MaxAir Technical – Software Install

A number of shell scripts are available to both perform configuration tasks or install additional software. At the time of writing, three scripts are available:

- /var/www/api/enable_rewrite.sh
- /var/www/var/www/add_on/amazon_echo/install.sh
- /var/www/add_on/homekit/install.sh

The scripts can be executed from a terminal session using the command bash 'script path', for example ***'bash /var/www/add_on/amazon_echo/install.sh'***

These scripts can also be executed from the user interface, see document '**software_install'.**

/var/www/api/enable_rewrite.sh

Used to configure the Apache web-server re-write function to allow PHP files in the /var/www/api directory to be executed without the need for the '.php' filename extension.

The script performs the following tasks:

1. If not already modified then modify - /etc/apache2/sites-available/000-default.conf
2. If not already modified then modify - /etc/apache2/sites-enabled/000-default.conf
3. Enable Apache mod-rewrite if not already enabled
4. If any changes have been made then restart Apache web-server

/var/www/add_on/amazon_echo/install.sh

Used to install the 'fauxmo' add-on to enable boost control through the use of the Amazon Echo Home Assistant.

The script performs the following tasks:

1. Exposes the service name associated with the add-on
2. Execute the shell script /var/www/api/enable_rewrite.sh to check/enable Apache mod-rewrite
3. Install 'pip3' if not already installed
4. Install fauxmo application software
5. Add Unprivileged Fauxmo User
6. Change Privileges for Fauxmo
7. Back Up and Update /etc/fauxmo/config.json, Adding Accessories for Each Zone
8. Create Unit File for service 'pihome_amazon_echo.service'
9. Enable the service
10. Start the service

/var/www/add_on/homekit/install.sh

Used to install the 'homebridge' add-on to enable boost control through the use of Apple's HomeKit application, which includes 'Siri' voice control.

The script performs the following tasks:

1. Exposes the service name associated with the add-on
2. Execute /var/www/api/enable_rewrite.sh to check/enable Apache mod-rewrite
3. Install the latest version of 'nodejs' dependant on processor type (armv6l for RPi Zero or else armv7l)
4. Use npm to install the 'homebridge' application software
5. Setup the Homebridge service
6. Use npm to install the WebHooks Plugin
7. Create the WebHooks Cache Directory
8. Back Up and Update /var/lib/homebridge/config.json, adding WebHooks Plugin and Accessories for Each Zone
9. Restart the Hombridge service

Installing from The User Interface

The user interface provides a means to execute any of the three script files, see document '**software_install'.** When a script file is selected for execution the full path name of the script file is placed in the 'script' field of a new row in the table '**sw_install**'. This table is checked once every 10 seconds using the task scheduler to execute the Python script ***/var/www/cron/sw_install.py***, the script checks the field '**pid**' for the last '**sw_install**' table entry and if found to be NULL it executes the installation script file as a background task, updating the '**pid**' and '**start_datetime**' fields with the PID for the installation script and the current datetime. The Python script loops until the PID is no longer current, indicating that the background script has finished, it then updates the '**stop_datetime**' field in the '**sw_install**' table. The user interface displays a dialogue indicating the script start, this is updated with the result of the script execution on completion.