# Seting Modul IP pada OtomaX

*Oleh: Yusuf AR*

*Note: Scroll down to see* [*English version*](#_Setup_IP_Module)

Pada dokumen ini kita akan membuat modul untuk ambil stok ke supplier lain dengan menggunakan Modul IP. Dengan asumsi:

* Software supplier menggunakan ST24
* IP supplier: 110.90.40.162
* URL yang disediakan supplier untuk menerima *request* adalah <http://110.90.40.162/xmlrequest.php>
* Format XML untuk transaksi:

<?xml version="1.0"?>

<methodCall>

<methodName>topUpRequest</methodName>

<params>

<param>

<value>

<struct>

<member>

<name>MSISDN</name>

<value><string>0818111222</string></value>

</member>

<member>

<name>REQUESTID</name>

<value><string>123456</string></value>

</member>

<member>

<name>PIN</name>

<value><string>6633</string></value>

</member>

<member>

<name>NOHP</name>

<value><string>0818111333</string></value>

</member>

<member>

<name>NOM</name>

<value><string>XR10</string></value>

</member>

</struct>

</value>

</param>

</params>

</methodCall>

* Format XML untuk laporan/report:

<?xml version="1.0"?>

<methodResponse>

<params>

<param>

<value>

<struct>

<member>

<name>RESPONSECODE</name>

<value><string>00</string></value>

</member>

<member>

<name>REQUESTID</name>

<value><string>123456</string></value>

</member>

<member>

<name>MESSAGE</name>

<value><string>TOP UP BERHASIL</string></value>

</member>

<member>

<name>TRANSACTIONID</name>

<value><string>666333</string></value>

</member>

</struct>

</value>

</param>

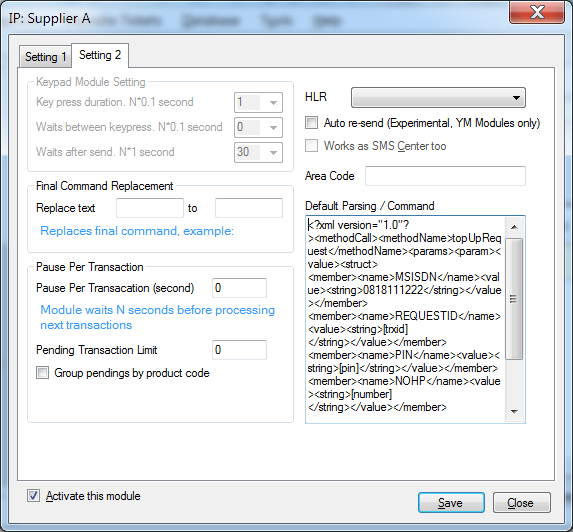
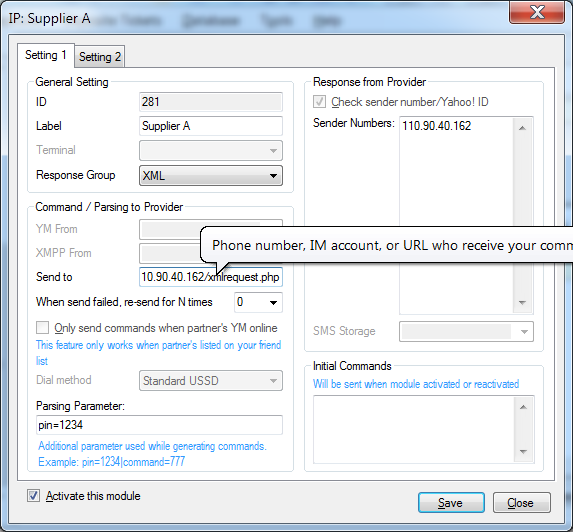
</params>

</methodResponse>

* IP Statik local kita: 110.90.40.99
* URL yang kita sediakan untuk menerima *report* dari supplier adalah: <http://110.90.40.99/xmlreport.php>

Tahapan:

1. Buat modul IP di OtomaX. Caranya kurang lebih sama dengan membuat modul jenis lainnya.



* Send to kita isi URL supplier: <http://110.90.40.162/xmlrequest.php>
* Response from provider kita isi IP static supplier: 110.90.40.162. Jika IP supplier lebih dari satu (1), bisa kita tambahkan dibaris berikutnya
* Default Parsing kita isi xml seperti dibawah. Perhatikan bahwa xml sudah kita modif dengan memasukkan parameter-parameter dari otomax seperti: [trxid], [nominal], [tujuan], dll.

<?xml version="1.0"?>

<methodCall>

<methodName>topUpRequest</methodName>

<params>

<param>

<value>

<struct>

<member>

<name>MSISDN</name>

<value><string>0818111222</string></value>

</member>

<member>

<name>REQUESTID</name>

<value><string>**[trxid]**</string></value>

</member>

<member>

<name>PIN</name>

<value><string>**[pin]**</string></value>

</member>

<member>

<name>NOHP</name>

<value><string>**[tujuan]**</string></value>

</member>

<member>

<name>NOM</name>

<value><string>XR**[nominal]**</string></value>

</member>

</struct>

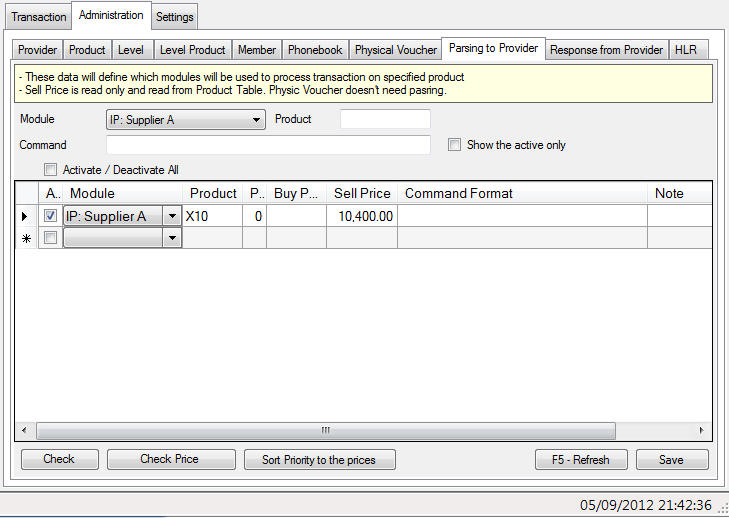
</value>

</param>

</params>

</methodCall>

1. Menentukan produk apa saja yang akan diproses oleh modul IP tersebut. Cukup tentukan produk apa saja, parsing/command dikosongkan karena akan mengambil parsing dari seting Modul (lihat atas).



Sampai tahap ini, kita sudah bisa melakukan transaksi dengan modul tersebut. Namun status transaksi akan menunggu jawaban karena supplier belum bisa mengirimkan report ke kita.

1. Agar supplier bisa mengirimkan report ke kita, maka kita perlu membuat halaman web yang bisa diakses oleh supplier. Sebagai contoh, kita akan membuat halaman PHP dengan nama xmlreport.php. Halaman PHP tersebut akan kita pasang di web server yang sudah terinstal PHP 5.2 dan extension mssql. Instalasi web server, PHP 5.2 dan extension tidak akan dibahas pada dokumen ini dan menjadi beban kerja masing-masing server.

Isi xmlreport.php adalah sbb:

<?php

function sqlEscape($sql) {

$fix\_str = stripslashes($sql);

$fix\_str = str\_replace("'","''",$sql);

$fix\_str = str\_replace("\0"," ",$fix\_str);

return $fix\_str;

}

$method = $\_SERVER['REQUEST\_METHOD'];

if ($method == 'GET') die("FAILED"); // abaikan jika 'GET'

$content\_type = $\_SERVER['CONTENT\_TYPE'];

// abaikan jika html form

if ($content\_type == 'application/x-www-form-urlencoded') die("FAILED");

$content = trim(@file\_get\_contents('php://input'));

$content = str\_replace("\r\n"," ",$content);

// escape for sql

$content = sqlEscape($content);

if (strlen($content) == 0) die("FAILED"); // abaikan jika isi kosong

// disesuaikan kemudian dengan seting modul, kolom "Terima dari"

$pengirim = $\_SERVER['REMOTE\_ADDR'];

// Connect to MSSQL

$server = '.\\SQLEXPRESS';

$link = mssql\_connect($server);

if (!$link) {

die('Something went wrong while connecting to MSSQL');

}

// Select database

mssql\_select\_db('otomax')

or die('Could not select a database.');

// masukkan data ke tabel inbox\_temp

$SQL = "insert into inbox\_temp (tgl\_entri,pengirim,pesan,status) values (getdate(),'$pengirim','$content',0)";

$result = mssql\_query($SQL)

or die('A error occured: ' . mysql\_error());

// Close the link to MSSQL

mssql\_close($link);

// jawaban untuk supplier

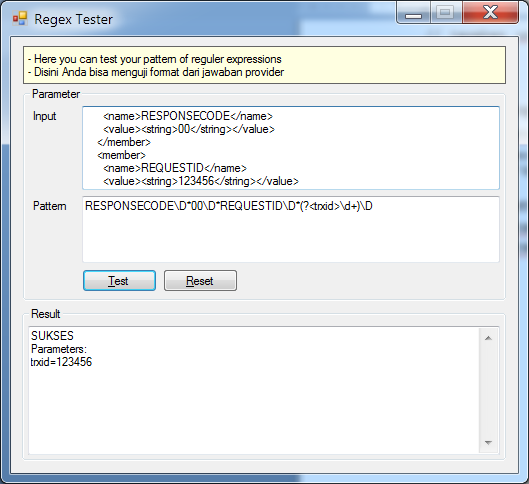
echo 'OK';

?>

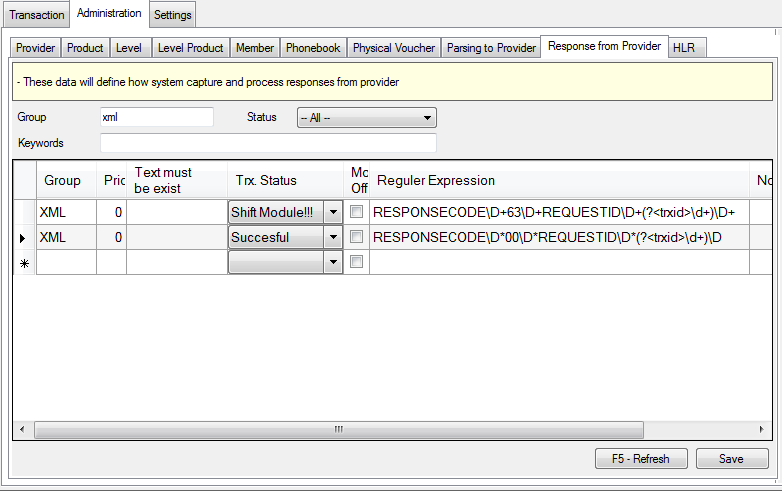
Inti dari isi program php tersebut adalah menangkap pesan yang dikirim dari supplier dan memasukkan data pesan tersebut ke tabel inbox\_temp:

$SQL = "insert into inbox\_temp (tgl\_entri,pengirim,pesan,status) values (getdate(),'$pengirim','$content',0)";

1. Melakukan setting pada halaman Administrasi – Jawaban, agar report dari supplier bisa dikenali dan ditangkap parameternya. Jika report dari supplier menyertakan trxid yang kita kirim, maka kita hanya perlu menangkap trxid tersebut. Dengan contoh report seperti diatas, maka kita bisa gunakan regular expression berikut: RESPONSECODE\D\*00\D\*REQUESTID\D\*(?<trxid>\d+)\D\*



Hasil dengan menggunakan Regex Tester



Aplikasi pada setting jawaban.

**Disclaimer**

Pengembang dan pembuat dokumen ini tidak bertanggung jawab atas segala kesalahan dan kerugian yang diakibatkan atas penggunaan modul IP. Contoh program yang ada pada dokumen ini hanya untuk mengilustrasikan logika / prinsip kerja dari modul IP. Waspadalah terhadap kejahatan cyber seperti hacking, SQL Injection, dll.

# Setup IP Module on OtomaX

*By: Yusuf AR*

In this document we will create a module to take stock of another supplier by using the IP Module. Assuming:

* Supplier use software that supports such method
* Supplier’s IP: 110.90.40.162
* URL provided by the supplier to accept the request is <http://110.90.40.162/xmlrequest.php>
* XML format for topup:

<?xml version="1.0"?>

<methodCall>

<methodName>topUpRequest</methodName>

<params>

<param>

<value>

<struct>

<member>

<name>MSISDN</name>

<value><string>0818111222</string></value>

</member>

<member>

<name>REQUESTID</name>

<value><string>123456</string></value>

</member>

<member>

<name>PIN</name>

<value><string>6633</string></value>

</member>

<member>

<name>NOHP</name>

<value><string>0818111333</string></value>

</member>

<member>

<name>NOM</name>

<value><string>XR10</string></value>

</member>

</struct>

</value>

</param>

</params>

</methodCall>

* XML format for response:

<?xml version="1.0"?>

<methodResponse>

<params>

<param>

<value>

<struct>

<member>

<name>RESPONSECODE</name>

<value><string>00</string></value>

</member>

<member>

<name>REQUESTID</name>

<value><string>123456</string></value>

</member>

<member>

<name>MESSAGE</name>

<value><string>TOP UP BERHASIL</string></value>

</member>

<member>

<name>TRANSACTIONID</name>

<value><string>666333</string></value>

</member>

</struct>

</value>

</param>

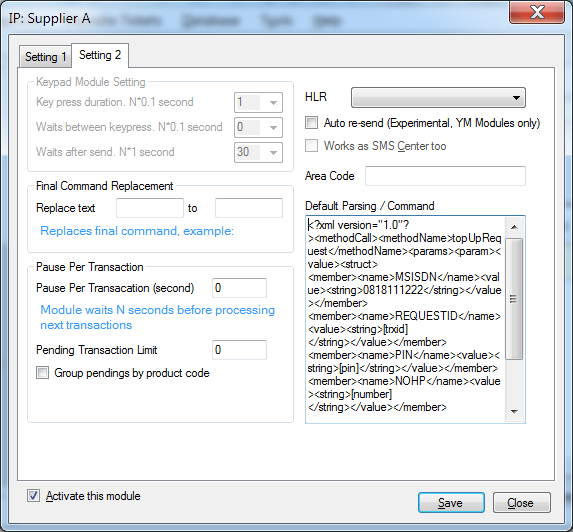
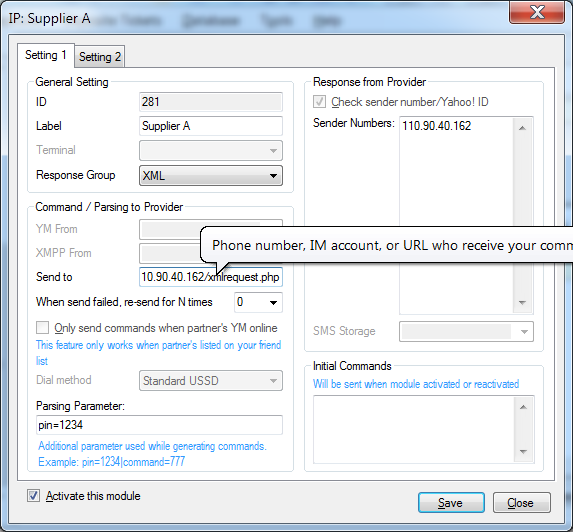
</params>

</methodResponse>

* Our Static IP: 110.90.40.99
* URL that we provide to receive the report of the supplier is <http://110.90.40.99/xmlreport.php>

Steps:

1. Create an IP module in OtomaX. The way about the same as making any other type of module.



* We fill URL supplier into “Send to”: <http://110.90.40.162/xmlrequest.php>
* “Response from provider” is filled with supplier’s static IP Address: 110.90.40.162. If they have more IP Addresses, add them in separated lines.
* Default Parsing is filled XML below. See that we modify the XML by inserting OtomaX’s parameters such as: [trxid], [nominal], [tujuan], etc.

<?xml version="1.0"?>

<methodCall>

<methodName>topUpRequest</methodName>

<params>

<param>

<value>

<struct>

<member>

<name>MSISDN</name>

<value><string>0818111222</string></value>

</member>

<member>

<name>REQUESTID</name>

<value><string>**[trxid]**</string></value>

</member>

<member>

<name>PIN</name>

<value><string>**[pin]**</string></value>

</member>

<member>

<name>NOHP</name>

<value><string>**[tujuan]**</string></value>

</member>

<member>

<name>NOM</name>

<value><string>XR**[nominal]**</string></value>

</member>

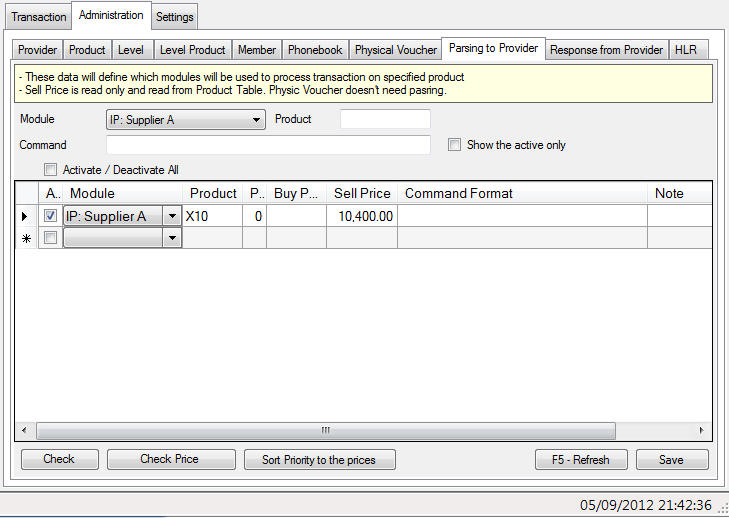
</struct>

</value>

</param>

</params>

</methodCall>

1. Determine which products that will be proceed by the module. Just fill the Product column because command will be taken from “Default Parsing” in the module’s setting (see above). 

At this point, transactions can be done already but status will be Pending, we have not create the web interface so supplier can send response to us.

1. We need to create a web interface so suppliers can send response to us. For an example we will create a PHP page named xmlreport.php. The PHP page will be installed on a Web Server that already installed with PHP 5.2 and an mssql extension. How to install PHP 5.2 and the extension is not described in this document, and is not a part of our regular support.

Scripts in xmlreport.php:

<?php

function sqlEscape($sql) {

$fix\_str = stripslashes($sql);

$fix\_str = str\_replace("'","''",$sql);

$fix\_str = str\_replace("\0"," ",$fix\_str);

return $fix\_str;

}

$method = $\_SERVER['REQUEST\_METHOD'];

if ($method == 'GET') die("FAILED"); // ignores when 'GET'

$content\_type = $\_SERVER['CONTENT\_TYPE'];

// ignores when an html form

if ($content\_type == 'application/x-www-form-urlencoded') die("FAILED");

$content = trim(@file\_get\_contents('php://input'));

$content = str\_replace("\r\n"," ",$content);

// escape for sql

$content = sqlEscape($content);

if (strlen($content) == 0) die("FAILED"); // ignores when empty

// adjust to module’s setting, "Sender Numbers" box.

$pengirim = $\_SERVER['REMOTE\_ADDR'];

// Connect to MSSQL

$server = '.\\SQLEXPRESS';

$link = mssql\_connect($server);

if (!$link) {

die('Something went wrong while connecting to MSSQL');

}

// Select database

mssql\_select\_db('otomax')

or die('Could not select a database.');

// masukkan data ke tabel inbox\_temp

$SQL = "insert into inbox\_temp (tgl\_entri,pengirim,pesan,status) values (getdate(),'$pengirim','$content',0)";

$result = mssql\_query($SQL)

or die('A error occured: ' . mysql\_error());

// Close the link to MSSQL

mssql\_close($link);

// tells the supplier that we already receive the response

echo 'OK';

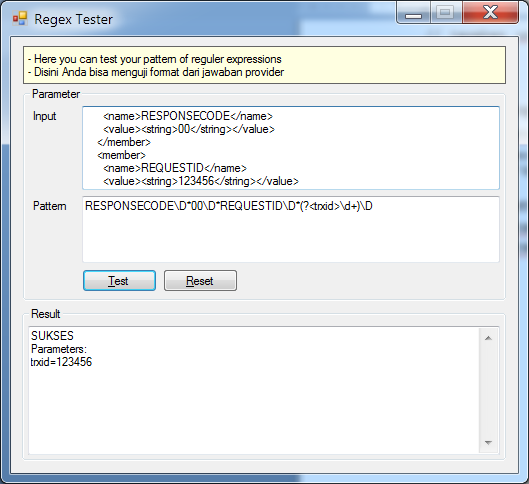
?>

The point of the script about is to capture responses from Suppliers, and insert the responses into inbox\_temp table:

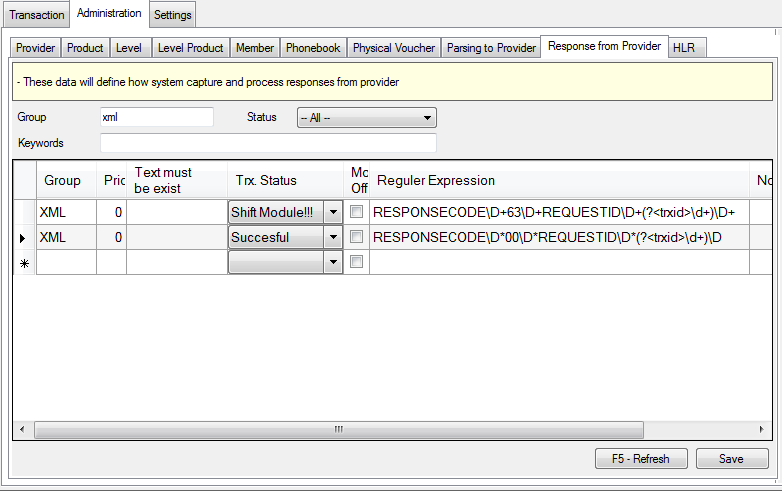
$SQL = "insert into inbox\_temp (tgl\_entri,pengirim,pesan,status) values (getdate(),'$pengirim','$content',0)";

1. Setup the Administration – Response so we can capture the parameters within the response. When response include trxid we sent, than it is the only parameter we need to catch. With the above response sampe, we can use this regular expression:

RESPONSECODE\D\*00\D\*REQUESTID\D\*(?<trxid>\d+)\D\*



Result with the Regex Tester



Setup on the Administration – Response from Provider page

**Disclaimer**

The developer and the author of this document are not responsible for any mistakes or failures caused by the IP module. Use at your own risk. Beware of cybercrime such as hacking, SQL Injection, etc.