

Installing a Regulus 1.0.1 Plugin

Revision A
February 2015
Part Number 06932

LIQUID
ROBOTICS

Installing a Regulus 1.0.1 Plugin

Copyright © 2015 Liquid Robotics, Inc.

Confidential computer software. Valid license from Liquid Robotics, Inc. required for possession, use or copying. Consistent with FAR 12.211 and 12.212, Commercial Computer Software, Computer Software Documentation, and Technical Data for Commercial Items are licensed to the U.S. Government under vendor's standard commercial license.

Trademarks

Liquid Robotics®, LRI, Wave Glider®, Wave Glider Management System, WGMS, and SHARC® are worldwide trademarks of Liquid Robotics, Incorporated. Other products and company names mentioned herein may be the trademarks of their respective owners.

Patents

The Wave Glider mechanism is patented under US Patent Number 7,371,136 and other patents pending. All aspects of the Wave Glider product have been developed at private expense by Liquid Robotics, Inc.

Warranty

The information contained herein is subject to change without notice. The only warranties for Liquid Robotics, Inc. products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as a warranty. LRI, Inc. is not liable for technical or editorial errors or omissions contained herein.

Revision History

Document Part Number	Revision	Publication Date	Changes
06932	A	February 2015	This is the initial release of this document.

Table of Contents

1. Introduction	1
2. Installing a Plugin	4
3. Deleting a Plugin	8
4. Plugin Configuration Options	9
5. Troubleshooting	12
Plugin Initialized but Disabled	12
Plugin Still Booting	13
A. Issuing a Write Console Script from the Vehicle Web Page	15

Chapter 1. Introduction

Regulus plugins can be installed on a Wave Glider SV3 from the vehicle Web page. A vehicle's Web page is accessed by connecting to the vehicle's wireless network, which has the same name as the vehicle. For example "SV3-032."

The vehicle Web page can also be accessed over a cellular network. Go to <http://swab2.liquidr.local/metadata/index.html> (the default login is **hawaii** and **f=ma rules**). After login, click on the vehicle name, then click on the vehicle's IP address to open its Web page. The following is an example of the vehicle Web page for SV3-009:

Introduction

LIQUID ROBOTICS

SV3-409331995 "SV3-009"

888 574 4574 (US)

927Wh

VR

INET

WP072

0.4km

WiFi

support@liquidr.com

2014-08-14T16:35:18.542Z

SV3-009: Overall vehicle status

Status

Config

Operate

Setup

Help

● airmarWaterspeed

● airmarWeatherDeviceDriver

● ais

● AIS

● AISReporter

● AMPManager

● AMPSPowerWrapper

● bulkDataExport

● canTransport

● cellModem

● commandReporter

● communicationsManager

● configuration

● configurationReporter

● CPUtempMonitor

● deviceManager

● ethernetSwitch

● fileManager

● fileSystemCleanup

● floatAhrs

● floatRecoveryDevice

● G1

● G2

● genericAlarmReporter

● GPSAirmarWeather

● GPSTorpedoGPS

● GPSUblox

● heading

● headingAirmarWeather

● headingHost_32_4

● headingRudder

● http

● internetAccessibility

● iridiumModem

● lightBar

● linuxState

● linuxWatchdog

● location

● log

● memoryMonitor

● navigation

● navigator

● PDCFilesystem

● PDCParser

● PeriodicThruster_1_0_periodicThruster

● picture

● radarReflector

● rudder

● runshell

● SI

● scripting

● serviceManager

● setParametersListener

● setStatusHandler

● smc

● speedOverGround

● SV2PayloadManager

● SV3PayloadManager

● TI

● telegraphOffice

● telemetry06Reporter

● thruster

● time

● torpedoGPS

● ubloxDeviceDriver

● umbilical

● uniqueIDManager

● uniview

● uptime

● vehicleHealth

● vehicleID

● waterSpeed

● weatherReporter

● weatherStation

● wirelessAccessPoint

● debugClient

● debugLAN

● debugServer

● G3

● lightBar

● S2

● S3

● virtualRelay

● otherSample

● sample

8/14/2014 9:35:16 AM

8/14/2014 9:35:17 AM

8/14/2014 9:35:17 AM

8/14/2014 9:35:17 AM

8/14/2014 9:35:19 AM

8/14/2014 9:35:20 AM

8/14/2014 9:35:21 AM

8/14/2014 9:35:21 AM

8/14/2014 9:35:22 AM

8/14/2014 9:35:22 AM

Happy

VWVH-Handler

GPGBA-Handler

PSH-Handler

MMSI 57624893 (unnamed), Pos=-99.534293, -104.799267, t=0.0, NOT_DEFINED

[Host(Node=2): P=119.8 kPa, T=30.9-C; Not responding: 11-12,14-20,32,34; Uptime: 0 days, 08:35:41] [Thruster(Node=5): P=147.2 kPa, T=26.5-C; HW OK; Uptime: -23 days, -571:34262-20557261]; HW OK; No Manifest]

Host: data.liquidr.com; Streaming: track rudderDebug alarmLog log :-)

2315 cantransport, Uptime: 20 days, 22:19:12

ON (ON) Requested by: operator Requires: ethernetSwitch

Last MD5: 3D 2B BA B4 12 C2 D8 FC C0 E6 7E 71 E7 02 80 2F

[virtualRelay, iridiumModem] No Default Channel, Last Command: Follow Sequential Course (68-73, target=73). Have 0 blocked messages.

Last MD5: 1D E9 AA 00 50 4E D0 60 3F 84 7E E0 67 07 36 6A

59.2-C

ON (ON) Requested by: cellModem

Free space: /dev/mmcblk0p3: 9Gb rootfs: 2Gb

heading=-0.100000, pitch=214748368.000000, roll=214748368.000000, numTurns=2147473152.000000, data is 0 old

OFF

On ip=192.168.103.0 mask=255.255.255.0 no payloads found in box

On ip=192.168.105.0 mask=255.255.255.0 no payloads found in box

reportAlarmsToShore=true, reportEventsToShore=true, reportTelemAlarmsToShore=true

At [20.060263333333333;-156.04967666666667]@2014-08-14T16:35:19.300Z, quality=80

At [19.99642785;-155.91350516666667]@2014-08-12T15:59:42Z, quality=65

At [20.060255833333333;-156.04966933333333]@2014-08-14T16:35:19Z, quality=100

Sub: HeadingRudder* Float: HeadingAirmarWeather*, HeadingHost_32_4

Heading 213.7 (turns=655.594)

Heading 9.5 (turns=212.026)

Heading 211 (turns=907.586)

pingInterval=30 min, autostartVPN=enabled, vpnStatus=openvpn running, defaultProvider=cellmodem, maxAllowablePingFailures=10, timeBetweenPingAttempts=30, totalNumFailedPingAttempts=0

ON. 0 queued msgs (Max=1), Channel is TALKING, IMEI:300434060709190, consecutiveErrors=0, sessionErrors=20494, writeErrors=0, readErrors=0

OFF

Aug 11 21:12:11 (none) kernel: [5892328.537231] (stc): add_channel_to_table: id 9

Watchdog Timeout = 300 sec

Using GPSUblox declination=9.626485

7.6Mb free, 120Mb max, 20.3Mb total

Targeting waypoint 072; desired heading=222.04884; speed-to-goal:1.3013706

Speed-to-goal = 1.3 knots

reportRate=2m

Power OFF, Automatic, status=0x00000000

2014-08-14T16:35:21.795Z

15568 wit283gps, GPRMC-Handler

GPZDA-Handler

ON (ON) Requested by: operator rudder

Float: 215-Sub: 218-Twist: 0.0

Uptime: 69 days, 20:58:06

pressureLimit: high=130.000000, low=105.000000; temperatureLimit: high=45.000000, low=5.000000; batteryLow=5.000000

SV3-009, Running Regulus software version of 2014-06-05_10-02-43, production build 292

0.9 min: 0.37 max: 1.88 mean: 0.92

reportLengthMinutes=10 T=25.6,WS=5.2,GS=7.8,WD=242.4,P=1.0, @ [20.060254999999998;-156.049685]@2014-08-14T16:35:22.300Z

Temp=25.6-C, Press=1022.0mb, Wind Speed=4.49kn, Wind Dir=257.4. Heading=215.7.

WAP is On

ExitedUnhappy

OFF (ON), not AMPSP Requires: ethernetSwitch

OFF (ON), not AMPSP Requires: ethernetSwitch

OFF (ON), not AMPSP Requires: ethernetSwitch

Off ip=192.168.104.0 mask=255.255.255.0 no payloads found in box

OFF (OFF)

Device is Off. Port is Off, []

Device is Off. Port is Off, []

Happy

ON. 0 queued msgs (Max=2147483647), Channel is TALKING.(44/45 success rate) sleepTime=300000 packetSize=8192

Dead

/root/Regulus/lib

another sleep

Recent log entries

rudder Decouple => 2.4029174

virtualRelay IOException on socket read

virtualRelay sendBytes threw null, cannot talk to WGMS

virtualRelay VR Error when sending bytes to shore

canTransport 2315 cantransport, Uptime: 20 days, 22:19:12

AMPSPManager Setting Desired Heading to 212.5 degrees

rudder Decouple => 2.2453578

rudder rack angle: 4.9; post angle: 4.4; Heading: 218.62648, Desired Heading: 221.12648, av decouple angle: 1.9960209 (calibrated), status=0x3400810B

torpedoGPS 15568 wit283gps

uptime Uptime: 69 days, 20:58:06

SV3

009

SV3

009

java

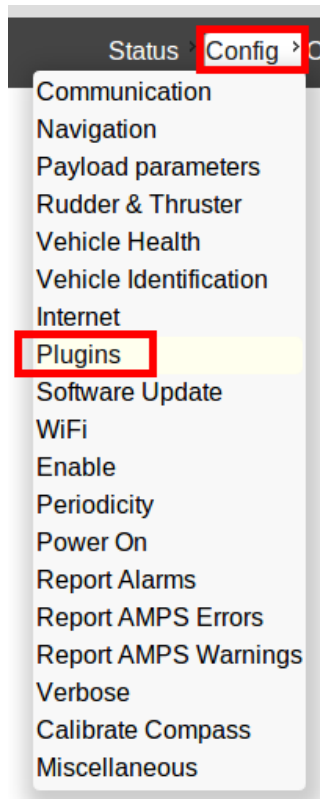
POWERED

java

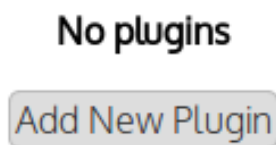
POWERED

2

From the vehicle's Web page, open the **Config** drop-down menu and click on **Plugins**:



The **Add New Plugin** window will appear:

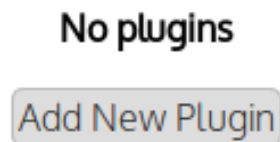


You can add as many plugins as you need, but you must add them one at a time.

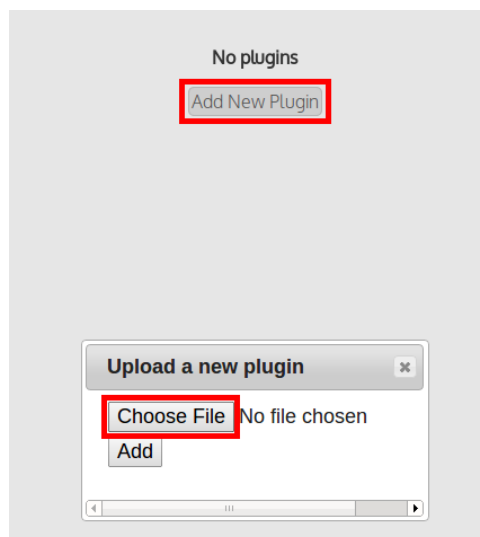
Chapter 2. Installing a Plugin

Plugins are installed by clicking on **Plugins** in the **Config** drop-down menu of the vehicle Web page to open the **Add New Plugin** window.

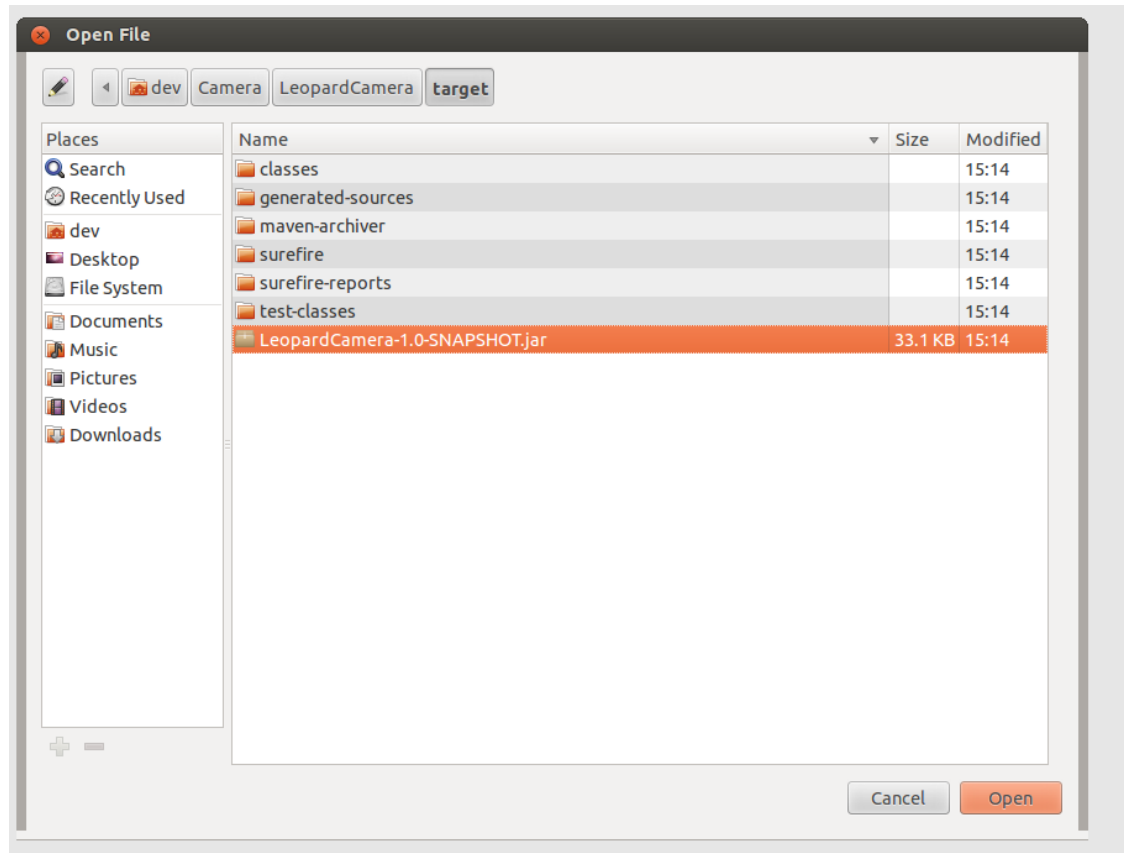
1. There may or may not be plugins already installed. If there are plugins installed, they will be listed above the **Add New Plugin** button, if not, **No plugins** will be displayed:



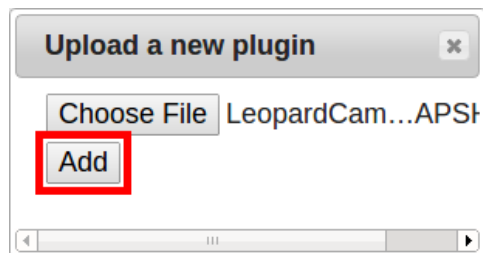
2. Click on **Add New Plugin** and the **Upload a new plugin** dialog will appear:



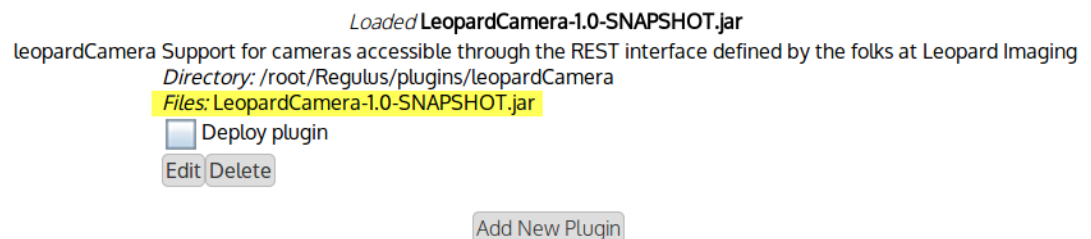
3. Click on **Choose File** in the **Upload a new plugin** dialog and a file browser will appear. This is where you select a plugin's JAR file, for example `LeopardCamera-1.0-SNAPSHOT.jar`:



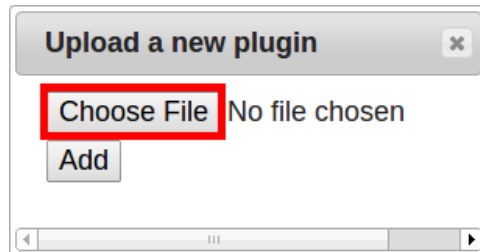
4. Click on **Add** in the **Upload a new plugin** dialog:



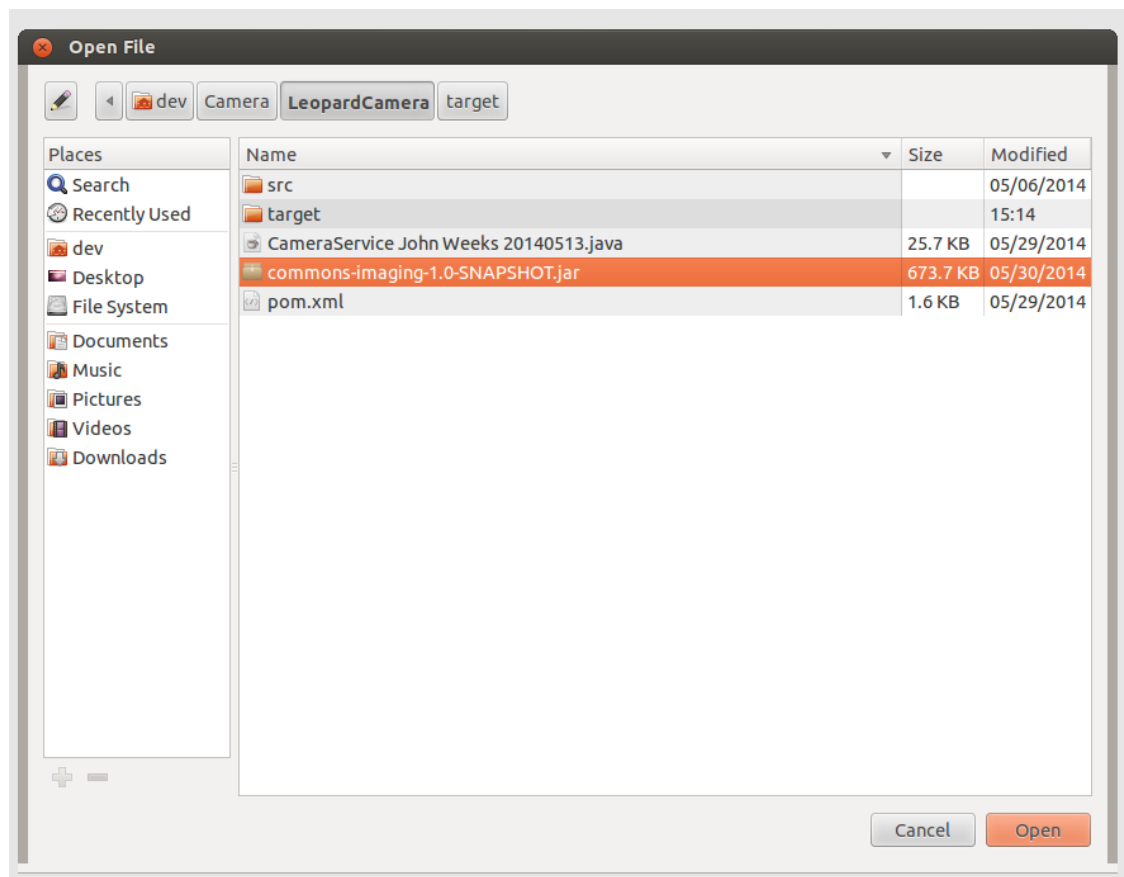
5. At this point the plugin has been added to the vehicle, but it is undeployed, as indicated by the empty **Deploy plugin** check-box in the loaded plugin description that appears at the top of the Add New Plugin window:



6. Include optional or required files that are associated with the plugin. Some plugins may have optional or required files that need to be added to the vehicle. These files are added in the same way that plugins are added. For example, the Camera plugin requires the commons-imaging-1.0-SNAPSHOT.jar library. Click **Choose File** in the **Upload a new plugin** dialog to open a file browser:



Select the commons-imaging-1.0-SNAPSHOT.jar file:



7. Click **Add** in the **Upload a new plugin** dialog. Now the commons-imaging-1.0-SNAPSHOT.jar file is ready to deploy along with the Leopard Camera plugin (LeopardCamera-1.0-SNAPSHOT.jar):

leopardCamera Support for cameras accessible through the REST interface defined by the folks at Leopard Imaging
Directory: /root/Regulus/plugins/leopardCamera
Files: LeopardCamera-1.0.1-SNAPSHOT.jar, commons-imaging-1.0-SNAPSHOT.jar
☐ Deploy plugin
Edit Delete

Add New Plugin

8. Click on the **Deploy plugin** check-box. Note, while the plugin is loading the check-box will display a question mark:

Loaded commons-imaging-1.0-SNAPSHOT.jar (no regulus.manifest)
leopardCamera Support for cameras accessible through the REST interface defined by the folks at Leopard Imaging
Directory: /root/Regulus/plugins/leopardCamera
Files: commons-imaging-1.0-SNAPSHOT.jar, LeopardCamera-1.0-SNAPSHOT.jar
☐ ? Deploy plugin
Edit Delete

Add New Plugin

9. When the plugin is loaded the Deploy plugin check-box will display a check mark:

Loaded commons-imaging-1.0-SNAPSHOT.jar (no regulus.manifest)
leopardCamera Support for cameras accessible through the REST interface defined by the folks at Leopard Imaging
Directory: /root/Regulus/plugins/leopardCamera
Files: commons-imaging-1.0-SNAPSHOT.jar, LeopardCamera-1.0-SNAPSHOT.jar
☒ Deploy plugin
Edit Delete

Add New Plugin



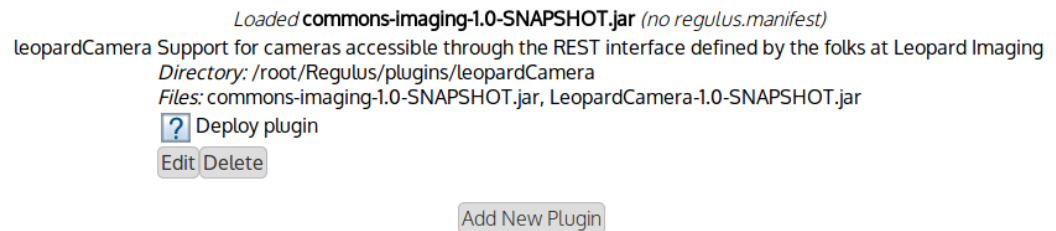
Note

At this point the plugin has been **deployed**, but it will not operate until it is **enabled**. See [Chapter 4: Plugin Configuration Options](#) for information on configuring and enabling plugins.

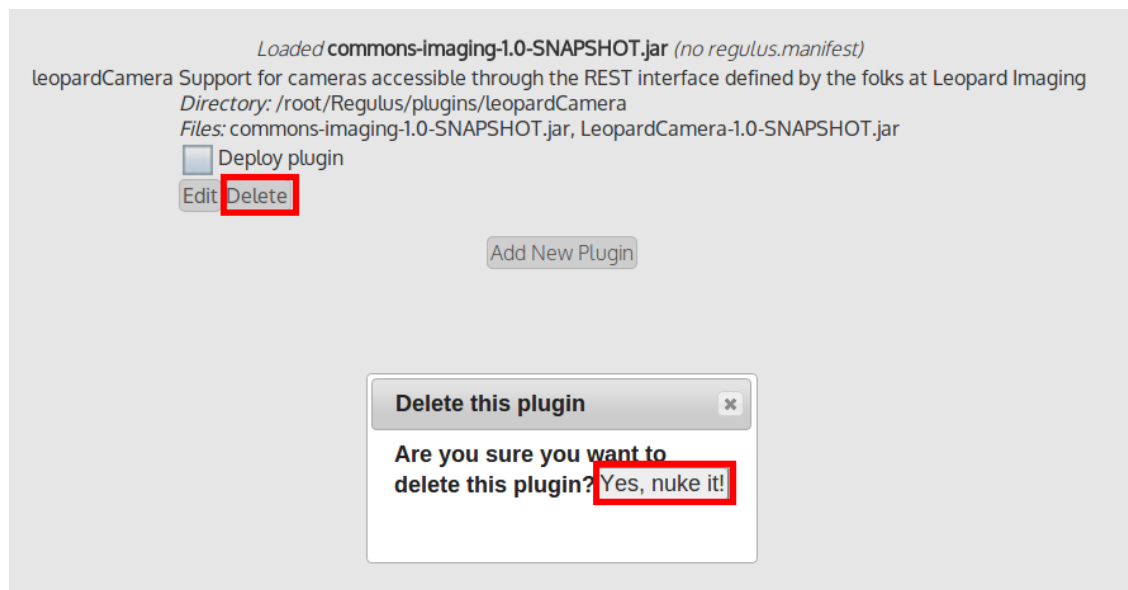
Chapter 3. Deleting a Plugin

While a plugin is deployed, it may be necessary to update it to a newer version. To do this you must delete the older version of the plugin and then add and deploy the newer version. Deleting a plugin is a two step process, first undeploy and wait for confirmation, then delete.

1. Undeploy and wait for confirmation by clicking on the checked **Deploy plugin** check-box - a question mark will appear temporarily while waiting for the plugin to undeploy:



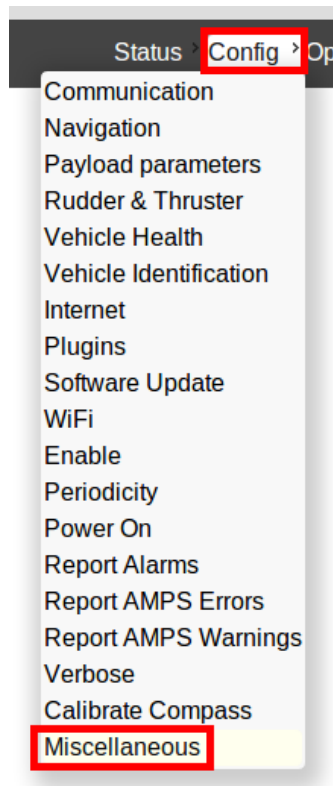
2. When the question mark is gone from the **Deploy plugin** check-box, click on the **Delete** button to open the **Delete this plugin** dialog. Click **Yes, nuke it**, and then the plugin will be deleted:



Chapter 4. Plugin Configuration Options

Once a plug-in is deployed, there may be configuration options you can select.

1. From the vehicle Web page, open the **Config** drop-down menu and click on **Miscellaneous**:



2. On the **Miscellaneous** page, all of the configuration options for each plugin are associated with the plugin name. **When you first deploy a plugin, the default options will include a check-mark in the check-box preceding the line item.** You have the option to modify the plugin configuration by adding or removing check-marks on this Web page. After confirming or changing the default configuration for a plugin, it must be enabled before it will begin operating. The default configuration for the **leopardCamera** plugin is shown in the following example of a vehicle's **Miscellaneous** Web page.

Plugin Configuration Options

Miscellaneous

☐ genericAlarmReporter: Report Alarms to Shore

☒ genericAlarmReporter: Report Events to Shore

☒ genericAlarmReporter: Report Telem06 Alarms to Shore

http: HTTP service port

☒ leopardCamera: A camera is present on this vehicle

☐ leopardCamera: Automatically take pictures

☒ leopardCamera: Deploy plugin

leopardCamera: Down scale factor for creating thumbnails

leopardCamera: If the scheduled picture period is less than this many minutes, the camera will be kept on in between photos.

leopardCamera: If the scheduled picture period is less than this many seconds, it's too short between photos.

leopardCamera: Number of seconds to pause to take a second picture after the first. 0=>don't take a second picture.

leopardCamera: Number of seconds to pause to wait for the glider to align to the target.

leopardCamera: plugin name/var/tag

☒ leopardCamera: Save photos in storage path

☐ leopardCamera: Send picture to bulk data collector

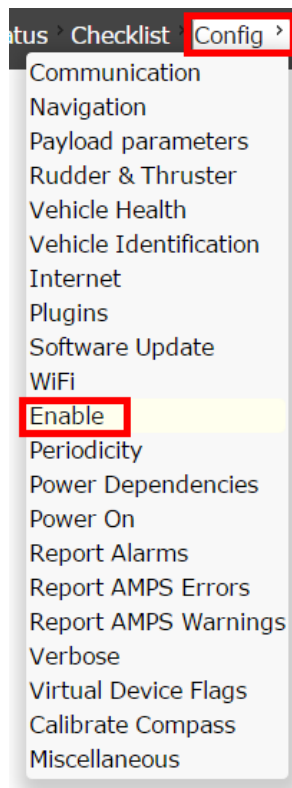
leopardCamera: Storage path to save pictures and thumbnails

leopardCamera: The expansion port to which the camera is connected.(G1, G2, G3)

leopardCamera: The IP address of the camera.

leopardCamera: Time (in minutes) between taking pictures.

3. To enable a plugin, from the vehicle Web page open the **Config** drop-down menu and click on **Enable**, then click in the check-box associated with the plugin you want to enable and a check-mark will appear indicating that the plugin has been enabled.



	Enable
☒ airmarWaterspeed	☒ airmarWeatherDeviceDriver
☒ bulkDataExport	☒ canTransport
☒ communicationsManager	☒ configurationReporter
☒ debugLAN	☐ debugServer
☒ expansionPort1	☒ expansionPort2
☒ fileManager	☒ fileSystemCleanup
☒ genericAlarmReporter	☒ http
☒ leopardCamera	☒ lightBar
☒ payloadPort1	☒ payloadPort2

4. Once enabled, the plugin begins it's startup sequence. If it is successful, it may report one or more events to WGMS.

For example, if the camera plugin startup sequence is successful, in the **Command Reason** column of the commands log on the WGSM Home page, the camera will:

- Report a Photo Startup Success event to WGMS.
- Upload a thumbnail image to WGMS and upload a high quality version of that photo to swab2.
- Report Camera Info including:

OK biosversion=LI-M38-IMX136-P

OK hardwareversion=M38_V11

OK softwareversion=1.2.064 build 1651

OK sensor=IMX136_2_3MP

OK pro_sn=1016000085 (this serial number is a unique ID that can be used to identify the camera).

SV3-032 Commands		
SV3-032 Commands		
 	Time Stamp	Command Reason
	02/04/2015 21:45:02	Camera Info: OK biosversion=LI-M38-I...
	02/04/2015 21:45:00	Photo Startup Success: Took a photo!
	02/04/2015 21:44:26	sending command file
	02/04/2015 21:42:59	Andrew Trapani issued webCmd:install...
	02/04/2015 21:42:43	Andrew Trapani issued webCmd:install...
	02/04/2015 21:42:37	Andrew Trapani issued webCmd:delete...
 Page 1  Goto 		

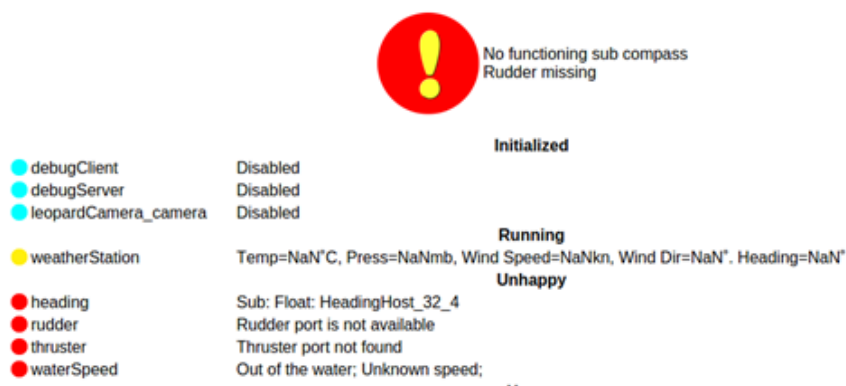
Chapter 5. Troubleshooting

To verify if a plugin has been enabled, go to the vehicle's **Overall vehicle status** Web page, using a cell or WiFi communications channel. If the plugin was successfully enabled, a green dot will appear next the plugin name. If the plugin failed to boot, a turquoise dot will appear next to the plugin name - one of the following procedures may resolve the issues.

Plugin Initialized but Disabled

If the vehicle's **Overall vehicle status** Web page displays a turquoise dot next to the plugin name, the plugin was initialized, but it was not enabled. Go to [Chapter 4: Plugin Configuration Options](#) and follow step 3 and step 4 to enable the plugin.

SV3-032: Overall vehicle status



If the plugin is still not working (Turquoise for Disabled), the following procedure may solve the problem:

1. **Restart Regulus** -there are two ways to do this:

a. **Turn the Vehicle Off and On**

Unplug the VMC

Plug it back in

b. Issue the following WGMS **Write Console Script** command:

```
runshell /etc/init.d/regulus restart
```

2. Wait for the Regulus to come back online.

3. Log in to the vehicle Web page using a cell or WiFi communication channel.

4. Run a series of WGMS write console script commands to **board id = 0 / task id = 0** and observe the responses, for example:

- `%leopardCamera_camera.getState()`

Response = **Runnable** - This is the desired response.

Response = **Initialized** - Resend the command, the plugin is still booting.

- `%leopardCamera_camera.startSS()`

Response = **OK** - This is the desired response.

- `%serviceManager.getState()`

Response = **Happy** - This is the desired response.

- `%serviceManager.getInfoString()`

Response = **Everything has started** - This is the desired response.

If all of the command responses you get match the ones displayed above, the plugin is properly installed.



Note

For information on issuing Write Console Script commands from the vehicle's Web page, see: [Appendix A: Issuing a Write Console Script from the Vehicle Web Page](#)

5. Go to [Chapter 4: Plugin Configuration Options](#) and complete step 4 to verify that the start up sequence worked.
6. Login to the vehicle's **Overall vehicle status** Web page and make sure there is a green dot next to the plugin name, for example:

● iridiumModem	adler integrity=10, timeBetweenMsgAttempts=00 ON. 0 queued msgs (Max=1). Channel is TALKING. s=810, writeErrors=0, readErrors=0
● leopardCamera_camera	<u>Camera is present, but must be manually triggered</u>
● lightBar	OFF

Plugin Still Booting

If the Overall Vehicle Status page looks like the following example for the leopardCamera, the plugin was unable to start (turquoise) because it is relying on other services to complete the boot sequence. In this case, the plugin is waiting for the location service (the GPS). If it has been 15 minutes since the boot attempt began, the vehicle is not receiving a GPS signal. Contact the Liquid Robotics Support Organization for assistance.

SV3-061: Overall vehicle status

Still Booting

Waiting for leopardCamera_camera(location), Sentinel_1_0_USSIAcousticSensor(heading,location)



No functioning sub compass
No functioning GPS
Compass is uncorrected

● leopardCamera_camera location
● Sentinel_1_0_USSIAcousticSensor heading,location

Initialized

● weatherStation

Temp=NaN°C, Press=NaNmb, Wind Speed=NaNkn, Wind Dir=NaN°, Heading=NaN°

Running

● heading
● HeadingHost_32_4
● HeadingRudder
● location
● serviceManager
● waterSpeed

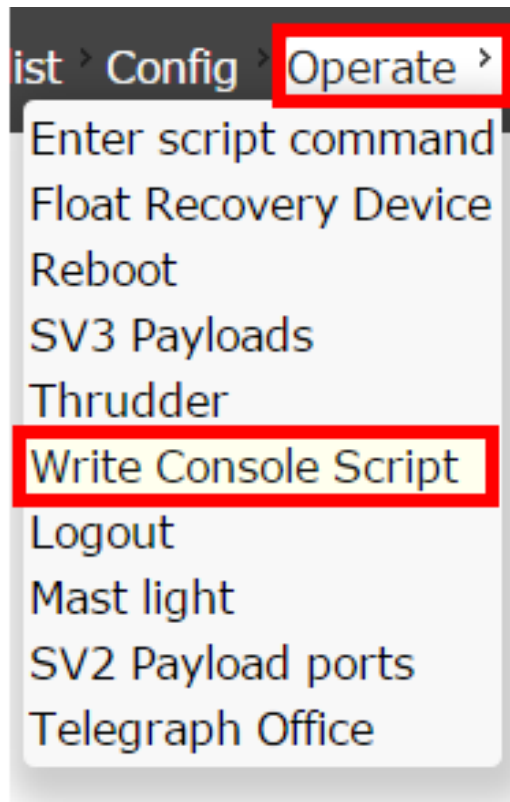
Unhappy
Sub: HeadingRudder Float: HeadingHost_32_4
Heading NaN (turns=NaN)
Heading NaN (turns=NaN)
No location provider available
Waiting for leopardCamera_camera(location), Sentinel_1_0_USSIAcousticSensor(heading,location)
Out of the water; Unknown speed;

Appendix A. Issuing a Write Console Script from the Vehicle Web Page

This section requires running Write Console Script commands on the vehicle. This can be done from the vehicle's Web page or from WGMS.

To issue a Write Console Script command from the vehicle's Web page:

1. Open the vehicle Web page, click on the **Operate** menu then select **Write Console Script**:



2. Enter a console command, for example, `%serviceManager.getState()`:

Write Console Script:

Board Address:

Function Code:

3. Click **Send Write Console Script Command** and wait. If nothing is returned after a few seconds, click the **Refresh** button to get the response:

Write Console Script Command Sent

Command Script: `runshell serviceManager.getState()`
Board Address: 0
Function Code: 0

Response

Refresh

4. In the following example, the **Happy** response indicates that the plugin is running properly.

Write Console Script Command Sent

Command Script: `%serviceManager.getState()`
Board Address: 0
Function Code: 0

Response

Happy

Refresh