

**T** · · Systems · · ·

**rvs<sup>®</sup> OFTP Proxy**

Version 6.02

User Manual

The products listed in this manual are protected by copyright.

rvs rvs<sup>®</sup> OFTP Proxy  
Version 6.02  
User Manual

© 2020 by T-Systems International GmbH  
Holzhauser Str. 4-8  
13509 Berlin

This manual is protected by copyright. All rights reserved. No part of this book may be used or reproduced in any form or by any means including photocopies, microfilm or any other means or stored in a database or retrieval system without obtaining prior permission from T-Systems. Rights are also reserved as far as lectures, radio and television is concerned.

We reserve the right to make changes to the content of this manual without giving prior notice.  
T-Systems is not liable for technical or printing errors or defects in this manual. Moreover, T-Systems shall not be liable for damage which is directly or indirectly caused by delivery, performance and use of this material.

---

<b>1</b>	<b>Introduction</b>	<b>5</b>
1.1	Basics	5
1.2	rvs <sup>®</sup> OFTP Proxy Architecture	5
1.3	Representation Means	6
1.4	Target Group	7
<b>2</b>	<b>Installation</b>	<b>9</b>
2.1	System Requirements	9
2.2	Obtaining a License	9
2.3	Set Up rvs <sup>®</sup> OFTP Proxy	9
2.4	New Installation rvs <sup>®</sup> OFTP Proxy	10
2.4.1	Installation on Windows Systems	10
2.4.2	Installation on Linux / UNIX Systems	13
2.5	Update Installation of rvs <sup>®</sup> OFTP Proxy	14
<b>3</b>	<b>Configuration</b>	<b>15</b>
3.1	Directory Structure	15
3.2	Setting Up of rvs <sup>®</sup> OFTP Proxy	15
3.2.1	Setting Up of Bastion RMI Parameters	15
3.2.2	Setting Up of Logging Parameters	16
3.2.3	Setting Up of Heartbeat Parameters	17
3.3	Setting up of a TLS Connection	18
3.3.1	Setting up of a RMI Communication via TLS Protocol	18
3.3.2	Key Management via keytool Script	18
<b>4</b>	<b>Start of rvs<sup>®</sup> OFTP Proxy</b>	<b>21</b>



## 1 Introduction

This chapter describes the configuration and the functionality of rvs<sup>®</sup> OFTP Proxy implementation in rvsEVO.

**Hint:** rvs<sup>®</sup> OFTP Proxy Version 6.00 onwards is only available for rvsEVO Version 6.00 onwards.

Up to version 5.04 rvs<sup>®</sup> OFTP Proxy is an universal software product, which works together with any other OFTP product.

### 1.1 Basics

Internet is increasingly used also for transferring the commercial and business-critical data of the content. Such advantages are deciding factors for this, such as its wide distribution, enormous available bandwidths and the reasonable transfer costs.

These advantages are accompanied by the risks to be considered seriously: in internet all data are susceptible to the attacks of their confidentiality and integrity. Their authentication is also not ensured without further ado. This may result into that the data are repudiated by its obvious sender. Such risk is encountered with suitable protocols, such as TLS and HTTPS. For CAD data and also for EDI and any other data, OFTP version 2 has been standardized as a secured internet-protocol.

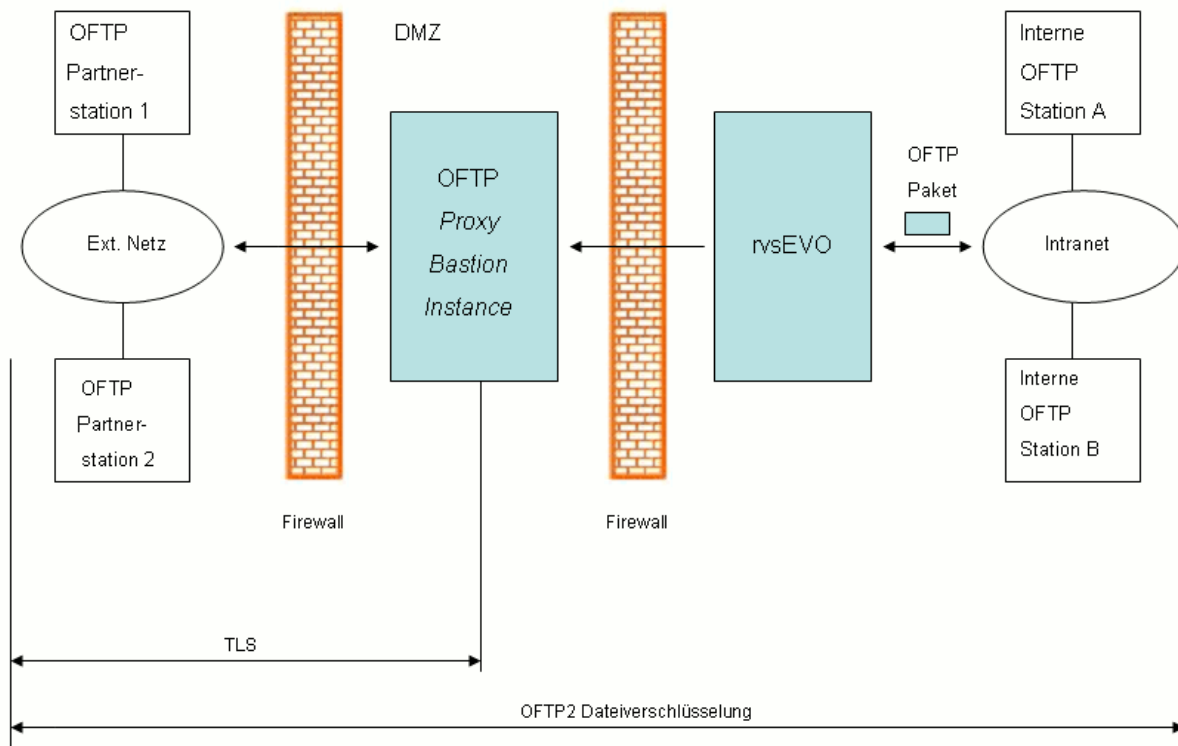
Any further risk arises through malware, which is spread widely and automatically in internet. Virus, worms, Trojan and other harmful programs can damage the IT-infrastructure of the company considerably. In order to minimize this risk, network areas are developed which are separated from the corporate networks through the firewalls (also referred to as demilitarized Zone - DMZ). For example, http-proxies are set in DMZ, which transport the user data without allowing direct connections between internet and intranet. rvs<sup>®</sup> OFTP Proxy offers the similar solution, which can be implemented in DMZ in order to transfer the controlled data of OFTP protocol version 1 and 2 between intranet and internet.

Please read the release notes `$RVS_HOME/doc/readme.txt` for information about supported platforms.

**Hint:** In chapter 1.3 "Representation Means" you can find a discription of `$RVS_HOME`

### 1.2 rvs<sup>®</sup> OFTP Proxy Architecture

Two instances The rvs<sup>®</sup> OFTP Proxy is run with two instances: bastion instance and a connection instance, which is integrated in rvsEVO. The following figure should illustrate these facts:



The connections are received from the external network by using bastion instance. These external connections are forwarded to rvsEVO.

The other way round rvsEVO establishes connections to the external network via bastion instance.

Communication between rvsEVO and bastion instance is executed exclusively through the connections built by rvsEVO. Communication is restricted to one port, the configurable RMI (Remote Method Invocation)-port of the bastion instance.

**Proxy listener** For being accessible for the external stations one or more Proxy listeners are launched by rvsEVO. rvsEVO uses the Proxy listener to establish a connection to the partner station.

Altogether, all functional configurations, keys and certificates required for TLS communication are stored exclusively in rvsEVO.

### 1.3 Representation Means

This chapter describes the typographic conventions used in this manual and explains the meaning of specially highlighted expressions.

#### Typographic conventions

Instructions begin with a bullet.

- Other lists begin with the en dash.

---

## Character Styles

<b>Courier</b>	Commands, menu commands, file names, path names, programs, examples, scripts, options, qualifiers, data sets, fields, modes, window names, dialog boxes and statuses
<b>BOLD and in CAPITAL LETTERS</b>	Parameters, environment variables, variables
<b>"Inverted commas"</b>	Links to other manuals, sections and chapters, literature
<b>Bold</b>	Important terms, names of operating systems, pro-penames, buttons, function keys.

### Directories

**\$RVS\_HOME** As user directories are found on different locations for the different operating systems we use the variable **\$RVS\_HOME** in this manual. Default value: `C:\Programs\rvsProxy` for Windows

Substitute the variable with your correct path.

### 1.4 Target Group

This manual is meant for regular users of rvs<sup>®</sup> OFTP Proxy as well as administrators. It provides an overview of the basic rvs<sup>®</sup> OFTP Proxy functions.

**Skills** The following skills are required to be able to use rvs<sup>®</sup> OFTP Proxy:

- good knowledge of the current operating system
- knowledge of the communications techniques in use (TCP/IP, TLS)

We recommend to read this manual before working with rvs<sup>®</sup> OFTP Proxy.





## 2 Installation

This chapter describes the system requirements as well as the rvs<sup>®</sup> OFTP Proxy installation procedure.

### 2.1 System Requirements

To successfully operate rvs<sup>®</sup> OFTP Proxy you need the following software:

- Software
- Operating system: Windows 8.1 / Windows 10 / Windows Server 2012 R2 / Windows Server 2016 / Windows Server 2019 or UNIX (AIX, Solaris/SunOS, HP-UX, Linux, zLinux).
  - Java runtime environment (JRE) Version 1.8\_XX or higher.

**Hint:** With Windows and Linux systems the Java Runtime Environment is distributed during the installation. With UNIX systems you can select an installed JRE during the installation.

**Attention:** Please note that there are different installers for 32-bit and 64-bit operating systems for Windows and Linux systems.

You can download rvs<sup>®</sup> OFTP Proxy from the following web page:

<https://servicenet.t-systems.com/business-integration/software/958408>

If that is not possible for you, please contact your sales partner:

Phone from Germany: 0800 664 77 45;

Phone from other countries: +49 375 606 19 902

E-Mail: rvs-service@t-systems.com

We can send the software also on DVD to you.

Networks: Communication line based on TCP/IP und TLS.

### 2.2 Obtaining a License

From rvs<sup>®</sup> OFTP Proxy version 6.00 onwards licensing occurs via rvsEVO.

### 2.3 Set Up rvs<sup>®</sup> OFTP Proxy

To establish a connection via rvs<sup>®</sup> OFTP Proxy please follow the procedure given below.

- Install rvs<sup>®</sup> OFTP Proxy. The versions of rvsEVO and rvs<sup>®</sup> OFTP Proxy must be identical.
- Add a Proxy Bastion in rvsEVO (see rvsEVO User Manual chapter „Add Proxy Bastion“)
- Configure a Proxy listener in rvsEVO (see rvsEVO User Manual chapter "Setting up of a Local Station" section „Add Listener“)

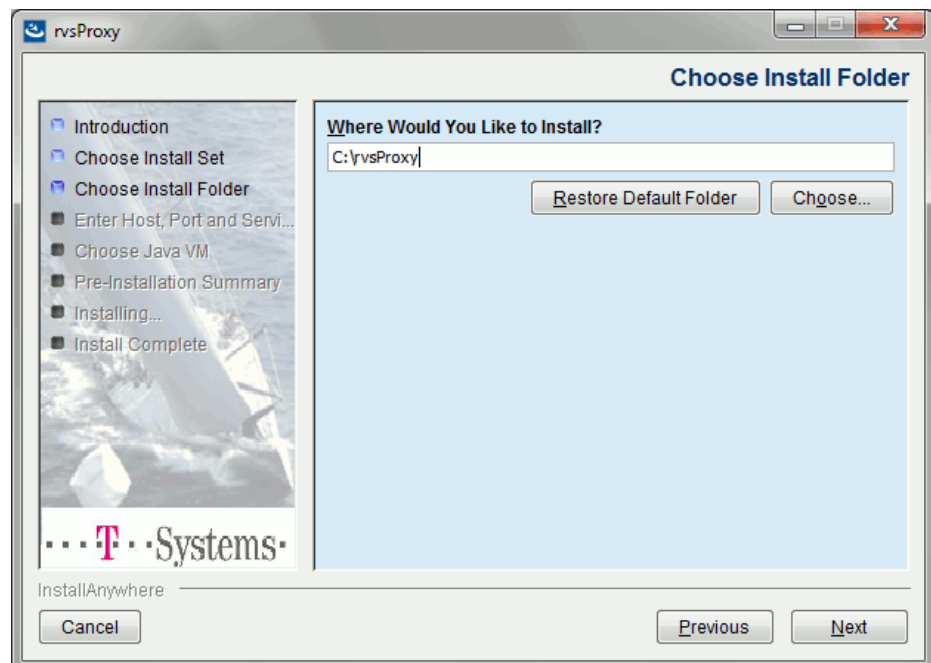
- Add a neighbour station in rvsEVO for communication via Proxy TCP/IP or Proxy TLS (see rvsEVO User Manual chapter "Add a Neighbour Station").

## 2.4 New Installation rvs<sup>®</sup> OFTP Proxy

This chapter describes the installation of rvs<sup>®</sup> OFTP Proxy. Please read chapter 2.1 "System Requirements" before installing the software.

### 2.4.1 Installation on Windows Systems

- Start Windows and log in as Windows user with administrator rights.
- Start the installer  
`rvsProxy-setup-X.XX.XX-Windows32.exe` or  
`rvsProxy-setup-X.XX.XX-Windows64.exe` (for 32 or 64 bit OS)  
 by double-clicking or using the Windows command `Start -> Run. X.XX.XX` accords to the version of rvs<sup>®</sup> OFTP Proxy.
- With the first window you can choose the language of the installation wizard.
- Thereafter you get information on installation of rvs<sup>®</sup> OFTP Proxy.
- In the dialog below you can indicate the rvs<sup>®</sup> OFTP Proxy destination directory. If you choose an existing directory you were asked to choose another directory name or to install an update. Choose a non-existing directory name for a new installation.



- Define the RMI parameters (Hostname or IP address, port and service name) for Bastion instance. The same parameters are needed for adding a Proxy Bastion in rvsEVO.

**Hints:** Using the NAT (**N**etwork **A**ddress **T**ranslation) procedure, the private IP address is to be added via `$(RVS_HOME)/bin/bastion.cmd` file (see chapter 3.2.1 "Setting Up of Bastion RMI Parameters").

Please note that the **Bastion Service Name** must be unique if multiple Proxy Bastion instances are defined in rvsEVO.

Enter Host, Port and Service Name

Please enter bastion hostname (or IP), port and service name for RMI.

Bastion Host

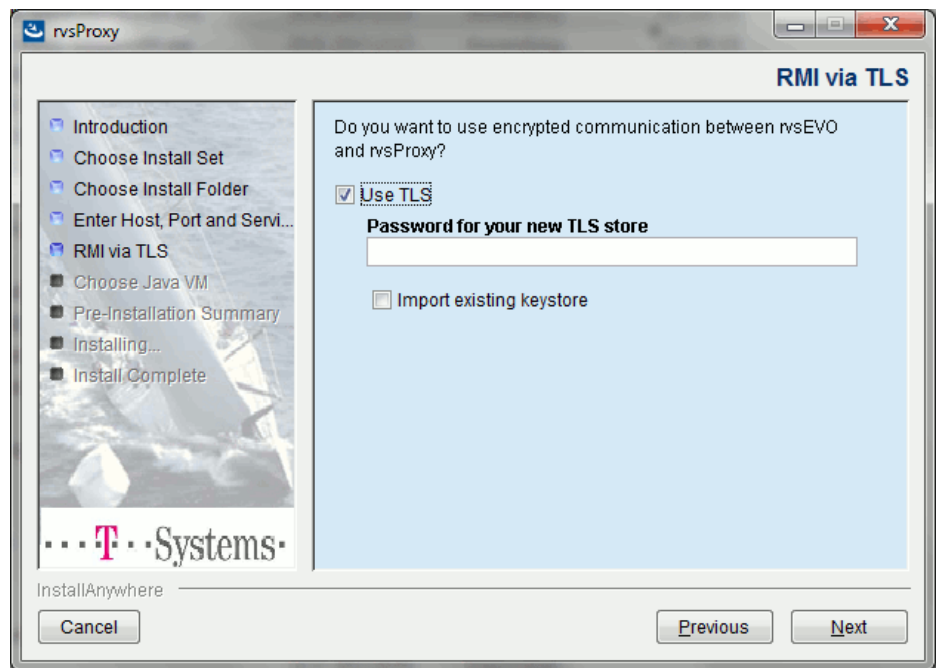
Bastion Port

Bastion Service Name

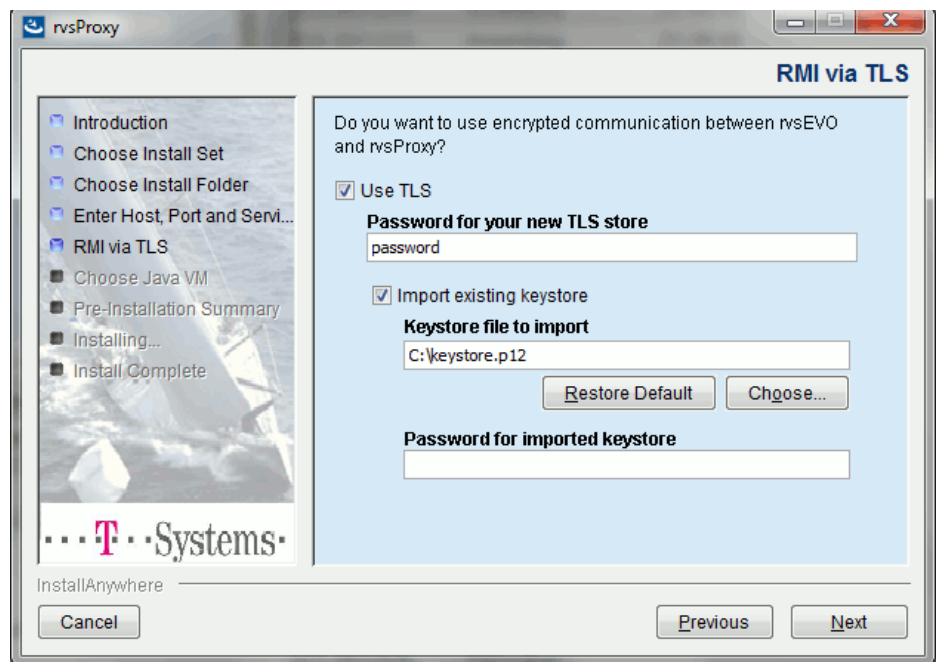
Cancel Previous Next

- From Version 6.01 onwards rvs<sup>®</sup> OFTP Proxy can communicate with rvsEVO via TLS protocol.

Activate the checkbox **Use TLS** for encrypted communication and define the password for the TLS store. By default a keypair is generated during installation.



Activate the checkbox **Import existing keystore** to import an existing keystore and define path and name of the keystore to be imported as well as the keystore password.



- In the next dialog you are given a brief overview of the selections you have made (installation directory, ...). The required and the currently available disk space is also displayed. Press the **Next** button to start installation and to copy the installation files into the directories you specified.

- The last dialog informs you about the successful installation of rvs<sup>®</sup> OFTP Proxy.

## 2.4.2 Installation on Linux / UNIX Systems

Installation on UNIX systems runs nearly analog to an installation on Windows systems (see chapter 2.4.1 "Installation on Windows Systems"). The installation file for Generic UNIX is named `rvsProxy-setup-X.XX.XX-GenericUnix.bin` (X.XX.XX accords to the version of rvs<sup>®</sup> OFTP Proxy).

Use the following installation files for Linux:

`rvsProxy-setup-X.XX.XX-Linux32.bin` for 32 bit OS or

`rvsProxy-setup-X.XX.XX-Linux64.bin` for 64 bit OS.

### Hints:

The installer is provided as zip file and must be unzipped before the installation is started.

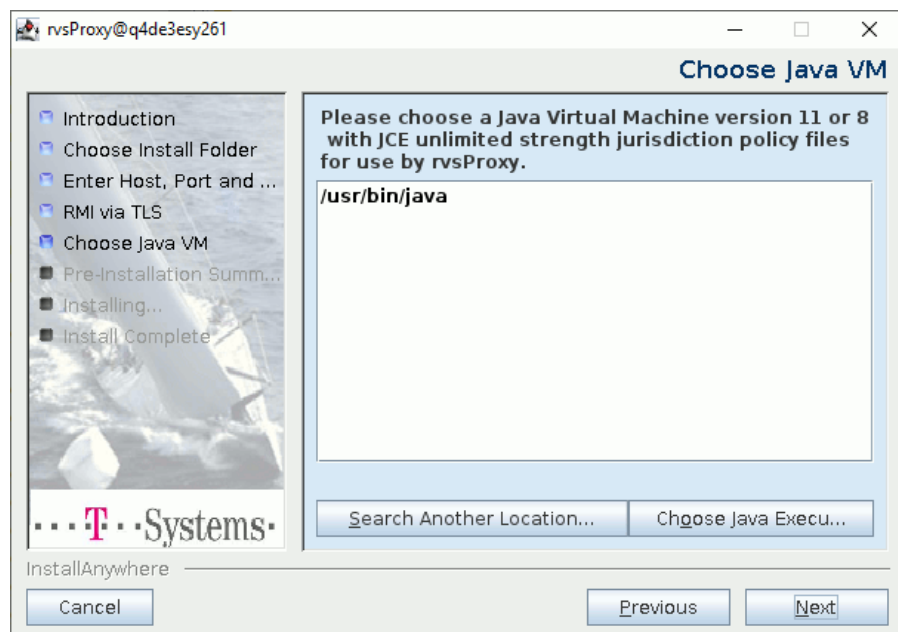
Call the installation script as shell script.

### Example:

```
sh ./rvsProxy-setup-6.02.05.bin
```

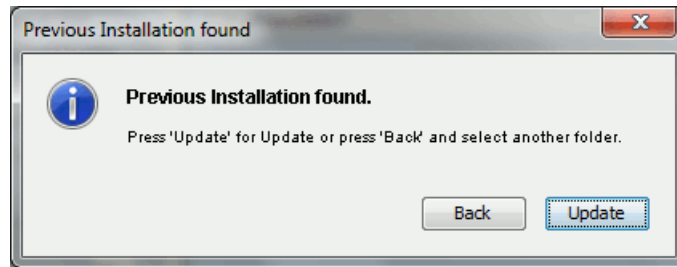
Java Runtime Environment

There is an additional dialog for defining the JRE (Java Runtime Environment) for rvs<sup>®</sup> OFTP Proxy operation. You are asked to choose the JVM (Java Virtual Machine) which is already installed on your system. The installer searches for installed installations and proposes the found versions in a dialog. rvs<sup>®</sup> OFTP Proxy has been released for use of JVM version 1.8.XX onwards.



## 2.5 Update Installation of rvs<sup>®</sup> OFTP Proxy

- Start the installer as described in chapter 2.4.
- Choose the directory of the installation to be updated in dialog window **Choose Install Folder**. Confirm your entry via a mouse click on the **Next** button.
- In the next dialog you were informed about the existing installation and asked to choose another directory name or to install an update. Chosse **Update** button for an update installation.



- Thereafter you are given a brief overview of selections you have made.
- The last dialog informs you about the successful update installation.

## 3 Configuration

The configuration of rvs<sup>®</sup> OFTP Proxy (basic configuration, stationlist, encryption ...) is done in rvsEVO. Only RMI parameters, logging level and the time interval for sending heartbeats is configured in the bastion instance.

### 3.1 Directory Structure

After installation of rvs<sup>®</sup> OFTP Proxy the installation folder includes the following directories:

- bin - batch (Windows) or shell scripts (Unix)
- conf - files of user configuration
- jre - Java Runtime Environment (only Windows and Linux)
- lib - libraries
- log - log files
- system - files of system configuration
- Uninstall\_rvsProxy - files for deinstallation of rvs<sup>®</sup> OFTP Proxy.

### 3.2 Setting Up of rvs<sup>®</sup> OFTP Proxy

\$RVS\_HOME/conf folder includes the following configuration files:

- RemoteListenerProperties.properties (heartbeat configuration)
- rvsLogger.xml (logging configuration)

The bastion RMI parameters can be changed in

\$RVS\_HOME\bin\bastion.cmd (Windows) / bastion.sh (UNIX) startscript and for the Windows service in bastionService.ini file.

#### 3.2.1 Setting Up of Bastion RMI Parameters

The RMI parameters of the bastion instance can be changed in \$RVS\_HOME\bin\bastion.cmd (Windows) / \$RVS\_HOME/bin/bastion.sh (UNIX) startscript.

Example (bastion.cmd):

```
-Djava.rmi.server.hostname=127.0.0.1
com.gedas.rvs.proxy.Bastion -u rvs -p gedas -h
127.0.0.1 -r 4567 -s service -t -kn
C:\rvsProxy2\system\proxy-ssl.p12 -kp
VWUv3Un7tVQIwkKIs0y646q73x9ElxVG
```

Hostname or IP address is to be changed twice: in

**-Djava.rmi.server.hostname** parameter and in parameter **-h**. **-u** and **-p** parameter should not be changed.

**Hints:**

For Windows service `$RVS_HOME/bin/bastionService.ini` file is to be edited.

When using NAT (**N**etwork **A**ddress **T**ranslation) procedure: the value of **-Djava.rmi.server.hostname** Java parameter is the public IP address and the value of **-h** parameter is the private IP address.

#### RMI Parameters

Parameter	Description
<b>-Djava.rmi.server.hostname</b>	Hostname or IP address of the server, on which the Bastion instance has been installed. When using NAT procedure: public IP address.
<b>-h &lt;hostname IP-Adresse&gt;</b>	Hostname or IP address of the server, on which the Bastion Instance has been installed. When using NAT procedure: private IP address
<b>-r &lt;port&gt;</b>	Port of the Bastion Instance for the RMI-communication, on which a listener is started.
<b>-s &lt;service name&gt;</b>	Name of the Bastion Service. Default: service
<b>-t</b>	RMI via TLS (Default: RMI via TCP)
<b>-kn</b>	Path and name of the keystore
<b>-kp</b>	Encrypted keystore password

**Important:** Set up the parameters also in rvsEVO (see rvsEVO User Manual chapter „5.6.3 Setting up Proxy Bastion“)

### 3.2.2 Setting Up of Logging Parameters

The logging of rvs® OFTP Proxy can be set up in `$RVS_HOME/conf/rvsLogger.xml` file.

monitor.log The monitor messages are saved in `$RVS_HOME/log/monitor.log` file. A new monitor.log file is generated daily and the old monitor.log file is renamed in „monitor.log“ plus date pattern plus counter. Also a new file is generated if the maximum file size is reached.

Maximum file size and maximum number of monitor.log files can be defined in XML element `appender name="monlog"` in `$RVS_HOME/conf/rvsLogger.xml` file:

- Maximum size: use the function `MaxFileSize`

**Syntax:**

```
<param name="MaxFileSize" value="10MB"/>
```

- Maximum number of log files: use the function **MaxRollFileCount**

**Syntax:**

```
<param name="MaxRollFileCount" value="50"/>
```



- rsvProxy.log** The error messages of rvs<sup>®</sup> OFTP Proxy server are saved in `$RVS_HOME/log/rsvProxy.log` file.
- Maximum file size and maximum number of proxy.log files can be defined in XML element `appender name="rsvlog"` in `$RVS_HOME/conf/rsvLogger.xml` file:
- Maximum size: use the function `MaxFileSize`
- Syntax:
- ```
<param name="maxFileSize" value="2097152"/>
```
- Maximum number of log files: use the function **maxBackupIndex**
- Syntax:
- ```
<param name="maxBackupIndex" value="10"/>
```
- Log Level** The log level for log messages in `rsvProxy.log` file can be defined in element `priority` in `$RVS_HOME/conf/rsvLogger.xml` file.
- The following values are possible: "DEBUG", "INFO", "WARN" or "ERROR".
- Hint:** Only edit the log level in element `priority` if instructed to do so by technical support.
- The log level for log messages in `monitor.log` file can be defined in XML logger element `Name=Monitor` in `$RVS_HOME/conf/rsvLogger.xml` file.
- The following values of parameter `level` are possible: "INFO", "WARN" oder "ERROR".

### 3.2.3 Setting Up of Heartbeat Parameters

You can configure a time interval for sending heartbeats to prevent the interrupt of the connection to rvsEVO if there is no transmission. This functionality can be set up in parameter `listener.controller-thread.sleep-time` and `listener.controller-thread.keep-alive-counter` in `$RVSPROXY_HOME/conf/RemoteListenerProperties.properties` file.

Use the `listener.controller-thread.sleep-time` parameter to set up the sleep time (in milliseconds) of the controller thread before it checks next time the status of the listener.

Use the `listener.controller-thread.keep-alive-counter` parameter to define the number of checkups before the next heartbeat is sent.

rsv<sup>®</sup> OFTP Proxy forwards certificates to rvsEVO for verifying.

Use the `rsv_evo.callback.tls_verification_timeout` parameter to define the period of time (in milliseconds) after which the connection to the partner is closed if rvsEVO sends no validity confirmation of the certificate.

Use the `rvs_evo.callback.polling_timeout` parameter to define the time interval (in milliseconds) between sending of heartbeats for maintenance of the callback line.

### 3.3 Setting up of a TLS Connection

From version rvs<sup>®</sup> OFTP Proxy 6.01 upwards the connection between Bastion Instance and Connection Instance (rvs<sup>®</sup> OFTP Proxy and rvsEVO) can be encrypted via TLS protocol. For encrypted communication a private keypair and a public X.509 certificate is needed. In chapter 2.4.1 "Installation on Windows Systems" you come to know how to generate a keypair during installation of rvs<sup>®</sup> OFTP Proxy. In chapter 3.3.2 "Key Management via keytool Script" you get information about creating keypairs via `keytool` script. The X.509 certificate is exported into the trustkeystore automatically.

#### 3.3.1 Setting up of a RMI Communication via TLS Protocol

Necessary steps for line encryption via TLS protocol between Bastion and Connection Instance:

- Install rvs<sup>®</sup> OFTP Proxy for using TLS or import a keypair into an existing rvs<sup>®</sup> OFTP Proxy .
- Copy the truststore which includes the X.509 certificate into the rvsEVO directory.
- Specify this truststore in Proxy Bastion parameters of rvsEVO (see rvsEVO User Manual chapter 'Proxy Bastion').

#### 3.3.2 Key Management via keytool Script

Use the `keytool.cmd` script for managing keypairs and certificates and creating a new keypairs.

##### Syntax:

```
keytool -k <Datei> -t <Datei> -p <Name> -u <Datei> -l
<Datei> -lp <Datei> -g -i <Datei> -ia -ip -e -h -? -
verbose
```

Requested Parameters of keytool Script.

Parameter	Description
<b>-k &lt;file&gt;</b> <b>-keystore</b>	Path and name of the target keystore file in rvs <sup>®</sup> OFTP Proxy.
<b>-t &lt;file&gt;</b> <b>-trustore</b>	Path and name of the target trustkeystore file for the X.509 certificate.
<b>-p &lt;name&gt;</b> <b>-password</b>	Password for the target keystore file.

## Requested Parameters of keytool Script.

Parameter	Description
<b>-u &lt;File&gt;</b> <b>-update &lt;File&gt;</b>	Update certificate in trustkeystore Further requested parameters: <b>-k, -t, -p</b>
<b>-g</b> <b>-generate</b>	Generate keypair and export self signed certificate into trustkeystore. Further requested parameters: <b>-k, -t, -p</b>
<b>-l &lt;File&gt;</b> <b>-list &lt;File&gt;</b>	List keypairs of specified PKCS#12 file Further requested parameter: <b>-lp</b>
<b>-lp &lt;File&gt;</b> <b>-list-password &lt;File&gt;</b>	Password of PKCS#12 file which keypairs should be listed. (See parameter <b>-l</b> )
<b>-i &lt;File&gt;</b> <b>-import &lt;File&gt;</b>	Import keypair from specified PKCS#12 file into keystore of rvs <sup>®</sup> OFTP Proxy and export the public certificate into the trustkeystore. Further requested parameters: <b>-k, -t, -p, ip, ia</b>
<b>-e &lt;data string&gt;</b> <b>-encrypt &lt;data string&gt;</b>	Encrypt given data string
<b>-ia &lt;name&gt;</b> <b>-import-alias &lt;name&gt;</b>	The key alias to import (See parameter <b>-i</b> )
<b>-ip &lt;password&gt;</b> <b>-import-password &lt;password&gt;</b>	Password of the source keystore (See parameter <b>-i</b> )
<b>-help</b> <b>-h</b>	Requests help information
<b>-?</b>	Requests help information
<b>-verbose</b>	Verbose message output

**Example: Import of key pair**

In the following example a key pair with alias `LOC.2` is imported from `NEUrmITlsKeyStore.p12` keystore with password `tls1` into `c:\rvsPROXY\system\rmITlsKeyStore.p12` keystore with password `tls`. The x.509 certificate is saved in `c:\rvsPROXY\system\rmITlsTrustStore.p12` truststore.

```
C:\rvsProxy\bin>keytool -i "c:\rvsPROXY\system\
NEUrmITlsKeyStore.p12" -k "c:\rvsPROXY\system\
rmITlsKeyStore.p12" -t "c:\rvsPROXY\system\
rmITlsTrustStore.p12" -- ip tls1 -p tls --ia LOC.2
```

**Hints:**

You get the alias via `keytool -list`. Please see the following example.

If a key pair is renewed (e.g. after expiring) also the truststore in rvsEVO must be exchanged.

**Example: list keystore**

In following example the content of keystore

`c:\rvsPROXY\system\rmiTlsKeyStore.p12` is displayed.

Password is `tls`.

```
C:\rvsProxy\bin>keytool -l "c:\rvsPROXY\system\rmiTlsKeyStore.p12" --lp tls
```

**Output:**

```
C:\rvsProxy\bin>set CLASSPATH=C:\rvsProxy\lib\rvs-proxy-6.02.00-SNAPSHOT.jar
```

```
C:\rvsProxy\bin>C:\rvsProxy\jre\bin\java.exe -Djava.security.manager=default -Dj
```

```
ava.security.policy=C:\rvsProxy\system\java.policy com.tsystems.rvs.evo.proxy.cl
```

```
i.KeyTool -l "c:\rvsPROXY\system\rmiTlsKeyStore.p12" -lp tls
```

```
* [LOC.2 valid to: 30.08.2017 14:21]
```

Alias is `LOC.2`

`valid to: 30.08.2017 14:21` indicates the expiration date of the key.

## 4 Start of rvs<sup>®</sup> OFTP Proxy

Windows By default rvs<sup>®</sup> OFTP Proxy is installed as a Windows service.

**Note:** The term service means a program that can be started from the operating system and works in the background.

rvs<sup>®</sup> OFTP Proxy as Windows service can also be started in the command line via the batch script `$RVS_HOME\bin\bastionService.exe`

### Syntax:

```
bastionService <-install [ini]> <-configure [ini]>
<-remove [ini]> <-debug [ini]> <-run [ini]>
```

Optional bastionService Parameters:

Parameter	Description
<b>-install [ini]</b>	Installs rvs <sup>®</sup> OFTP Proxy as service. <code>\$RVS_HOME\bin\bastionService.ini</code> is used by default.
<b>-configure [ini]</b>	For new configuration of an installed rvs <sup>®</sup> OFTP Proxy. <code>\$RVS_HOME\bin\bastionService.ini</code> is used by default.
<b>-remove [ini]</b>	Removes rvs <sup>®</sup> OFTP Proxy as service. <code>\$RVS_HOME\bin\bastionService.ini</code> is used by default.
<b>-debug [ini]</b>	For starting a console application for debugging. <code>\$RVS_HOME\bin\bastionService.ini</code> is used by default.
<b>-run [ini]</b>	Starts rvs <sup>®</sup> OFTP Proxy as service on console. <code>\$RVS_HOME\bin\bastionService.ini</code> is used by default.

**Hint:** After installation of rvs<sup>®</sup> OFTP Proxy as service, you can find rvs<sup>®</sup> OFTP Proxy in the list of system services (Start -> Control Panel -> Administrative Tools -> Services). If you want to start rvs<sup>®</sup> OFTP Proxy with each system start, you can set the startup type to Automatic by choosing the `Automatic` entry from the combo box in the **Startup type** input field.

With `$RVS_HOME\bin\bastion.cmd` script you can start rvs<sup>®</sup> OFTP Proxy via command line. In this case the RMI parameters of `bastion.cmd` file are used.

UNIX / Linux Start rvs<sup>®</sup> OFTP Proxy on UNIX / Linux systems via the shell script `$RVS_HOME\bin\bastion.sh`.



---

Symbols  
\$RVS\_HOME 7  
B  
bastion 5  
bastionService Parameters 21  
H  
heartbeats 17  
J  
JRE 9, 13  
JVM 13  
L  
logging 16  
M  
monitor.log 16  
N  
NAT 11  
O  
Operating system 9  
R  
RemoteListenerProperties.properties 15  
RMI parameters 15  
rvsLogger.xml 15  
rvsProxy.log 17  
T  
Typographic conventions 6