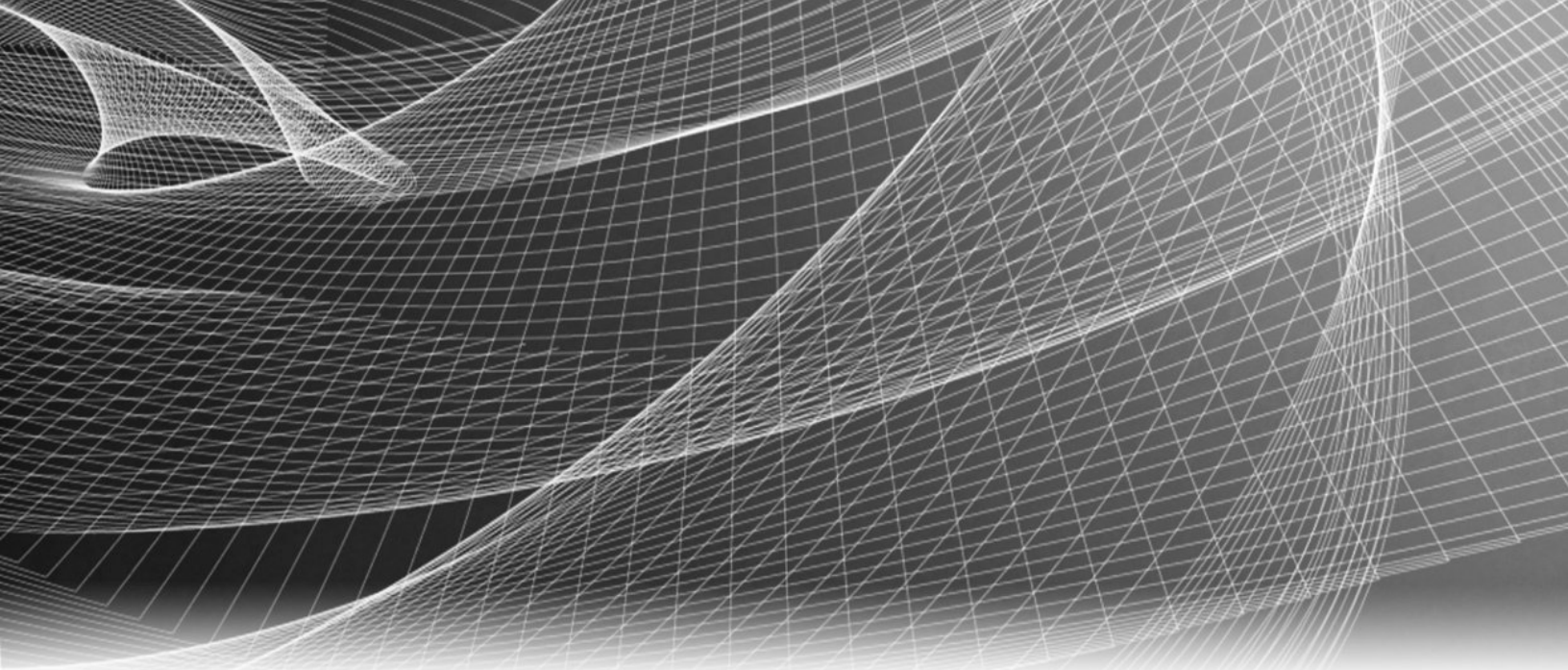




EMC® Watch4net®

Version 6.4u2

Webservice Gateway Admin Guide

P/N: pending

Rev : 40032

Copyright ©2011 - 2014 EMC Corporation. All rights reserved. Published in the USA.

Published October, 2014

EMC believes the information in this publication is accurate as of its publication date. The information is subject to change without notice.

The information in this publication is provided as is. EMC Corporation makes no representations or warranties of any kind with respect to the information in this publication, and specifically disclaims implied warranties of merchantability or fitness for a particular purpose. Use, copying, and distribution of any EMC software described in this publication requires an applicable software license.

EMC², EMC, and the EMC logo are registered trademarks or trademarks of EMC Corporation in the United States and other countries. All other trademarks used herein are the property of their respective owners.

For the most up-to-date regulatory document for your product line, go to EMC Online Support (<https://support.emc.com>).

Contents

Preface	iii
Purpose	iii
Intended Audience	iii
Prerequisites	iii
Document Organization	iii
Common Abbreviations and Acronyms	iii
Where to get help	v
1 Overview	1
2 Configuration	3
1 SSL Settings	4

Preface

Purpose

This guide gives a quick overview of the *Webservice Gateway* and how it is configured. Even if it is mostly intended to work out-of-the-box, there are a few configuration changes that an administrator would like to check.

Intended Audience

This guide is addressed to the *APG* Administrator.

Prerequisites

This guide assumes that you have an *APG* 6.0 installation ready with the `webservice-gateway` module installed.

Document Organization

1. Overview	This is a high level overview of the <i>Webservice Gateway</i> and its usage.
2. Configuration	This sections explains how to change the <i>Webservice Gateway</i> settings.

Common Abbreviations and Acronyms

WS

This is an acronym for *webservice*.

Where to get help

EMC support, product, and licensing information can be obtained as follows:

PRODUCT INFORMATION - For documentation, release notes, software updates, or information about EMC products, go to EMC Online Support at:

<https://support.emc.com>

TECHNICAL SUPPORT - Go to EMC Online Support and click Service Center. You will see several options for contacting EMC Technical Support. Note that to open a service request, you must have a valid support agreement. Contact your EMC sales representative for details about obtaining a valid support agreement or with questions about your account.

Overview

The *Webservice Gateway* is the entry point for all the *APG* management webservices. It allows remote clients to manage any manageable module of an *APG*.

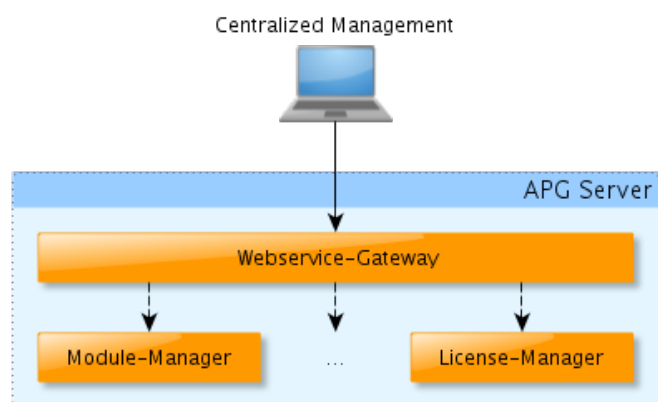


Figure 1: Overview

The *Webservice Gateway* typically receives requests from a centralized management console and forwards it to the module able to handle it.

For example, to remotely install a license, the *Webservice Gateway* would redirect the client request to the *License-Manager*. More precisely, it forwards client requests to <https://apg.server:48443/Tools/License-Manager/Default/> to the default instance of the license manager.

2

Configuration

The *Webservice Gateway* is configured to be the only entry point of the whole *APG* server. As soon as it is installed, it allows to remotely manage any module that has a management webservice.

Out-of-the-box, the *Webservice Gateway* is ready to use at <https://apg.server:48443> using the default user (admin:changeme). The user list can be configured in `conf/users`.

Most of the *Webservice Gateway* configuration is done through properties in `conf/modules.properties`. This is typically where an administrator would change the protocol (http/https), the listening address, the listening port, the SSL parameters (see section 1), and a few other advanced settings.

```
#=====
# Gateway settings
#=====
# the webservice url (0.0.0.0 binds on all interfaces)
com.watch4net.webservice.server.url=https://0.0.0.0:48443/
# the service heartbeat (a service that doesn't respond for one minute is considered
dead)
com.watch4net.webservice.port.threshold=60000

#=====
# Authentication settings
#=====
# the webservice authentication mode, only BASIC is supported
com.watch4net.webservice.auth.factory=com.watch4net.apg.webservice.proxy.auth.
BasicAuthenticatorFactory
# the user database for the BASIC authentication
com.watch4net.webservice.auth.basic.users=conf/users

#=====
# SSL settings
#=====
# the keystore location
com.watch4net.webservice.server.ssl.keystore.file=conf/keystore
# the keystore password
com.watch4net.webservice.server.ssl.keystore.pass=changeit
```

1 SSL Settings

Here is the detailed list of supported SSL settings when the *Webservice Gateway* is configured to use HTTPS protocol:

algorithm	The certificate encoding algorithm to be used. This defaults to SunX509.
keystore.file	The pathname of the keystore file where you have stored the server certificate to be loaded. By default, the pathname is the file <code>.keystore</code> in the operating system home directory of the user that is running the <i>Webservice Gateway</i> . If your <code>keystore.type</code> doesn't need a file use (<i>empty string</i>) for this parameter.
keystore.pass	The password used to access the server certificate from the specified keystore file. The default value is <code>changeit</code> .
keystore.type	The type of keystore file to be used for the server certificate. If not specified, the default value is <code>JKS</code> .
keystore.provider	The name of the keystore provider to be used for the server certificate. If not specified, the list of registered providers is traversed in preference order and the first provider that supports the <code>keystore.type</code> is used.
protocol	The version of the SSL protocol to use. If not specified, the default is <code>TLS</code> .
key.alias	The alias used to for the server certificate in the key-store. If not specified the first key read in the key-store will be used.
key.pass	The password used to access the server certificate from the specified keystore file. The default value is <code>changeit</code> .
trustmanager.class	The name of a custom trust manager class to use to validate certificates. The class must have a zero argument constructor and must also implement <code>javax.net.ssl.X509TrustManager</code> . If this attribute is set, the trust store attributes may be ignored.
trustmanager.maxcertlen	The maximum number of intermediate certificates that will be allowed when validating certificates. If not specified, the default value of 5 will be used.
truststore.file	The trust store file to use to validate certificates. The default is the value of the <code>javax.net.ssl.trustStore</code> system property. If neither this attribute nor the default system property is set, no trust store will be configured.

<code>truststore.pass</code>	The password to access the trust store. The default is the value of the <code>javax.net.ssl.trustStorePassword</code> system property. If that property is null, the value of <code>keystorePass</code> is used as the default. If an invalid trust store password is specified, a warning will be logged and an attempt will be made to access the trust store without a password which will skip validation of the trust store contents. If the trust store password is defined as (<i>empty string</i>) then no password will be used to access the store which will also skip validation of the trust store contents.
<code>truststore.type</code>	The type of key store used for the trust store. The default is the value of the <code>javax.net.ssl.trustStoreType</code> system property. If that property is null, the value of <code>keystore.type</code> is used as the default.
<code>truststore.provider</code>	The name of the truststore provider to be used for the server certificate. The default is the value of the <code>javax.net.ssl.trustStoreProvider</code> system property. If that property is null, the value of <code>keystore.provider</code> is used as the default. If neither this attribute, the default system property nor <code>keystore.provider</code> is set, the list of registered providers is traversed in preference order and the first provider that supports the <code>truststore.type</code> is used.
<code>truststore.algorithm</code>	The algorithm to use for truststore. If not specified, the default value from the JVM is used.
<code>session.cachesize</code>	The number of SSL sessions to maintain in the session cache. Use 0 to specify an unlimited cache size. If not specified, a default of 0 is used.
<code>session.timeout</code>	The time, in seconds, after the creation of an SSL session that it will timeout. Use 0 to specify an unlimited timeout. If not specified, a default of 86400 (24 hours) is used.
<code>crlfile</code>	The certificate revocation list file to use to validate certificates.
<code>ciphers</code>	The comma separated list of encryption ciphers to support for HTTPS connections. If specified, only the ciphers that are listed and supported by the SSL implementation will be used. By default, the default ciphers for the JVM will be used.

sslEnabledProtocols	The comma separated list of SSL protocols to support for HTTPS connections. If specified, only the protocols that are listed and supported by the SSL implementation will be enabled. If not specified, the JVM default is used.
---------------------	--