

---

Article

[Jose-Tomas Salvador](#) · Apr 8, 2020 6m read

## Configure an environment to use Docker on Hyper-V Ubuntu virtual machines running on a Windows 10 host

This time I want to talk about something not specific to InterSystems IRIS, but that I think is important if you want to work with Docker and your server at work is a PC or laptop with Windows 10 Pro or Enterprise.

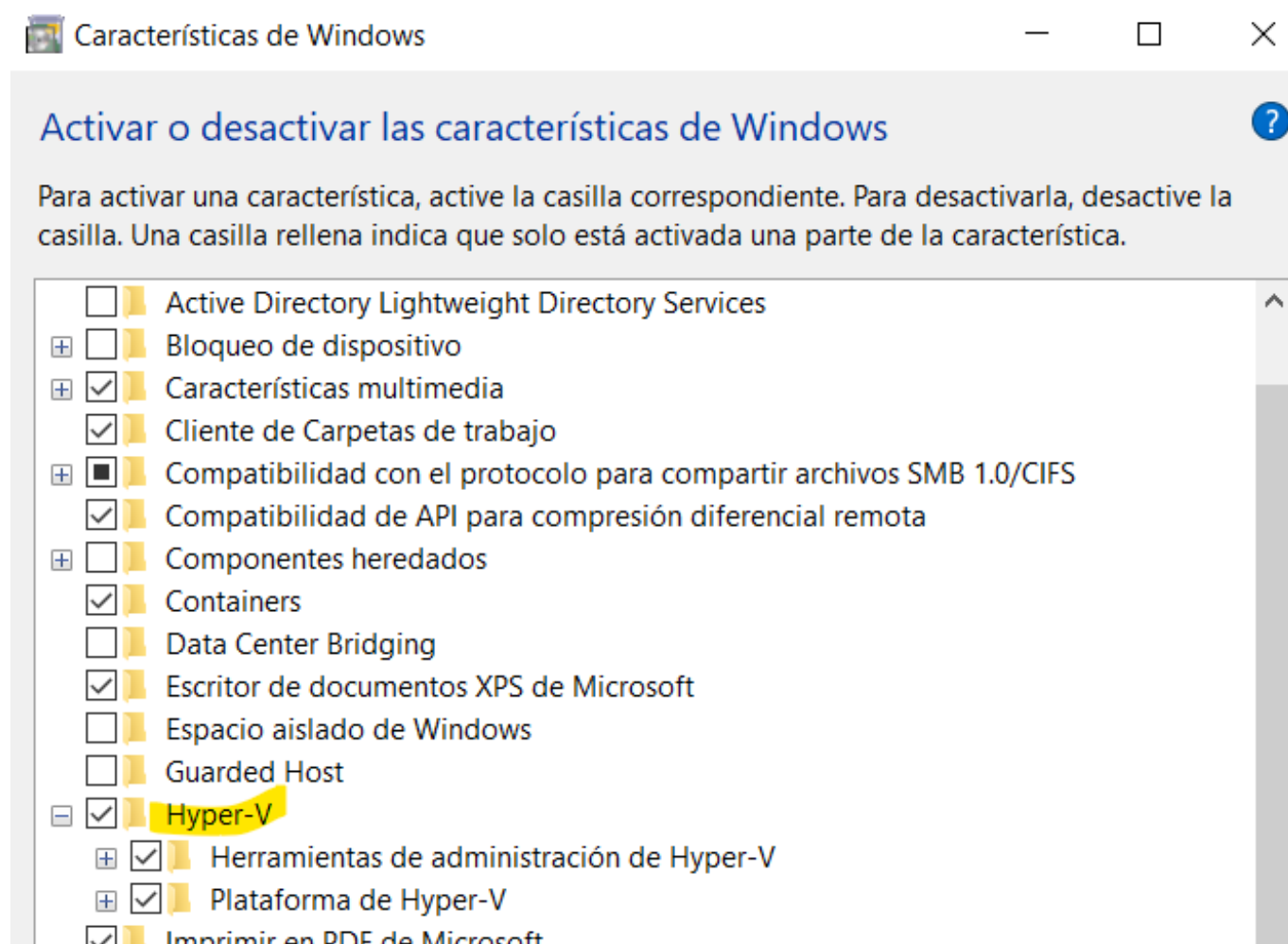
As you likely know, containers technology comes basically from Linux world and, nowadays, is on Linux hosts where it shows maximum potential. Those who use Windows on a normal basis see that both, Microsoft and Docker, have done important efforts during these last years that allow us to run containers based on Linux images on our Windows system in a really easy way... but it's something not supported for production systems and, this is the big problem, is not reliable if we want to keep persistent data outside of containers, in the host system,... mostly due to the big differences between Windows and Linux file systems. In the end, Docker for Windows itself uses a small Linux virtual machine (MobyLinux) to run the containers... it does it transparently for the Windows user... and it works perfectly well if, as I said, you don't require that your databases survive longer than the container...

Well,...let's get to the point,... the point is that many times, to avoid issues and simplify, we need a full Linux system and, if our server is based on Windows, the only way of having it is through a virtual machine. At least till WSL2 in Windows is released, but that will be another story and sure it'll take a bit of time to become robust enough.

In this article, I'll tell you, step by step, how to install an environment where you'll be able to work, if you need it, with Docker containers on an Ubuntu system in your Windows server. Let's go...

### 1. Enable Hyper-V

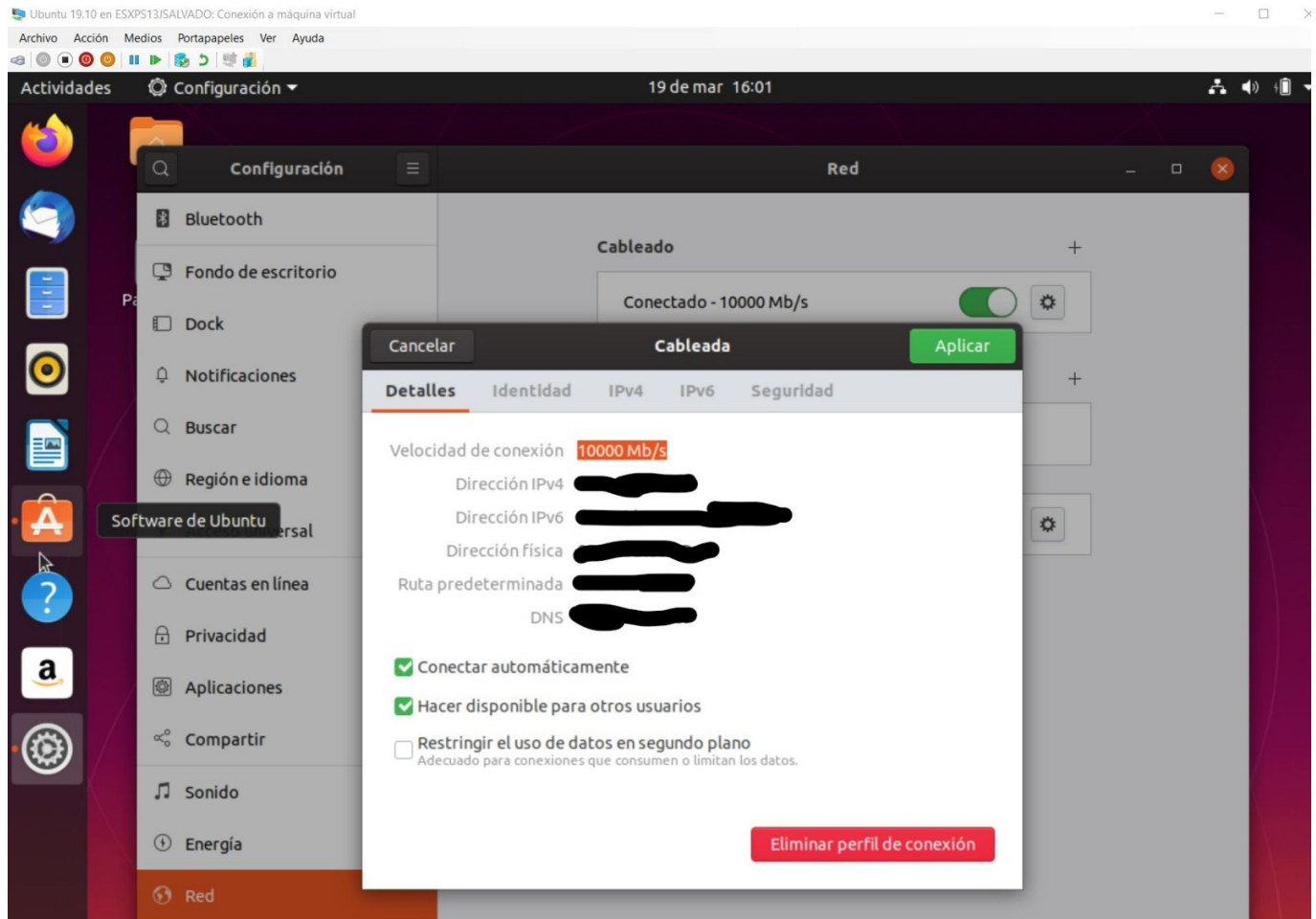
If you don't have it enabled yet, go to add Windows Features and enable Hyper-V. You'll need to restart (text images is in Spanish, but that's my current locale. I hope that altogether with instructions will help to "decrypt" it if you don't know the Don Quixote's lingua )



## 2. Create an Ubuntu virtual machine on Hyper-V

I don't think there is any easier way of creating a virtual machine (VM). Just open the window of the Hyper-V Manager, and go to option [Quick Create...](#) (right up in your screen) and create your virtual machine using any of Ubuntu versions already offered (you could download an iso file of whatever other Linux and so create the VM with a different distro). In my case, I've chosen the last Ubuntu release available: 19.10. Anyway, everything you'll see here is valid for 18.04. In 15 or 20 minutes, depends on what the image takes to download for you, you'll have your new VM created and ready.

**Important:** Leave the option of Default Switch as it's offered . This will guarantee you have access to internet from both, the host and the virtual machine.

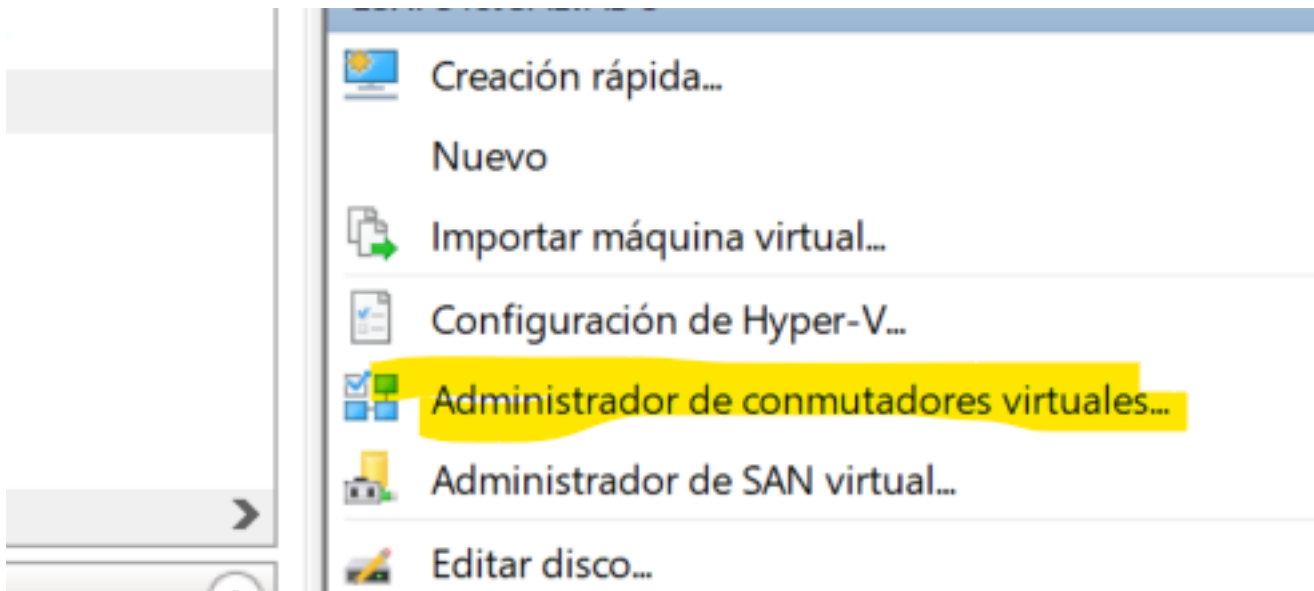


### 3. Create a local subnet

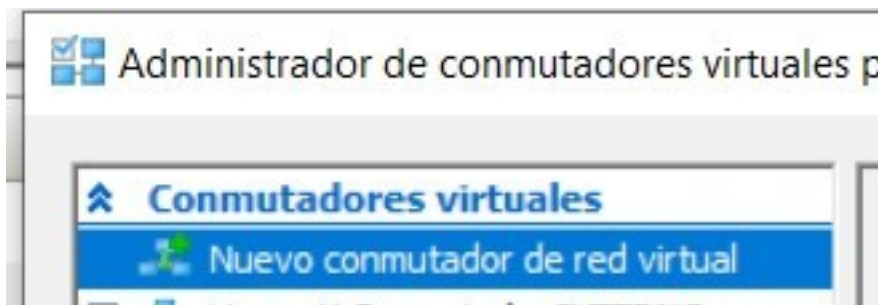
One of the problems of using virtual machines that I've found very often has to do with network configuration... sometimes works, others don't, or it works if I'm connected with Wi-fi but not by cable or the opposite, or if I establish a VPN in the windows host, then I lose internet access in the VM, or communication between the VM (Linux) and the host (Windows) gets broken... in short... it's crazy! Makes me not to trust in my environment when I use my laptop for development, small and quick demos or for presentations, where most likely access to internet is not as important as being sure that communications between my host and my VM(s) work in a reliable way.

With an ad-hoc local subnet, shared between your Windows host and your virtual machines, you solve it. To let them communicate among each other, you use that subnet and that's it. You just have to assign specific IPs to your host and your VMs and all set.

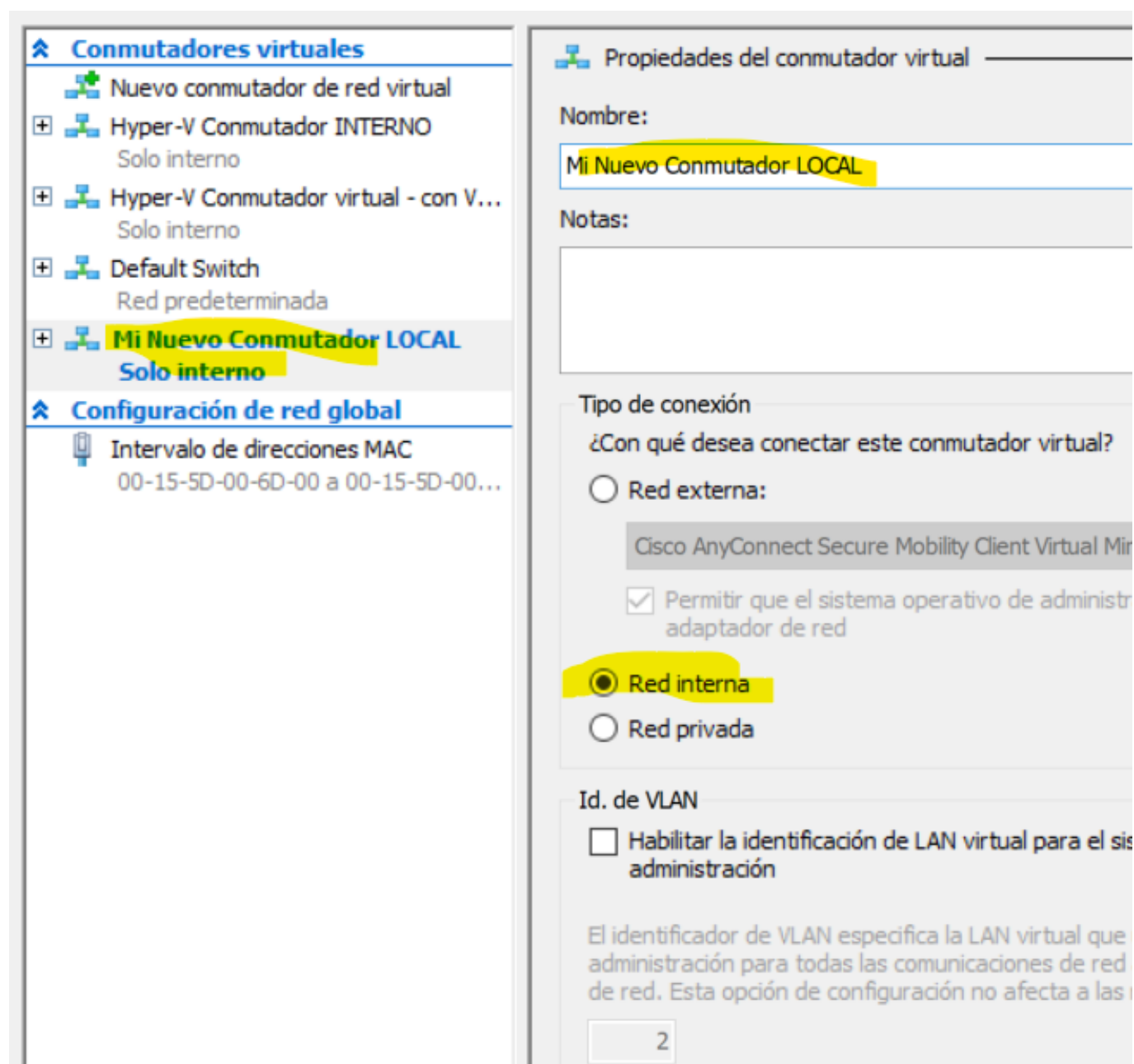
It's really easy to do it with these steps. Just go to [Virtual Switch Manager...](#) that you'll find in your Hyper-V Manager:



Once there, go to option [New Virtual Switch](#) (it would be like a new network card for the VM afterwards):



Be sure you define it as [Internal Network](#) , choose the name we want and leave the other options by default



Now, if we go to Windows Control Panel --> Network and Sharing Center, we'll see that we already have there the switch we just created:


Panel de control > Todos los elementos de Panel de control > Centro de redes y recursos compartidos

## Ver información básica de la red y configurar conexiones

Ver las redes activas

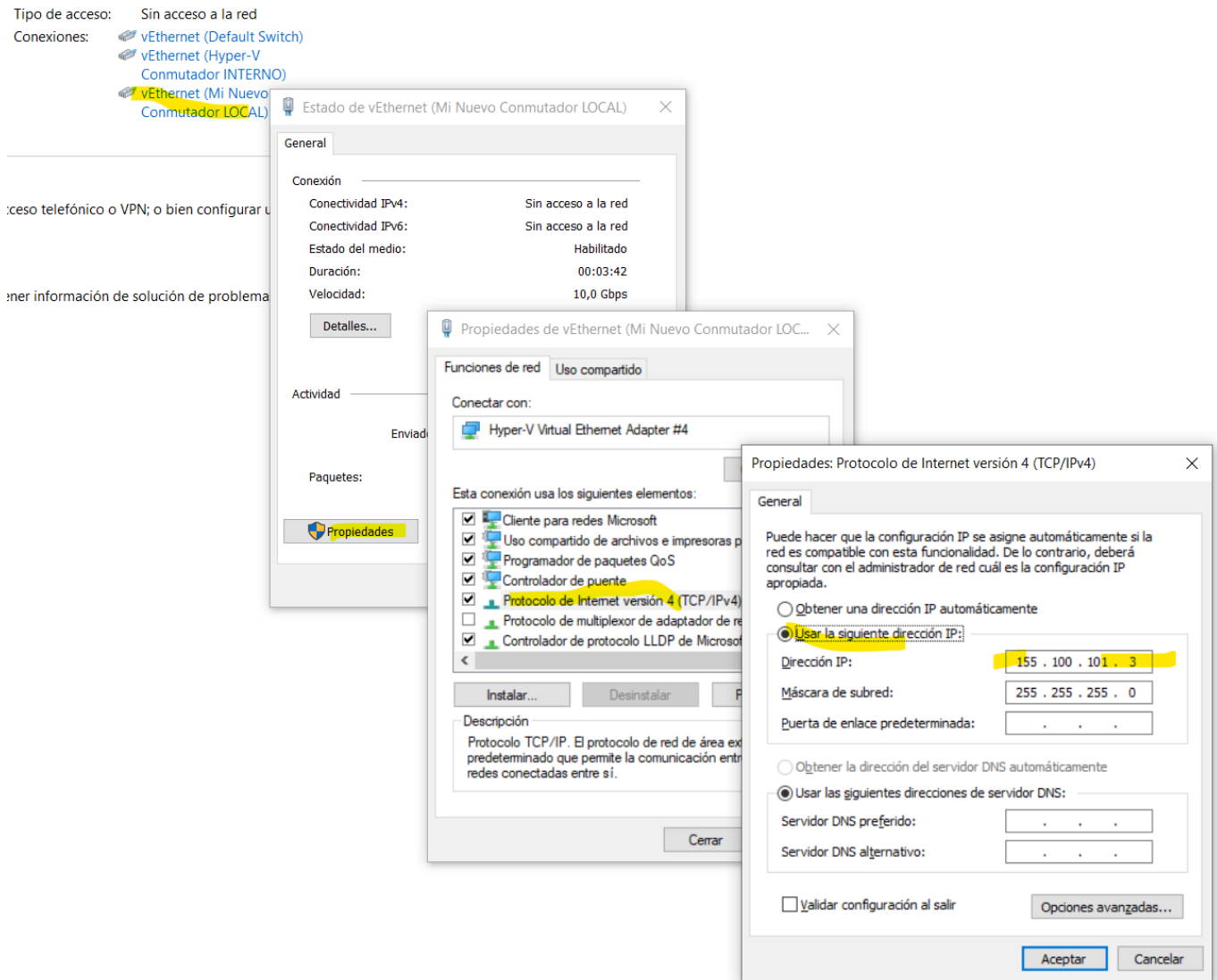
duración del	<b>iscinternal.com</b> Red de dominios	Tipo de acceso: Internet Conexiones: Cisco AnyConnect Secure Mobility Client Connection
duración de uso lanzando	<b>INSANITY</b> Red privada	Tipo de acceso: Sin acceso a Internet Conexiones: Wi-Fi [REDACTED]
streaming	<b>Red no identificada</b> Red pública	Tipo de acceso: Sin acceso a la red Conexiones: vEthernet (Default Switch) vEthernet (Hyper-V Conmutador INTERNO)
	<b>Identificando...</b>	Tipo de acceso: Sin acceso a la red Conexiones: vEthernet (Mi Nuevo Conmutador LOCAL)

Cambiar la configuración de red

 [Configurar una nueva conexión o red](#)  
Configurar una conexión de banda ancha, de acceso telefónico o VPN; o bien configurar un

#### 4. Configure the local Subnet shared by the host and the Virtual Machines

At this point you can finish the configuration of your new local network. To do that, set your cursor over the connection **Mi Nuevo Conmutador LOCAL**, click and go to **Properties**, and from there to **IPv4 protocol** so to assign a fixed IP address:

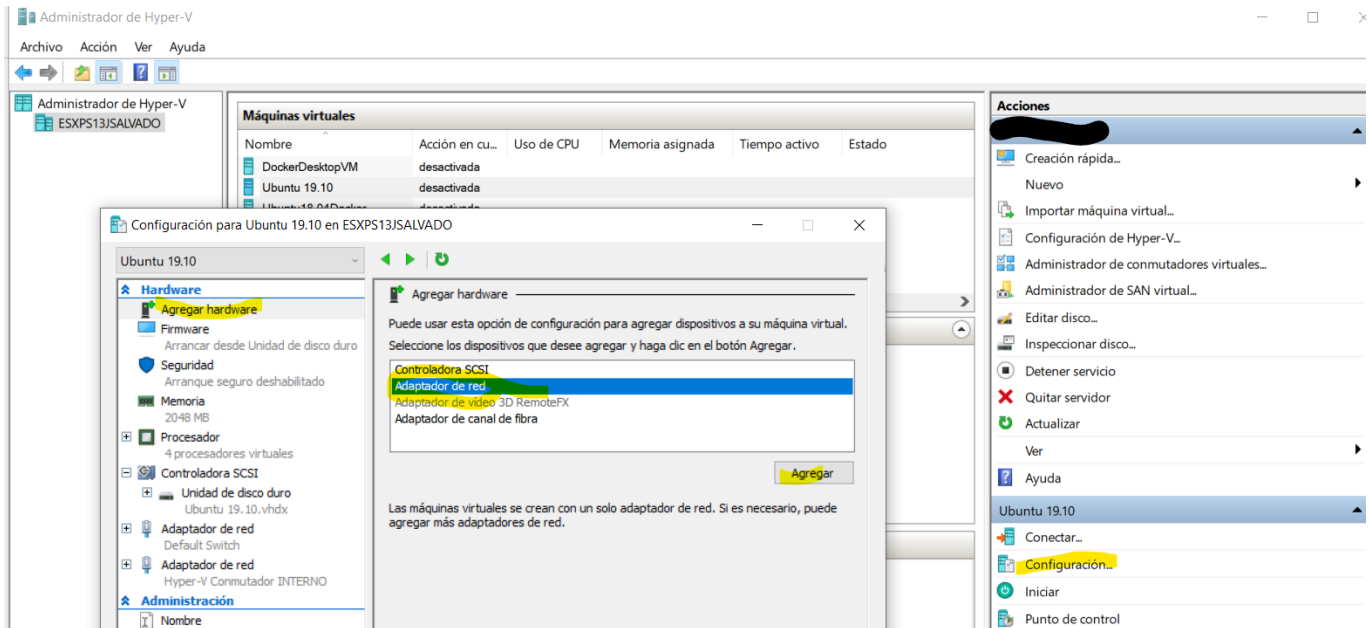


**Important:** The IP that you assign here will be your host (Windows) IP in this local subnet.

##### 5. Link and configure your new local network to your virtual machine

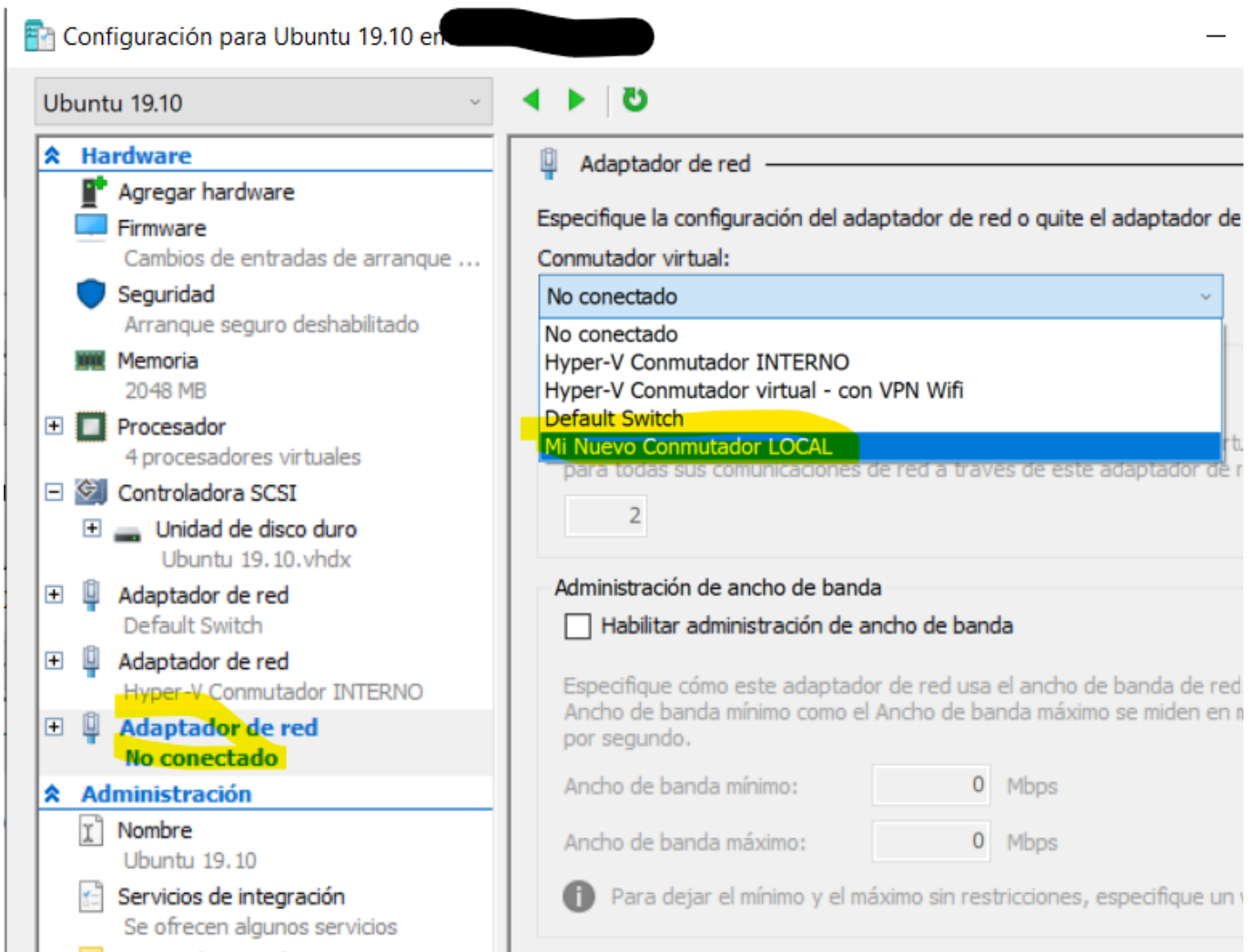
Now come back to your Hyper-V Manager. If your VM is running, stop it. Once stopped, go to its configuration and add the new internal virtul switch:





(Note.- In the image you can see another switch, the Hyper-V Conmutador INTERNO . It's for another subnet that I have. But it's not necessary for you in this configuration)

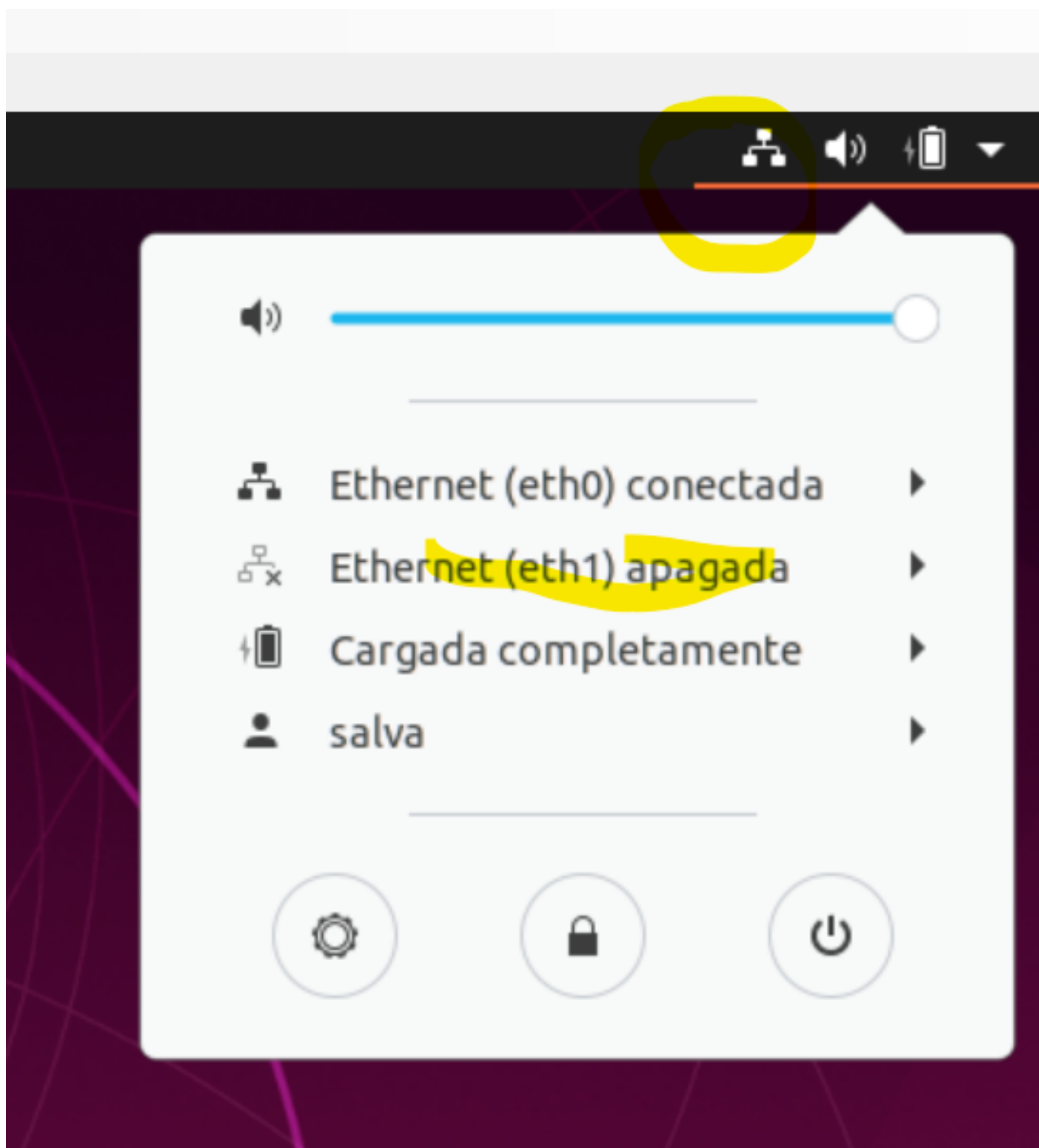
Once you click on Add, you will just have to select the switch that you previously created:



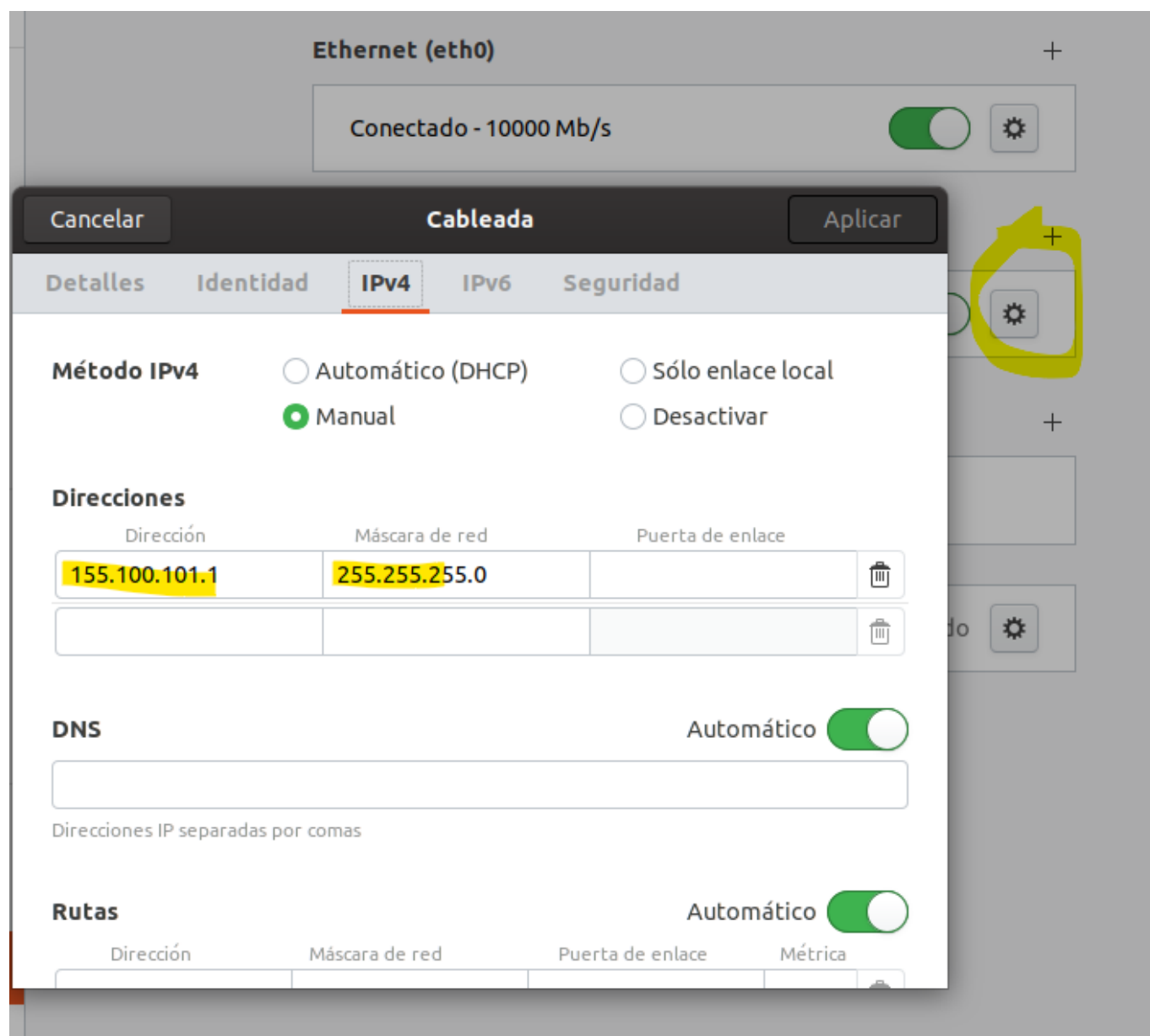
Well, once this is done, click Apply, Accept, ... and you're ready! You just can start and login again in your virtual



machine to finish the configuration of the internal connection. To do that, once the VM starts, clicnk on the network icon (right up) and you'll see that you have 2 networks: eth0 y eth1. The eth1 appears as disconnected... by now:



Go into the configuration of the Ethernet (eht1) and assign a fixed IP for this local subnet, for example: 155.100.101.1, and the subnet mask: 255.255.255.0



and that's all. Here you have your virtual machine, identified with IP 155.100.101.1 sharing the same subnet with your host.

#### 7. Allow access to Windows 10 from your virtual machine


You will likely discover that Windows 10 doesn't allow by default connection from other server and, for your windows system, the VM you've just created is precisely that, an external and potentially dangerous server... so you'll have to add a rule in the Firewall to be able to connect to your host from those virtual machines. How? Very easy, just look for Windows Defender Firewall in your Windows Control Panel, go to [Advance Configuration](#) and create a new Entry Rule:

The screenshot shows the 'Asistente para nueva regla de entrada' (New Incoming Rule Wizard) in Windows Firewall. The 'Acciones' (Actions) pane on the right shows 'Reglas de entrada' (Incoming Rules) selected, with 'Nueva regla...' (New Rule...) highlighted. The main window is titled 'Tipo de regla' (Rule Type) and asks '¿Qué tipo de regla desea crear?' (What type of rule do you want to create?). The 'Pasos:' (Steps) pane on the left shows 'Tipo de regla' as the current step. The main area has three radio button options: 'Programa' (Program), 'Puerto' (Port), and 'Predefinida:' (Predefined:). The 'Puerto' option is selected and highlighted with a yellow circle. Below the 'Predefinida:' option, there is a text box containing '@FirewallAPI.dll,-80200' and a description: 'Regla que controla las conexiones de una experiencia con Windows.'

You can set one port or one or many range of them... (also you can set the rule for all the ports)...

The screenshot shows the 'Asistente para nueva regla de entrada' (New Incoming Rule Wizard) in Windows Firewall, Step 2: 'Protocolo y puertos' (Protocol and ports). The 'Pasos:' (Steps) pane on the left shows 'Protocolo y puertos' as the current step. The main area asks '¿Se aplica esta regla a TCP o UDP?' (Does this rule apply to TCP or UDP?). There are two radio button options: 'TCP' and 'UDP'. The 'TCP' option is selected and highlighted with a yellow circle. Below this, it asks '¿Se aplica esta regla a todos los puertos locales o a unos puertos locales específicos?' (Does this rule apply to all local ports or to specific local ports?). There are two radio button options: 'Todos los puertos locales' (All local ports) and 'Puertos locales específicos:' (Specific local ports:). The 'Puertos locales específicos:' option is selected and highlighted with a yellow circle. Below it, there is a text box containing '51771-51779,52771-52779' and an example: 'Ejemplo: 80, 443, 5000-5010'.

The action we want is Allow Connection...

 Asistente para nueva regla de entrada

### Acción

Especifique la acción que debe llevarse a cabo cuando una conexión coincide con las condiciones especificadas en

**Pasos:**


- Tipo de regla
- Protocolo y puertos
- Acción**
- Perfil
- Nombre

¿Qué medida debe tomarse si una conexión coincide con las condiciones?

☒ **Permitir la conexión**  
Esto incluye las conexiones protegidas mediante IPsec y las que no lo son.

☐ Permitir la conexión si es segura

For all network types...

 Asistente para nueva regla de entrada

### Perfil

Especifique los perfiles en los que se va a aplicar esta regla.

**Pasos:**

- Tipo de regla
- Protocolo y puertos
- Acción
- Perfil**
- Nombre

¿Cuándo se aplica esta regla?

☒ **Dominio**  
Se aplica cuando un equipo está conectado a su dominio corporativo.

☒ **Privado**  
Se aplica cuando un equipo está conectado a una ubicación de red privada, como una red doméstica o del lugar de trabajo.

☒ **Público**  
Se aplica cuando un equipo está conectado a una ubicación de redes públicas.

Give a name to your rule...

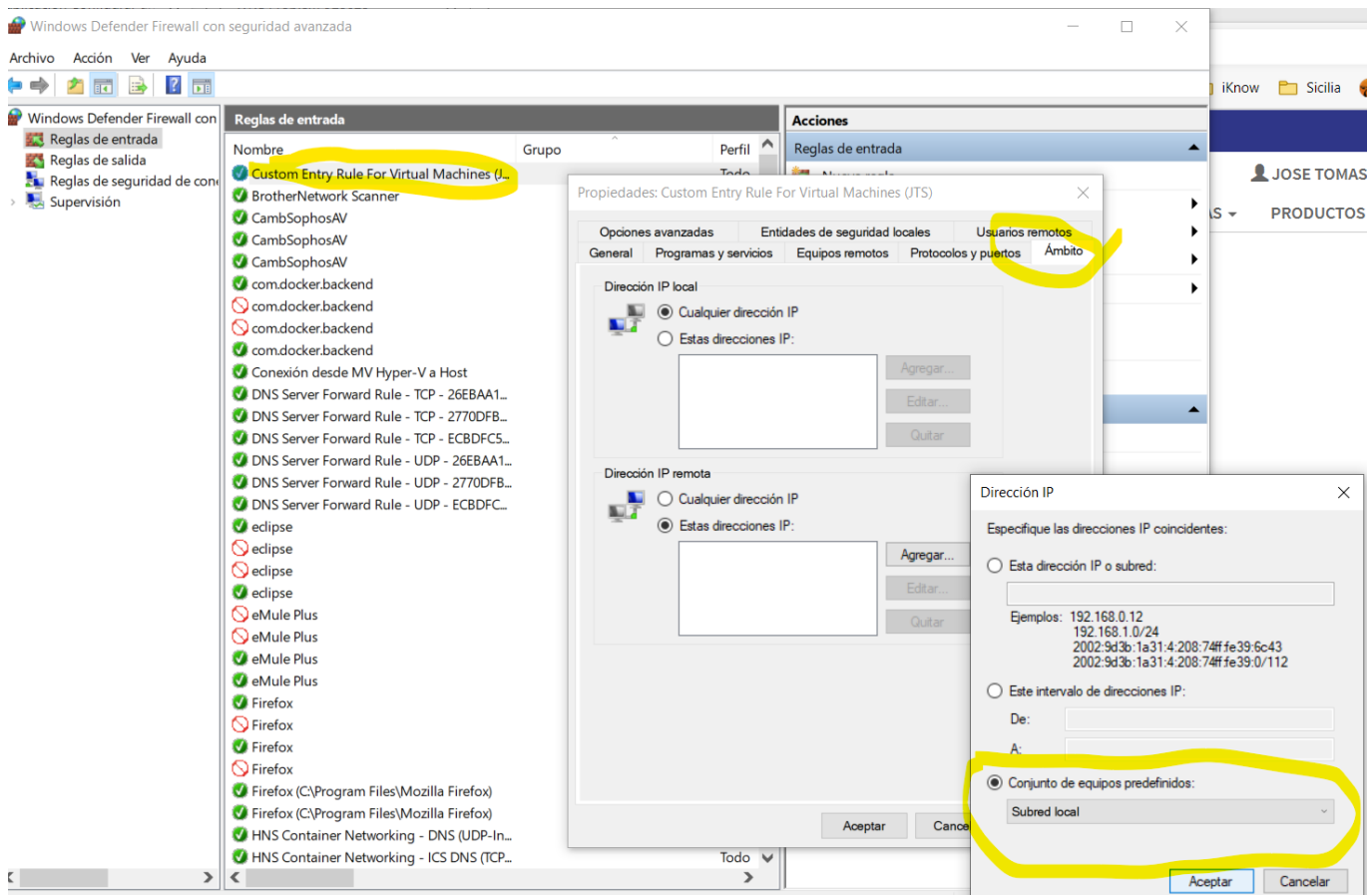
- Acción
- Perfil
- Nombre**

Nombre:

Descripción (opcional):

This is a rule to give open access to my Windows 10 host from Hyper-V Virtual machines located on my own local subnet.

And important, immediately after that, open again the properties of your newly created rule and limit the application scope, so to apply only for connections within your local Subnet...



## 8. READY. Install Docker and whatever other application in your new Ubuntu Virtual Machine

Once you go through the entire installation process and you have your new VM ready and up-to-date, with internet access, etc.. you can install the apps you want... Docker as a minimum, that was the idea from the beginning, also you can install your VPN client if you need it for connection with your company network, VS Code, Eclipse+Atelier,...

Specifically, to install Docker, within your VM, you can follow the instructions you'll find here: <https://docs.docker.com/install/linux/docker-ce/ubuntu/>

Be sure that your Docker runtime is working, download some test image, etc... and that's it.

With this... ¡You're all set!, now you'll be able to have containers running with no limitations (apart from your hardware capacity) in your Ubuntu VM, to which you'll be able to connect from your Windows 10 host, from a browser or an app and the other way around, from your Ubuntu VM to your Windows 10 host. All of that using your IP addresses set in your shared local subnet, that will work no matter if you have a VPN established or not, if you access to internet through your Wi-fi adapter or through your ethernet cable.

Ah... one last advice. If you want to exchange files between Windows 10 and your virtual machines, a very useful and simple option is using [WinSCP](#). It's free and it works really well.

Well, surely there are other configurations... but this is the one that I use and has proved to be the more reliable. I hope you find it useful also. If I've avoid you any headache, this article will have been worthy.

¡Happy Coding!

[#Best Practices](#) [#Development Environment](#) [#Docker](#) [#Microsoft Windows](#) [#System Administration](#) [#Tips & Tricks](#) [#Ubuntu](#) [#InterSystems IRIS](#)

URL:<https://community.intersystems.com/post/configure-environment-use-docker-hyper-v-ubuntu-virtual-machines-running-windows-10-host>