

Contact Verification Server



Contact Verification Server

Quick Start Guide

COPYRIGHT

Companies; names; and data used in examples herein are fictitious unless otherwise noted. No part of this document may be reproduced or transmitted in any form or by any means; electronic or mechanical; for any purpose; without the express written permission of Melissa Data Corporation. This document and the software it describes are furnished under a license agreement; and may be used or copied only in accordance with the terms of the license agreement.

Copyright © 2012 by Melissa Data Corporation. All rights reserved.

Information in this document is subject to change without notice. Melissa Data Corporation assumes no responsibility or liability for any errors; omissions; or inaccuracies that may appear in this document.

TRADEMARKS

Address Object; Phone Object; Name Object; Email Object, and GeoCoder Object are registered trademarks of Melissa Data Corp. Windows is a registered trademark of Microsoft Corp.

The following are registered trademarks of the United States Postal Service®: United States Postal Service; USPS, ZIP.

CONTACT

MELISSA DATA CORPORATION

22382 Avenida Empresa

Rancho Santa Margarita; CA 92688

Phone: 1-800-MELISSA (1-800-635-4772)

Fax: 949-589-5211

E-mail: info@MelissaData.com

Internet: www.MelissaData.com

For the most recent version of this document, visit

<http://www.melissadata.com/tech/contact-verification-server.htm>

Document Code: CVSQSG

Revision Number: 111216.002

INSTALLING THE CONTACT VERIFICATION SERVER

The Contact Verification Server should be added to your company network server room by your IT department. They are welcomed to change the domain name on the server. However, for the update script and testers hosted on the machine to work properly, you must update a file (see WebSmartTestingTool section).

If you would like to change the host header, go to the IIS manager console, select the WebSmart site, and click *Bindings* on the right pane. It is good practice to refer to the server by machine name instead of IP address.

For fast, future support, Melissa Data may request that your IT department provide Melissa Data support with the access and login information to use Remote Desktop Protocol to connect to the server. We suggest opening a new port instead of using the default 3389 port. We may also request you install third party remote support software GotoAssist if needed. This is strictly optional.

TCP Ports

The following ports should be open on the appliance.

- RDP: 6892 or custom
- FTP: 21
- HTTP: 80
- SMTP: 25 or custom port defined in configuration script.

Setting Up the Web Services

The Contact Verification Server comes with the AddressCheck, Email, Phone, Name, and Geocoder Web services configured and ready for use. Each of these services has a SOAP, XML, and REST protocol interface available. However, a few configurations specific to the user still need to be added.

Each service has its own directory under C:\WebSmart. Under each service directory, we have three directories, one for each protocol: SOAP; XML; and REST. In each of those directories is contained a web.config file to control certain aspects of that service.

The following configurations should be adapted to the individual user in every web.config.

```
<add key="SubjectEmail" value="DQWS 2: Web  
Services application error!"/>
```

This is the subject title for error emails.

```
<add key="toEmail" value="admin@example.com"/>
```

Email address that should receive error emails from the Web service.

```
<add key="CCEmail" value="admin@example.com"/>
```

Additional email address that should receive error emails from the Web service.

```
<add key="smtpServer"  
value="mail.melissadata.com"/>
```

SMTP mail server to use for sending error emails.

```
<add key="fromEmail"  
value="webmaster@melissadata.com"/>
```

The from email address for error emails.

Setting up Scripts

The Contact Verification Server contains scripts that will automatically check for, download, unpack, install, and test new updates. These scripts require a valid download link that is provided to you by your Melissa Data Sales Representative. They can also be retrieved from the monthly download email you receive with a valid subscription. They also require a valid email address in case of any problems. These scripts are located under C:\Scripts.

1. Change into the C:\Scripts\MD_Scripts directory
2. Edit the params.cmd file to make sure the following are correct:

mdscripts variable is set up by default. Do not change this unless you move C:\Scripts. This is not recommended.

```
set mdscripts="c:\Scripts"
```

Set your email and smtp server for email updates on the status and error for updates.

```
set mdemail=set.me@domain.com
```

```
set mdsmtplib=smtpserver.domain.com
```

Set your server name variable to identify different machines if you have more than one CVS.

```
set mdserver=servername
```

Set up your download links. This should be set up by default but may need to be changed if we provide you with a new link. You must preface exception characters with a '%' character.

```
set mddqs="http:xxxxx"
```

```
set mdgeo="http:xxxxx"
```

3. **Extremely Important:** Make sure the MD_LICENSE Environment variable is set to your DQT License for all products. Here is a how-to page for this:

<http://www.melissadata.com/tech/set-environment-variable.htm>

4. Set up scheduled tasks for the four scripts to run every day. They will check for updates and exit gracefully if there are not any.

It is important to setup the task scheduler under an administrator or domain administrator account. Also you will need to change the administrator permissions to full for the C:\Scripts folder and also for each of the batch files in MD_Scripts. As the batch files also have to write some data, the MD_Data directory for each script should also have explicit full permissions.

Detailed Steps.

- 1) Make sure that the task is set to "configure for Windows Vista or Windows 2008" on the first page of the task properties (under the "general" tab)
- 2) Make sure that the task is set to "start in" the folder that contains the batch file: open the task properties, click on the "actions" tab, click on the action and then the "edit" button at the bottom. In the "Edit Action" Window there is a field for "start in (optional)" that you set to the path to the batch file.
- 3) Make sure that the task is running as an account that has explicit "Full access" permissions to all these things: The .bat file itself, the folder containing the .bat file, and the target files/folders that are affected by the .bat script.
- 4) Make sure that the account running the task is a member of the local "administrators" group for this machine
- 5) Make sure that the task is set to "run whether logged on or not"
- 6) The Task should run successfully with expected output when you right-click on the task and select "run" If it does that then it will run successfully when you are logged off.

Make sure to enable these to run as an Administrator user for example: (Change your scripts path and username/password). You must also give the C:\Script directory full control permissions for the account running the scheduled task.

```
SCHTASKS /Create /SC DAILY /TN dqs-update /TR  
c:\PathToScripts\MD_scripts\mdDQSUpdate.cmd /u  
Administrator /p Password
```

Using Testers

Several testing tools are available out of the box on the Contact Verification Server to verify that the data files are installed correctly and that the services are up and running.

Version Checker Web Page

Version Checker is a Web page that displays all the data files dates and expirations as well as build numbers for the libraries. It is already hosted in IIS on the server. The default Web address is <http://appliance.domain.com/Default.aspx>. A shortcut to this page is located on the desktop.

WebSmartTestingTool

In C:\Scripts\MD_Tools, there is a command line tool named WebSmartTestingTool.exe. This will test all or individual services with a basic test to ensure they are up and running. This exe uses the WSTconfig.ini file in the same directory for the URL of each service. For help, type WebSmartTestingTool.exe -h.

Note If you change the name of the machine, you must go into WSTconfig.ini and change the service links in there. This testing tool is used to verify an update ran correctly and the services are up and running. If the links are not valid, the update will fail and data path reverted to the previous update.

Updating Data Files and Libraries

Once configured, the automated scripts will automatically download and install all required updates. They will only download an update if it is new and found on the server. We recommend you analyze and understand how to use the update scripts on the appliance.

You can set these scripts to run on a periodic basis (daily, weekly, monthly...) as a scheduled task. Alternatively, you can have staff perform the updates manually by executing these scripts on a regular basis. Note that these scripts will take the services offline briefly as it restarts IIS. Please take this fact into consideration when designing your update procedures.

NOTE ON PERIODIC CLEANUP

The scripts create a Date stamped folder where the zip files are unpacked. Then, a junction point in the root is created that the WebSmart CVS services are configured to use.

This mechanism allows the CVS services to be always set to use the updated data from the junction point. The CVS update scripts retain the older data files so that they can roll back if an update fails. However, it will be necessary to delete the older file directories in C:\Scripts\MD_Data once every quarter. One each for Geocoder and DQS will need to be deleted.

For example: if two DQ directories are found, you may delete the earlier one:

1. dq2011-05-27 <-- Safe to Delete
2. dq2011-06-27 <-- Live pointing to MDUPDATE Junction point.

Scripts: Advanced

We will now give you a general overview of the update batch scripts. This should give you a better understanding of how the scripts work. Within the C:\Scripts directory:

MD_tools:

This directory contains dll's and executables used by the batch scripts. Most important is WebSmartTestingTool.exe. This executable will try to hit each web service and look for a valid response. If this is not able to execute correctly, the update scripts will fail and revert back to the previous data files.

MD_Scripts:

This directory contains the update scripts themselves. Execute them manually or set them up as scheduled tasks. mdDQSUpdate.cmd updates the data files for the addresscheck, email, phone, and name web services. An update for this script is available every month. mdEWSUpdate.cmd updates the ewe.txt file that contains street names scheduled for future inclusion into the USPS database. It is updated weekly. mdGeoUpdate.cmd updates the data files for the geocoder web service. Updates are available quarterly.

Also important are the dq-install.txt and go-install.txt files. The file name of the update download is logged here so we do not download the same update continually. To force an update, you can delete this file or make it blank.

MD_logs:

This directory contains logs that are generated every time an update batch script runs. When you execute an update batch script, you can come to this directory to check its progress. After an update, this log is emailed to the email address set up in params.cmd.

MD_data:

This directory contains the data files used by the Web services. For every update, a new directory is created. After the download and extraction is complete, the batch script will change the symbolic link for C:\mdupdate from the previous data file directory to the current data file directory. This is done using the linkd command.

LIBRARY UPDATES

Note: The update scripts only update the data files for the CVS: Annually or upon recommendation from MD support it is recommended to update the DQ Suite libraries in the WebSmart Protocol - Bin Directories for each service and each protocol respectively to keep the Builds of the DQ libraries up to date.

C:\WEBSMART\ADDRESSCHECK

```
|—aspnet_client
|   |—system_web
|   |   |—2_0_50727
|—REST
|   |—bin <- mdAddr.dll
|—SOAP
|   |—bin <- mdAddr.dll
|—XML
|   |—bin <- mdAddr.dll
```

C:\WEBSMART\phonecheck

```
|—aspnet_client
|   |—system_web
|   |   |—2_0_50727
|—REST
|   |—bin <- mdPhone.dll
|—SOAP
|   |—bin <- mdPhone.dll
|—XML
|   |—bin <- mdPhone.dll
```

C:\WEBSMART\email

```
|—aspnet_client
|   |—system_web
|   |   |—2_0_50727
|—REST
|   |—bin <- mdEmail.dll
|—SOAP
|   |—bin <- mdEmail.dll
|—XML
|   |—bin <- mdEmail.dll
```

```
C:\WEBSMART\geocoder
```

```
|---aspnet_client
|   |---system_web
|   |   |---2_0_50727
|---REST
|   |---bin <- mdGeo.dll
|---SOAP
|   |---bin <- mdGeo.dll
|---XML
|   |---bin <- mdGeo.dll
```

Using the Web Services

The Web services on the Contact Verification Server are compatible with the services that we host here at Melissa Data. Therefore, you can use the sample code available on our website (<http://www.melissadata.com/tech/websmart.htm>) and simply change the URL to the CVS. The documentation for these services are also available at the link above.

Clustering and Failover

If you receive more than one Contact Verification Server, you can cluster them together to distribute load and provide failover. At Melissa Data, we use Microsoft Network Load Balancing to provide this. All requests come in through a single URL and are distributed to all active servers. If a server goes down or needs to be updated, we take the server off the cluster, drain the server of incoming requests, and perform maintenance/support.

Additionally, a benefit of using the Contact Verification Server is the availability of Melissa Data WebSmart™ Cloud as failover. If the CVS is out of service, clients can simply re-direct to Melissa Data's endpoint. To set this up, it is required to get a valid CustomerID from your sales representative and set it in the Web Service client request.

