SYSTEM ADMINISTRATION GUIDE

Release 9.2-MP3 (October 25, 2019)

StiboSystems
MASTER DATA MANAGEMENT

Table of Contents

System Administration

There are many administrative tasks that are needed to keep a STEP system running at optimum levels. This section of the documentation introduces those responsible to the common tasks, challenges, and issues that often occur when working with administration of a complete STEP system and infrastructure.

Some of the information presented may not be applicable to all systems. Any questions should be directed to your Stibo Systems' representative or the Stibo Systems' Technical Support department.

- Installing STEP

Installing STEP

The installation of a STEP system is comprised of multiple components:

- An Oracle 18c RDBMS server software installation is recommended
- A STEP database
- An installation of the Java Runtime Environment that runs the application
- STEP 9.2 or an Apache HTTP Server 2.4 installation

Important: The complete list of platform and software support is available in the **9.2 Platform and Software Support** section of the **System Release and Patch Notes** documentation.

To install a STEP system you must perform the following steps:

1. Review the system requirements. These requirements can be found in the **System Requirements** section of the documentation.
2. Run all required pre-installation tasks. These tasks are detailed in the **Pre-Installation Tasks** section of the documentation.
3. Run the installation script (install.pl).

Before proceeding with the installation, however, an installation scenario must be decided on that best fits the client's needs. The two primary questions that must be answered are: should you use a Single or a Clustered Application Server setup, and who needs to be involved in the setup?

Single vs. Clustered Application Server setup

Determine whether the system being installed consists of one or more Application Servers. The installation must be performed on all servers in the setup. For more information, see the **Pre-Installation Tasks** section of the documentation.

Who Needs to be Involved in the Installation?

When running the installation script to install a STEP system (including all third-party software) a number of pre-installation steps are performed before actually installing the software itself.

The pre-installation steps require that the user running the installation scripts is a privileged user. This might be an issue on Linux where certain kernel parameters must be set, and OS groups and users must be created. On Windows, the STEP Installer must be a member of the Administrator group. For more information, see the **System Requirements** section of the documentation.

System Requirements

The system requirements for running STEP on a host machine largely depends on the operating system.

The supported operating systems include:

- Linux
- Windows
- MacOS

The complete list of platform and software support is available in the **9.2 Platform and Software Support** section of the **System Release and Patch Notes** documentation.

Linux System Requirements

The supported operating systems are:

- Red Hat Enterprise Linux 7.5+ or 6.10 64-bit
- Oracle Enterprise Linux 7.5+ or 6.10 64-bit

Important: All Red Hat Enterprise Linux / Oracle Enterprise Linux 6 versions will be desupported with the 9.3 release.

Unless explicitly mentioned, OS prerequisites are implemented by the installation scripts (when run as root or with sudo on Linux).

The install scripts will (if needed):

- Create OS user(s) and group
- Adapt kernel parameters
- Install required RPMs if the server has a connection to Red Hat Network (RHN)

Important: The Infrastructure Recommendations guide provides more detailed information regarding requirements.

User Accounts

The OS user accounts include:

User Account Name	Must Be Member of Group(s)	Description
oracle	dba	Account must be created on the database server and must use the Bash shell. The user is the owner of the Oracle database software. If the user and/or group does not exist, the installation scripts will automatically create them. Group ID (GID) default: 501 User ID (UID) default: 501

User Account Name	Must Be Member of Group(s)	Description
stibosw	stibosw	Account must be created on the application server(s) and must use the Bash shell. The user is the owner of the STEP application software. If the user and/or group does not exist, the installation scripts will automatically create them. Group ID (GID) default: 500 User ID (UID) default: 500

Note: The installation scripts are implemented and tested using the Bash shell. Therefore, it is important that this is used if pre-creating the OS user accounts.

Server Setup

In a single server setup, with the database and application server running on one machine, both accounts must be created. This is often the case for test systems.

As indicated in the table above, there might be more than one application server, such as in the case of a clustered system setup. In these instances, the user account must exist on all application servers.

Service Accounts vs Logon Accounts

If security policies require separation of service and logon accounts, two separate accounts must be created. The service account password may never expire and must not be changed.

User Account Names and IDs

User account names do not need to match what is shown in the table. If required, create account names that adhere to corporate standards.

Group and user IDs must adhere to corporate standards if the default values are not compliant. Properties for these include:

- user.db (username)
- user.db-id
- user.db-primary-group-id
- user.app (username)

- user.app-id
- user.app-primary-group-id

Note: User names are case sensitive.

Windows System Requirements

The supported operating systems are:

- Windows Server 2016 64-bit (Standard or Datacenter Edition)
- Windows Server 2012 R2 64-bit (Essentials, Standard, or Datacenter Edition)

The following accounts are required for a complete STEP installation. The user accounts can either be local accounts (recommended) or domain accounts. The requirements below are valid in both scenarios.

User Account Name	Must Be Member of Group(s)	Description
`oracle`	`Administrators` `Remote Desktop Users`	Account must be created on the database server. This user is the owner of the Oracle database software. If this user does not already exist, it will automatically be created by the installation (local user).
`stibosw`	`Administrators` `Remote Desktop Users`	Account must be created on the application server(s). This user is the owner of the STEP application and HTTP server software. If this user does not already exist, it will automatically be created by the installation (local user).

Password Properties

Because both users act as logon users for Windows services, it is important that the password does not change or expire. Doing so would prevent the required services from starting.

Single Server Setup

In a single server setup, with the database and application server running on one machine, both accounts must be created. This is often the case for test systems.

As indicated in the table above, there might be more than one application server, such as in the case of a clustered system setup. In these instances, the user account must exist on all application servers.

Service Accounts vs Logon Accounts

If security policies require separation of service and logon accounts, two separate accounts must be created. The service account password may never expire and must not be changed.

Windows OS Services

The installer package is dependent on the 'Windows Management Instrumentation' Windows Service being enabled and started.

MacOS System Requirements

The supported operating systems are:

- MacOS X 10.14
- MacOS X 10.13

User Account Name	Must Be Member of Group(s)	Description
stibosw	stibosw	Account must be created on the DTP server. This user is the owner of the DTP server software.

A server running MacOS X is only required if there is a need for running Adobe InDesign Server on MacOS.

Installation Prerequisites

Before installing a STEP system, the following software must be available on the servers:

- OpenJDK / AdoptJDK Java 1.8.0 (JDK or JRE) 64-bit
- Perl 64-bit
- SPOT foothold distribution file (Stibo software)

Java

The STEP application requires AdoptOpenJDK 1.8 64-bit to run. As part of the STEP deployment (patch procedure), AdoptOpenJDK 1.8 JDK will be downloaded and installed. This Java version is regularly tested and released with the STEP application, but the SPOT tool itself also requires Java. Therefore, OpenJDK / AdoptOpenJDK Java must be installed manually on the servers in order to download and install third-party software and the STEP application.

Java for Windows

For Windows, download AdoptJDK Java JRE 1.8 64-bit from https://adoptopenjdk.net/. It must be 64-bit.

Follow the steps below to install AdoptJDK Java on Windows:

1. Login with an Administrator account
2. Download AdoptOpenJDK binaries (64-bit) and save it to any directory (see the above URL)
3. Unzip the binaries in the chosen directory (E: is used in the below example commands) by running the following commands in command prompt:

   ```
   unzip -d E:\java OpenJDK%U-jdk_x64_windows_hotspot_%u%b%.zip>
   ```

4. Create a link for the newly unzipped Java binaries to allow for easy update when a newer version is installed:

   ```
   mklink /d E:\java\jre.current E:\java\jdk%u%-b%-jre
   ```

5. Add the following to the System Environment variable PATH:

   ```
   E:\java\jre.current\bin
   ```

Java for Linux

On Linux, OpenJDK 1.8 64-bit can be downloaded using Yum. OpenJDK Java is located in the standard Red Hat and Oracle Linux repositories.

Follow the steps below to install OpenJDK Java on Linux:

1. Login as the root user.
2. Install Java by running the following command in a Terminal:

   ```
   yum install java-1.8.0-openjdk.x86_64
   ```

Perl

Perl for Windows

When running SPOT the first time ActivePerl will be downloaded and the installer will be launched. For more information on installing ActivePerl, see the **Installing ActivePerl on Windows** section of the documentation.

Perl for Linux

As Perl is standard on Linux, nothing has to be done.

SPOT Foothold Distribution

Running the installation requires an Installation Package provided by Stibo Systems. The Installation Package contains all the required third-party software and scripts used to run the installation. For more information, see the **Pre-Installation Tasks** section of the documentation.

The installation package is automatically downloaded to the servers from the Stibo Systems release server using a foothold distribution.

The foothold zip file contains required scripts and customer specific certificates that allow for the download of third-party software from the Stibo Systems release server. The foothold zip can be received via e-mail from Stibo.

Important: All servers must be able to access the Stibo Update Server via HTTPS (port 443): https://updates.stibosystems.com.

Pre-Installation Tasks

Before beginning the installation, the system must have been set up according to the **System Requirements** section of the documentation. Additionally, the installation instructions described below assume that the volumes / partitions have been configured as listed in the **Default Filesystem Layout** section of the documentation.

Important: The foothold distribution file (received by email) is required to get started. The email also includes a required password to unzip the foothold file. This file should be provided by Stibo Systems.

User Account Controls

If using Windows, the Use Account Control must be disabled during installation.

1. Go to **Control Panel** > **User Accounts** > **Change User Account Control settings** and set notification to 'Never notify'.
2. Restart server to activate settings

Deploying the Installation Package

The foothold distribution file must be placed in a specific directory on the server to get started. This directory is referred to as STEP_HOME.

Important: The steps described in this topic must be executed on all servers included in the setup: Database, Application, and DTP servers.

Create STEP_HOME Directory

On Linux (as a root user)

1. Go to `/opt` directory:

   ```
   cd /opt
   ```

2. Create `stibo/step` directories:

   ```
   mkdir -p stibo/step
   ```

On Windows

1. Go to E: drive
2. Create `stibo\step` directories using Explorer or from command line:

   ```
   mkdir stibo\step
   ```

Unzip Foothold Distribution File

The foothold zip file (`<customer-name>-foothold.zip`) must now be placed in the STEP_HOME folder. This can be done using SCP/SFTP (Linux) and FTP, or copy / paste through RDP (Windows).

On Linux (as root user)

1. Go to the STEP_HOME directory created above:

   ```
   cd /opt/stibo/step
   ```

2. Unzip the foothold file using the supplied password:

   ```
   unzip -P <password> <customer-name>-foothold.zip
   ```

On Windows

1. Go to the STEP_HOME directory created above:

   ```
   E: <enter>
   cd stibo\step <enter>
   ```

2. Unzip the foothold file by right-clicking the file and selecting `Extract All`. Enter the supplied password when prompted.

Now that the foothold distribution file has been deployed on all servers, the user can download and install third-party software and the STEP software.

Prepare SPOT and Download Installation Scripts

Prepare SPOT for the first time and download the installation scripts.

On Linux

Run the following commands:

```
[root@test step]# ./spot --wrapper
Generating wrapper /opt/stibo/step/spot with baked in JAVA_
HOME=/usr/lib/jvm/java-1.8.0-openjdk-1.8.0.191.b12-1.el7_6.x86_64/jre
[root@test step]# ./spot --install
To apply this recipe 1 bundle(s) need to be installed.
The prepared recipe was stored as:
/opt/stibo/step/admin/spot/recipes/prepared/2019-01-11-20-19-55.spr
```

On Windows

Run the following commands:

```
E:\stibo\step>spot --wrapper
Generating wrapper E:\stibo\step\spot.bat with baked in JAVA_
HOME=E:\java\jdk8u192-b12
```

```
E:\stibo\step>spot --install

To apply this recipe 1 bundle(s) need to be installed.

The prepared recipe was stored as:
E:\stibo\step\admin\spot\recipes\prepared\2019-01-11-20-19-55.spr
```

Results

After the last command has run, the following STEP_HOME/admin folder will be created:

```
STEP_HOME/
        |-- admin/
                |-- install/
                |       |-- apache-http-server/
                |       |-- config/
                |       |-- documentation/
                |       |-- oracle-database-11g/
                |       |-- oracle-database-12c/
                |       |-- oracle-instant-client/
                |       |-- oracle-single-server/
                |       |-- os-prerequisites/
                |       |-- pm/
                |       |-- step-audit
                |       |-- step/
                |       |-- step-database/
                |       |-- toolbox-backup/
                |       |-- toolbox-db/
                |       |-- tools/
                |-- install.pl
                |-- spot/
                |-- var/
                        |-- <servername>/
```

These installation scripts comprise what is known as the Installation Package.

Installation

Once the Installation Package has been deployed, the actual installation can begin. The STEP_HOME/admin folder contains the install.pl (Perl) script, which is used to launch the download and install menu.

For more information on the STEP_HOME directory and the Installation Package, see the **Pre-Installation Tasks** section of the documentation.

The installation of STEP is divided into two phases:

- Downloading the Software
- Installing the Software

Downloading the Software

The software must be downloaded before installation.

Note: The third-party software downloaded and used for the STEP installation can be found as HTML documentation in: `STEP_HOME/admin/install/documentation/html/software.html`

1. Log onto the server (database or application server) as `root` (Linux) or `Administrator` (Windows)
2. In a terminal window / command prompt, go to STEP_HOME/admin:

   On Linux

   ```
   cd /opt/stibo/step/admin
   ```

   On Windows

   ```
   E: <enter>
   cd stibo\step\admin <enter>
   ```

3. Run the install.pl script:

   On Linux

   ```
   ./install.pl
   ```

   On Windows

   ```
   install.pl
   ```

When running the script for the first time, the user is prompted to create a folder to hold the downloaded software.

The `Software folder` defaults to `<workarea>/install-software`. This directory can be created if you select 'y', or you can select 'n' and specify another location:

```
[root@test admin]# ./install.pl
2019-01-14 21:19:53 Info     install.pl -> /opt/stibo/step/admin/install.pl
   -> Arguments:
2019-01-14 21:19:53 Command  install.pl ->
/opt/stibo/step/admin/install/config/fetch-software.pl --
log=/opt/stibo/step/diag/logs/test/install.log
2019-01-14 21:19:53 ERROR    fetch-software.pl -> Default software folder not
found: /workarea/install-software
Do you want to create '/workarea/install-software' [y/n]:
```

Note: The workarea directory defaults to `/workarea` (Linux) and `L:` (Windows). A different location must be specified in the database server, usually, `/opt/install-software` (Linux), or `E:\install-software` (Windows) since `/workarea` and `L:\` are only standard partitions for the application server.

The software directory holds all the downloaded software. The default directory can be changed later by pressing `'c'` and entering a new location.

Sharing the Software Directory Between Servers

If possible, the software directory should be shared between all servers (both database and application servers) in the setup. This way the installation script will be able to see the same directory and install configuration.

It is common for the database and application server(s) to not have a shared filesystem. In these scenarios, you will have to use a local filesystem and replicate the configuration file (install.properties). On the database server, you would only download the database-related software and on the application server(s) you would only download the application related software.

Choosing Software to Download

After the Software folder is created, the following menu is displayed:

```
--------------------------------------------------------------------------------
The following software have been marked for download

Software directory: /opt/install-software
--------------------------------------------------------------------------------
    No software marked for download



--------------------------------------------------------------------------------
Choose software for download by selecting a platform

    Id: Platform
--------------------------------------------------------------------------------
     1: Generic
     2: Linux
     3: Mac
     4: Windows
--------------------------------------------------------------------------------
     c: Change software directory
     d: Download files (if not already downloaded) and exit
     q: Exit without downloading
     a: Switch to advanced mode (not recommended)
--------------------------------------------------------------------------------

Choice:
```

Depending on the role of the server you are connected to, and what platform you are installing on, you now must choose what software to download. This is done by choosing the **Id** (1 – 4) in the menu.

Example (Linux)

In the following example the user is downloading software required for a database server installation on the Linux platform (sub-menu Id=2):

```
-------------------------------------------------------------------------------
Please select software to download for 'Linux'
-------------------------------------------------------------------------------

    Id: Software                                                Action
-------------------------------------------------------------------------------
     a: ** Oracle Database 12c **
        oracle-database-12c                                     skip
        oracle-database-12c-patchset-update                     skip

     b: ** Oracle Instant Client 12c **
        oracle-instant-client                                   skip

     c: ** Oracle tools **
        oracp                                                   skip

-------------------------------------------------------------------------------
   a-z: Toggle between 'download' and 'skip'
     0: Back to main menu
-------------------------------------------------------------------------------

Choice:
```

1. Once Linux has been selected, the user would type `'a'` to mark the group 'Oracle Database 12c' for download as this group is required software for the database server.

```
--------------------------------------------------------------------------------
Please select software to download for 'Linux'
--------------------------------------------------------------------------------

    Id: Software                                                Action
--------------------------------------------------------------------------------
     a: ** Oracle Database 12c **
        oracle-database-12c                                     skip
        oracle-database-12c-patchset-update                     skip

     b: ** Oracle Instant Client 12c **
        oracle-instant-client                                   skip

     c: ** Oracle tools **
        oracp                                                   skip

--------------------------------------------------------------------------------
   a-z: Toggle between 'download' and 'skip'
     0: Back to main menu
--------------------------------------------------------------------------------

Choice: a
```

2. Next, the user would enter `'0'` to return to the main menu which has updated to reflect the previous selections.

```
--------------------------------------------------------------------------------
Please select software to download for 'Linux'
--------------------------------------------------------------------------------

    Id: Software                                                Action
--------------------------------------------------------------------------------
     a: ** Oracle Database 12c **
        oracle-database-12c                                     download
        oracle-database-12c-patchset-update                     download

     b: ** Oracle Instant Client 12c **
        oracle-instant-client                                   skip

     c: ** Oracle tools **
        oracp                                                   skip

--------------------------------------------------------------------------------
   a-z: Toggle between 'download' and 'skip'
     0: Back to main menu
--------------------------------------------------------------------------------

Choice: 0
```

3. The user also requires a 'starter' database (referred to as STEP standard) which must be downloaded as well. It is located under the Generic sub-menu (Id=1).

```
--------------------------------------------------------------------------------
The following software have been marked for download

Software directory: /opt/install-software
--------------------------------------------------------------------------------
** linux **
    oracle-database-12c
    oracle-database-12c-patchset-update


--------------------------------------------------------------------------------
Choose software for download by selecting a platform

    Id: Platform
--------------------------------------------------------------------------------
     1: Generic
     2: Linux
     3: Mac
     4: Windows
--------------------------------------------------------------------------------
     c: Change software directory
     d: Download files (if not already downloaded) and exit
     q: Exit without downloading
     a: Switch to advanced mode (not recommended)
--------------------------------------------------------------------------------

Choice: 1
```

4. There are multiple versions of the 'starter' database available from the menu. Each choice is compatible with a range of STEP versions. Choose `'c'` if installing the latest version of STEP.

```
--------------------------------------------------------------------------------
Please select software to download for 'Generic'
--------------------------------------------------------------------------------

    Id: Software                                                Action
--------------------------------------------------------------------------------
     a: ** STEP base (>= step-8.1) **
        stepbase-datapump.step-8.1                              skip

     b: ** STEP base (>= step-8.2) **
        stepbase-datapump.step-8.2                              skip

     c: ** STEP base (>= step-8.3) **
        stepbase-datapump.step-8.3                              skip

--------------------------------------------------------------------------------
   a-z: Toggle between 'download' and 'skip'
     0: Back to main menu
--------------------------------------------------------------------------------

Choice: c
```

5. Type `'0'` to return to the main menu.
6. To start the download, type `'d'`.

```
--------------------------------------------------------------------------------
The following software have been marked for download

Software directory: /opt/install-software
--------------------------------------------------------------------------------
** generic **
    stepbase-datapump.step-8.3
** linux **
    oracle-database-12c
    oracle-database-12c-patchset-update




--------------------------------------------------------------------------------
Choose software for download by selecting a platform

    Id: Platform
--------------------------------------------------------------------------------
     1: Generic
     2: Linux
     3: Mac
     4: Windows
--------------------------------------------------------------------------------
     c: Change software directory
     d: Download files (if not already downloaded) and exit
     q: Exit without downloading
     a: Switch to advanced mode (not recommended)
--------------------------------------------------------------------------------

Choice: d
```

7. The download process begins.

```
2019-01-15 18:54:21 Command  fetch-software.pl -> /opt/stibo/step/spot --
baseurl=updates:/install --download=stepbase-datapump/stepbase-datapump-11.zip -
-output=/opt/install-software
Stibo Patch Operations Tool - test - test - test
Downloading [########################################] 100% 1724/1724 k
/workarea/install-software/stepbase-datapump/stepbase-datapump-11.zip
2019-01-15 18:54:24 Command  fetch-software.pl -> /opt/stibo/step/spot --
baseurl=updates:/install/software --download=oracle-database-
12c/linux/linuxx64_12201_database_1of2.zip,oracle-database-
12c/linux/linuxx64_12201_database_2of2.zip,oracle-database-12c-patchset-
update/linux/p6880880_122010_Linux-x86-64.zip,oracle-database-12c-patchset-
update/linux/p28163133_122010_Linux-x86-64.zip --output=/opt/install-software
Stibo Patch Operations Tool - test - test
Downloading [########################################] 100% 3856/3856 M
```

The files are downloaded under the chosen `Software folder` which, in this example, is `/opt/install-software`. In an application server installation, this defaults to `/workarea/install-software` (Linux), or `L:\install-software` (Windows).

```
[root@test install-software]# pwd
/opt/install-software
[root@test install-software]# ls -l
total 16
drwxrwxrwx. 3 root root 4096 Jan 15 18:57 oracle-database-12c
drwxrwxrwx. 3 root root 4096 Jan 15 18:57 oracle-database-12c-patchset-update
drwxrwxrwx. 3 root root 4096 Jan 15 18:58 oracle-instant-client
drwxrwxrwx. 2 root root 4096 Jan 15 18:54 stepbase-datapump
```

Installing the Software

Once the software has been downloaded to the `Software folder`, the installation will begin.

Note: All installation properties are documented as HTML documentation in: `STEP_HOME/admin/install/documentation/html/install-properties.html`

Initial Installation Steps

1. The initial installation steps require you to provide the names of the database, application, and DTP servers. This must be the FQDN (Fully Qualified Domain Name) of the server(s).

```
--------------------------------------------------------------------------------
Please enter minimum required properties:
--------------------------------------------------------------------------------
host.application-servers =
   Group 'STEP installation setup'
   Specifies the application server(s) in the STEP system.
   Should be fully qualified hostnames (in lower case).
   NOTE: Comma separated.
   - Default value ''
   - Must match pattern
   '^[a-z0-9-]*(\.[a-z0-9-]+)*(\s*,\s*[a-z0-9-]*(\.[a-z0-9-]+)*)*$'
   - This property is mandatory (not null)

Enter value: <Application Server FQDN>

host.database-servers =
   Group 'STEP installation setup'
   Specifies the database server in the STEP system.
   Should be fully qualified hostnames (in lower case).
   NOTE: Comma separated.
   - Default value ''
   - Must match pattern
   '^[a-z0-9-]*(\.[a-z0-9-]+)*(\s*,\s*[a-z0-9-]*(\.[a-z0-9-]+)*)*$'
   - This property is mandatory (not null)

Enter value: <Database Server FQDN>
```

In the event that there is more than one application server, specify them all as a comma-separated list. This is the host name(s) of the server(s). If a DTP server is not part of the setup, just leave the property empty.

2. You will be prompted to provide the **step.system-name**. This is a unique name that identifies the system, and must exist in Stibo's internal systems as the license is mapped to this system name. This name must be agreed upon by the customer and created as a licensed system in Stibo.

   Some examples include:

   - <customer>-development / <customer>-dev
   - <customer>-production / <customer>-prod

```
step.system-name =
   Group 'STEP configuration setup'
   The STEP system name
   - Default value ''
   - This property is mandatory (not null)

Enter value []: test-install
```

3. You will be prompted to provide the **step.takeout-ur**. This is the version of STEP to be installed to the system. Pressing `Enter` will set the value as default, which is the latest version.

```
Enter value [to:step/trailblazer/latest.spr]: [Enter]
```

Installation Menu

After the **step.takeout-ur**is provided and pressing `Enter`, the following installation menu is shown:

```
--------------------------------------------------------------------------------
Date and time              : 2019-01-15 19:16:16
Server                     : testdb.ec2.internal
Server role(s)             : database-server
OS User(s)/OS Group(s)     : oracle:dba
OS Caption                 : Red Hat Enterprise Linux Server release 7.6 (Maipo)
Virtual server             : yes
SPOT is online             : yes
STEP version               : 0.0
Current user               : root
Switch user method         : sudo -u <user> <cmd>
Tablespace (stepsysdata)   : 8192 MB (initial 8192 MB), 1 file(s)
Tablespace (stepsysblob)   : 8192 MB (initial 8192 MB), 1 file(s)

Primary database server    : testdb.ec2.internal
Primary application server : testapp.ec2.internal

Configuration file         : /workarea/install-software/install.properties
Log file                   :
/opt/stibo/step/diag/logs/testdb.ec2.internal/install.log

First/last script          : 1 -> 10
Dry run                    : no
--- -------- ---------------------------------------------------------------
Id  Action   Script description
--- -------- ---------------------------------------------------------------
 1  Run      OS prerequisites
 2  Run      Apply the latest STEP DB Server toolbox
 3  Run      Apply the latest STEP DB Backup toolbox
 4  Run      Install and configure Oracle 12c database software
 5  Run      Install Oracle OPatch and Patch Set Update for Oracle 12c Database
 6  Run      Create database instance with Oracle 12c DBCA
 7  Run      Create STEP database schemas
 8  Run      Create STEP database users
 9  Run      Configure STEP database
10  Run      Import STEP Base database

--- --------------------------------------------------------------------------
Key Command
--- --------------------------------------------------------------------------
** DOWNLOAD **
 d  Download third party software, tools etc. required for the installation
 u  Download and install latest install scripts

** CONFIGURATION **
 e  Edit configuration file
 p  List properties by property name
 v  List properties by property value
 c  List properties that differ from default

** INSTALL **
 f  Change first script
 l  Change last script
 i  Start installation
 q  Quit

** HELP/DEBUG **
 h  Help
 n  Toggle dry run mode
 r  Revision log for install package
 s  Show/hide script
 w  Show/hide reason why steps are marked with action 'Skip'
--- --------------------------------------------------------------------------

Choice:
```

The line `First/last` script indicates which installation scripts are run or skipped. If a script is skipped you can type ‘w’ to see why this script is being skipped. If you only want to run some scripts, you can use ‘f’ and ‘l’ to indicate the first and last script to run.

Default Configuration Properties

The installation consists of a large set of default configuration properties which can be modified by overwriting the defaults in the following file: `<Software folder>/install.properties`.The file can be modified directly from the installation menu by selecting option ‘e.’ Upon saving the file, the installation process will return to the installation menu.

Options ‘p,'v,’ and ’c’ are optional to list the various available properties. The wildcard '*’ can be used with ‘p’ and ‘v’ options to show all properties.

The OS user(s) and group(s) are examples of this, and can be changed to match the customer's requirements.

For some examples, see the **Configuration Examples** section of the documentation.

For information on the filesystem layout, see the **Default Filesystem Layout** section of the documentation.

Toggle Dry Run

Option ‘n’ will toggle dry run mode on or off. The default is off. When dry run mode is enabled, and an installation is started by selecting option ‘i,’ the installation goes through its install routine based on the supplied properties, and builds a file that self-documents the scripts and commands that will be executed during the installation process without installing any components.

The file created during a dry run is `<STEP_HOME>/logs/<hostname>/dry-run/dry-run.txt`.

Any files that are generated as part of the installation routine will be placed in: `<STEP_HOME>/diag/logs/<hostname>/dry-run/<username>`.

Note: Ensure to toggle dry run to 'off' to start an actual installation that physically installs the components.

Completing the Installation

When ready, typing ‘i’ will start the installation. If errors are found during installation, they must be solved before the installation can continue. Upon a successful installation, STEP will be started automatically on the application server.

An installation log file is found in `<STEP_HOME>/logs/<hostname>/install.log`.

Note: When installing the database server in Windows, the system administrator will have to run all options (excluding 'OS prerequisites') as the oracle user or equivalent, otherwise the actions will be marked as ‘Skip’.

Application Server Prompt Examples

```
--------------------------------------------------------------------------------
The following software have been marked for download

Software directory: /workarea/install-software
--------------------------------------------------------------------------------
** linux **
    oracle-instant-client




--------------------------------------------------------------------------------
Choose software for download by selecting a platform

    Id: Platform
--------------------------------------------------------------------------------
     1: Generic
     2: Linux
     3: Mac
     4: Windows
--------------------------------------------------------------------------------
     c: Change software directory
     d: Download files (if not already downloaded) and exit
     q: Exit without downloading
     a: Switch to advanced mode (not recommended)
--------------------------------------------------------------------------------

Choice: d
```

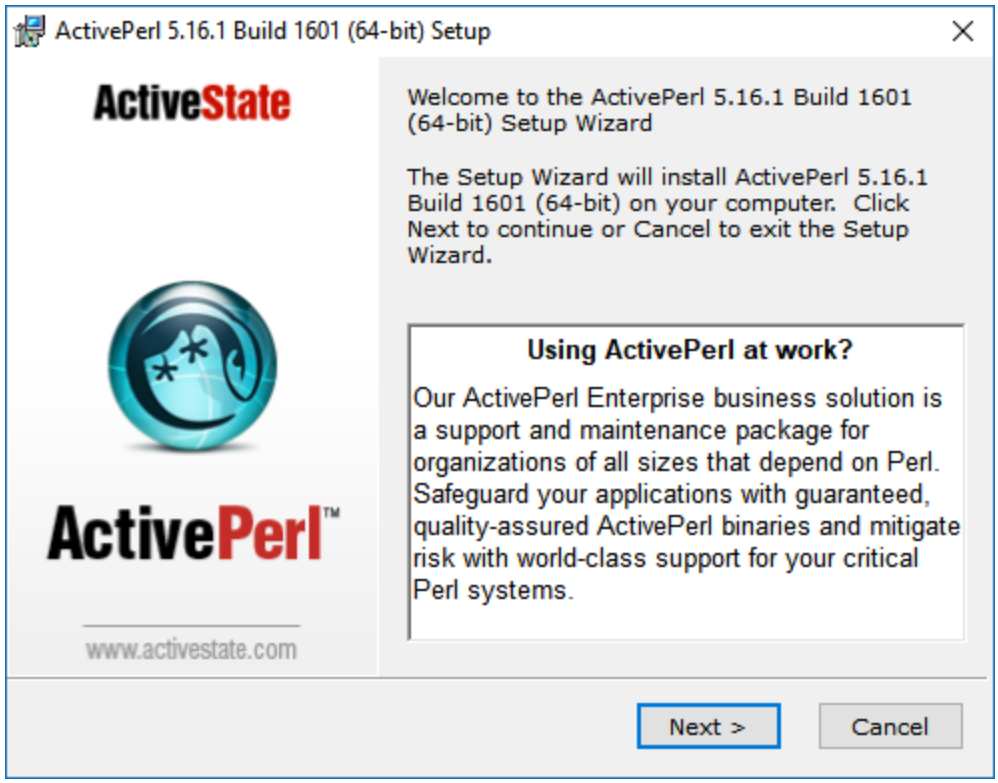
```
--------------------------------------------------------------------------------
Date and time              : 2019-01-22 15:02:28
Server                     : ip-172-31-36-178.ec2.internal
Server role(s)             : application-server
OS User(s)/OS Group(s)     : stibosw:stibosw
OS Caption                 : Red Hat Enterprise Linux Server release 7.6 (Maipo)
Virtual server             : yes
SPOT is online             : yes
STEP version               : 0.0
Current user               : root
Switch user method         : sudo -u <user> <cmd>

Primary database server    : testdb.ec2.internal
Primary application server : testapp.ec2.internal

Configuration file         : /workarea/install-software/install.properties
```

```
Log file                    : /opt/stibo/step/diag/logs/ip-172-31-36-
178.ec2.internal/install.log

First/last script           : 1 -> 4
Dry run                     : no
-------------------------------------------------------------------------------

--- -------- ---------------------------------------------------------------
Id  Action   Script description
--- -------- ---------------------------------------------------------------
 1  Run      OS prerequisites
 2  Run      Install and configure the Oracle instant client software
 3  Run      Install and configure the Apache HTTP server software
 4  Run      Install and configure the STEP software

--- -------------------------------------------------------------------------
Key Command
--- -------------------------------------------------------------------------
** DOWNLOAD **
 d  Download third party software, tools etc. required for the installation
 u  Download and install latest install scripts

** CONFIGURATION **
 e  Edit configuration file
 p  List properties by property name
 v  List properties by property value
 c  List properties that differ from default

** INSTALL **
 f  Change first script
 l  Change last script
 i  Start installation
 q  Quit

** HELP/DEBUG **
 h  Help
 n  Toggle dry run mode
 r  Revision log for install package
 s  Show/hide script
 w  Show/hide reason why steps are marked with action 'Skip'
--- -------------------------------------------------------------------------
```

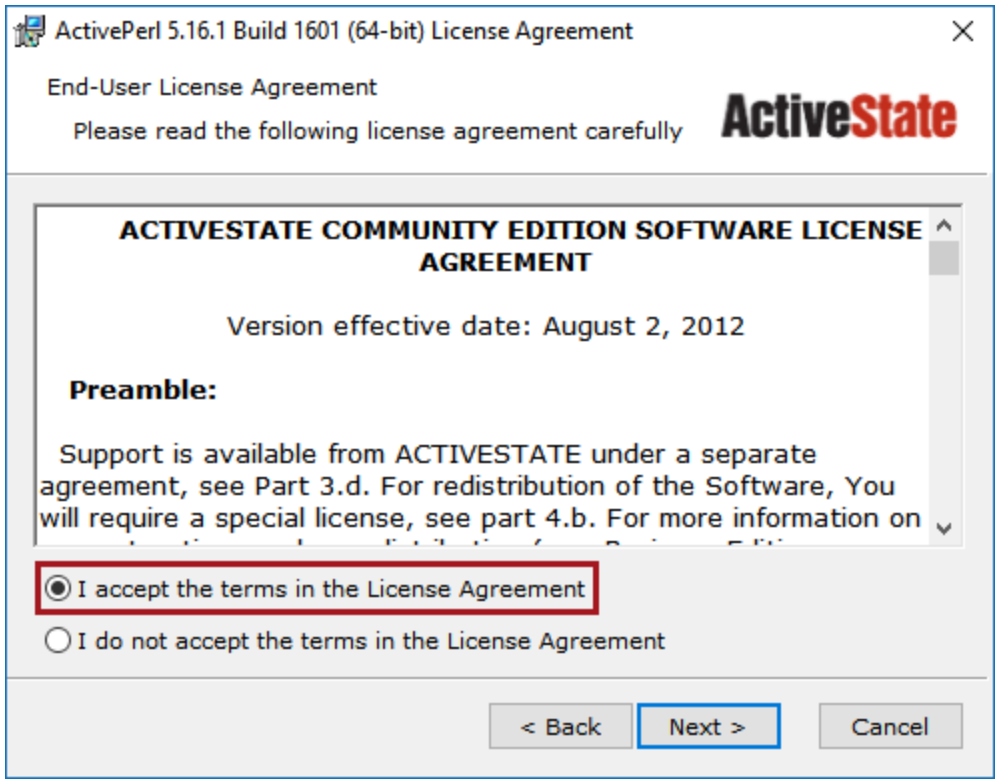
Installing ActivePerl on Windows

The ActivePerl installer is automatically launched when running `STEP_HOME\spot-install` for the first time.

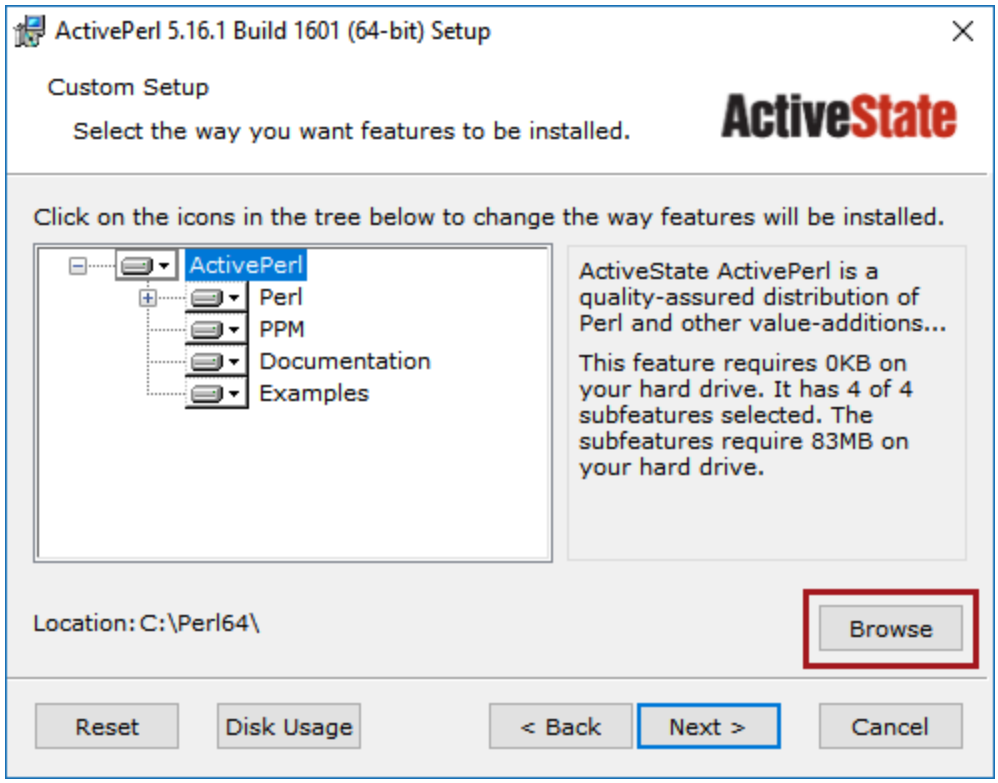
1. Click **Next** to proceed with the installation.



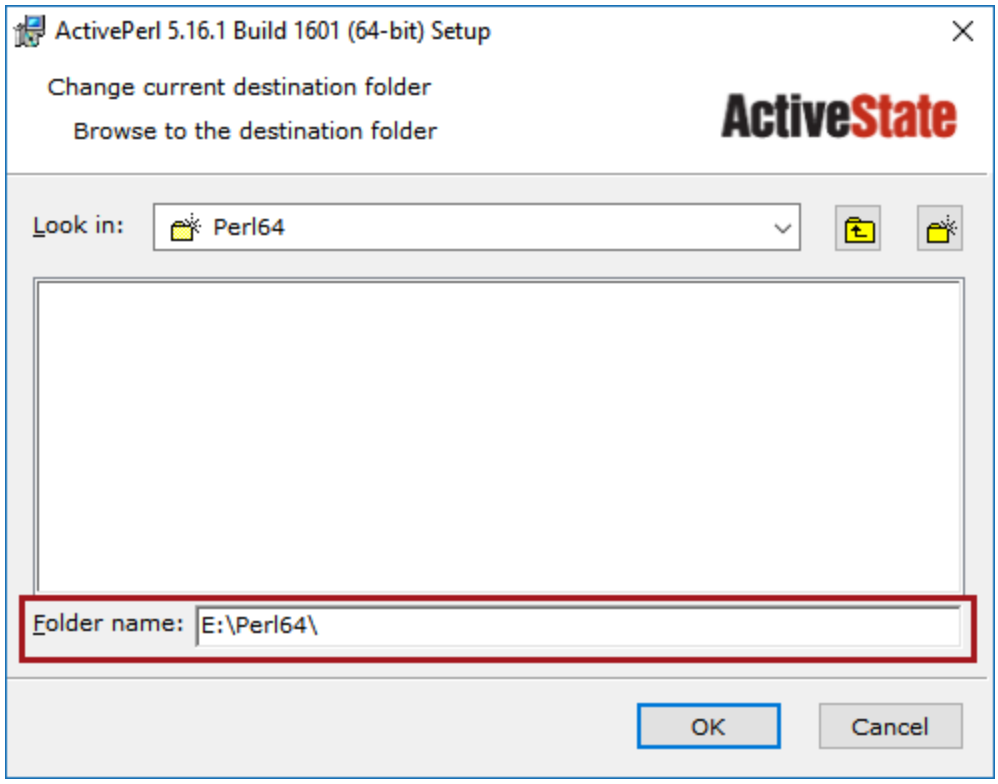
2. Accept the End-User License Agreement and click **Next**.



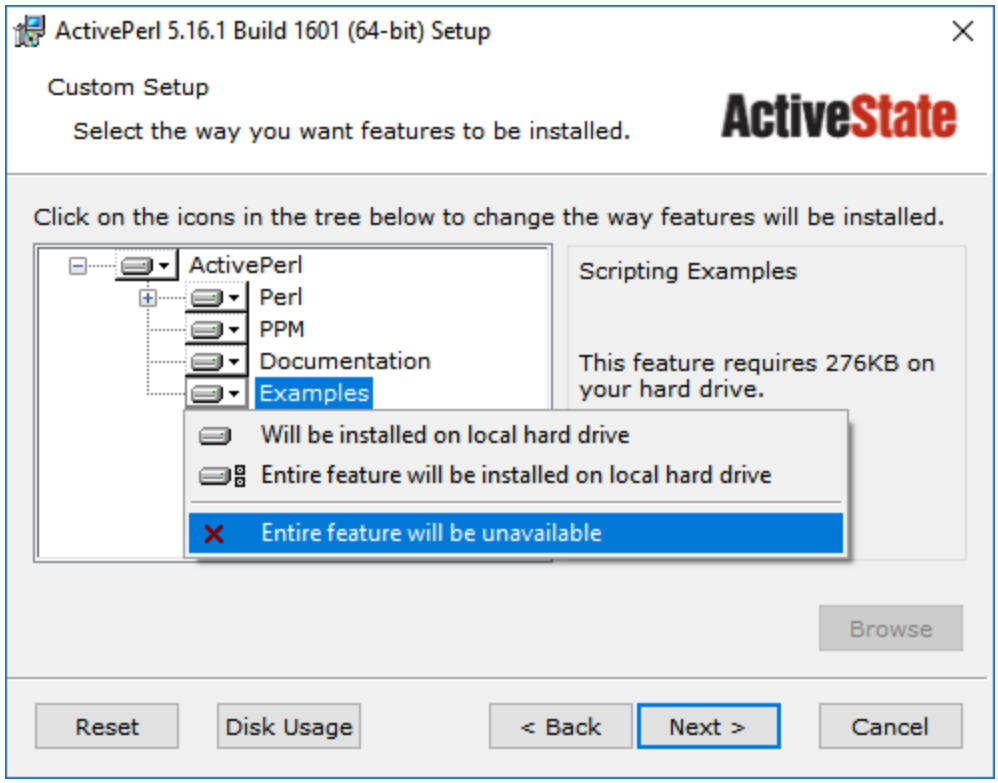
3. Click **Browse** to open the 'Change current destination' folder dialog.



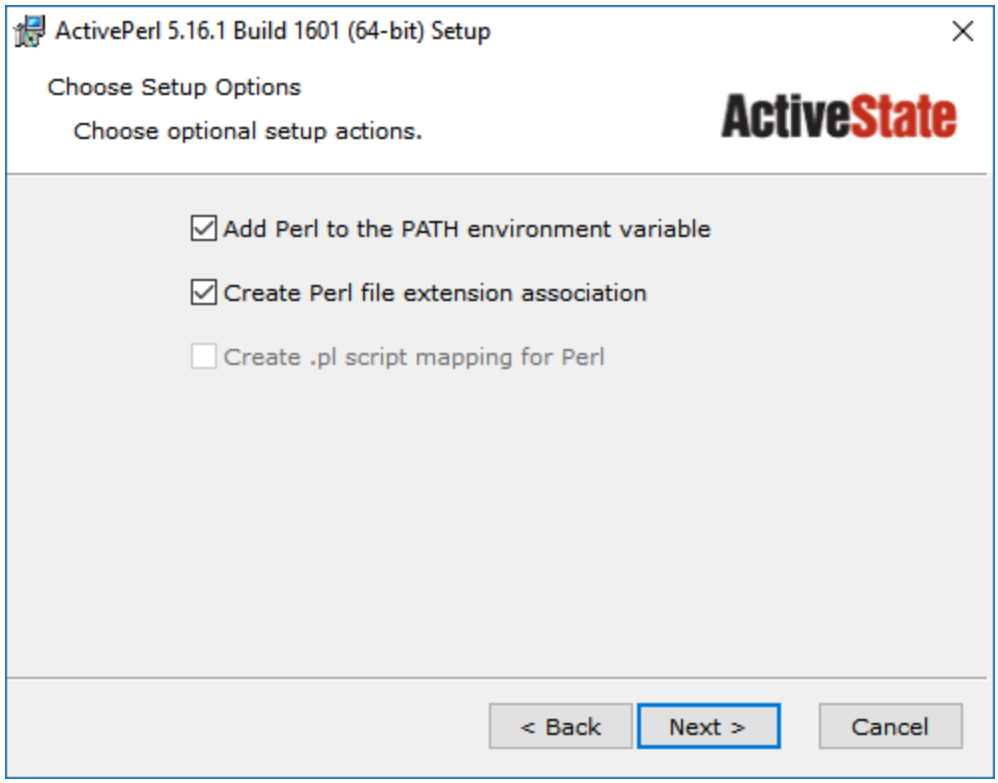
4. Enter the correct path in the 'Folder name' parameter and click **OK**. The location in the image is for example purposes.



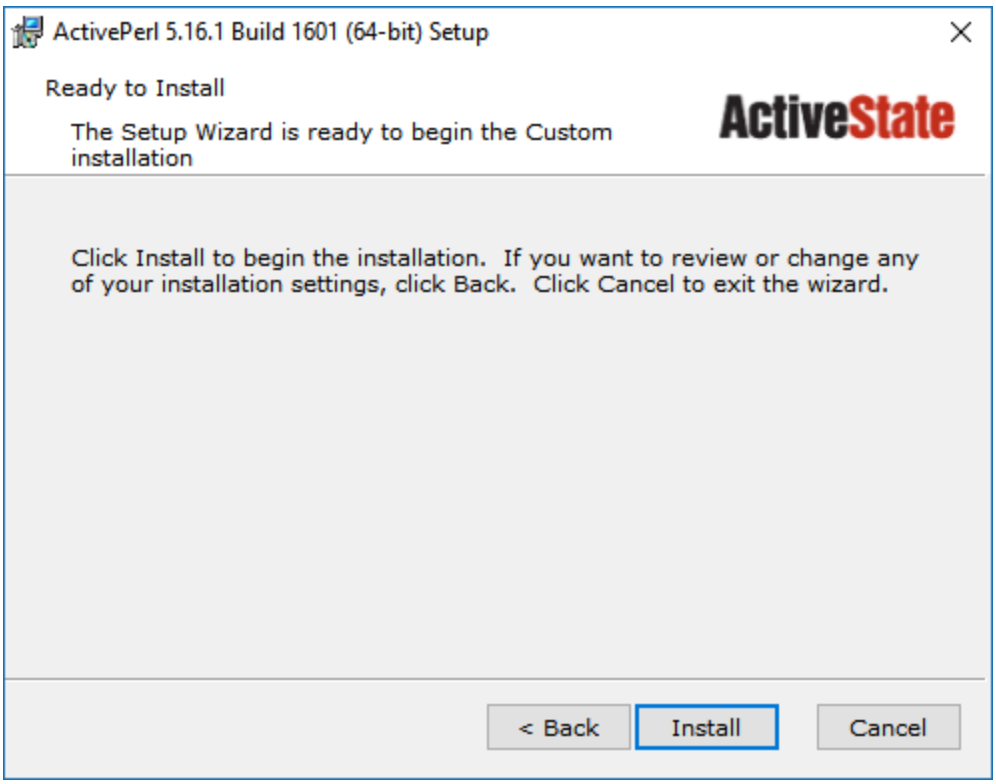
5. Right-click on 'Examples' in the navigation tree, and select 'Entire feature will be unavailable'. Click **Next**.



6. On the 'Choose Setup Options' screen, leave the defaults selected, and click **Next**.



7. On the 'Ready to Install' screen, click **Install**.



8. Once the installation is complete, click **Finish** to exit the installer. You may also check the 'Display the release notes' box if you wish to view the release notes upon closing the installer.

Default Filesystem Layout

The default filesystem layouts expected by the installation scripts are detailed in the tables below.

This information can also be found in the Hardware Recommendations document (provided by Stibo Systems).

Linux

Database Server

Server Internal Storage:

Partition	Description	Size (Net)
/	System	10 GB (or more)
/opt	Software	40 GB
/tmp	Temp file system	10 GB
/var	Variable files	10 GB
/home	Home	5 GB

Server External Storage:

Partition	Description	Size (Net)
/database/dbredo1	Redo1	10 GB
/database/dbredo2	Redo2	10 GB
/database/db1	Database	XX GB
/database/dblog	DB-Log	XX GB
/database/backup	Online Backup	XX GB

Note: The online backup partition (staging area for RMAN) is optional depending on any existing backup solutions on an existing storage setup.

Application Server

Server Internal Storage:

Partition	Description	Size (Net)
/	System	10 GB (or more)
/opt	Software + Application temp	100 GB
/tmp	Temp file system	10 GB
/var	Variable files	10 GB
/home	Home	5 GB

Server External Shared Storage:

Partition	Description	Size (Net)
/workarea	Workarea	100 GB
/upload	Upload	50 GB

Note: In a clustered application server setup, this storage must be shared with read / write access by all application servers.

Windows

Database Server

Server Internal Storage:

Partition	Description	Size (Net)
C:\	System	40 GB
E:\	Software	40 GB

Server External Storage (SAN):

Partition	Description	Size (Net)
F:\	Redo1	10 GB
G:\	Redo2	10 GB
H:\	Database	XX GB
I:\	DB-Log	XX GB
K:\	Online Backup	XX GB

Note: The online backup partition (staging area for RMAN) is optional depending on any existing backup solutions on an existing storage setup.

Application Server

Server Internal Storage:

Partition	Description	Size (Net)
C:\	System	40 GB
E:\	Software + application temp	100 GB

Server External Storage (SAN):

Partition	Description	Size (Net)
L:\	Workarea	100 GB
M:\	Upload	50 GB

Note: In a clustered application server setup, this storage must be shared with read / write access by all application servers.

Cluster SSH Configuration

When configuring a Linux application server cluster, the OS account that runs the STEP software (default 'stibosw') must be able to login via SSH between the application servers without being prompted for a password. This secure access is accomplished by using public key authentication.

SSH Setup

1. Select one machine to be the temporary master. This machine is where the commands will be run.
2. On the master machine, enter the following command with the stibosw account:

```
ssh-keygen -t dsa
```

Note: Accept all defaults.

3. On each other machine in the cluster except the master, enter the following commands with the stibosw account:

```
ssh-keygen -t dsa
ssh master cat ~/.ssh/id_dsa.pub >> ~/.ssh/authorized_keys
chmod 600 ~/.ssh/authorized_keys
```

Note: Accept all defaults.

4. The stibosw account on the master machine can now ssh to all other machines using public key authentication. Repeat the above steps so any application server in the cluster can act as the 'master.'
5. Once the ssh keyexchange is done, it is possible for the stibosw user on the cluster to ssh to any other machine using public key authentication.

Configuration Examples

Application Server

The following configuration steps are typically required:

- Specify the fully qualified hostnames of the STEP servers (mandatory)
- Specify the fully qualified hostname of the database server (mandatory)
- Specify the System Name (mandatory)
- Specify a STEP takeout URL
- Specify a filesystem for software if it differs from the default
- Specify a filesystem for Workarea and Upload if they differ from default
- Specify a filesystem for STEP temp-files if it differs from default
- Specify Java heapsize for STEP if it differs from default
- Specify database name and/or service if they differ from default step
- Specify name and/or password for STEP database users if they differ from default

Examples

```
host.application-servers = stepapp.stibo.com
host.database-servers = stepdb.stibo.com
step.system-name = step-test
#step.takeout-url =
#filesystem.app-software =
#filesystem.app-workarea =
#filesystem.app-upload =
#filesystem.step-temp-files =
#step.applicationserver-maximumheapsize =
#step.applicationserver-initialheapsize =
```

If the name of the STEP OS user differs from the default 'stibosw':

```
user.app = <my-os-app-user>
```

If the service name of the database differs from dbca.db-name:

```
dbca.service-name = <my-db-service-name>
```

If the name and/or password of the stepsys / stepview database user differs from default:

```
database.stepsys-owner = <my-db-stepsys>
```

```
database.stepsys-owner-passwd  = <my-db-stepsys-password>
database.stepview-owner = <my-db-stepview>
database.stepview-owner-passwd = <my-db-stepview-password>
```

Note: Leading blanks are not valid.

Database Server

The following configuration steps are typically required:

- Specify the fully qualified hostnames of the STEP servers (mandatory).
- Specify the fully qualified hostname of the database server (mandatory).
- Specify the System Name (mandatory).
- Specify a filesystem for software if it differs from the default.
- Specify a filesystem for DB-Workarea and DB-Upload if they differ from default, or set to blank if they are not present on the database server (ignored for single server installations).
- Specify filesystems for the database if they differ from default.
- Configure Oracle edition if it differs from default. See the Configure Oracle Server Edition section below for more information.
- Manually specify the amount of memory that is available to the database and/or use memory target. See the Configure Oracle Server Edition section below for more information.
- Specify the size of the tablespaces STEPSYSDATA and STEPSYSBLOB.

Examples

```
host.application-servers = stepapp.stibo.com
host.database-servers = stepdb.stibo.com
step.system-name = step-test
#filesystem.db-software =
filesystem.db-workarea =
filesystem.db-upload =
#filesystem.database =
#filesystem.dblog =
#filesystem.redo1 =
#filesystem.redo2 =
database.stepsysdata-size-gb = 32
database.stepsysblob-size-gb = 320
database.stepsysblob-initial-file-size-gb = 16
```

If the name of the database OS user differs from the default Oracle:

```
user.db = <my-os-db-user>
```

If the name of the database differs from the default step:

```
dbca.db-name = <my-db-name>
```

If the service name of the database differs from dbca.db-name:

```
dbca.service-name = <my-db-service-name>
```

If the instance name of the database differs from dbca.db-name:

```
oracle.sid = <my-instance-name>
```

If the name and/or password of the stepsys / stepview database user differs from default:

```
database.stepsys-owner = <my-db-stepsys>
database.stepsys-owner-passwd = <my-db-stepsys-password>
database.stepview-owner = <my-db-stepview>
database.stepview-owner-passwd = <my-db-stepview-password>
```

If the database is a RAC database, then specify the SCAN name:

```
oracle.database-listener-reference = <scan-name>
```

Note: Leading blanks are not valid.

Configure Oracle Instance Memory

By default, SGA_TARGET and PGA_AGGREGATE_TARGET are used. SGA and PGA are set to 5/6 and 1/6 of the accountable amount of memory for the database instance.

If you want to use MEMORY_TARGET, then you must specify:

```
dbca.use-memory-target = true
host.db-use-hugepages = false  (if Linux)
```

The available amount of memory for the database instance is calculated from either of the following rules (available-host-memory):

- For a dedicated database server, the available amount of memory is set to 2/3 of host.memory-ram.
- For a server which is both an application and database server, the available amount of memory is set to 2/5 of host.memory-ram.

The maximum allowable amount of memory for the database instance (dbca.max-db-memory) is set to 68719476736 (64GB) by default.

The accountable amount of memory for the database instance is calculated from the following rules:

- dbca.max-db-memory < available-host-memory
  - The calculation is based on dbca.max-db-memory
- dbca.max-db-memory >= available-host-memory
  - The calculation is based on available-host-memory

- dbca.max-db-memory is set to 0
  - The calculation is based on available-host-memory

If the amount of available-host-memory is higher than dbca.max-db-memory and you want to use it, raise dbca.max-db-memory or set it to 0 to disable the limit. Alternatively, you can lower the value if you want your database instance to use less than the calculated available-host-memory.

Configure Oracle Server Edition

By default, the Oracle server software is installed as the standard edition (SE). To install the enterprise edition, you must specify:

```
oracle.install-edition-short-name = EE
```

The Oracle enterprise edition has a list of optional components. These components come at an additional cost. All components are installed, but you have the choice to enable / disable specific or all components. By default, the install script is configured to install Oracle enterprise with all optional components disabled.

To disable all optional components:

```
oracle.enterprise-edition-options-selection = true
oracle.enable-enterprise-edition-options-list
```

Note: The options list must be blank.

To enable specific optional components:

```
oracle.enterprise-edition-options-selection = true
oracle.enable-enterprise-edition-options-list = <comp1-name>:version, <comp2-name>:version
```

The components can also be enabled or disabled after installation by using the command utility chopt.

The optional components are registered in the v$option view.

Upgrading to Oracle Enterprise Edition

To upgrade to the Enterprise Edition, you must be able to install the exact same Oracle server software (including patches). If the required software is unavailable or you want to upgrade to a new Oracle release, you must upgrade both the server software and the database. See the Oracle upgrade guide for instructions.

Note: The Oracle upgrade guide is provided by Oracle's own documentation.

Upgrading in Linux

1. Stop all Oracle services related to the Oracle home you intend to uninstall. Note that the listener may be servicing other Oracle homes.
2. Backup the Oracle home you intend to uninstall.
3. Copy the db-parameter files, password files, listener.ora, tnsnames.ora, and sqlnet.ora from the Oracle home you intend to uninstall, and send them to a location outside Oracle base.
4. Comment out <ORACLE_SID> entries, in /etc/oratab, related to the Oracle home you intend to uninstall.
5. Uninstall the current Oracle server software installation from a location outside the Oracle home you intend to uninstall:

   ```
   $ <ORACLE_HOME-TO-UNINSTALL>/uninstall/uninstall
   ```

Important: When the program says 'de-configure', the term means to remove. If the list of Oracle databases shown is not empty, then there is something wrong. You should not continue unless you truly want to remove the database.

6. Install the Oracle server software again (including patches).
7. Stop the listener if it is running.
8. Restore db-parameter files, password files, listener.ora, tnsnames.ora, and sqlnet.ora.
9. Start the listener.
10. Start the database.

The database is now upgraded to Enterprise Edition.

Upgrading in Windows

1. Stop all Oracle services related to the Oracle home you intend to uninstall. Note that the listener may be servicing other Oracle homes.
2. Backup the Oracle home you intend to uninstall.
3. Copy db-parameter files, password files, listener.ora, tnsnames.ora, and sqlnet.ora from the Oracle home you intend to uninstall, and send them to a location outside Oracle base.

4. Backup registry settings 'ORA_<ORACLE_SID>_%'.
5. Remove the listener if it has the same name as a listener in another Oracle home. To do this, manually execute Net Configuration Assistant (NETCA) in the Oracle home you intend to uninstall.
6. Delete Windows service 'OracleService<ORACLE_SID>', using ORADIM, belonging to the Oracle home you intend to uninstall.
7. Leave all command prompts and Explorer files that are located in the Oracle home you intend to uninstall.
8. Uninstall the current Oracle server software installation from a location outside the Oracle home you intend to uninstall:

   ```
   $ <ORACLE_HOME-TO-DEINSTALL>\deinstall\deinstall.bat
   ```

Important: When the program says 'de-configure', the term means to remove. If the list of Oracle databases shown is not empty, then there is something wrong. You should not continue unless you truly want to remove the database.

9. Install the Oracle server software again (including patches).
10. Recreate the listener by executing Net Configuration Assistant (NETCA) in the re-installed ORACLE_HOME.
11. Stop the listener if it is running.
12. Restore db-parameter files, password files, listener.ora, tnsnames.ora, and sqlnet.ora.
13. Start the listener.
14. Recreate Windows service 'OracleService<ORACLE_SID>' using ORADIM. Restore registry settings for the service as needed.
15. Start the database.

The database is now upgraded to Enterprise Edition.

Offline Installations

Customers with strict security policies that prohibit external access to https://updates.stibosystems.com will prevent the standard installation from working, resulting in a failed installation. In these cases, it may be necessary to perform an 'offline' installation of STEP for both Application and Database servers.

An offline installation comprises two steps. The first step is to prepare the relevant installation recipes (install the latest STEP version) as well as creating a software repository of the software required during the installation. This software must be fetched from Stibo's updates.stibosystem.com server. This step must be performed in a location and on a machine that has access to https://updates.stibosystem.com. The content created during this stage will need to be copied to the target server intended for the installation. The second step is to perform the installation on the target server using the files / content created from the first step using the online machine.

Note: Java JRE 1.8 is required for the tasks listed in the below sections.

Online Machine Tasks

1. Unzip the specific customer foothold.zip file into '<STEPHOME>'
2. `<STEPHOME>/spot --wrapper`
3. `<STEPHOME>/spot --install`
4. `<STEPHOME>/spot --offline-recipe --snapshot=install.spr`
5. `<STEPHOME>/spot --accessmode=offline --`

   `prepare=to:step/trailblazer/latest.spr --`

   `target=install.spr --output=latest-fullfat.spr`

6. `<STEPHOME>/admin/install/config/fetch-software.pl`

   Download the required 3rd party software (e.g. /workarea/install-software).

7. The following files will need to be transferred to the target server and will be used during the installation of STEP:
   - install.spr
   - latest-fullfat.spr
   - zipped / tarball of directory containing the downloaded software (e.g., workarea/install-software)

Target Server Tasks

1. Unzip the specific customer foothold.zip file into '<STEPHOME>'
2. `<STEPHOME>/spot --wrapper`

3. `<STEPHOME>/spot --apply=install.spr --sync`
4. `<STEPHOME>/admin/install.pl --swdir=<SWDIR>`

Note: When prompted for the 'step.takout-url' property, enter `latest-fullfat.spr` including the full path to the file.

---

5. The installation will continue to process as normal.

Note: During the installation of the STEP software, an error will be thrown testing a connection to the Stibo Updates server. Press ‘c’ to ignore and continue with the installation.

Patching STEP

The system architecture of the STEP platform is split up into separate components, each of which may access other components through a set of component APIs. This component-based architecture satisfies the otherwise contradictory requirements for longer time between releases and fast introduction of new improvements. Customers can choose to upgrade specific components in order to take advantage of new features and important updates, whilst keeping the core of STEP and other unrelated components unchanged. This approach reduces the risk and workload involved in testing new updates.

When upgrading components individually, the customer may choose not to upgrade to the newest STEP release. If the feature is available in a new component, that component can be installed on its own. If the feature is available as an upgrade to an existing component, that component may be upgraded while keeping other components as they are.

Note: Available component updates are made visible on a STEP system similarly to the way updates are to mobile phone apps, i.e., with release notes detailing the new features and fixes available relative to the current installation, and with instructions on how to perform the update.

Components have separate release cycles limited only by the dependencies introduced when one component uses another component. Each component declares its dependency on other components through principles where a given component version may depend on a specified range of versions of another component.

For additional information on patching STEP components, see the following topics within this documentation:

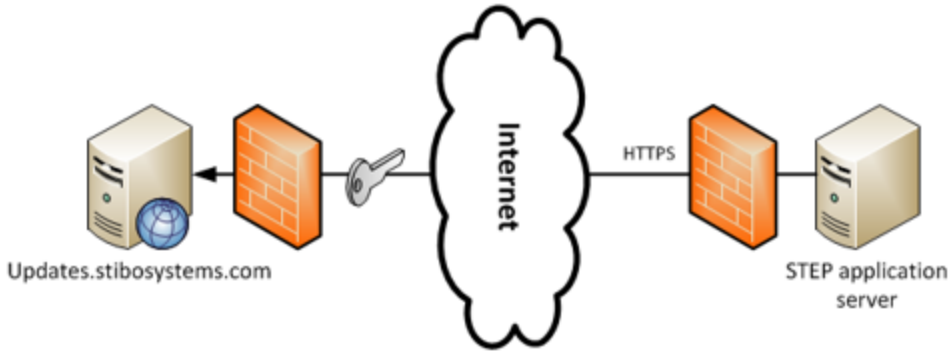
- Patching Methods
- SPOT Program
- STEP Patching Procedures
- Patching Security
- Configuring a Private Updates Mirror

Patching Methods

Patch operations in the STEP system are defined by the specific component(s) being installed / upgraded. These component updates are downloaded either directly from one of the Stibo Systems Global Updates Mirrors (Release Server) or from a private updates mirror at the customer can be used to execute these operations. The connection to either of the two uses an encrypted network connection over HTTPS. Connections are always initiated from the customer side. The update mirror will at no time initiate a connection to the STEP environment.

Direct Connection to Release Server

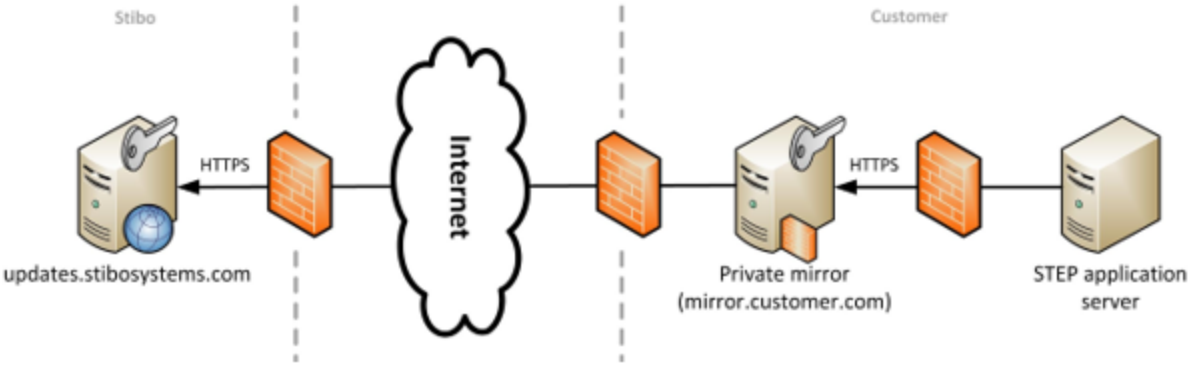
Downloading updates directly from a Release Server is the default method for patching. Using this method, the STEP environment is configured to allow an encrypted connection by HTTPS to the release server. This method offers the best security.



Advantages to using the Direct Connection method include: faster support from Stibo Systems by providing complete version information and a simplified infrastructure.

Private Updates Mirror

As an alternative to accessing the release server directly, it is possible to set up a Private Updates Mirror and configure SPOT on the internal STEP servers to use the mirror instead.



Advantages

The advantages of using the Private Updates Mirror method are:

- If the internet connection or the global updates server breaks down, the already downloaded files will still be available.
- The internet connection bandwidth consumed is reduced by avoiding repeated downloads.
- The network configuration is simpler as only the mirror needs to access the updates server, while the individual SPOT instances can be configured to talk only to the private mirror on the internal network.

Requirements

To run a private mirror server, you need:

- A 64-bit Linux host, not shared with STEP.
- java 8 64-bit (and updated version will be installed by SPOT, so the OS version is okay for bootstrapping).
- Enough storage to hold the entire mirror (400 GB will suffice).
- Outgoing internet access to the Stibo Systems updates servers on port 443.
- Incoming aces from the private network on port 443 for the SPOT hosts.
- A DNS entry on the local network that can be expected to never change, so *mirror.customer.com* would be preferable to *pc2016-02-13-room7-linux-test-dl120g9.dhcp.customer.com*.

Upstream Root Mirrors

The root mirrors that the private mirror connects to can be listed using `spot --mirrors`, but these are the current hosts:

- *dk1.updates.stibosystems.com*: Primary root mirror.
- *dk2.updates.stibosystems.com*: Secondary root mirror.
- *updates.stibosystems.com*: Fail-over mirror on a shared IP between the two root mirrors.

Outgoing TCP access on port 443 must be allowed to each of the root mirror IP addresses from the private mirror, this way the mirror has more upstream mirrors to pick from if one fails.

SPOT Program

The Stibo Patch Operations Tool (SPOT) program initiates an encrypted exchange between the customer site and the update mirrors at Stibo Systems. This program runs off either the STEP application server or on a dedicated SPOT support installation PC.

The communication sequence between the SPOT program and the update mirror is as follows:

1. SPOT stores the current thin snapshot of version information to updates.stibosystems.com.
2. SPOT fetches the desired recipe of the software bundles to download.
3. SPOT downloads the actual bundles.
4. SPOT stores the updated thin snapshot of the version information to updates.stibosystems.com.

Storing the thin snapshots to updates.stibosystems.com serves two purposes. It enables:

- Stibo Systems to support the STEP environment by providing complete version information.
- Easy creation of the exact software configuration for additional Test / QA environments and in the case of disaster recovery.

Both the metadata (including the thin snapshot) and the bundle recipe, together with the actual bundles, are cached by SPOT and only the files that are actually needed are ever downloaded, so the amount of data transferred is as low as possible.

The SPOT program can be found in the home directory of the STEP installation on the application server. On a Linux server, this will typically be in `/opt/stibo/step`. On a Windows server, this will typically be in `E:\stibo\step`.

Using the Upgrade Command

To help simplify the analysis process and make it easier to work with components, the `--upgrade` command can be executed to look for possible upgrades to the components installed on a STEP system. Users can also run the command to search for a component not already installed to verify availability and compatibility with their version of STEP.

The `--upgrade` (or `-u`) command is used to calculate the newest possible version of the listed components that can be installed given different restrictions on how large of an upgrade is allowed. This command never changes the STEP system or performs any automatic upgrading.

To further facilitate the process, the upgrade options output that is displayed upon running the `--upgrade` command includes a recipe file that can later be applied to the system.

Outlined below are the different upgrade command options with examples. The name of the actual component(s) should be used in place of what is shown in the examples.

Upgrade Options	Examples
Upgrading one component	To upgrade the Experian component, use: `--upgrade=experian`
Upgrading several components	To upgrade both Experian and Loqate, use: `--upgrade=experian,local-loqate`
Upgrading the baseline (the STEP version such as 9.0, 9.1)	The baseline can also be upgraded by using the component name *step*: `--upgrade=step` Pick the release of the baseline by specifying a prefix: `--upgrade=step:9.1` When a prefix is specified, the newest version matching the prefix will be tried.

Installation candidates (components that have not yet been installed) can also be found by using the `--upgrade` command, as described above.

Upgrade levels

The calculation used to determine upgrades can potentially produce suggestions for up to five levels of upgrade. Only the upgrades that bring newer versions of the listed components will be shown in the result. Below, the options shown are sorted by how aggressive the update would be with regard to introducing new component versions.

Level	Description
Listed	This is the most conservative upgrade possible where only the listed components are touched.
Dependents	This level allows upgrading of: • The listed components • The components that depend on the listed components
Dependencies	This level, listed with `DEPENDENCIES_BUT_NOT_BASELINE` in the file name, allows upgrading of: • The listed components • The components that depend on the listed components

Level	Description
	• The components that the listed components depend on, but not STEP baseline
Baseline within Maintenance Patch	This level, listed with `BASELINE_WITHIN_MP` in the file name, allows upgrading of all components, including the STEP baseline, but only to the latest maintenance patch of the same release as the one currently installed. For example, if the system has step-8.2-mp1 installed, then this level would look for the newest MP of that release, possibly 8.2-mp3, but not 8.3.
Baseline	This is the least conservative upgrade level, which allows upgrading of all components, including the STEP baseline to the latest released version.

The upgrade options are shown on screen with the upgrade file recipes listed. Users can use standard commands to view a detailed change log, prepare for an installation, and to apply changes to their STEP system.

For example: `--upgrade=inmemory`

```
Found 3 possible upgrades to choose from:
================================================================================
Option 1: Upgrade only the listed components
Components:
       * assetloader: Keep at 7.0.14 (newest available: 7.0.24)
       * inmemory: Upgrade from 7.0.10 (newest available: 7.0.23)
       * spot: Keep at 7.0.48 (newest available 7.0.65)
File: /home/step/admin/spot/recipes/upgrade/upgrade.LISTED.2017-11-01-15-24-
28.spr
================================================================================

Option 2: Upgrade to latest maintenance patch within the same STEP release
+ All components
Components:
       * assetloader: Keep at 7.0.14 (newest available: 7.0.24)
       * inmemory: Upgrade from 7.0.10 to 7.0.15 (newest available: 7.0.23)
       * spot: Keep at 7.0.48 (newest available 7.0.65)
       * step: Upgrade from 8.0-mp3-2016-09-06-14-12-00 to 8.0-mp4-2016-10-04-10-10-
27 (newest available: 8.2-mp3-2017-11-02-07-39-51)


File: /home/step/admin/spot/recipes/upgrade/upgrade.BASELINE_WITHIN_MP.2017-11-
01-15-24-28.spr

================================================================================

Option 3: Upgrade to latest STEP release (full upgrade)
+ All components
```

```
Components:
       * assetloader: Keep at 7.0.14 (newest available: 7.0.24)
       * inmemory: Upgrade from 7.0.10 to 7.0.14 (newest available: 7.0.23)
       * spot: Keep at 7.0.48 (newest available 7.0.65)
       * step: Upgrade from 8.0-mp3-2016-09-06-14-12-00 to 8.1-mp5-2017-10-02-16-10-
00 (newest available: 8.2-mp3-2017-11-02-07-39-51)


File: /home/step/admin/spot/recipes/upgrade/upgrade.BASELINE.2017-11-01-15-24-
28.spr
```

If the system is ignoring any components or if the system cannot find a way to upgrade the components specified, the applicable messaging will be shown on the screen. All ignored versions will not be considered when trying to find an upgrade.

Starting with STEP 8.3 and with all subsequent versions, the `--upgrade` command can be used in place of the installation commands. For example, `--upgrade=wikimetadata` or `--upgrade=acrolinx`.

STEP Patching Procedures

All commands listed are valid for any STEP environment, counting single application server setups and clusters.

Back Up the STEP Database and Application

Before patching STEP, a fallback procedure should be developed to mitigate any risk. Ideally, full back ups of the STEP database should be maintained, and the option to restore the database to a specific point in time should be available.

Note: The requirement for each individual patch may vary. Refer to the relevant release note for more information.

Back ups of the STEP application should also be maintained, including all files provided in STEP_HOME/config.properties and a snapshot of the STEP system itself.

To take a snapshot of the STEP system:

```
cd /opt/stibo/step
./spot --snapshot=/workarea/<snapshot-env-date>.spr
```

Prepare the Patch

The patch should be downloaded in advance to avoid unnecessary downtime for deployment.

A STEP core patch may look like the following command:

```
./spot --prepare=to:step/trailblazer/step-trailblazer-<release>.spr
```

Sometimes, customers have their own components in addition to the STEP core, and the command could look something like this:

```
./spot --prepare=to:step/trailblazer/step-trailblazer-
<release>.spr,to:customer/<customer>/<customer>-addon/7.0/<customer>-addon-
7.0.x.spr
```

Install the Patch

The patch should be installed by the following command:

```
./spot --apply=to:step/trailblazer/step-trailblazer-<release>.spr
```

With customer components included, the command looks like this:

```
./spot --apply=to:step/trailblazer/step-trailblazer-
<release>.spr,to:customer/<customer>/<customer>-addon/7.0/<customer>-addon-
7.0.x.spr
```

STEP will automatically stop and start during the patch session.

In case of any deprecated parameters in the configuration, follow the instructions on the screen that explain how to correct and restart STEP.

```
./spot --start
```

Fallback

In the event of errors during patching, it may be necessary to restore STEP to a previous state.

Depending on the contents of the patch the following steps should be completed when reverting the patch:

1. Stop STEP

   ```
   ./spot --stop
   ```

2. Restore database
3. Restore configuration files
4. Redeploy STEP using a snapshot

   ```
   ./spot --apply=/workarea/<snapshot-env-date>.spr --sync --syncmode=delete
   ```

Using the snapshot and the above `--sync --syncmode=delete` command will entirely recover STEP and delete any files related to a failed patch-session.

Note: Refer to the relevant release note to check if restoring the database is required for the patch in question.

Patching Security

Stibo Systems only distributes software via the updates.stibosystems.com server or one of the official mirrors.

The update mirror web server is configured to only communicate via HTTPS (never plain HTTP) on port 443, with only the high security cipher suites using the Apache SSLCipherSuite 'HIGH' option and only communicating with clients which have a proper client certificate issued by the build system certificate authority (CA) of Stibo Systems. This Stibo-specific CA was created solely for the purpose of certifying various STEP-related infrastructures.

Unlike a standard website where an external CA-signed certificate is used for ease of access by multiple clients (users), the updates server has only one client that is allowed to communicate with it: the SPOT client. For this reason, Stibo Systems believes this to be a safer and stronger security approach – over using an external CA certificate – as it is not possible for a cyberattacker to use a fake certificate from a compromised external CA to gain access.

By taking this approach, some auditing tools may register a false positive and flag the server's certificate as self-signed. Because of this, security teams should configure these tools to trust Stibo Systems' CA to certify *stibosystems.com* domains

The client certificate required for communicating with the update mirror is included in the STEP installation package, and is used by the SPOT program to fetch both the software required for the initial installation and future application updates. Only the certificate used by the updates server will be trusted by SPOT for downloading these installation bits and updates.

All the certificates involved use 2048-bit RSA keys, so the system is considered secure against any man-in-the-middle attacker for the foreseeable future. Even with a valid client certificate, the operations allowed are severely limited to downloading only the licensed software produced by Stibo Systems and to saving customer-specific thin snapshots that do not contain software, so a compromised client would not be able to affect other customers or compromise other clients.

The SPOT program caches all files locally and validates contents using a SHA-1 hash before using the cached files, so the amount of traffic is kept as low as possible while ensuring the integrity of the cached files.

At no point will the STEP software communicate customer data back to the update mirrors at Stibo Systems. The thin snapshots uploaded to the release server contain only a list of versions of the installed STEP software components and they are only used by Stibo Systems to provide the best support to the STEP system.

Configuring a Private Updates Mirror

Requirements for running a private mirror server include:

- A 64-bit Linux host, not shared with STEP.
- 64-bit Java 8 (an updated version will be installed by SPOT, so the OS version is okay for bootstrapping).
- Enough storage to hold the entire mirror (currently 400 GB will suffice).
- Outgoing internet access to the Stibo Systems updates servers on port 443.
- Incoming access from the private network on port 443 for the SPOT hosts.
- A DNS entry on the local network that can be expected to never change, so *updates.example.com* would be preferable to *pc2016-02-13-room7-linux-testdl120g9.dhcp.example.com*.

The root mirrors that the private mirror connects to can be listed using `spot --mirrors`, but these are the current hosts:

- *dk1.updates.stibosystems.com*: Primary root mirror.
- *dk2.updates.stibosystems.com*: Secondary root mirror.
- *updates.stibosystems.com*: Fail-over mirror on a shared IP between the two root mirrors.

Outgoing TCP access on port 443 must be allowed to each of the root mirror IP addresses from the private mirror, this way the mirror has more upstream mirrors to pick from if one fails.

The mirror server listens on three ports:

- *10080*: The Admin port of dropwizard, which is used to serve HTTP requests that allow monitoring the health of the server. The init script uses this port to check if the server is running.
- *10081*: The stop port of Jetty. The init script uses this port to shut down the server in an orderly fashion. This port should not be accessed from outside the machine itself.
- *10082*: The HTTPS service port that serves the actual mirror. This port should not be accessed from outside the machine itself.

These ports are all internal to the host that the server runs on and external systems should not connect directly to them (with the possible exception of having a monitoring system talking to port 10080.)

Important: Do not configure any STEP systems to talk to the mirror on port 10082. Port redirection (as described in the next section) must be set up.

IPTables Rules

It is impossible to listen to port 443 when running the Java process as an unprivileged user. To account for this, a set of iptables rules must be used.

There are two ways to install the required rules: either run the mirror script as root when starting the server or set up iptables at the OS level. If the init script is called by root, then it will install the needed port redirection, but if the administrator tasked with maintaining the mirror does not have sudo access to this script, then the rules can be inserted into the */etc/sysconfig/iptables* config file, allowing the OS to load the rules at boot time.

These rules redirect all incoming requests to TCP port 443 over to port 10082 where the server listens.

To configure iptables on the server, switch to the root user and run the following command to view the current settings:

```
[root@mirror mirror]# /sbin/iptables -L -n --line-numbers
Chain INPUT (policy ACCEPT)
num  target     prot opt source      destination
1    ACCEPT     all  --  0.0.0.0/0   0.0.0.0/0   state RELATED,ESTABLISHED
2    ACCEPT     icmp --  0.0.0.0/0   0.0.0.0/0
3    ACCEPT     all  --  0.0.0.0/0   0.0.0.0/0
4    ACCEPT     tcp  --  0.0.0.0/0   0.0.0.0/0   state NEW tcp dpt:22
5    REJECT     all  --  0.0.0.0/0   0.0.0.0/0   reject-with icmp-host-
prohibited

Chain FORWARD (policy ACCEPT)
num  target     prot opt source       destination
1    REJECT     all  --  0.0.0.0/0    0.0.0.0/0   reject-with icmp-host-
prohibited

Chain OUTPUT (policy ACCEPT)
num  target     prot opt source       destination
```

In the output, there will be a line that displays `REJECT` as the `INPUT` type, and in its first column `(num)`, the line number is shown ('5' in the above example.) This line number will be the starting line for adding entries to the iptables configuration.

Once this information is known, run the following commands to add the needed port-opening entries:

```
[root@mirror mirror]# /sbin/iptables -I INPUT <line_number> -p tcp -m tcp --dport
443 -j ACCEPT
[root@mirror mirror]# /sbin/iptables -I INPUT <line_number> -p tcp -m tcp --dport
10082 -j ACCEPT
```

In the example above, the line number shown is '5', and therefore, the commands would look like the following commands:

```
[root@mirror mirror]# /sbin/iptables -I INPUT 5 -p tcp -m tcp --dport 443 -j
ACCEPT
[root@mirror mirror]# /sbin/iptables -I INPUT 6 -p tcp -m tcp --dport 10082 -j
ACCEPT
```

Afterwards, add the entries for redirection by executing these commands:

```
[root@mirror mirror]# /sbin/iptables -t nat -A PREROUTING -p tcp -m tcp --dport
443 -j REDIRECT --to-ports 10082
[root@mirror mirror]# /sbin/iptables -t nat -A OUTPUT -o lo -p tcp -m tcp --dport
443 -j REDIRECT --to-ports 10082
```

Once that has been done, the added entries can be checked by running the commands that follow:

```
[root@mirror mirror]# /sbin/iptables -L -n --line-numbers
Chain INPUT (policy ACCEPT)
num  target     prot opt source       destination
1    ACCEPT     all  --  0.0.0.0/0    0.0.0.0/0   state RELATED,ESTABLISHED
2    ACCEPT     icmp --  0.0.0.0/0    0.0.0.0/0
3    ACCEPT     all  --  0.0.0.0/0    0.0.0.0/0
4    ACCEPT     tcp  --  0.0.0.0/0    0.0.0.0/0   state NEW tcp dpt:22
5    ACCEPT     tcp  --  0.0.0.0/0    0.0.0.0/0   tcp dpt:443
6    ACCEPT     tcp  --  0.0.0.0/0    0.0.0.0/0   tcp dpt:10082
7    REJECT     all  --  0.0.0.0/0    0.0.0.0/0   reject-with icmp-host-prohibited

Chain FORWARD (policy ACCEPT)
num  target     prot opt source       destination
1    REJECT     all  --  0.0.0.0/0    0.0.0.0/0   reject-with icmp-host-prohibited

Chain OUTPUT (policy ACCEPT)
num  target     prot opt source       destination
```

```
[root@mirror mirror]# /sbin/iptables -L -n --line-numbers -t nat
Chain PREROUTING (policy ACCEPT)
num  target     prot opt source       destination
1    REDIRECT   tcp  --  0.0.0.0/0    0.0.0.0/0    tcp dpt:443 redir ports 10082

Chain INPUT (policy ACCEPT)
num  target     prot opt source       destination

Chain OUTPUT (policy ACCEPT)
num  target     prot opt source       destination
1    REDIRECT   tcp  --  0.0.0.0/0    0.0.0.0/0    tcp dpt:443 redir ports 10082

Chain POSTROUTING (policy ACCEPT)
num  target     prot opt source       destination
```

If everything looks to be correct, save the configuration so that it will be loaded each time the system reboots using these commands:

```
[root@mirror mirror]# /sbin/service iptables save
[root@mirror mirror]# /sbin/service iptables stop
[root@mirror mirror]# /sbin/service iptables start
```

Important: If the mirror server is a RHEL 7.x system, the above `/sbin/service iptables stop` and `/sbin/service iptables start` commands should be replaced with the following: `/bin/systemctl stop iptables` and `/bin/systemctl start iptables`.

Once complete, the */etc/sysconfig/iptables* config file should look similar to the following:

```
# Generated by iptables-save v1.4.7 on Tue Jun 21 14:16:10 2016
*nat
:PREROUTING ACCEPT [4595:497811]
:INPUT ACCEPT [1:28]
:OUTPUT ACCEPT [0:0]
:POSTROUTING ACCEPT [0:0]
-A PREROUTING -p tcp -m tcp --dport 443 -j REDIRECT --to-ports 10082
-A OUTPUT -o lo -p tcp -m tcp --dport 443 -j REDIRECT --to-ports 10082
COMMIT
# Completed on Tue Jun 21 14:16:10 2016
# Generated by iptables-save v1.4.7 on Tue Jun 21 14:16:10 2016
*filter
:INPUT ACCEPT [0:0]
:FORWARD ACCEPT [0:0]
:OUTPUT ACCEPT [434:47393]
-A INPUT -m state --state RELATED,ESTABLISHED -j ACCEPT
-A INPUT -p icmp -j ACCEPT
-A INPUT -i lo -j ACCEPT
-A INPUT -p tcp -m state --state NEW -m tcp --dport 22 -j ACCEPT
-A INPUT -p tcp -m tcp --dport 443 -j ACCEPT
-A INPUT -p tcp -m tcp --dport 10082 -j ACCEPT
-A INPUT -j REJECT --reject-with icmp-host-prohibited
-A FORWARD -j REJECT --reject-with icmp-host-prohibited
COMMIT
# Completed on Tue Jun 21 14:16:10 2016
```

Installing a Private Mirror

1. Satisfy all requirements mentioned above.
2. Make a note of the DNS name which all the SPOT hosts will be using. In this example we will call it *mirror.customer.com*.
3. Contact Stibo Systems Technical Services and request that a system name is created for the mirror. This must be human readable and unique. The system name in this example is *your-mirror*.
4. Create a directory for the mirror.

5. Unzip the SPOT foothold (must be newer than the March 2016 release).
6. Run: `./spot --enroll=mirror:your-mirror:mirror.customer.com`.
7. Run: `./spot --apply=to:updates/mirror/latest.spr`.
8. Edit the *mirror.yaml* file and review the options in the file. Some of them, particularly those dealing with mailing of errors, will need to be changed.
9. Run: `./mirror start`.
10. Your mirror should now be running on *mirror.customer.com*.
11. On a system with STEP installed, run the following command:

    ```
    ./spot --updates=https://mirror.customer.com --ping
    ```

12. As `root`, create a symlink to the mirror script into the appropriate sysv init directories using a command similar to the following:

    ```
    ln -s <mirror_home>/mirror /etc/rc3.d/S90stibo-updatesmirror
    ```

For example:

```
ln -s /home/mirror/mirror /etc/rc3.d/S90stibo-updatesmirror
```

Important: Do not run any of these commands as `root`. Make sure an unprivileged user exists for this mirror — e.g., `mirrorsw`.

Once the private mirror has been configured, it can be verified by running the following command on the STEP application server as the `stibosw` (or equivalent) user:

```
[stibosw@app1 step]$ ./spot --mirrors
Stibo Patch Operations Tool
Priority Id          Name                          Url                                [X]
100      <customer>  <customer> local mirror       https://mirror.customer.com         X
30       global      Auto failover mirror          https://updates.stibosystems.com
20       dk1         Primary mirror in Aarhus DK   https://dk1.updates.stibosystems.com
10       dk2         Secondary mirror in Aarhus DK https://dk2.updates.stibosystems.com

Please use spot --mirrors --updates={Url} to set the upstream mirror
```

The mirror can also upgrade itself using the init script by running: `./mirror upgrade`. The upgrade command calls the `spot --apply=to:updates/mirror/latest.spr` and `./mirror restart` commands.

If the iptables' rules have been added to the RHEL config file, the init script no longer needs root access and can be started by an unprivileged user. This is done by editing said user's *crontab* entries (such as `crontab -e`) and adding the following line:

```
@reboot <mirror_home>/mirror start
```

For example:

```
@reboot /home/mirror/mirror start
```

Preemptive Download

The mirror server can download files before the STEP systems ask for them. Doing this allows most files to be served from the local mirror without waiting for the upstream mirror, so better performance can be expected. This comes at the cost of more disk space being utilized and the possibility of downloading files that end up never being needed.

The download option has three possible values:

- `download: HISTORIC`: Downloads all the files available from the upstream mirror, regardless of age. This requires about 1.5 TB of space.
- `download: RELEASED`: Downloads newly released code as soon as it becomes available, this is the default and will steadily consume space. About 2 GB are consumed per month.
- `download: ON_DEMAND`: Nothing is downloaded until a client asks for it.

Note: When new content (e.g., monthly maintenance patches, add-on components, hotfixes, etc.) is downloaded to the mirror, it will be saved to the server's `<mirror_home>/content/takeout` directory (e.g., `/home/mirror/content/takeout`).

As no user is actively waiting for the preemptive downloads to complete and because the downloads can be quite large, the bandwidth consumed by the background downloads can be limited via the `bulkDownloadSpeedInMbitPerSecond` configuration option. The default limit is 10 Mb/s, so the expected lag after a release of STEP until the mirror is in sync should be less than an hour.

The bulk download speed limit is applied to the download of newly released files and historically released files separately, so if a historic download is running, then the two bulk processes can consume twice the speed limit in total.

If downloads take a long time to complete, it could be because the network or the upstream mirror is overloaded. To avoid contributing to the problem the bulk download threads will sleep for a while after completing a download. The amount of time to sleep after a download can be specified using the `bulkBackoffFactor` option, which defaults to '1.5'.

For example, if a download takes two seconds then a *bulkBackoffFactor* of '1.5' means that the process will sleep three seconds before downloading the next file.