

# Role STORE - Install Applications and Data (Debian/Win2019)

## Installation of required additional software

### Install GIT

Git is required, to checkout the software and the default mysql-client is required (for running the update scripts).

If you are installing STORE on a machine which will contain the role DATA too, then do not install the default mysql client, as this will collide with MySQL. Instead, install the MySQL software first (see role DATA).

```
apt-get -y install git default-mysql-client
su jtel
cd
git config --global user.name "jtel Support"
git config --global user.email "support@jtel.de"
git config --global http.postBuffer 1048576000
```

### Configure GIT proxy server (optional)

If you need a proxy for GIT:

```
git config --global http.proxy http://<proxy_server>:<port>
```

## Downloading the jtel software for the first time

**Note:** for redundant systems, do this on the **ACTIVE** storage node.

### Change to jtel User

The following operations are no longer performed in the context of the *root* user but **must** be performed in the context of the *jtel* user. To do this, you can either log in as user *jtel* in a separate SSH session or, if you are already logged in as user *root*, you can switch to the context of the *jtel* user. This is done with the following command:

```
su jtel
cd
```

### Clone jtel Software

Next, the jtel Software is cloned. This will be done in 2 steps, as a user name and password is required for each one.

```
cd /srv/jtel/shared
git clone https://bitbucket.org/jtelgmbh/jtel.git JTEL
```

```
cd /srv/jtel/shared
git clone https://bitbucket.org/jtelgmbh/jtelcarrierportal.git JTELCarrierPortal
```

```
cd /srv/jtel/shared
git clone https://bitbucket.org/jtelgmbh/jtel-system-java-services.git jtel-system-java-services
```

## Clone Customer Software

If specific software has been programmed for the customer, the customer-specific software repository must also be downloaded. This is done by a command which is structured as follows, for the example customer acme:

### Download a custom repository

```
cd /srv/jtel/shared
git clone https://bitbucket.org/jtelgmbh/cacme-software.git acme
```

## Change to required Software Release

In the downloaded software repositories, the software release to be used must now be selected. In this case it is release 3.29:

### Select the desired release

```
cd /srv/jtel/shared/JTELCarrierPortal
git config pull.rebase false
git checkout release/stable-3.32

cd /srv/jtel/shared/JTEL
git config pull.rebase false
git checkout release/stable-3.32

cd /srv/jtel/shared/jtel-system-java-services
git config pull.rebase false
git checkout release/stable-3.32

# For customer specific software only
# cd /srv/jtel/shared/acme
# git checkout develop
cd /srv/jtel/shared
```

Note, that the customer specific release is usually develop, as customer releases are not branched or tagged when the jtel Software is branched or tagged.

## Create Required Directories

The last step is to create the directories for central logging, standard data import and the main data directory. This uses a basic data directory download which fits the Init.sql script (used later in role DATA).

```
cd /srv/jtel/shared
mkdir -p LogFiles LogFilesCall Import/{Clients,ServiceNumbers}/{Done,In,Problems}
wget https://cdn.jtel.de/downloads/Data/Data.zip
unzip Data.zip
rm -f Data.zip
cp -a JTELCarrierPortal/Data .
cp -a JTEL/Data/system/gui Data/system
cd /srv/jtel/shared/JTELCarrierPortal/Update
bash ./get_binaries.sh
```

## Cleaner processes

A CRON job is used to clean up the directories of the portal.

**With redundancy, configure this on both STORE.** Since the directory is only mounted on one, but we don't know which one is active, the command must be executed on both.

Note: execute as **ROOT**.

### Cleaner processes in /etc/cron.daily

```
# Make Directory
mkdir -p /home/jtel/jtel_dailyjobs

# Does the dailyjobs script exist? If not, create.
if [ ! -f /home/jtel/jtel_dailyjobs/jtel_dailyjobs.sh ]; then
# No - so create it
cat << EOF > /home/jtel/jtel_dailyjobs/jtel_dailyjobs.sh
#!/bin/bash
EOF
# Create cron script
cat << EOF > /home/jtel/jtel_dailyjobs/jtel_dailyjobs
#!/bin/bash
su - jtel -c '/home/jtel/jtel_dailyjobs/jtel_dailyjobs.sh'
EOF
# Link to cron.daily
ln -s /home/jtel/jtel_dailyjobs/jtel_dailyjobs /etc/cron.daily
# Make executable
chmod +x /home/jtel/jtel_dailyjobs/jtel_dailyjobs
chmod +x /home/jtel/jtel_dailyjobs/jtel_dailyjobs.sh
fi

# Create the cleaner script
cat << 'EOF' > /home/jtel/jtel_dailyjobs/jtel_portal_cleaner.sh
#!/bin/bash
find /srv/jtel/shared/Data/clients/*/reports -type f -mtime +2 -delete
EOF

# Make executable
chmod +x /home/jtel/jtel_dailyjobs/jtel_portal_cleaner.sh

# Append job to jtel_dailyjobs.sh
cat << EOF >> /home/jtel/jtel_dailyjobs/jtel_dailyjobs.sh
bash /home/jtel/jtel_dailyjobs/jtel_portal_cleaner.sh
EOF

# Change ownership of everything to jtel
chown -R jtel:jtel /home/jtel/jtel_dailyjobs
```