



Backup Gateway Documentation

Written by Jateen Mistry
Revised: 18th May 2004,
Revised: 29th May 2004 [Adam Beaumont]

SUPPORT:

For support related issues please logon to the support forum at <http://aqcorporate.com/support.php> and raise a new ticket. The support screen loads in a new window so please ensure all pop-up blocking software is **disabled** for the support site.

Contact us

If you have a sales related question please call us on
08707 449227
08707 449228 (fax)

If you have any technical questions regarding backup gateways please raise a support ticket.

sms2email.com

sms2email is a service provided by (aq)



Systems Overview

(aq) operate a network with 3 points of presence, Figure 1, with our main services running out of 2 interlinked data centres (2 and 4). A third point of presence (5) runs our tertiary backup services such as mirrored backup HTTP post SMS Gateways (Figure 2) allowing API users access to a redundant route should our main systems be inaccessible for any reason.

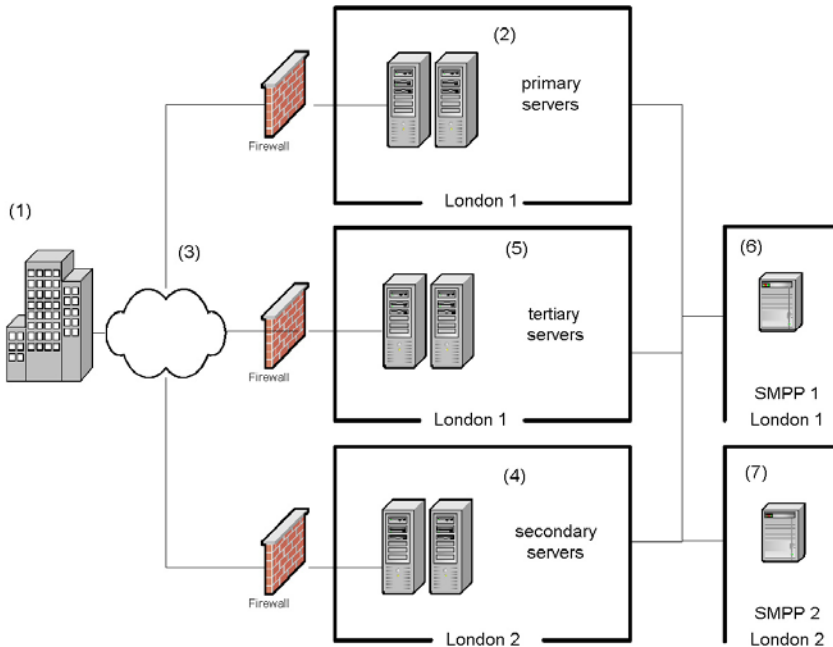


Figure 1, schematic overview of geographic arrangement of network architecture.

Figure 1. Networks Schematic – Geographic.

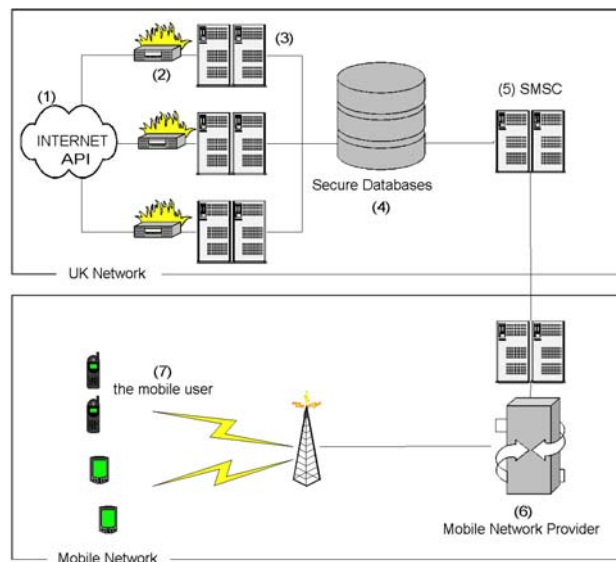


Figure 2. IP based API to GSM Schematic.

Using the backup gateways

The standard method to send a message via sms2email can be broken down into the following steps:

1. Connect to the sms2email server
2. Send message to server
3. Receive response and handle it appropriately.

If for some reason the connection between yourselves and sms2email became unavailable or if the server was temporarily down, the above method would not allow your messages to get through. It would not get past step 1.

For this reason we recommend that you insert some program logic into your application that would be able to use our backup gateways should the above problems occur.

The backup gateways servers and the order in which to try and connect are listed below:

1. **gw1.sms2email.com**
2. **gw11.sms2email.com**
3. **gw2.sms2email.com**
4. **gw22.sms2email.com**

Please Note: The backup servers listed above are the actual server addresses. An example of a full URL to send a text message via the “postmsg.php” interface is:

<http://gw11.sms2email.com/sms/postmsg.php>

All the gateways listed on the developer info web page are accessible via all of the servers listed above, simply replace www.sms2email.com with eg : gw1.sms2email.com



Program Logic in Pseudocode

The following pseudocode is the logic you would need to implement in your application.

```
array_servers = "gw1.sms2email.com", "gw11.sms2email.com",  
                "gw2.sms2email.com", "gw22.sms2email.com"  
  
connected = FALSE  
for every item in array_servers AS current_server  
{  
    handle = connect(current_server)  
    if(handle is valid)  
    {  
        connected = TRUE  
  
        //Send the message to  
        //http://current_server/sms/postmsg.php  
        //deal with response  
  
        CloseHandle(handle)  
        break/exit loop  
    }  
}  
if(connected is FALSE)  
{  
    //could not connect at all. Deal with this error.  
    exit  
}
```

It basically loops through the list of servers and sends out to the first one that is available. This should almost always result in the connection being established to **gw1.sms2email.com**

The following pages contain actual code examples of how to connect to sms2email using the backup gateway servers. The examples are in PHP, Perl and ASP.

sms2email.com

sms2email is a service provided by (aq)



Example in PHP

```
// your aqsms username and password
$username = "yourusername";
$password = "yourpassword";
$path = "/sms/postmsg.php";
$connected = FALSE;
$message = "This is a test message";
$number = "44712345678";

// encodes your message for sending on the web
$message = urlencode($message);

// the request string
$smsurl =
"$path?username=$username&password=$password&to_num=$number&message=$message";

// connects to server to send message

$array_servers = array("gw1.sms2email.com", "gw11.sms2email.com",
                      "gw2.sms2email.com", "gw22.sms2email.com");
foreach($array_servers as $server)
{
    //this line sets the server we wish to connect to
    $connecturl = "http://$server".$smsurl;
    if($content_array = file($connecturl))
    {
        $connected = TRUE;
        $content = implode("", $content_array);
        // check for response
        if (ereg("AQSMS-AUTHERROR",$content))
            echo "There was an authentication error";
        elseif (ereg("AQSMS-NOMSG",$content))
            echo "There was no message or mobile number";
        elseif (ereg("AQSMS-OK",$content))
            echo "The Message was queued successfully";
        elseif (ereg("AQSMS-NOCREDIT",$content))
            echo "Your account has no credit";

        break; //this breaks the foreach loop
    }
}

if($connected == FALSE)
{
    //none of sms2email servers were available.
    //Deal with this situation appropriately
    echo "There was an error connecting to all sms2email servers";
}
```



Example in PHP using CURL and sending via HTTPS

```
$connected = FALSE;
$array_servers = array("gw1.sms2email.com", "gw11.sms2email.com",
                      "gw2.sms2email.com", "gw22.sms2email.com");

foreach ($array_servers as $current_server)
{
    $ch = curl_init(); // initialise curl handle
    $data =
        "username=xxxxx&password=xxxxx&message=a%20txt%20message&to_num=447766404142";

    //the commented out line shows how to connect via normal HTTP, not HTTPS
    //$url = "http://$current_server/sms/postmsg.php";
    $url = "https://$current_server/sms/postmsg.php";

    curl_setopt($ch, CURLOPT_SSL_VERIFYHOST, 0); // don't try and do any clever ssl

    //the above line is used because the SSL cert will not always match the gateway name

    curl_setopt($ch, CURLOPT_URL, $url); // set url to post to
    curl_setopt($ch, CURLOPT_POST, 1); //want to use HTTP POST not GET
    curl_setopt($ch, CURLOPT_POSTFIELDS, $data); //the data to post
    curl_setopt($ch, CURLOPT_RETURNTRANSFER, 1); // return response into a variable

    //send whole request
    $result = curl_exec($ch);
    curl_close($ch);

    /*
    at this stage, the message has been sent to
    the sms2email servers and we have the response
    We need to deal with the response in an appropriate way.
    */
    if ($result !== FALSE)
    {
        $connected = TRUE;
        switch($result)
        {
            case "AQSMS-OK":
                //the message was successfully queued with sms2email
                //deal with this case
                break;
            case "AQSMS-NOCREDIT":
                //the message was not sent as there is
                //insufficient credit on your account
                //buy more credit and retry.
                //deal with this case
                break;
            case ".....":
                //end the for loop
                break;
        }
    }
}
if($connected == FALSE)
{
    //none of sms2email servers were available.
    //Deal with this situation appropriately
    echo "There was an error connecting to all sms2email servers";
}
```

sms2email.com

sms2email is a service provided by (aq)



Example in Perl

```
#!/usr/bin/perl -w

use LWP::UserAgent;
use HTTP::Request::Common;
use URI::Escape;

%post_data = ();
$post_data{'username'} = "username";
$post_data{'password'} = "password";
$post_data{'to_num'} = "4477123456";
$post_data{'orig'} = "+4477123456";
$post_data{'message'} = "this is my message !";
$post_type = "http://";
@post_servers = ("gw1.sms2email.com", "gw11.sms2email.com",
                "gw2.sms2email.com", "gw22.sms2email.com");
$post_path = "/sms/postmsg.php";

my $ua = LWP::UserAgent->new(agent => 'sms2email perl client');

foreach $server (@post_servers) {
    my $resp = $ua->request(
        POST $post_type.$server.$post_path,
        Content_Type => 'form-data',
        Content => [ %post_data ]
    );

    if ($resp->is_success) {
        print $resp->content;
        last;
    } else {
        print "Error: ".$post_type.$server.$post_path." -
            ".$resp->status_line."\n";
    }
}
}
```

Example in ASP

```
<%
Dim reply
reply = sendsms("username", "pwd", "this is a message",
"44796111222", "+447871122334", 0, 1)
Response.write reply
%>

<%
Function sendsms(username, password, message, to_num, orig_num,
flash, secure)
On Error Resume Next
Dim Gateway, Prefix, Suffix, httpObj, AQresponse, i
message = Server.URLEncode(message)
username = Server.URLEncode(username)
password = Server.URLEncode(password)
to_num = Server.URLEncode(to_num)
orig_num = Server.URLEncode(orig_num)
Gateway =
Array("gw1.sms2email.com", "gw11.sms2email.com", "gw2.sms2email.com", "g
w22.sms2email.com")
Suffix = "/sms/postmsg.php"
If secure = 1 Then
Prefix = "https://"
Else
Prefix = "http://"
End If
postString = "username=" & username & "&password=" & password &
"&to_num=" & to_num & "&message=" & message & "&flash=" & flash &
"&orig=" & orig_num
Set httpObj = Server.CreateObject("Msxml2.ServerXMLHTTP.3.0")
httpObj.setTimeouts 0,2000,2000,2000
httpObj.setOption 2, 13056
i = 0
Do
Err.Clear
httpObj.Open "POST", Prefix & Gateway(i) & Suffix, False
httpObj.setRequestHeader "Content-Type", "application/x-www-form-
urlencoded"
httpObj.Send postString
i = i + 1
Loop Until Err.Number = 0
AQresponse = httpObj.responseText
Set httpObj = nothing
sendsms = AQresponse
End Function
%>
```


Example in ColdFusion

```

<!-- begin code -->
<!-- set up the 4 servers -->
<cfset urltotry =
"http://gw1.sms2email.com/sms/postmsg.php,http://gw11.sms2email.com/sms/post
msg.php,http://gw2.sms2email.com/sms/postmsg.php,http://gw22.sms2email.com/s
ms/postmsg.php">
<CFSET trynext="1">
<cfset session.numbers_sent="">

<cfloop index="listurl" list="#urltotry#">
  <cfif trynext is "1">
    <cftry>
      <cftransaction>
        <cfhttp url="#listurl#" method="post" resolveurl="no"
        thrownerror="yes" timeout="2000">
          <cfhttpparam type="FORMFIELD" name="to_num"
            value="#Replace('#attributes.to_num#', '0', '44')#">
          <cfhttpparam type="FORMFIELD" name="message"
            value="#attributes.message# sent from #listurl#">
          <cfhttpparam type="FORMFIELD" name="username"
            value="#attributes.sms2email_username#">
          <cfhttpparam type="FORMFIELD" name="password"
            value="#attributes.sms2email_password#">
          <cfhttpparam type="FORMFIELD" name="orig"
            value="#attributes.sms_orig#">
          <cfif attributes.flash is "Y">
            <cfhttpparam type="FORMFIELD" name="flash" value="1">
          </cfif>
          <cfif attributes.reply is "N">
            <cfhttpparam type="FORMFIELD" name="orig"
              value="#attributes.keyword#">
          <cfelse>
            <cfhttpparam type="FORMFIELD" name="orig"
              value="447766404142">
          </cfif>
        </cfhttp>
        <cfoutput><br>sent from #listurl#</cfoutput>
      </cftransaction>
      <CFSET trynext="0">
      <cfif session.numbers_sent is "">
        <cfset session.numbers_sent = "#attributes.to_num#">
      <cfelse>
        <cfset session.numbers_sent =
          "#attributes.to_num#,#session.numbers_sent#">
      </cfif>
      <cfcatch type="any">
        <CFSET trynext="1">
        <!-- <cfoutput>error #listurl#</cfoutput> -->
      </cfcatch>
    </cftry>
  </cfif>
</cfloop>
<!-- | <cfoutput>#session.numbers_sent#</cfoutput><BR> -->

```



Example Shell Script using Curl

```
#!/bin/sh
#
# Sample script to send an SMS email notification to SMS2Email's HTTP gateways
# Username and password associated with SMS2Email account
username=
password=
number=
message="Test message"
if [ $# -eq 0 ]; then
    echo "Usage: $0 -n [number] -m [message] -u [username] -p [password]";
    echo "";
    echo "[number]    = SMS number to send message to";
    echo "[message]    = Text of message you want to send";
    echo "[username]    = Username associated with SMS2Email account";
    echo "[password]    = Password associated with SMS2Email account";
    echo "
    Both the username and password options are optional and";
    echo "
    override the account credentials defined in this script.";
    echo "";
    exit 1;
fi
# Get command line arguments
while [ "$1" != "" ] ; do
    case $1
    in
        -n)
            # Get the SMS number that we should send message to
            number=$2;
            shift 2; ;;
        -m)
            # Get the message we should send
            message=$2;
            shift 2; ;;
        -u)
            # Get the username
            username=$2;
            shift 2; ;;
        -p)
            # Get the password
            password=$2;
            shift 2; ;;
        *)
            echo "Unknown option: $1"
            exit 1; ;;
    esac
done
message_sent_ok=0;
for server in gw1 gw11 gw2 gw22; do
    RESPONSE=`curl -s -d username=$username -d password=$password -d to_num=$number -d
message="$message" http://$server.sms2email.com/sms/postmsg.php`
    if [ "$?" -eq "0" ]; then
        case $RESPONSE
        in
            AQSMS-OK)
                # Message was queued ok
                message_sent_ok=1;
                echo "Message posted OK to HTTP gateway $server"
                exit 0; ;;
            AQSMS*)
                # Some kind of fatal error occurred
                echo "Fatal error received from HTTP gateway $server: $RESPONSE"
                exit 1; ;;
            *)
                # No response or invalid response
                ;;
        esac
    fi
done
```

sms2email.com

sms2email is a service provided by (aq)

