

# Konfigurieren des Windows VM zum CGM-SRV-Modul auf dem CGR1xxx

## Inhalt

[Einführung](#)

[Voraussetzungen](#)

[Anforderungen](#)

[Verwendete Komponenten](#)

[Hintergrundinformationen](#)

[Konfigurieren](#)

[Erstellen eines Windows VM-Images](#)

[Installieren Sie KVM auf Ihrem Linux-Rechner](#)

[Überprüfen der KVM-Installation](#)

[Erstellen einer Windows-VM](#)

[Bereitstellen des Windows VM-Image auf CGM-SRV](#)

[Überprüfen](#)

[Fehlerbehebung](#)

## Einführung

In diesem Dokument werden die erforderlichen Schritte zum Erstellen und Ausführen eines virtuellen Windows-Systems (VM) auf dem Modul Connected Grid (CGM) - System Server (SRV) beschrieben.

## Voraussetzungen

### Anforderungen

Cisco empfiehlt, über Kenntnisse in folgenden Bereichen zu verfügen:

- Linux
- Kernel Based Virtual Machine (KVM)
- Verständnis von Virtualisierungskonzepten

### Verwendete Komponenten

Die Informationen in diesem Dokument basieren auf den folgenden Software- und Hardwareversionen:

- Connected Grid Router (CGR) 1120
- CGM-SRV-XX-Modul
- Die Konfigurationsschritte für CGM-SRV werden vor diesem Leitfaden ausgeführt:
- Windows 7 Installations-ISO

- VNC-Viewer (Virtual Network Computing)

Die Informationen in diesem Dokument wurden von den Geräten in einer bestimmten Laborumgebung erstellt. Alle in diesem Dokument verwendeten Geräte haben mit einer leeren (Standard-)Konfiguration begonnen. Wenn Ihr Netzwerk in Betrieb ist, stellen Sie sicher, dass Sie die potenziellen Auswirkungen eines Befehls verstehen.

## Hintergrundinformationen

Wenn Sie IOx-Anwendungen oder virtuelle Systeme auf der CGR1000-Plattform ausführen möchten, können Sie das CGM-SRV-Computing-Modul verwenden. Das CGM-SRV-Modul ist eigentlich ein kleiner Server, der eine x86-Multicore-CPU, einen Speicher und einen Speicher enthält. Sowohl der CGR 1120 als auch der CGR 1240 können über eines dieser Module verfügen, um IOx-Funktionen hinzuzufügen.

Zum Zeitpunkt der Dokumenterstellung sind zwei Typen verfügbar:

Stock Keep Unit (SKU)	Solid-State-Laufwerke (SSD)	RAM	CPU
CGM-SRV-64	64 GB (50 GB nutzbar)	4 GB	4 Core, 800 MHz
CGM-SRV-128	128 GB (100 GB nutzbar)	4 GB	4 Core, 800 MHz

Jedes Modul verfügt außerdem über zwei USB-Speicherports und eine eigene externe Gigabit-Ethernet-Schnittstelle.

Wie bei allen anderen IOx-fähigen Geräten kann das Modul verschiedene IOx-Anwendungen hosten. Aufgrund der größeren Kapazität des CGM-SRV-Moduls kann es jedoch auch ein vollständig konfiguriertes Windows- oder Standard-Linux-Modul (z. B. Ubuntu oder CentOS) ausführen.

## Konfigurieren

### Erstellen eines Windows VM-Images

Um eine Windows VM auf dem CGM-SRV-Modul bereitzustellen, müssen Sie zunächst ein Image im QEMU-QCOW-Format erstellen, das die Windows-Installation enthält. Eine Möglichkeit zum Erstellen eines solchen Images ist KVM und virtuell auf einem Linux-Rechner.

Bei den weiter oben genannten Schritten handelt es sich nicht um den CGR1xxx oder CGM-SRV. Sie sind lediglich erforderliche Schritte zum Erstellen eines grundlegenden QCOW-Image für Windows 7 VM, das Sie im nächsten Schritt des CGM-SRV bereitstellen können.

Für diese Anleitung können Sie mit einer neu installierten CentOS7-Minimalinstallation beginnen. Die Schritte für andere Linux-Distributionen müssen ähnlich sein, können sich jedoch leicht unterscheiden.

### Installieren Sie KVM auf Ihrem Linux-Rechner

Schritt 1: Zunächst muss geprüft werden, ob das Host-System VM-Erweiterungen unterstützt. Auf der x86-Plattform sind dies entweder AMD-V oder Intel VT-X. Die meisten, wenn nicht sogar alle, modernen x86-CPU's unterstützen diese Erweiterungen. Selbst bei Ausführung einer VM bieten die meisten Hypervisoren die Möglichkeit, diese Erweiterungen zu übergeben/emulieren.

Um zu überprüfen, ob die installierte CPU diese Erweiterungen unterstützt, müssen Sie überprüfen, ob das `vmx` (für VT-X) oder `svm` (für AMD-V) Flag in der `cpuinfo-output` vorhanden ist.

```
[root@cen7 ~]# egrep -c '(vmx|svm)' /proc/cpuinfo
2
```

Wenn die Ausgabe dieses Befehls 0 ist, bedeutet dies, dass keine gefundene CPU die VM-Erweiterungen unterstützt. In diesem Fall können Sie überprüfen, ob diese Erweiterungen im BIOS oder Hypervisor aktiviert sind, wenn Sie eine VM zum Ausführen dieses Systems verwenden.

Schritt 2: Im nächsten Schritt wird eine Bridge erstellt, um ein Netzwerk für die VM bereitzustellen, die Sie auf KVM ausführen können.

Zunächst müssen Sie IP Forwards im Kernel aktivieren:

```
[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
```

Um die Bridge zu erstellen, muss die IP-Konfiguration von der eigentlichen Schnittstelle zur Bridge selbst verschoben werden, da dies die Schnittstelle ist, die die IP-Adresse besitzt.

Nachdem Sie eine Standardinstallation abgeschlossen haben, befindet sich die Netzwerkkonfiguration in `/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
```

Schritt 3: Wie Sie sehen können, gibt es derzeit eine Schnittstelle (neben der Loopback-Schnittstelle), die `eno16777736`. Sie müssen die IP-bezogene Konfiguration in eine Bridge-Schnittstelle verschieben, die Sie als `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
```

Schritt 4: Danach müssen Sie die IP-Konfiguration von der realen Schnittstelle bereinigen und mit der `virbr0`-Bridge verbinden:

```
[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
```

Schritt 5: Sobald die Netzwerkkonfiguration abgeschlossen ist, können Sie KVM installieren:

```
[root@cen7 ~]# sudo yum install kvm virt-manager libvirt virt-install qemu-kvm xauth dejavu-lgc-sans-fonts -y
...
Complete!
```

Schritt 6: Nach Abschluss der Installation empfiehlt es sich, diesen Computer neu zu starten, um die neu installierten Module und die Netzwerkkonfiguration anzuwenden:

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

## Überprüfen der KVM-Installation

Schritt 7: Nachdem der Neustart abgeschlossen ist, sollten Sie auf den Rechner über dieselbe IP zugreifen können, die auf der Bridge-Schnittstelle konfiguriert wurde. Sie müssen überprüfen, ob das KVM-Kernelmodul geladen ist:

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

Schritt 8: Wenn dies in Ordnung ist, können Sie versuchen, eine Verbindung mit dem Virus herzustellen:

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

Schritt 9: Ein letzter Schritt besteht darin, Port 5900 auf der Firewall auf diesem Computer für den VNC-Zugriff auf die Windows-Installation zu öffnen:

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

## Erstellen einer Windows-VM

Da Sie nun über ein System verfügen, das mit der KVM-Installation funktioniert, können Sie ein neues VM auf KVM starten und die Windows-Installationsdialoge durchlaufen.

Schritt 1: Kopieren Sie die Windows 7 Installations-ISO in Ihre VM (oder machen Sie sie über das Netzwerk zugänglich):

```
[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
```

Schritt 2: Erstellen Sie eine neue KVM-VM, und lassen Sie sie von Windows 7 ISO starten:

```
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

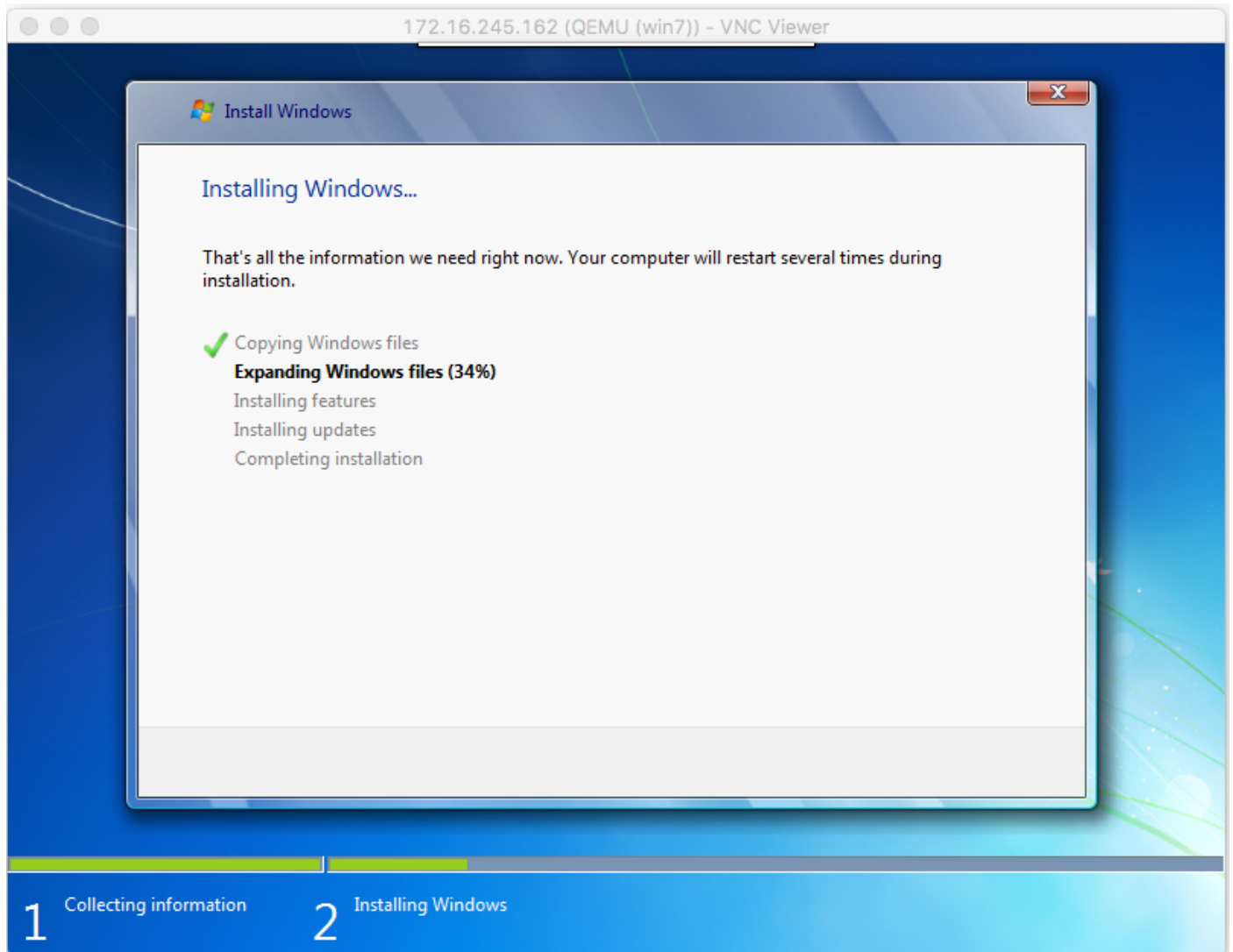
00:00:00

Creating domain...

00:00:00 | 0 B

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

Schritt 3: Nachdem das virtuelle System gestartet wurde, können Sie mit der Verwendung des VNC Viewer eine Verbindung zur IP-Adresse des Host-Rechners auf Port 5900 herstellen und die standardmäßige Windows-Installation wie im Abbild gezeigt beenden:



Wenn Windows zum Zeitpunkt der Installation neu startet, kann es erforderlich sein, das virtuelle System mit einem Virus neu zu starten, wenn dies nicht automatisch geschieht:

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

Schritt 4: Fahren Sie das virtuelle System nach Abschluss der Installation herunter. Sie haben nun ein QCOW-Image dieser Installation im Pfad angegeben, der bei der Erstellung des virtuellen Systems bereitgestellt wird: `/var/lib/libvirt/images/win7.img`. Dieser Image-Typ kann auf dem CGM-SRV bereitgestellt werden, um Windows auszuführen.

## Bereitstellen des Windows VM-Image auf CGM-SRV

Nachdem Sie jetzt den richtigen Image-Typ für die CGM-SRV haben, können Sie mit der Bereitstellung beginnen.

Schritt 1: Erstellen Sie ein Profil für einen Iox-Client, das Ihrer Konfiguration entspricht:

```
[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
```

In diesem Beispiel entspricht 10.X.X.X der ausgehenden Schnittstelle auf dem CGR1000, auf dem die Network Address Translation (NAT) für die Weiterleitung an Port 8443 auf dem CGM-SRV konfiguriert wurde.

Schritt 2: Nachdem der ioxclient konfiguriert ist, lassen Sie uns das zuvor erstellte Image in **vm.img** umbenennen, um ein Bit zu vereinfachen und es mithilfe von Secure Copy (SCP) mit ioxclient in CGM-SRV zu kopieren.

Konvertieren Sie optional das Disk-Image in das QCOW2-Format, da dies von der CGM-SRV erwartet wird. Neuere Versionen von virt-manager scheinen die Disk Images standardmäßig im QCOW3 Format zu erstellen.

Mit dem folgenden Befehl können Sie das Bild auf einfache Weise konvertieren:

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

Sobald Sie sicher sind, dass das Bild im richtigen Format ist, fahren Sie mit dem Umbenennen fort und kopieren Sie:

```
[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:/]
```

Dieser Transfer kann eine Weile dauern, da die Übertragungsraten von etwa 3-4 MB/s über Cisco IOS® zur CGM-SRV variieren. Die Datei wird im CGM-SRV-Modul auf **/mnt/data/vm/vm.img** kopiert.

Schritt 3: Während die Übertragung läuft (oder abgeschlossen ist), können Sie die **package.yaml-**

Datei erstellen. Diese Datei beschreibt IOx, was genau Sie bereitstellen möchten und wie Sie sie verpacken.

```
[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
```

Wie Sie in dieser Datei **package.yaml** sehen können, finden Sie Informationen unter **file://vm.img**, die dem tatsächlichen Speicherort von **mnt/data/vm/vm.img** auf dem CGM-SRV-Modul entspricht.

Schritt 4: Der nächste Schritt besteht in der Verpackung mit dem Einsatz von **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
```

Schritt 5: Nachdem Sie das Paket erstellt haben, können Sie es auf unserer CGM-SRV installieren. Die IOx-Anwendung/VM wird in diesem Beispiel als **win7** bezeichnet:

```
[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
```

Schritt 6: Bevor Sie die Win7 IOx VM aktivieren können, müssen Sie eine Payload-JSON-Datei erstellen, die das VNC-Kennwort für diese VM festlegt:

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

Schritt 7: Mit der **vnc.json** Payload können Sie die Win7 IOx VM aktivieren:

```
[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
```

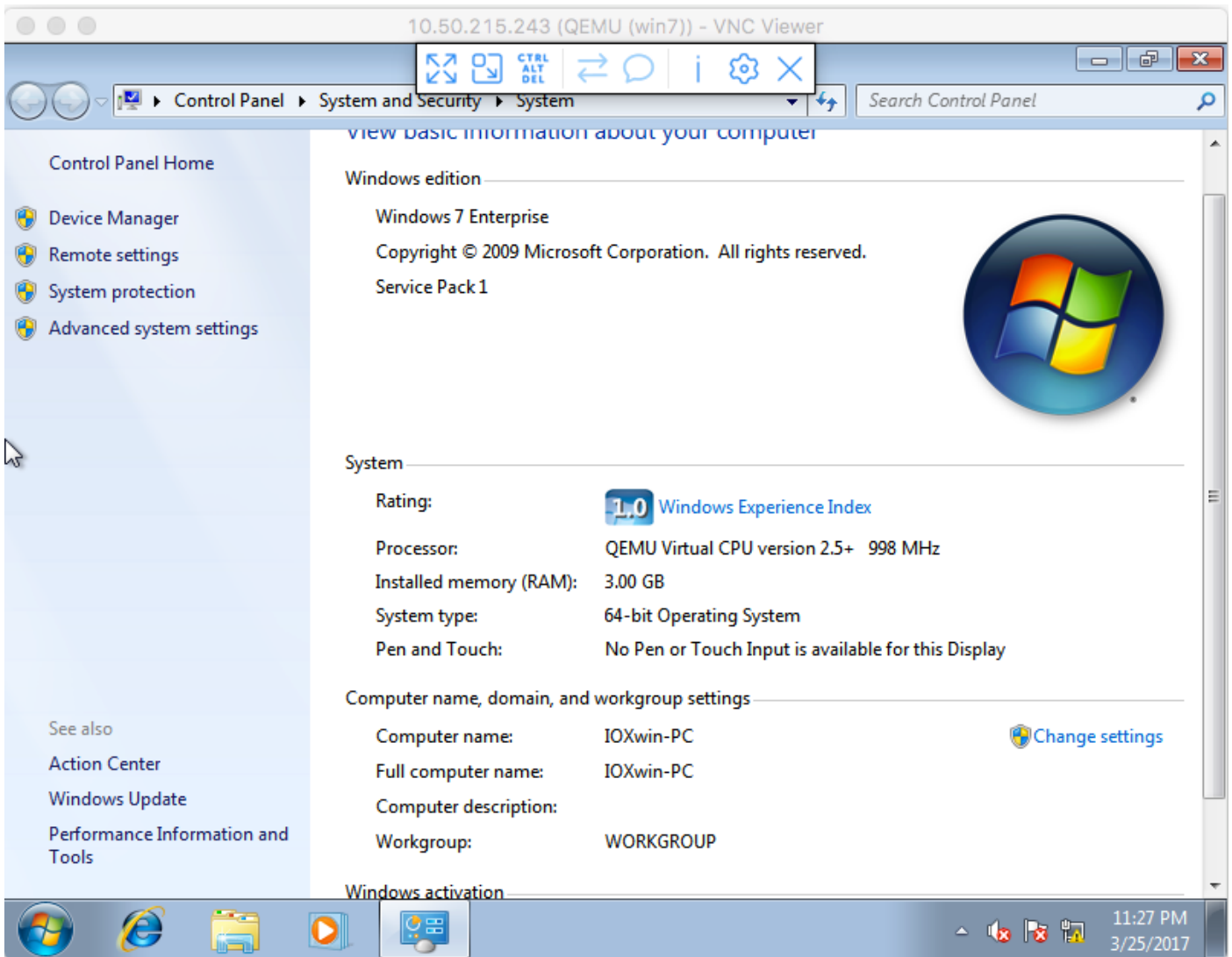
Schritt 8: Der letzte Schritt mit ioxclient ist der Start der VM:

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

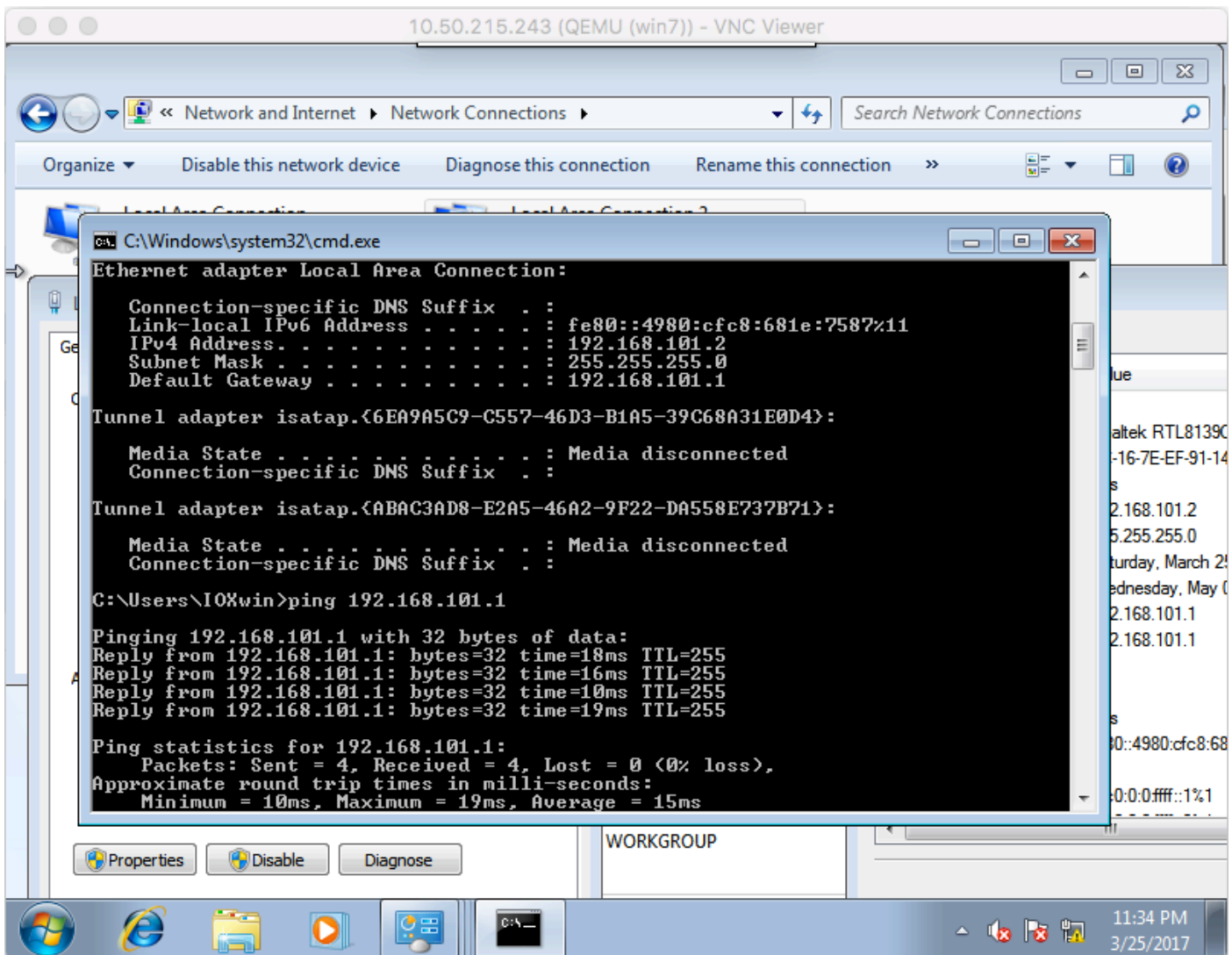
Zu diesem Zeitpunkt wird das Windows VM auf dem CGM-SRV ausgeführt, und Sie können mit der Verwendung beginnen.

Um Zugriff auf die Windows-Konsole zu erhalten, können Sie VNC Viewer auf der ausgehenden Schnittstelle des CGR1000 und des Ports 5900 verwenden, wie im Bild gezeigt:





Aus Netzwerksicht haben Sie beschlossen, eth0 und eth1 der Win7 IOx VM mit der **package.yaml**-Datei zu übergeben, wie im Bild gezeigt:



Wie Sie sehen können, haben diese Schnittstellen eine IP vom DHCP-Server erhalten, der unter Cisco IOS® ausgeführt wird und ohne weitere Konfiguration verwendet werden kann.

## Überprüfen

In diesem Abschnitt überprüfen Sie, ob Ihre Konfiguration ordnungsgemäß funktioniert.

So prüfen Sie, ob die VM ausgeführt wird:

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

Sie können den Status auch über den lokalen Manager überprüfen, wie in der Abbildung gezeigt:

#### 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

## Fehlerbehebung

Dieser Abschnitt enthält Informationen, die Sie zur Fehlerbehebung bei Ihrer Konfiguration verwenden können.

Um Probleme bei der Bereitstellung zu beheben, überprüfen Sie die Ausgabe von `ioxclient` oder `/var/log/caf.log` auf dem CGM-SRV-Host-Betriebssystem.

Stellen Sie sicher, dass NAT korrekt konfiguriert ist, um auf alle Ressourcen zuzugreifen (Cisco Application-Hosting Framework (CAF), Secure Shell (SSH), VNC).