HAI System Plugin for HomeSeer

Requirements

- HomeSeer version 2.1.0.0 or greater is required.

Installation

1. To install the HAI System plug-in for HomeSeer version 2.x, select the “Updater” button on the web interface, or select “Updates” from the “Help” menu. Follow the on-screen instructions for the Updater.
2. Select the HAI System plug-in on the package selection screen, then follow the on-screen instructions to finish the update process, and restart HomeSeer.
3. After restarting HomeSeer, click the “Setup” tab and then the “Interfaces” tab on the web site. Click “Disabled” to enable the plugin, and then click “Save” at the bottom of the screen.

Introduction

This plug-in supports HAI models Omni, Omni Pro, Omni LT, Omni II, and Omni Pro II

Here are the current known limitations or untested functionality pieces in this version:

- Using the panel as an X-10 interface is not supported.
- Auxiliary Temperature Sensors are supported for temperature sensing only – the associated relay/thermostat functionality is not supported. (Programmable Energy Saver Module)

Please read the section titled "How It Works - What To Expect" for important information to help you get started using this plug-in.

How It Works – What to Expect

The plug-in is installed for use in HomeSeer in the normal fashion for an I/O plug-in. You MUST be running a version of HomeSeer that is greater than or equal to 2.1.0.0. Version 2 of the .NET Runtime is required to use this plug-in and the 2.1 version of HomeSeer.

When the plug-in is added to HomeSeer, you must restart HomeSeer for any of the plug-in features to take affect.

When HomeSeer is started, if the COM port of the HAI panel has not been specified, then the plug-in configuration web page will appear asking for the COM port of the HAI system. This is the setup page and can also be accessed by clicking on the Config button on the Interfaces screen of HomeSeer in the HAI System plug-in row.

After entering the COM port, you will also need to enter the panel code, and a starting model number. The panel master code is preferred, although the PC Access (if PC Access is enabled) and Installer code may also be used. Use of ANY other code will cause the plug-in to be able to perform some functions but not all, which means the plug-in will seem to operate intermittently rather than as it was designed. The panel code entered is encrypted before being stored in the hspi_HAI.ini file in the config folder. If you change the code, you may do so at this screen, or you can edit the hspi_HAI.ini file directly. When you edit the INI file directly, enter the 4 digit code and save the file, and when the plug-in is restarted, it will detect the unencrypted code and then it will modify the entry to replace it with the encrypted code.

If the COM port, panel code, and panel model have all been configured, then the next level of configuration information is checked. If no panel limits have been entered, then the panel limits portion of the configuration page will be required to be completed. This page allows you to enter information on the different zones, units, buttons, etc. that you wish to have access to. For example, if you are only using 14 zones in your system that has a maximum of 16 installed, enter 14 for the number of zones of security and the various status displays, triggers and condition screens will not display your unused zones.

After entering your limits, the limits will be checked against the known limits for your model of panel. Any discrepancies will be noted and you will be given the opportunity to make any necessary corrections.

After entering your limits and they have been validated, clicking "Save" will save your settings and will trigger the plug-in to again check to see if it has enough information to communicate with the panel.

At startup, you should see a line in the HomeSeer log similar to this to verify communication with the panel:

~HAI System~!~Panel Model: 4 (Omni Pro) with Panel Software Version: 1.8

If the HAI system configuration is completed and the panel code entered is valid, then you will see this line to indicate that the plug-in has begin operation:

~HAI System~!~Performing Startup/Recovery Check of the System.

And finally, the completion of the startup/recovery check should be indicated with the following line:

~HAI System~!~All HAI Items and Units are Up-To-Date. Triggers and Conditions are now active.
When you see the above line, the plug-in is fully functional. Do not continue with the next step until you have verified that the above line appears in your log.

Once valid communications has been established, you can name the zones, units, thermostats, etc. in the system. The names you entered in your HAI system are typically very limited - 12 to 15 characters depending upon the item being named. This plug-in gives you much more flexibility in naming things - it is almost unlimited. The way the plug-in does this is to store the names locally, in its INI file, since the panel can only hold 12 to 15 characters. However, it is difficult to remember, for example, what unit 134 is when you are entering the names, and so this is where downloading names from the panel comes in handy. In order to help you understand what each zone, unit, thermostat, etc. is in the system, there is a function that allows you to download the current names from the panel. This should be done only once since downloading names will overwrite any names that you may have entered using the plug-in (which may be longer, more descriptive names). Here is an example - you have a "Family Room Chandelier" at address B1 which may be unit 17 in the system. When the plug-in first starts, the name shown on the HAI Units web page will be "Unit 17". After you download names from the panel, the name might be the name you entered in the HAI system which is "Family Rm Chndr". You can then change the name to the easier to understand "Family Room Chandelier"; but if you ever repeat the procedure to download names from the panel, the name will revert to "Family Rm Chndr".

From the configuration web page (/HAIWebConfig) find the link under "Name Items / Delete Devices / Logging Options" titled "Name Configuration Page". This link will take you to the names configuration page where a large button on the right hand side of the page is labelled "Get Names from Panel". Clicking this button will walk you through the procedure to download names from the panel.

If you wish to set the names manually, or wish to change the name of something later, you can select the names category from the "Select Name Category" drop-down list, and the listing of items in that category will be displayed. The category and its category item number will be displayed along with the current name assigned to it. Blank names mean that no name was entered or downloaded from the panel for that item. Using the earlier example on an HAI Omni Pro panel, unit 134 will appear with a category and unit of: "Expansion Unit/Flag: 134". After entering the names you wish to have, click the 'Save' button at the bottom of the page BEFORE you change to a different category to prevent losing your changes.

At this point, all of the basic configuration steps have been completed and you are ready to begin using the system and learning more about the capabilities of using it, including the creation of triggers, conditions, and actions in HomeSeer.

**What's New**

Version 2.1.2567.29695

- `SetPanelDateTime` was added to set the security panel's date, time, and daylight savings mode to the HomeSeer computer's settings. The call should be made when the time is at exactly 0 seconds as the command to set the date/time/ds mode does not include the seconds. The script command is a function that returns a Boolean result of True/False indicating whether the command was a success or not. Here is an example (using VBScript):

```vbs
Sub Main

dim bRet

dim HAI

set HAI = hs.Plugin("HAI_System")

bRet = HAI.SetPanelDateTime

hs.WriteLog "HAI Time", "Result is " & bRet

End Sub
```

- When the panel cannot communicate with a thermostat, the HVAC web page will reflect this error condition, and if thermostat devices are used, the status of each thermostat device will show "Comm Error"

- When a thermostat is reporting a Freeze Alarm, the condition is now reflected in the thermostat current temperature device status, and it is also displayed on the HVAC web page for that thermostat.

- If you try to view the Security, Units, or HVAC web pages before the system is done initializing, a message will be displayed letting you know that it is not possible to view or control the system until communications and synchronization is complete.

**New in Previous 2.1 Releases**

Along with the advent of HomeSeer version 2.0, the plug-in has undergone several more enhancements to support HomeSeer 2 and other software:

- There are 4 new devices created along with the main interface devices that are always created. The popular Previous System Status device supported through a script in the past is now created by the plug-in - this allows you to create automation events based upon the security system being disarmed when the previous mode was Armed Away versus the previous mode being Armed Night. A battery level device now exists - if the level indicated on the device is 13.47 Volts, then the value of the device is 1347. Thus, device value comparisons can be done to determine when the voltage is getting low as when the battery is getting old and needs replacing. Two other devices are indicators of when any zone or any of the special indicators (e.g. AC trouble, Freeze warning) change. These devices can be used to trigger an event/script that can interrogate further to determine if action should be taken.

- More of the system setup/configuration has been moved to a web page to make the plug-in more HomeSeer 2 compatible.

Since the original version of this plug-in in 2003 there have been a number of enhancements/improvements:

- The plug-in now gathers information about the configuration of the system from the panel itself. This eliminates the need to specify fire/smoke/CO2 detector zones or name them in a special way so that the plug-in can identify them. Thermostats and temperature sensors are also detected this way.

- Devices are now created for zones, units, and HVAC items in the system although each group and each item in each group may have a setting to override and prevent the creation of a device for it. This feature facilitates the use of the plug-in with other plug-ins that depend upon devices as the go-between including TouchPad, Media Center, and MainLobby.

- Messages to be displayed on the system consoles may now have their text set/changed and the plug-in can display these messages. In other words, it is possible now to have your HAI consoles display a message indicating the number of voicemail messages you have, even though this
is a dynamic value.

- The plug-in actions page now includes device actions for X-10 and ALC devices, including support for direct level set if the device supports this.
- When unit devices are created, the ALC lighting units are included in order to provide a device interface to those lighting loads. X-10 devices are NOT created and should instead be created in HomeSeer and should utilize HomeSeer's X-10 interface for control and status. (This is due to significant shortcomings in the use of the panel as an X-10 interface.)
- A web based user configuration page exists that allows for the configuration allowing or denying specific HomeSeer users from accessing the built-in plug-in web pages. This allows for one individual to be able to change HVAC settings without having access to the security system page.
  
  Note: The security status device in the HAI System location now allows the security system to be armed or disarmed through a device interface - please be aware of this.
- All of the devices created by the plug-in may be renamed or have their location changed without affecting the operation of the plug-in.
- Polling of the panel (required for interfacing with this panel) now uses a polling algorithm designed to obtain the status of non-event driven items (e.g. thermostats) much more frequently.

How It Works – Devices and Screens

This section describes the various methods for interacting with the plug-in and thus your HAI panel. Since this plug-in was written using the new version 2 plug-in API by HomeSeer, there are some new functions in this plug-in never before available. The new API has provided a means by which interaction with the system does not need to be done using devices – virtual or otherwise. Some devices are used however and they are described in this section. The plug-in will create devices to represent zones, units, HVAC items, etc., but these may be disabled. Only the special control devices in the HAI System location are created and used at all times.

Devices

The plug-in will automatically create 15 devices and put them in the device group “HAI System”. If for whatever reason you need to delete these devices, there is a button labeled “Reset Configuration” on the setup screen. (Go to View… Options… Interfaces… in HomeSeer, then select HAI_System in the I/O plug-in section and click the “Setup” button.) The reset configuration button will delete all of the HAI related devices. When the plug-in is re-initialized (e.g. restarted with HomeSeer) it will detect the absence of these devices and will re-create them. If you use the reset configuration option, please do so just before shutting down HomeSeer.

Note: The Reset Configuration option deletes ALL plug-in owned devices, not just those in the HAI System location.

The following devices are created by the plug-in:

Security Status

This is the arm status of the panel. It will contain one of the following status (string) values: Disarmed, Armed Day, Armed Night, Armed Away, Armed Vacation, Armed Day Instant, or Armed Night Delayed. Additionally, all of these status strings except “Disarmed” may be followed by the string " (Delay)” during the time that the arming exit delay is in effect. The device values are 0 through 6 respectively, 101 through 106 during the arming exit delay. When creating device value triggers in HomeSeer, if you wish to know about the panel being armed in Away mode for example, you can build your trigger for this device to have a value which becomes equal to 103 to trigger an event at the start of the exit delay, or 3 to trigger an event for the fully-armed Away mode. The string status and values are summarized in the following table:

<table>
<thead>
<tr>
<th>Security Status Table</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Status</strong></td>
</tr>
<tr>
<td>Disarmed</td>
</tr>
<tr>
<td>Armed Day</td>
</tr>
<tr>
<td>Armed Night</td>
</tr>
<tr>
<td>Armed Away</td>
</tr>
<tr>
<td>Armed Vacation</td>
</tr>
<tr>
<td>Armed Day Instant</td>
</tr>
<tr>
<td>Armed Night Delayed</td>
</tr>
<tr>
<td>Armed Day (Delay)</td>
</tr>
<tr>
<td>Armed Night (Delay)</td>
</tr>
<tr>
<td>Armed Away (Delay)</td>
</tr>
<tr>
<td>Armed Vacation (Delay)</td>
</tr>
<tr>
<td>Armed Day Instant (Delay)</td>
</tr>
<tr>
<td>Armed Night Delayed (Delay)</td>
</tr>
</tbody>
</table>

Previous Security Status

This is the immediate previous arm status of the panel. It will contain one of the following status (string) values: Disarmed, Armed Day, Armed Night, Armed Away, Armed Vacation, Armed Day Instant, or Armed Night Delayed. Additionally, all of these status strings except “Disarmed” may be followed by the string " (Delay)” during the time that the arming exit delay is in effect. The device values are 0 through 6 respectively, 101 through 106 during the arming exit delay. When creating device value triggers in HomeSeer, if you wish to know about the panel being armed in Away mode for example, you can build your trigger for this device to have a value which becomes equal to 103 to trigger an event at the start of the exit delay, or 3 to trigger an event for the fully-armed Away mode.
This device is of particular use because it can tell you if a Disarm is after Armed Night (e.g. you just woke up) or if it is after Armed Away (you just got home) which are important things to know in automating events in the home.

The string status and values are summarized in the following table:

<table>
<thead>
<tr>
<th>Status</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Disarmed</td>
<td>0</td>
</tr>
<tr>
<td>Armed Day</td>
<td>1</td>
</tr>
<tr>
<td>Armed Night</td>
<td>2</td>
</tr>
<tr>
<td>Armed Away</td>
<td>3</td>
</tr>
<tr>
<td>Armed Vacation</td>
<td>4</td>
</tr>
<tr>
<td>Armed Day Instant</td>
<td>5</td>
</tr>
<tr>
<td>Armed Night Delayed</td>
<td>6</td>
</tr>
<tr>
<td>Armed Day (Delay)</td>
<td>101</td>
</tr>
<tr>
<td>Armed Night (Delay)</td>
<td>102</td>
</tr>
<tr>
<td>Armed Away (Delay)</td>
<td>103</td>
</tr>
<tr>
<td>Armed Vacation (Delay)</td>
<td>104</td>
</tr>
<tr>
<td>Armed Day Instant (Delay)</td>
<td>105</td>
</tr>
<tr>
<td>Armed Night Delayed (Delay)</td>
<td>106</td>
</tr>
</tbody>
</table>

**Special Status**

As the name implies, this device tracks special panel status conditions. These conditions have status (string) values and numerical values just as the Security Status does. Here is the list of values that the status (string) setting of the device may have followed by the numerical value representation of the status:

<table>
<thead>
<tr>
<th>Status</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Phone Line Dead</td>
<td>0</td>
</tr>
<tr>
<td>Phone Line Ringing</td>
<td>1</td>
</tr>
<tr>
<td>Phone Line Off Hook</td>
<td>2</td>
</tr>
<tr>
<td>Phone Line On Hook</td>
<td>3</td>
</tr>
<tr>
<td>AC Power Off</td>
<td>4</td>
</tr>
<tr>
<td>AC Power Restored</td>
<td>5</td>
</tr>
<tr>
<td>Battery Low</td>
<td>6</td>
</tr>
<tr>
<td>Battery OK</td>
<td>7</td>
</tr>
<tr>
<td>Digital Communicator Trouble</td>
<td>8</td>
</tr>
<tr>
<td>Digital Communicator OK</td>
<td>9</td>
</tr>
<tr>
<td>Energy Cost Low</td>
<td>10</td>
</tr>
<tr>
<td>Energy Cost Mid-Range</td>
<td>11</td>
</tr>
<tr>
<td>Energy Cost High</td>
<td>12</td>
</tr>
<tr>
<td>Energy Cost Critical</td>
<td>13</td>
</tr>
</tbody>
</table>

**Alarm Type**

Should you have the misfortune of having an alarm occur on your premises, this is the device that will hold the type of alarm that occurred. As with the previous two devices, there are status (strings) and values and they are:

<table>
<thead>
<tr>
<th>Status</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Burglary</td>
<td>1</td>
</tr>
<tr>
<td>Fire</td>
<td>2</td>
</tr>
<tr>
<td>Gas</td>
<td>3</td>
</tr>
<tr>
<td>Auxiliary</td>
<td>4</td>
</tr>
<tr>
<td>Freeze</td>
<td>5</td>
</tr>
<tr>
<td>Water</td>
<td>6</td>
</tr>
<tr>
<td>Duress</td>
<td>7</td>
</tr>
<tr>
<td>Temperature</td>
<td>8</td>
</tr>
</tbody>
</table>

**Last Code Used**

When the security system is commanded to perform a function, especially a security function, a user code is required. This device holds the number for
the user code (value) and the name, if names were provided, of the user code. Please note that these are not the 4 digit user security code but rather the position in the list of user codes that the particular security code was entered into. For example, the master code for the system is stored in user code 1. There are 5 special user codes in user code positions 251 through 255 documented in the following table. The status (string) that will be displayed as well as the numerical value are indicated. For all other user codes, the user code number will be the numerical value. Actions carried out by the plug-in in events will always use the user code you provided at the HAI plug-in setup screen which is the PC Access (254) code or the Master (1) code.

### Special User Code Table

<table>
<thead>
<tr>
<th>Special User Code</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Duress Code</td>
<td>251</td>
</tr>
<tr>
<td>Key Switch</td>
<td>252</td>
</tr>
<tr>
<td>Quick Arm</td>
<td>253</td>
</tr>
<tr>
<td>PC Access</td>
<td>254</td>
</tr>
<tr>
<td>Programmed</td>
<td>255</td>
</tr>
</tbody>
</table>

**Button Pushed**

The HAI system uses the moniker “button” for macro operations. When you “push” HAI’s buttons, commands associated with the particular button in the HAI programming are carried out. Buttons may be “pushed” by another program, by a user activating a button at a console, or programmatically through an action via this plug-in. Whenever a button is “pushed” in the system, the status (string) value for this device will be the associated name if provided, and the value for this device will be set to the button number that was “pushed”. If no name is provided, the status (string) value will be “Button” followed by the button number.

**Freeze Alarm, Fire Emergency, Police Emergency, Auxiliary Emergency, Duress Alarm, Battery Low Trouble, AC Powerfail Trouble, Phone Line Dead Trouble and Digital Communicator Trouble**

These special alarm conditions all operate in a similar fashion. When the alarm is present, their value is 1 and their status is “Trouble”. When the alarm condition indicated by the device title is not present, then the value is zero and the status is “OK”.

**Battery Level**

The battery level device shows the current battery level and provides a device interface for detecting when the level goes above or below a certain value.

If the battery level device is showing (for example) 12.21 Volts, then the device VALUE will be 1221 (the voltage without the decimal).

When the battery level goes below approximately 11.60 Volts, then the Battery Low Trouble alarm will occur on the panel.

**Change Indicators**

There are two special change indicator devices: Zone Change Indicator and Special Status Change Indicator.

Often it is difficult and time consuming to create an event that can be triggered when ANY zone changes state - e.g. when a window is opened or a door closed. These change indicators help with this by momentarily toggling state from "Changed" to "No Change". Since this transition is very fast, you will very likely not see it on the HomeSeer device status display, but it does change and thus can be used for a Device Value Change trigger in HomeSeer.

When the plug-in first starts, the display status of these devices may be "Reset" depending upon your version of HomeSeer. The device value for reset is 99. When the device changes to Changed or Not Changed, the value of the device is 1 and 0 respectively.

**Devices - HVAC**

These devices represent the various HVAC elements in the system not represented in any of the other device types, and you may change their name and location as you see fit. They are initially created by the plug-in using the location "HAI HVAC" and with the name that you assigned to the thermostat or aux temp zone.

Since there are several different parameters to an HVAC unit's status (e.g. current temperature, thermostat mode, etc.) the various status values are different depending upon the unit type.

Additionally, control options will be available depending upon the unit type. When a control option is available, the control column appears in HomeSeer's HTML UI, and a right-click on the device in the Windows UI provides control options. Note that when a control option is chosen but it is not an option that exists as well as a status value, the command will be carried out and the status of the device will immediately change to reflect the new value - thus the change is momentary, but can still trigger events.

The string status and values are summarized in the following table:

### HVAC Device Status Table

<table>
<thead>
<tr>
<th>Status</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>
These devices represent the various units in the system not represented in any of the other device types, and you may change their name and location as you see fit. They are initially created by the plug-in using the location "HAI Units" and with the name that you assigned to the unit.

Since there are several different parameters to a unit's status (e.g. light levels, counters incrementing and decrementing, outputs on/off, etc.) the various status values are different depending upon the unit type.

Additionally, control options will be available depending upon the unit type. When a control option is available, the control column appears in HomeSeer's HTML UI, and a right-click on the device in the Windows UI provides control options. Note that when a control option is chosen but it is not an option that exists as well as a status value, the command will be carried out and the status of the device will immediately change to reflect the new value - thus the change is momentary, but can still trigger events.

The string status and values are summarized in the following table:

<table>
<thead>
<tr>
<th>Devices - Units</th>
<th>Status Value</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Thermostat and Auxiliary Sensor Temperature Devices</strong></td>
<td>0 - 1000 through 1255</td>
</tr>
<tr>
<td>Unknown</td>
<td>0</td>
</tr>
<tr>
<td>(All Temperature Values)</td>
<td>1000 through 1255</td>
</tr>
<tr>
<td>Off</td>
<td>1</td>
</tr>
<tr>
<td><strong>Thermostat Mode Devices</strong></td>
<td></td>
</tr>
<tr>
<td>Unknown</td>
<td>0</td>
</tr>
<tr>
<td>Off</td>
<td>1</td>
</tr>
<tr>
<td>Heat</td>
<td>2</td>
</tr>
<tr>
<td>Cool</td>
<td>3</td>
</tr>
<tr>
<td>Auto</td>
<td>4</td>
</tr>
<tr>
<td>Aux/Emerg (status only)</td>
<td>5</td>
</tr>
<tr>
<td><strong>Thermostat Fan Devices</strong></td>
<td></td>
</tr>
<tr>
<td>Unknown</td>
<td>0</td>
</tr>
<tr>
<td>Auto</td>
<td>1</td>
</tr>
<tr>
<td>On</td>
<td>2</td>
</tr>
<tr>
<td><strong>Thermostat Heat/Cool Setpoint Devices</strong></td>
<td>0 - 1000 through 1255</td>
</tr>
<tr>
<td>Unknown</td>
<td>0</td>
</tr>
<tr>
<td>(All Temperature Values)</td>
<td>1000 through 1255</td>
</tr>
<tr>
<td>For values between 50F/C and 88.7F/87.5C, the temperature appears as a setpoint control value. For all other temperatures, it is a status display value only.</td>
<td></td>
</tr>
<tr>
<td>*Note: The panel protocol provides no mechanism for determining an auxiliary temperature sensor from a humidity sensor, so values only are displayed without a unit display (e.g. it will not display ° or % after the value.</td>
<td></td>
</tr>
</tbody>
</table>

**Devices - Units**

These devices represent the various units in the system not represented in any of the other device types, and you may change their name and location as you see fit. They are initially created by the plug-in using the location "HAI Units" and with the name that you assigned to the unit.

Since there are several different parameters to a unit's status (e.g. light levels, counters incrementing and decrementing, outputs on/off, etc.) the various status values are different depending upon the unit type.

Additionally, control options will be available depending upon the unit type. When a control option is available, the control column appears in HomeSeer's HTML UI, and a right-click on the device in the Windows UI provides control options. Note that when a control option is chosen but it is not an option that exists as well as a status value, the command will be carried out and the status of the device will immediately change to reflect the new value - thus the change is momentary, but can still trigger events.

The string status and values are summarized in the following table:

<table>
<thead>
<tr>
<th>Unit Device Status Table</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Status</strong></td>
</tr>
<tr>
<td><strong>Lighting Units</strong></td>
</tr>
<tr>
<td>Unknown</td>
</tr>
<tr>
<td>On</td>
</tr>
<tr>
<td>Off</td>
</tr>
<tr>
<td>1% through 99%</td>
</tr>
<tr>
<td>Dim 1 through Dim 9</td>
</tr>
<tr>
<td>Bright 1 through Bright 9</td>
</tr>
<tr>
<td>Scene A through Scene L</td>
</tr>
<tr>
<td><strong>Output &amp; Control Trigger Units</strong></td>
</tr>
<tr>
<td>Unknown</td>
</tr>
<tr>
<td>Closed (On)</td>
</tr>
<tr>
<td>Open (Off)</td>
</tr>
<tr>
<td><strong>Internal Flag (Counter) Units</strong></td>
</tr>
<tr>
<td>Unknown</td>
</tr>
<tr>
<td>0 through 255</td>
</tr>
<tr>
<td>Set to (values 0 through 255 in increments of 16) 1000, 1016, 1032, etc.</td>
</tr>
<tr>
<td>Increment 10</td>
</tr>
<tr>
<td>Decrement 10</td>
</tr>
<tr>
<td>Increment 5</td>
</tr>
<tr>
<td>Decrement 5</td>
</tr>
<tr>
<td>Increment 1</td>
</tr>
<tr>
<td>Decrement 1</td>
</tr>
</tbody>
</table>
**Devices - Zones**

These devices represent the various zones in the system, and you may change their name and location as you see fit. They are initially created by the plug-in using the location "HAI Zones" and with the name that you assigned to the zone.

Since there are several different parameters to a zone's status (e.g. latch alarm status, arming status, etc.) the various status values are combined in different groups to create the status values that the zones can be in, which are listed below.

Additionally, non-fire/CO2 zones (zones which cannot be bypassed) will have the device control option enabled, and the control options will include bypassing and restoring the zone. If the device's value is set to the value for bypass or restoral of the zone, note that the command will be carried out and the status will immediately change to the appropriate new value (e.g. Secure, Bypassed by User).

The string status and values are summarized in the following table:

```
Zone Device Status Table

<table>
<thead>
<tr>
<th>Status</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Secure</td>
<td>1</td>
</tr>
<tr>
<td>Not Ready</td>
<td>2</td>
</tr>
<tr>
<td>Trouble</td>
<td>3</td>
</tr>
<tr>
<td>Secure, Armed</td>
<td>11</td>
</tr>
<tr>
<td>Not Ready, Armed</td>
<td>12</td>
</tr>
<tr>
<td>Secure, Disarmed</td>
<td>21</td>
</tr>
<tr>
<td>Not Ready, Disarmed</td>
<td>22</td>
</tr>
<tr>
<td>Secure, Bypassed by User</td>
<td>31</td>
</tr>
<tr>
<td>Not Ready, Bypassed by User</td>
<td>32</td>
</tr>
<tr>
<td>Secure, Bypassed by System</td>
<td>41</td>
</tr>
<tr>
<td>Not Ready, Bypassed by System</td>
<td>42</td>
</tr>
<tr>
<td>Bypass</td>
<td>10</td>
</tr>
<tr>
<td>Restore</td>
<td>20</td>
</tr>
</tbody>
</table>
```

**Screens**

This plug-in registers some new screens for your HTML HomeSeer GUI. Unfortunately these screens cannot be duplicated in the Windows UI environment. These custom web pages allow you to view and command various features in your HAI system. All of the pages use the limits set in the limits screen described earlier.

Important Note: If you bypass the links provided by HomeSeer and use your own custom links with your HTML HomeSeer UI, then you will NOT see these new links appear automatically with the activation of the plug-in. You will need to modify your custom links page to add the appropriate URL (specified in the descriptions next) to your links.

The HTML Display and Control Screens are:

- Security Page
- HVAC (If you have thermostats or auxiliary temperature sensors)
- HAI Units
- Web Security Configuration

By default, users authenticating access to the HomeSeer web server as admin are granted access to all of the built-in web pages. Once you access the Web Security Configuration page, access to the built-in web pages is only granted to the users indicated on this page. Be careful that you do not access this page unless you are ready to configure the access rights or you could be locked out of further web page access.

**Security Page**

This page is at the URL: http://HSServerName/Security

This page will appear similar to this:

Security Link Page Example
The first section on this page shows the current panel arming status ("Armed Night" in the example above) and provides hyperlinks which may be clicked on to immediately perform the selected arm or disarm function.

The lower section lists each zone in the system (up to the zone limit entered) and provides the zone’s current status, latch alarm status, arming status, and analog value. Additionally, for all zones EXCEPT for fire zones, hyperlinks to bypass or restore each zone are provided as well. At the top of the zone list is a special entry which provides a hyperlink to restore all zones that may have been previously bypassed.

**HVAC**

This page is at the URL: http://HSServerName/HVAC and will ONLY appear if the number of thermostats or auxiliary temperature sensors configured in your system is one or more.

This page will appear similar to this:

![HVAC Link Page Example](image)

From this screen you can read all of the status information available from a thermostat connected to the system, and you can set/change all modes and setpoints as well. You can select a new mode by clicking on the down arrow of the drop down lists and you can enter a numerical value in the setpoint fields. Once you have made mode selections or entered setpoint values, click the “Make Changes” button and all of the changes made will be set to the appropriate thermostat.

**HAI Units**

This page is at the URL: http://HSServerName/HAIUnits

This page will appear similar to this:

![HAI Units Link Page Example](image)
This page lists all of the HAI “units” as well as “buttons”. The order in which they are listed are: X-10/ALC Units, Hardware Expansion Output (Flag) Units, Hardwire Outputs, Buttons, and Counter Units. To help identify this division using the above example, here are the cutoff points: The last X-10 unit is “Fireplace (9)”, the last hardware expansion output (used as flags) is “FLAG (140)”, the last physical hardwired output is “RELAY OUTPUT (196)”, the last button is “Away Nite (8)” and the last counter unit is “COUNTER (204)”. Following the name of each item is its item number in parenthesis. With the exception of buttons, all other items are units and follow the order from 1 to n where n is the maximum number of units your particular HAI model of panel supports. The number of units (n) is divided according to a table provided in the owner’s manual into groups for each unit type.

After the item name, the next two columns are the unit’s status and value respectively. The last column, labeled “Command” provides buttons for immediate operations appropriate for the unit type. At this time, X-10/ALC devices do not support the DIM or SET LEVEL commands, and counters may be set to zero (Off), incremented or decremented from this screen – to set them to a specific value, see the “actions” page.

**Web Configuration**

By default, users who access HomeSeer's HTML interface with an admin level access will be granted access to all HAI plug-in web pages. If the user authenticates as a guest, none of the plug-in web pages will be accessible. The web security configuration page allows you to configure admin users to have a secondary access check - e.g. the user "Sammy" can be configured to access the HVAC web page but not the security page, while the user "Richie" is allowed access to everything.

Once you access the web configuration page, it will appear similar to the one shown below - click on the parts of the image below to learn more about the various configuration options.
Triggers, Conditions, and Actions

The HAI plug-in supports a robust, feature rich set of triggers, conditions, and actions which are defined using HomeSeer’s Windows or HTML user interfaces. The UI presentation was optimized for the feature rich HTML environment and is the recommended UI to use when working with HAI triggers, conditions, and actions in events.

Triggers

There are several different categories of HAI triggers and several options in each category. The following table describes the category that the trigger falls into, and this corresponds with the bold category heading found on the trigger page for HAI System triggers. After the category the trigger, its condition and optional value are shown:

Table of HAI Plug-in Triggers

<table>
<thead>
<tr>
<th>Category</th>
<th>Trigger</th>
<th>First Parameter</th>
<th>Condition</th>
<th>Value Options</th>
</tr>
</thead>
<tbody>
<tr>
<td>Zone Triggers</td>
<td>Zone Status</td>
<td>Zone</td>
<td>Status (Secure, Not Ready, Trouble)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Zone Latch Alarm</td>
<td>Zone</td>
<td>Status (Secure, Tripped, Reset after Tripped)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Zone Arming</td>
<td>Zone</td>
<td>Status (Disarmed, Armed, Bypassed by User, Bypassed by System)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Zone Analog Value</td>
<td>Zone</td>
<td>Comparison (Becomes Less Than, Becomes Equal To, Becomes Greater Than)</td>
<td>Value (0 to 255)</td>
</tr>
<tr>
<td>Thermostat Triggers</td>
<td>Thermostat Temp</td>
<td>Thermostat</td>
<td>Comparison (Becomes Less Than, Becomes Equal To, Becomes Greater Than)</td>
<td>Value (0 to 255)</td>
</tr>
<tr>
<td></td>
<td>Thermostat Mode</td>
<td>Thermostat</td>
<td>Equal To (Off, Auto, Heat, Cool, Emerg/Aux)</td>
<td>Valid Temperature</td>
</tr>
<tr>
<td></td>
<td>Thermostat Fan</td>
<td>Thermostat</td>
<td>Equal To (On, Auto)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Thermostat Hold</td>
<td>Thermostat</td>
<td>Equal To (Holding, Not Holding)</td>
<td></td>
</tr>
</tbody>
</table>
Additional triggers may be set using the devices described in the “How it Works – Screens and Devices” section and the HomeSeer built-in “Device Value Change” trigger. Note: The device status (string) values do not conform to X-10 status values, so only the numerical values and the device value trigger may be used with these devices.

**Conditions**

As with triggers, several conditions have been provided in the HAI plug-in. These conditions may be OR’ed together to form complex checks on several zones in the HAI system or other important factors. Note that conditions placed on a trigger are checked immediately upon the trigger being met, but condition-only triggers are evaluated on a regular basis approximately once a minute. For this reason, it is not recommended that you try to OR together several zone status conditions hoping to fire an event if a zone status changes on one of several zones as it is very possible for a zone status to change and then change back to its previous value all between the time the condition-only triggers are checked. To trigger an event from a zone status change, create several events with one zone as the trigger and the other zones as conditions on the trigger.

Here is a description of the conditions available with the HAI plug-in:

**Table of HAI Plug-in Conditions**

<table>
<thead>
<tr>
<th>Condition Type</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>Zone Status=, Latch Alarm=, Arming Status= (Analog value is not available at this time.)</td>
<td></td>
</tr>
<tr>
<td>Thermostat Mode=, Fan=, Hold= (Temperature and setpoints are not available at this time.)</td>
<td></td>
</tr>
<tr>
<td>Unit Unit=ON, Unit=OFF</td>
<td></td>
</tr>
<tr>
<td>Counter Counter=0, Counter&lt;&gt;0</td>
<td></td>
</tr>
<tr>
<td>Counter Arming Condition Is (one of the panel arm status options), Is Not (one of the panel arm status options)</td>
<td></td>
</tr>
</tbody>
</table>

The options that were indicated as not being available are due to a current limitation of the plug-in API.

**Actions**

All of the actions possible with the HAI system were made available to the event actions page except having the HAI system do your taxes. Because the actions associated with an event are not “added” to a list like conditions are, several of the same type of action were provided in many cases so that several actions of an identical sort can be done without having to create many similar events. The following table describes the actions available and how many of each type of action you may set per event.

**Table of HAI Plug-in Actions**

<table>
<thead>
<tr>
<th>Action Category</th>
<th>Action</th>
<th>Parameter(s)</th>
<th>Number Available per Event</th>
</tr>
</thead>
<tbody>
<tr>
<td>Arm/Disarm</td>
<td>Arm/Disarm</td>
<td>Panel Security Mode</td>
<td>1</td>
</tr>
<tr>
<td>Thermostat</td>
<td>Raise Heat Setpoint by, Lower Heat Setpoint by</td>
<td>Degrees</td>
<td>4</td>
</tr>
<tr>
<td>Thermostat</td>
<td>Raise Cool Setpoint by, Lower Cool Setpoint by</td>
<td>Degrees</td>
<td>4</td>
</tr>
<tr>
<td>Thermostat</td>
<td>Raise Heat Setpoint to, Set Cool Setpoint to</td>
<td>Degrees</td>
<td>4</td>
</tr>
<tr>
<td>Thermostat</td>
<td>Enable Hold, Disable Hold</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>Thermostat</td>
<td>Set Fan Mode to Auto, On</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>Thermostat</td>
<td>Set Mode to Off, Auto, Cool, Heat</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>Zone Bypass, Restore</td>
<td>8 of each</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Button Push</td>
<td>4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Expansion Unit (Flag) Operation</td>
<td>Unit (Flag or Relay) Operation On, Off</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>Voltage Output</td>
<td>Output (Relay) Operation On, Off</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>Counter Operation (Set to value, Increment, Decrement)</td>
<td>Value</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>Messages Show/Clear a Message</td>
<td>Message to Show or Clear</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>Messages Set Message Text</td>
<td>Message and Text to set</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>(or device address to have it set to the contents of a device)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Lighting Device Control</td>
<td>Turn on, off, or set a light to a specific level. Light unit, operation (on, off)</td>
<td>8</td>
<td></td>
</tr>
</tbody>
</table>
**Note:** Thermostat actions DO NOT check the status of the HOLD setting and WILL carry out the desired action regardless of the thermostat hold mode. To honor the thermostat hold mode setting, add a hold status condition to the event trigger setting the condition to “Not Holding” for the desired thermostat.

**Scripting Interface**

The HAI plug-in has a full scripting interface so that you may write scripts to perform complex actions or to voice-enable your HAI system. Script commands with a version 2.0 plug-in API are written using the following format:

```plaintext
Return_value=object.Plugin("plug-in_name").sub_or_function(parameters)
```

- `Return_value` is a variable name to accept the returned value (if applicable for the sub or function).
- `Object` is the HomeSeer object reference, which is “hs” for scripts run by HomeSeer.
- `Plug-in_name` is “HAI_System” for the HAI plug-in.
- `Sub_or_function` is the name of the sub or function you wish to call.
- `Parameters` are parameters that may be required by the sub or function call

**Example:**

To retrieve the current temperature from the upstairs (zone 2) thermostat:

```plaintext
UpstairsTemp=hs.plugin("HAI_System").HAI_WhatIsTemp(2)
```

To retrieve the current security system status:

```plaintext
SecStat= hs.plugin("HAI_System").SecurityStatus
```

**Query Type Scripting Functions**

The commands under this section return information from the system and do not change or control items in the interface.

**Thermostat Query Commands**

**HAI_WhatIsTemp**

HAI_WhatIsTemp(zone as integer) as string

Returns the current temperature for a zone. Use `Val()` to convert it to a double (numeric) value.

**GetMode**

GetMode(zone as integer) as string

Returns the current operating mode for a thermostat zone. (Off, Auto, Cool, Heat, Emerg/Aux)

**GetFanMode**

GetFanMode(zone as integer) as string

Returns the current fan mode for a thermostat zone. Possible return values are On and Auto.

**GetHoldMode**

GetHoldMode(zone as integer) as string

Returns the current holding mode for a thermostat zone. (Holding, Not Holding)
HAI_WhatIsSetpoint

HAI_WhatIsSetpoint(zone as integer, mode as string) as string

Returns the current setpoint for a thermostat zone. Mode indicates the desired setpoint to retrieve, Heat or Cool. The first character is all that is checked and is “h” or “H” for heat, “c” or “C” for cool. Use Val() to convert the returned value to a double (numeric).

ThermostatName

ThermostatName(zone as integer) as string

Returns the assigned name of a given thermostat zone number.

WhatNumber

WhatNumber(thermostat as string, type as string)

Returns the thermostat zone number given a match of the provided thermostat name string. Name string is case insensitive, partial matches are allowed. The name provided must be specific enough to be unique amongst the others in its type class or the first one matching will be returned. Type is “T” or “t”.

Auxiliary Temperature Sensor Query Commands

GetAuxTemp_Temp

GetAuxTemp_Temp(zone as integer) as double

Returns the current temperature for an auxiliary temperature sensor zone.

WhatNumber

WhatNumber(sensor as string, type as string)

Returns the temperature sensor zone number given a match of the provided sensor name string. Name string is case insensitive, partial matches are allowed. The name provided must be specific enough to be unique amongst the others in its type class or the first one matching will be returned. Type is “A” or “a”.

Security Query Commands

SecurityStatus

SecurityStatus() as string

Returns the current security system arming status as a string. Possible values are:

Disarmed, Armed Day, Armed Night, Armed Away, Armed Vacation, Armed Day Instant, Armed Night Delayed, Armed Day (Delay), Armed Night (Delay), Armed Away (Delay), Armed Vacation (Delay), Armed Day Instant (Delay), Armed Night Delayed (Delay)

SecurityValue

SecurityValue() as integer

Returns the current security arming status value. Possible values are 0 (Disarmed), 1-6 (arming modes described in SecurityStatus), and 101-106 (arming modes during exit delay).

ZoneStatus

ZoneStatus(zone as integer, optional request as integer) as string
Returns the current status for a zone, default request type is 1 for zone status.

Request 2 is Arming Status

Request 3 is Latch Alarm Status

**ZoneValue**

ZoneValue(zone as integer) as integer

Returns the current analog value for a zone

**GetCodeInfo**

GetCodeInfo(code as string) as string

GetCodeInfo returns information about security codes. It is useful for determining the code number for a particular code, and the code number can be used with the HAI_Lookup_UserCode command to get the name for that user code.

The value returned is in the format "number,type" where the number is the user code number, and the type is the access level or type of access code.

Normal user codes are values 1-99. 0 is an invalid code, 251 is the duress code, and 999 is a special indicator for the Installer or PC Access codes.

The type for codes 1-99 is either **Master**, **Manager**, or **User**.

The type for code 251 is always **DURESS**

The type for code 999 can be **Installer**, **PC Access**, or **Installer & PC Access**

**Example Return Values:**

4,User

1,Master

999,Installer

**HAI_Lookup_UserCode**

HAI_Lookup_UserCode(code as integer) as string

Returns the user code name for a given user code number (1-255).

**ZoneName**

ZoneName(zone as integer) as string

Returns the assigned name for a given zone number.

**WhatNumber**

WhatNumber(zone as string, type as string)
Returns the zone number given a match of the provided zone name string. Name string is case insensitive, partial matches are allowed. The name provided must be specific enough to be unique amongst the others in its type class or the first one matching will be returned.

Type is “Z” or “z” for Zones.

Type is “C” or “c” for User Codes.

**Exit_Delay**

Exit_Delay() as integer

The Exit_Delay command should be used after the plug-in has fully initialized and obtained configuration information from the panel. Exit_Delay returns the value of the exit delay time (in seconds) that the panel is configured to use.

**Unit Query Commands**

**UnitStatus**

UnitStatus(unit as integer) as string

Returns the current status string for a unit.

**UnitValue**

UnitValue(unit as integer) as integer

Returns the current value for a given unit.

**UnitName**

UnitName(unit as integer) as string

Returns the assigned name for a given unit number.

**ButtonName**

ButtonName(button as integer) as string

Returns the assigned name for a given button number.

**WhatNumber**

WhatNumber(unit as string, type as string)

Returns the unit number given a match of the provided unit name string. Name string is case insensitive, partial matches are allowed. The name provided must be specific enough to be unique amongst the others in its type class or the first one matching will be returned.

Type is “X” or “x” for X-10/ALC Units.

Type is “E”, “e”, “F”, or “f” for Expansion/Flag Units.

Type is “H” or “h” for Hardware Output Units.

Type is “O” or “o” for Other Output Units (e.g. OmniLT class additional outputs).

Type is “I” or “i” for Internal Counters.

Type is “B” or “b” for Buttons.
System and Interface Query Commands

InterfaceStatus

InterfaceStatus() as integer

Returns the current HAI plug-in interface status. Possible values are: 0 – OK, 1 – Error sending data to the panel (in recovery mode), 2 – Error with the initialization (plug-in is non operational), 3 – Not Ready (plug-in is operational, but cannot go to fully operational mode until a configuration problem is addressed)

HAI_Version

HAI_Version() as string

Returns the HAI model number, model name in parenthesis, and panel firmware version and revision number.

Version

Version() as string

Returns the plug-in version number.

Command Type Scripting Functions

These script elements are commands which can change items in the HAI system.

Thermostat Commands

HAI_Lower_Temp

HAI_Lower_Temp(amount as integer, zone as integer, mode as string) as boolean

Lowers the temperature setting of a given zone by amount. Mode string first character of “h” or “H” for Heat, “c” or “C” for Cool, anything else to adjust the temperature for the thermostat’s current mode. Returns True if the command was successful.

HAI_Raise_Temp

HAI_Raise_Temp(amount as integer, zone as integer, mode as string) as boolean

Raises the temperature setting of a given zone by amount. Mode string first character of “h” or “H” for Heat, “c” or “C” for Cool, anything else to adjust the temperature for the thermostat’s current mode. Returns True if the command was successful.

HAI_Set_Fan_Mode

HAI_Set_Fan_Mode(mode as string, zone as integer) as boolean

Sets a thermostat zone’s fan mode. Returns True if the command was successful. Mode string first character of “O” sets the mode to ON, “A” sets the mode to Auto.

HAI_Set_Hold_Mode

HAI_Set_Hold_Mode(mode as integer, zone as integer) as boolean

Sets a given thermostat zone’s hold mode. Returns True if the command was successful. Mode of 0 is to disable holding, 1 enables hold mode.

HAI_Set_Temp

HAI_Set_Temp(temp as integer, zone as integer, mode as string) as boolean

Sets a given thermostat zone’s heating or cooling temperature setpoint. Returns True if the command was successful. Mode is “cooling” for cool setting (case insensitive) or anything else for heat setpoint.
HAI_SetThermMode

HAI_SetThermMode(mode as string, zone as integer) as boolean

Sets a given thermostat zone’s mode. Returns True if the command was successful. Mode first character is “A” for Auto, “O” for Off, “H” for Heat, “C” for Cool.

Security Commands

HAI_Arm_Disarm

HAI_Arm_Disarm(mode as integer, optional code as string) as boolean

Arms or disarms the system. Returns True if the command was successful. Mode is one of the following: 0 (Disarm), 1 (Arm Day Mode), 2 (Arm Night Mode), 3 (Arm Away Mode), 4 (Arm Vacation Mode), 5 (Arm Day Instant Mode), or 6 (Arm Night Delayed Mode)

If the optional code parameter is provided, the code is validated against the security system and the command will be carried out only if the code is a valid Master, Manager, or User code. If the code is not valid, a message is also written to the HomeSeer log.

HAI_RestoreZone

HAI_RestoreZone(zone as integer, optional All as boolean) as boolean

Restores a given zone. Returns True if the command was successful. Set All to True to restore all zones and the zone number is ignored.

HAI_BypassZone

HAI_BypassZone(zone as integer) as boolean

Bypasses a given zone. Returns True if the command was successful.

Unit Commands

HAI_Push_Button

HAI_Push_Button(button as integer) as boolean

“Presses” a given button number. Returns True if the command was successful.

HAI_Unit_On

HAI_Unit_On(unit as integer) as boolean

Turns on a given unit. Returns True if the command was successful. This command will not work with counters.

HAI_Unit_Off

HAI_Unit_Off(unit as integer) as boolean

Turns off a given unit. Returns True if the command was successful. This command will also work with counters to set them to zero.

HAI_Unit_SetLevel

HAI_Unit_SetLevel(unit as integer, percent as integer) as boolean

Sets a given unit to a percent level. Returns True if the command was successful. The percentage is any value from 0 to 100 inclusive.

HAI_Counter_Set

HAI_Counter_Set(unit as integer, value as integer) as boolean

Sets a counter to value. Returns True if the command was successful. Value is in the range 0 to 255.

HAI_Counter_Change

HAI_Counter_Change(unit as integer, direction as integer) as boolean
Increments or decrements a given counter unit. Returns True if the operation was successful. If direction is < 0, the counter is decremented. If direction is positive, the counter is incremented.

**Interface Commands**

**Set_Log_Debug**

Set_Log_Debug(setting as boolean)

Enables or disables logging of plug-in internal notations to the HAIPlugIn.log file in the HomeSeer directory. WARNING: Turning on this option will cause the HAIPlugIn.log file to grow very large very quickly. Please use this command under the supervision of HomeSeer Technical Support. It is not recommended that you enable this feature with a logging level (see the next command) greater than 2 for more than a few minutes.

This setting is instance temporary. That is, the log setting will revert back to the setting saved in the INI file the next time HomeSeer is started. To change the permanent setting, go to the Interfaces configuration tab in the HomeSeer Windows UI, and in the I/O section select “HAI_System” and then press “Setup”. On the configuration screen that will be displayed, there is a checkbox to enable or disable logging. When you press “OK”, this setting is saved in the INI file. Use the above script command to turn logging on or off at specific times to help capture a problem without generating a lot of useless log entries.

**Set_Debug_Level**

Set_Debug_Level(level as integer)

This determines the level of detail or information that is logged when logging is enabled. A value of zero has the same effect as calling Set_Log_Debug(False).

The valid range of values other than zero is 1 through 4. Here are the descriptions for each level of debugging. Note: As this information is designed to aid technical support, these descriptions provide a rough summary of the information logged and is not absolute. That is, there are some events and data values that are not logged or are logged under a different level than what these descriptions imply.

WITH EACH HIGHER LEVEL OF LOGGING SELECTED, ALL LOWER VALUE LEVELS ARE INCLUDED AUTOMATICALLY

1 = Events retrieved from the HAI Panel

2 = HAI Current Value Changes, which in turn includes some trigger and condition information

3 = Functions/Subs and Information relating to communications between the software and the HAI panel. This level generates a large amount of information in the log.

4 = Subroutine and Function Entrance and Exits. This level generates an enormous amount of information in the log. Not recommended for a weak computer or one with a slow hard drive!

**WriteHAILog**

WriteHAILog(info as text)

Writes the given text to the HAIPlugIn.log file with a date/time stamp.

**SetPanelDateTime**

SetPanelDateTime was added to set the security panel's date, time, and daylight savings mode to the HomeSeer computer's settings. The call should be made when the time is at exactly 0 seconds as the command to set the date/time/ds mode does not include the seconds. The script command is a function that returns a Boolean result of True/False indicating whether the command was a success or not. Here is an example (using VBScript):

```vbnet
Sub Main
    dim bRet
    dim HAI
    set HAI = hs.Plugin("HAI_System")
    bRet = HAI.SetPanelDateTime
    hs.WriteLog "HAI Time","Result is " & bRet
End Sub
```

Since the plug-in uses the computer's current date and time, it is important that not only is your computer's date and time set properly, but that the timezone is correct for the determination of daylight savings mode.

**Other Plugin Features**

There are other plugin features which you may take advantage of by adding configuration parameters to your HomeSeer configuration INI file. This file is located in your CONFIG directory and is called hspi_HAI.INI. In this file, locate the section (enclosed by braces) called hspi_HAI. All HAI plugin related configuration parameters are in this section.

**TemperatureUnit**
If you wish to have the temperature values displayed by the plug-in to be in Celsius, then set this INI to “c” or “C” such as:

TemperatureUnit=C

Any value other than “c” or “C”, or if it is omitted, will result in Fahrenheit for the temperature unit.

PollingInterval

This allows you to override the default polling interval for event messages from the panel. Event messages are what update most zones, units, buttons, etc. The default interval is 500ms, and you can set values in the range of 200ms to 5000ms (5 seconds). Polling too frequently (low value) will cause poor performance of the system and polling not often enough (high value) will cause the system to respond to changes very slowly. The syntax for using this parameter is as follows:

PollingInterval=x

(Where x is a numerical value from 200 to 5000)

gAutoRefresh

Even if HomeSeer's auto-refresh feature is disabled, you can still have this on your built-in HAI plug-in web pages by setting this value to the number of seconds you wish to have the page refresh interval set to. For example:

gAutoRefresh=30

... provides a 30 second automatic refresh rate.

Device Creation Control

The plug-in creates devices to represent various items controlled by the plug-in such as zone devices, unit devices, and thermostat devices.

If you are not using another plug-in or script in your system that requires the use of devices, you may OMIT the creation of these devices with the INI settings described in this section.

bUseDev_Z

You may use this INI setting to prevent the plug-in from creating ANY devices to represent the zones in the system. For control over individual zones, please refer to NoZoneDevice.

The format of the INI setting is:

bUseDev_Z=True|False

If you do not wish to have any zone devices created, use False.

bUseDev_U

You may use this INI setting to prevent the plug-in from creating ANY devices to represent the units in the system. For control over individual units, please refer to NoALCDevice, NoFlagDevice, NoOutDevice, and NoCountDevice.

The format of the INI setting is:

bUseDev_U=True|False

If you do not wish to have any unit devices created, use False.

bUseDev_T
You may use this INI setting to prevent the plug-in from creating ANY devices to represent the thermostats in the system. For control over individual thermostats, please refer to NoHVACDevice.

The format of the INI setting is:

```
bUseDev_T=True|False
```

If you do not wish to have any thermostat devices created, use False.

**bUseDev_A**

You may use this INI setting to prevent the plug-in from creating ANY devices to represent the ALC Lighting Units in the system. For control over individual ALC units, please refer to NoALCDevice.

The format of the INI setting is:

```
bUseDev_A=True|False
```

If you do not wish to have any ALC devices created, use False.

**bUseDev_O**

You may use this INI setting to prevent the plug-in from creating ANY devices to represent the output units in the system. For control over individual outputs, please refer to NoOutDevice.

The format of the INI setting is:

```
bUseDev_O=True|False
```

If you do not wish to have any output devices created, use False.

**NoZoneDevice-#**

You may use this INI setting to prevent the plug-in from creating a single zone device to represent the zone status and to provide device control over the zone.

The format of the INI setting is:

```
NoZoneDevice-#=True
```

Replace # with the zone number that you do NOT wish to have a device created to represent zone status and control. You may have as many of these entries in the INI file as you wish.

**NoZoneDevice-#**

You may use this INI setting to prevent the plug-in from creating a single zone device to represent the zone status and to provide device control over the zone.
The format of the INI setting is:

NoZoneDevice-#=True

Replace # with the zone number that you do NOT wish to have a device created to represent zone status and control. You may have as many of these entries in the INI file as you wish.

**NoALCDevice-#**

You may use this INI setting to prevent the plug-in from creating a single ALC lighting unit device to represent the unit status and to provide device control over the unit.

The format of the INI setting is:

NoALCDevice-#=True

Replace # with the unit number that you do NOT wish to have a device created to represent ALC device status and control. You may have as many of these entries in the INI file as you wish.

**NoALCDevice-#**

You may use this INI setting to prevent the plug-in from creating a single ALC lighting unit device to represent the unit status and to provide device control over the unit.

The format of the INI setting is:

NoALCDevice-#=True

Replace # with the unit number that you do NOT wish to have a device created to represent ALC device status and control. You may have as many of these entries in the INI file as you wish.

**NoOutDevice-#**

You may use this INI setting to prevent the plug-in from creating a single output unit device to represent the hardware output's status and to provide device control over the unit.

The format of the INI setting is:

NoOutDevice-#=True

Replace # with the unit number that you do NOT wish to have a device created to represent unit status and control. You may have as many of these entries in the INI file as you wish.

**NoCountDevice-#**

You may use this INI setting to prevent the plug-in from creating a single counter unit device to represent the counter status and to provide device control over the unit.
The format of the INI setting is:

NoCountDevice-#=True

Replace # with the unit number that you do NOT wish to have a device created to represent unit status and control. You may have as many of these entries in the INI file as you wish.

NoHVACDevice-#

You may use this INI setting to prevent the plug-in from creating devices to represent the thermostat status and to provide device control over the thermostat.

The format of the INI setting is:

NoHVACDevice-#=True

Replace # with the thermostat number that you do NOT wish to have a device created to represent thermostat status and control. You may have as many of these entries in the INI file as you wish.

Polling Control

There are several INI settings that control the system polling. Polling is a critical component of the plug-in because it is the only method for communicating with the panel.

Here is an overview of how polling works in the plug-in:

At the primary polling interval, which is 500ms by default, the plug-in polls the panel for any events that have occurred. Events include zones changing status, units turning on and off (but not dim/bright), buttons being pushed, or security events.

After checking and processing these events, the plug-in then executes 50 secondary polling cycles. The secondary polling has two functions - first, it updates the status of items that are never reported in a system event (e.g. thermostat status), and second it makes sure that the status of all of the zones and units in the system is correct.

All of the zones, units, thermostats, security status and special alarms are added together to determine the maximum number of ‘widgets’ that are polled on each cycle - this number can be as large as 520 items in a fully loaded large panel.

When the secondary polling group is 1, the system status is obtained. When it is 2, the status of the special alarms (e.g. freeze alarm) is obtained. For polling groups 3 through n, the polling is done as follows:

When polling zones, the status of several zones is requested at once rather than taxing the panel by requesting the status of each individual zone. If you have 16 zones in use in your panel, then 50 secondary poll iterations will include a single zone status request for zones 1 through 16. If you have 80 zones in your system, then the first secondary poll iteration will retrieve the status of zones 1 through 50, and the next one will get zones 51 through 80. Unfortunately, the panel cannot handle status requests of such large numbers of zones (or units, or thermostats, etc.) and therefore the plug-in has INI settings to control the number of widgets to request the status of for each item type in the panel. These INI settings can be adjusted to achieve optimum performance of the plug-in and the polling.

Secondary polling quantity for Units, Zones, and Temperature.

If your Poll_U setting is changed from its default setting of 25 to a setting of 40, then the status of up to 40 units will be requested all at once from the panel.

If you exceed the default values that the polling control settings provide by a great deal, you run the risk of receiving many checksum errors as the panel is unable to assemble the response message due to the overhead of other processing that needs to be done. This will cause your secondary polling to be inconsistent. If you decrease the default polling control settings provided by a significant amount, then you will affect the total amount of time it takes the panel to get the status of items that do not generate system events. For example, if you change a thermostat's cool setpoint, then this will typically be reflected in the status of the thermostat within about 20 seconds using the default value, but could take more than a minute if the plug-in is requesting too few status update items at a time.

Poll_U
This controls how many units to request the status for during a secondary polling interval as described here.

Poll_U=#

# represents the value you wish to set. The default value for units is 25. This value may not be less than 1 or greater than 50 - if a value is specified outside of this range, the default value will be used.

**Poll_Z**

This controls how many zones to request the status for during a secondary polling interval as described here.

Poll_Z=#

# represents the value you wish to set. The default value for zones is 16. This value may not be less than 1 or greater than 50 - if a value is specified outside of this range, the default value will be used.

**Poll_T**

This controls how many thermostat/temp sensors to request the status for during a secondary polling interval as described here.

Poll_T=#

# represents the value you wish to set. The default value for thermostats/temp sensors is 4. This value may not be less than 1 or greater than 12 - if a value is specified outside of this range, the default value will be used.

**Support**

For Support:

Help Desk: www.homeseer.com (select HelpDesk from the Support menu)

Support is also available on our message board at:

http://board.homeseer.com

Look for the forum dedicated to the HAI System plugin.