

MessageMedia AutoText and SMS Listener Install Checklist

01/27/2025 1:27 pm EST

AutoText Two-Way Texting Integration Only Works With The Following SMS Providers:

1. - Clickatell (Legacy)
2. - Message Media (Legacy)

AutoText One-Way Texting Integration Works With The Following SMS Providers:

1. Legacy (all existing support)
2. Bulk SMS
3. Clickatell
4. Message Media
5. Plivo
6. Twilio

For two-way Autotext the SMS Listener is typically installed on the BoldNet Server. If there is no BoldNet server, the Pre-Install checklist below will need to be done on the media gateway or another server of their choice, and will need to have ports opened externally. If BoldNet is already installed, the Pre-Install Checks will already be in place on that machine. 2 way also requires VRT and VRT lines in the license.

Pre-Install Checklist

- Are the most recent update files downloaded?
- Does the installation user have full admin privileges?
- The following Roles and Features must be added to the MediaGateway Server:

1. Application Server (Role)
2. NET Framework 4.5 with ASP.NET 4.5 and HTTP redirection (Feature)
3. Net Framework 3.5 (Feature) See the right for specific details.
4. Telnet Client (Feature)
5. Windows Powershell (Feature)

The following are only needed for two-way without a BoldNet server. See the above call out for more information.

1. Application Server - Web Server IIS Support (Role)
2. Web Server IIS
3. Web Server IIS → Common HTTP Features → HTTP Redirection
4. Web Server IIS → Application → ASP.NET
5. Web Server IIS → Security → Basic Authentication
6. Web Server IIS → Performance → Dynamic Content Compression (Manitou 1.6.0 and higher)
7. Web Server IIS → Management Tools → IIS 6 Management Compatibility and sub-items (Windows 2008 and higher)

- Can the Manitou server ping the BoldNet server by name?
- Is MediaGateway installed and properly licensed?
- Is the SMS Listener Setup downloaded? (Located in Critical Software on FileServer01)
- Is the MediaGateway2 service set to run as a local user?
- Open a run dialog and run services.msc
- Within the Services window, right click on "Bold MediaGateway2" and select Properties
- Open the Log On tab
- Select the "This Account:" Radio Button
- Enter in the MSSQL Account details
- Select "OK" to save changes and close the Services window
- Are the necessary ports open?

AutoText Install Checklist

- Open MediaGateway
- Select the Connector option under the UniversalConnector Node.
- Enter in the password (pikespeak3) and select "Add"
- Change the Connector Type to SMS Gateway and specify a Connector Device Name (Typically "AutoText")
- Specify the Provider as the Service the customer is using.
- The settings to add will change slightly based on the provider, however, the information can be found at the bottom.
- The sending number will need to include the Country Code if the customer is located outside of the US.
- The "Test To Number" will not include the Country Code
- Specify the Default Country Number and Internal Country Prefix in the Media Gateway Options
- Select the "Options" Node
- Open the Default Menus page
- Save all Changes made (File → Save)
- Import the Basic AutoText Menu ATBASIC, and ATNOTIFY2 for 2 way.
- Select the "Menu Items" Node
- Right click on Menu Configurations and select Import
- Import MS_AutoText Basic Notification (located in the MS_AutoText Menus folder within the SMS Listener Setup package, and the ATNOTIFY2 menu in the same directory.
- Save all Changes made (File → Save)
- Add the AutoText Line Driver
- Select the "Line Drivers" Node
- Right click on Menu Configurations and select Import
- Select the next empty Line Driver dropdown and select the connector previously created (Typically "AutoText")
- Change the Line Function to AutoText
- Right Click on the currently blank Properties Space and select Properties
- Change the Menu to ATBASIC, or ATNOTIFY2 if using 2way (The menu previously imported)
- Select OK
- Save all Changes made (File → Save)
- Reload and Restart all Devices (
- Create a script message to send with AutoText
- Open the Supervisor Workstation on the Primary Server
- Open the Script Messages Page (Maintenance → Script Messages)
- Enter Edit mode and select "Add"

- Give a Description for the Script Message (Typically "AutoText")
- Change the Type to Generic and select "OK"
- Enter in a basic message to be sent within the "Script Text" field.
- At the end of your message within the "Script Text" field, add {\$MCATBASIC} or {\$MCATNOTIFY2} for 2 way.
- Select Save and Close the Supervisor Workstation
- Add the Reverse Channel Commands
- Open the Operator Workstation on the Primary Server
- Open the Monitoring Company (Maintenance → Monitoring Company)
- Select Reverse Commands from the Jump To menu on the Right
- Enter Edit Mode
- Expand MediaGateway on the inner Left Menu
- Select "SMS - SMS Message"
- Match the Optional Parameters to the Parameters shown in the image below

□

- Save the changes
- Test sending a message with AutoText.
- Open the Operator Workstation if not already open
- Create a test account, usually autotexttest
- Open the Contact List from the Jump To menu on the right
- Enter Edit Mode and select "Add"
- Specify a name for the contact in the pop-up window, usually BoldTest
- Under the Contact section select the drop down arrow next to Site and select "AutoText"
- Select the Icon of the hand located next to the drop down arrow
- Specify Media Gateway as the Output Device type
- Select AutoText as the Service Provider
- Select the AutoText script created Earlier
- Specify a number to test with that can receive text messages
- Select OK on both windows, and then Save changes.
- Select "Action Patterns" from the Jump To menu on the right
- Enter Edit Mode
- Select Customer from the menu on the inner left, and then select Add
- Specify "AUTOTEXT" as the Action ID and "Testing AutoText" as the Description
- On the new window, verify the Command is set to Contact, if not select the drop down and change it to Contact.
- Within the Contact List, select the AutoText number we created and select "Add Command"
- Save Changes
- Select Systems from the Jump To menu on the right
- Enter Edit mode
- Under the Event Actions Programming Section specify a new event (Typically *BA or any other unused alarm generating event code)
- The TX Area and Zone can all remain as the default of *
- Leave Alarm as Default
- Set the action ID to "AUTOTEXT" or the named Action Pattern created previously
- Select Save
- Open the Manual Signal Utility (Tools → Manual Signal)

The next steps the customer must be aware of. A live signal will be sent into their queue

- Select an available Transmitter
- Specify the Event we programmed for the action pattern (BA)
- Send the Signal
- Open the Alarm Queue (Operations → Alarm Queue)
- Double click on the Alarm sent in
- Double click on the Action Pattern action for AutoText
- Verify the script message has been received on the contact's phone.

SMS Listener Install Checklist

The following only needs to be done if the customer is using two-way. These steps need to be done on the Web Server which is typically BoldNet, but it could be any server including the MediaGateway. Refer to the Call out at the top of the page for more information.

- Download the SMS Listener Setup Package to their BoldNet server. (Or server that will have Web capabilities.)
- Within the setup package, run AspNetMVC3ToolsUpdateSetup.exe as administrator
- Once complete, run the AspNetMVC4Setup installer.
- Copy the SmsListener folder to the bold wwwroot directory (Typically the default directory at C:\inetpub\wwwroot)
- Create a SMS temp directory, DO NOT USE TEMP or TEMP2. (should generally be C:\MessageMedia)
- Share the created SMS directory with "Everyone"
- Right click on the folder and select properties
- Open the Security Tab and Select "Edit..."
- Select "Add..."
- Type in Everyone in the object names box and select "OK" (You may need to change the location to the local machine if on a domain)
- Change the Permissions for Everyone to Allow Full control
- Select "OK"
- Open the Sharing Tab and Select "Share"
- Everyone should already be listed as a name, if not type in Everyone and select "Add"
- Verify Everyone has the Read/Write Permission Level
- Select "Share", then "Done"
- Close the TEMP Properties window
- Within the SmsListener directory in wwwroot Open SmsListener.cshtml with notepad and select all and then copy/ paste the following into the cshtml.

```
@functions {
public string LogRequest()
{
List<Tuple<string, string>> hostMapping = new List<Tuple<string,string>>();

const string BASE_FILE_PATH = "C:\\MessageMedia\\";

hostMapping.Add(new Tuple<string, string>("wan1-fg.m4u.com.au", "MESSAGEMEDIA"));
hostMapping.Add(new Tuple<string, string>("webpush.m4u.com.au", "MESSAGEMEDIA"));

string hostName = "";
try
{
hostName = Dns.GetHostEntry(Request.UserHostAddress).HostName;
```

```

}
catch(System.Net.Sockets.SocketException)
{
    hostName = Request.UserHostAddress;
}

Tuple<string, string> hostMap = hostMapping.Find(m => m.Item1.ToUpper() == hostName.ToUpper());

if (hostMap != null)
{
    using (System.IO.StreamWriter file = new System.IO.StreamWriter(string.Format("{1}{0}-{2}{3}", hostMap.Item2, BASE_FILE_PATH, Guid.NewGuid().ToString(), ".sig")))
    {
        /* -- Debugging code --
        file.WriteLine(string.Format("Query string: {0}", Request.QueryString.ToString()));
        file.WriteLine(string.Format("Verb: {0}", Request.HttpMethod));
        if (Request.HttpMethod == "POST")
        {
            System.IO.StreamReader sr = new System.IO.StreamReader(Request.InputStream);
            file.WriteLine(string.Format("Request body: {0}", sr.ReadToEnd()));
            file.WriteLine();
            file.WriteLine();
        }
        */

        file.WriteLine(string.Format("BOLDPROVIDER={0}", hostMap.Item2));
        foreach (string key in Request.QueryString.Keys)
        {
            String val = Request.QueryString[key];
            if (key == "phone" && val.Length == 11 && val[0] == '1')
            {
                val = val.TrimStart('1');
            }
            file.WriteLine(string.Format("{0}={1}", key.Replace("=", ""), val));
        }
        if (Request.HttpMethod == "POST")
        {
            System.IO.StreamReader sr = new System.IO.StreamReader(Request.InputStream);
            string postBody = sr.ReadToEnd();
            if (postBody.Length > 0)
            {
                string[] parameters = postBody.Split("&".ToCharArray());
                foreach (string p in parameters)
                {
                    string[] parts = p.Split("=".ToCharArray());

                    if (parts.Length == 2)
                    {
                        file.WriteLine(string.Format("{0}={1}", HttpUtility.UrlDecode(parts[0]).Replace("=", ""), HttpUtility.UrlDecode(parts[1])));
                    }
                }
            }
        }

        return "OK";
    }
    using (System.IO.StreamWriter file = File.AppendText(string.Format("C:\\MessageMedia\\SmsListener_Log.txt")))

```

```

{
    file.WriteLine(string.Format("Request blocked from [{0}]", hostName));
}
return "YOUR HOSTNAME " + hostName + " IS NOT AUTHORIZED";
}
}

<!doctype html>
<html>
<head>
<title></title>
</head>
<body>
@{
    Write(LogRequest());
}
</body>
</html>

```

Open Internet Information Services

- Expand the server name → Sites → Default Web Site (may be different if there is a specified Bold site)
- Right click the SmsListener directory and select "Convert to Application"
- Verify the Alias is set to "SmsListener" and the Application pool is set to ".NET 4.0 Integrated" (Typically the DefaultAppPool)
- Select "OK"
- The SmsListener site is now live, test navigating to the site.
- Open a browser
- Navigate to <https://<Web Server URL>/smslistener/smslistener.cshtml>
- The top of the loaded page should show "YOUR HOSTNAME <Your IP Address> IS NOT AUTHORIZED"
- Open the Messagemedia directory previously created, there should not be a document in the folder called SmsListener_Log.txt
- Open the text document and verify there is an entry for the blocked request
- Close the text document and any other remaining open windows
- On the MediaGateway Server, open the MediaGateway
- Select the "Connector" Node from the left
- Open the AutoText connector created from previous steps and select "EDIT"
- Change the Response directory to the TEMP directory created on the Web server (typically formatted as \\SERVERNAME\MessageMedia)
- Select Update and then save changes (File → Save)
- Re-load and restart all devices (File → Re-load & Restart all Devices)
- Check the Activity Log for any errors while the line drivers restart
- Upload a response Menu into the MediaGateway to test receiving responses
- In the MediaGateway select the Menu Items node
- Right Click on Menu Configurations and Select Import
- Import the AutoText Notify and Cancel Menu
- Assign the Menu to your AutoText Line Driver
- In the MediaGateway, select the Line Drivers node
- Right Click in the Properties area of the AutoText Line Driver
- Select the newly imported menu (ATNOTIFY2)
- Select OK

- Save all changes (File → Save)
- Re-load and Restart all Devices (File → Re-load & Retart All Devices)
- Change the Script for testing AutoText from the Basic script to the Notify and Close script
- Open the Supervisor Workstation on the Primary Server
- Open the Script Messages Page (Maintenance → Script Messages)
- Enter Edit mode and select the previously created AutoText menu.
- Change the script text to say something similar to "A {ET} alarm {ZD} {PO} occurred on {DT} at {TM}. Please send 1 to cancel."
- Change the ending Menu identification from {\$MCATBASIC} for 1 way to {\$MCATNOTIFY2} for 2 way
- Select Save and Close the Supervisor Workstation
- Test sending a signal and closing the alarm via SMS messaging
- Open the Operator Workstation
- Create an autotext test account named AutotextTest.
- Add a contact BoldTest with your phone number for testing.
- Use the output device type Autotext
- Use the service provider Autotext
- Verify the post-processing is programmed to use your auto text script. I typically set it to *BA, zone *, area *, using the autotext script.
- Send a manual signal following the same process as before
- Grab the signal and start the action pattern.
- Once the SMS message has been received, reply to the message with "1"
- Watch the MediaGateway log to verify the response is received.
- Once the response has been received, verify the alarm has now been canceled with Autoclient.
- Once the alarm has shown as canceled repeat the test signal and send a 2 to request a callback, verify it added a comment.
- Once verified close the alarm.
- Close the Supervisor Workstation and any other windows open.
- Notify the customer the work is done!

Troubleshooting AutoText

Common Error #1 - The MediaGateway is showing errors in the Activity Log that it can not access the Response Director TEMP. This error will display if the file is not shared or the MediaGateway is running as a System Account instead of a Local User

Steps to Troubleshoot

- Open a File Explorer Window
- Navigate to the shared Temp directory (\\ComputerName\MessageMedia)
- If the directory opens, the file is accessible, if not verify permission and sharing settings
- If the directory opens but MediaGateway is still stating that it cannot access the directory
 - This will mean the MediaGateway is still running as a System Account
 - Open a run dialog and run services.msc
 - Within the Services window, right click on "Bold MediaGateway2" and select Properties
 - Open the Log On tab
 - Select the "This Account:" Radio Button

- Enter in the MSSQL Account details
- Select "OK" to save changes and close the Services window
- On the Primary Server open the MSM
- Right click on the MediaGateway and select Stop
- Right click on the MediaGateway and select Start
- Open the MediaGateway Console and verify the error messages are no longer occurring

Provider Information

- They stopped using the old HTTP API, they now host the legacy API over HTTPS here is that legacy HTTP API. The details for this are below:
- <https://http-api.m4u.com.au/api/send.php?phone=XXXXXX&username=XXXXXX&password=XXXXXX&message=Test&id=123456789>
- The fields for this are as follows:
 - phone: This is the phone number, with the country code. If no country code is provided, +61 (Australia) is assumed. International destinations are supported, however, you must enter in full international format with a leading + sign.
 - username: This is the MessageMedia username.
 - password: This is the password associated with the username.
 - message: This is the message to be sent. This must be html encoded for special characters (eg spaces as %20, etc). A table of the encoded values is at the following link: http://www.w3schools.com/TAGS/ref_urlencode.asp
 - id (optional): 9 character numeric id for the message. This will be shown under the ID field in reports. If left blank, messages will have an id of 0.
- While testing with a browser the response you would get on success is a HTTP 200 with 'OK' as the body.

CDYNE
