



ADMIN NODE MANUAL

Admin Node Manual

Overview.....	2
Jenkins Jobs Definition	2
Deployment	2
Utilities.....	3
Administration.....	4
Nodes Group Parameter.....	5
Deployment Instruction.....	6
Terminology.....	6
Preliminary Steps.....	6
Deployment Actions	6
Deployment	6
Verification	7
Self-Deployment Training.....	9
Preliminary Steps.....	9
Deployment	9
Verification	9
Server Restart Instruction.....	11
Preliminary Steps.....	11
JBoss restart via Jenkins (admin node available)	11
JBoss restart via Linux console (no admin node available)	11
Graceful JBoss shutdown via Linux console (no admin node available)	11

Overview

The purpose of this document is to provide the user manual explaining how to perform tasks using the admin node. It contains descriptions of the operations and parameters available when the admin node is installed as well as the procedures that have to be executed to restart the server.

Jenkins Jobs Definition

Deployment

Includes operations used to deploy the application.

101 - Start JBoss – job used to start JBoss on the selected UniPay/UniBroker node(s). Used for updates and patches only.

- NODES_GROUP – allows selection of group of nodes which will be affected by the starting of the JBoss service. To learn more about the parameter, see [the section below](#).

102 - Stop Jboss – job used to stop JBoss on the selected UniPay/UniBroker node(s). Used for updates and patches, before the *Apply DB Changes* job.

- NODES_GROUP – allows selection of group of nodes which will be affected by the stopping of the JBoss service. To learn more about the parameter, see [the section below](#).

110 - Build Application – job used to download a branch from the GIT repository and build UniPay and UniBroker modules. Used for updates and patches only. All parameters (branch, profile name, etc) are set in *inventory* configuration file.

120 - Apply DB Changes – job used to update a current database. Used for updates and patches only.

- MIGRATION_SCHEMA – allows selection of how the system will migrate from the current database to the updated one. Used only for large databases, when the system has to be stopped for 20 or more minutes for the database to be updated.

Note: currently, only *no migration* radio button can be selected.

130 - Deploy Unipay – job used to deploy UniPay and its components to the selected node(s). Used for updates and patches only.

- NODES – allows selection of one or more nodes in UniPay cluster environment which will be affected by the redeploy occurring as part of this process.
- RESTART_JBOSS – indicates that JBoss will be stopped prior to deployment and started after that. Required if *REDEPLOY_UNIPAY* parameter is selected
- ARCHIVE_LOGS – indicates that all logs will be compressed by gzip after JBoss will be stopped, if *RESTART_JBOSS* parameter is selected.
- REDEPLOY_UNIPAY – indicates that UniPay will be redeployed during the deployment process using UniPay rebuild, which was built from sources downloaded from a repository. IF CHECKED OFF, RESTART_JBOSS MUST BE SELECTED.
- REDEPLOY_REPORTS – indicates that report templates will be redeployed during the deployment process.
- REDEPLOY_CAMEL – indicates that Camel will be redeployed during the deployment process.

- REDEPLOY_UTIL – indicates that additional utilities will be redeployed during the deployment process.
- REDEPLOY_UNIREAD – indicates that UniRead module (used for terminal transactions processing through the Cloud) will be redeployed during the deployment process.

140 - Deploy Unibroker – job used to deploy UniPay and its components to the selected node(s). Used for updates and patches only.

- NODES – allows selection of one or more nodes in the UniBroker cluster environment which will be affected by the redeploy occurring as part of this process.
- RESTART_JBOSS – indicates that JBoss will be stopped prior to deployment and started after it. Required if *REDEPLOY_UNIBROKER* parameter is selected.
- ARCHIVE_LOGS – indicates that all logs will be compressed by gzip after JBoss is stopped, if *RESTART_JBOSS* parameter is selected.
- REDEPLOY_UNIBROKER – indicates that UniBroker will be redeployed during the deployment process using UniBroker rebuild, which was built from sources downloaded from a repository. IF CHECKED OFF, RESTART_JBOSS MUST BE SELECTED.
- REDEPLOY_CAMEL – indicates that Camel will be redeployed during the deployment process.

Utilities

Includes operations used to diagnose and fix possible issues related to deployment/redeployment process.

200 - Ping Nodes – job used to check the connection between the admin and the selected node via SSH. Must be used prior to updates/patches, when a particular issue arises, or when there is a delay between the checks. Result of the ping is displayed in the Build Log.

- NODES_GROUP – allows selection of group of nodes which will be affected by the executed ping. To learn more about the parameter, see [the section below](#).

201 - Ping Nodes (sudo) – job used to check the connection between the admin and the selected node via SSH, as well as check [superuser](#) privileges availability. For example, if **200 - Ping Nodes** shows that everything is fine, you can use **201 - Ping Nodes (sudo)** to see whether there is a problem with privileges. Must be used prior to updates/patches, when a particular issue arises, or when there is a delay between the checks. Result of the ping is displayed in the Build Log.

- NODES_GROUP – allows selection of group of nodes which will be affected by the executed ping. To learn more about the parameter, see [the section below](#).

210 - Sync Node Resources – job used to synchronize resources between the selected nodes via [csync2](#) tool. The job runs *csync2 -xv* command on the selected nodes. When run, csync2 tool performs file reconciliation and necessary synchronization. Used for diagnostic purposes. By default, the job is run automatically, but can be run manually, if needed.

- NODES_GROUP – allows selection of group of nodes which will be affected by the executed synchronization. To learn more about the parameter, see [the section below](#).

220 - Clean up JBoss temp folders – job used to clean up legacy files and temporary folders.

- INCLUDE_ARCHIVE_FOLDER – indicates that the *archive* folder, which stores archived copies of merchants' digital resources, such as merchant statements, will be cleaned up.
- ARCHIVE_FOLDER – location of the *archive* folder. This parameter is read-only. If you need to change it, address the issue to the gateway support.

- INCLUDE_LOG_FOLDER – indicates that the *log* folder, which stores various log files, will be cleaned up.
- LOG_FOLDER – location of the *log* folder. This parameter is read-only. If you need to change it, address the issue to the gateway support.
- INCLUDE_WORK_FOLDER – indicates that the *work* folder, which stores temporary application files, will be cleaned up.
- WORK_FOLDER – location of the *work* folder. This parameter is read-only. If you need to change it, address the issue to the gateway support.
- STORAGE_PERIOD – maximum file age in days. Any files within the *log*, *work* and *archive* folders that are older than the specified number of days will be removed.

300 - Check for JDK Updates – job used to update Java Development Kit. The job verifies if JDK update is available in the Ubuntu repository (for Ubuntu) or distribution repository (for CentOS). If available, it downloads and installs the new version of JDK. Used each time the new java version is released and during updates/patches.

- NODES_GROUP – allows selection of group of nodes which will be affected by executed JDK updates check. To learn more about the parameter, see [the section below](#).

310 - Install JBoss – job used to install a new JBoss version or to re-configure the current one.

- NODES_GROUP – allows selection of group of nodes which will be affected by JBoss installation. To learn more about the parameter, see [the section below](#).

320 - Patch JBoss – job used to update a current JBoss version with available patch.

- NODES_GROUP – allows selection of group of nodes which will be affected by JBoss patching. To learn more about the parameter, see [the section below](#).
- PATCH_FILE – allows selection of a patch file that must be used for JBoss update.

330 - Execute JBoss CLI Script – job used to run a JBoss CLI script for configuring JBoss settings (for example, a number of available loggers and access policy for them) without stopping JBoss. Used in cases of emergency.

- NODES_GROUP – allows selection of group of nodes which will be affected by the executed JBoss CLI script. To learn more about the parameter, see [the section below](#).
- SCRIPT – content of the JBoss CLI script that should be executed.

340 - Add SSL Certificate to UniBroker – job used to add a certificate for UniBroker host to the truststore. After the certificate is added, JBoss has to be restarted. Note that if the host, specified in the certificate, does not correspond to the real one, UniBroker will not be run.

- CERTIFICATE_ALIAS – name of the certificate. Name of a certificate must be unique; usually, it corresponds to the host name. NOTE: If CERTIFICATE_ALIAS is set as *hostname:port* and CERTIFICATE_DATA is not specified, a certificate will be downloaded automatically from the specified *hostname:port*.
- CERTIFICATE_DATA – content of the certificate.

Administration

Includes operations used to modify Jenkins configuration file.

Restore Jenkins Config - restores a current Jenkins configuration.

Update Jenkins Config - updates Jenkins configuration from the repository, if an update is available.

Nodes Group Parameter

Jenkins jobs can be applied to either individual nodes or a group of nodes. Nodes groups are defined in the *inventory* file, which is used in conjunction with the *ansible scripts* for UniPay deployment.

The following groups are available at the moment:

- unipay-entry - UniPay master node entry;
- unipay - includes all UniPay nodes;
- unibroker-entry - UniBroker master node entry;
- unibroker - includes all UniBroker nodes;
- uniadmin-entry - UniAdmin node entry;
- uniadmin - UniAdmin node;
- jboss-entry-nodes - includes unipay-entry and unibroker-entry;
- jboss-nodes - includes all nodes where JBoss is installed (unipay-nodes and unibroker-nodes);
- java-entry-nodes - includes jboss-entry and uniadmin-entry;
- java-nodes - includes all nodes where Java is installed (jboss-nodes and uniadmin-node);
- nodes - includes all nodes in java-entry-nodes and java-nodes groups.

Note that all entry groups are used only for Ping Nodes jobs.

Deployment Instruction

Terminology

- outage – a period when the UniPay application is unavailable.
- hotfix - deployment of a fixed version of the application. The changes are made exclusively in Java, without SQL being touched. No outage is applied.
- patch - deployment of a fixed version of the application. Changes can be made both in Java and SQL (data modification in the database or creation of new SQL tables). A short-term outage (2-3 min) is possible.
- update - deployment of a new/enhanced version of the application. Changes are made both in Java and SQL (database structure is modified). A long-term outage (20-30 min) is required to complete the update.

Preliminary Steps

- 1) To perform hotfix/patch/update, you must go through the corresponding training, conducted by the gateway support. If the training is needed, request it via the designated [form](#). Make sure that you have specified the desired deployment level (hotfix/patch/update) in the request. Be advised that trainings can take up to 1-2 weeks.
- 2) Every time a new hotfix/patch/update is available, you receive an email with the release notification, which contains necessary technical information. Make sure you have received such an email before the deployment. If you have not received the email notification about the expected employment, request it from the gateway support.
- 3) Confirm the hotfix/patch/update deployment time with UT team. This is needed for cases when an outage is going to be present so UT team will not react to the notifications about server being unavailable.

After preliminary steps are done, the server can be deployed.

Deployment Actions

During the hotfix/patch/update, build and deployment of the application are performed. As a part of the deployment, certain actions may be required from your end for hotfix/patch/update to be made correctly. These actions can be pre- or post-deployment. For example:

- **Pre-update action** - execute a particular CLI script to skip additional node restart;
- **Post-update action** - modify [provider-profile-defaults.properties](#) file and replace all \$\$ characters with ss.

Deployment

For a hotfix:

- 1) Update the admin node via the [Administration/Update Jenkins Config](#) Jenkins job.
- 2) Run build via the [Deployment/110 - Build Application](#) Jenkins job.

- 3) Complete pre-update actions if available (a list of pre-update actions is provided by the UT team – if nothing is provided, there is no pre-update actions).
- 4) Deploy UniPay via the [Deployment/130 - Deploy Unipay](#) job. All parameters (except REDEPLOY_UNIREAD and REDEPLOY_UTIL, if Cloud and Util are not used within the system) have to be checked off.
- 5) Deploy UniBroker via the [Deployment/140 - Deploy Unibroker](#) job. All parameters have to be checked off.
- 6) Complete post-update actions if available (a list of post-update actions is provided by the UT team - if nothing is provided, there is no post-update actions).

For a patch/update:

- 1) Update the admin node via the [Administration/Update Jenkins Config](#) Jenkins job.
- 2) Run build via the [Deployment/110 - Build Application](#) Jenkins job.
- 3) Complete pre-update actions if available (a list of pre-update actions is provided by the UT team - if nothing is provided, there is no pre-update actions).
- 4) Stop the JBoss service on all unipay and unibroker nodes via the jenkins [Deployment/102 - Stop Jboss](#) job.
- 5) Run [Deployment/120 - Apply DB Changes](#) Jenkins job.
- 6) Deploy UniPay via the [Deployment/130 - Deploy Unipay](#) job. All parameters (except REDEPLOY_UNIREAD and REDEPLOY_UTIL, if Cloud and Util are not used within the system) have to be checked off.
- 7) Deploy UniBroker via the [Deployment/140 - Deploy Unibroker](#) job. All parameters have to be checked off.
- 8) Complete post-update actions if available (a list of post-update actions is provided by the UT team - if nothing is provided, there is no post-update actions).

Verification

When deployment is done, make sure that changes have been applied successfully and the server is updated to certain version and revision. To do this, run the application, open the [System](#) form on the [System perspective](#) and check whether the values specified in the email received prior to hotfix/patch/update are present within the form.

For a hotfix:

To verify that deployment has been completed successfully, check the [Implementation-Version](#) field via the System form on the System perspective. Deployed version of the application has to be present in this field in the format [release].[revision]-[date], for example, 7.1.1911296a34-b20180503.

For a patch/update:

To verify that deployment has been completed successfully, check the [Version](#) and [Implementation-Version](#) fields via the System form on the System perspective. Deployed version of the application has to be present in the [Implementation-Version](#) field in the format [release].[revision]-[date], for example, 7.1.1911296a34-b20180503. The version of the last applied database delta has to be present in the [Version](#) field, for example, 31288.

Status Information

System	
ZK Framework	5.0.8
OS Name	Linux
OS Version	4.4.0-116-generic
OS Architecture	amd64
JAVA VM	25.162-b12
JAVA Runtime	1.8.0_162-b12
Node Code	1
SQL Dialect	MySQLInnoDBDialect
SQL Version	5.6.33-0ubuntu0.14.04.1
Host	dev-manual.local
Database	
Version	31288
DB Name	unipay
DB Username	unipayw@localhost
DQM Username	unipayr@localhost
Info	
Verification Session Date	

MANIFEST.MF (101-imerchant-lib.jar)	
Build-Data	
Profile-Name	jenkins
OS-Version	4.4.0-116-generic
SVN-Author	
SVN-Date	2018-05-03T17:50:30+03:00
OS-Architecture	amd64
OS-Username	jenkins
OS-Name	Linux
OS-BuildTime	05/03/2018 18:00:30:030 EEST
OS-Hostname	\$(env.HOSTNAME)
SVN-Path	origin/dev
SVN-Revision	1911296a34
Vendor-Data	
Implementation-Version	7.1.1911296a34-b20180503
Specification-Version	7.1
Specification-Vendor	United Thinkers LLC
Implementation-Vendor	United Thinkers LLC

Memory	
Used	834,573,888
Free	171,010,496
Total	1,005,584,384
Maximum	1,005,584,384

permanent refresh

refresh

Reset Perspective State Cache

Reset Merchant and User Cache

Reset Permission Cache

Reset Response Codes Cache

Reset Terminal Cache

Reset SQL Increment Generator

Run the garbage collector

Set to [false] - last date update

Set to [true] show comment

File Path Validation

Reset Hibernate Query Plan Cache

Self-Deployment Training

Preliminary Steps

- 1) Request the self-deployment training via the designated [form](#). Make sure that you have specified the desired deployment level (hotfix/patch/update) in the request. Be advised that trainings can take up to 1-2 weeks.
- 2) Confirm the time of the self-deployment training with UT team.

After preliminary steps are done, the training can be started.

Deployment

For a hotfix:

- 1) Update the admin node via the [Administration/Update Jenkins Config](#) Jenkins job.
- 2) Run build via the [Deployment/110 - Build Application](#) Jenkins job.
- 3) Deploy UniPay via the [Deployment/130 - Deploy Unipay](#) job. All parameters (except REDEPLOY_UNIREAD and REDEPLOY_UTIL, if Cloud and Util are not used within the system) have to be checked off.
- 4) Deploy UniBroker via the [Deployment/140 - Deploy Unibroker](#) job. All parameters have to be checked off.

For a patch/update:

- 1) Update the admin node via the [Administration/Update Jenkins Config](#) Jenkins job.
- 2) Run build via the [Deployment/110 - Build Application](#) Jenkins job.
- 3) Stop the JBoss service on all unipay and unibroker nodes via the jenkins [Deployment/102 - Stop Jboss](#) job.
- 4) Run [Deployment/120 - Apply DB Changes](#) Jenkins job.
- 5) Deploy UniPay via the [Deployment/130 - Deploy Unipay](#) job. All parameters (except REDEPLOY_UNIREAD and REDEPLOY_UTIL, if Cloud and Util are not used within the system) have to be checked off.
- 6) Deploy UniBroker via the [Deployment/140 - Deploy Unibroker](#) job. All parameters have to be checked off.

Verification

When deployment is done, make sure that changes have been applied successfully and the server is updated to certain version and revision. To do this, run the application, open the [System](#) form on the [System perspective](#) and check whether the values specified in the email received prior to hotfix/patch/update are present within the form.

To verify that self-deployment has been completed successfully, check the following fields:

- **Implementation-Version** field. Deployed version of the application has to be present in this field in the format [release].[revision]-[date], for example, *7.1.133f72eb6b-bug-b20180919*.

- **Practice Data** field. Sequential number of a hotfix/patch/update has to be present in this field, for example, 1 / 1.

GATEWAY System

Statistics | Vel. Tracking | Keys | Jobs | Files | Reporting | Codes | IQ | Branding | Agreements | Logs | Sanitization | Providers | System | Status | Settings | Diagnostics

System Information

System	
ZK Framework	5.0.8
OS Name	Linux
OS Version	4.4.0-135-generic
OS Architecture	amd64
JAVA VM	25.131-b11
JAVA Runtime	1.8.0_131-b11
Node Code	1
SQL Dialect	MySQLInnoDBDialect
SQL Version	5.7.22-log
Host	

Database	
Version	31421
DB Name	
DB Username	
DQM Username	

Info	
Verification Session Date	
Practice Data	1 / 1

MANIFEST.MF (101-imerchant-lib.jar)	
Build-Data	
OS-Architecture	amd64
OS-BuildTime	09/19/2018 18:04:00:000 EEST
OS-Hostname	
OS-Name	Linux
OS-Username	
OS-Version	4.4.0-135-generic
Profile-Name	jenkins-cub
SVN-Author	
SVN-Date	2018-09-12T18:37:37+03:00
SVN-Path	origin/dev-bug-4342-2
SVN-Revision	13372eb6b
Vendor-Data	
Implementation-Vendor	
Implementation-Version	7.1.13372eb6b-bug-b20180919
Specification-Vendor	
Specification-Version	7.1

Memory	
Used	542,002,776
Free	352,432,552
Total	894,435,328
Maximum	954,728,448

Permanent Refresh
 Terminal Activity Trace Refresh

Reset Cache

Actions

Lock Info

server locks:

my locks:

Current server time is: 10/09/2018 02:15:47 AM EEST (GMT+0300) - Eastern European Summer Time

Server Restart Instruction

The procedures below describe how JBoss has to be restarted. Other approaches can lead to the data loss and malfunctioning of the admin node services after restart is done.

Preliminary Steps

If there is a suspect of the memory leak in JBoss, follow the steps below. Otherwise, please skip them. To verify that memory leak is present, check the logs (search for the *OutOfMemoryError*) or the corresponding diagrams in the JavaMelody service. If present, follow these steps before the server restart:

- 1) Run the command to generate classes taking the most memory:
`jcmd $PID GC.class_histogram > /tmp/class_histogram.txt`
- 2) Run the command to generate the heap dump:
`jcmd $PID GC.heap_dump /tmp/heap_dump.hprof`
- 3) Run the command to generate the thread dump:
`jcmd $PID Thread.print >/tmp/threads.txt`
- 4) Archive generated files and send to the gateway support team for further analysis.

JBoss restart via Jenkins (admin node available)

- 1) Select **Manage Services (UniBroker)** or **Manage Services (UniPay)** respectively under the **Management** tab.
- 2) Check off the nodes that should have JBoss restarted under the **Nodes** section.
- 3) Check off **unibroker-jboss7** or **unipay-jboss7** respectively under the **Services** section.
Note: Please contact support if another JBoss version is used in your configuration.
- 4) Select **Restart** under the **Action** section.
- 5) Click **Build** button.

JBoss restart via Linux console (no admin node available)

- 1) To restart JBoss on UniPay particular node, run the command below:
`service unipay-jboss7 restart`
- 2) To restart JBoss on UniBroker particular node, run the command below:
`service unibroker-jboss7 restart`

Graceful JBoss shutdown via Linux console (no admin node available)

- 1) To stop JBoss, run the command below:
`jboss-cli.sh --connect --command=":shutdown"`
- 2) To start JBoss, run the command specific to the daemon that you have set up for JBoss.