



# Proxy 1

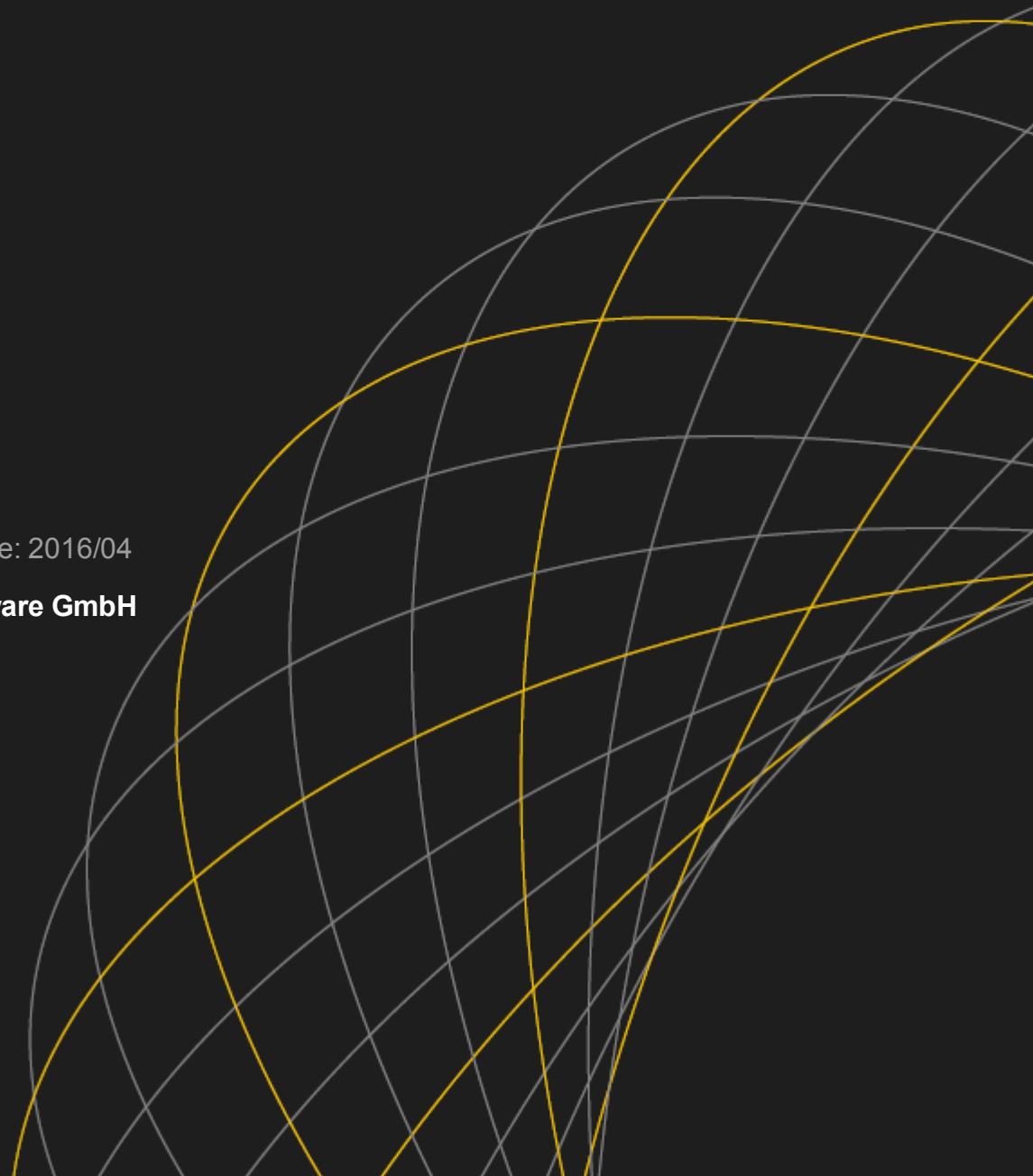
ONE Automation Platform

# Documentation

Version: 1.0.2

Publication Date: 2016/04

**Automatic Software GmbH**



# 1 Welcome to the World of Automic!

Automating your business processes – that's what the Automation Engine is about. Financial transactions, software deployment or database operations, to name just a few, with any of these you may adapt the Automation Engine individually to your needs – and relieve every day work from recurrent tasks. This documentation is meant to provide all the necessary information for making the best use of the Proxy. It is the perfect addition to the Automation Engine and supplies features which make the Automation Engine even more scalable and add special configuration options.

We are constantly busy developing the Proxy and the Automation Engine further – and look forward to hearing from you!

For any kind of suggestions, wishes and questions just contact our [support](#)! Your input will help us to make our products even better.

We hope you will be successfully using the Proxy – and wish you the fun in using it that we have developing it.

**Your Automic Team**

# Copyright

Automic® and the Automic logo® are trademarks owned by Automic Software GmbH (Automic). All such trademarks can be used by permission only and are subject to the written license terms. This software/computer program is proprietary and confidential to Automic Software and is only available for access and use under approved written license terms.

This software/computer program is further protected by copyright laws, international treaties and other domestic and international laws and any unauthorized access or use gives rise to civil and criminal penalties. Unauthorized copying or other reproduction of any form (in whole or in part), disassembly, decompilation, reverse engineering, modification, and development of any derivative works are all strictly prohibited, and any party or person engaging in such will be prosecuted by Automic.

No liability is accepted for any changes, mistakes, printing or production errors. Reproduction in whole or in part without permission is prohibited.

© Copyright Automic Software GmbH. All rights reserved.

---

# Contents

<b>1 Welcome to the World of Automic!</b>	<b>1</b>
<b>2 Proxy</b>	<b>4</b>
2.1 General Considerations	4
2.2 Load Balancing	5
2.2.1 Inactive Load Balancing	5
2.2.2 Active Load Balancing	5
2.3 Terminology	6
<b>3 Installation</b>	<b>8</b>
3.1 Workflow	8
<b>4 Configuring the Client Proxy</b>	<b>11</b>
4.1 Structure	11
4.2 Example	14
<b>5 Release Notes - PROXY Version 1</b>	<b>18</b>
<b>6 Technical Support</b>	<b>20</b>

## 2 Proxy

The Proxy is an additional service program that can only be used in combination with the Automation Engine.

Proxy is supplied as a separate product.


### 2.1 General Considerations


 The Proxy as of version 2 works with Automation Engine version 11.2 and above exclusively.

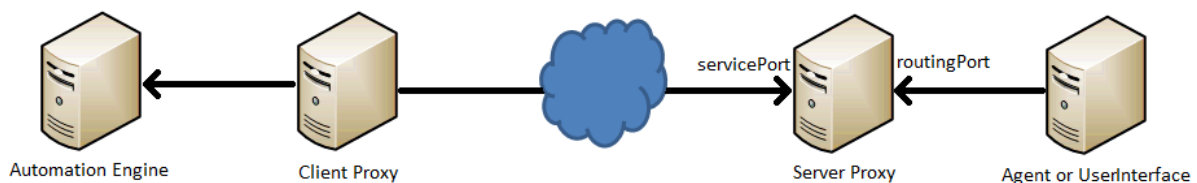
Different components such as agents, dialogs (UserInterface), and APIs (CallAPI, JavAPI) connect to the server component of the Automation Engine (referred to as the "AutomationEngine" from v9). A dedicated connection to the communication processes (CPs) of the server component is set up for each component. Each of these connections must be specifically permitted when using a firewall, which involves increased configuration effort.


The Proxy application allows you to combine the CP connections of the Automation Engine (AE) components (agents, dialogs, etc.) into one connection and reverse the direction in which the connection was set up. This requires an instance of the Proxy on both the component and server side. The client proxy concerns every instance of the Proxy that runs on the side of the CPs. An instance that runs on the side of the AE components (such as agents) is referred to as a server proxy. Each client proxy also requires a server proxy. The client and server proxy are run as a proxy pair, meaning that the server proxy allows connection of one client proxy.


This way, it is only necessary to configure the connections from the Proxy in the firewall once. New components (such as agents) that should connect to the server component must be allocated to one respective server proxy only.

 **Tip:** Run at least two proxy pairs in parallel for the same AE component and link them to one another (INI file of the client proxy: [OTHER\_SP\_LIST] section). This increases reliability in case one proxy pair fails.

 **Information:** The server proxy does not require an INI file as it receives the relevant settings from the client proxy via the service connection. Both the server and client proxy are configured with the INI file on the client proxy side.




 The proxy is invisible in "system overview" of the Automation Engine

 You may use the proxy together with the net area function in Automation Engine, where the agent would connect via the proxy with the CPs contained in one net area. This applies, when using `cpSelection = 1` in the proxy's .INI. For details on net area configuration and setup please see the relevant chapter in the Automation Engine documentation.

## 2.2 Load Balancing


The client proxy can select a communication process depending on the load of the communication process of an AE system (load balancing). If load balancing is activated, the client proxy searches for other communication processes from the same AE system and automatically connects to the one that has the fewest connections.

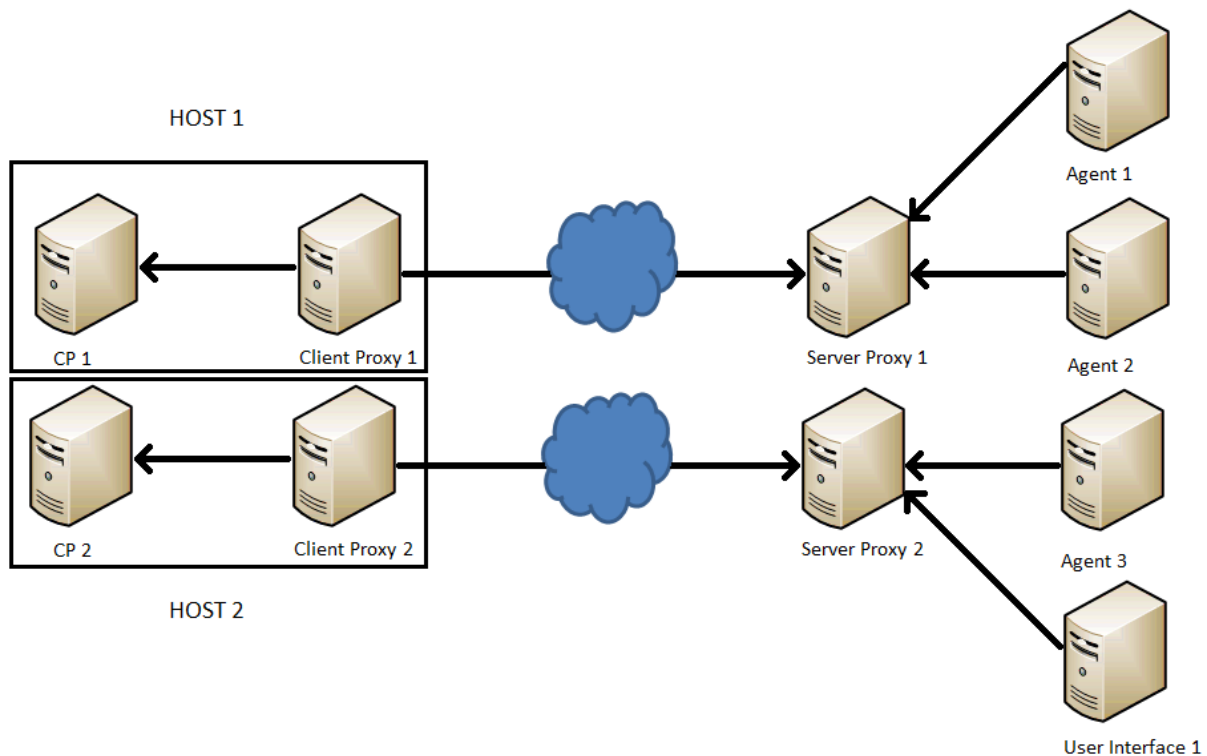
 You can activate and deactivate load balancing on the communication process side in the client proxy's [INI](#) file (parameter: `cpSelection=`).

The client proxy connects to a specific CP if load balancing is deactivated.

### 2.2.1 Inactive Load Balancing

Load balancing is deactivated in the following example. Client Proxy 1 and 2 are only connected to one single CP.

 This setting is required if you are using UserInterfaces and CallAPIs.

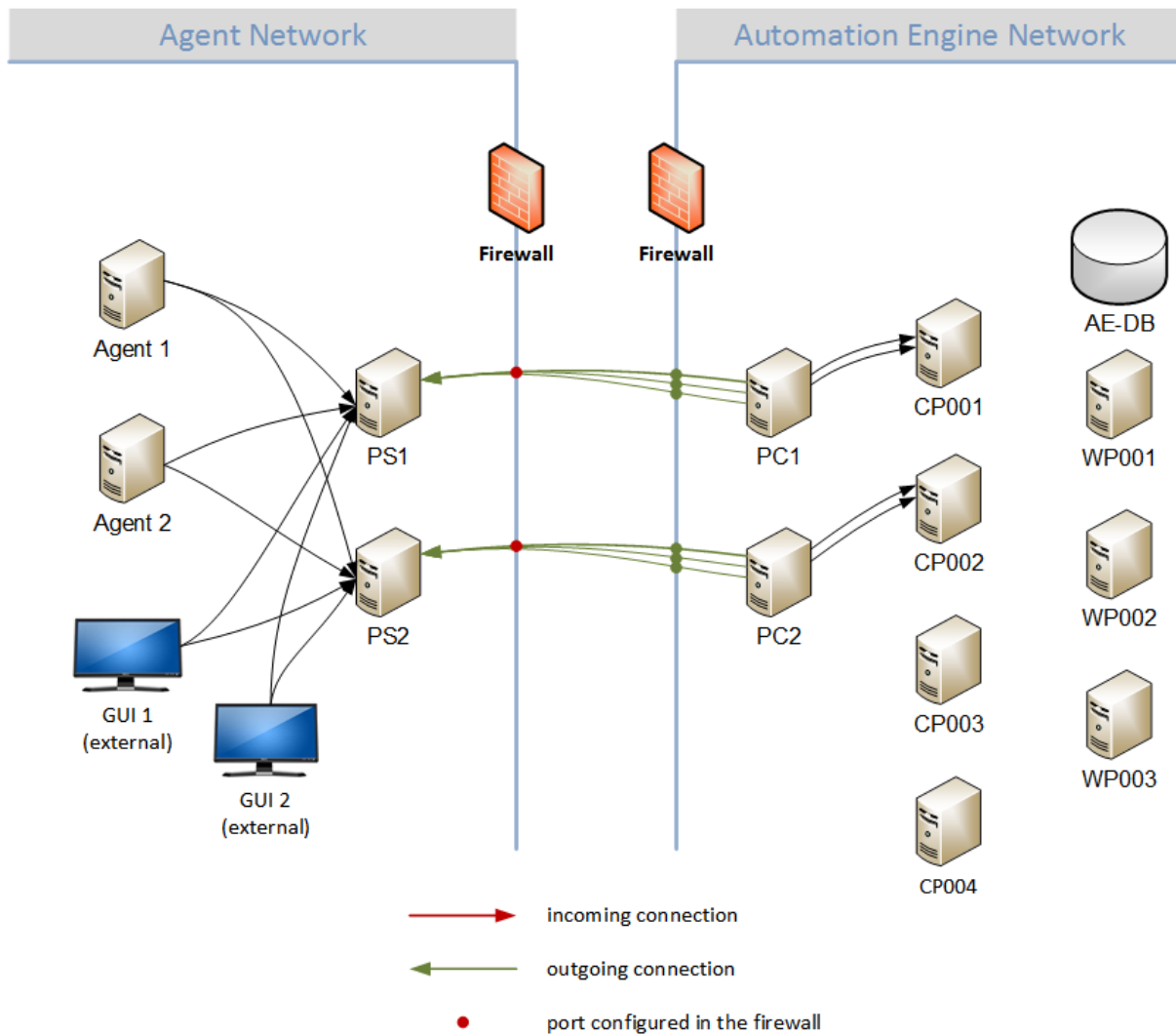


### 2.2.2 Active Load Balancing

Load Balancing is activated in the following example. When establishing the connection, the client proxies select the CP from all of the available CPs depending on the load (only CPs within the same net area are taken into account in AutomationEngine v9 SP 3 and above). Client Proxy 1 forwards connections from Agents 1 and 2 from Server Proxy 1 to CP1 and CP2. Client Proxy 2 forwards connections from Agent 3 and UserInterface 1 to CP3 and CP4.

 Load balancing may only be activated if agents are connected to the proxies.

---



## 2.3 Terminology

Keyword	Description
Proxy	The Automatic product that combines and reroutes CP connections.
Client proxy	The Proxy instance that runs on the side of the communication processes. Establishes a connection to the server proxy and forwards its connections to a CP.
Server proxy	The Proxy instance that runs on the side of the AE components, which connect to a CP. Reroutes the connections from agents, APIs, dialogs, etc. to the client proxy.
Server component	<p>The core component of the Automation Engine. Consists of multiple communication processes (CPs) and work processes (WPs).</p> <p>Also referred to as "Automation Engine" in versions 9 and 10. Known as "UC4 Server" in previous versions.</p>

Keyword	Description
Communication process (CP)	Part of the server component to which AE components (such as agents) connect.
Work process (WP)	Part of the server component that takes over processing.
Routing port	The port of the proxy server to which AE components such as agents and UserInterfaces connect.
Load balancing	The client proxy performs a load-dependent CP selection. You can activate and deactivate this in the INI file.

---



## 3 Installation

Installation of the Proxy is carried out in several steps.

 These installation instructions apply to Windows and UNIX.

### 3.1 Workflow


#### 1. Decoding the Supplied File

- Use the program [UCYBCRYPT.EXE](#) to decrypt the file proxy.jar.ucc. You will find it in the folder IMAGE:\TOOLS\ENCRYPT in the Automation Engine folder. Now call the program through the command line with the following parameters:

**UCYBCRYPT.EXE -d -f proxy.jar.ucc -l License file**

The license file is supplied together with the Automation Engine itself (*customer number.TXT*).

- The result is the file proxy.jar.

 Before attempting the decryption process, please verify that the relevant entry for PROXY is included in the license file.

#### 2. Java Runtime

Using the Proxy requires Oracle Java 1.7 (or a higher version).

Use the following command to check the Java VM version:

**java -version**

Pay attention to the correct order of the Java directories in the %PATH% (Windows) and \$PATH (UNIX) environment variable if multiple JREs or SDKs are installed. The first Java VM is always used.


You can download the latest Java version from <http://java.oracle.com>.

#### 3. Installing the Proxy

The following are required to operate two instances of the Proxy: the client proxy and the server proxy. This installation step must be performed on the computer on which the client proxy should run as well as on each computer on which the server proxy should run.


Create a dedicated directory for the proxy. For example (Windows): "C:\Automic\Proxy\bin"

Copy the delivered files to this directory.

 It is only necessary to copy the \*.ini file to the client proxy directory. The server proxy does not require an \*.ini file.

#### 4. Creating the SSL Certificate

Note that SSL encryption is required for the connections between client and server proxy. To do so, you must create a certificate. Connections between client and server proxy are only accepted if both have the same certificate.

 Note that SSL authentication is required for the connections between client and server proxy and cannot be deactivated.


Use the Java keytool in the bin folder of the Java program directory to create a self-signed certificate.

For example:

C:\Program Files\Java\jre7\bin\keytool.exe

The following command creates the keystore.jks file that saves a certificate, which is valid for 365 days. The KeyStore file is protected by the password "passwd."

```
keytool -genkey -keyalg RSA -alias selfsigned -keystore keystore.jks -  
storepass passwd -validity 365  
-keysize 2048
```

 The password for the keystore.jks may be encrypted, using the UCYBCRYPT.EXE from the Utilities in Automation Engine.

Next, answer the questions about the company and press ENTER to confirm. Example:

```
What is your first and last name?  
[Unknown]: www.automic.com  
What is the name of your organizational unit?  
[Unknown]: Development  
What is the name of your organization?  
[Unknown]: Automic  
What is the name of your city or town?  
[Unknown]: Vienna  
What is the name of your state or province?  
[Unknown]: Vienna  
What is the two-letter country code for this unit?  
[Unknown]: AT  
Is CN=www.automic.com, OU=Development, O=Automic, L=Vienna, ST=Vienna, C=AT  
correct?  
[no]: yes  
  
Enter key password for <selfsigned>  
(RETURN if same as KeyStore password):
```

**5.**

### Configuring the Client Proxy

Modify the client proxy's [\\*.ini file](#) accordingly.

**6.**

### Starting the Server Proxy

Start the Proxy instance running on the agent side, APIs, etc. (= server proxy) by using the following command line parameters:

- **servicePort=**  
Port of the proxy server to which the client proxy connects. If this parameter is not specified, the
-

default port 4321 is automatically used.

- **keyStore=**  
Path and name of the KeyStore file
- **keyStorePwd=**  
KeyStore file password

For example:

```
java -cp proxy.jar com.uc4.proxy.Server -keyStore=keystore.jks -  
keyStorePwd=passwd -servicePort=4321
```

**7.**

## Starting the Client Proxy

Start the proxy instance on the CP side (= client proxy) without entering command line parameters.

For example:

```
java -jar proxy.jar
```

The client proxy creates a log file immediately after starting and determines whether it is possible to establish a connection to a communication process. The client proxy connects to the server proxy upon successful connection to the CP.




Use the Automic ServiceManager to start the client proxy and the server proxy as a service.


---

## 4 Configuring the Client Proxy






Structure and possible settings for the proxy.ini file of the Proxy.


 Mandatory parameters are highlighted in **red**.



 The proxy.ini configuration file is only necessary for the client proxy. The client proxy connects to the server proxy (whose port is forwarded via command lines), whereby it is assigned the necessary settings. The initial connection is referred to as the service connection, over which the client and server proxy send signals.



 Note: When using two agents (during file transfers), the proxy cannot be used between these two agents.

### 4.1 Structure

Section/Parameter	Description
[GLOBAL]	
<b>cpSelection=</b>	<p>Load balancing.</p> <p>The following values are allowed: "Y," "1," "y," or "N," "0," "n" (default value)</p> <p>Set to "Y" (or "y," "1") and the Proxy performs a load balancing process. This means that the Proxy automatically selects the communication process that has the fewest connections. If load balancing is activated, the <b>cpName=</b> setting must be set to a value that is unique for each proxy instance.</p> <p> The [CP_LIST] section is automatically extended using the proxy application if additional communication processes are found.</p> <p>Set to "N" (or "n," "0") to deactivate load balancing. This way, a load-dependent CP selection is not performed and the client proxy connects to the communication process that was defined in the [CP_LIST] section.</p> <p> For cpSelection=N, one CP entry to which the client proxy is connected needs to be defined in the [CP_LIST] section.</p> <p> cpSelection=Y may only be used if only agents should connect to the proxy! UserInterfaces, CallAPIs, and Java-APIs require the "N" setting (or "n," "0").</p>
<b>cpName=</b>	<p>The name of a proxy pair that is unique within a netArea (Automation Engine v9 SP 3 and above) or the entire AE system (v6, v8, or v9, SP 1 and 2).</p> <p> This setting may only be set if the cpSelection parameter has been set to "Y" (or "y," "1").</p> <p> The name may comprise up to 32 characters.</p>

Section/Parameter	Description
<b>serverProxy=</b>	<p>The address of the server proxy to which the client proxy should connect.</p> <p>Permissible format: &lt;IP address or host name&gt;:&lt;Port&gt;</p>
<b>routingPort=</b>	<p>The number of the server proxy's routing port.</p> <p>Default value: 2217</p> <p>The server proxy monitors the routing port for incoming connections from AE components (agents, UserInterfaces, etc.).</p> <p> <b>Note:</b> The routing port and the service port may not be identical.</p>
<b>checkMemoryConsumption=</b>	<p>The period of time in seconds between memory consumption checks.</p> <p>Each check calculates the current Java memory consumption and writes it to the log file. If memory consumption is over 90 percent, the server proxy no longer accepts connections until consumption drops below 80 percent.</p> <p>Default value: 30 seconds</p>
<b>language=</b>	<p>The language of the log file items.</p> <p>The following values are permitted: "D" (German), "E" (English, default), or "F" (French)</p>
<b>logging=</b>	<p>The path and name of the client proxy's log file.</p> <p>The hash symbols (##) can be used as a placeholder for the increasing sequential numbering of the log files. The number "00" represents the most up-to-date log file. It may contain as many log files as specified in the logCount= setting.</p>
<b>serverLogging=</b>	<p>The path and name of the server proxy's log file.</p> <p>If not specified, the client proxy's log file is used (see the description of logging=). The ## placeholder may also be used for serverLogging=.</p>
<b>logCount=</b>	<p>The number of log files that should be saved.</p> <p>Default value: 1</p>
<b>LogMaxDays=</b>	<p>The number of days after which the log file should be automatically switched.</p> <p>Default value: 1</p>
<b>LogMaxSize=</b>	<p>The maximum size of the log file. A new log file is created if the log file reaches this size.</p> <p>Default value: 10000 KB</p>
<b>[TCP/IP]</b>	
<b>bindaddr=</b>	<p>This parameter defines the IP address which the client proxy uses to connect to the server proxy. You may enter the port number of your choice. bindaddr= doesn't have a default value.</p>

Section/Parameter	Description
cpResponseTime=	<p>The maximum length of time in seconds that is waited to see whether a CP responds to a connection attempt. If the CP does not send any packets during this time, it is assumed that it is not accessible. The proxy no longer accepts new connections if the CP is not accessible.</p> <p>Default value: 10 seconds</p>
connect=	<p>The length of time in seconds waited after an unsuccessful connection attempt to the CP or server proxy. If the connection cannot be established, the next attempt occurs after this period.</p> <p>Default value: 60 seconds</p>
keepAliveTime=	<p>The period of time in seconds in which packets are sent between the server and client proxy to maintain the service connection.</p> <p>Default value: 60 seconds</p>
tcp_nodelay=	<p>Set to "N" (or "n," "0") to use the Nagle algorithm for proxy connections, if it is supported by the operating system.</p> <p>The following values are permitted: "Y," "1," "y," (default value) or "N," "0," "n"</p>
soKeepAlive=	<p>Set to "Y" (or "y," "1") to activate the sending of KeepAlive packets for the proxy's TCP/IP sockets, if it is supported by the operating system. This applies to all connections.</p> <p>The following values are permitted: "Y," "1," "y," (default value) or "N," "0," "n"</p>
sendBufferSize=	<p>Size of the TCP/IP temporary storage for messages to be sent in bytes.</p> <p> The default value of the system environment is used for the value "0."</p> <p>Default value: 8192 (bytes)</p>
receiveBufferSize=	<p>Size of the TCP/IP temporary storage for messages to be received in bytes.</p> <p> The default value of the system environment is used for the value "0."</p> <p>Default value: 8192 (bytes)</p>
connectDelayMS=	<p>Period of time in milliseconds that should be waited between setting up multiple simultaneous connections to the CP.</p> <p>Default value: 10 milliseconds</p>
maxConnectionCount=	<p>Maximum number of connections the proxy will accept.</p> <p>If this number is reached, additional connections are only accepted when connections are removed again.</p> <p>Default value: 2000</p>

Section/Parameter	Description
traceLevel=	<p>Activates additional debug outputs in the log file.</p> <p> A separate trace file is not created!</p> <p>Possible values:</p> <ul style="list-style-type: none"> <li>• "0" (default value) = No additional outputs</li> <li>• "1" = Source, destination, and size are logged when a packet is sent or received</li> <li>• "2" = The hex dump of the packet content is logged when a packet is sent or received.</li> <li>• "3" = The relevant socket, option (for example, soKeepAlive=), and value are logged when modifying the socket options.</li> <li>• "4" = If an SSL action takes place, the action, resulting status, and the number of consumed/generated bytes are logged.</li> </ul>
<b>[SSL]</b>	
keyStore=	<p>Path to the KeyStore file, which contains a certificate that was created using the Java keytool.</p> <p>This parameter ensures that the connections between the server proxy and the client proxy are encrypted with SSL and authenticated. The client and server proxy require the same KeyStore file.</p>
keyStorePwd=	<p>KeyStore file password</p> <p>The password is set when creating the KeyStore file.</p>
<b>[OTHER_SP_LIST]</b>	<p>Addresses of all server proxies between which the agents (or UserInterfaces or CallAPIs) may choose for load balancing and failover purposes.</p> <p>Valid format: spx = &lt;Server name or IP address&gt;:&lt;Routing port&gt;</p> <p>Example:</p> <p>sp1 = HOST01:8809 sp2 = HOST02:8810</p>
<b>[CP_LIST]</b>	<p>List of communication processes (CPs) of the server component to which the client proxy should connect.</p> <p>Valid format: cpx = &lt;Server name or IP address&gt;:&lt;Port&gt;</p> <p>Example:</p> <p>c1 = HOST01:2217</p> <p> The CP list may only have one entry if the cpSelection= parameter is deactivated (value "N," "n," or "0").</p>

## 4.2 Example

Illustration 1: cpSelection=0:

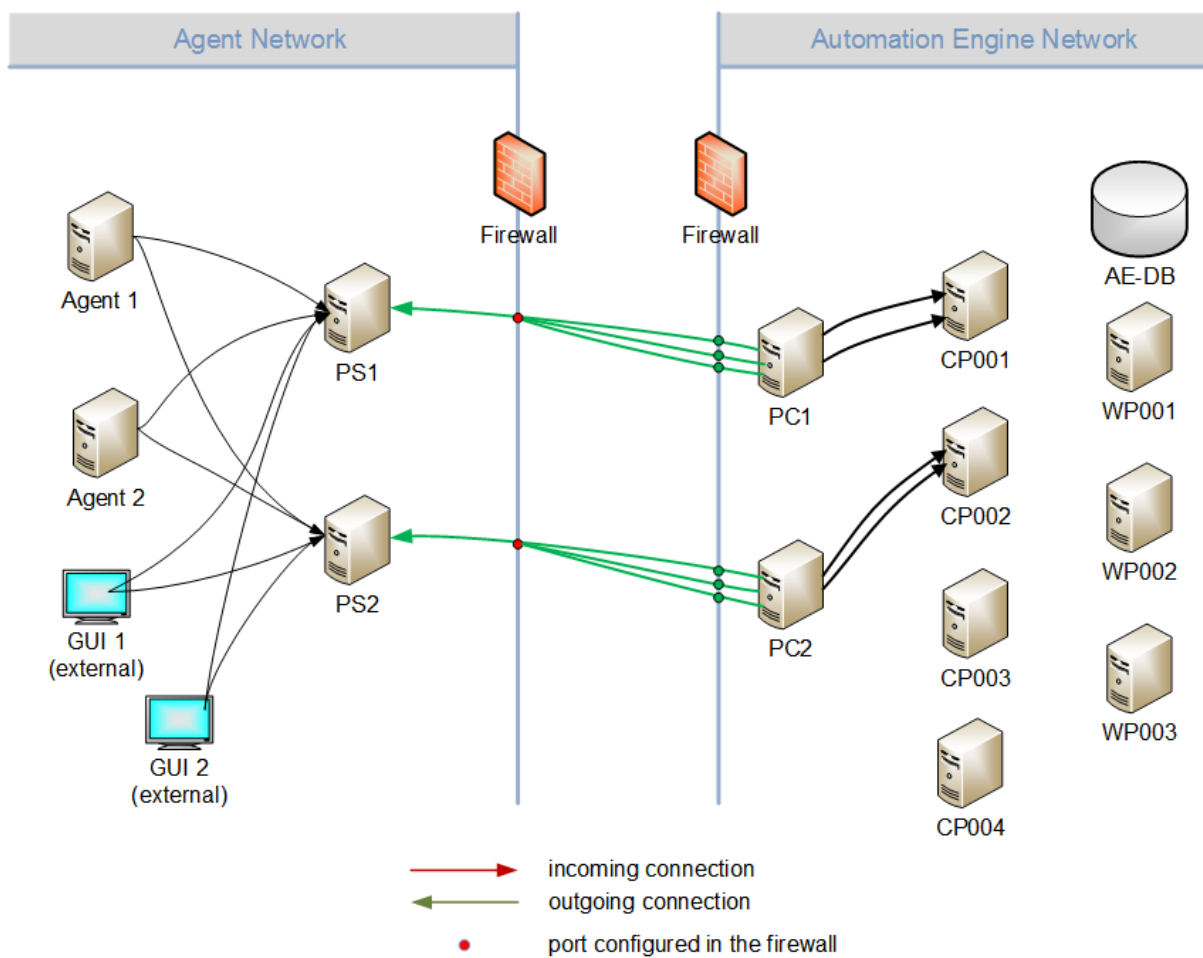
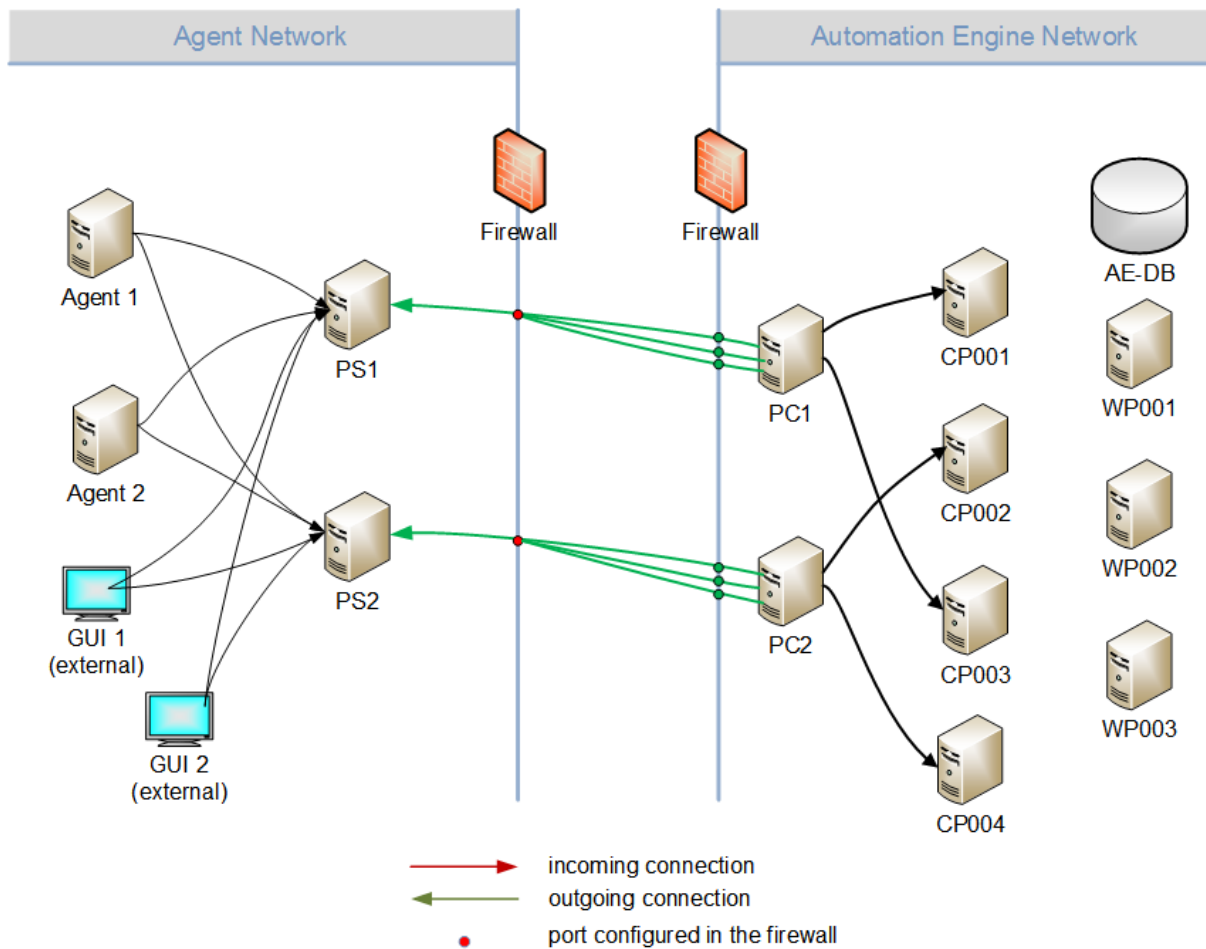
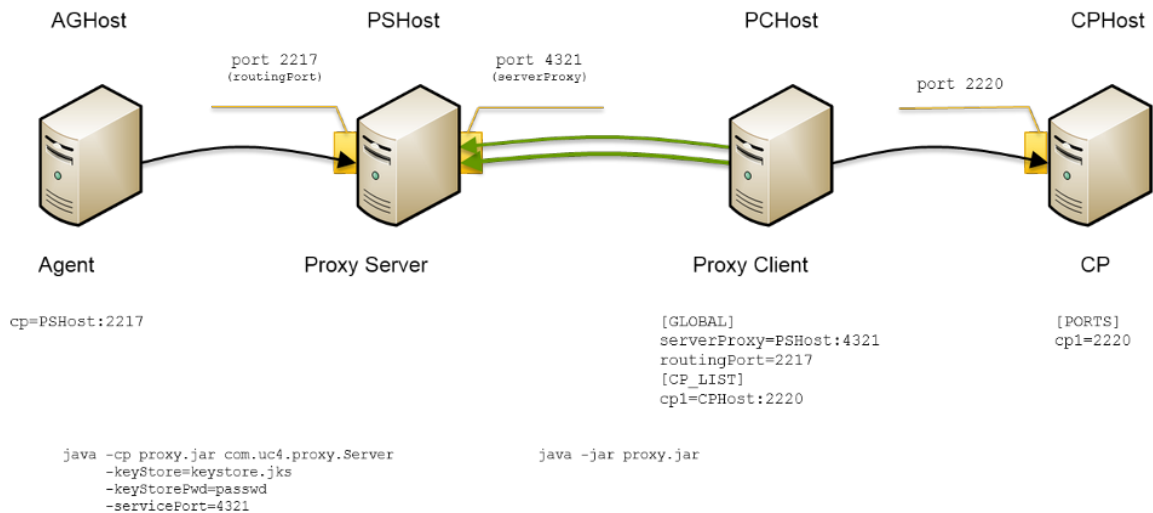


Illustration 2: cpSelection=1:





**Illustration 3: Port configuration:**



```
[GLOBAL]
cpSelection=1
cpName=PROXY_01
serverProxy=HOSTSP01:4321
routingPort=2217
checkMemoryConsumption=10
language=(D,E)
logging=../TEMP/PROXY_LOG_##.TXT
serverLogging=../TEMP/SPROXY_LOG_##.TXT
```

```
logCount=10
logMaxDays=1
logMaxSize=9000K

[TCP/IP]
bindaddr=192.168.20.21
cpResponseTime=10
connect=60
keepAliveTime=60
tcp_nodelay=1
soKeepAlive=1
sendBufferSize=8192
receiveBufferSize=8192
connectDelayMS=10
maxConnectionCount=20000
traceLevel=0

[SSL]
keyStore=keystore.jks
keyStorePwd=passwd

[OTHER_SP_LIST]
sp1=HOSTSP02:2217
sp2=HOSTSP03:2217

[CP_LIST]
cp1=HOSTSRV01:2216
cp2=10.0.0.100:2217
cp3=10.0.0.100:2218
```

---

# 5 Release Notes - PROXY Version 1

The following is the change history of the new functions and improvements we implemented, as well as bug fixes or known errors of the PROXY.

## Automatic Release Policy

### Keep your systems up to date

Automatic recommends that you always install the latest Service Pack or Hotfix. Both contain valuable corrections and bug fixes between Major and Minor Releases, where new features and enhancements are introduced.

### Automatic release policy

- **Major Release:** This is the main version of a software release. It is identified by the first segment of the entire version number (such as the 10 in **10.0.0**).
- **Minor Release:** This includes new features, modifications and corrections that may contain major changes such as database modifications. It is identified by the second segment of the entire version number (such as the 2 in **11.2.0**).  
Major and Minor releases for Automation Engine are feature releases supplied at 9-month intervals.
- **Service Pack:** This is a patch for a release and contains corrections for errors. New features or modifications are not included. Service packs are identified by the third segment of the entire version number (such as the 2 in **10.0.2**).
- **Hotfix:** This is a minor sub-release to remove malfunctions and defects. Hotfixes are indicated by an HF number after the version number (such as the 1 in **10.0.2 HF 1**).  
Service Packs and Hotfixes are maintenance releases supplied at 4-month intervals.

## New Functions

### General

#### New Password Encryption for keystore.jks

As of version 1.0.1 it is possible to encrypt the password for the keystore.jks by using the encryption tool of the Automation Engine.

#### New Parameter for Setting Particular IP Address

The parameter `bindaddr=` defines the IP address which the client proxy uses to connect to the server proxy. You may enter a port number of your choice. `bindaddr=` doesn't have a default value.

## Improvements

### General

### **Port Configuration Extended**

The possibilities to configure ports have been extended. You may now define either the port itself or the IP address and port.

### **Default Value Changed for maxConnectionCount in Configuration**

The default value for the parameter maxConnectionCount was changed to 2000.

---

## **Documentation**

### **Installation Process Description Updated**

The description of the installation process has been updated with new illustrations and a few new additions to the text itself to improve clarity.

---

## **Bug Fixes**

For detailed and up-to-date information, see the *Patch Descriptions* section in the [Automatic Download Center](#).

---

## **Known Issues**

For detailed and up-to-date information, see the *Known Bugs & Workarounds* section in the [Automatic Download Center](#).

---

## 6 Technical Support

---

We have a support team you can trust. Our team of professionals is ready to support you, anytime and anywhere. Three support centers located in Europe, the United States, and Asia Pacific build the core of the Automic support organization.

---

### Support Resources: Download or Browse

The [Download Center](http://downloads.automic.com/de/services/download-center.html) (<http://downloads.automic.com/de/services/download-center.html>) is the place where you find everything you need to know about your Automic solution to make sure you are using our products to their fullest potential. It's all in one place: from service hotfixes, release notes, to guides - for all our products. You will also find patch descriptions, known bugs or workarounds - and everything to download, for you to choose from by version, product or release.

Additionally you will find our documentation at [docs.automic.com](https://docs.automic.com). For all our products the latest released version of documentation is up, structured by type, version and product. This way you may browse through the respective categories at leisure - or leave your comments there.

We look forward to hearing from you!

### Technical Support Team

Our [Technical Support Team](mailto:support@automic.com) ([support@automic.com](mailto:support@automic.com)) makes sure that your closest Automic experts are never more than a few hours flight away, no matter on which continent your subsidiaries and data centers are located. Our products are designed to provide global connectivity for international companies. You are employing them on a global scale and therefore you can expect global services.

---