameter name	Port number /range (default)	Description
app.http.port	9085	Файл <AppServer_HOME>/applications/console/conf/application.conf
Web server parameters		
Listen	8080	Web server HTTP port. File <AppServer_HOME>/conf/webserver/httpd-main.conf (if any exists). When updating WAY4 Application Server from a version that is higher than 1.7.1402, the file <AppServer_HOME>/applications/apache_24/conf/webserver/httpd-main.conf.
Listen	8443	Web server HTTPS port used for a protected connection (for more information, see "Configuring the Web Server Component"). File <AppServer_HOME>/conf/webserver/httpd-ssl.conf.

More information about WAY4 Health Monitoring Gen2 ports and their numbers can be found in the document "Administering WAY4 Health Monitoring Gen2 (provided according to an additional agreement), Distributed Management Console ports are described in the document "WAY4™ Distributed Management Console" (provided according to an additional agreement).