Configuring eGauge Alerts (Classic Interface method)

> Firmware version 4.6 and newer can have alerts configured through the Modern Interface.

Overview

The eGauge can be configured to send alerts based on a variety of trigger conditions. Alerts must be configured through the eGauge interface, and the eGauge needs to be powered on and connected to the internet in order to send alerts. There are three possible alert destinations: SMTP (email or SMS-capable phone numbers via an email-to-SMS gateway if the cellular provider supports), the eGuard alert service, or a custom URI for a JSON POST (advanced users).

SMTP emails credentials may be supplied, and the eGauge will use this email account to generate alerts. Some services such as Gmail may restrict logins to browsers, or disallow the login if devices in different locations are attempting to all log in to send alerts. For large larger deployments a service such as SendGrid may be used.

The eGuard Alert Service is a more simplified email alert delivery service and only requires you have an eGauge.net account.

The following article covers basic alert configuration and provides some sample alerts. Additional information is available on an eGauge-specific basis by navigating to http://**DEVNAME**.egaug.es/fundoc.html?alert where **DEVNAME** is the device name of your specific eGauge. To take advantage of all alert features, the eGauge should be on the latest firmware.

> For meters shipped after January 1, 2024 this information may be found at: http://**DEVNAME**.egauge.io/fundoc.html?alert

Contents

Alert Basics

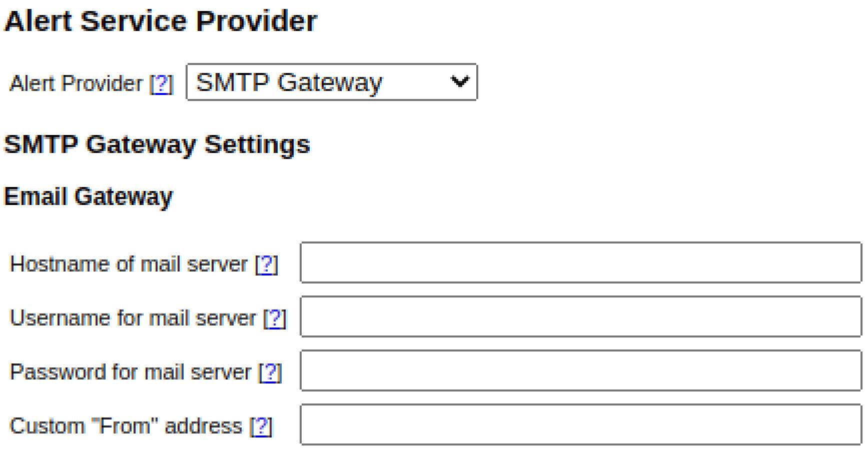
Alerts may be configured from the **Settings → Alerts** page and viewed from **View → Alerts**. There are two types of alerts: system alerts and user-defined alerts. System alerts can report conditions such as when the device configuration is changed or when the connection to a remote device has been established. User-defined alerts are built arbitrary conditions that, when true, trigger the alert. For example, you could define an alert that triggers when solar production for a period is below a certain threshold value, or an alert that triggers when the register monitoring Oven usage has been above a certain value for a certain time. More examples are **available here**.

Configuring the Alert Service Provider

Choose the View/Edit Gateway & Alert Destinations button from the top of the Alerts page to configure how alerts are sent. The page will request credentials in order to make changes if none have been previously cached.

SMTP Gateway

> Legacy meters (eGauge2 and EG30xx series) support TLS 1.1, while newer meters (such as EG4xxx) support TLS 1.2.



Alert Service Provider

Alert Provider [?] SMTP Gateway

SMTP Gateway Settings

Email Gateway

Hostname of mail server [?]

Username for mail server [?]

Password for mail server [?]

Custom "From" address [?]

The SMTP Gateway Alert Service Provider allows the eGauge to send alerts directly to email addresses, SMS-enabled phones, or a mixture of the two. This functionality requires an internet connection, but *does not* require the eGauge to be connected to the proxy server at d.egauge.net. The following fields are required:

Hostname of mail server

Normally, eGauge attempts to deliver email directly to the destination address. Similarly, it attempts to deliver SMS directly to an SMS-gateway. However, if a firewall prevents the device from directly establishing such connections, as is commonly the case for consumer-grade Internet-service, you will have to set the value of this setting to the hostname of a mail server which can forward the messages to the final destination. The mail server may either be a host on the same LAN (e.g., within a company or school network) that will accept email delivery without authentication or it may be am external mail server where you have a valid user account. By specifying the username and password for that account, the device is then able to deliver email through that mail server (ie, the alert messages from the eGauge will originate from your username on that mail server). As an example, if you have a Gmail account, you can set the hostname to smtp.gmail.com. By specifying your Google account’s username and password, you can then have alerts delivered via Gmail.

Username for mail server

When non-empty, this setting specifies the username the device uses to authenticate itself to the mail server. If empty, mail is delivered without authentication. Note that this option is required for almost all mail servers.

Password for mail server

This setting specifies the password the device uses to authenticate itself to the mail server. It is used only if Username is not empty.

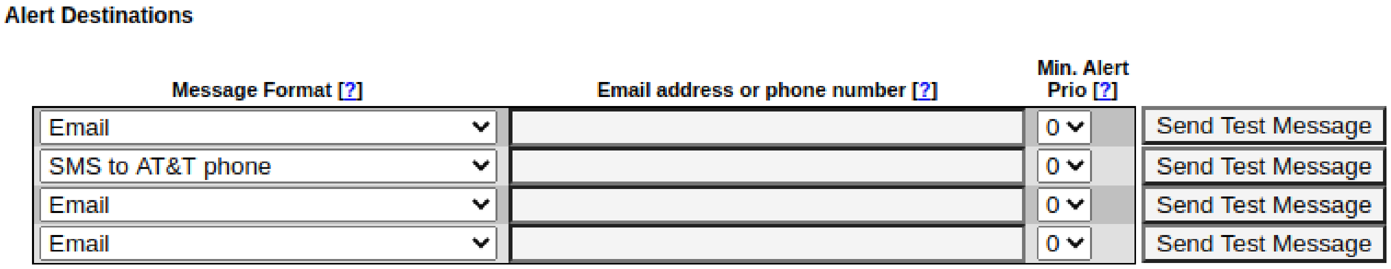
> Caution: on legacy meters (eGauge2 and EG30xx series) the password is transmitted to the eGauge over an unencrypted channel. Only change this password from a computer that's connected to the same LAN as the eGauge and only after clicking on the **LAN Access** link on the eGauge main page. As an added security measure, create a dedicate email account at the mail server for sending eGauge alerts.

Using SendGrid

> Compatibility Notice: Beginning in June 2023, only EG4xxx will be compatible with the SendGrid SMTP gateway. Legacy meters such as EG30xx and eGauge2 will fail to send alerts via the SendGrid SMTP gateway.

SendGrid credentials are entered in the SMTP Alert Service Provider fields. For more information on using SendGrid or a similar service, please refer to this article.

Setting Alert Destinations



Alert Destinations

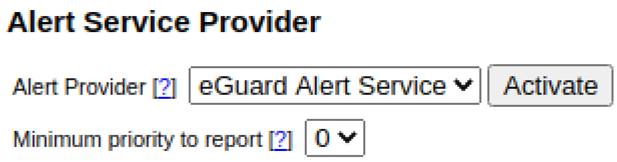
Message Format [?]	Email address or phone number [?]	Min. Alert Prio [?]	
Email		0	Send Test Message
SMS to AT&T phone		0	Send Test Message
Email		0	Send Test Message
Email		0	Send Test Message

Message Format: select the appropriate SMS carrier or email format.
Email address or phone number: enter the appropriate destination for the alert.
Min. Alert Prio (Minimum Alert Priority): minimum level of alerts this destination should receive (see below).

Up to four alert-destinations can be defined. Alerts are prioritized. For each alert-destination, a **minimum priority** can be defined. Only alerts whose priority is equal to or greater than the minimum priority are reported to an alert-destination. Once an alert-destination has been notified, only alerts of higher priority result in a new notification to that destination until the alert has been acknowledged or deleted via the alerts page, or after 24 hours have passed.

eGuard Alert Service

> Legacy meters (eGauge2 and EG30xx) require**HTTPS certificate validation to be disabled** to activate the eGuard Alert Service. This is due to a bug with an older SSL library used on legacy meters and the eGauge.net certificate provider. This should only be used if the alert information being sent is not sensitive.



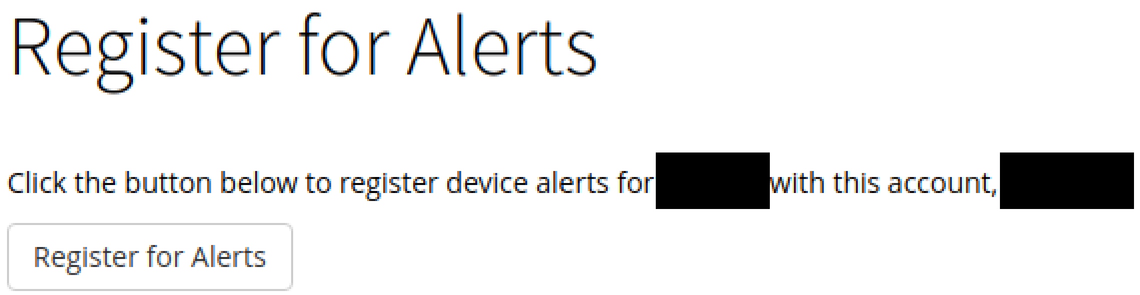
Alert Service Provider

Alert Provider [?] eGuard Alert Service

Minimum priority to report [?] 0

The eGuard alert service provides an alternative to configuring the eGauge with SMTP credentials. This is especially useful for users with a large number of devices. The meter must be in an eGuard group controlled by the user. More information on eGuard is available here. Also note that eGuard features built-in alerts - those are covered in this article.

To use the eGuard Alert Service, simply select "eGuard Alert Service" and click the "Activate" button. A new window will open, and you will be prompted to log in to eGuard. Once logged in (or if you are already logged in), eGuard will confirm you want to register this device for alerts. Click "Register for Alerts" to confirm.

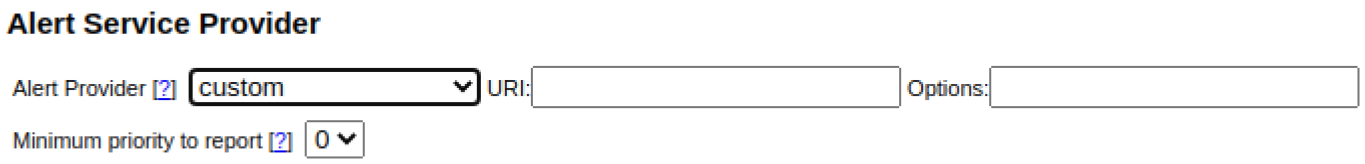


Register for Alerts

Click the button below to register device alerts for [redacted] with this account, [redacted]

Minimum priority to report: Setting this to a value other than zero will omit any alerts with a priority set lower than that value. This can be useful when certain alerts are not required (eg, set all system alerts to zero, set minimum priority to report to 1, then set all user-defined alerts to 1).

Custom Alert Destinations



Alert Service Provider

Alert Provider [?] custom URI: Options:

Minimum priority to report [?] 0

Custom alerts may be utilized by advanced users to send JSON-formatted data as a POST to a user-provided URI (alert destination).

Alert Provider: Must be set to "custom".

URI: The URI to send the JSON POST to. Should be unique to the device, such as with a GET token to uniquely identify the device making the POST.

Options: A comma separated list of options available below:

- deflate: Use "deflate" content-encoding when posting alerts.
- gzip: Use "gzip" content-encoding when posting alerts.
- secure: For HTTPS connections, fail if the alert provider server's certificate cannot be verified as being valid.

> Do not use multiple compression schema, i.e., do not use gzip AND deflate on the same device.

Minimum priority to report: All alerts with a priority level equal to or greater than this will be POSTed to the URI when triggered. To prevent some or all system alerts from being reported, this may be set to "1" or greater. When alerts below the minimum priority level are triggered, they are only logged on the device locally and do not create a POST.

> An example of the JSON post contents is available here.

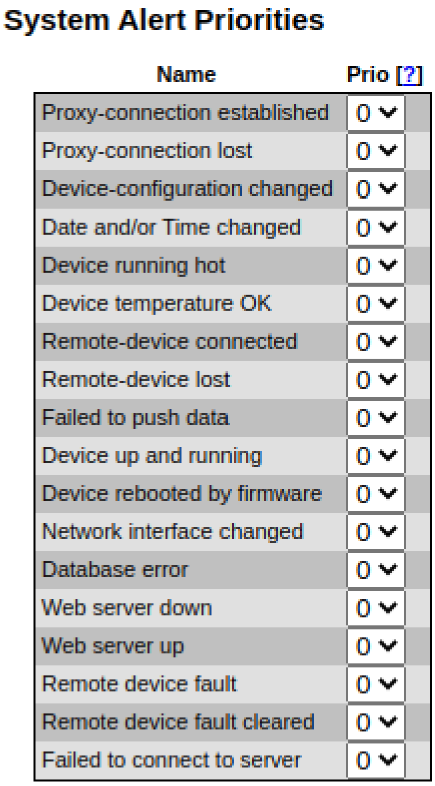
Configuring Alerts

Alerts are reported with a delay of approximately 30 seconds and are automatically acknowledged 24 hours after reporting them. These rules ensure you will be promptly informed of any alert conditions for a device without a deluge of SMS or email messages. Alerts of higher priority are reported even if there are pending alerts of a lower priority.

There are two types of alerts: System alerts and User-defined alerts

System Alerts

System alerts are predefined but you can choose the priority with which they are reported. This allows control over which recipients receives which system alerts (if any) and which alerts are more important. To set an alert priority, use the dropdowns in the "Prio" column. If there are certain system-alerts that you do not wish to have reported at all, select priority 0 and ensure that all alert destinations have a minimum alert priority of at least 1. Note that alerts with a priority of 0 will still be logged on the Alerts page, but no notifications will be sent for those alerts if all alert destination priorities are higher than 1.



System Alert Priorities

Name	Prio [?]
Proxy-connection established	0
Proxy-connection lost	0
Device-configuration changed	0
Date and/or Time changed	0
Device running hot	0
Device temperature OK	0
Remote-device connected	0
Remote-device lost	0
Failed to push data	0
Device up and running	0
Device rebooted by firmware	0
Network interface changed	0
Database error	0
Web server down	0
Web server up	0
Remote device fault	0
Remote device fault cleared	0
Failed to connect to server	0

Proxy-connection established/lost: tracks when a connection to the proxy server at d.egauge.net is opened or closed. If this occurs frequently it can indicate an unstable network connection.

Device-configuration changed: reports when a device’s configuration is changed, and which account has made the modification.

Date and/or Time changed: reports when the device date or time is changed (either by the user or automatically).

Device running hot: reports if the eGauge’s internal temperature reaches a significantly high temperature.

Device temperature OK: reports when the eGauge’s temperature returns to a safe level.

Remote-device connected: tracks when a connection is established to a remote device (including Modbus devices and remote eGauges).

Remote-device lost: tracks when a connection to a remote device is lost (including Modbus devices and remote eGauges).

Failed to push data: reports if a data push is set, and the eGauge is unable to successfully push data.

Device up and running/Device rebooted by firmware: tracks when the meter is rebooted, and when the meter comes back online from a reboot or power outage.

Network interface changed: tracks when the meter switches from an Ethernet (ETH) to HomePlug (PLC) connection. This may happen immediately after a reboot and can generally be ignored.

Database error: typically reports when the device configuration is changed. The occasional database error is considered normal, but if this alert triggers multiple times per day and no configuration changes are being made it may indicate an issue. If this happens, contact eGauge support at support@egauge.net.

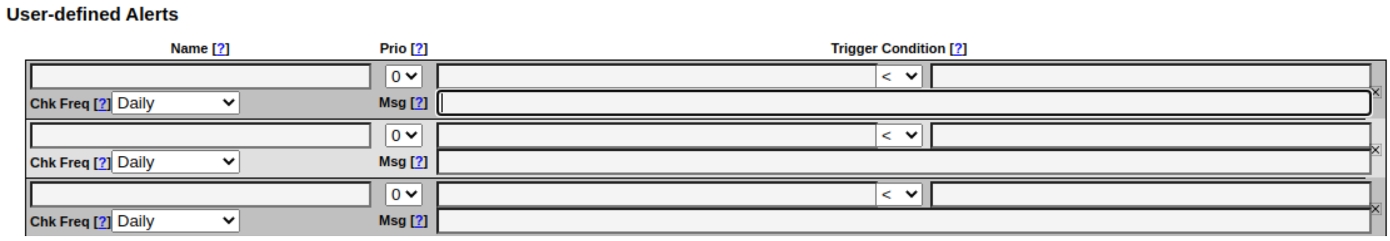
Web server down/Web server up: reports when the eGauge's internal webserver stops and starts. This will generally happen as the result of a reboot, and may happen as a normal occurrence during regular operation (for example, watchdog resets). If this alert triggers multiple times per day for several days it may indicate an issue.

Remote device fault/Remote device fault cleared: reports when Sunspec fault codes trigger on a remote Modbus device, and when those fault codes are cleared.

Failed to connect to server: reports when certain outbound connections fail, typically alert POSTs when using a custom alert destination.

User Defined Alerts

User-defined alert patterns allow the flexible detection and reporting of various conditions. For example, an alert could be defined which, on a second-by-second basis, checks whether a register value is outside of its permitted range (e.g., whether a voltage or frequency is above or below a certain threshold).



User-defined Alerts

Name [?]	Prio [?]	Trigger Condition [?]		
	0		<	
Chk Freq [?] Daily	Msg [?]			
	0		<	
Chk Freq [?] Daily	Msg [?]			
	0		<	
Chk Freq [?] Daily	Msg [?]			

The alert fields are described below:

Name: The name of the alert. This should be short but informative enough to convey the nature of the alert.

Trigger Condition: The trigger condition consists of three parts: left-hand-side (lhs), comparison operator, and right-hand-side (rhs). The comparison operator may be one of less-than (`<`), less-or-equal (`<=`), equal (`=`), not-equal (`!=`), greater-orequal (`>=`), or greater-than (`>`). The lhs is compared to the rhs based on this operator and, if true, the alert is triggered.

Chk Freq (Check Frequency): select the frequency with which the trigger condition is to be checked. eGauge evaluates all alert conditions whenever the device starts up and hence may evaluate the conditions more frequently than requested. Apart from the first time a condition is checked on start up, hourly conditions are evaluated during the first minute of each hour, daily conditions during the first hour after midnight, weekly conditions during the first hour of Sunday, monthly conditions during the first hour of the first day of the month, and annual conditions during the first hour of the first day of the year. “Every second” conditions are evaluated each second, “Every minute” conditions once a minute.

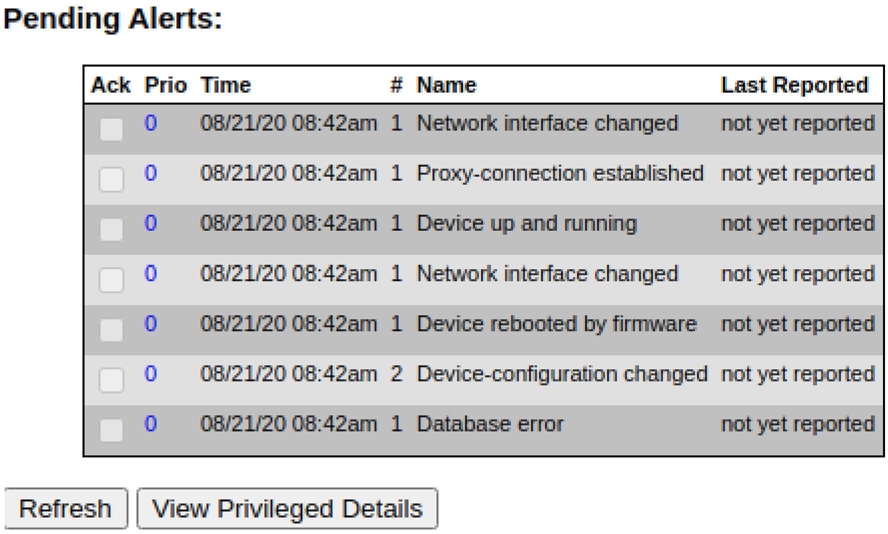
> Choose the lowest check frequency possible as evaluating too many conditions too often may slow down the device. If a slow-running condition (eg, a condition using the peak_risk() function) is evaluated, evaluation of other conditions may be delayed until the evaluation of that condition is completed.

Msg (Message): use this field to define a custom-message to be displayed along with the alert name. If left empty, a default message is included which shows the value of the lhs, the operator, and the rhs of the trigger-condition. A well-written message will explain the alert - for example, on a "Low Production" alert the message might be "Caution: Low production on Inverter 1 (north side)". The placeholders `%l` and `%r` can be used in the message field to include the calculated value for the lhs and rhs of the equation.

> Examples of user-defined alerts are available in the User-defined Alert Examples section near the end of this document. Click here to jump to that section.

Viewing and Acknowledging Alerts

You can view and acknowledge alerts on your device under **View → Alerts**. By default, a list of triggered alerts will be visible. For more information on each alert and the option to acknowledge or delete an alert, click the "View Privileged Details" button.



Pending Alerts:

Ack	Prio	Time	#	Name	Last Reported
☐	0	08/21/20 08:42am	1	Network interface changed	not yet reported
☐	0	08/21/20 08:42am	1	Proxy-connection established	not yet reported
☐	0	08/21/20 08:42am	1	Device up and running	not yet reported
☐	0	08/21/20 08:42am	1	Network interface changed	not yet reported
☐	0	08/21/20 08:42am	1	Device rebooted by firmware	not yet reported
☐	0	08/21/20 08:42am	2	Device-configuration changed	not yet reported
☐	0	08/21/20 08:42am	1	Database error	not yet reported

Ack (Acknowledged): indicates if this alert has been acknowledged. Once acknowledged, the alert will be reported again should it reoccur and its priority is sufficiently high. Alerts are automatically acknowledged after 24 hours. To ensure new alerts are reported, alerts should be acknowledged when they are received.

Prio (Priority): the priority of the corresponding alert.

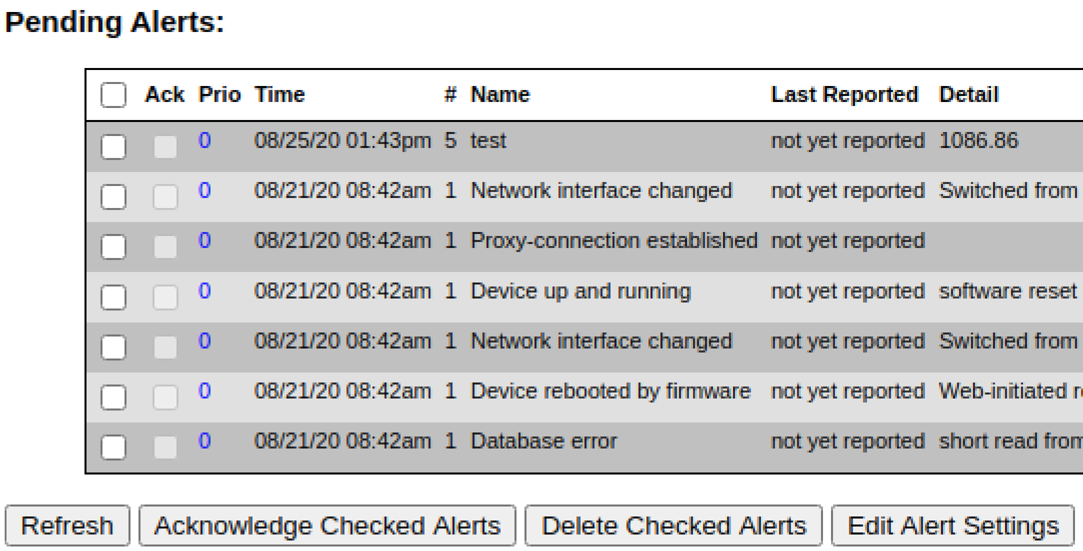
Time: date and time of the most recent occurrence of the alert.

#: number of times this alert has occurred (note that this isn't necessarily the number of times the alert has occurred since the device was installed).

Name: name of the alert.

Last Reported: date and time when the alert was last reported to at least one of the alert-destinations.

To view detailed alert information as well as acknowledge and delete reported alerts, click the "View Privileged Details" button. Valid credentials are required to see this information and acknowledge/delete alerts.



Pending Alerts:

□	Ack	Prio	Time	#	Name	Last Reported	Detail
□	□	0	08/25/20 01:43pm	5	test	not yet reported	1086.86
□	□	0	08/21/20 08:42am	1	Network interface changed	not yet reported	Switched from eth0 to qca0.
□	□	0	08/21/20 08:42am	1	Proxy-connection established	not yet reported	
□	□	0	08/21/20 08:42am	1	Device up and running	not yet reported	software reset
□	□	0	08/21/20 08:42am	1	Network interface changed	not yet reported	Switched from none to eth0.
□	□	0	08/21/20 08:42am	1	Device rebooted by firmware	not yet reported	Web-initiated reboot
□	□	0	08/21/20 08:42am	1	Database error	not yet reported	short read from epoch.dat (136 < 152)

The **Detail** column contains additional information about each alert. For example, one instance of the "Network interface changed" alert provides the additional detail that the network interface was changed from "none" to "eth0" (this happens immediately after a reboot) while the other (newer) instance of the "Network interface changed" alert provides the additional detail that the network interface changed from eth0 to qca0. This happened very quickly (within the same minute), and is normal behavior for a meter coming back online after a reboot.

To modify alerts, check off any alerts you wish to delete or acknowledge, and click the appropriate button. Deleting alerts here
will remove them from the reported alert page until it occurs again.

User-defined Alert Examples

> For available functions on your particular firmware version, visit http://**DEVNAME** /fundoc.html?alert where DEVNAME is your eGauge device name.

General Notes

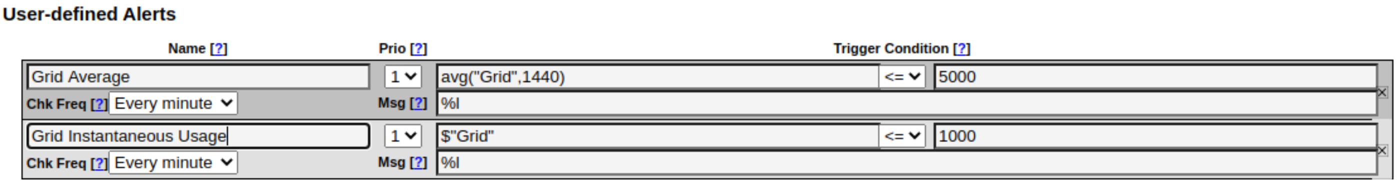
`$"REG NAME"` returns the instantaneous value of the register REG NAME, while `"REG NAME"` points a function at a specific register (but doesn't necessarily return the register's value). When using functions such as avg() or others listed in the function documentation, **do not** include the dollar sign.

When creating a message (Msg), there are several shortcuts which can be used to include values from the alert condition itself:

`%l` will return the value of the left–hand–side
`%L` will return the formula of the left–hand–side
`%r` will return the value of the right–hand–side
`%R` will return the formula of the right–hand–side
`%%` will return a single percent sign (eg, %l %% would read as <value from left side of the comparison> %)

Basic Examples

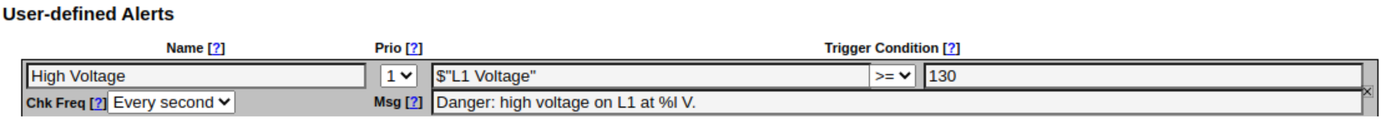
In the following example, "Grid Average" will return the daily average of the Grid register if that value is less than or equal to 5000W, while "Grid Instantaneous Usage" will return the instantaneous reading of the Grid register if that value is less than or equal to 1000W.



User-defined Alerts

Name [?]	Prio [?]	Trigger Condition [?]		
Grid Average	1	avg("Grid",1440)	<=	5000
Chk Freq [?] Every minute	Msg [?]	%l		
Grid Instantaneous Usage	1	$"Grid"	<=	1000
Chk Freq [?] Every minute	Msg [?]	%l		

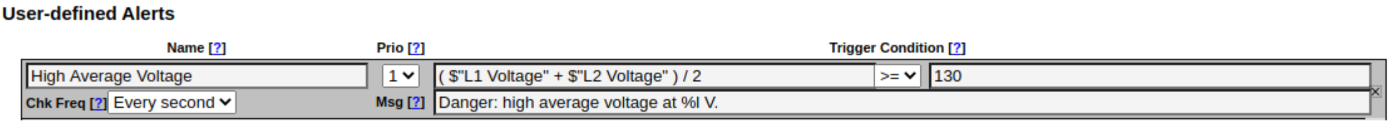
The next example will trigger if the value of the L1 voltage register is greater than or equal to 130V (which could indicate a dangerous condition for devices connected to that service).



User-defined Alerts

Name [?]	Prio [?]	Trigger Condition [?]		
High Voltage	1	$"L1 Voltage"	>=	130
Chk Freq [?] Every second	Msg [?]	Danger: high voltage on L1 at %l V.		

More complex math can also be performed on either side of the alert expression. For example, the following alert obtains the average voltage from two references, and triggers if that value is greater than or equal to 130V.



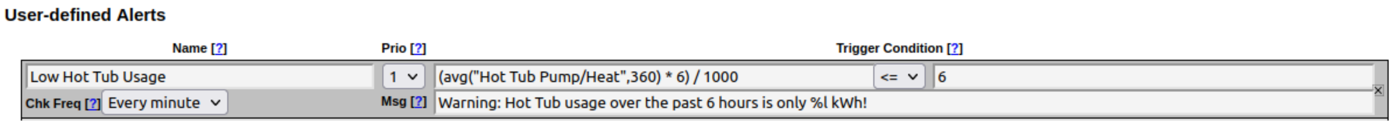
User-defined Alerts

Name [?]	Prio [?]	Trigger Condition [?]		
High Average Voltage	1	($"L1 Voltage" + $"L2 Voltage") / 2	>=	130
Chk Freq [?] Every second	Msg [?]	Danger: high average voltage at %l V.		

It's also possible to calculate cumulative values (kWh) over a period and trigger an alert based on those values. In the following example, let's assume an outdoor hot tub has a 6kW pump/heater, which cycles every 3 hours for 30 minutes at a time. Thus, every 6 hours there should be 6 kWh of energy used. Any less could indicate a pump or heater failure, and the hot tub could freeze.

`(avg("Hot Tub Pump/Heat",360)*6) / 1000` will take the average power (W) read on the register Hot Tub Pump/Heat over the last 360 minutes (60 minutes in an hour, 6 hours). Then, the average power is multiplied by 6 hours to get Wh, the total energy used over the 6 hour period. Finally, we divide by 1000 to convert Wh to kWh.

That value is then compared to the value on the right-hand-side, in this case 6. If the alert is triggered the alert will be sent.



Low Hot Tub Usage | 1 | (avg("Hot Tub Pump/Heat",360) * 6) / 1000 | <= | 6

Chk Freq [?] Every minute | Msg [?] Warning: Hot Tub usage over the past 6 hours is only %l kWh!

Ternary operator

The syntax of the “?” ternary operator (also referred to as a conditional or conditional test) is `condition?value_if_true:value_if_false` and can be nested. This is a fundamental component of many alerts, especially more complex alerts.

Boolean expressions

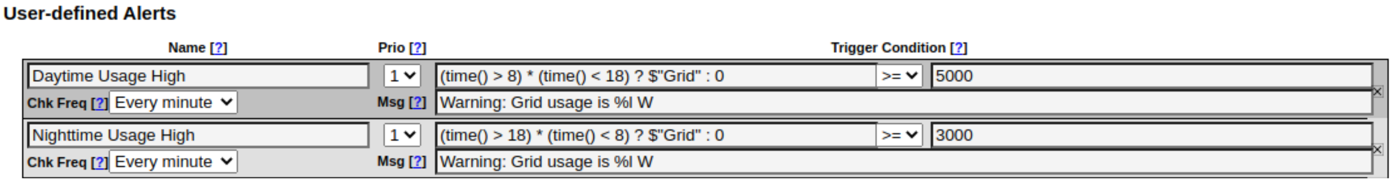
Simple boolean expressions may be used within an alert:

`(5 > 4)` will return 1. Conversely, `(5 < 4)` will return 0.

The boolean value can be multiplied by another value (including a register value). For example, `($"Grid" < 7000) * $"Grid"` returns the value of “Grid” if “Grid” is greater than 7000 W, and returns 0 if the value of "Grid" is less than 7000W.

To break this down: if `$"Grid" < 7000` is true, it will return a 1. `1 * $"Grid"` returns the value for the "Grid" register. If `$"Grid" < 7000` is false, it will return a 0. `0 * $"Grid"` is 0. Remember, `$"REGNAME"` returns the instantaneous value of the register.

Let's look at how the `time()` function can be used with the ternary operator and boolean expressions to trigger an alert at a specific time:



Daytime Usage High | 1 | (time() > 8) * (time() < 18) ? $"Grid" : 0 | >= | 5000

Chk Freq [?] Every minute | Msg [?] Warning: Grid usage is %l W

Nighttime Usage High | 1 | (time() > 18) * (time() < 8) ? $"Grid" : 0 | >= | 3000

Chk Freq [?] Every minute | Msg [?] Warning: Grid usage is %l W

These two alerts work together to trigger if the Grid value is greater than 5000W during daytime hours and greater than 3000 during nighttime hours.

time() returns the current time as a number from 0 up to (but not including) 24 with minutes as a fractional value. For example, 11:30am would be 11.5. We use two booleans here:

`time() > 8 * time() < 18`

If the time is > 8 (8am) and less than 18 (6pm), the booleans work out to 1 * 1 or 1.

If either boolean is false, the output from the booleans is 0. 0 * 1 or 1 * 0 both equal 0.

This gives us a ternary expression of either `1 ? $"Grid" : 0` or `0 ? $"Grid" : 0` (remember ternary expressions work out as `condition?value_if_true:value_if_false`). Thus, if the booleans evaluate to 1, the left side of the formula returns the value of $"Grid". If the booleans evaluate to 0, the left side of the formula returns 0.

Moving on to the alert expression: if the booleans work out to zero (ie, if the time range is not correct), the left side of the alert returns 0. This can never be greater than 5000, so the alert never triggers. If the booleans work out to 1, the left side of the alert returns the value of the "Grid" register. If the value of the "Grid" register is >= 5000, the alert triggers.

Example and description of POST data

```
{
  "now": "1568419537.35",
  "alerts": [
    {
      "id": 1804290019,
      "priority": 7,
      "occurrences": 12,
      "first_occurence": 4462.5,
      "last_occurence": 389.31,
      "name": "Device-configuration changed",
      "detail": "By owner."
    },
    {
      "id": 1804290035,
      "priority": 0,
      "occurrences": 1,
      "first_occurence": 0,
      "last_occurence": 0,
      "name": "Device rebooted by firmware",
      "detail": "Howdy do?"
    }
  ]
}
```

- `now` is a 64-bit UNIX timestamp, possibly with a fractional (sub-second) part, formatted as a decimal integer string.
- `alerts` is a list of reported alerts:
- `id` is a number that uniquely identifies an alert. It is used, for example, to acknowledge or clear an alert.
- `priority` is the user-assigned priority level of the alert (0 being the lowest priority and 7 the highest priority).
- `occurrences` gives a count of how many times the alert has occurred since it was last cleared.
- `first_occurence` and `last_occurence` specify how many seconds ago the alert occurred for the first time and the last time, respectively, relative to `now`. That is, the specified number should be subtracted from `now` to get the absolute UNIX timestamp of when the alert occurred first and last, respectively.
- `name` is the name of the alert that occurred.
- `detail` provides additional detail on the alert that occurred.

Please visit kb.egauge.net for the most up-to-date documentation.