



## The Cloud Alert Messaging Gateway

### **User Guide**

Document Version 2.2  
Copyright (C) 2019 Cloud Alert Ventures W.L.L. All Rights Reserved.  
Java, Oracle and the Oracle Database are trademarks/copyrights of Oracle Corporation.  
Microsoft Windows is a trademark of Microsoft Corp.  
Google Chrome is a trademark of Google LLC.  
Mozilla Firefox is a trademark of Mozilla Corporation.  
<http://cloudalert.cloud>

## Table of Contents

Introduction.....	3
Concepts.....	3
Internal Components.....	5
The Administration Interface.....	6
Details of the Administration Interface.....	7
Integrating with your Applications and Databases.....	10
APIs.....	10
SMPP Server.....	11
Plugins.....	12
Adding a database driver.....	12
Compiling Plugins.....	12
File and Folder Structure.....	14
Log files.....	14
Windows Service.....	16
Installation and uninstallation.....	16
Migrating an installation to another computer.....	16
Additional information and Support.....	17

## Introduction

---

Cloud Alert is a comprehensive solution for the alert and notification requirements of financial institutions and other enterprises.

One of the components of the Cloud Alert Platform is the Messaging Gateway - an on-site deployed system that integrates with the organization's databases to send alerts and OTP messages to handsets. This document explains its operation and administration.

If you do not have a copy of the gateway, Starter editions of the software are available for free at:

<http://cloudalert.cloud>

## Concepts

---

Before using the Messaging Gateway, certain concepts must be understood:

- Networks
- Connections
- Failover
- Message Status

### Networks

In the Messaging Gateway's terminology, a 'Network' refers to a service which provides delivery of messages, regardless of whether its GSM, CDMA, OTT or others. Examples are: Vodafone's GSM Network, Cloud Alert, GMail, etc.

### Connections

A 'Connection' refers to a single socket/http connection to a server of the network. Multiple connections can be defined within a network, but they must all match the type of the network. Examples are: SMPP connection to Vodafone's GSM Network; Cloud Alert Connection to Cloud Alert's Network; SMTP Connection to GMail's Network.

### Failover

The Messaging Gateway is capable of failover, i.e., a message is tried through one network, and if it fails to deliver, it is then tried on another network. The failure to delivery can be instantaneous, say, a rejection of the message by the destination server, or it can be on lack of a delivery report from the network within a defined period of time.

### Message Status

Most admins are familiar with the existing messages statuses of 'In Queue', 'Sent', 'Delivered', etc. With failover, an additional set of statuses have been added.

The list of all possible statuses are now:

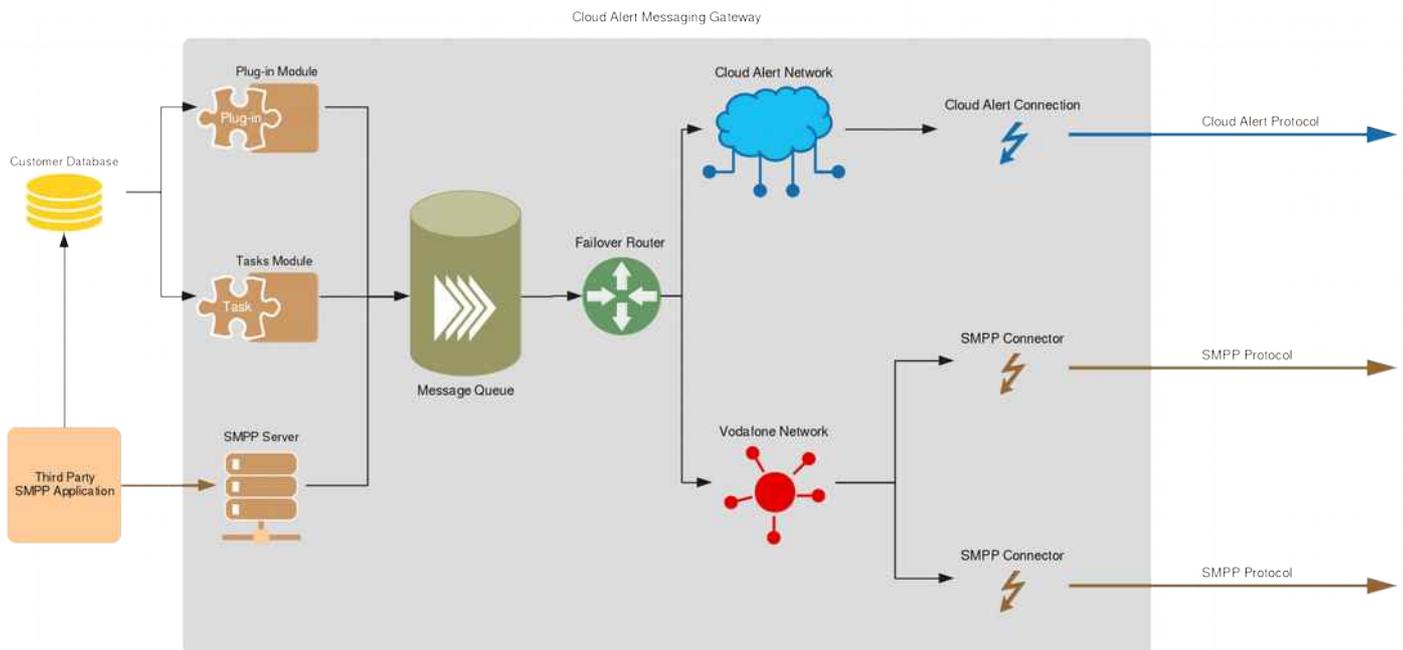
Status	Description
Awaiting Routing	The message has been posted to the queue, but has not been routed yet.
In Queue	The message has been routed, and is in the queue for that network.
Sent	The message has been sent on that network, ie, the server has accepted it. Its delivery or failure has not been confirmed.
Failed	The message has been rejected by the network. The failover may reroute this message to another network, or fail it completely. If rerouted, its status will change back to 'In Queue'. If no more failover networks are present, its status will change to 'Totally failed'.
Undeliverable	The message was initially accepted by the server, but the server later responded that it cannot be delivered through it. The failover may reroute this message to another network, or fail it completely. If rerouted, its status will change back to 'In Queue'. If no more failover networks are present, its status will change to 'Totally failed'.
Totally Delivered	The message delivery has been confirmed by at least one of the networks. The Messaging Gateway will no longer try to send this message.
Totally Failed	The message has been tried through all defined networks, but cannot be sent through any of them. The Messaging Gateway will no longer try to send this message.

Messages may be in certain statuses for very short periods, hence never visible in that status.

## Internal Components

The Messaging Gateway comprises of the following components:

- Message Queue
- Plugin Module
- SMPP Server
- REST API
- Cloud Alert Connection
- SMPP Connections



### Message Queue

The message queue holds messages within the Messaging Gateway until they reach a final delivery or failure status. The queue has been implemented as a table within the Oracle Database supplied with the gateway.

The queue receives messages from Plugins, Tasks, the integrated SMPP Server and the APIs. For test purposes, the 'Send a Test Message' page in the console also allows you to post messages into the queue.

### Plugin Module

Plugins are Java classes that are loaded by the Plugin Module. These classes can be coded by the vendor or admins to any requirement; typically, extracting records from databases and posting messages into the Gateway's queue.

### SMPP Server

Should your organization already possess an existing SMPP client application, it can be integrated with the Messaging Gateway's built-in SMPP Server by altering the application's settings to point to the Messaging Gateway. This helps you reuse the integration that your SMPP client application already has with your

databases.

## APIs

Applications can integrate with the Gateway using the XML and REST APIs present on its built-in web server.

## Cloud Alert Connection

The Messaging Gateway is typically configured with a Cloud Alert connection to send messages. The failover router is configured to try all messages through this connection first, and on failure to retry through the other network.

## SMPP Connections

The Messaging Gateway can be connected to your organization's existing SMPP service provider to send SMS messages to handsets which do not have the Cloud Alert app installed.

## The Administration Interface

---

The administration of the system can be carried out from any web browser. The system includes an integrated web server for this purpose.

### Launching the administration interface

The administration interface can be launched from the same machine where the gateway is installed, by using the following URL:

```
http://localhost:2199
```

Alternatively, the ip-address may be used:

```
http://127.0.0.1:2199
```

The administration can also be carried out from any other machine on the network. The URL in this case will use the hostname or IP-address of the machine where the gateway has been installed.

### Logging into the administration interface

The console allows passwordless logins from the same machine.

If a login page has been displayed in your browser, enter the password to login. The default administration password is **'password'**. The password is not case-sensitive.

## Details of the Administration Interface

### The home page

On successfully logging on, the home page is displayed, which contains links to the features of the interface.

In addition, brief information is present on the right bar of the home page.

### View Logs

This section helps you view the current activity of the system.

The log information is displayed on the page, and is automatically refreshed every second.

The auto-refresh can be enabled and disabled by the links at the bottom of the page.

This visual log is limited to 50 entries to conserve system memory. For older logs, or logs beyond a system restart, you will need to refer to the log files.

Tip: Turn off the refresh if you wish to scroll upwards in the logs.

### View Connected SMSCs

This displays the outbound connections from the gateway. All defined connections are displayed, even if they are currently not connected. The color of the connection indicates the status of the connection.

The outbound and inbound packet speeds are displayed per connection, alongwith other brief statistics.

### View Connected Applications

This displays the inbound SMPP connections to the gateway from your third-party SMS applications. It displays only the connections that are currently connected.

### Configure Plugins

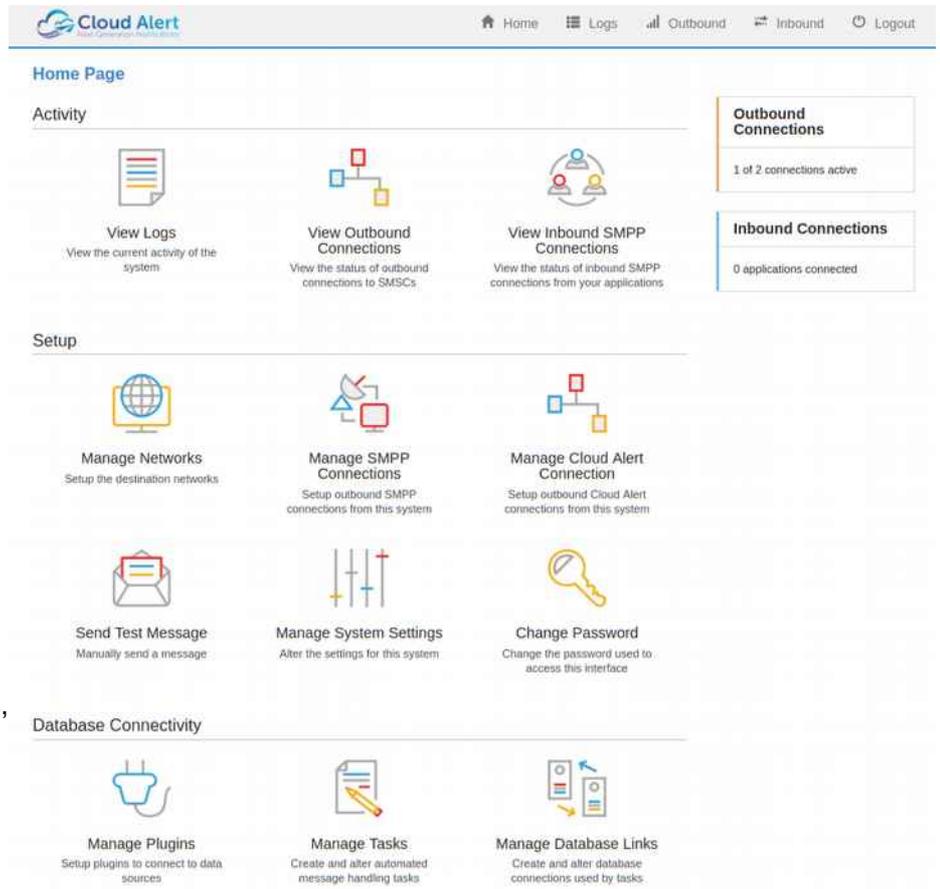
This section helps you configure the plugins of the system. The system may have been provided pre-configured to you with ready plugins.

Each plugin setting includes the following information:

**Name:** Enter any name descriptive of the plug, for your reference, for example 'Card Payment Reminder' .

**Plugin Class Name:** This is the class name of the plugin. The class name is usually the name of the file of the plugin, not including any .class extension. If the class is part of a package, a fully-qualified name will have to be entered. (for example, 'com.bank.AccountsSMSPlugin')

Caution: This field is case-sensitive on some operating systems.



**Param 1-10:** These are parameters to the plugin. These can be used to provide any additional information required by the plugin. Additional details regarding the parameters will be provided by the developers of each plugin.

On altering the settings of any plugin, you need to restart the gateway. The gateway will prompt you - click 'Restart Now' to restart.

## Send Test Message

This page helps you send a message to test the connectivity of the gateway to the servers.

**Phone number:** The phone number should be in the international format, with country/network codes, but without any leading 0s or '+' signs.

**Sender ID:** The sender id field is used only if the message is sent over SMS. For Cloud Alert, the name displayed on the handset is the name provided during the signup at Cloud Alert.

**Routing Options:** This option can be used to override the default setting, so that a particular connection can be tested.

**Sending options:** This option allows you to schedule your outbound message.

**Content:** The content can be larger than 160 characters. If SMS, the system uses concatenation to send the whole message.

The page displays a confirmation of the posting of the message, with the Message ID of the record in the queue.

## Manage Networks

This section helps you define the networks through which messages can be sent.

The fields included with each network are:

**Network Name:** Any name you wish to provide to this network. This information is not sent over the network. It will be present in the logs to help you identify the network.

**Network ID:** Any 3-character code you wish for identifying this network. This code cannot be changed later, and will be used in other pages and logs to refer to this network.

**Network Type:** The protocol of this network. This cannot be changed once defined.

**Comment:** Any comments you wish to enter, for your reference only. This information is not sent over the network.

The other settings affect the performance of the messaging engine, and can be left at their default values.

## Manage Outbound SMPP Connections

This page helps you define outbound SMPP connections for your SMS messages.

The fields included with each connection are:

**Connection Name:** Any name you wish to provide to this connection. This information is not sent over the connection. It will be present in the logs to help you identify the connection.

**Server name:** The hostname or IP address of the SMPP server to connect to. This field may be hardcoded to Cloud Alert's SMPP server.

**Port number:** The port to connect to.

**Login name:** The username/system ID on the server.

**Login Password:** The password of your SMPP account on the server.

**Network:** The network that this connection will be associated with. Since SMPP is a telecom-specific protocol, only networks of type GSM/CDMA can be selected here.

**Node:** The node in your Messaging Gateway cluster that should open the connection.

The rest of the settings relate to the performance and stability of the connection. They can be changed if requested by your SMPP provider.

## Manage Cloud Alert Connection

This page helps you manage your Cloud Alert connection. Note that only one simultaneous connection to Cloud Alert can be configured.

If a connection is already configured, its information is displayed.

The fields are:

**Connection Name:** Any name you wish to provide to this connection. This information is not sent over the connection. It will be present in the logs to help you identify the connection.

**Network:** The network that this connection will be associated with. Since the Cloud Alert protocol is a proprietary encrypted protocol, only networks of type Cloud Alert can be selected here.

**Node:** The node in your Messaging Gateway cluster that should open the connection.

**User name:** The name of your organization. This name will be displayed on the handsets as the sender of the messages. This field cannot be changed without recreating your account on the server.

**Email address:** Your organization's email address. This address will be displayed to handsets. Its recommended that you enter a common email address, or that of your call center. This field cannot be changed without recreating your account on the server.

**Comments:** Any comments you wish to add. This information will not be sent.

**Logo:** The logo of your organization. This will be displayed to users. The logo file must be in the JPG or PNG formats, with size 512x512 pixels. This logo cannot be changed without recreating your account on the server.

**Authorization code:** A code provided by Cloud Alert that authorizes you to signup. This code will be sent to you over email.

## Manage System Settings

The important settings in this section are:

**The network code for initial send try :** The 3-character code of the network that messages are initially tried on, ie, when a new message is posted to the queue, its routed to this network. Typically the code of your Cloud Alert connection.

**The network code that failover fails from:** The failover module monitors this network for messages that havent been delivered. Typically the code of your Cloud Alert connection.

**The network code that failover fails to:** When the failover module processes a message, it assigns it to a different network. This network code is entered here. Typically the network code of your telecom provider/outbound SMPP connection.

**Reroute messages when no DLR for x seconds:** Once a message is sent over Cloud Alert, the failover module waits for the specified number of seconds to receive a success delivery report. If not received within this duration, the message is failed over to the failover network.

**Do not send if user offline more than secs:** Cloud Alert maintains information about app users who are connected to the service based on pings from the app. If the last ping from the app was longer than the specified time, the message will not be sent over Cloud Alert, and will be failed over.

## Change Password

This page allows you to change the administration password. It is recommended that you change the default password as soon as possible.

## Integrating with your Applications and Databases

---

The Messaging Gateway integrates with your database through one of the following:

- **APIs:** REST and XML APIs are present. They are detailed in the REST API Guide and the XML API Guide.
- **SMPP Server:** The SMPP server helps you integrate SMPP-based applications with the gateway.
- **Plugins:** Plugins are Java classes used to integrate with your databases through Java code. The creation of plugins is detailed in the Plugin Developers Guide.

## APIs

---

The gateway includes http-based APIs that can be invoked by your applications. For example:

```
http://192.168.1.1/RESTIntegrator_SendSMS_1?  
u=myuser&p=mypass&s=mycompany&r=true&d=441234534500&t=Hi%20There  
&21&o=Nv&c=notif
```

posts an outbound message.

Similarly, an XML-based API is present at:

```
http://192.168.1.1/RESTIntegratorXML_SendSMS_1
```

Refer to the REST API Guide and XML API Guide for more details.

## SMPP Server

---

The Gateway includes an SMPP Server, to which applications that include an SMPP client can connect to. Enter the following settings into your SMPP client application:

SMPP Server Host: <The hostname/IP of the Messaging Gateway>

SMPP Server Port: 2775

Username/System id: admin

Password: <The password of the administration console>

Source TON : 5

Destination TON:1

Source NPI:9

Destination NPI:1

Bind type: Transceiver

### Notes

- Messages submitted to the gateway will be added to the queue, and follow the same failover settings as messages posted through the plugins or APIs. The queue can be managed through the Manage Queue section in the management portal.
- The SMPP Server does not support message concatenation.
- The SMPP Server Port can be reconfigured in the gateway's Manage Settings.

## Plugins

---

Plugins are Java classes that you create which can be loaded by the gateway. The gateway invokes functions in the classes, which you code to extract data from your data sources. Thus, the plugin holds the glue code that integrates your databases and other data sources with the gateway.

Plugins are compiled into class files and placed into the `c:\cag\dbc\plugins` directory.

## Adding a database driver

---

For your plugins & tasks to connect to your databases, you may need to add additional database drivers. The Messaging Gateway already includes database drivers for Oracle and SQL Server. Note that an Oracle driver is required for the Messaging Gateway to connect to its private store.

The drivers are maintained in the `c:\cag\jdbc` directory. Each driver is a jar file, and has to be renamed to a filename from `d1.jar` to `d9.jar`. This mechanism has been so designed since the Messaging Gateway's classpath cannot be changed, thus necessitating file renames.

In the default installation, the files `d1` and `d2` are the drivers for Oracle and SQL Server respectively. The files `d3` to `d9` are dummy files - replace them with your own driver jar.

Since renaming a file may cause you to lose track of them, a comments text file has been included. Ensure that you update the comments file on any change.

Additionally, it's recommended that a copy of the driver jar files be placed in the 'drivercopy' directory, with their original names. While this directory is not used by the Messaging Gateway, it is a useful reference to assist you.

Note that the default Oracle driver is not compatible with Oracle 12c. It only supports SIDs, not service names - update to the latest jdbc driver to avail of these features. A newer driver 'ojdbc7.jar' is present in the drivercopy directory - replace `d1.jar` with it if required. Note that this may affect the connectivity between the Messaging Gateway and its private store; it's recommended to contact support before changing the `d1.jar` file or adding additional Oracle drivers.

The Messaging Gateway is compatible with class versions up to Java 8. Note that updating the Java installation on your OS does not increase this limit.

## Compiling Plugins

---

Once you have created your plugin's .java file as per the instructions in the Plugin Developers Guide, you need to compile it to create a .class that can be loaded by the gateway. The `javacplugin.bat` file in the `\cag\dbc\plugins` directory will help in the compilation process.

- Open a command prompt, and `cd` to the `\cag\dbc\plugins` directory.
- Enter '`javacplugin myjavafile.java`', replacing `myjavafile.java` with the actual name of your java file.
- The compilation process will either display errors, warnings, or nothing if the compilation was successful. Ensure that the .class file is now present.
- The name of the .class file, without its extension, can now be entered into the Administration Console - Configure Plugins - Add - Plugin Class Name. Click Save.
- The gateway must now be restarted for the plugin to be loaded. The View Logs section displays any errors during the loading of the plugin.

Note that your plugin can comprise of multiple classes. In this case, compile all of the java files into classes, but enter the name of only the main file in the Configure Plugins section. The main file is the file that extends `DBCDBPlugin` or `DBCGenericPlugin`.

If you wish to use a third-party IDE (for example, Eclipse or Idea) to create your plugins, ensure that the

\cag\dbc\pluginstubs directory is added or copied into your project. The database drivers in \cag\lib\jdbc may also need to be added.

Refer to the Plugin Developer's Guide for more details.

---

## File and Folder Structure

The recommended location to install the Messaging Gateway is the \cag directory at the root of any drive. Below are the files and folders that make up the installation. Other directories may be present for internal use.

File/Folder	Description
c:\cag\cag.exe	The main executable. This executable cannot be launched directly; it must be installed as a service.
c:\cag\conf\basic.conf	This file contains settings required for the gateway to connect to its private store.
c:\cag\dbc\plugins\	The plugins you create must be placed here.
c:\cag\dbc\pluginstubs\	Files required for the compilation of plugins.
c:\cag\scripts\	Batch files and useful sql scripts.
c:\cag\doc\	Documentation.
c:\cag\dbc\settings.db2	This file contains the plugin settings.
c:\cag\dbc\tasks.db	This file contains the task definitions.
c:\cag\dbc\ctasks\	This directory contains the class files of generated tasks.

## Log files

The system maintains logs of the following, and in the following folders:

Folder	Description
c:\cag\logs\status\	This folder contains the status-related information of the system. Includes logs displayed in the View Logs section of the administration console. Each file corresponds to a single day, and the files are stored in the text format.
c:\cag\logs\sent\	This folder contains a log of all messages sent by the system. Each file corresponds to a single day, and the files are stored in the CSV format.
c:\cag\logs\dlr\	This folder contains a log of all delivery reports received from SMSCs. Each file corresponds to a single day, and the files are stored in the CSV format.
c:\cag\logs\received\	This folder contains a log of all inbound messages received from SMSCs. Each file corresponds to a single day, and the files are stored in the CSV format.
c:\cag\logs\out\	The log of general operations of the system.
c:\cag\logs\debug\	Debugging logs.

## Log File Formats

**CSV:** This format, also known as comma-separated-values, is a tabular data format. Files of this type can be opened in Microsoft Excel or OpenOffice.

**Text:** Files of this format can be opened in any text editor, including Notepad, Wordpad, or Microsoft Word.

## Logging details

Each logging file is suffixed with the date that it pertains to, with the date in reverse format to aid in sorting.

The system holds open only the file of the current day. Past files are closed, and can be moved, edited or deleted by other applications while the system is in operation.

If the system is unable to open the log file of the current day for writing, for example, if it is currently held open by Microsoft Excel, then the system will generate a new file with a numeric suffix ( For example, if file sent160318.csv could not be opened, the system will create a new file sent160318\_1.csv, which it will write to. This is to ensure that logs are not lost.

The switching from one log file to the next takes place at midnight. The operation may be delayed by upto 10 seconds. For accurate day-wise accounting of messages sent around midnight, it is recommended that the user also refer to the last file of the earlier day.

## Windows Service

---

The gateway installs as a Windows Service. Its Windows options can be managed by selecting Start - Control Panel - Administrative Tools - Services, and then searching for 'Cloud Alert Messaging Gateway'.

The gateway can be started and stopped using the icons on the top of this window.

## Installation and uninstallation

---

The service is automatically installed as part of the initial setup. If you wish to manually install or uninstall the service, the following steps can be carried out.

Navigate to Start - All Programs - Accessories - Command Prompt. Right-click on the Command Prompt icon and select 'Run as Administrator'.

In the window that opens, change directory to the \cag\scripts directory:

```
cd \cag\scripts
```

Execute the installservice.bat or uninstallservice.bat as required.

```
installservice.bat
```

## Migrating an installation to another computer

---

The gateway can be easily moved to another computer without having to reinstall or reconfigure.

The following steps need to be carried out. These steps do not include the migration of the Oracle Database - you will need to separately install Oracle on the new computer and migrate the 'CAG' user if required.

### On the source computer

- Uninstall the gateway by executing \cag\scripts\uninstallservice.bat in an administrative command prompt.
- Copy the entire \cag directory to the other computer's root drive. Note that it needs to be installed at the root of any drive, for example, C:\cag or D:\cag. Other locations are currently not supported.

### On the target computer

- Open \cag\conf\basic.conf and ensure that your Oracle settings are still valid on the new computer.
- Install the gateway by executing \cag\scripts\uninstallservice.bat in an administrative command prompt.
- Locate the gateway service in Start - Control Panel - Administrative Tools - Services, and then searching for 'Cloud Alert Messaging Gateway', and select the 'Start Service' icon to start it.
- Navigate your browser to <http://127.0.0.1:2199> to ensure that the gateway is operational.

## **Additional information and Support**

---

Should you have any additional questions, comments or feedback on the Messaging Gateway, contact us over email at:

`support@cloudalert.cloud`

or visit our website at

<https://cloudalert.cloud>