

Table des matières

Prérequis	1
Côté PC principal :	1
Côté PC sur machine virtuelle	1
Configuration réseau Windows.....	1
Configuration réseau Machine Virtuelle sous linux (Debian).....	2
Division du flux vidéo et activation	4

Le but de ce rapport est de montrer toutes les configurations nécessaires au bon fonctionnement de la caméra afin de pouvoir accéder à la vidéo via QGroundControl et un code Python sur une machine virtuelle en parallèle.

Prérequis

Côté PC principal :

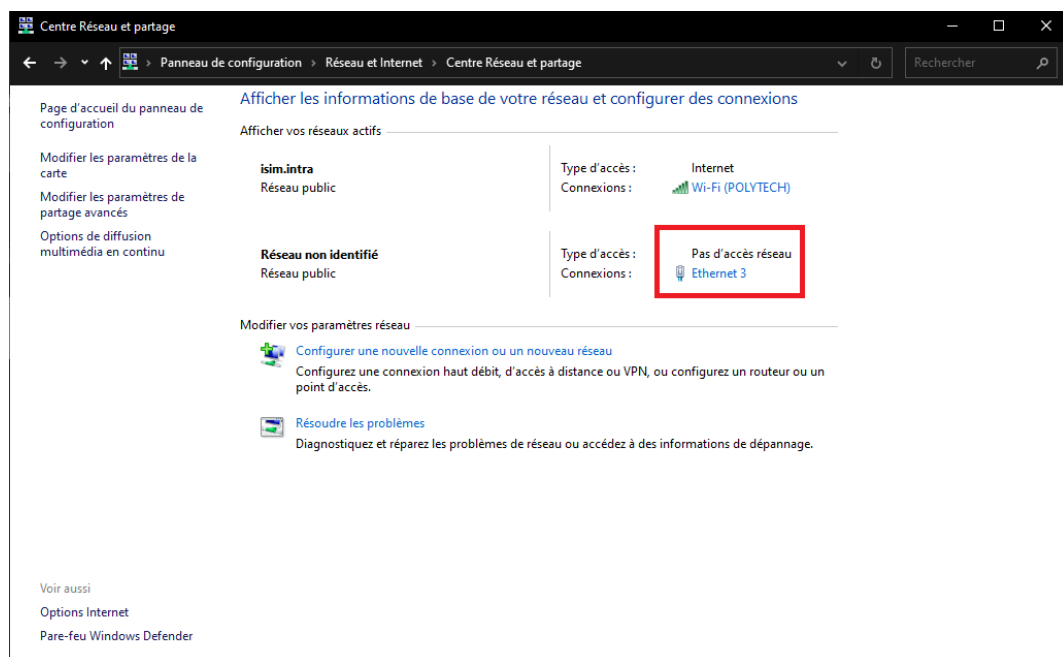
Rien en particulier.

Côté PC sur machine virtuelle

- Python 2
- Les librairies gstreamer, OpenCV 2 et Numpy

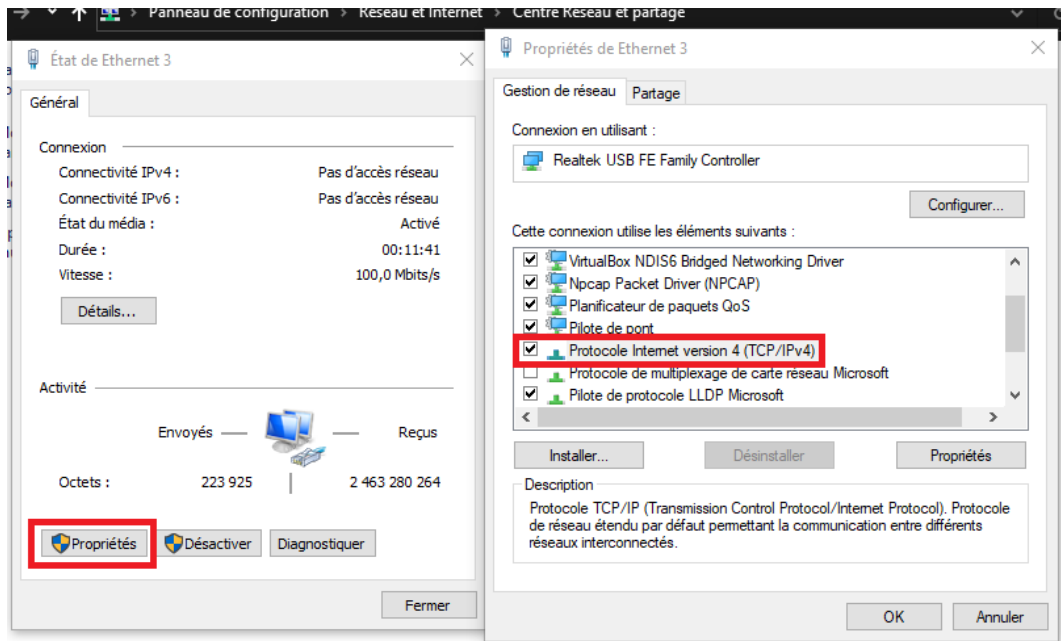
Configuration réseau Windows

Aller dans le panneau de configuration, dans Réseau et Internet puis dans Centre de Réseau et partage après avoir connecté la prise USB du Bluerov alimenté. Vous devriez voir un réseau Ethernet comme sur l'image ci-dessous.

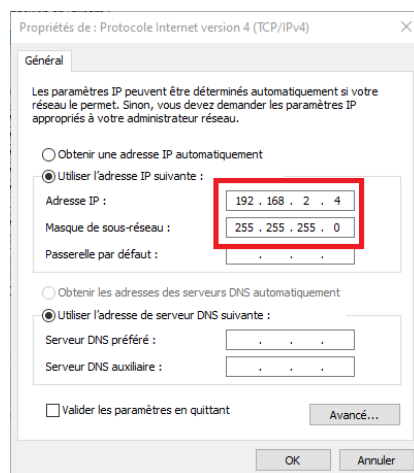


Cela va ouvrir une fenêtre, cliquez sur Propriété puis sur Protocole Internet version 4 (TCP/IPv4).

Accès caméra bluerov via VM



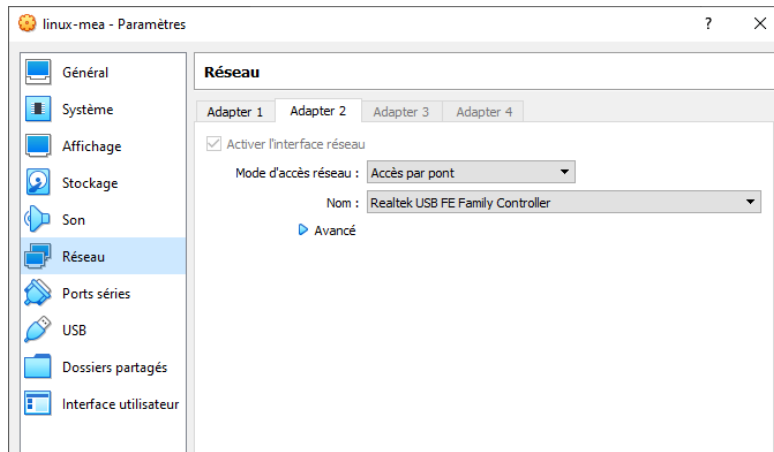
Une petite fenêtre va s'ouvrir afin d'entrer les paramètres comme sur la capture d'écran ci-dessous. Ne choisissez pas 192.168.2.1 ou 192.168.2.2 puisqu'elles sont respectivement réservées à la machine virtuelle et au robot.



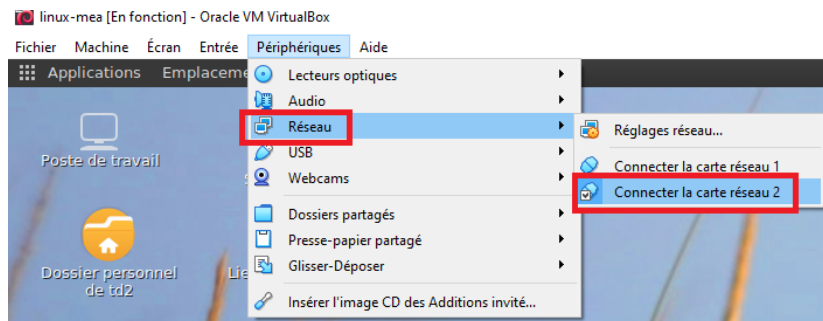
Configuration réseau Machine Virtuelle sous linux (Debian)

Ici l'exemple sera appliqué sous une distribution Debian tournant sous une machine virtuelle avec VirtualBox. Le plus important est de comprendre la procédure afin de pouvoir le reproduire sous vos conditions à vous. Allez dans les paramètres de votre machine virtuelle, dans le menu Réseau afin d'ajouter un nouvel adaptateur. Celui-ci devra être configuré en accès par pont (bridge) en sélectionnant Realtek USB FR Family Controller afin que le réseau créé par le Bluerov passe directement dans la machine virtuelle.

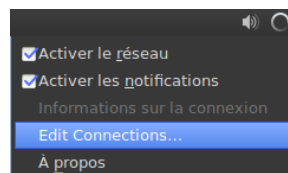
Accès caméra bluerov via VM



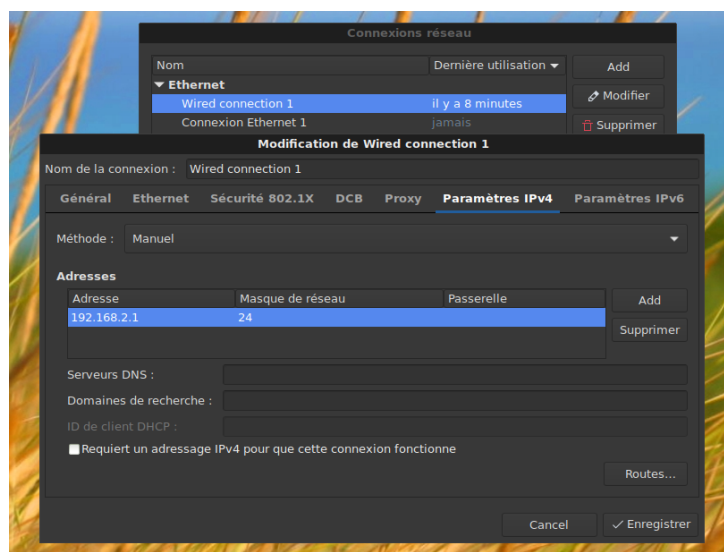
Lancez la machine virtuelle et sélectionnez dans les paramètres l'activation de l'adaptateur réseau créé.



Faites ensuite clic droit sur l'icône du réseau et entrez dans les paramètres en cliquant sur Edit Connections.

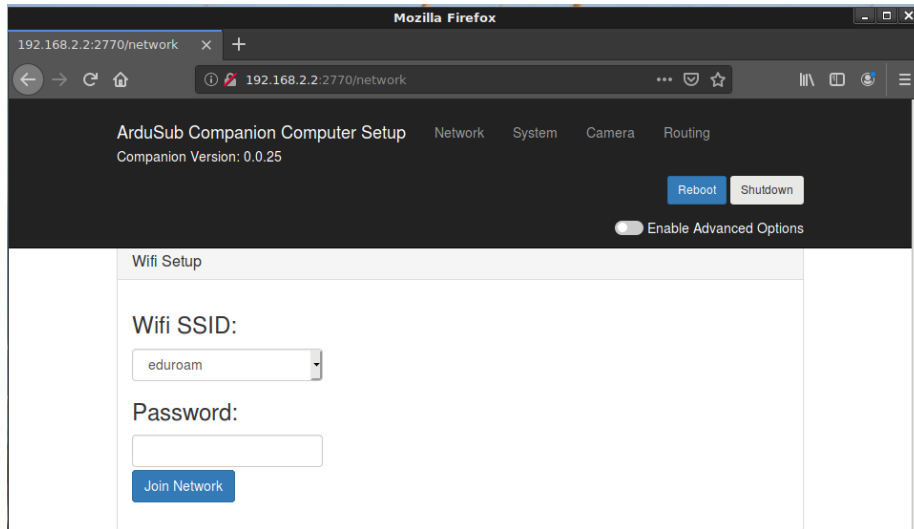


Editez l'adaptateur réseau en cliquant sur Modifier et dans l'onglet Paramètres IPv4, ajoutez l'adresse 192.168.2.1 ainsi qu'un masque réseau de 24.



Accès caméra bluerov via VM

Validez et connectez-vous sur le réseau modifié en cliquant sur l'icône réseau ainsi que sur le nom de l'adaptateur. Ici Wired connection 1. Vérifiez ensuite la validité de votre connexion en ouvrant un navigateur et en tapant 192.168.2.2 :2770/network. Cela devrait vous amener sur la page ci-dessous.



Division du flux vidéo et activation

Pour cette partie, il faut d'abord avoir fait les configurations données dans les deux parties précédentes. Connectez-vous sur 192.168.2.2:2770/camera et mettez ces paramètres et validez :

! multiudpsink clients=192.168.2.4:5600,192.168.2.4:4777,192.168.2.1:5600,192.168.2.1:4777

Camera Setup

Streaming Settings

Active Camera: /dev/video1

Format: H264

Frame Size: 640x480

Frame Rate (FPS): 30

gstreamer options:

! h264parse
! queue
! rtph264pay config-interval=10 pt=96
! multiudpsink
clients=192.168.2.4:5600,192.168.2.4:4777,192.168.2.1:5600,192.168.2.1:4777

Default gstreamer options

Apply gstreamer options

Nous avons mis 192.168.2.4 afin de lancer QGround Control sur Windows en parallèle du script Python sur la machine virtuelle.

Sur la machine virtuelle, lancez le script python, si tout se passe bien, une fenêtre devrait apparaître avec le flux vidéo de la caméra. Si ce n'est pas le cas et qu'il n'y a pas de message d'erreur, il y a très certainement un problème de configuration quelque part.