Upgrade instance

Upgrade

Upgrading **ActiveServer** is a simple process that only requires the `as.jar` to be replaced with the new version.

To ***upgrade*** your existing **ActiveServer** instance to the latest version:

1. Stop the **ActiveServer** instance node.
2. Open the **ActiveServer** directory and back up the old `as.jar` file (simply copy or save in an archive) in case a roll back is required.
3. Back up the **ActiveServer** database (refer to your database documentation for back up processes and requirements specific to your database).
4. Download and extract the new **ActiveServer** package into a temporary directory and copy the `as.jar` file to your **ActiveServer** directory.
5. Start up the **ActiveServer** instance node. **ActiveServer** will automatically upgrade the database as necessary during the startup. Then the upgrade process is complete.

> **Clustering environment upgrade**
>
> In a clustering environment where multiple ActiveServer nodes are deployed with the same database, the database upgrade process will be handled automatically so that only one node at a time can migrate the database. The startup process of the rest of the nodes in the cluster will be blocked until the database migration is done by the first node.

Upgrading to v4.0.0 or above

> **Install Java 25 before upgrading**
>
> **ActiveServer** v4.0.0 requires Java 25. Install Java 25 on every node **before** replacing `as.jar` . On an earlier Java version the instance will not start.

Copyright ©2019 GPayments Pty Ltd. All rights reserved.

Rollback version

If you need to roll back after upgrading and you are currently running v3.2.0 or earlier, roll back to v3.2.1. If you are running a later version, roll back to your previous version.

1. Install Java 25 on each node and confirm that `java -version` reports version 25 for the user that runs **ActiveServer**.
2. Replace `startup.sh` / `startup.bat` with the scripts from the new package, then re-apply any customisations (for example `AS_PROFILES`, `-Xmx`, proxy settings). The new scripts check the Java version before starting.
3. Follow the upgrade steps above.

Rollback

If you need to roll back to the previous version, keep the Java 21 installation on each node so the instance can be started on the earlier Java version.

Upgrading to v1.3.3 or above

From v1.3.3, **ActiveServer** supports **DS Profiles** which allows a single instance to connect to both card scheme production Directory Servers and the GPayments TestLabs Directory Server. However, there are two important updates that are required for clients to review when you upgrade your **ActiveServer** to v1.3.3 or above:

- Adding `trans-type=prod` to API requests in the 3DS Requestor
- Configuring additional ports for the GPayments TestLabs to access

Important: No existing Directory Server settings or certificates will be changed during this upgrade.

Transaction type parameter

When you upgrade **ActiveServer** to v1.3.3, by default all Auth API requests will be sent to the GPayments TestLabs directory server, for testing purposes. When moving into production, to send the API request to the card scheme directory server, the **`trans-type` query parameter** must be appended to the API URL.

Copyright ©2019 GPayments Pty Ltd. All rights reserved.

This parameter is required for the `InitAuth` and `Auth` processes if you would like to send the authentication requests to the production Directory Server. If this parameter is not provided, the authentication requests will be sent to the GPayments TestLabs.

For example, `InitAuth` will be called with following URL `https://api.testlab.3dsecure.cloud/api/v2/auth/brw/init?trans-type=prod`.

> **Update 3DS requestor first!**
>
> **Important:** If you are already in production, to ensure the upgrade process proceeds smoothly the `trans-type=prod` parameter can be added to your existing 3DS Requestor code before the upgrade without any issues. This will ensure that all existing implementations will send API requests to the production DS that is already configured as soon as the upgrade takes place.
>
> Not adding the `trans-type=prod` to the 3DS requestor before upgrading to v1.3.3 will result in all requests going to the TestLabs Directory Server after the upgrade.
>
> If you are currently only using the TestLabs, then this step is not required, as API requests will continue to be sent to the TestLabs after upgrading.

Configure GPayments TestLabs

If ongoing access to the TestLabs is required, additional ports must be opened for TestLabs DS communications. This will allow testing with TestLabs in parallel with using the production card scheme Directory Servers. The additional ports should be externally accessible from the GPayments TestLabs Directory Server. The settings should be added to your `application-prod.properties`, with example properties shown below:

Copyright ©2019 GPayments Pty Ltd. All rights reserved.

```
##------------------------------------------------------------------
## GPayments TestLabs DS ports
##
## DS ports used to connect from GPayments TestLabs directory servers
##------------------------------------------------------------------

## Visa TestLabs
as.testlab.visa.port=9800
## Set to false to disable DS HTTPS listening port
# as.testlab.visa.enabled=false

## Mastercard TestLabs
as.testlab.mastercard.port=9801
## Set to false to disable DS HTTPS listening port
# as.testlab.jcb.enabled=false

## American Express TestLabs
as.testlab.amex.port=9802
## Set to false to disable DS HTTPS listening port
# as.testlab.amex.enabled=false

## Discover TestLabs
as.testlab.discover.port=9803
## Set to false to disable DS HTTPS listening port
# as.testlab.discover.enabled=false

## JCB TestLabs
as.testlab.jcb.port=9804
## Set to false todisable DS HTTPS listening port
# as.testlab.jcb.enabled=false
```

For full information on setting the `application-prod.properties`, refer to the quickstart guide.

Tip

After upgrading to v1.3.3, you can safely remove all DS settings and certificates related to the GPayments TestLabs. Alternatively, this step can be completed gradually as you replace the settings with the production settings provided by the card schemes.

Copyright ©2019 GPayments Pty Ltd. All rights reserved.

Rollback

In case you need to ***roll back*** to your previous version of **ActiveServer**:

1. Stop the **ActiveServer** instance node.
2. Open the **ActiveServer** directory and back up the old `as.jar` file (simply copy or save in an archive) if required.
3. Restore your previous `as.jar` into the **ActiveServer** directory.
4. Restore the **ActiveServer** database that you have previously backed up.
5. Start **ActiveServer**.

Copyright ©2019 GPayments Pty Ltd. All rights reserved.