NOTICE TILE FOR CRESTRON HOME

Revision: 1.5
Date: 30 July 2026

SUPPORTED INSTALLATION

This is a platform driver for Crestron Home processors on OS 4 and above only.

MODULE DESCRIPTION

The driver is designed to act as a messaging centre for notifications, alerts and other messages natively via Crestron programmable actions but also through HTTP, TCP and UDP listeners for external messages. A simple API is provided to program events to create notices within the tile.

Features include:

- Manage up to 100 notices per tile, sorted by date or priority level.
- View read/unread status for notices and mark them as read.
- Dismiss notices individually or in bulk, with configurable auto-dismissal.
- Custom button actions that can trigger up to 10 custom events per tile.
- Notice priorities (low/medium/high) to allow for conditional triggers.
- Sensitive notices that require a PIN to be entered for secure dismissal.
- Integration with Crestron Home audio interrupts and chimes, with the ability to conditionally trigger interrupts based on a notice priority level.
- Programmable events triggered on incoming notices or read/dismiss all.
- Integrates with Pushover to forward incoming notices as mobile push notifications.

Disclaimer: The driver supports communications via LAN from devices external to Crestron but does NOT secure internet-facing HTTP by default. If implemented, all security—such as secure port forwarding—is the integrator's responsibility. The driver is a simple user notification system and NOT for emergency use such as a fire or security alerts. Ultamation accepts no responsibility for such incidents.

The driver does not require an active Internet connection to function, but licensing is activated initially via a cloud connection. See 'Licencing' below for more details.

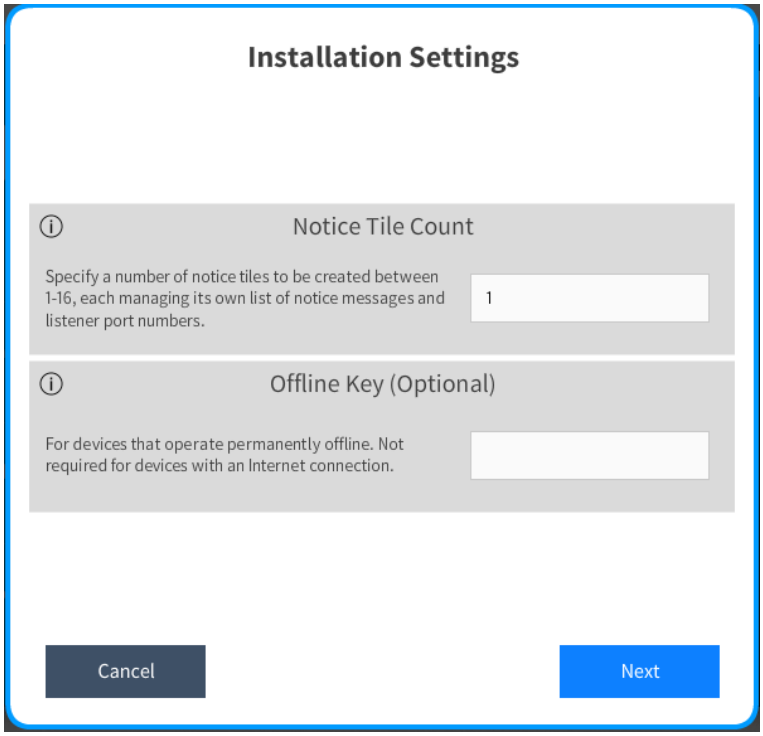
Contents

INSTALLATION

The driver can be found under Drivers > Platform > Ultamation > Notice Tile

Platform Device

The platform manages the devices and can be added to a hidden "Equipment" room. Please ensure that you only add one platform into the system.



Notice Tile Count

If you require more than one Notice Tile device, you may specify between 1 to 16. Each will manage its own individual list of notifications and have its own listener port numbers.

Ensure that all Notice Tiles in the system are using unique port numbers.

If your system requires multiple tiles that share notifications, such as a 'global' notification to all tiles, we recommend using our Wildcard listener as a proxy that receives the notice and passes it to the tiles via the 'Add Notice Message' programmable action (see 'Adding Notice Messages' below for more information).

Offline Key

The driver is licenced via the cloud, but an optional 'offline' key can be request under special circumstances. Please FIRST refer to the 'Licencing' section below and then contact support@ultamation.com if you REQUIRE offline activation. The same offline key can be used for multiple driver instances in the same system.

If you do not have a licence, the system will function for ONE hour and then control will be suspended. The offline licence key can be entered at any time after initial setup without having to reconfigure each tile.

Pushover Credentials

The Pushover credentials allow incoming notice messages to send push notifications.

If you are using exactly one Notice Tile device, or you'd like all Notice Tiles to share the same Pushover credentials, you will always want to enter them in the platform. This means that all Notice Tile devices will send push notifications through the platform, unless explicitly overridden.

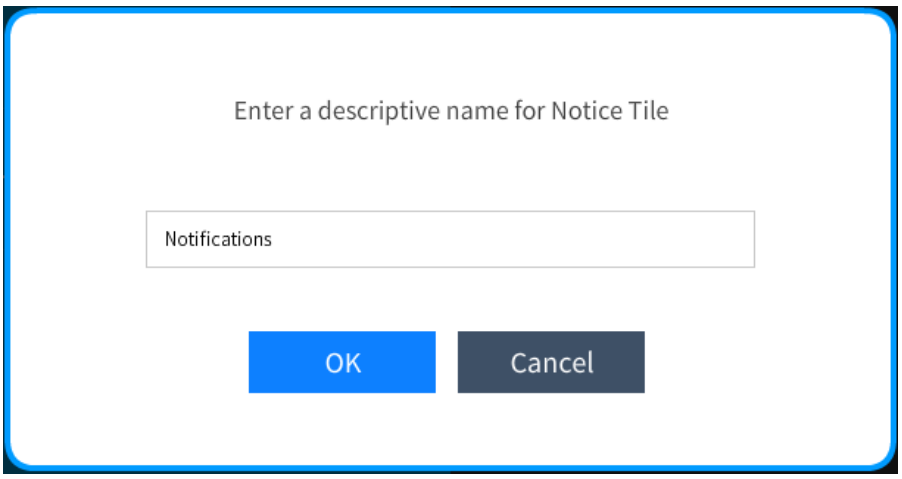
Please see "Configure Push Notifications" for the details on integrating Pushover with the Notice Tile.

Notice Tile Device

Once the platform is added the devices are added to 'Managed Platforms' under the name of the platform.

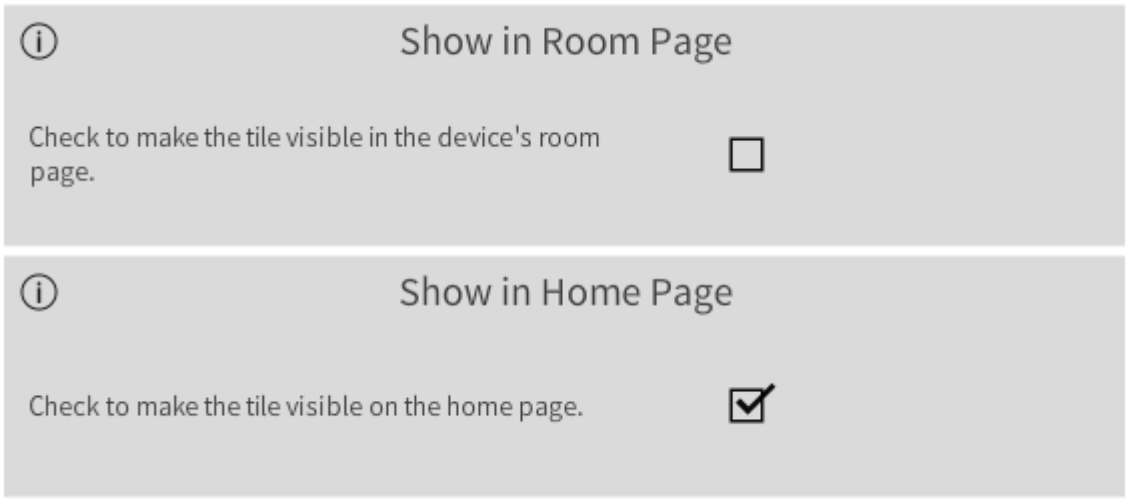
The 'Notice Tile' is the device which receives, stores and displays notices. You may choose to display this tile on the home, the room or both.

When adding to a room, the name of the device can be anything you like, but 'Notifications' works great for the intended use case.



Tile Visibility

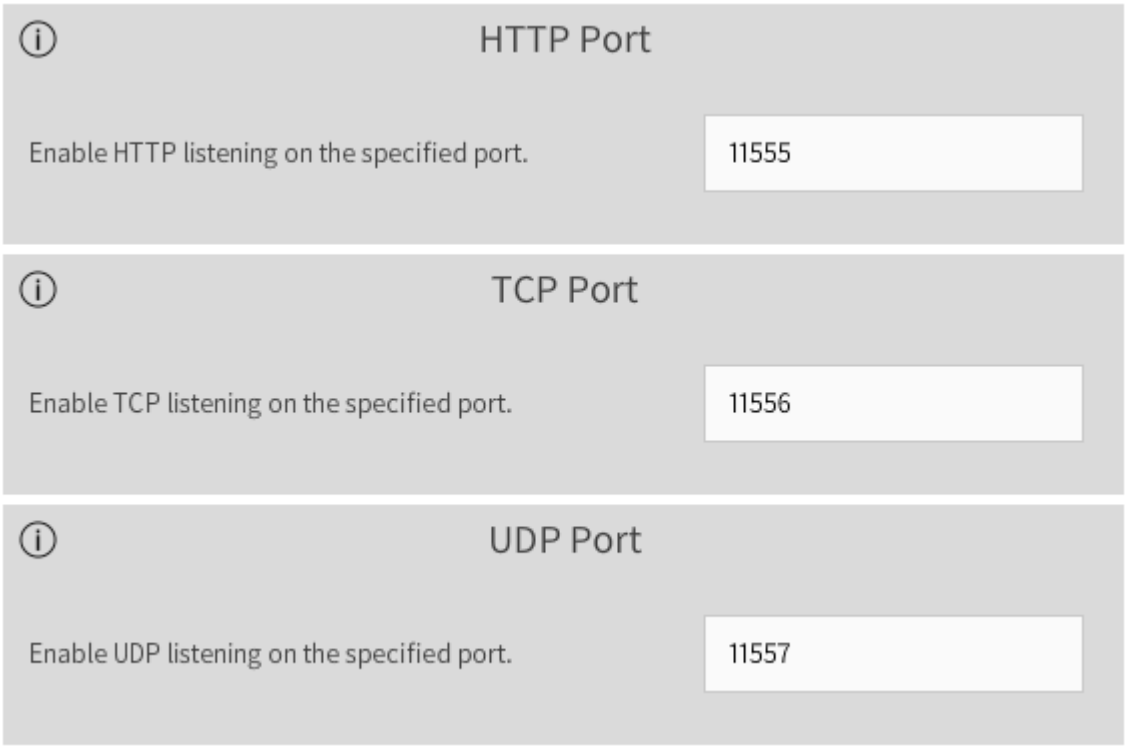
The first installation step is related to where you have added the device as mentioned above. Enabling each checkbox will toggle the visibility in the desired area.



Listener Ports

The next step is to provide the listener ports if intending to send messages via HTTP, TCP or UDP.

The port is crucially important, and you should select a port that is not used by any other service on the Crestron Processor. The default ports we have provided work in most circumstances. This port should be noted down as this is part of the address you will need for whatever service is sending a request. If you wish to not use a listener, leave it blank.



On a successful port assignment, the listeners will be active. If you wish to stop a listener at any point simply blank out the port number.

If you need to verify that the listeners are running or stopped, please refer to the Troubleshooting section at the end of the document.

User Configuration

The final installation step presents items related to optional user configuration.

- Sensitive Dismiss Pin: Enable 'sensitive' notices by providing a 4-digit PIN the user must enter to dismiss. If left blank, incoming notices will never be marked as sensitive. See 'Sensitive Dismiss' below for how this works.
- Enable Interrupts: If you intend on using the interrupts feature, enable this checkbox to present interrupts settings to the user (see 'User Settings' below), otherwise please uncheck.

Pushover Credentials

The Pushover credentials allow incoming notice messages to send push notifications.

Credentials entered here in the Notice Tile device will override credentials that are set in the platform device.

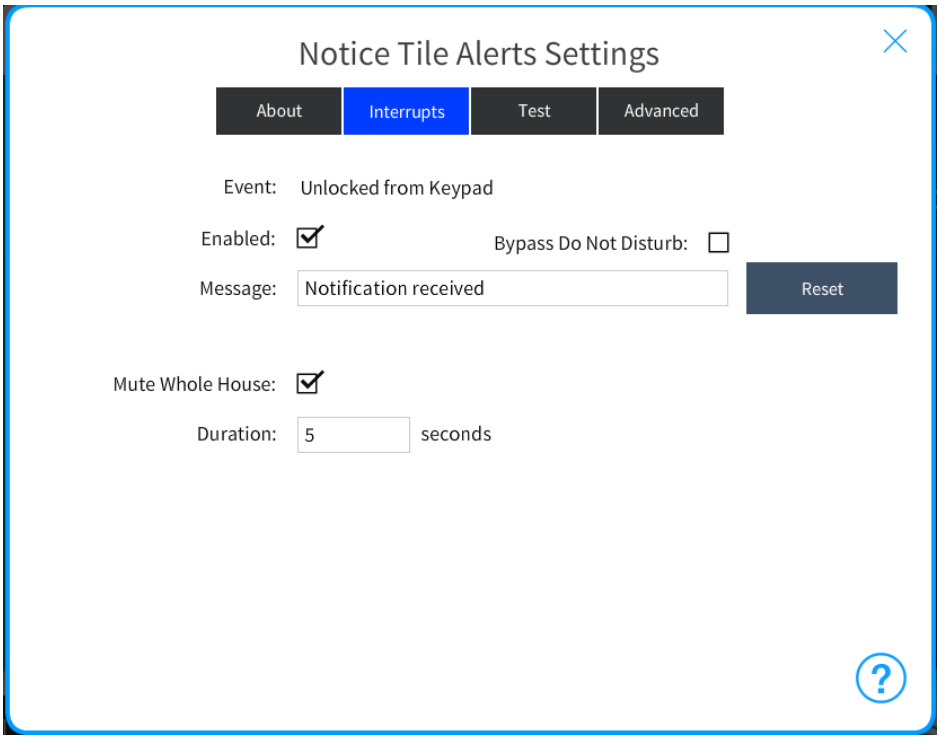
Please see "Configure Push Notifications" for the details on integrating Pushover with the Notice Tile.

Alerts Device (Interrupts)

A second device 'Notice Tile Alerts' is provided under the 'Managed Platforms' folder.

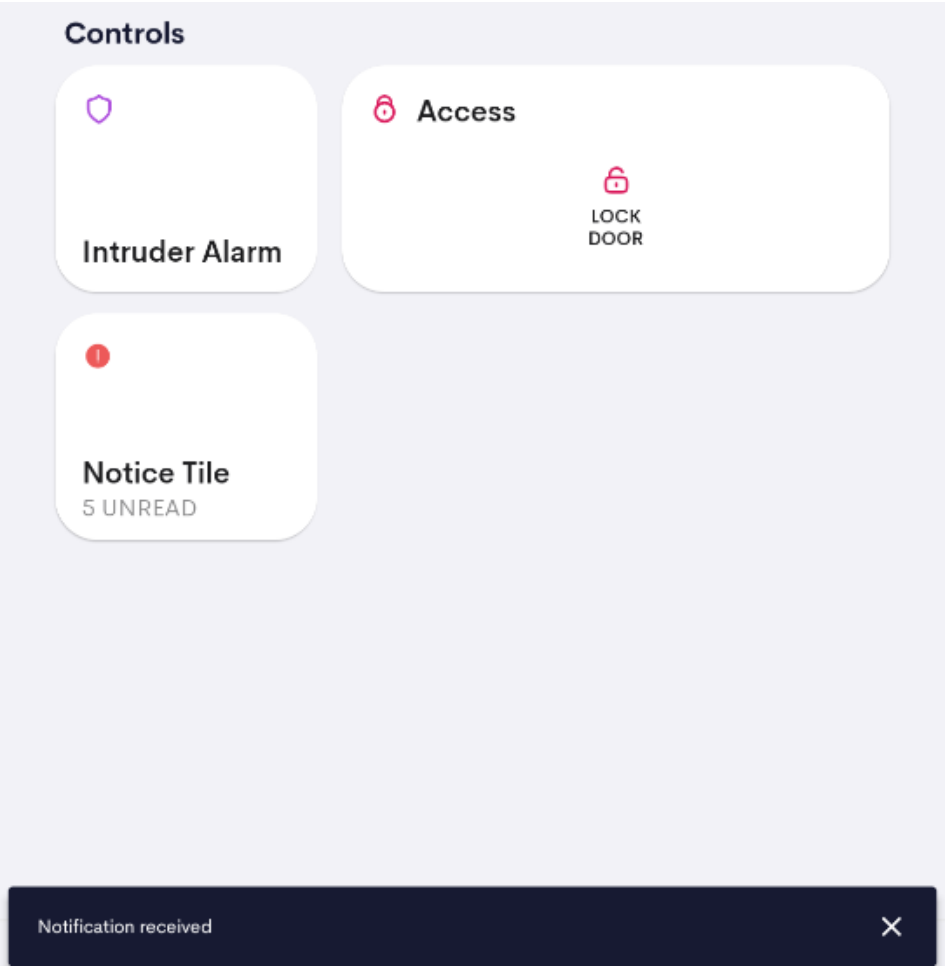
This is a lock device type, which is the way we are currently able to trigger simple interrupts. This should be added to a hidden 'Equipment' room to prevent the device being shown to the user.

The message you provide will be shown when the interrupt is triggered.



Interrupts can trigger an audible chime from a touch panel, or when the Crestron Home mobile app is open. We recommend interrupts to be configured with Crestron Home's do not disturb settings. The user also has settings to control which priority levels trigger interrupts and can enable/disable them at any time directly from the Notice Tile.

When a notice is received the interrupt also appears within Crestron Home as shown.



Change the Chime Sound

The chime can be adjusted as shown by the Crestron Home documentation under 'Audio Interrupts and Chimes':

https://docs.crestron.com/en-us/8525/Content/CP4R/Operation/Settings/Settings-Audio-Interrupts-Chimes.htm

For the Notice Tile, "Interrupt 3" is our personal favourite!

CREATING TRANSLATION FILES

To create a translation file, connect to the Crestron Home processor using an FTP client and navigate to user/Data/UsedThirdPartyDrivers. There should be a folder named "ultamation.noticetileplatform.cloud.ultamationlimited", which will contain one or more folders with names corresponding to version number of the driver.

Open the folder with the version currently in use. Inside, open the translations folder, and copy the en-US.json file to your PC. You can then change the translation values and save the json file as a new json file with the correct language code as the name.

For example, to translate the driver to French (France), the following steps would be taken:

- Open the translations folder and copy en-US.json.
- Translate keys to desired language, changing Priority to Priorité, etc.
- Save the file using the language code as the filename e,g. fr.json.

Now, copy the new json file to the same folder that the en-US.json file came from, and repeat for all languages you wish to create translations for. Reboot the processor, and the translations should then be usable.

Language codes and region codes

Filenames must use a standard code, either:

- Language only – fr.json, de.json, es.json
- Language and region – fr-FR.json, fr-CA.json, de-CH.json

If the translations don't show, and the device running Crestron Home is set to French, please check that the file is named correctly.

Configure Push Notifications

To enable push notification support with the Notice Tile, we have integrated with a service called Pushover, which can be found at https://pushover.net/, a reputable service with support for iOS, Android and desktop. Please note this is a paid service, see https://pushover.net/licensing for information on further costs.

The following information serves as a guide but does not act as a replacement for the Pushover documentation. Please see https://pushover.net/api#registration for more detailed information on how to register your application and users.

Pushover Account

The first step is to create a Pushover account for the integrator or homeowner. How you decide to configure your account is up to you, and the following guide is to show how to retrieve the information you need to integrate with the Notice Tile.

Regardless of which entity owns the account, the initial account will be the "Root User" and will be referred to as such in the following information.

Each subsequent user who wishes to receive push notifications to their devices will need to have their own Pushover account and install the application on their device. We will connect those users to the root account later.

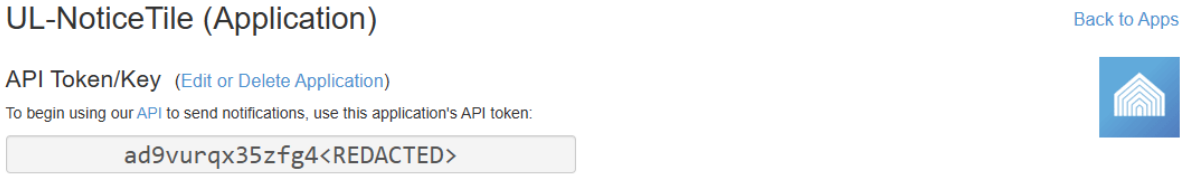
Application API Key

From the root user's account, once logged in, you'll need to create a new application. Each application can send 10,000 free messages per month before needing to upgrade.

Navigate to the dashboard and scroll to the bottom to the "Your Applications" section and click "Create an Application/API Token" (or navigate to https://pushover.net/apps/build) where you'll be prompted with a form to provide the following fields:

- **Name:** This should be named something relevant to the user, such as "Home Notifications" or if used for more specific cases such as "Sump Pump".
- **Description:** Optional but should likely extend the name provided above.
- **URL:** Not required.
- **Icon:** The icon shown to the user when they receive push notifications.

Once your application is created, you will be provided with the Application API Key.

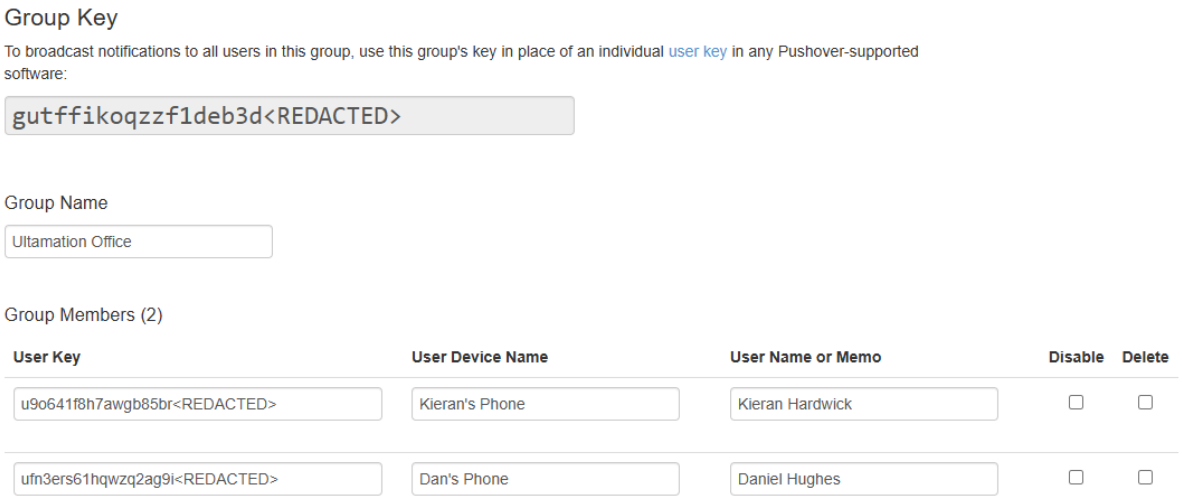


User Key

A delivery group allows multiple devices to receive the same messages. If there is only ever one registered device you may skip this step and use that individual devices "User Key" instead – but otherwise, to register multiple devices, continue reading.

Create a delivery group from the Pushover dashboard or navigate directly to https://pushover.net/groups/build. The form asks that you provide the User Key, which is found within the Pushover client application's settings.

You can edit this group later to add more user keys if necessary. Once all user keys have been added click "Create Delivery Group", and you will be provided with a "Group Key", which you shall now use as the "User Key" installer setting.



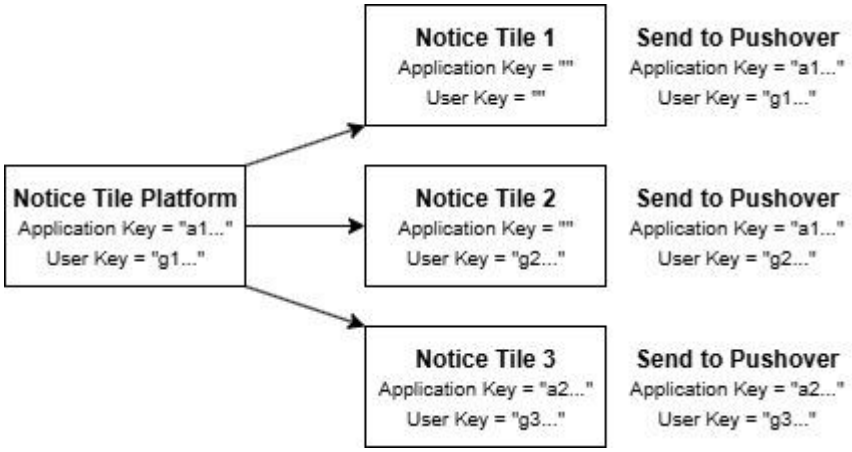
Pushover Installer Settings

You can enter your Pushover credentials in the platform, in a Notice Tile, or in both.

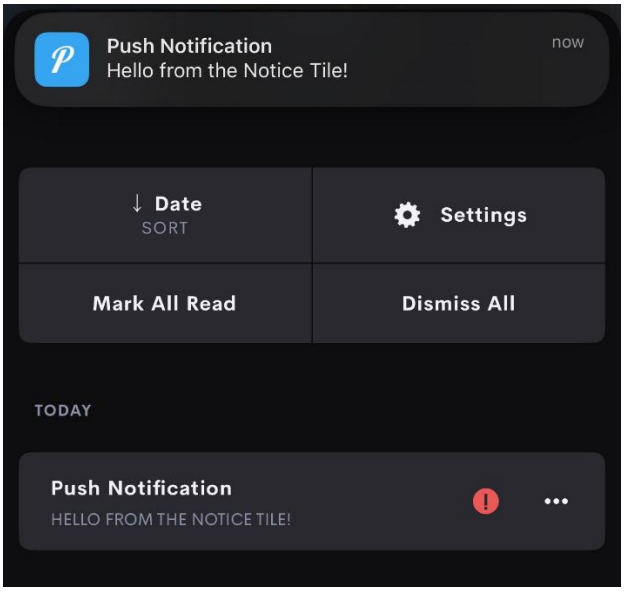
The platform serves as the root device and holds the default (or "global") credentials. By default, each Notice Tile device you create will use these platform-level credentials to send push notifications, unless you specifically override them.

If you want a specific Notice Tile to send notifications using different credentials, you can provide them directly in that Notice Tile device. These will override the corresponding credentials from the platform on a one-to-one basis, meaning you can override each credential individually (e.g., just the API key or user key, without affecting the others).

The diagram below shows how this process looks for a system with multiple Notice Tiles.



Once the Pushover credentials are correctly configured, new notices will send push notifications to the group's users.



ADDING NOTICE MESSAGES

The notice message content and suitability are at the discretion of the integrator. For recommended formatting, Google Material Design offers guidance for notification content: https://m3.material.io/foundations/content-design/notifications.

Message Schema

The general schema for all notice messages, regardless of protocol.

Field	Description	Requirement
Title	Title of the notice	1-30 characters
Message	Main body of the notice	1-60 characters
Priority	To determine importance	low/medium/high
Sensitive	Notice must be dismissed via PIN NOTE: this will ONLY be enabled if a dismiss PIN is provided via user attributes	true/false
Time-To-Live	Time in minutes of when the notice will dismiss itself	0 to 2,147,483,647 – where 0 is unlimited
Button Action	A button that can trigger a custom event. See below for details.	Custom string format (see below)

NOTE: Duplicate messages are suppressed for 60 seconds. A new notice with identical values as one received within the last 60 seconds will be ignored. However, it should be the notice sender's responsibility to prevent duplication and a reasonable send frequency.

Button Action

The button action field was introduced in version 1.3. This is an optional parameter that if omitted, makes no difference by default. When provided, a button is rendered on the notice that can trigger one of the 10 custom events from the Notice Tile device.

We recommend that you consider a suitable time-to-live for these notices to ensure that they remain relevant and users are not prompted to perform actions on stale data.

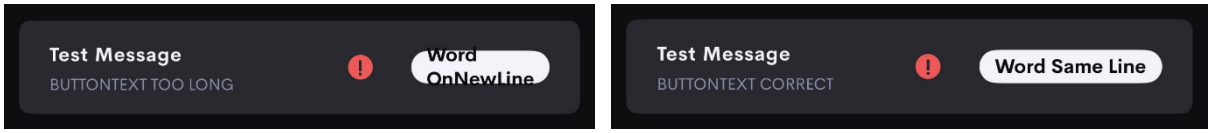
Button Action Format

```
customEventId,buttonText:confirmText
```

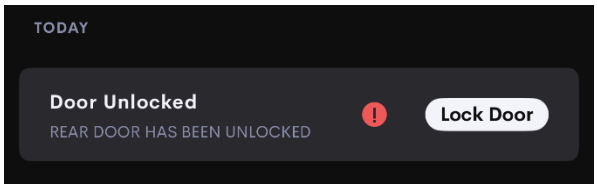
- customEventId: the number of the custom event you want to trigger when the user confirms the action. (See programmable events for more information).
- buttonText: the text shown on the button (1 to 15 characters)
- confirmText: the confirmation message to the user (1 to 75 characters). Can be left blank, which defaults the message to "Are you sure?", but we recommend you provide a meaningful message.

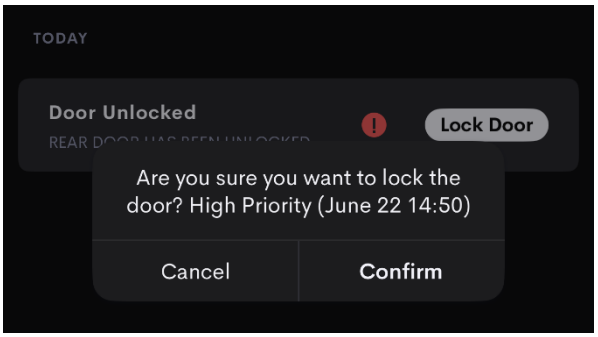
We recommend testing the button text length on the screen sizes your end users will be using. Longer text may cause unsightly wrapping.

Please also see these two examples; left where button text wraps, right where it does not.

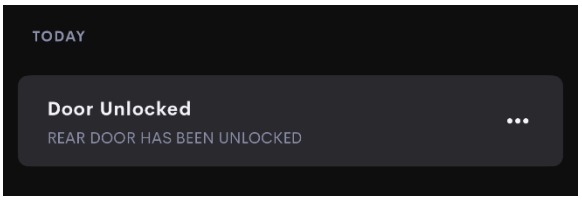


When the action button is pressed, the user is prompted to confirm the action.





Once the action is confirmed, the assigned custom event will be triggered and the notice will now be displayed as a regular notification, marked as read.



HTTP Listener

The HTTP listener allows both GET and POST methods and supports optional values. The following fields may be omitted and will be replaced with the default if they are missing.

Field	Default Value
Priority	low
Sensitive	false
Time-To-Live	0 (unlimited duration)
Button Action	*No button will show*

HTTP POST

The HTTP POST allows a body to be provided with the standard JSON data types.

URL Format

```
http://<processor_ip>:<http_port>
```

Headers

```
Content-Type: application/json
```

Body Example

```
{
    "title": "Door Unlocked",
    "message": "Rear door has been unlocked",
    "priority": "high",
    "sensitive": false,
    "ttl": 5,
    “buttonAction”: “1,Lock Door:Are you sure you want to lock the door?”
}
```

HTTP GET

The HTTP GET adds notices via URL parameters only, formatted as strings.

URL Format

```
http://<processor_ip>:<http_port>?title={title}&message={message}&priority={priority}&sensitive={sensitive}&ttl={ttl}&buttonAction={buttonAction}
```

Example

```
http://192.168.30.60:11555
?title=Door Unlocked
&message=Rear door has been unlocked
&priority=high
&sensitive=false
&ttl=5
&buttonAction=1,Lock Door:Are you sure you want to lock the door?
```

TCP/UDP

The TCP and UDP messages are simple pipe delimited string values, as like the HTTP GET URL parameters. These protocols do NOT allow for optional or default values.

The message format requires the fields to be ordered as shown below with a pipe '|' delimiter and followed by a new line '\n'.

New Parameters (v1.3+)

The original 5-field format from v1.2 and earlier is fully supported — existing messages require no changes. Fields introduced in v1.3 and later are optionally appended to the end of the message: omit them using the hash symbol (#) to preserve existing behaviour.

As new fields are introduced in future versions, they will always be appended in order. Because the format is positional, if you wish to use a later field you must explicitly pass # for any earlier optional fields you want to skip. It is therefore good practice to use # for any optional field you are not using, so that adding a new field later requires only appending to your existing message rather than restructuring it.

Message Format

```
Title|Message|Priority|TTL|Sensitive|ButtonAction\n
```

Example

```
Door Unlocked|Rear door has been unlocked|high|5|false|1,Lock Door:Are you sure you want to lock the door?\n
```

Example – Opting out of ButtonAction

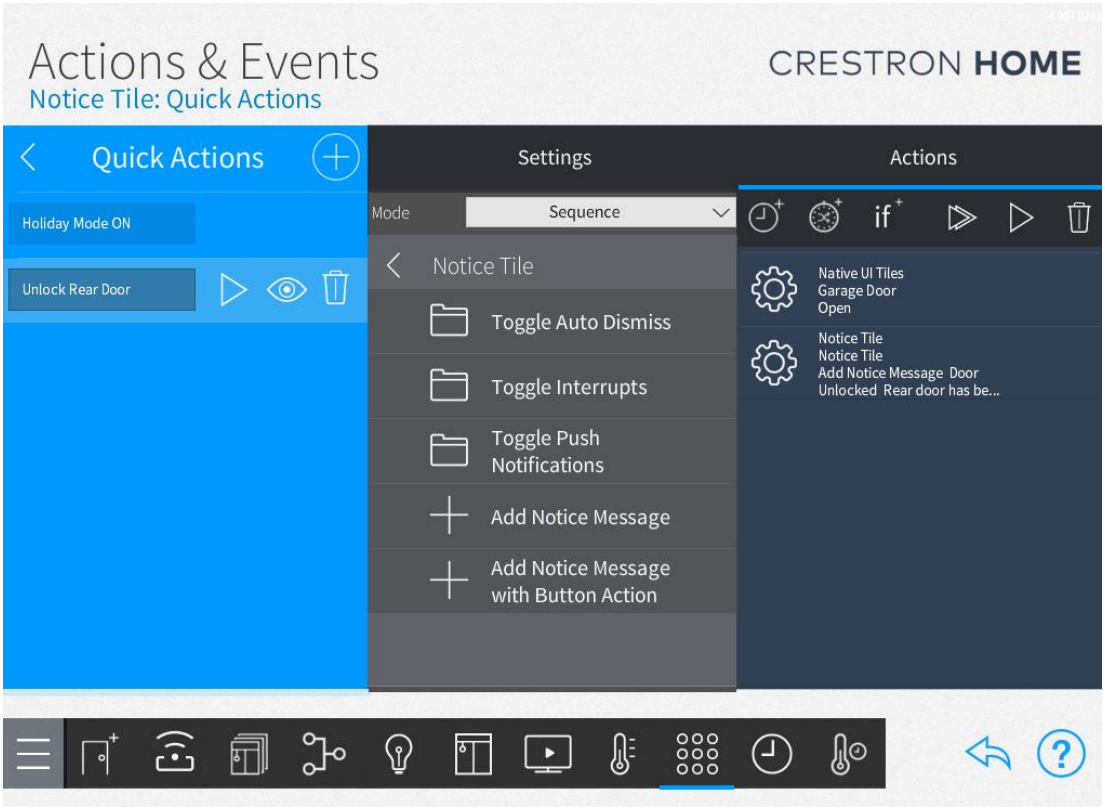
```
Door Unlocked|Rear door has been Unlocked|high|5|false|#\n
```

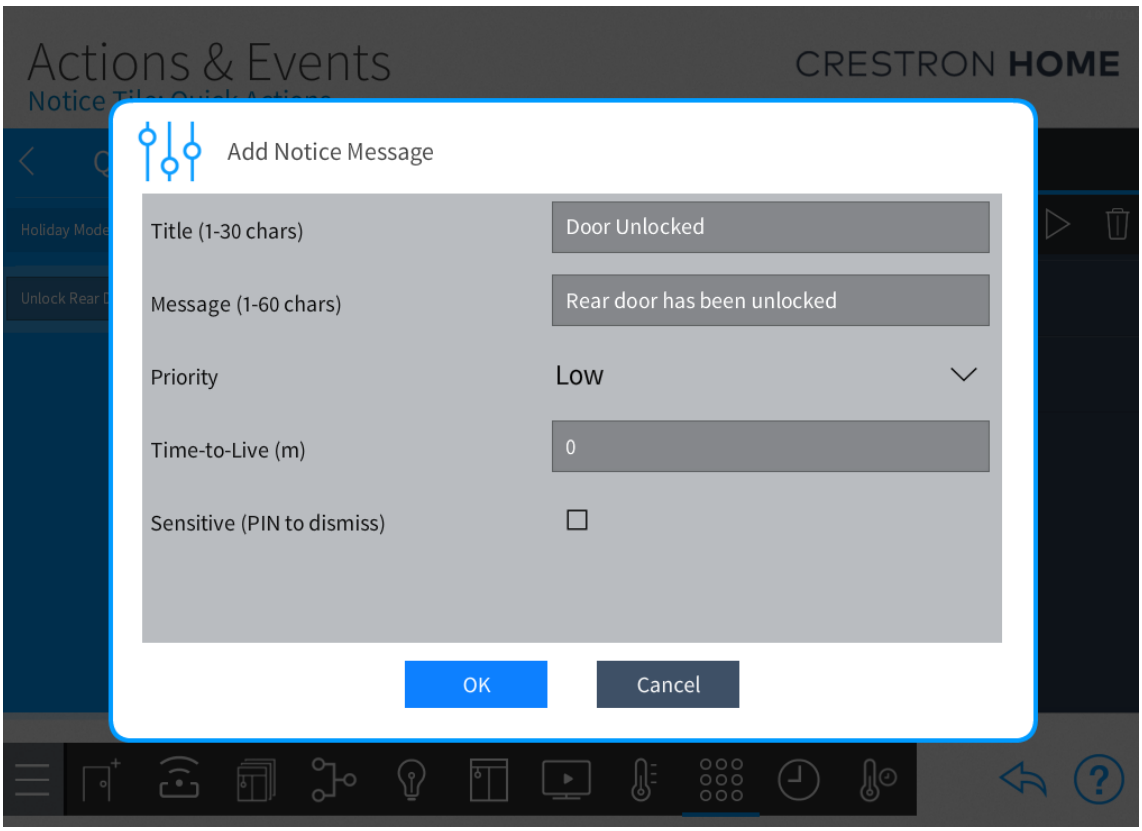
Example – Backwardly compatible schema from v1.2 and before, *still works in v1.3+*

```
Door Unlocked|Rear door has been Unlocked|high|5|false\n
```

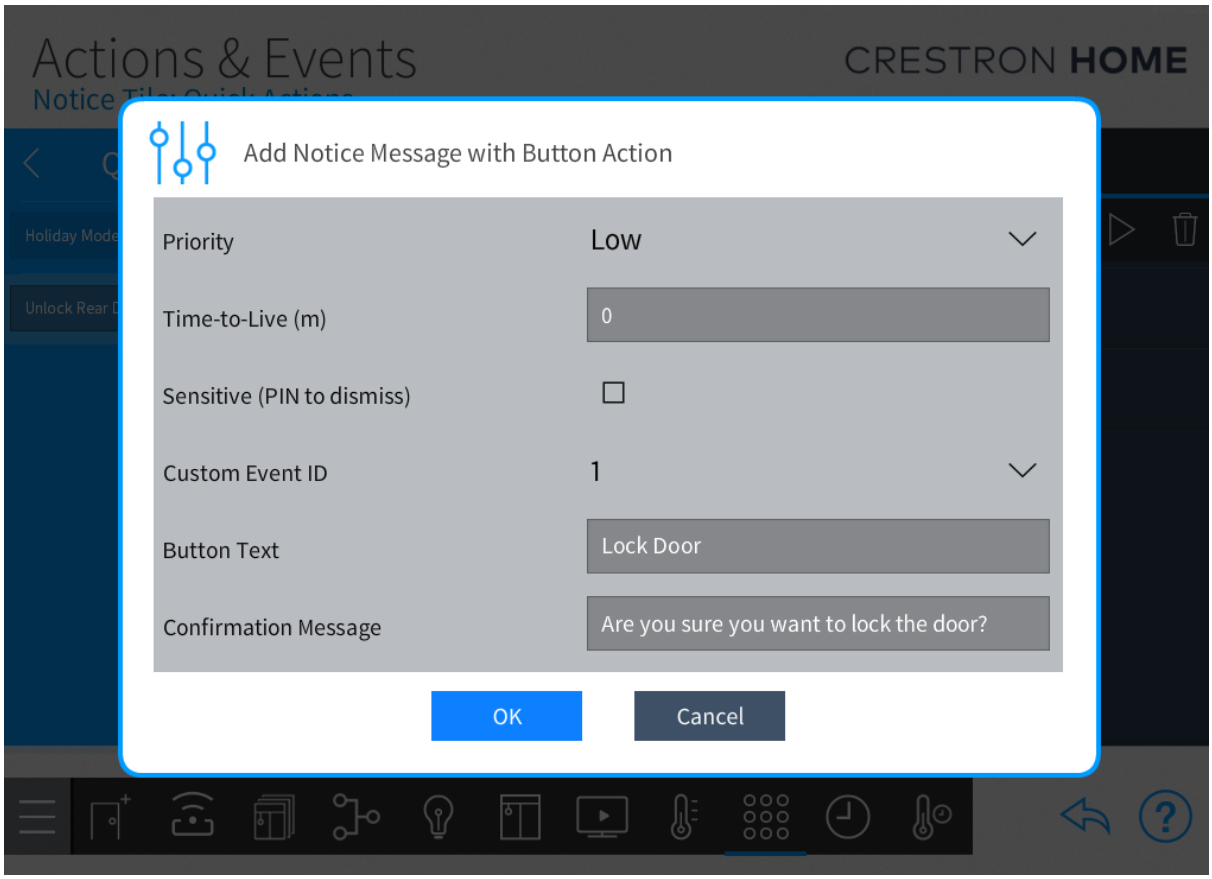
Add Notice Message via Programmable Action

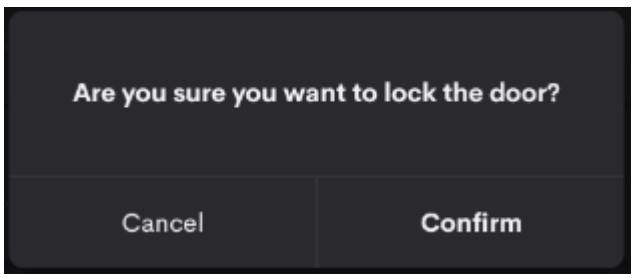
Notifications can be created by any programmable event or quick action using the Notice Tile's programmable action 'Add Notice Message'.





The second programmable action is the “Add Notice Message with Button Action” that provides fields for the button action.

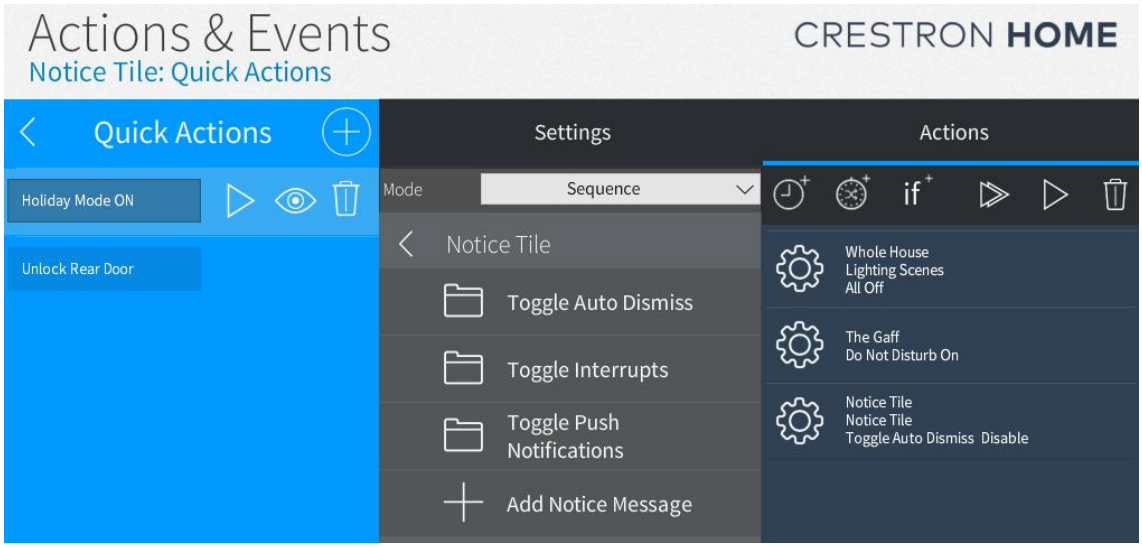




ACTIONS & EVENTS

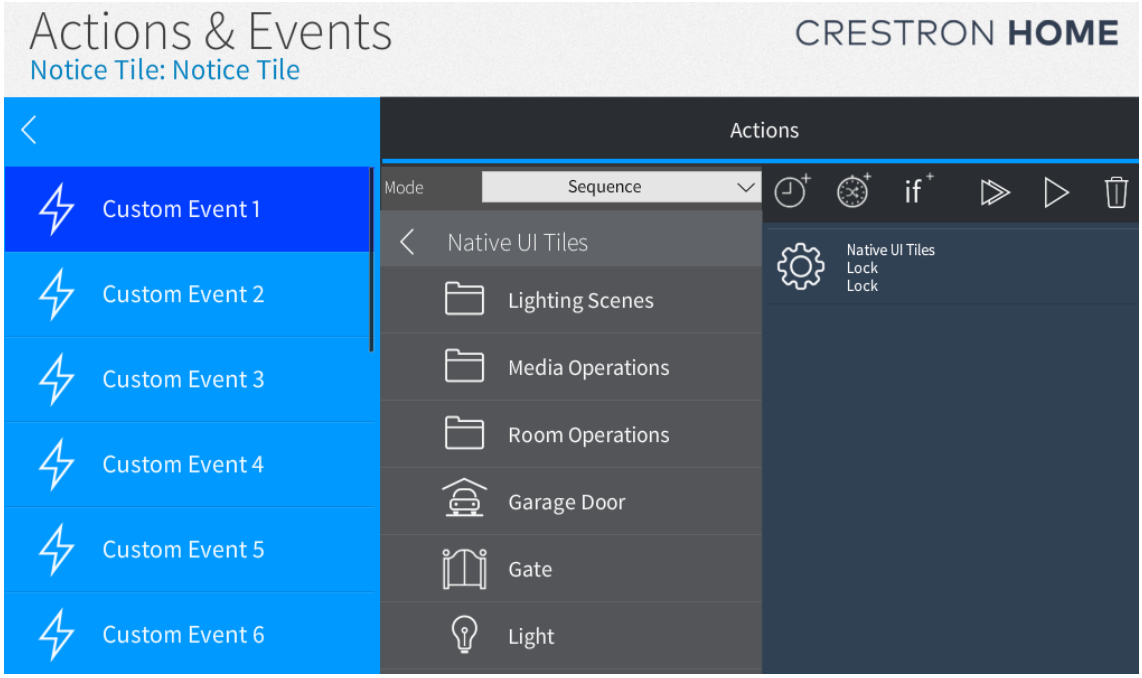
Programmable Actions

The primary programmable action for this driver is Add Notice Message. Additional toggle actions provide programmatic control over the user settings available on the Notice Tile — for instance, auto-dismiss can be disabled whilst holiday mode is active to ensure important notifications are retained during the user's absence.

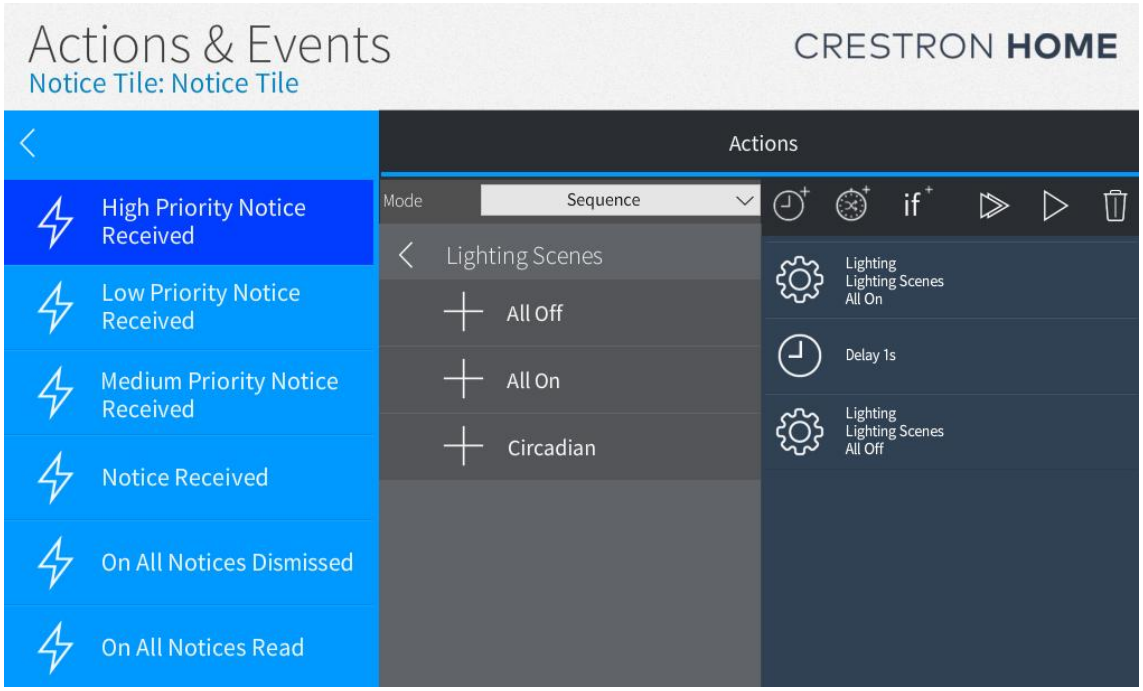


Programmable Events

Custom events are triggered by button actions as described previously. Ten events are available, each identified by a numeric ID corresponding to the custom event number. Each event fires once upon confirmation of its associated button action, where one has been specified.



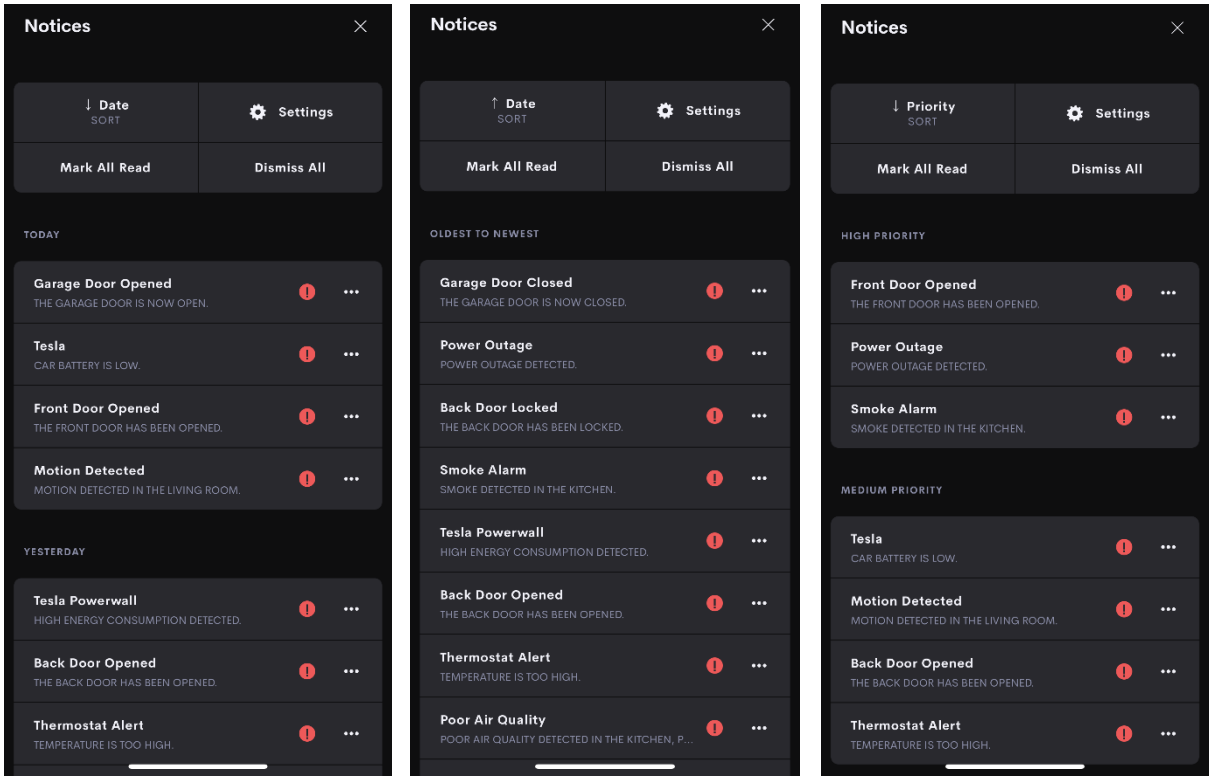
Further events from the driver are triggered when differing priorities of notices are received, or when notices are dismissed or marked read. This may be useful to set up automations such as an indicator that a message is received or is awaiting acknowledgement.



USER GUIDE

Sorting

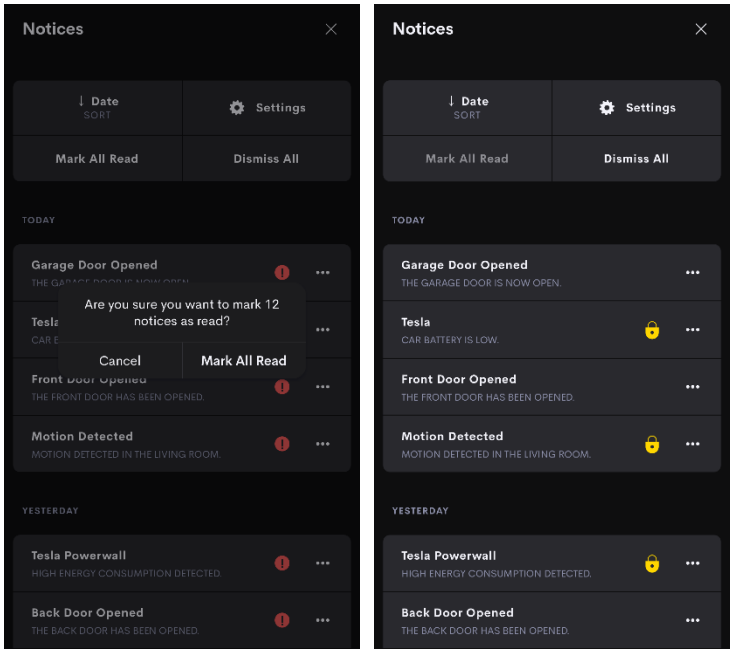
From the Notice Tile the user can view and manage up to 100 notice messages at once, where they can be sorted by date or priority in ascending and descending order.



Sorted by date ascending (left), date descending (middle) and priority ascending (right).

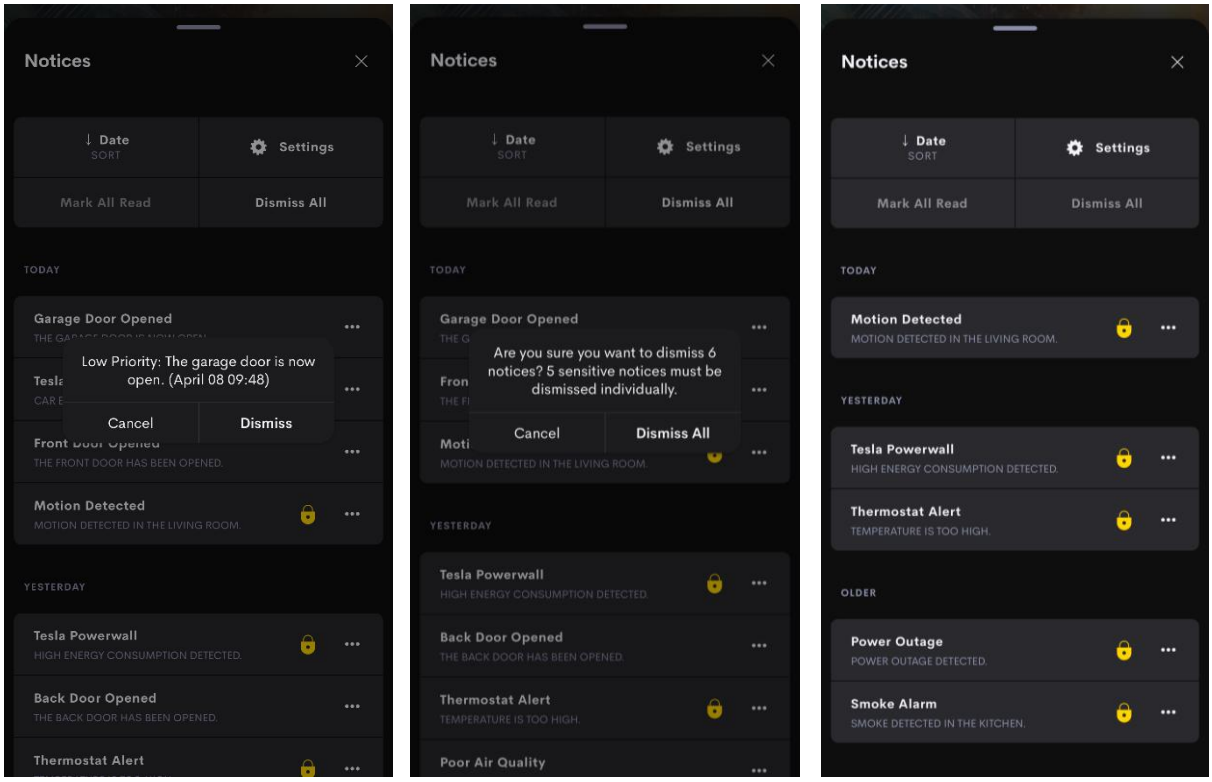
Mark Read

As presented within the tile, notices with the red alert icon are unread, which is also reflected on the tile. Notices can be marked as read via the 'Mark All Read' button. Once marked as read, sensitive messages (if enabled) will display the lock icon.



Dismiss

Non-sensitive notices can be dismissed individually via the navigation button, or dismissed in bulk via the 'Dismiss All' button, which will exclude sensitive notices.



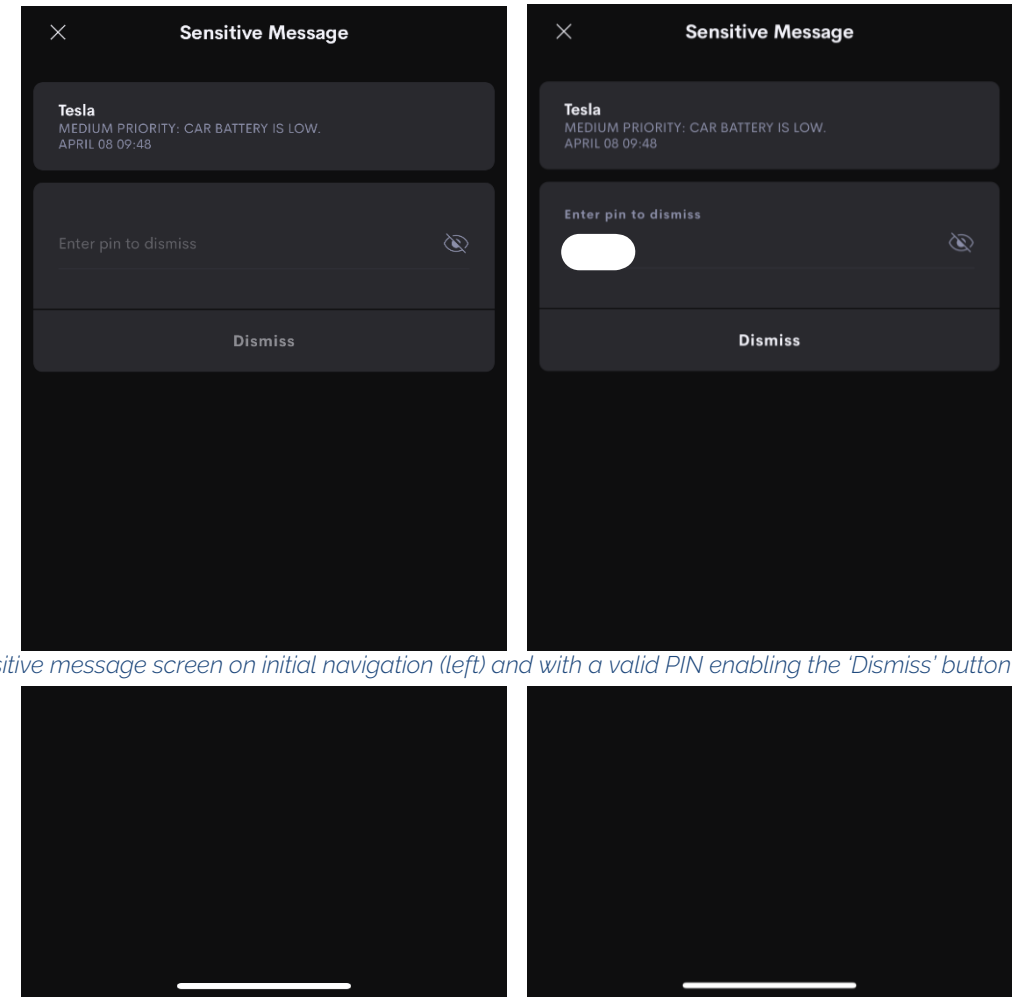
Dismissing a notice individually (left), dismissing all notices (middle) and remaining sensitive notices (right)

Sensitive Dismiss

The sensitive notice feature is useful for notices that require administrative intervention. For example, to prevent a 'regular' user accidentally dismissing a notification of a critical incident which requires attention.

This may be implemented however you'd like, however please use sparingly – these notices *must* be dismissed one by one! You *may* provide a TTL for sensitive messages, however it is recommended to provide an unlimited TTL to ensure the administrator acknowledges the message.

Sensitive messages can be dismissed by using the same navigation button as all other messages, where you will be navigated to a page with a text entry and button. When the correct PIN is entered (the 'Sensitive Dismiss PIN' provided within the installer settings), the button will become enabled and allow the message to be dismissed.

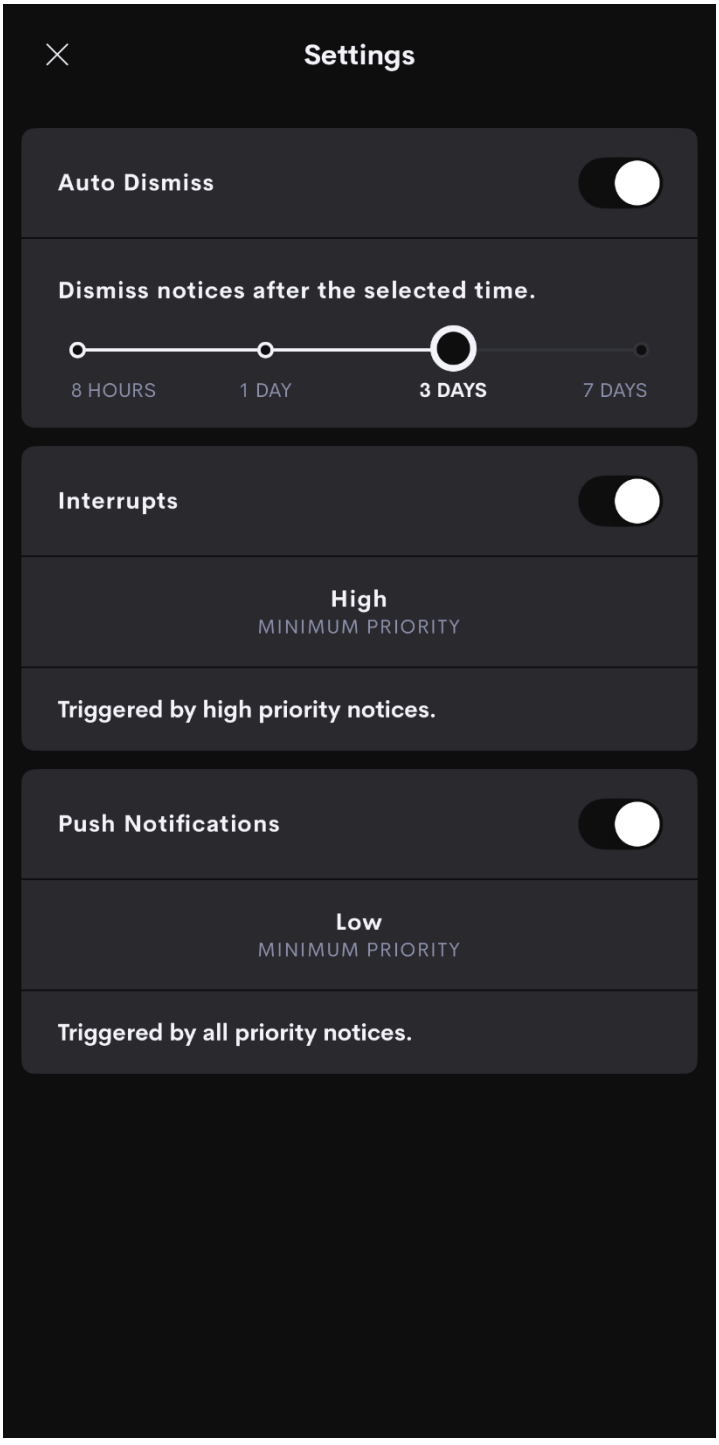


Sensitive message screen on initial navigation (left) and with a valid PIN enabling the 'Dismiss' button (right).

User Settings

The 'Settings' button will navigate to the user settings page, where the following settings can be configured:

- Auto Dismiss: Automatically dismiss non-sensitive messages after the specified time. Sensitive messages require individual dismissal with PIN entry.
- Interrupts: Toggle to enable/disable interrupts and set the minimum priority that triggers them. (*This is only visible if the 'Enable Interrupts' installer setting is enabled).*
- Push Notifications: Toggle to enable/disable push notifications and set the minimum priority that sends them. *(This is only visible if Pushover is configured).*



TROUBLESHOOTING

If you experience any issues when installing the driver, there are some troubleshooting steps that should be taken. Hopefully these will resolve the primary issue, but if not, will provide valuable information when you then contact support (see below).

Although the Crestron Home configuration app includes some simple logging features, Crestron's Toolbox is far more powerful when interrogating the system for logging and messages.

If you are not familiar with Crestron's Toolbox application, please refer to the Crestron training materials which cover where to obtain the software and how to install and use it.

Within Crestron Toolbox, open the Console Tool and connect to the Crestron Home processor.

Verifying Correct Listener Operation

After setting the listener ports and clicking `Next`, the integration will report the current state to the processors primary log.

The view this, type the "ERR" command in the Console and you will see various messages including some relating to the Notice Tile. Successful listener startup will look something like this:

```
Ultamation Info:NoticeTile - HttpNoticeListener.Start: Listener started on port 11555
Ultamation Info:NoticeTile - UdpNoticeListener.Start: Listener started on port 11557
Ultamation Info:NoticeTile - TcpNoticeListener.Start: Listener started on port 11556
```

If there is a conflict or an error occurs binding to the socket, you will see an error like so:

```
Ultamation Error:NoticeTile - HttpNoticeListener.Start: Failed to start HttpNoticeListener
listener on port 11555: Address already in use
```

If you wish to stop an existing listener, the port number should be made blank, and you should see the following info log:

```
Ultamation Info:NoticeTile - TcpNoticeListener.Stop: Listener stopped on port 11556
```

SUPPORT

If you have any issues with a driver or installation please let us know by contacting Ultamation support on support@ultamation.com and please include as much detail about your issue as possible, such as a recent processor error log.

Licence verification messages are posted to the error log, so please ensure you have checked this.

LICENCING

This driver (including software, images and all other associated assets distributed as part of the purchased download package) is licenced on a PER PROCESSOR basis.

A purchase should not be completed without correct information as refunds cannot be issued for errors or changes made to details following purchase.

This is an electronic product and there is no physical delivery.

The driver is provided without any warranty with respect to the reliability of the controlled device or changes to device protocol. We will endeavour, through best efforts, to maintain the driver's functionality and any bug fixes will be provided free-of-charge. Additional functionality may be released as a variation of this driver and this will be a separate, purchasable, product.

CLOUD LICENCE

This driver contacts Ultamation's licencing server at startup. If the server finds a matching licence for the driver and processor then the driver will be licenced. Otherwise, the driver will check the offline licence key. If you purchase a licence AFTER you have loaded the driver, please reboot the system to see changes take effect.

If you purchased a licence **before** it was migrated to the cloud service, i.e. you have a licence key already, you must enter this into the **Offline Key** user attribute. If you purchased the driver **after** it was migrated, and you don't have a licence key, no further action is required.

If no licence exists for the product/processor the driver will enter a short trial period (ONE HOUR) to allow for verification of correct control or evaluation.

To request an OFFLINE key, please contact support@ultamation.com with your order details and a brief explanation why you REQUIRE offline activation. Ultamation reserve the right to refuse offline activation.

NOTE: Once an offline key has been issued no further licence changes will be granted. Moving the product to a new processor will require an additional licence purchase.