Building an “in-house” notification service for Visonic Powerlink2

By junav (junav2@hotmail.com)

Rev 1.0

April 2016

Introduction

PowerMax panels can be connected to Internet through a especial module known as PowerLink.

In the rest of the document I will refer specifically to the version 2 of this module called PowerLink2 (PL2).

With this feature PowerMax users can:

1) Access to alarm panel from web devices, like PC of smartphones, through a HTTP server integrated in PL2
2) Receive e-mail notifications of events detected by alarm panel

In order to receive e-mail notification, it is necessary to use an IP service provided by someone, typically a security company (Central Station) that charge for the service. Visonic PowerMax panels with PL2 has a communication setting with de IP of the selected Central Station ("IP RCVR 1").

Visonic, as a mere equipment manufacturer, has been offering, free of charge, an event notification service called PowerManaged (formely IPMP) through a server (212.179.58.186) located in Israel. Despite the natural reticence of using it, I think this service has been used for many PL2 owners, like me. This service had been active free of charge for years and it was an essential element of the PM "autonomous" alarm reporting system.

In 2011, Tyco adquires Visonic. Tyco is a very large fire protection and security company.

In march of 2016, Tyco-Visonic sent an e-mail to Visonic customers informing them about the imminent shutdown of its free of charge email event notification service. With the effective shutdown of this service, PL2 module will not be able to address email messages to the Powermax owners' mailbox.

This document is a guide for stablish an autonomous (i.e local or "in-house") notification service that, from the PL2 point of view, mimic the Visonic IPMP service. Free of charge, of course.

Notification diagrams

The following pictures depict the event notification process and the elements that are involved.
Figure 1, is the current process where is marked with a red cross the service withdrawn by Visonic.

Figure 2 depict the first solution that I test. This was been the best solution but, if you have a dymanic IP from your ISP, this solution is not reliable as I will explain later.

Figure 3 shows the implemented solution in my house. It uses the free mail services of gmail for sending event notifications to my current mailbox that is a hotmail one.

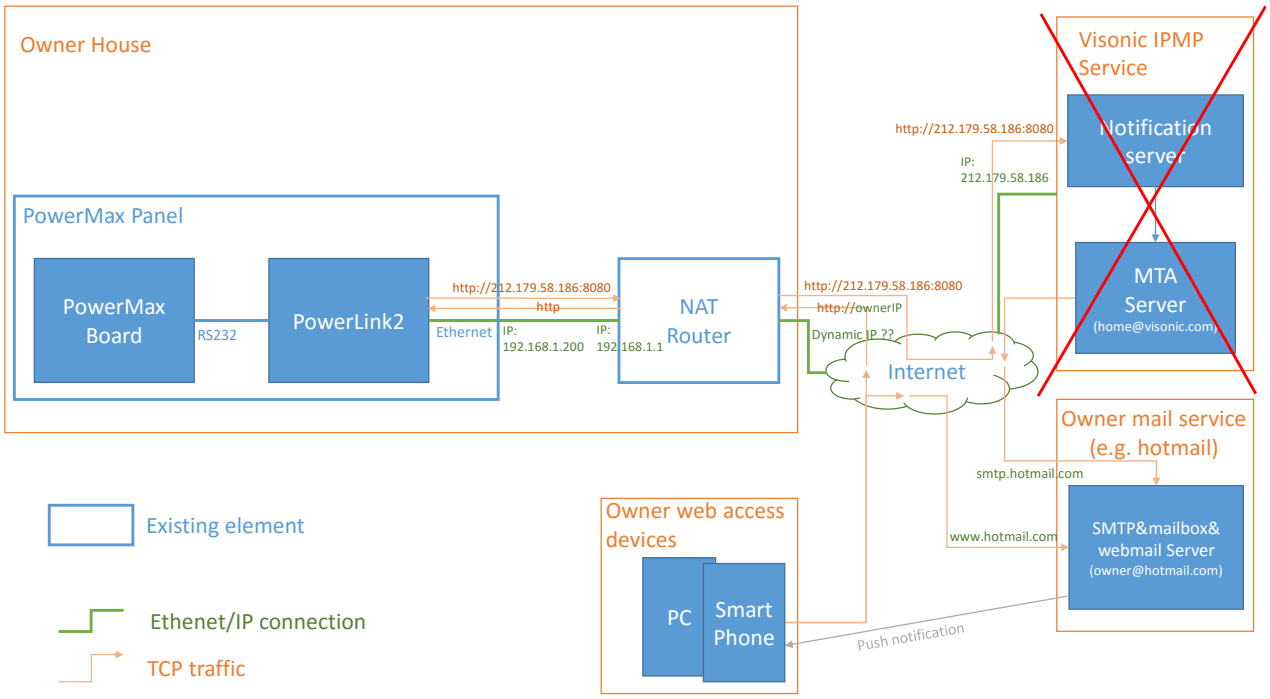

Figure 1- Powerlink2 event notification through Visonic IPMP service (to be discontinued)

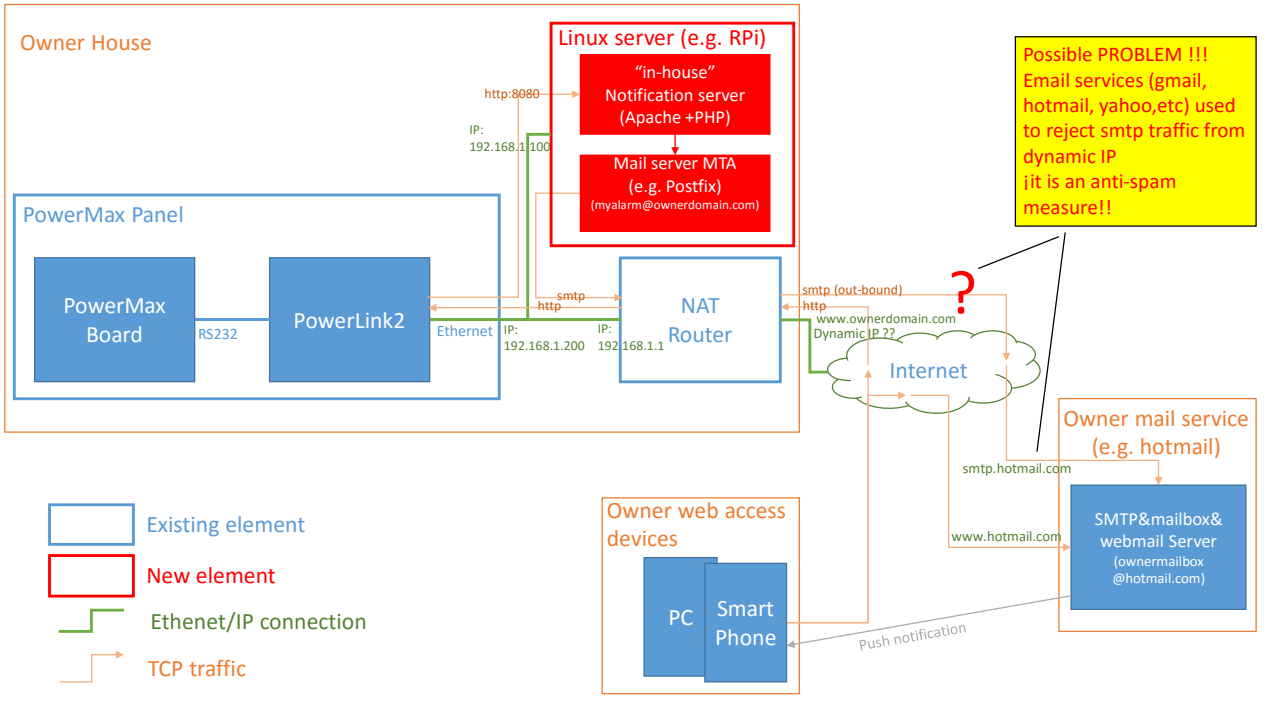

Figure 2- Powerlink2 event notification using a local notification server and a local email server

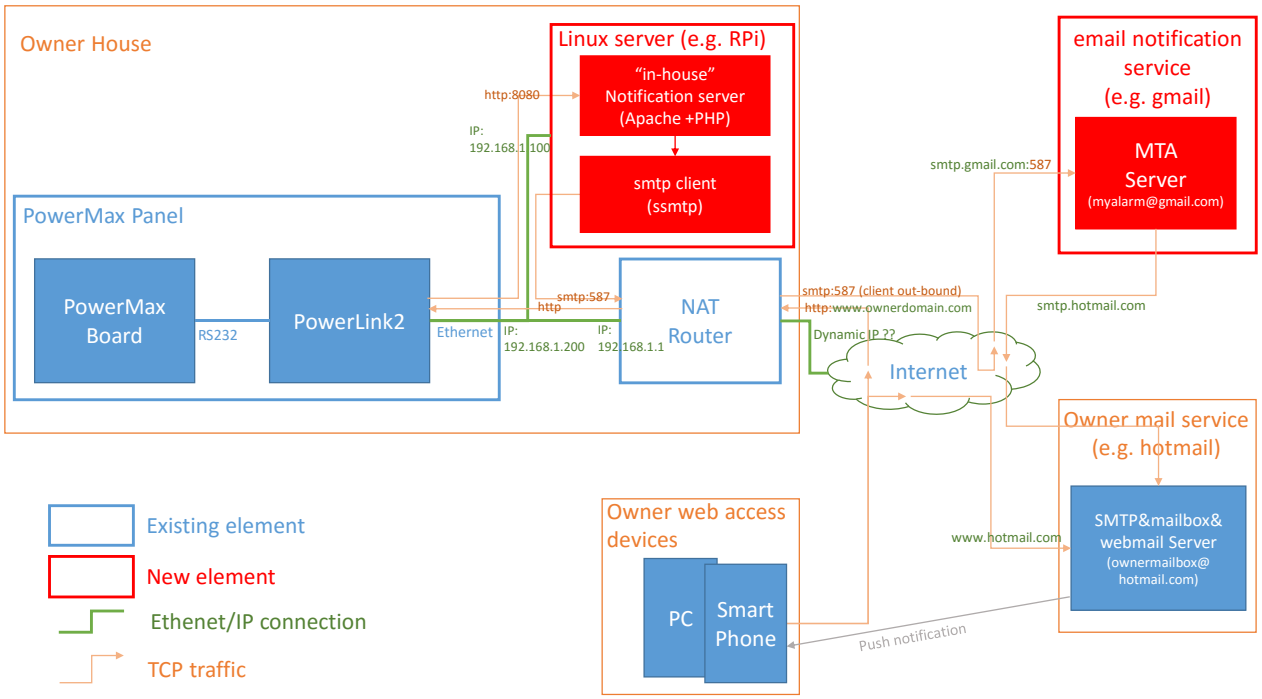

Figure 3- Powerlink2 event notification using a local notification server and a external email service

Sniffing traffic between PowerLink2 and Visonic IPMP server

Using Wireshark and Ethernet switch with a mirror port capability, I was several hours monitoring IP traffic between PowerLink2 and Visonic IPMP server.

In this way, I could see that PL2 send to IPMP server two types of messages:

- Periodic update (keep alive) messages
- Event notify messages

Each message must be replied with an answer from IPMP server to PL2. So emulating the answers of these PHP scripts will be possible to stablish the proper connection between PL2 and another "autonomous" notification server. PL2 will not notice the change.

Like had been advertised by others forum member, it is somewhat disturbing confirm that this traffic (that travel through wild Internet to Israel) is in clear text, that is, without any cypher algorithm. In these messages goes, not only the presence and absence, but mail addresses and telephone numbers.

Update messages (keep alive)

PL2 send every 50s an http (GET) call through 8080 port to a php script in the Visonic server (http://212.179.58.186:8080/scritpts/update.php). Visonic server respond with a string of data. It must be a keep-alive (or heartbeat) mechanism.

Periodic update

Every 50 seconds PL2 send a http GET message to 8080 tcp port of IMPM server.

Call:

```
GET /scripts/update.php?serial=052eeb&id=052EEB&account=001234&ver_sw=6.1.14&ver_hw=1
23&ver_var=6005&upgrade_status=0&configuration_status=0 HTTP/1.1\r\n
Host: 212.179.58.186:8080\r\n
Accept: */*\r\n
```

Answer:

```
HTTP/1.0 200 OK\r\n
Content length: 32\r\n
Content-Type: text/plain; charset=UTF-8\r\n
\r\n
status =0&ka_time =50&allow =0&\n
```

Event notify messages

When an event occur, PL2 send an http (POST) call through 8080 port to an another php script in the Visonic server (http://212.179.58.186:8080/scritpts/notify.php) carrying a XML data stream. This call must be responded with another XML data stream.

Armado parcial (Partial armed?)

Call:

```
POST /scripts/notify.php HTTP/1.1\r\n
Host: 212.179.58.186:8080\r\n
Accept: */*\r\n
Content-Type: text/xml\r\n
User_Agent: persist/1.0\r\n
Connection: keep-alive\r\n
Content length: 749\r\n
\r\n
<?xml version='1.0'?>
<notify>
	<pmax_account>001234</pmax_account>
	<index>16990</index>
	<serial>052eeb</serial>
	<time>1459463087</time>
	<priority>2</priority>
	<event id='81' type='Arm. Parcial'/>
	<profile id='3'  type='Open / Close'/>
	<device id='3' type='Usuario'/>
	<location id='1' type='admin'/>
	<zone id='1'/>
	<partition>1</partition>
	<forward_automation>true</forward_automation>
	<userlist>
		<user name='admin' email='ownermailbox@hotmail.com' phone='555
	666 777' is_sms='0' is_mms='0' is_email='1'/>
		<user name='carmen' email='wifemailbox@gmail.com' phone='555 777
	888' is_sms='0' is_mms='0' is_email='1'/>
		<user name='juan' email='neigbourgmailbox@yahoo.com' phone='555
	555 555' is_sms='0' is_mms='0' is_email='1'/>
	</userlist>
</notify>
```

Answer:

```
HTTP/1.0 200 OK\r\n
Content length: 42\r\n
Content-Type: text/xml; charset=UTF-8\r\n
\r\n
<?xml version '1.0'?> <index>16990</index>
```

Desarmado (disarmed?)

Call

```
POST /scripts/notify.php HTTP/1.1\r\n
Host: 212.179.58.186:8080\r\n
Accept: */*\r\n
Content-Type: text/xml\r\n
User_Agent: persist/1.0\r\n
Connection: keep-alive\r\n
Content length: 746\r\n
\r\n
<?xml version='1.0'?>
<notify>
	<pmax_account>001234</pmax_account>
	<index>16988</index>
	<serial>052eeb</serial>
	<time>1459461082</time>
	<priority>2</priority>
	<event id='85' type='Desarmado'/>
	<profile id='3'  type='Open / Close'/>
	<device id='3' type='Usuario'/>
	<location id='1' type='admin'/>
	<zone id='1'/>
	<partition>1</partition>
	<forward_automation>true</forward_automation>
	<userlist>
		<user name='admin' email='ownermailbox@hotmail.com' phone='555 666 777'
		is_sms='0' is_mms='0' is_email='1'/>
		<user name='carmen' email='wifemailbox@gmail.com' phone='555 777 888'
		is_sms='0' is_mms='0' is_email='1'/>
		<user name='juan' email='neigbourgmailbox@yahoo.com' phone='555 555 555'
		is_sms='0' is_mms='0' is_email='1'/>
	</userlist>
</notify>
```

Answer:

```
HTTP/1.0 200 OK\r\n
Content length: 42\r\n
Content-Type: text/xml; charset=UTF-8\r\n
\r\n
<?xml version '1.0'?> <index>16988</index>
```

Step by step guide for notification solution of figure 3

I will assume that you already has a computer configured with Linux, e.g. a Raspberry Pi with Raspbian, with a local IP (e.g. 192.168.1.100).

Step 1 Get an email account in a free email service

This account will be used to send mail notification messages (field from:).

I am going to suppose that this account will be *myalarm@gmail.com.* (En el caso de hotmail se comprueba que va muy lento enviando correos entre cuentas de este mismo proveedor).

Once created, probably using the webmail interface, we must to test it sending a message to a normally used mailbox, e.g. *ownermailbox@hotmail.com*

In the case of gmail it is necessary to activate "access for less secure applications." This will allow for mailings from the sSMTP client that will be configured in next step.

Step 2. Install and configure sSMTP email client

Install the sSMTP application (https://wiki.archlinux.org/index.php/SSMTP). For this, excute with root privileges:

```
# apt-get  install ssmtp
```

Nota: Previamente es preciso asegurar que no está instalado otro servicio de correo como sendmail o postfix. Por ejemplo para desinstalar sendmail:

```
apt-get remove sendmail
apt-get purge sendmail*
```

Obsérvese la presencia del asterisco (http://askubuntu.com/questions/460820/completely-remove-sendmail)

Then, it is necesary edit sSMTP configuration file:

```
# nano /etc/ssmtp.conf

#
# Config file for sSMTP sendmail
#
# The person who gets all mail for userids < 1000
# Make this empty to disable rewriting.
root=postmaster

# The place where the mail goes. The actual machine name is required no
# MX records are consulted. Commonly mailhosts are named mail.domain.com
# mailhub=mail

# Where will the mail seem to come from?
#rewriteDomain=

# The full hostname
hostname=vcamserver.vigilanet.com

# Are users allowed to set their own From: address?
# YES - Allow the user to specify their own From: address
# NO - Use the system generated From: address
#FromLineOverride=YES

### Para Gmail
AuthUser=myalarm@gmail.com
```

```
AuthPass=MiPassw0rd
FromLineOverride=YES
mailhub=smtp.gmail.com:587
UseSTARTTLS=YES

### Para Hotmail (remove comments)
#AuthUser=myalarm@outlook.com
#AuthPass= MiPassw0rd
#FromLineOverride=YES
#mailhub=smtp.live.com:587
#UseSTARTTLS=YES
```

Now you must test that your ssmtp really send mails using gmail account as origin (MTA):

```
# ssmtp ownermailbox@hotmail.com
subject: Alarma en dormitorio
Prueba de envío de notificación de alarma por correo electrónico
```

End the message with ctrl-D.

Check that in a few seconds the message is received in your private account (*ownermailbox@hotmail.com*) from the alarm account (*myalarm@gmail.com*).

Step 3 Install Apache and PHP

To install Apache and PHP in a Raspberry PI microcomputer you can follow this URL:

https://www.raspberrypi.org/documentation/remote-access/web-server/apache.md

NOTE: It is not necessary install MySQL database engine for this notification service.

After the installation, Apache server must be **configured to listen 8080 TCP** port (normally listen ports are 80 and 443).

http://serverfault.com/questions/28041/how-do-i-make-apache-web-server-listen-on-two-different-ports

Step 4 Install notification server PHP scripts

Create a subdirectory */scripts* in the DocumentRoot directory of Apache server (e.g. */var/www/scripts*). Copy in this directory the scripts files *update.php* y *notify.php.* These files must have execution and read permissions (use chmod command).

Check that script *update.php* work under apache server. Using a web browser **enter the following URL** (changing accordingly de IP address).

```
http://192.168.1.100:8080/scripts/update.php?serial=052eeb&id=052EEB&account=001234&ver_sw=6.1.14&ver_hw=123&ver_var=6005&upgrade_status=0&configuration_status=0
```

If the local notification server is working correctly the web **browser will show:**

```
status=0&ka_time=50&allow=0
```

Checking *notify.php* script is more dificult since it uses a POST command.

Step 6 Modify the PowerMax communication parameter to point new notification server

Do the following

1. From the PowerMax control panel, enter "INSTALLER MODE" menu using the

Master Installer Code.

2. Enter the "5.DEFINE COMM." menu.

3. Enter the "3:C.S.REPORTING" sub-menu.

4. Program the following menus:

- "REPORT EVENTS" – Select the type of events that the control panel will report to the Central Station.
- "1st/2nd/3rd REPORT METHOD" – Define the 1st/2nd/3rd priority of method used to report events. Select the "broadband" option for PowerLink2.
- "IP RCVR 1/2" – Enter the **IP address of new local notification server** acting as Central Station the PowerLink2 will report to (e.g. 192.168.1.100)

Step 7 Test the reception of mails with of event notification

Arm the PowerMax and check the reception of a mail message from new created account of mail service for sending these notifications.

Step 8 Install notification server PHP scripts

After the system is working correctly you can backup script *notify.php* an modify it for your convenience.

Notification server scripts

notify.php

```
<?php
    ###############################################################################################################
    #notify.php   v0.1                                                                                            #
    #Recibe las notificaciones XML sobre HTTP enviadas por modulo Powerlink de la alarma PowerMax Pro             #
    #Receive XML notifications  over HTTP sent from PowerLink module of PowerMax Pro panel                       #
    #autor/author: junav (junav2@hotmail.com)                                                                    #
    #Fecha/Date: 06/04/2016                                                                                       #
    ###############################################################################################################


    ##############################################
    #obtiene documento xml enviado via post     #
    ##############################################
    $xmlinput = new DOMDocument();
    $xmlinput->loadXML(file_get_contents("php://input"));
    $elementos = simplexml_load_string($xmlinput->saveXML());
    $mensajeXML=$xmlinput->saveXML();

    ##################################################
    #crea la respuesta HTTP                          #
    ##################################################
    # se obtiene el valor del tag "index" y se devuelve en la respuesta como un data stream XML
    $index=$elementos->index;
    $xml = new DOMDocument();
    $Element = $xml->createElement("index",$index);
    $xml->appendChild($Element);
    $header = "Content-Type:text/xml";
    header($header);
    print $xml->saveXML();

    ##############################################
    #crea correos a enviar a usuarios           #
    ##############################################
    #pasa el tiempo que viene en el tag "time" como numero de segundos a hora y fecha local
    $date=date_create();
    date_timestamp_set($date,$elementos->time);
    #compone el asunto con la información más relevante
    $subject= $elementos->event['type']." (event ".$elementos->event['id'].") ".date_format($date,"H:i:s d-m-
Y");
    #compone un cuerpo de correo en htnl para poder emplear los enlaces a las cámaras
    $message = '
        <html>
        <head>
          <title>Notificación central alarma</title>
        </head>
        <body>';

    $message .= '<h2>Prioridad:';
    $message .= $elementos->priority;
    $message .= ' Zona:';
    $message .= $elementos->zone['id']." ";
    $message .= '<br>Perfil:';
    $message .= $elementos->profile['type']."(" . $elementos->profile['id'] . ")";
    $message .= '<br>Dispositivo:';
    $message .= $elementos->device['type']."(" . $elementos->device['id'].")";
    $message .= '<br>Origen:';
    $message .= $elementos->location['type'] . "(" . $elementos->location['id'] . ")";
    $message .= '</h2>';
    $message .= '<h2>';
    $message .= $elementos->event['type']." (event ".$elementos->event['id'].") a las
".date_format($date,"H:i:s d-m-Y");
    $message .= '</h2>';
    $message .= '<p>
                 Otros mensajes <br>
                 O enlaces
                 </p>';
    $message .= '<h3>Fuente de datos XML recibida: </h3>';
    $message .= '<p> <![CDATA[';
    $message .=  $mensajeXML;
    $message .=   ']]> </p>';
    $message .=   '
        </body>
        </html>
    ';

    $headers  = 'MIME-Version: 1.0' . "\n";
    $headers .= 'Content-type: text/html; charset=UTF-8' . "\n";
    $headers .= 'From: central alarma' . "\n" .
              'Reply-To: myalarm@gmail.com' . "\n" .
              'X-Mailer: PHP/' . phpversion();
```

```
    ################################################
    #envia correos usuarios indicados por la alarma #
    ################################################
    foreach($elementos->userlist->user as $usuarios) {
        $to=$usuarios[email];
        mail($to, $subject, $message, $headers);
    }
    ################################################
    #envia correos adicionales                     #
    ################################################
    #$to="other_mail@gmail.com";
    #mail($to, $subject, $message, $headers);
?>
```

update.php

```
<?php

    ##########################################################################################################
    #update.php   v0.1                                                                                       #
    #Recibe llamadas periódicas GET de HTTP enviadas por modulo Powerlink de la alarma PowerMax Pro     #
    #Receive periodic calls through GET command of HTTP sent from PowerLink module of PowerMax Pro panel#
    #autor/author: junav (junav2@hotmail.com)                                                               #
    #Fecha/Date: 08/04/2016                                                                                  #
    ##########################################################################################################

    $header = "Content-Type:text/plain; charset=UTF-8";
    header($header);
    print ("status=0&ka_time=50&allow=0&\n");
?>
```

Annex A: WEB resources

http://voksenlia.net/powerlink/

http://www.securitybydefault.com/2012/01/alarma-visonic-powermax-pro-un-estudio.html