

CRICKETS AUTOMATION

Training and Tutorials

xvisionIOT

Running on

Docker container

Property notice

The information published in this document contains rights and industrial secrets of Crickets Automation or its suppliers and cannot be reproduced in any way or transmitted to others, nor be made available in archiving systems.

The information is provided solely for the purpose of evaluating X Vision products and may not be used for other purposes.

The information contained in this document is subject to change without notice. Crickets Automation assumes no responsibility for any errors that this document may contain.

Unless otherwise specified, any reference to companies, names, data and addresses used in screen reproduction and in the examples is purely coincidental and has the sole purpose of illustrating the use of the Crickets Automation product. No part of this document may be reproduced or transmitted or made available in filing systems in any form or electronic or mechanical means, for any use without the written permission of Crickets Automation.

Copyright © 2019-2020, Crickets Automation. All rights reserved.

Registered trademark

All trademarks and registered trademarks are the property of their respective owners.

Abstract

This guide explains how to run xvisionIOT on Docker container for ARM v7 host.

Docker is an impressive technology that allows applications to run in separated containers, dramatically increasing security and easy of use.

Setup of Docker Container is done by Portainer, a simple and effective software to manage images and containers.



This technical note applies to xvisionIOT for ARM v7 cpu.

1

Container setup

If you have got a xvisionIOT license protected by USB hardware dongle, follow the instructions below. If you have opted for a software key, follow the instructions below ignoring hardware key, then proceed to step 2).

- 1) Insert USB Protection dongle in a USB port of your appliance
- 2) Start **Portainer** and login *
- 3) In *Portainer* menu select *Containers* and choose *+Add container*
- 4) Set Name with a string of your choice (i.e. xvisionIOT)
- 5) Set Image to **crickets/xvisioniot** (use latest version available)
- 6) In **Network Ports configuration** add the following ports mapping using *+publish a network port*:
 - 8010 as 80
 - 8011 as 443
 - 27015 as 27015
 - 502 as 502 (Modbus slave, if needed)

portainer.io

Create container
Containers > Add container

Portainer support admin
my account log out

Home LOCAL

Dashboard

App Templates

Stacks

Containers

Images

Networks

Volumes

Events

Host

SETTINGS

Extensions

Users

Endpoints

Name: xvisionIOT

Image configuration

Image: crickets/xvisioniot. Registry: DockerHub

Always pull the image ?

Network ports configuration

Publish all exposed network ports to random host ports ?

Manual network port publishing

host	8010	→	container	80	TCP	UDP	<input type="button" value="trash"/>
host	8011	→	container	443	TCP	UDP	<input type="button" value="trash"/>
host	27015	→	container	27015	TCP	UDP	<input type="button" value="trash"/>

Note: you must remap to unassigned ports (free ones)

7) Go to *Advanced container settings*. Choose *Command & logging* and select *Interactive & TTY*

Advanced container settings

Command & logging Volumes Network Env Labels Restart policy Runtime & Resources Capabilities

Command e.g. /usr/bin/nginx -t -c /mynginx.conf

Entry Point e.g. /bin/sh -c

Working Dir e.g. /myapp User e.g. nginx

Console **Interactive & TTY (-i -t)** Interactive (-i) None TTY (-t)

8) *Network*: leave *bridge*

9) *Env*: set the following using *+add environment variable*

TZ = Europe/Rome (choose your timezone)

SIMULATION = 1 (if you want to start demo simulation)

Advanced container settings

Command & logging Volumes Network **Env** Labels Restart policy Runtime & Resources Capabilities

Environment variables + add environment variable

name	TZ	value	Europe/Rome	
name	SIMULATION	value	1	

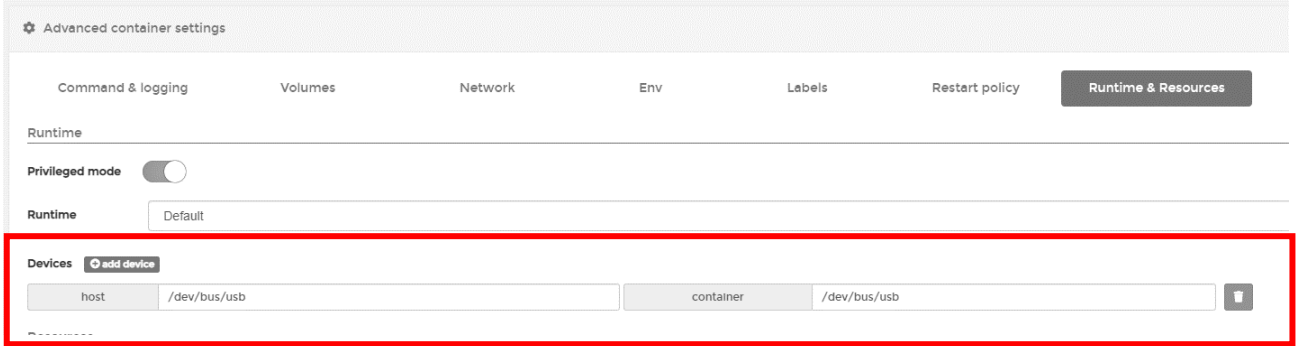
10) *Restart Policy*: select *Always*

Advanced container settings

Command & logging Volumes Network Env Labels **Restart policy** Runtime & Resources Capabilities

Restart policy Never **Always** On failure Unless stopped

11) *Runtime & Resources*: switch on *Privileged mode* and Add device host =**/dev/bus/usb** container=**/dev/bus/usb**



12) In *Actions* select *Deploy* the container to create and run the new container



Warning: the new created container has no project inside, so you must publish the project of your choice.

(*) run Portainer from Putty, if not already running:

```
docker run -d -p 8000:8000 -p 9000:9000 -v /var/run/docker.sock:/var/run/docker.sock -v portainer_data:/data portainer/portainer
```

2

Software key setup

If you have got a xvisionIOT license protected by a software key, follow the instructions below.

Warning: software key is available only for Raspberry compatible devices. Please check before buy.

- 1) Remove xvisionIOT USB protection dongle, if any, and restart the container
- 2) In Portainer select your xvisionIOT container and open its console by pressing ">_" located on the same row, then press *Connect*

Name	State	Quick actions	Stack	Image	Created	IP Address
angry_thompson	created		-	portainer/portainer	2020-02-01 11:31:17	172.
awesome_rubin	created		-	portainer/portainer	2020-01-21 09:37:39	172.
xvisionIOT	running	>_	-	crickets/xvisioniot:2.0.8	2020-02-01 16:43:20	172.
nifty_ganguly	running	>_	-	portainer/portainer	2020-02-01 11:59:18	172.
netpi-nodered	running	>_	-	hilschernetpi/netpi-nodered.latest	2019-10-19 13:25:09	-
festive_meitner	stopped		-	a5dc2f30e19a	2020-02-01 15:15:39	-
affectionate_spence	stopped		-	portainer/portainer	2020-01-27 11:29:42	172.
friendly_spence	stopped		-	portainer/portainer	2020-01-20 14:20:26	172.
beautiful_meninsky	stopped		-	portainer/portainer	2020-01-11 17:17:49	172.
blissful_khorana	stopped		-	portainer/portainer	2020-01-10 15:52:31	172.

>_ Execute

Command

Use custom command

User

Connect

- 3) Move to xvisionIOT directory `cd /data/webhmi`
- 4) Get super user privileges: `sudo -s`
- 5) Run `./ExportLicense`, that creates the `hardwareID.dat` file
- 6) Edit `hardwareID.dat` and copy its content (**do not modify it**):

nano hardwareID.dat



```
GNU nano 2.7.4 File: hardwareID.dat
P4Aq6Xl5ViS32wLKu/yzquMpT5Eb0cvCt5M27VoNtM=
[ Read 1 line ]
^G Get Help      ^C Write Out    ^W Where Is    ^K Cut Text    ^J Justify    ^C Cur Pos
^X Exit          ^R Read File    ^\ Replace     ^U Uncut Text  ^T To Spell   ^_ Go To Line
```


select all the text with the mouse, then copy it to clipboard (right click copy), next paste it in the body of the email you will send to Crickets Automation to request authorization (info@crickets-it.com).

Close editor without saving (**CTRL+X**)

- 7) Wait for unlock code from Crickets. Crickets will send you the unlock code after receiving the hardwareID as at point 5).
- 8) Create the unlock file (if exists, first rename or move it to a backup location) and edit it:
`cd /data/webhmi`
`nano license.dat`
- 9) Copy the unlock code in the clipboard and paste it in the `license.dat` file. Save and close the file (**Ctrl+X**, type **Y**, **Ctrl+O**).















10) Restart your container selecting *Containers* from menu, then select the container and press *Res*

Container list  Portainer support  admin
[my account](#) [log out](#)

Containers Columns  Settings

▶ Start ■ Stop ☠ Kill **↺ Restart** ⏸ Pause ▶ Resume 🗑 Remove + Add container

Q Search...

<input type="checkbox"/>	Name	State  Filter 	Quick actions	Stack	Image	Created	IP Adc
<input type="checkbox"/>	angry_thompson	created	 	-	portainer/portainer	2020-02-01 11:31:17	172.1
<input type="checkbox"/>	awesome_rubin	created	 	-	portainer/portainer	2020-01-21 09:37:39	172.1
<input checked="" type="checkbox"/>	xvisionIOT	running	   	-	crickets/xvisioniot:2.0.8	2020-02-01 16:43:20	172.1
<input type="checkbox"/>	nifty_ganguly	running	   	-	portainer/portainer	2020-02-01 11:59:18	172.1

Verify your license

1) browse license information typing in the container console:

cat /data/webhmi/license.inf

```
root@120d3b3021fa:/data/webhmi# cat license.inf
Serial=0391831001
Credits=4096
Clients=3
Internet clients=4
Pages=2
Drivers=22
Devices=15
C - Runtime=
D - DataLogger=
E - Trend=
G - Recipe=
N - EMail=
R - Script=
Driver=MOD
Driver=MODTCP
Driver=SIETCP
root@120d3b3021fa:/data/webhmi#
```

Cricket's Automation

Headquarters:

via Brunelli Maioni, 44
28021 Borgomanero (NO) Italy

Contacts:

tel. +39 0322836020
tel. +39 0322846938

Web:

email info@crickets-it.com
web www.crickets-it.com