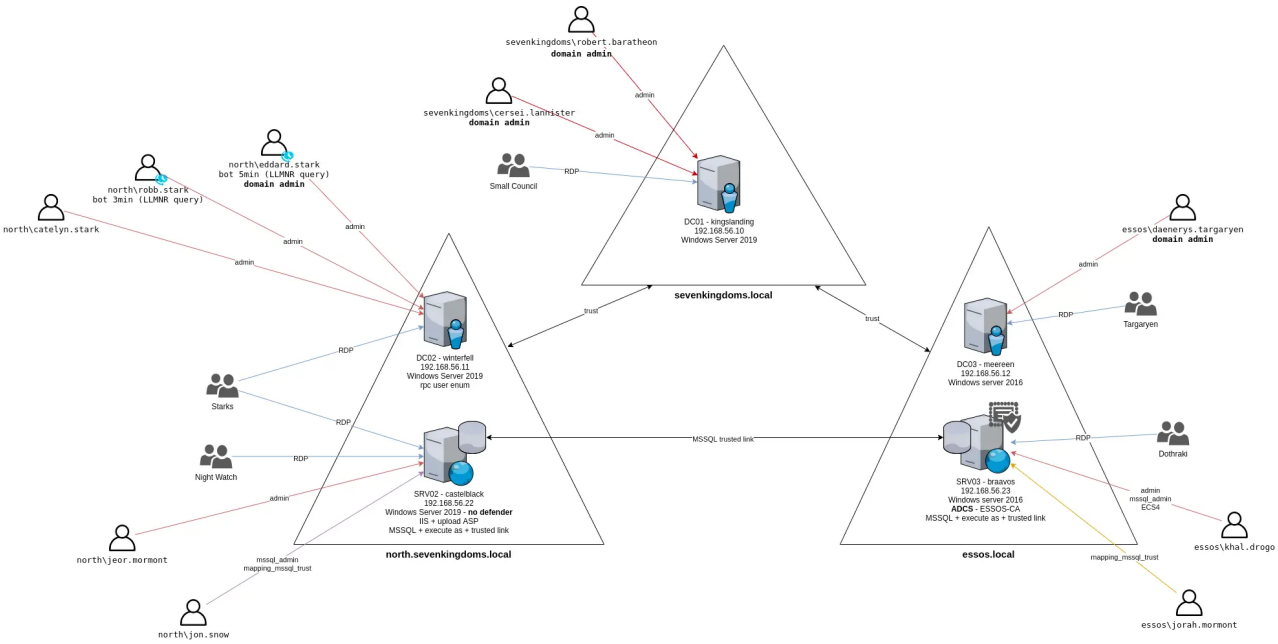
Setting up a domain environment for the game of active directory (GOAD)

lca.xlog.app/game-of-active-directoryGOAD-yu-huan-jing-da-jian

lca

The second version of "Game Of Active directory", project address: https://github.com/Orange-Cyberdefense/GOAD

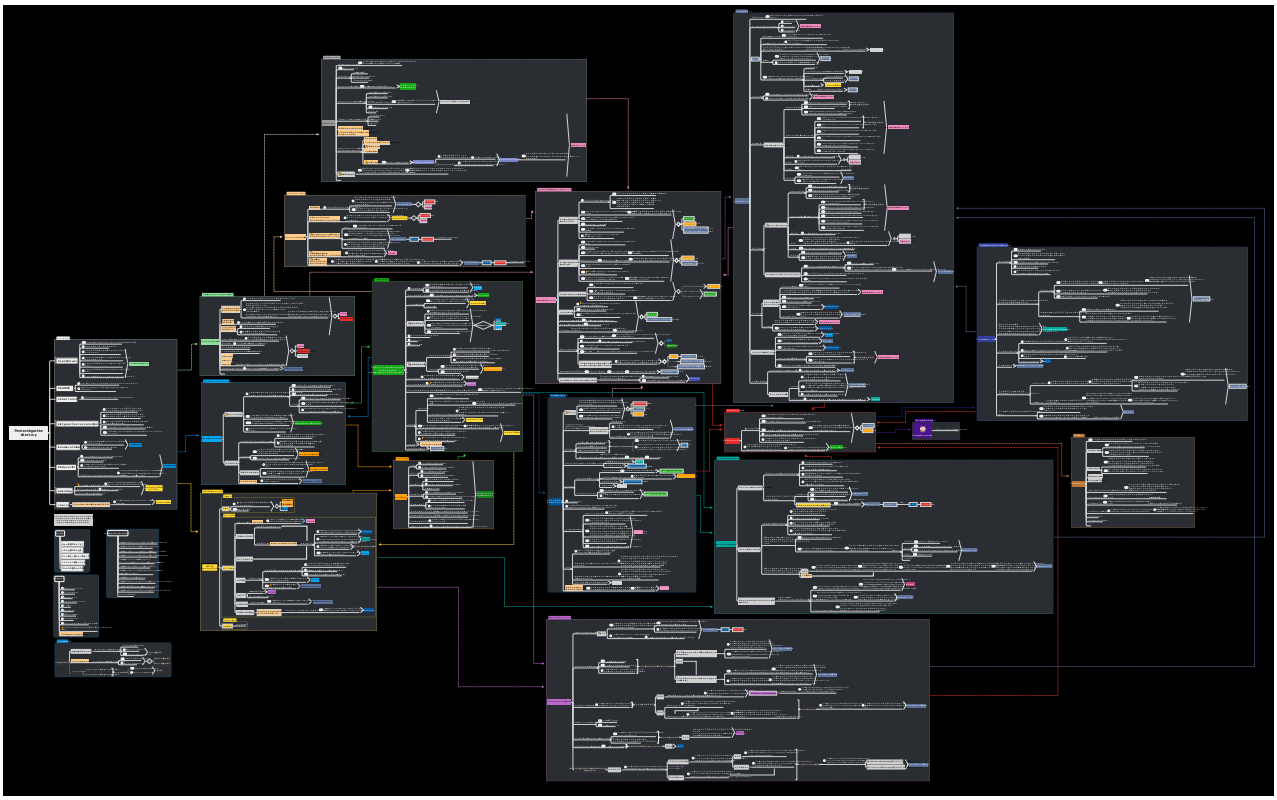
The domain target environment installs 5 Windows instances (three DCs and two regular domain hosts) through vagrang, as shown in the topology diagram:



Some roadmaps provided by the official (**vulnerability points**):

- Password-reuse-between-computer-(PTH)
- Spray-User=-Password
- Password-in-description
- SMB-share-anonymous
- SMB-not-signed
- Responder
- Zerologon
- Windows-defender
- ASREPRoast
- Kerberoasting
- AD-Acl-abuse
- Unconstraint-delegation
- Ntlm-relay
- Constrained-delegation
- Install-MSSQL
- MSSQL-trusted-link

- MSSQL-impersonate
- Install-IIS
- Upload-asp-app
- Multiples-forest
- Anonymous-RPC-user-listing
- Child-parent-domain
- Generate-certificate-and-enable-ldaps
- ADCS---ESC-1/2/3/4/6/8
- Certifry
- Samaccountname/nopac
- Petitpotam-unauthent
- Printerbug
- Drop-the-mic
- Shadow-credentials
- Mitm6
- Add-LAPS
- GPO-abuse
- Add-Webdav
- Add-RDP-bot
- Add-full-proxmox-integration
- Add-Gmsa-(receipe-created)
- Add-azure-support
- Refactoring-lab-and-providers
- Protected-Users
- Account-is-sensitive
- Add-PPL
- Add-Gmsa
- Groups-inside-groups
- Shares-with-secrets-(all,-sysvol)



Original image:

https://orange-cyberdefense.github.io/ocd-mindmaps/img/pentest_ad_dark_2022_11.svg

Host environment#

Virtual machine based on the target	VMware
Operating system	Ubuntu 22.04
Allocated memory	24G
Disk space	500G

01 Install Ubuntu#

The first step is to install an Ubuntu 22.04 virtual machine based on VMware. The following steps are based on this Ubuntu 22.04 virtual machine.

02 Update#

```
sudo apt update
sudo apt upgrade
```

03 Install VirtualBox#

```
sudo apt install virtualbox
```

04 Install Vagrant#

```
wget https://releases.hashicorp.com/vagrant/2.2.19/vagrant_2.2.19_x86_64.deb
sudo apt install ./vagrant_2.2.19_x86_64.deb
vagrant --version
```

05 Install Python#

```
sudo apt install python3-pip
pip3 --version
```

06 Install Python virtual environment#

```
sudo apt install python3-venv
```

07 Clone the GOAD V2 repository#

Git tool needs to be installed first

```
sudo apt-get install git-all
```

Clone to the user's home directory

```
cd ~/
git clone https://github.com/Orange-Cyberdefense/GOAD.git
```

08 Create a Python virtual environment#

```
python3 -m venv venvGOAD
```

09 Activate the virtual environment#

```
cd GOAD/ansible
source ~/venvGOAD/bin/activate
```

10 Install the Ansible module#

```
pip install ansible-core
#or
python3 -m pip install ansible-core==2.12.6
```

11 Install pywinrm#

```
pip install pywinrm
```

12 Install Galaxy dependencies#

```
ansible-galaxy install -r requirements.yml
```

13 System installation#

Before installation, you can use the goad.sh script in the GOAD directory to check if the environment is ready

```
./goad.sh -t check -l GOAD -p virtualbox -m local
```

Here are the solutions to some installation problems

Problem 1: Proxy

```
1. ERROR: Could not install packages due to an OSError: Missing dependencies for
SOCKS support.
2. fatal: [srv03]: UNREACHABLE! => {"changed": false, "msg": "ssl: Missing
dependencies for SOCKS support.", "unreachable": true}
```

If you encounter socks-related issues, you need to disable the proxy. Since you need to install the operating system, if you use the domestic network to pull, the speed will be very slow. So I set up a proxy in Ubuntu, so that the download speed of the operating system is very fast. You can temporarily turn it off first, and then turn on the proxy when the download speed of the system is too slow.

Solution:

```
unset ALL_PROXY
unset all_proxy
```

Problem 2: VMware does not support virtualization

Solution:

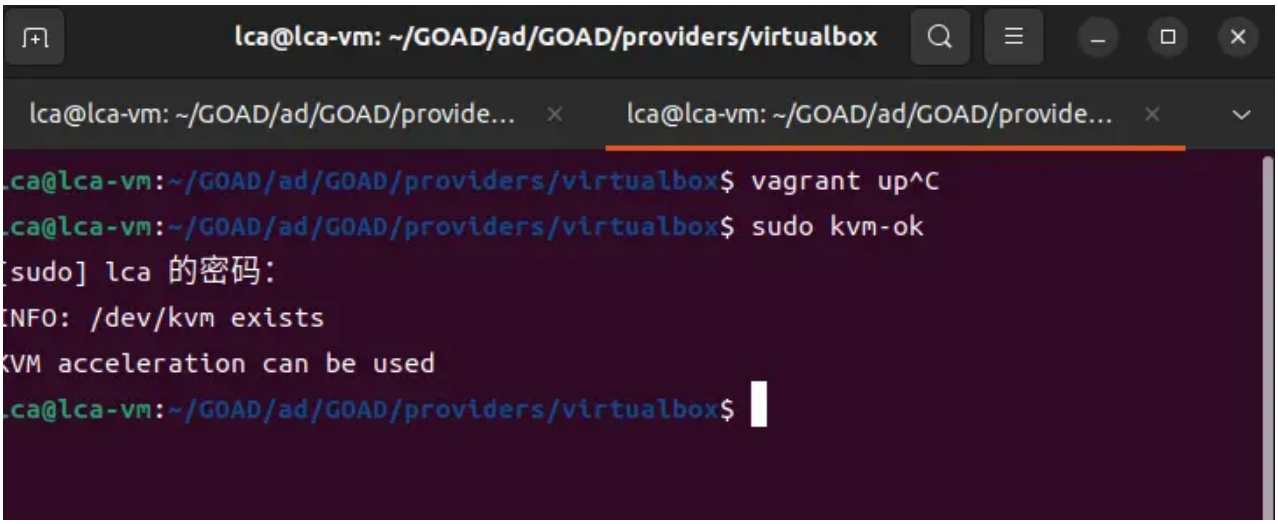
You can refer to: Solve the problem that the virtual machine VM opens the virtualization Intel-VT-x/EPT or AMD-V/RVI(V) and the computer blue screens or displays that this platform does not support virtualization

This is because the installation of Docker and Hyper-V on the host system conflicts, so you need to disable the relevant functions of Hyper-V

You can use the following command to check if the virtual machine supports KVM virtualization

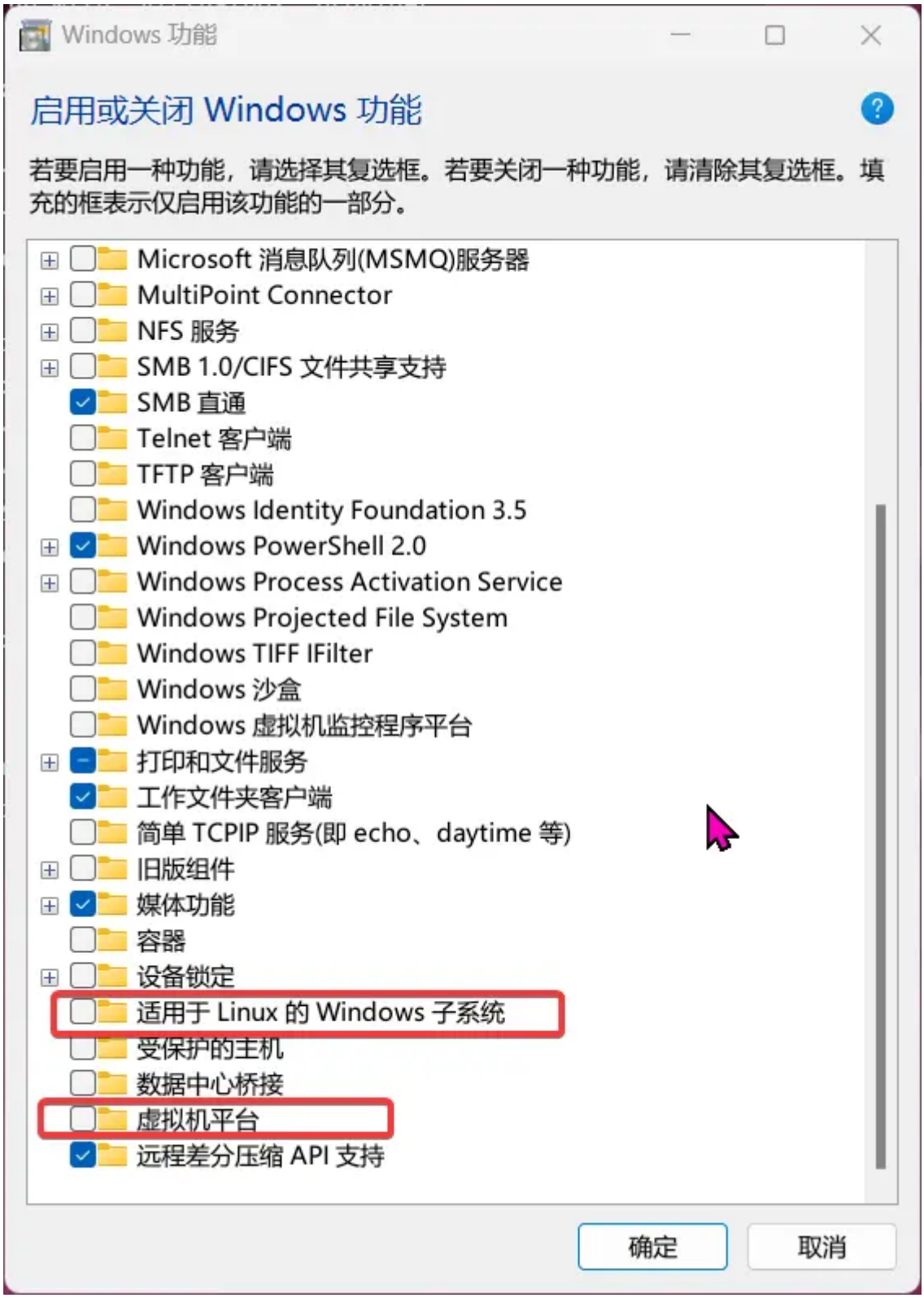
```
sudo apt install -y cpu-checker
sudo kvm-ok
```

If it is the following output, then it supports KVM and will not report this error

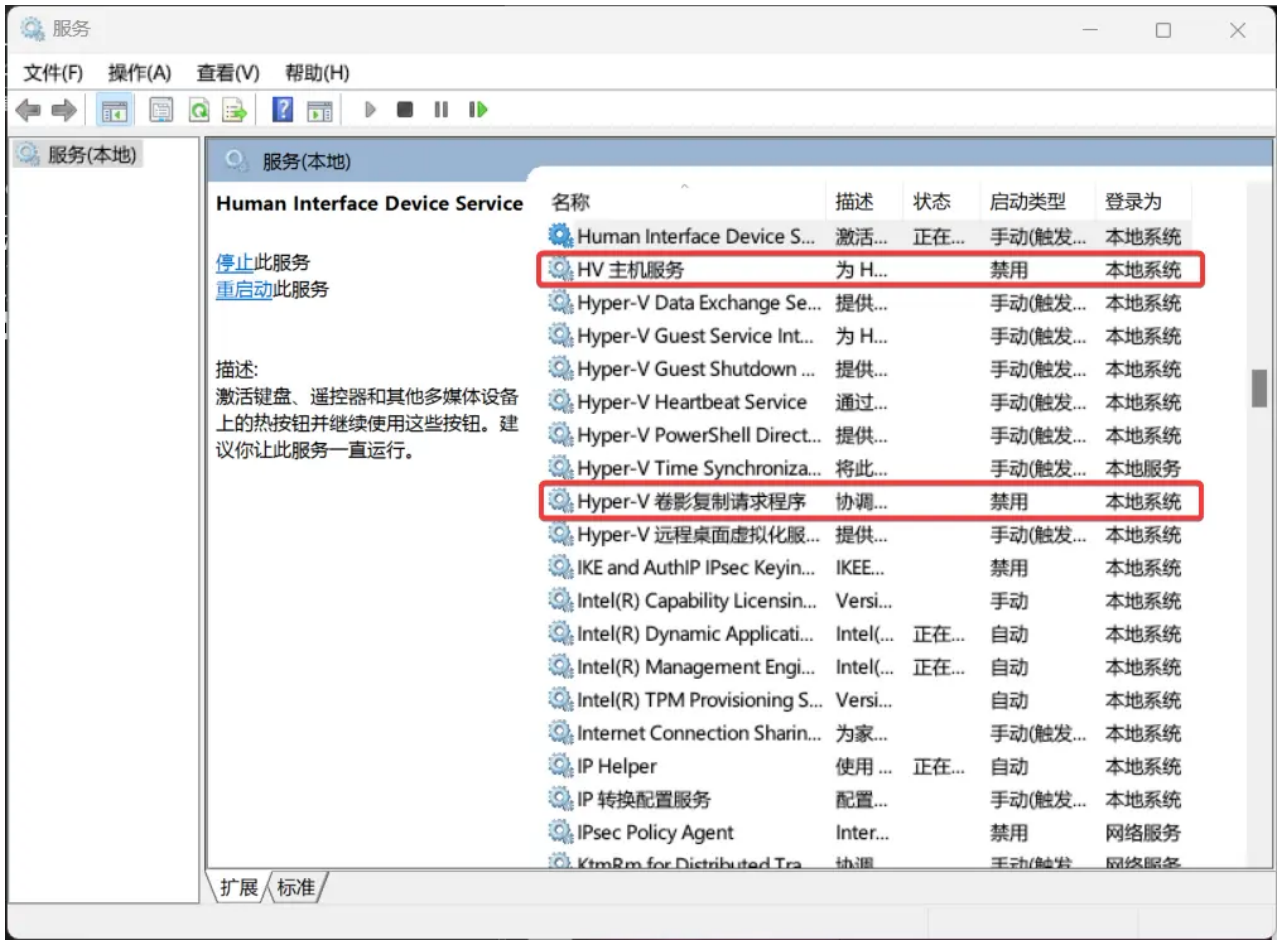


If it is a different result, please refer to the above article to resolve it, that is, the following steps

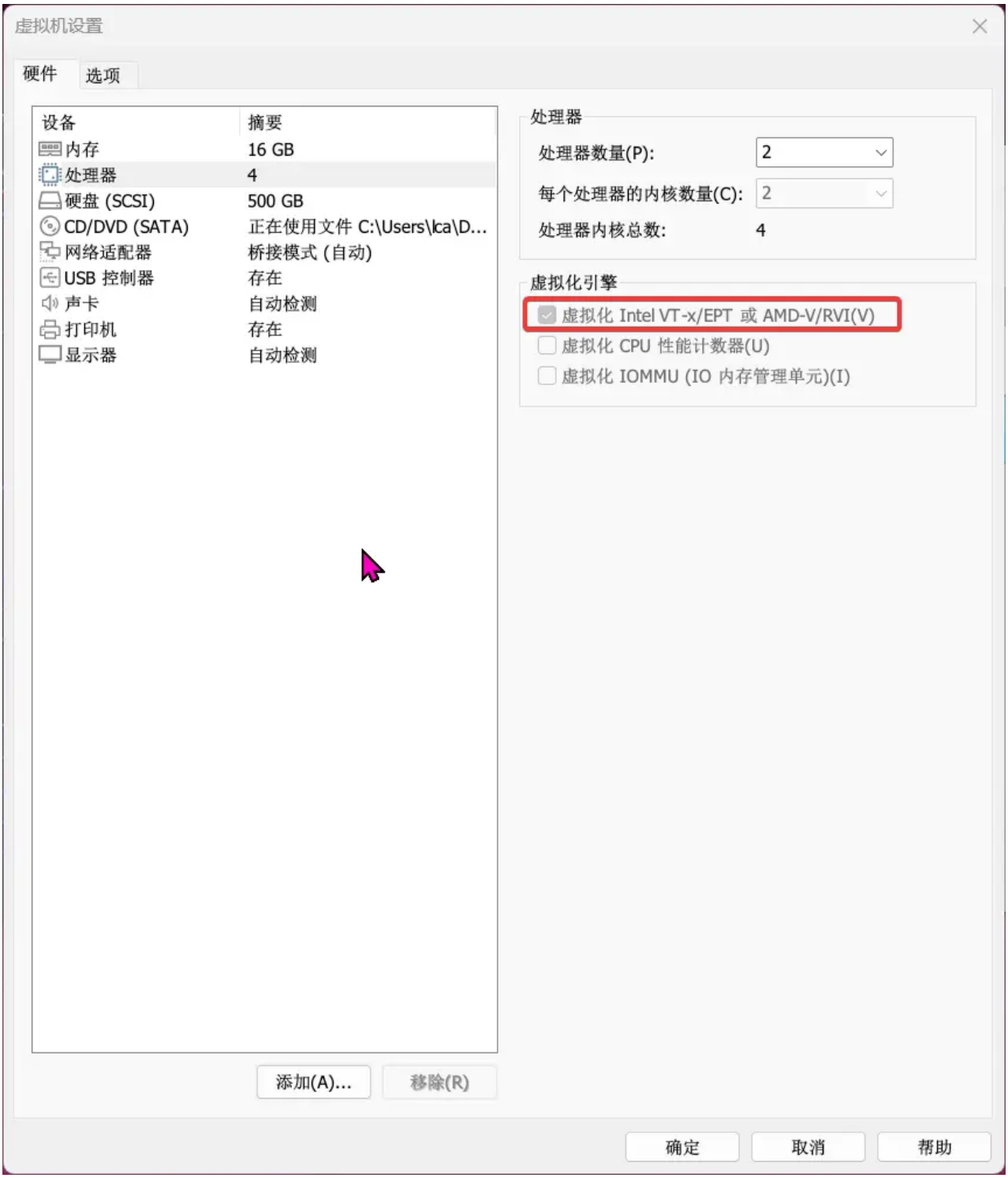
1. Disable some virtualization functions, and also turn off Hyper-V



2. Disable related services



3. In the virtual machine settings-Processor-Virtualization Engine, select the following options

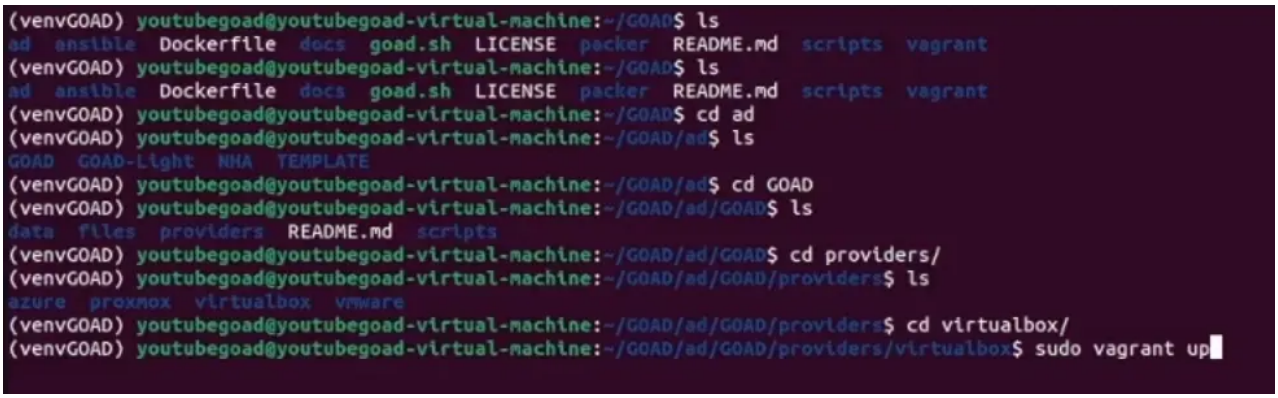


Problem 3: 'base' could not be found

```
default: Box 'base' could not be found. Attempting to find and install…
```

Solution:

Go to `~/GOAD/ad/GOAD/provider/virtualbox` and execute `vagrant up`



Problem 4: Memory issue

If the pulled system exits abnormally, it means that the memory is not enough

If none of the above problems occur, then start installing the target machine environment, as above

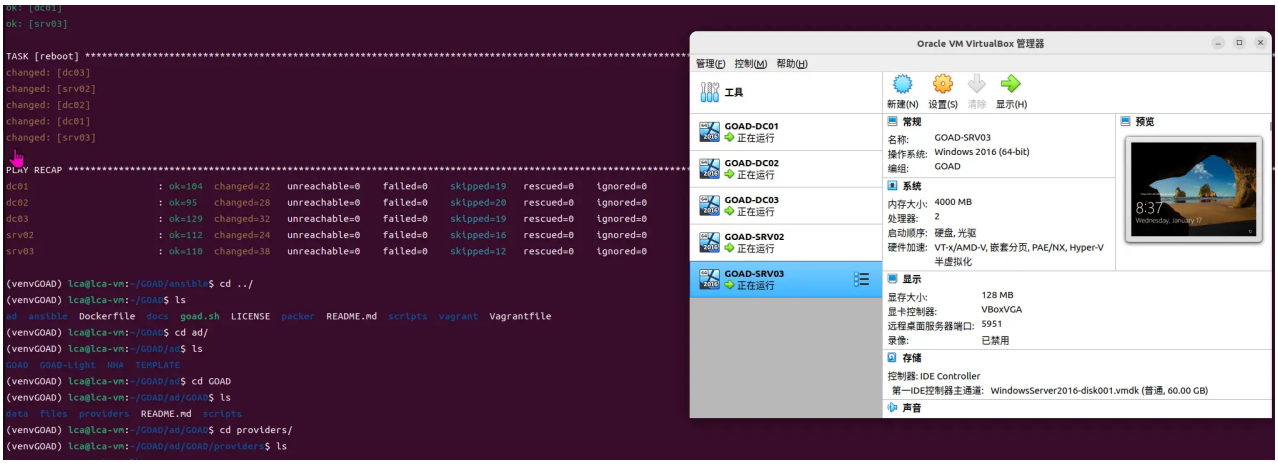
Go to `~/GOAD/ad/GOAD/provider/virtualbox` and execute `vagrant up`

Note: If the speed is too slow, use a proxy

If you have Clash running on your host, you can specify the IP of the host plus port 7890 in the proxy in the network settings of Ubuntu.

14 Celebrate#

After two nights of hard work, I finally see the following results 😭



15 References#

Video:
https://www.youtube.com/watch?v=haiTcZpqdQg

Articles:
https://mayfly277.github.io/posts/GOADv2/
https://github.com/quincyntuli/GOAD-v2-Installation-Notes
https://github.com/Orange-Cyberdefense/GOAD

1. 1. Host environment
2. 2. 01 Install Ubuntu
3. 3. 02 Update
4. 4. 03 Install VirtualBox
5. 5. 04 Install Vagrant
6. 6. 05 Install Python
7. 7. 06 Install Python virtual environment
8. 8. 07 Clone the GOAD V2 repository
9. 9. 08 Create a Python virtual environment
10. 10. 09 Activate the virtual environment
11. 11. 10 Install the Ansible module
12. 12. 11 Install pywinrm
13. 13. 12 Install Galaxy dependencies
14. 14. 13 System installation
15. 15. 14 Celebrate
16. 16. 15 References