

# Configuración de la VM de Windows al módulo CGM-SRV en CGR1xxx

## Contenido

[Introducción](#)

[Prerequisites](#)

[Requirements](#)

[Componentes Utilizados](#)

[Antecedentes](#)

[Configurar](#)

[Crear una imagen de VM de Windows](#)

[Instale KVM en su equipo Linux](#)

[Verificar instalación KVM](#)

[Crear una VM de Windows](#)

[Implementar imagen de VM de Windows en CGM-SRV](#)

[Verificación](#)

[Troubleshoot](#)

## Introducción

Este documento describe los pasos necesarios para crear y ejecutar una máquina virtual de Windows (VM) en el módulo Connected Grid Module (CGM) - System Server (SRV).

## Prerequisites

### Requirements

Cisco recomienda que tenga conocimiento sobre estos temas:

- Linux
- Máquina virtual basada en núcleo (KVM)
- Comprender los conceptos de virtualización

### Componentes Utilizados

La información que contiene este documento se basa en las siguientes versiones de software y hardware.

- Routers de red conectada (CGR) 1120
- módulo CGM-SRV-XX
- Los pasos de configuración para CGM-SRV se ejecutan antes de esta guía:
- ISO de instalación de Windows 7
- Visor de Virtual Network Computing (VNC)

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. Si tiene una red en vivo, asegúrese de entender el posible impacto de cualquier comando.

## Antecedentes

Cuando desee ejecutar aplicaciones IOx o VM en la plataforma CGR1000, puede utilizar el módulo de cálculo CGM-SRV. El módulo CGM-SRV es en realidad un servidor pequeño que contiene una CPU x86 de varios núcleos, memoria y almacenamiento. Tanto el CGR1120 como el CGR1240 pueden tener uno de estos módulos para agregar capacidades IOx.

En el momento de redactar el presente informe, hay dos tipos disponibles:

Unidad de mantenimiento de existencias (SKU)	Unidades de estado sólido (SSD)	RAM	CPU
CGM-SRV-64	64 GB (50 GB utilizables)	4 GB	4 núcleos 800 MHz
CGM-SRV-128	128 GB (100 GB utilizables)	4 GB	4 núcleos 800 MHz

Cada módulo también tiene dos puertos USB para el almacenamiento y su propia interfaz Gigabit Ethernet externa.

Al igual que con cualquier otro dispositivo compatible con IOx, el módulo puede alojar diferentes tipos de aplicaciones IOx, pero debido a la mayor capacidad del módulo CGM-SRV, también puede ejecutar un dispositivo Windows completamente configurado o un dispositivo Linux estándar (por ejemplo, Ubuntu o CentOS).

## Configurar

### Crear una imagen de VM de Windows

Para implementar una VM de Windows en el módulo CGM-SRV, primero debe crear una imagen en el formato QEMU QCOW que contiene la instalación de Windows. Una forma de crear una imagen de este tipo es con KVM y virsh en una máquina Linux.

Los pasos que se mencionan más adelante no incluyen el CGR1xxx o CGM-SRV en absoluto, sino que son simplemente los pasos necesarios para crear una imagen básica de VM QCOW de Windows 7 que puede implementar en el siguiente paso del CGM-SRV.

Para esta guía, puede comenzar con una instalación mínima de CentOS7 recién instalada. Los pasos para otras distribuciones de Linux deben ser similares pero pueden ser ligeramente diferentes.

### Instale KVM en su equipo Linux

Paso 1. Lo primero que hay que hacer es comprobar si la máquina host admite extensiones VM. En la plataforma x86, son AMD-V o VT-X de Intel. La mayoría, si no todas, las CPU x86 modernas admiten estas extensiones. Incluso cuando se ejecuta una VM, la mayoría de los hipervisores proporcionan la opción de pasar/emular estas extensiones.

Para verificar si la CPU instalada soporta esas extensiones, debe verificar si el indicador vmx (para VT-X) o svm (para AMD-V) existe en la salida `cpuinfo`.

```
[root@cen7 ~]# egrep -c '(vmx|svm)' /proc/cpuinfo
```

```
2
```

Si el resultado de este comando es 0, esto significa que ninguna CPU encontrada soporta las extensiones VM. En ese caso, puede verificar si estas extensiones están habilitadas en su BIOS o hipervisor cuando utilice una VM para ejecutar esta máquina.

Paso 2. El siguiente paso es crear un puente para proporcionar una red para la máquina virtual que se puede ejecutar en KVM.

En primer lugar, debe habilitar el reenvío de IP en el núcleo:

```
[root@cen7 ~]# echo "net.ipv4.ip_forward = 1"|sudo tee /etc/sysctl.d/99-ipforward.conf
```

```
net.ipv4.ip_forward = 1
```

```
[root@cen7 ~]# sysctl -p /etc/sysctl.d/99-ipforward.conf
```

```
net.ipv4.ip_forward = 1
```

Para crear el puente, la configuración IP necesita moverse de la interfaz real al propio puente, ya que ésta es la interfaz que posee la dirección IP.

Después de completar una instalación estándar, la configuración de red está en **/etc/sysconfig/network-scripts:**

```
[root@cen7 ~]# ls -l /etc/sysconfig/network-scripts/ifcfg-*
```

```
/etc/sysconfig/network-scripts/ifcfg-eno16777736
```

```
/etc/sysconfig/network-scripts/ifcfg-lo
```

Paso 3. Como puede ver, actualmente hay una interfaz (además de la interfaz de loopback), llamada eno16777736. Debe mover la configuración relacionada con IP a una interfaz de bridge a la que puede llamar virbr0:

```
[root@cen7 ~]# vi /etc/sysconfig/network-scripts/ifcfg-virbr0
```

```
[root@cen7 ~]# cat /etc/sysconfig/network-scripts/ifcfg-virbr0
```

```
DEVICE=virbr0
```

```
TYPE=BRIDGE
```

```
ONBOOT=yes
```

```
BOOTPROTO=static
```

```
IPADDR=172.16.245.162
```

```
NETMASK=255.255.255.0
```

```
GATEWAY=172.16.245.2
```

```
DNS1=8.8.8.8
```

Paso 4. Después de eso, debe limpiar la configuración IP de la interfaz real y conectarla al puente virbr0:

```
[root@cen7 ~]# vi /etc/sysconfig/network-scripts/ifcfg-eno16777736
```

```
[root@cen7 ~]# cat /etc/sysconfig/network-scripts/ifcfg-eno16777736
```

```
UUID=46f0f247-e164-40cc-866b-9133458d9df8
```

```
DEVICE=eno16777736
```

```
ONBOOT=yes
```

```
BRIDGE=virbr0
```

```
HWADDR=00:0c:29:ce:96:38
```

Paso 5. Una vez finalizada la configuración de red, puede continuar e instalar KVM:

```
[root@cen7 ~]# sudo yum install kvm virt-manager libvirt virt-install qemu-kvm xauth dejavu-lgc-
```

```
sans-fonts -y
```

```
...
```

```
Complete!
```

Paso 6. Una vez finalizada la instalación, lo mejor es reiniciar esta máquina para aplicar los módulos recién instalados y la configuración de red:

```
[root@cen7 ~]# init 6
```

## Verificar instalación KVM

Paso 7. Una vez que se haya completado el reinicio, debería poder acceder a la máquina en la (misma) IP configurada en la interfaz de bridge. Debe verificar si el módulo kernel KVM está cargado:

```
root@cen7 ~]# lsmod|grep kvm
kvm_intel          200704  0
kvm                589824  1 kvm_intel
irqbypass         16384   1 kvm
```

Paso 8. Si esto parece correcto, puede intentar conectarse con virsh:

```
[root@cen7 ~]# sudo virsh -c qemu:///system list
 Id      Name                               State
-----
```

Paso 9. Un último paso es abrir el puerto 5900 en el firewall de esta máquina para el acceso VNC a la instalación de Windows:

```
[root@cen7 ~]# firewall-cmd --zone=public --add-port=5900/tcp --permanent
success
[root@cen7 ~]# firewall-cmd --reload
success
```

## Crear una VM de Windows

Ahora que tiene un sistema que funciona con la instalación KVM, puede encender una nueva VM en KVM y ejecutar a través de los diálogos de instalación de Windows.

Paso 1. Copie la ISO de instalación de Windows 7 en su VM (o haga que sea accesible a través de la red):

```
[root@cen7 ~]# scp jedepuyd@172.16.X.X:/home/jedepuyd/win7install.iso /var
jedepuyd@172.16.X.X's password:
win7install.iso                               100% 4546MB  62.1MB/s
01:13
```

Paso 2. Cree una nueva VM KVM y déjela arrancar desde la ISO de Windows 7:

```
root@cen7 ~]# virt-install --connect qemu:///system -n win7 -r 1024 --vcpus=2 --disk
path=/var/lib/libvirt/images/win7.img,size=9 --graphics vnc,listen=0.0.0.0 --noautoconsole --os-
type windows --os-variant win7 --accelerate --network=bridge:virbr0 --hvm --cdrom
/var/win7install.iso
```

```
Starting install...
Allocating win7.img
```

| 9.0 GB

00:00:00

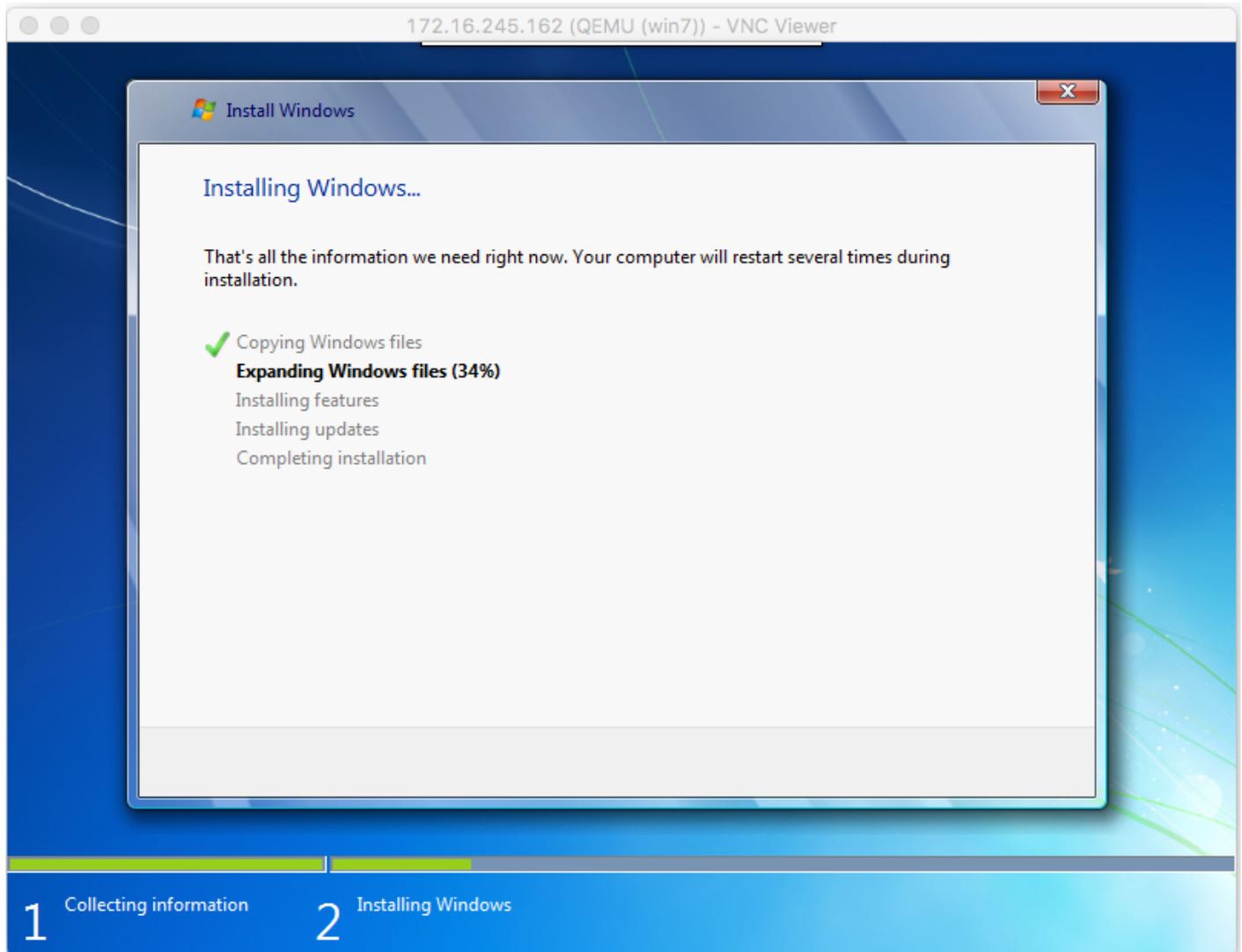
Creating domain...

| 0 B

00:00:00

Domain installation still in progress. You can reconnect to the console to complete the installation process.

Paso 3. Una vez que la máquina virtual se ha iniciado, puede conectarse con el uso del visor VNC a la IP de la máquina host en el puerto 5900 y finalizar la instalación estándar de Windows como se muestra en la imagen:



Si Windows se reinicia en el momento de la instalación, podría ser necesario reiniciar la máquina virtual con virsh si esto no se realiza automáticamente:

```
[root@cen7 ~]# virsh start win7  
Domain win7 started
```

Paso 4. Una vez finalizada la instalación, apague la máquina virtual. Ahora tiene una imagen QCOW de esta instalación en la ruta proporcionada cuando crea la VM:

`/var/lib/libvirt/images/win7.img`. Este tipo de imagen se puede implementar en el CGM-SRV para ejecutar Windows.

## Implementar imagen de VM de Windows en CGM-SRV

Ahora que tiene el tipo correcto de imagen para ejecutar en el CGM-SRV, puede comenzar a implementarlo.

Paso 1. Configure un perfil para ioxclient que se corresponda con su configuración:

```
[root@cen7 ~]# ./ioxclient profiles create
Enter a name for this profile : CGR1120_20
Your IOx platform's IP address[127.0.0.1] : 10.X.X.X.X
Your IOx platform's port number[8443] :
Authorized user name[root] : admin
Password for admin :
Local repository path on IOx platform[/software/downloads]:
URL Scheme (http/https) [https]:
API Prefix[/iox/api/v2/hosting/]:
Your IOx platform's SSH Port[2222]:
Your RSA key, for signing packages, in PEM format[]:
Your x.509 certificate in PEM format[]:
Activating Profile CGR1120_20
Saving current configuration
```

En este ejemplo, 10.X.X.X corresponde a la interfaz saliente en el CGR1000 en el que se configuró la traducción de direcciones de red (NAT) para reenviar al puerto 8443 en el CGM-SRV.

Paso 2. Ahora que el ioxclient está configurado, cambiemos el nombre de su imagen creada anteriormente a **vm.img** para simplificar un bit y copiarla con el uso de Secure Copy (SCP) con ioxclient a CGM-SRV.

Opcionalmente, convierta la imagen de disco al formato QCOW2 como es lo que el CGM-SRV espera. Las versiones más recientes de virt-manager parecen crear las imágenes de disco de forma predeterminada en el formato QCOW3.

Puede convertir fácilmente la imagen con este comando:

```
[root@cen7 ~]# qemu-img convert -f qcow2 -O qcow2 /var/lib/libvirt/images/win7.img
/var/lib/libvirt/images/win7.img
```

Una vez que esté seguro de que la imagen está en el formato correcto, continúe con el cambio de nombre y copie:

```
[root@cen7 ~]# mv /var/lib/libvirt/images/win7.img /root/vm.img
[root@cen7 ~]# ./ioxclient platform scp /root/vm.img
Currently active profile : CGR1120_20
Command Name: plt-scp
Saving current configuration
Downloaded scp keys to pscp.pem
Running command : [scp -P 2222 -r -i pscp.pem /root/vm.img scpuser@10.50.215.246:/]
```

Esta transferencia podría tardar un tiempo, las velocidades de transferencia de aproximadamente 3-4 MB/s a CGM-SRV a través de Cisco IOS®. El archivo se copia en **/mnt/data/vm/vm.img** en el módulo CGM-SRV.

Paso 3. Mientras la transferencia está en curso (o completa), puede crear el archivo **package.yaml**. Este archivo describe a IOx qué desea implementar exactamente y cómo empaquetarlo.

```
[root@cen7 ~]# vi package.yaml
[root@cen7 ~]# cat package.yaml
descriptor-schema-version: 2.2
```

info:

```
author-link: http://www.cisco.com/ author-name: Jens Depuydt description: Windows 7 VM for
CSR-SRV name: win7 version: 1.0 app: type: vm cpuarch: x86_64 resources: profile: custom cpu:
600 disk: 10 memory: 3072 network: - interface-name: eth0 - interface-name: eth1 graphics: vnc:
true startup: ostype: windows qemu-guest-agent: false disks: - target-dev: hda file:
file://vm.img
```

Como puede ver en este **package.yaml**, se refiere a **file://vm.img** que corresponde con la ubicación real de **mnt/data/vm/vm.img** en el módulo CGM-SRV.

Paso 4. El siguiente paso es empaquetar con el uso de **ioxclient**:

```
[root@cen7 ~]# ./ioxclient pkg .
Currently active profile : default
Command Name: package
No rsa key and/or certificate files to sign the package
Checking if package descriptor file is present..
Validating descriptor file /root/package.yaml with package schema definitions
Parsing descriptor file..
Found schema version 2.2
Loading schema file for version 2.2
Validating package descriptor file..
File /root/package.yaml is valid under schema version 2.2
Created Staging directory at : /var/folders/sp/f9qn2fsn0d5fkj7szps6qvvr0000gn/T/638513626
Copying contents to staging directory
Checking for application runtime type
Couldn't detect application runtime type
Creating an inner envelope for application artifacts
Excluding .DS_Store
Generated /var/folders/sp/f9qn2fsn0d5fkj7szps6qvvr0000gn/T/638513626/artifacts.tar.gz
Calculating SHA1 checksum for package contents..
Package MetaData file was not found at
/private/var/folders/sp/f9qn2fsn0d5fkj7szps6qvvr0000gn/T/638513626/.package.metadata
Wrote package metadata file :
/private/var/folders/sp/f9qn2fsn0d5fkj7szps6qvvr0000gn/T/638513626/.package.metadata
Root Directory : /private/var/folders/sp/f9qn2fsn0d5fkj7szps6qvvr0000gn/T/638513626
Output file: /var/folders/sp/f9qn2fsn0d5fkj7szps6qvvr0000gn/T/559089521
Path: .package.metadata
SHA1 : 262f763740c182f95358be84514a76ac11e37012
Path: artifacts.tar.gz
SHA1 : 3d89ccd35fe5318dd83a249a26cb8140d98d15bb
Path: package.yaml
SHA1 : aa42f949b707df07a83a17344e488c44eb585561
Generated package manifest at package.mf
Generating IOx Package..
Package generated at /root/package.tar
```

Paso 5. Después de crear el paquete, puede instalarlo en nuestro CGM-SRV. La aplicación/VM IOx se denomina **win7** en este ejemplo:

```
[root@cen7 ~]# ./ioxclient app install win7 package.tar
Currently active profile : default
Command Name: application-install
Saving current configuration
```

Installation Successful. App is available at :  
<https://10.X.X.X:8443/iox/api/v2/hosting/apps/win7> Successfully deployed

Paso 6. Antes de poder activar la VM Win7 IOx, debe crear un archivo JSON de carga útil que establezca la contraseña VNC para esta VM:

```
[root@cen7 ~]# vi vnc.json
[root@cen7 ~]# cat vnc.json
{
  "resources": {
    "graphics": {"vnc-password": "password"}
  }
}
```

Paso 7. Con el uso de la carga **vnc.json**, puede activar la VM Win7 IOx:

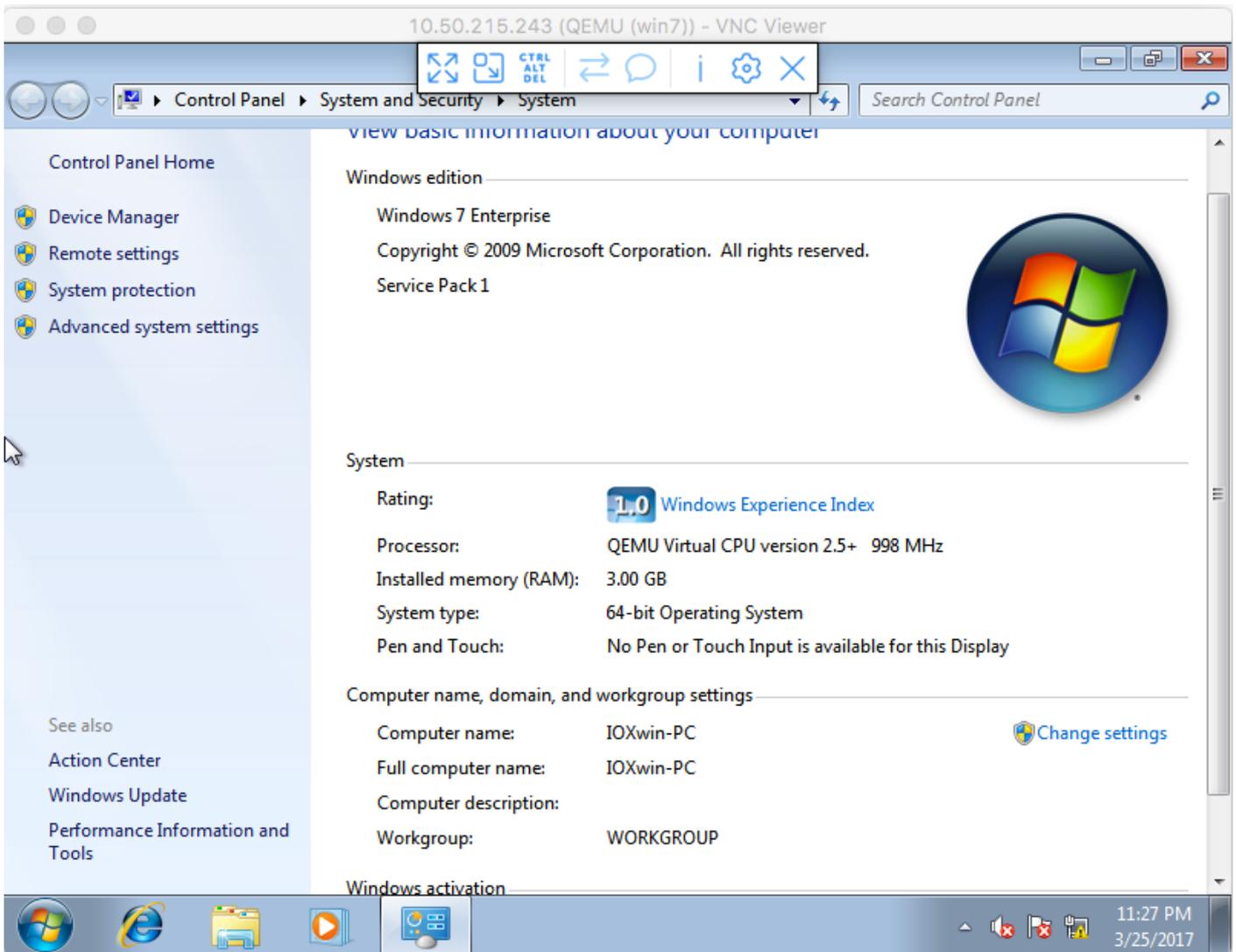
```
[root@cen7 ~]# ./ioxclient app activate win7 --payload vnc.json
Currently active profile : default
Command Name: application-activate
Payload file : vnc.json. Will pass it as application/json in request body..
App win7 is Activated
```

Paso 8. El último paso con ioxclient es iniciar la VM:

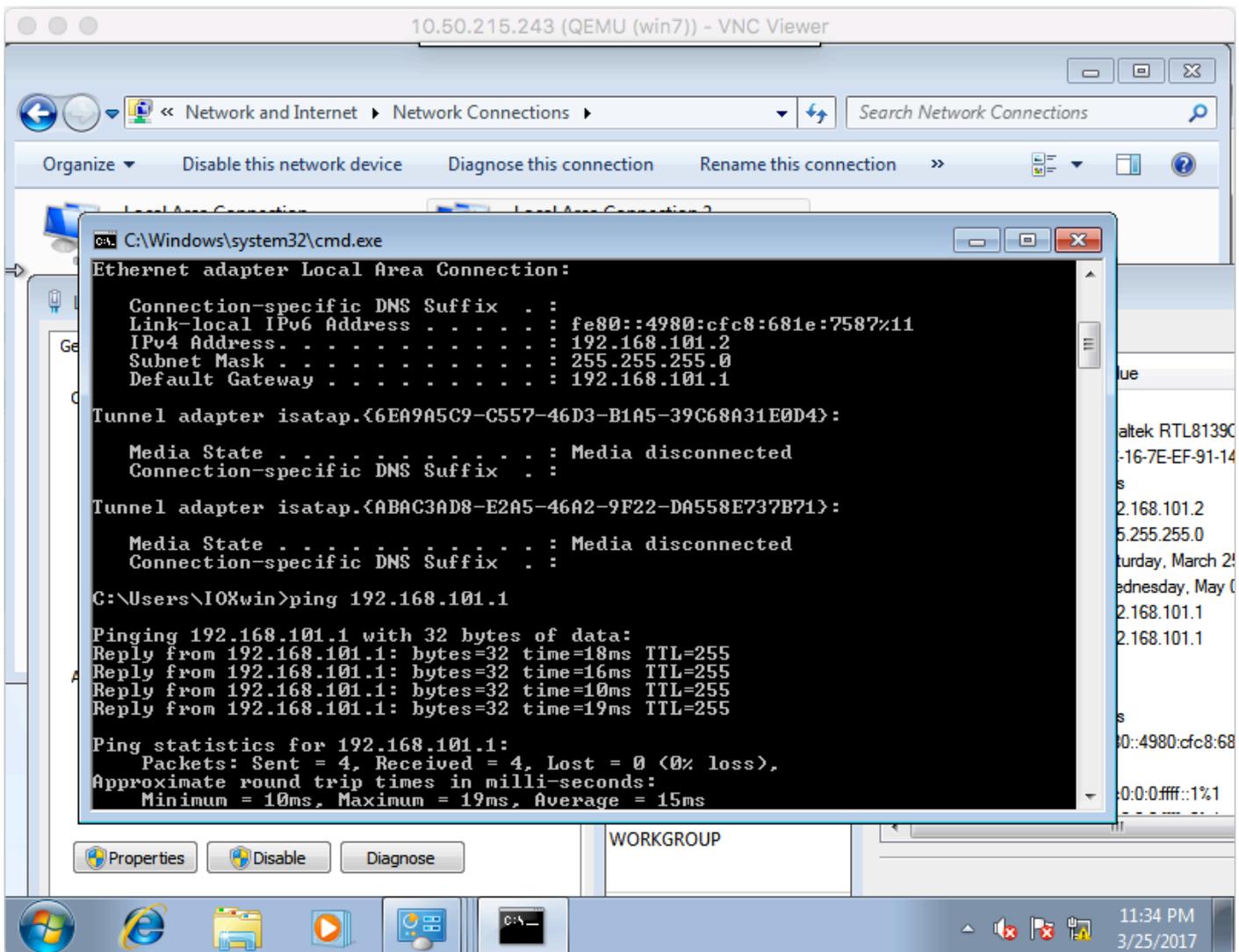
```
[root@cen7 ~]# ./ioxclient app start win7
Currently active profile : default
Command Name: application-start
App win7 is Started
```

En este momento, la VM de Windows se ejecuta en el CGM-SRV y puede comenzar a utilizarlo.

Para obtener acceso a la consola de la máquina Windows, puede utilizar el visor VNC en la interfaz saliente en el CGR1000 y el puerto 5900, como se muestra en la imagen:



Desde una perspectiva de red, usted eligió dar eth0 y eth1 a la VM Win7 IOx con el uso del archivo `package.yaml` como se muestra en la imagen:



Como puede ver, estas interfaces obtuvieron una IP del servidor DHCP que se ejecuta en Cisco IOS® y se puede utilizar sin más configuración.

## Verificación

Utilice esta sección para confirmar que su configuración funcione correctamente.

Para verificar si la VM se ejecuta:

```
[root@cen7 ~]# ./ioxclient app list
Currently active profile : CGR1120_20
Command Name: application-list
Saving current configuration
List of installed App :
 1. win7      --->  RUNNING
```

También puede comprobar el estado desde el Administrador local, como se muestra en la imagen:

#### Application information

ID:	win7
State:	RUNNING
Name:	win7
Cartidge Required:	* None
Version:	1.0
Author:	Jens Depuydt
Author link:	<a href="http://www.cisco.com/">http://www.cisco.com/</a>
Application type:	vm
Description:	Windows 7 VM for CSR-SRV

#### Requested Resource

Cpu:	600 cpu-units
Memory:	3072 MB
Profile:	custom
Disk:	10 MB
Vcpu:	1

#### Network information

interface-name:	<a href="#">eth0</a> <a href="#">eth1</a>
-----------------	-------------------------------------------

#### App Access

Console Access	<code>ssh -p {SSH_PORT} -i win7.pem appconsole@10.50.215.243</code>
VNC Access	VNC password :password

## Troubleshoot

En esta sección se brinda información que puede utilizar para resolver problemas en su configuración.

Para resolver problemas con la implementación, verifique el resultado de `ioxclient` o `/var/log/caf.log` en el CGM-SRV host OS.

Asegúrese de que NAT esté configurada correctamente para acceder a todos los recursos (marco de alojamiento de aplicaciones (CAF) de Cisco, Secure Shell (SSH) y VNC).