OxOffice Online

技術手冊

提供 OxOffice Online 在安裝、設定、整合及佈署的相關文件說明。

- OxOffice Online 主程式安裝說明
  - 在 CentOS 7 環境下安裝 (社群版)
  - 在 Ubuntu 18.04 環境下安裝 (社群版)
  - 在 CentOS 7 PVE LXC 環境下安裝 (社群版)
  - 在 Ubuntu 18.04 PVE LXC 環境下安裝 (社群版)
  - 在 CentOS 7 環境下安裝 OxOffice Online v4 社群版
  - 在 Rocky Linux 8 環境下安裝 OxOffice Online v4 社群版
  - 在 Ubuntu 20.04 環境下安裝 OxOffice Online v4 社群版
  - 在 PVE 的 LXC 下以 CentOS 7 環境安裝 OxOffice Online v4 社群版
  - 在 PVE 的 LXC 下以 Rocky Linux 8 環境安裝 OxOffice Online v4 社群版
  - 在 PVE 的 LXC 下以 Ubuntu 20.04 環境安裝 OxOffice Online v4 社群版
  - 在 Rocky Linux 8 環境下安裝 OxOffice Online v5 社群版
  - 在 Ubuntu 24.04 環境下安裝 OxOffice Online v5 社群版
- OxOffice Online 佈署設定
  - 設定啟用 OxOOL SSL 憑證
  - 設定讓其它的 Nextcloud 來存取 OxOOL
- OxOffice Online 社群版編譯說明
  - 在 CentOS 7 環境下編譯 OxOOL
  - 在 Ubuntu 18.04 環境下編譯 OxOOL
  - 在 CentOS 7 環境下編譯 OxOffice Online v4 社群版
  - 在 Rocky Linux 8 環境下編譯 OxOffice Online v4 社群版

在 Ubuntu 20.04 環境下編譯 OxOffice Online v4 社群版
在 Rocky Linux 9 環境下編譯 OxOffice Online v4 社群版
在 Rocky Linux 8 環境下編譯 OxOffice Online v5 社群版
在 Ubuntu 24.04 環境下編譯 OxOffice Online v5 社群版

OxOffice Online v4 模組開發
OxOOL v4 模組編譯手冊
OxOOL 模組開發手冊

OxOffice Online 設定檔說明
OxOffice Online 管理主控臺說明

OxOffice Online

主程式安裝說明

本文件將說明如何從無到有重新安裝 OxOffice Online (OxOOL) 應用。如果沒有安裝過 Nextcloud 的朋友，也可以參考 Nextcloud 快速安裝體驗（ https://docs.ossii.com.tw/books/nextcloud），在同一台安裝來測試玩看看。

OxOffice Online 主程式安裝說明

在 CentOS 7 環境下安裝 (社群版)

- 下載、安裝 CentOS 7 最新版本 (最新版本: 2009 載點 Minimal、Desktop) – 安裝 CentOS

  可參考文件 [1] [2] [3]

建議請記得設定成固定 IP, 可參考上述安裝文件

- 以 root 重新登入後執行以下指令來更新、安裝相關的指令及 OxOOL Community 套件

```
yum update -y
yum install vim net-tools unzip wget -y
cd /etc/yum.repos.d
wget http://www.oxoffice.com.tw/yum.repo/oxool-community.repo
yum update -y
yum groupinstall "OxOOL Community Group" -y
```

- 關閉 SELinux

```
vim /etc/selinux/config # vim 可代換成任何您慣用的文字編輯器
# 尋找 SELinux=enforcing 此行並將 enforcing 改為 disabled 或 permissive 並存檔離開
```

- 設定 firewalld 並開啟預設的 9980 port

```
firewall-cmd --zone=public --permanent --add-service=http
firewall-cmd --zone=public --permanent --add-port=9980/tcp
```

- 設定開機要啟動的服務

```
systemctl enable oxool
reboot
```

- 重新啟動主機, 重新使用 root 登入並確認 OxOOL Community 正常啟動

```
netstat -tlnp
```

結果如下:

```
tcp        0      0 127.0.0.1:9981□         0.0.0.0:*□LISTEN      644/oxool
tcp6       0      0 :::9980□□               :::*      □LISTEN      644/oxool
```

如果出現以上的結果，代表架設成功囉!

附註: 如果要與 Nextcloud 檔案儲存服務整合，請參考 Nextcloud 20 架設教學。

□ Ubuntu 18.04 □□□□ (□□)

- □□□□ Ubuntu 18.04 □□□□□ 64 □ Server □□□□□ □□ □□□□□
  Desktop □ 64 □□□□□□□□□□ [1]□ [2]

```
□□□□□□□ IP□□□□□□□□□ [3]
```

- □□ User □□□□□□□□□□□□□□□□□□□□ OxOOL Community □□

```
sudo apt update
sudo apt upgrade -y
sudo apt install vim openssh-server net-tools curl -y
# □□□□□□□□□□□□□□□□ yes □□
```

curl http://www.oxoffice.com.tw/apt/ubuntu/oxool-deb.key | sudo apt-key add echo "deb [arch=amd64] http://www.oxoffice.com.tw/apt/ubuntu/ oxool-v3 main" > ~/oxool.list sudo mv ~/oxool.list /etc/apt/sources.list.d/ sudo apt update sudo apt install oxool oxoffice* -y

- □□□□□□□□□

```
sudo systemctl enable oxool
sudo reboot
```

- □□□□□□□□□□□□□□□□ OxOOL Community □□□

```
sudo netstat -tlnp
```

□□□□

```
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program
name
tcp        0      0 127.0.0.1:9981          0.0.0.0:*               LISTEN
22383/oxool
```

```
tcp        0      0 127.0.0.53:53           0.0.0.0:*               LISTEN      6523/systemd-
resolv
tcp        0      0 0.0.0.0:22              0.0.0.0:*               LISTEN
4389/sshd
tcp        0      0 127.0.0.1:25            0.0.0.0:*               LISTEN
2575/master
tcp6       0      0 :::9980                 :::*                    LISTEN
22383/oxool
tcp6       0      0 :::22                   :::*                    LISTEN
4389/sshd
tcp6       0      0 ::1:25                  :::*                    LISTEN      2575/master
```

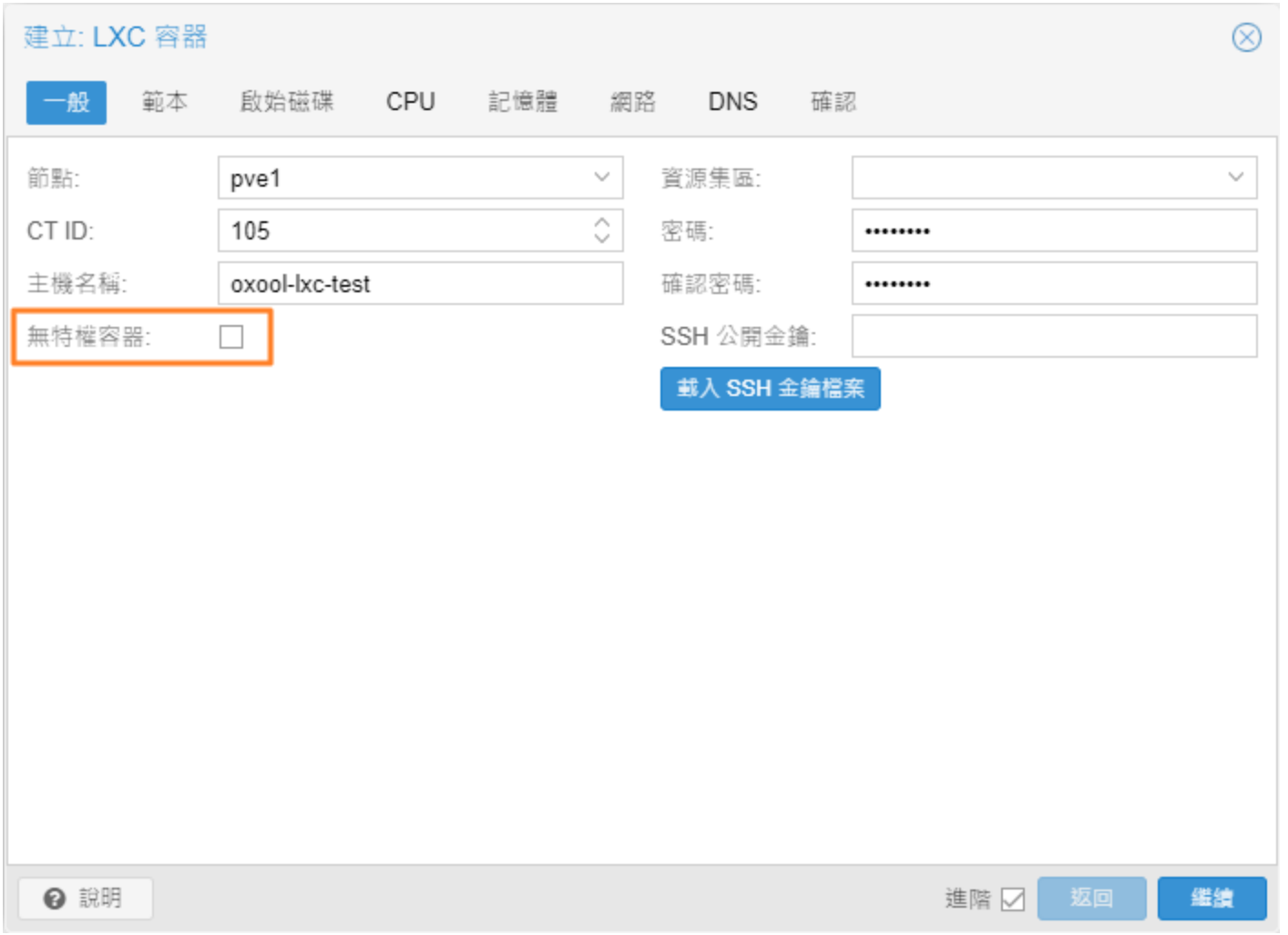
□□□□□□ **ezgo** □□□ **Franklin** □□□□ ~

□□□□□□□ Nextcloud □□□□□□□□□□□□ Nextcloud 20 □□□□ □

在 CentOS 7 PVE LXC 環境下安裝 (社群版)
請先使用 PVE 下載「 centos-7-default_20190926_amd64.tar.xz」的 LXC 範本。



建立一個新的「 CT
」，參考內容如下圖，但目前建議將「無特權容器」選項取消勾選，否則部份功能會有問題。


建立: LXC 容器
一般
範本
啟始磁碟
CPU
記憶體
網路
DNS
確認
節點:
pve1
資源集區:
CT ID:
105
密碼:
主機名稱:
oxool-lxc-test
確認密碼:
無特權容器:
SSH 公開金鑰:
載入 SSH 金鑰檔案
說明
進階
返回
繼續

建立: LXC 容器

一般 範本 啟始磁碟 CPU 記憶體 網路 DNS 確認

Key ↑	Value
cores	2
hostname	oxool-lxc-test
memory	2048
nameserver	8.8.8.8
net0	bridge=vmbr0,name=eth0,ip=192.168.3.211/32,gw=192.168.3.1,firewall=1
nodename	pve1
ostemplate	local:vztmpl/centos-7-default_20190926_amd64.tar.xz
pool	
rootfs	OSSIIPVE1:8
searchdomain	8.8.8.8
swap	2048
vmid	105

☐ 建立完成後開機

進階 ☑ 返回 完成

Task viewer: CT 105 - 建立

輸出 狀態

停止

```
extracting archive '/var/lib/vz/template/cache/centos-7-default_20190926_amd64.tar.xz'
Total bytes read: 422809600 (404MiB, 55MiB/s)
Detected container architecture: amd64
Creating SSH host key 'ssh_host_ecdsa_key' - this may take some time ...
done: SHA256:v9clmA/kbCbgeoJG1XG0zkt2GEgD4jQjVT7kh0ik0A0 root@oxool-lxc-test
Creating SSH host key 'ssh_host_rsa_key' - this may take some time ...
done: SHA256:DrDj/hQMM9jr8I3Td721i0CRtpHcGt6WkOJ4Hlo5z9s root@oxool-lxc-test
Creating SSH host key 'ssh_host_dsa_key' - this may take some time ...
done: SHA256:JxIzMaRfoPj2eoaiYAxaQOSFNdn0eICnCH1L9gV60Ws root@oxool-lxc-test
Creating SSH host key 'ssh_host_ed25519_key' - this may take some time ...
done: SHA256:s6y4Na/X9+m95rUW85OM5QFc8cNnDQthVcQitUEDlbU root@oxool-lxc-test
TASK OK
```

- □□□□□□ (□□ 10 □□□□□□)□□□□□□□□□□

```
yum install vim openssh* net-tools unzip wget -y
cd /etc/yum.repos.d
wget --no-check-certificate http://www.oxoffice.com.tw/yum.repo/oxool-community.repo
yum update -y
yum groupinstall "OxOOL Community Group" -y
```

- □□□□□□□□□□

```
systemctl enable oxool
reboot
```

- □□□□□□□□□ OxOOL Community □□□□

```
netstat -tlnp
```

□□□□□

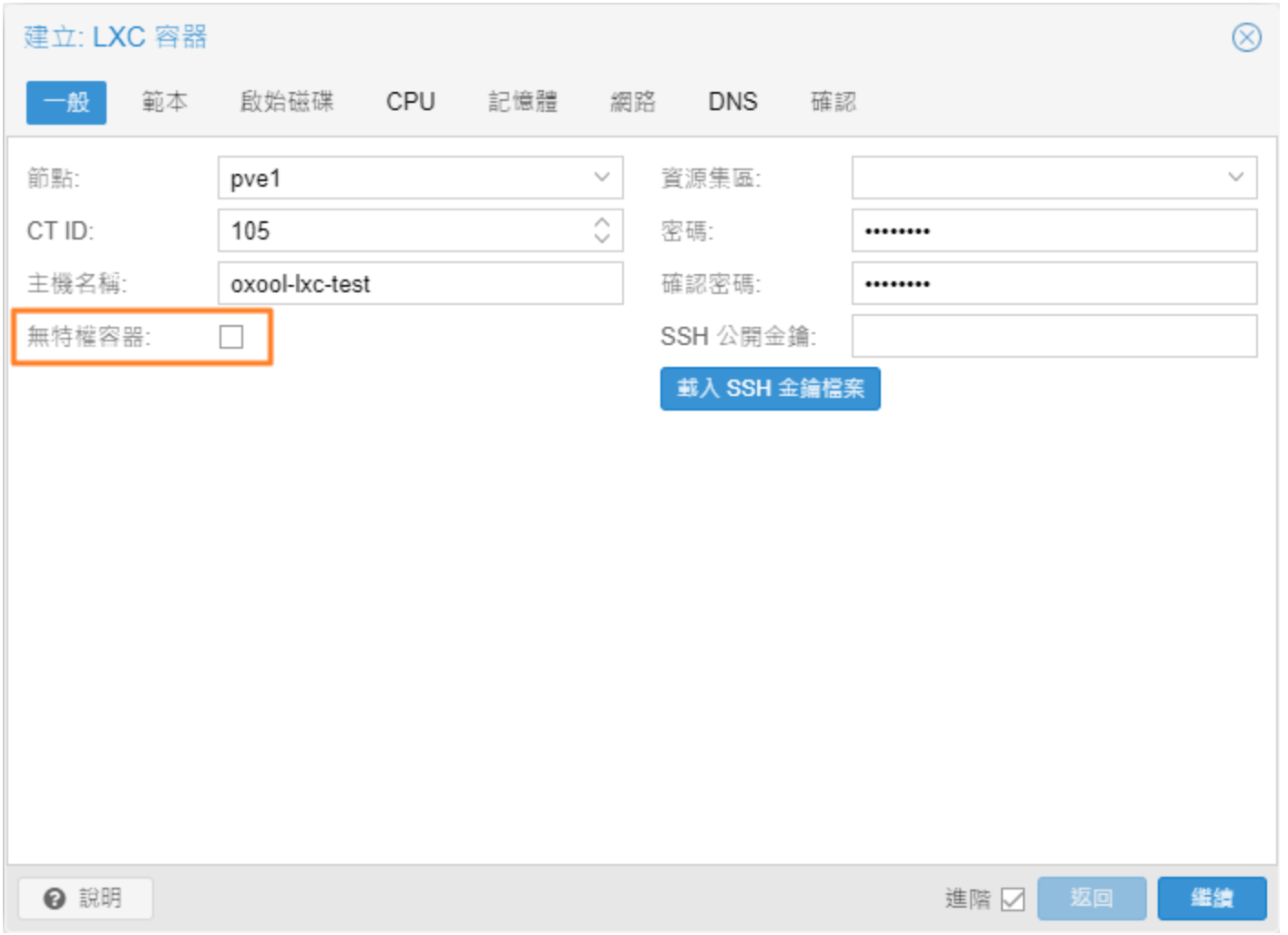
```
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program
name
tcp        0      0 127.0.0.1:9981          0.0.0.0:*               LISTEN
1049/oxool
tcp6       0      0 :::9980                 :::*                    LISTEN      1049/oxool
```

□□□□□□□□□□□□□□□□□□□ →□□□□□□□□□□□ OxOOL □□□□ LXC □□□

在 Ubuntu 18.04 PVE LXC 環境下安裝 (社群版)

- 請先使用 PVE 下載「 ubuntu-18.04-standard_18.04.1-1_amd64.tar.gz」的 LXC 範本。

- 建立一個新的「 CT 」，參考內容如下圖，但目前建議將「無特權容器」選項取消勾選，否則部份功能會有問題。


建立: LXC 容器
一般
範本
啟始磁碟
CPU
記憶體
網路
DNS
確認
節點:
pve1
資源集區:
CT ID:
105
密碼:
主機名稱:
oxool-lxc-test
確認密碼:
無特權容器:
SSH 公開金鑰:
載入 SSH 金鑰檔案
說明
進階
返回
繼續

建立: LXC 容器

一般 範本 啟動磁碟 CPU 記憶體 網路 DNS 確認

Key ↑	Value
cores	2
hostname	oxool-lxc-test
memory	2048
nameserver	8.8.8.8
net0	bridge=vmbr0,name=eth0,ip=192.168.3.211/32,gw=192.168.3.1,firewall=1
nodename	pve1
ostemplate	local:vztmpl/ubuntu-18.04-standard_18.04.1-1_amd64.tar.gz
pool	
rootfs	OSSIIPVE1:8
searchdomain	8.8.8.8
swap	2048
vmid	105

☐ 建立完成後開機

進階 ☑ 返回 完成

Task viewer: CT 105 - 建立

輸出 狀態

停止

```
extracting archive '/var/lib/vz/template/cache/ubuntu-18.04-standard_18.04.1-1_amd64.tar.gz'
Total bytes read: 637347840 (608MiB, 65MiB/s)
Detected container architecture: amd64
Creating SSH host key 'ssh_host_ed25519_key' - this may take some time ...
done: SHA256:MLsz7ueUypNPXlmVkLTIp4dyuqt6d8Dr1UX7/UpHRDk root@oxool-lxc-test
Creating SSH host key 'ssh_host_ecdsa_key' - this may take some time ...
done: SHA256:dT5WqOKTfwu1jv25t+XhbwtJ8xVnSTpk5I3TwGfexuo root@oxool-lxc-test
Creating SSH host key 'ssh_host_dsa_key' - this may take some time ...
done: SHA256:deoBi6GkKZco02laOyGs15uTFJ8CanNyvQd51WZXO2I root@oxool-lxc-test
Creating SSH host key 'ssh_host_rsa_key' - this may take some time ...
done: SHA256:VDbC9Eps/5X6nFooOt7pjYdA1Qs9Vv3jhU8KM7uWfNM root@oxool-lxc-test
TASK OK
```

- □□□□□□ (□□ 20 □□□□□)□□□□□□□□□□

```
apt update
apt upgrade -y
apt install vim openssh-server net-tools curl gnupg2 -y
# □□□□□□□□□□□□□□□□□□□□ yes □□

curl http://www.oxoffice.com.tw/apt/ubuntu/oxool-deb.key | apt-key add
echo "deb [arch=amd64] http://www.oxoffice.com.tw/apt/ubuntu/ oxool-v3 main" > ~/oxool.list
mv ~/oxool.list /etc/apt/sources.list.d/
apt update
apt install oxool oxoffice* -y
```

- □□□□□□□□□□

```
systemctl enable oxool
reboot
```

- □□□□□□□□□□ OxOOL Community □□□□

```
netstat -tlnp
```

□□□□□

```
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program
name
tcp        0      0 127.0.0.1:9981          0.0.0.0:*               LISTEN
22383/oxool
tcp6       0      0 :::9980                 :::*                    LISTEN
22383/oxool
```

□□□□□□□□□□□□□□□□□□□□ →□□□□□□□□□□□□ OxOOL □□□□ LXC □□□

OxOffice Online 主程式安裝說明

在 CentOS 7 環境下安裝 OxOffice Online v4 社群版

請先下載並安裝最新版本的 CentOS 7 (建議安裝 Minimal 版本)，安裝完成後，請執行以下指令來更新系統，安裝必要的軟體包與 OxOffice Online v4 社群版 (以下指令均以一般使用者身份執行，若使用 root 執行，則請自行去除 sudo)

```
sudo yum upgrade -y
sudo yum install vim net-tools unzip wget -y
cd /etc/yum.repos.d
sudo wget http://www.oxoffice.com.tw/rpm/el/oxool-community-v4-centos7.repo
sudo yum upgrade -y
sudo yum groupinstall "OxOOL Community Group" -y
```

停用 SELinux

```
sudo vim /etc/selinux/config # vim 可代換成任何您慣用的文字編輯器
# 尋找 SELinux=enforcing 此行並將 enforcing 改為 disabled 或 permissive 並存檔離開
```

設定防火牆並放行預設的 9980 port

```
sudo firewall-cmd --zone=public --permanent --add-service=http
sudo firewall-cmd --zone=public --permanent --add-port=9980/tcp
sudo firewall-cmd --reload
```

讓開機時預設啟動 OxOOL 並重新啟動

```
sudo systemctl enable oxoolwsd
sudo reboot
```

使用 netstat 確認 OxOOL 正常啟動

```
sudo netstat -tlnp
```

看到有類似以下的輸出就代表 OxOOL 正常啟動

```
tcp6       0      0 :::9980□□               :::*                    □LISTEN      644/oxool
```

□ Rocky Linux 8 □□□□ OxOffice Online v4 □□□

□□□□□□□□□□□□ Rocky Linux 8□□□□ Rocky Linux 8 □□□□
□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□ OxOffice Online v4
□□□□□□□□□□□□□□□□□□□□□□ root □□□□□□□□□ sudo□

```
sudo dnf config-manager --set-enabled powertools
sudo dnf upgrade -y
sudo dnf install epel-release -y
sudo dnf install vim net-tools unzip wget -y
cd /etc/yum.repos.d
sudo wget http://www.oxoffice.com.tw/rpm/el/oxool-community-v4-el8.repo
sudo dnf upgrade -y
sudo dnf groupinstall "OxOOL Community Group" -y
```

□□ SELinux

```
sudo vim /etc/selinux/config # vim □□□□□□□□□□□□□□□□□
# □□ SELinux=enforcing □□□□ enforcing □□ disabled □ permissive □□□□□
```

□□□□□□□□□□ 9980 port

```
sudo firewall-cmd --zone=public --permanent --add-service=http
sudo firewall-cmd --zone=public --permanent --add-port=9980/tcp
sudo firewall-cmd --reload
```

□□□□□□□ OxOOL □□□□

```
sudo systemctl enable oxoolwsd
sudo reboot
```

□□ netstat □□ OxOOL □□□

```
sudo netstat -tlnp
```

□□□□□□□□□□□□□□□ OxOOL □□□□□

```
tcp6       0      0 :::9980□□               :::*                    □LISTEN      644/oxool
```

OxOffice Online 主程式安裝說明

在 Ubuntu 20.04 環境下安裝 OxOffice Online v4 社群版

請先下載並安裝最新版本的 Ubuntu 20.04 (可參考 Ubuntu Server 20.04 基本安裝)，安裝完成後，請執行以下指令來更新系統，安裝必要的軟體包與 OxOffice Online v4 社群版

```
sudo apt update
sudo apt upgrade -y
sudo apt install vim openssh-server net-tools curl -y
# 如果有重新啟動服務的提示，請選擇 yes 即可
curl http://www.oxoffice.com.tw/deb/OSSII.key | sudo apt-key add
cd /etc/apt/sources.list.d/
sudo wget http://www.oxoffice.com.tw/deb/oxool-community-v4-focal.list
sudo apt update
sudo apt install oxool -y
```

讓開機時預設啟動 OxOOL 並重新啟動

```
sudo systemctl enable oxool
sudo reboot
```

使用 netstat 確認 OxOOL 正常啟動

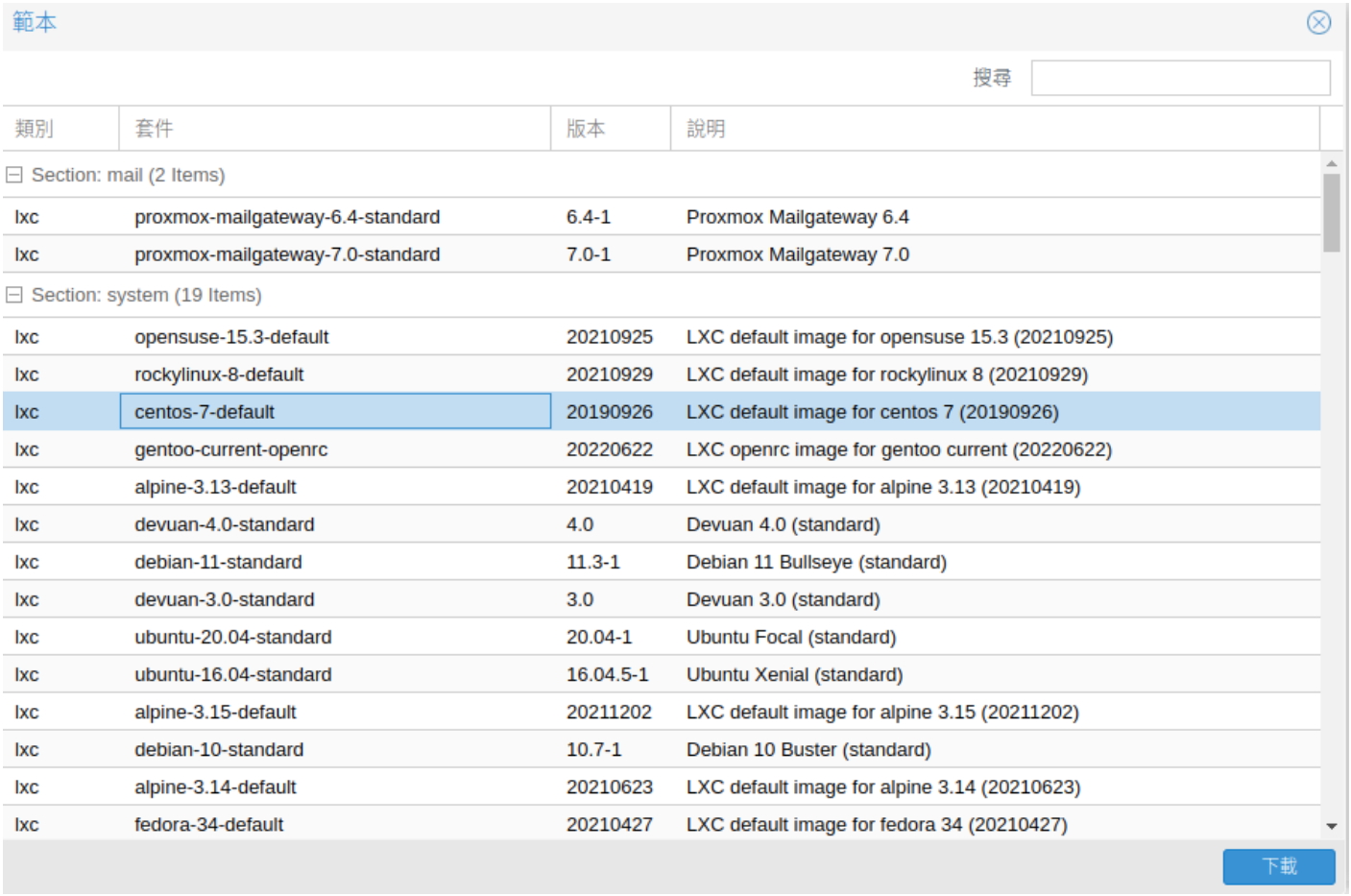
```
sudo netstat -tlnp
```

看到有類似以下的輸出就代表 OxOOL 正常啟動

```
tcp6       0      0 :::9980□□               :::*                    □LISTEN      644/oxool
```

□ PVE □ LXC □□ CentOS 7 □□□ OxOffice Online v4 □□

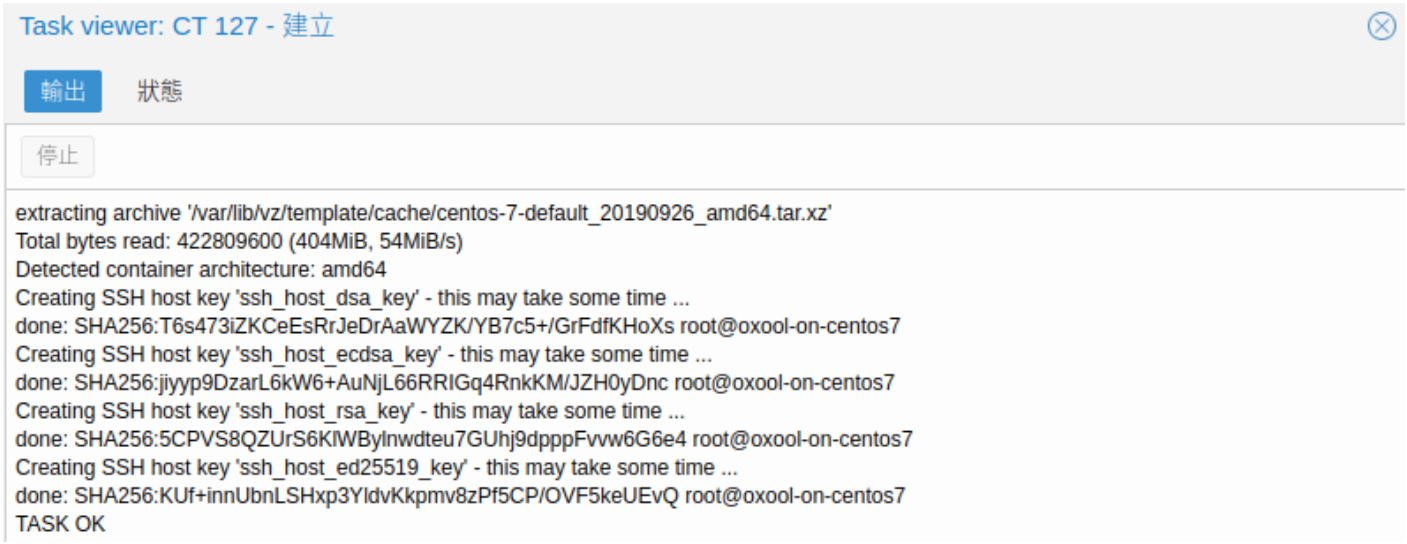
□□□ PVE □□ CT □□□□□□ CentOS 7 □□□□□□□□□□□ centos-7-default □□□□□□□□□□



□□□□□□□□□□□□□□□□□ CentOS 7 □□



點擊右上角的「建立 CT」建立新的 LXC
容器，可參考以下圖片進行設定，每個步驟完成後請按對話方塊右下角的「繼續」，到最後一步請按「完成」



□□□□□□□□□□□□□□ LXC □□□□ root
□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□

```
yum install vim openssh* net-tools unzip wget -y
cd /etc/yum.repos.d
wget http://www.oxoffice.com.tw/rpm/el/oxool-community-v4-centos7.repo
yum upgrade -y
yum groupinstall "OxOOL Community Group" -y
```

□□□□□□□□ OxOOL □□□□□

```
systemctl enable oxoolwsd
reboot
```

□□ netstat □□ OxOOL □□□□

```
netstat -tlnp
```

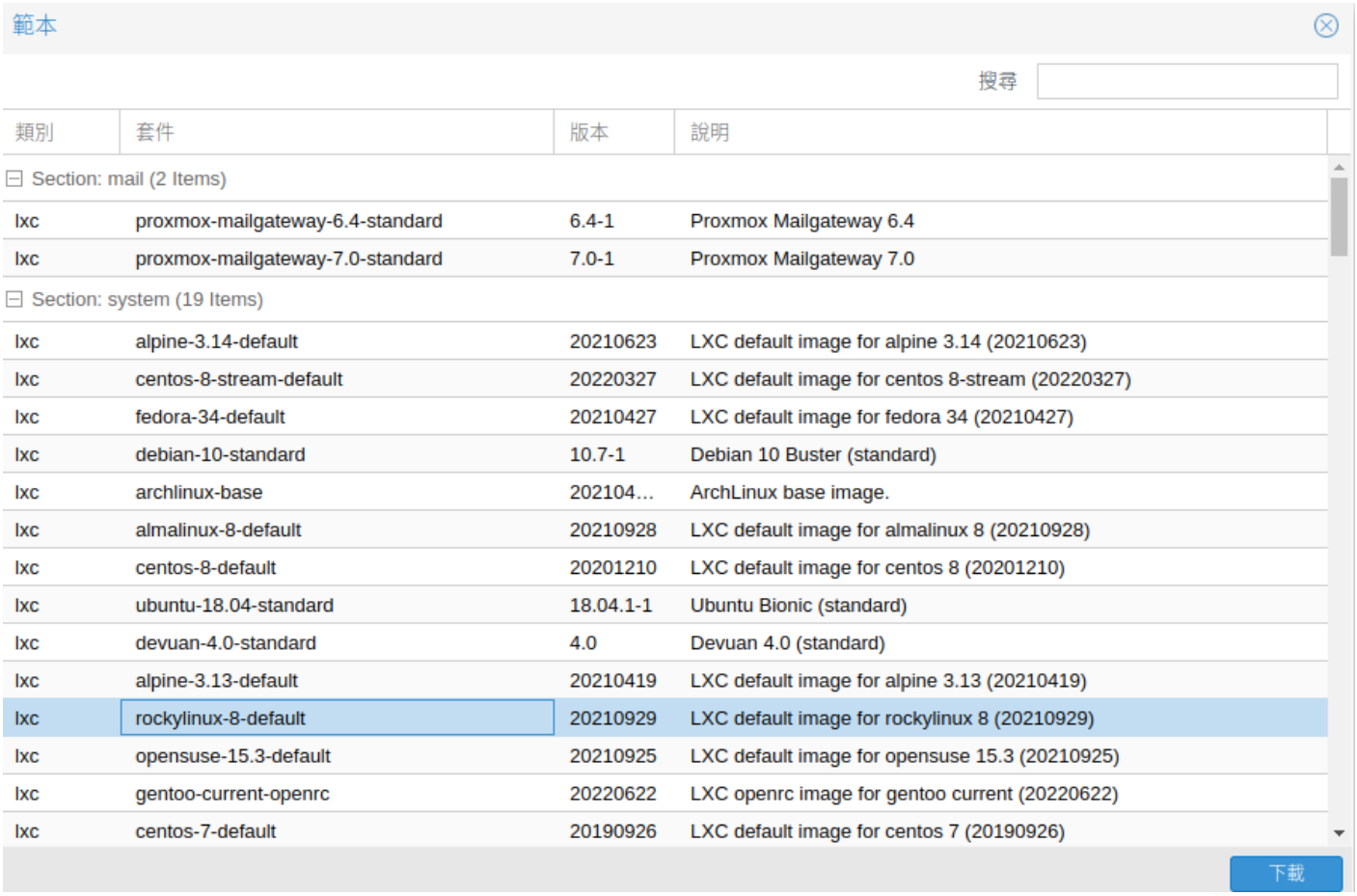
□□□□□□□□□□□□□□ OxOOL □□□□

```
tcp6       0      0 :::9980□□               :::*                    □LISTEN      644/oxool
```

□□□□□□□□□□□□□□□□□□□□ →□□□□□□□□□□□□□ OxOOL □□□□ LXC □□□

□ PVE □ LXC □□ Rocky Linux 8 □□□ OxOffice Online v4 □□□

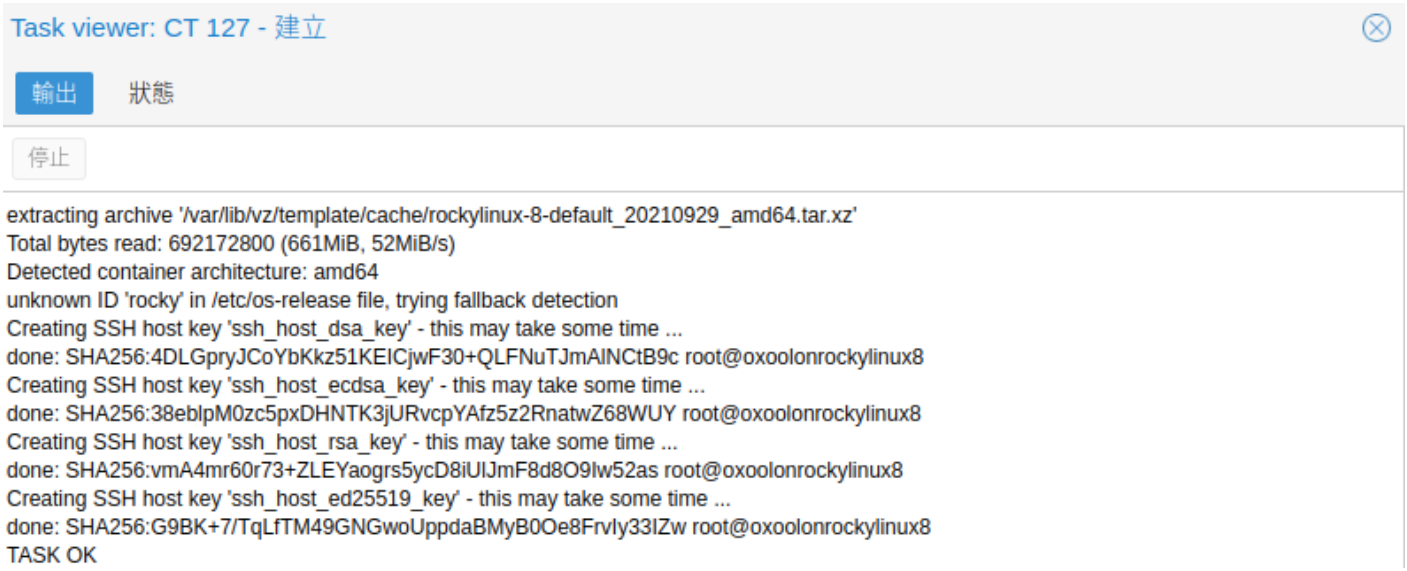
□□□□ PVE □□ CT □□□□□□□ Rocky Linux 8 □□□□□□□□□□□□ rockylinux-8-default □□□□□□□□□□□□



□□□□□□□□□□□□□□□□□ Rocky Linux 8 □□



點擊右上角的「建立 CT」建立新的 LXC
容器，可參考以下圖片進行設定，每個步驟完成後請按對話方塊右下角的「繼續」，到最後一步請按「完成」



□□□□□□□□□□□□□□ LXC □□□□ root
□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□

```
dnf config-manager --set-enabled powertools
dnf install vim openssh* net-tools unzip wget -y
cd /etc/yum.repos.d
wget http://www.oxoffice.com.tw/rpm/el/oxool-community-v4-el8.repo
dnf upgrade -y
dnf groupinstall "OxOOL Community Group" -y
```

□□□□□□□□ OxOOL □□□□□

```
systemctl enable oxoolwsd
reboot
```

□□ netstat □□ OxOOL □□□□

```
netstat -tlnp
```

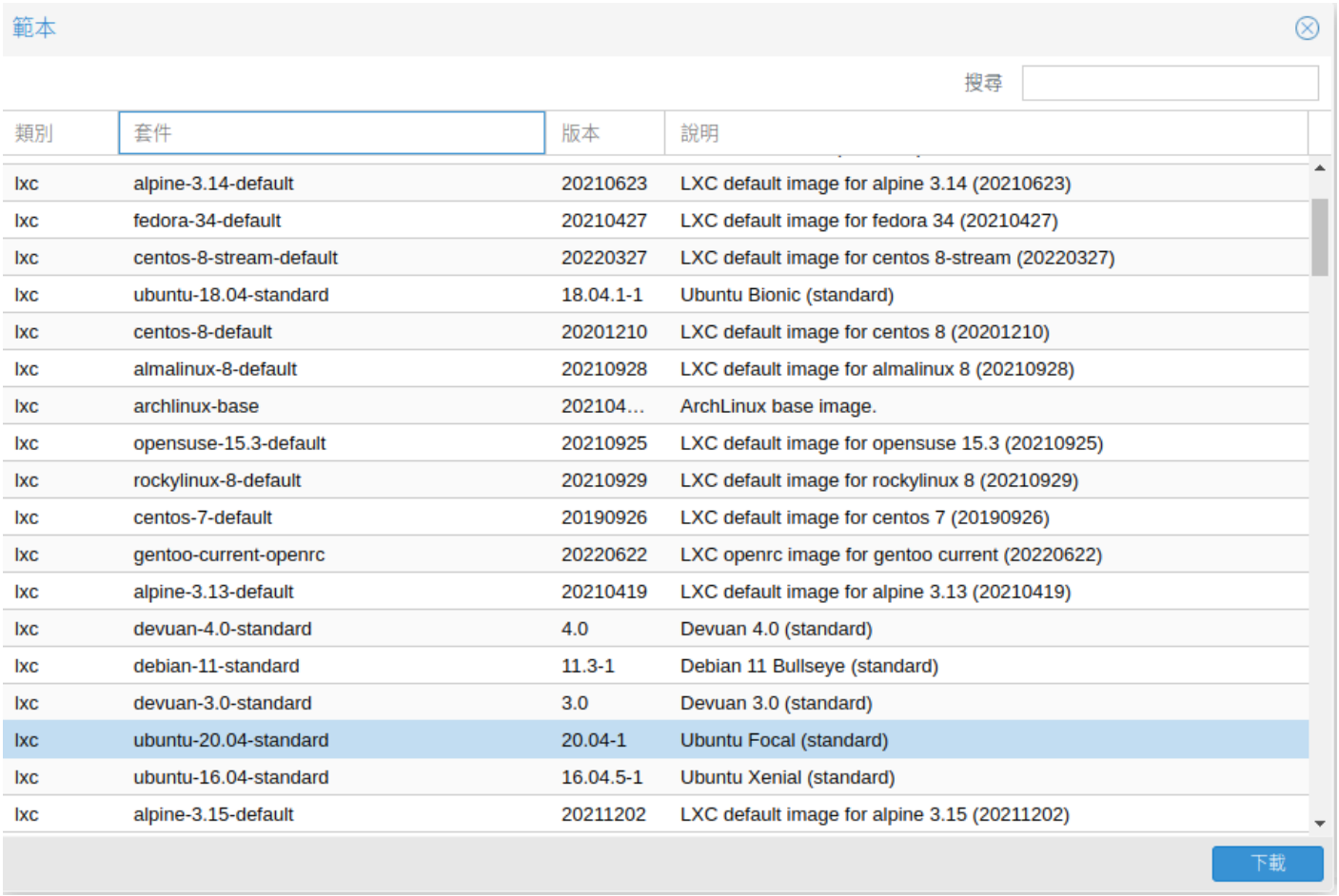
□□□□□□□□□□□□□□ OxOOL □□□□

```
tcp6       0      0      :::9980□□        :::*      □LISTEN     644/oxool
```

□□□□□□□□□□□□□□□□□□□□ →□□□□□□□□□□□□□ OxOOL □□□□ LXC □□□

□ PVE □ LXC □□ Ubuntu 20.04 □□□ OxOffice Online v4 □□□

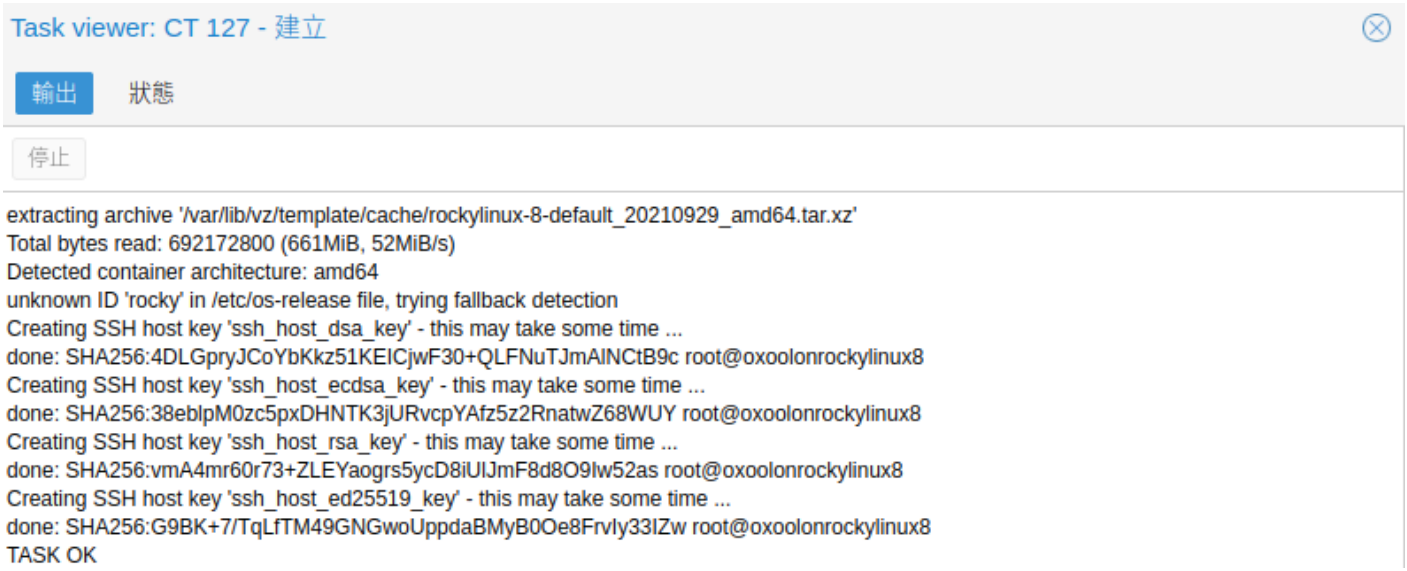
□□□□ PVE □□ CT □□□□□□□ Ubuntu 20.4 □□□□□□□□□□□□ ubuntu-20.04-standard □□□□□□□□□□□□



□□□□□□□□□□□□□□□□□ Ubuntu 20.04 □□



點擊右上角的「建立 CT」建立新的 LXC
容器，可參考以下圖片進行設定，每個步驟完成後請按對話方塊右下角的「繼續」，到最後一步請按「完成」



□□□□□□□□□□□□□□ LXC □□□□ root
□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□

```
apt update
apt upgrade -y
apt install vim openssh-server net-tools curl gnupg2 wget -y
# □□□□□□□□□□□□□□□□□□ yes □□
curl http://www.oxoffice.com.tw/deb/OSSII.key | sudo apt-key add
cd /etc/apt/sources.list.d/
wget http://www.oxoffice.com.tw/deb/oxool-community-v4-focal.list
apt update
apt install oxool -y
```

□□□□□□□□ OxOOL □□□□□

```
systemctl enable oxool
reboot
```

□□ netstat □□ OxOOL □□□□

```
netstat -tlnp
```

□□□□□□□□□□□□□ OxOOL □□□□

```
tcp6       0      0 :::9980□□               :::*                    □LISTEN      644/oxool
```

□□□□□□□□□□□□□□□□□□□ →□□□□□□□□□□□□ OxOOL □□□□ LXC □□□

□ Rocky Linux 8 □□□□ OxOffice Online v5 □□□

□□□□□□□□□□□□ Rocky Linux 8□□□□ Rocky Linux 8 □□□□
□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□ OxOffice Online v5
□□□□□□□□□□□□□□□□□□□□□□□ root □□□□□□□□□ sudo□

```
sudo dnf config-manager --set-enabled powertools
sudo dnf upgrade -y
sudo dnf install epel-release -y
sudo dnf install vim net-tools unzip wget -y
sudo wget -P /etc/yum.repos.d http://www.oxoffice.com.tw/rpm/el/oxool-community-v5-el8.repo
sudo dnf upgrade -y
sudo dnf groupinstall "OxOOL Community Group" -y
```

□□ SELinux

```
sudo vim /etc/selinux/config # vim □□□□□□□□□□□□□□□□□
# □□ SELinux=enforcing □□□□ enforcing □□ disabled □ permissive □□□□□
```

□□□□□□□□□□ 9980 port

```
sudo firewall-cmd --zone=public --permanent --add-service=http
sudo firewall-cmd --zone=public --permanent --add-port=9980/tcp
sudo firewall-cmd --reload
```

□□□□□□□□ OxOOL □□□□□

```
sudo systemctl enable oxoolwsd
sudo reboot
```

□□ netstat □□ OxOOL □□□□

```
sudo netstat -tlnp
```

□□□□□□□□□□□□□□□ OxOOL □□□□□

```
tcp6       0      0 :::9980□□               :::*                    □LISTEN      644/oxool
```

OxOffice Online 主程式安裝說明

在 Ubuntu 24.04 環境下安裝 OxOffice Online v5 社群版

請先下載並安裝最新版本的 Ubuntu 24.04，安裝完成後，請執行以下指令來更新系統，安裝必要的軟體包與 OxOffice Online v5 社群版

```
sudo apt update
sudo apt upgrade -y
sudo apt install vim openssh-server net-tools curl -y
# 如果有重新啟動服務的提示，請選擇 yes 即可
sudo curl -o /etc/apt/keyrings/OSSII.asc http://www.oxoffice.com.tw/deb/OSSII.key
sudo curl -o /etc/apt/sources.list.d/oxool-community-v5-noble.list
http://www.oxoffice.com.tw/deb/oxool-community-v5-noble.list
sudo apt update
sudo apt install oxool -y
```

讓開機時預設啟動 OxOOL 並重新啟動

```
sudo systemctl enable oxool
sudo reboot
```

使用 netstat 確認 OxOOL 正常啟動

```
sudo netstat -tlnp
```

看到有類似以下的輸出就代表 OxOOL 正常啟動

```
tcp6       0      0 :::9980□□               :::*                    □LISTEN      644/oxool
```

OxOffice Online □□□

□ OxOffice Online □□□□□□□□□

設定啟用 OxOOL SSL 憑證

若要設定 OxOOL 服務的主機運作在 SSL 協定上時，請確認已經有申請到正式的 hostname ，接下來透過以下設定方式啟動 SSL:

- 憑證申請的流程可參考「 SSL For Free 免費 SSL 憑證申請，使用 Let’s Encrypt 最簡單方法教學! 」這篇文章。
- 申請好了之後，請將相關的憑證上傳至 OxOOL 主機上，建議可將憑證檔案放 在 /etc/oxool/ssl 目錄下，申請完成後會有 3 個檔案。
  - ca_bundle.crt
  - certificate.crt
  - private.key
- 接下來可利用 vim 調整 /etc/oxool/oxoolwsd.xml 的內容，找到以下的段落，修改 <enable>、<cert_file_path>、<key_file_path>、<ca_file_path> 這 4 個參數的值及路徑 (實際上的路徑可依照自身的情況調整)。

```
<ssl desc="SSL settings">
    <enable type="bool" desc="Controls whether SSL encryption is enable (do not disable for production deployment). If default is false, must first be compiled with SSL support to enable." default="true">true</enable>
    <termination desc="Connection via proxy where oxoolwsd acts as working via https, but actually uses http." type="bool" default="true">false</termination>
    <cert_file_path desc="Path to the cert file" relative="false">/etc/oxool/ssl/certificate.crt</cert_file_path>
    <key_file_path desc="Path to the key file" relative="false">/etc/oxool/ssl/private.key</key_file_path>
    <ca_file_path desc="Path to the ca file" relative="false">/etc/oxool/ssl/ca_bundle.crt</ca_file_path>
    <cipher_list desc="List of OpenSSL ciphers to accept" default="ALL:!ADH:!LOW:!EXP:!MD5:@STRENGTH"></cipher_list>
    <hpkp desc="Enable HTTP Public key pinning" enable="false" report_only="false">
        <max_age desc="HPKP's max-age directive - time in seconds browser should remember the pins" enable="true">1000</max_age>
        <report_uri desc="HPKP's report-uri directive - pin validation failure are reported at this URL" enable="false"></report_uri>
        <pins desc="Base64 encoded SPKI fingerprints of keys to be pinned">
```

```
        <pin></pin>
        </pins>
    </hpkp>
</ssl>
```

- 存檔後，重新啟動 OxOOL 即可以 SSL 啟用 OxOOL。

```
# systemctl restart oxool
```

- 最後服務的位置為:

```
https://yourhostname:9980
```

- PS: 如果您是與 Nextcloud / ownCloud 系統介接，並以 SSL 的模式執行，但 OxOOL 無法正常執行時，請務必記得以下建議:

1. 在設定 richdocument 的位置時，建議使用 FQDN 而不是 IP。
2. 如果 Nextcloud / ownCloud 及 OxOOL 的 SSL 模式都設定正常但 OxOOL 仍無法正確開啟時，請將 ca_bundle.crt 的檔案 copy 到 Nextcloud 主程式目錄路行，並取代檔案 ca-bundle.crt (記得檔案名稱不要修改)，應該就可以正常開啟。

```
[nextcloud主程式目錄]/resources/config/ca-bundle.crt
```

設定讓其它的 Nextcloud 來存取 OxOOL

在撰寫說明文件的過程中, demo 環境都是設定在「 Private IP 環境」中, 若您的佈署的環境為以下二種情境者, 需額外進行進階的設定, 二種情境如下:

- 使用真實 IP / FQAN 對外開放服務 (例如: 117.56.68.133 / odf.nat.gov.tw)
- Nextcloud 主機和 OxOOL 主機是分開架設的

- 請進入 OxOOL 主機, 修改 /etc/oxool/oxoolwsd.xml 的內容, 找到以下這段:

```
<wopi desc="Allow/deny wopi storage. Mutually exclusive with webdav." allow="true">
<host desc="Regex pattern of hostname to allow or deny." allow="true">localhost</host>
<host desc="Regex pattern of hostname to allow or deny."
allow="true">10\.[0-9]{1,3}\.[0-9]{1,3}\.[0-9]{1,3}</host>
<host desc="Regex pattern of hostname to allow or deny."
allow="true">172\.1[6789]\.[0-9]{1,3}\.[0-9]{1,3}</host>
<host desc="Regex pattern of hostname to allow or deny." allow="true">172\.2[0-9]\.[0-
9]{1,3}\.[0-9]{1,3}</host>
<host desc="Regex pattern of hostname to allow or deny." allow="true">172\.3[01]\.[0-
9]{1,3}\.[0-9]{1,3}</host>
<host desc="Regex pattern of hostname to allow or deny." allow="true">192\.168\.[0-
9]{1,3}\.[0-9]{1,3}</host>
<host desc="Regex pattern of hostname to allow or deny." allow="true">117\.56\.68\.133</host>
<host desc="Regex pattern of hostname to allow or deny." allow="true">odf\.nat\.gov\.tw</host>
<host desc="Regex pattern of hostname to allow or deny." allow="false">192\.168\.1\.1</host>
<max_file_size desc="Maximum document size in bytes to load. 0 for unlimited."
type="uint">0</max_file_size>
</wopi>
```

- 請依格式加入以下二段:

```
<host desc="Regex pattern of hostname to allow or deny." allow="true">117\.56\.68\.133</host>
<host desc="Regex pattern of hostname to allow or deny." allow="true">odf\.nat\.gov\.tw</host>
```

- □□□□□□□ OxOOL

```
# systemctl restart oxool
```

OxOffice Online

社群版編譯說明

如果想編譯在 GitHub 上最新版本的 OxOffice Online 社群版本，請參考本文件說明該如何建立 OxOffice Online 社群版的編譯環境及編譯流程。

□ CentOS 7 □□□□ OxOOL

- □□□ □ CentOS 7 □□□□□ (□□□) □□□□□ CentOS 7 □□ OxOOL □□□□□□□□□□ SELinux □□□□□□□□□□□□□□□□□□□□□□□□□□□□□□ root □□□□□□□□□

- □□□□□□□□ OxOOL

```
sudo systemctl stop oxool
sudo systemctl disable oxool
```

- □□□□ OxOOL □□□□□

```
sudo yum install epel-release -y
sudo yum update -y
sudo yum install git yum-utils libpcap libcap-devel poco-devel ccache python36-polib python36-lxml -y
sudo yum install libtool gcc-c++ fontconfig-devel cppunit-devel pam-devel rpm-build -y
```

- □□□□□ Node.js

```
curl -fsSL https://rpm.nodesource.com/setup_16.x | sudo bash -
sudo yum install nodejs -y
```

- □ OxOOL □□□ clone □□

```
git clone https://github.com/OSSII/oxool-community.git
```

- □□□□ OxOOL □□

```
cd oxool-community
./autogen.sh
./configure
make dist
```

```
rpmbuild -tb oxool-x.x.x.tar.gz #在做完以上步驟後，clone 下來的目錄內應會有一個檔名類似 oxool-3.2.11.tar.gz 的壓縮檔，將後面的 oxool-x.x.x.tar.gz 代換為此檔的檔名即可
```

- 編譯完成的 rpm 會放在以下路徑

```
~/rpmbuild/RPMS/x86_64
```

套件直接就可以拿來安裝~編譯完成。

OxOffice Online 社群版編譯說明

在 Ubuntu 18.04 環境下編譯 OxOOL

- 請參考 在 Ubuntu 18.04 環境下安裝 (社群版) 這篇文章的 Ubuntu 18.04 安裝 OxOOL 套件的過程。
- 先關閉原本安裝的 OxOOL

```
sudo systemctl stop oxool
sudo systemctl disable oxool
```

- 安裝 git 指令

```
sudo apt install git -y
```

- 將 OxOOL 源始碼 clone 回來

```
git clone https://github.com/OSSII/oxool-community.git
```

- 安裝 Ubuntu 18.04 編譯環境所需的套件

```
cd oxool-community
./install_build_require_ubuntu.sh
```

- 安裝並設定 Node.js

```
curl -fsSL https://deb.nodesource.com/setup_16.x | sudo -E bash -
sudo apt install -y nodejs
```

- 編譯及測試執行 OxOOL

切換到 oxool-cummunity 程式的主目錄下，開始編譯程式。

```
cd oxool-community
./autogen.sh # 過程中若是提示找不到 fontconfig，請安裝 libfontconfig1-dev；若提示缺少 Poco 標頭，則請安裝 libpoco-dev
make -j2
```

如果成功就會看到以下訊息請您輸入編譯帳號的密碼:

```
[sudo] password for oxoffice:
```

輸入後，會繼續編譯直到完成，然後輸入以下指令測試是否能運行成功:

```
make run
```

如果看到以下訊息，就代表已經可以成功測試了。

```
Ready to accept connections on port 9980.
```

- 打包 OxOOL deb 套件

請在專案的根目錄執行

```
debuild -b -uc -us # 若您已先執行過上面的編譯動作，在執行此步驟前請先在您的 oxool-community 目錄執行 git clean -fdx
指令，如此才能正常打包
```

最後打包的 deb 結果會生成在專案的上層目錄

```
cd ..
ls oxool_3.x.x-1.community_amd64.deb
```

套件直接就可以拿來安裝~編譯完成。

在 CentOS 7 環境下編譯 OxOffice Online v4 社群版
請先下載並安裝最新版本的 CentOS 7
，然後執行以下指令更新系統與安裝必要的軟體包（此處是以一般使用者身份登入為準，且編譯時不建議使用
root 身份），其中 devtoolset-11 是用來安裝較新版本的 GCC 套裝軟體，數字為 GCC 的主要版本，請至少使用 GCC 8 以上的版本。

sudo yum upgrade -y
sudo yum install vim net-tools unzip wget -y
cd /etc/yum.repos.d
sudo wget http://www.oxoffice.com.tw/rpm/el/oxool-community-v4-centos7.repo
sudo yum install epel-release -y
sudo yum upgrade -y
sudo yum install git yum-utils libpcap libcap-devel libgit2-devel poco-devel ccache python36-devel python36-polib python36-lxml -y
sudo yum install libtool gcc-c++ fontconfig-devel cppunit-devel pam-devel rpm-build -y
sudo yum install oxoffice* poco* gumbo* -y
sudo yum install centos-release-scl
sudo yum install devtoolset-11

停用 SELinux

vim /etc/selinux/config # vim 可代換成任何您慣用的文字編輯器
尋找 SELinux=enforcing 此行並將 enforcing 改為 disabled 或 permissive 並存檔離開

安裝並設定 Node.js

curl -fsSL https://rpm.nodesource.com/setup_16.x | sudo bash -
sudo yum install nodejs -y

將 OxOffice Online v4 社群版的原始碼 clone 下來

git clone https://github.com/OSSII/oxool-community.git
cd oxool-community

```
git checkout v4
```

編譯 OxOffice Online v4 社群版（此處的 devtoolset-11 需根據上面選擇安裝的版本而定）

```
scl enable devtoolset-11 bash
./autogen.sh
./configure
make dist
rpmbuild -tb oxool-x.x.x.tar.gz
```

編譯完成的 RPM 軟體包會放在以下路徑

```
~/rpmbuild/RPMS/x86_64
```

此處的 RPM 軟體包即可用來安裝。

OxOffice Online 社群版編譯說明

在 Rocky Linux 8 環境下編譯 OxOffice Online v4 社群版

請先安裝 Rocky Linux 8 (可參照 Rocky Linux 8 基本安裝)，然後執行以下指令更新系統與安裝必要的軟體包 (此處是以一般使用者身份登入為準，且編譯時不建議使用 root 身份)

```
sudo dnf config-manager --set-enabled powertools
sudo dnf upgrade -y
sudo dnf install vim net-tools unzip wget -y
cd /etc/yum.repos.d
sudo wget http://www.oxoffice.com.tw/rpm/el/oxool-community-v4-el8.repo
sudo dnf install epel-release -y
sudo dnf upgrade -y
sudo dnf install git yum-utils libpcap libcap-devel libgit2-devel poco-devel ccache python3-devel python3-polib python3-lxml -y
sudo dnf install libtool gcc-c++ fontconfig-devel cppunit-devel pam-devel rpm-build -y
sudo dnf install oxoffice* poco* -y
```

停用 SELinux

```
vim /etc/selinux/config # vim 可代換成任何您慣用的文字編輯器
# 尋找 SELinux=enforcing 此行並將 enforcing 改為 disabled 或 permissive 並存檔離開
```

安裝並設定 Node.js

```
sudo dnf module enable nodejs:18 -y
sudo dnf install nodejs -y
```

將 OxOffice Online v4 社群版的原始碼 clone 下來

```
git clone https://github.com/OSSII/oxool-community.git
cd oxool-community
git switch v4
```

編譯 OxOffice Online v4 社群版

```
./autogen.sh
./configure
make dist
rpmbuild -tb oxool-x.x.x.tar.gz
```

編譯完成的 RPM 軟體包會放在以下路徑

```
~/rpmbuild/RPMS/x86_64
```

此處的 RPM 軟體包即可用來安裝。

□ Ubuntu 20.04 □□□□ OxOffice Online v4 □□□

□□□□ Ubuntu 20.04□□□□ Ubuntu Server 20.04 □□□□ □□□□□□□□□□□□□□□□□□□□

```
sudo apt update
sudo apt upgrade -y
sudo apt install vim openssh-server net-tools curl git -y
# □□□□□□□□□□□□□□□□□□ yes □□
curl http://www.oxoffice.com.tw/deb/OSSII.key | sudo apt-key add
cd /etc/apt/sources.list.d/
sudo wget http://www.oxoffice.com.tw/deb/oxool-community-v4-focal.list
sudo apt update
sudo apt install oxoffice* libpoco-dev
```

□□□□□□□□□□□□□

```
sudo apt update
sudo apt install -y build-essential libsqlite3-dev libcurl4-openssl-dev libcppunit-dev libcap-
dev libcap2-bin libgit2-dev libtool libpng-dev automake m4 wget curl autoconf pkg-config
openssl ccache fontconfig libfontconfig1-dev libpam-dev
sudo apt install -y devscripts debhelper dh-systemd dh-exec
sudo apt install -y libodbc1 libpcre16-3 libpcre3-dev libpcre32-3 libpcrecpp0v5
sudo apt install -y hunspell
sudo apt install -y python3-polib python3-lxml
```

□□□□□ Node.js

```
cd ~
curl -fsSL https://deb.nodesource.com/setup_20.x | sudo -E bash -
sudo apt install -y nodejs
```

□ OxOffice Online v4 □□□□□□□ clone □□

```
cd ~
git clone https://github.com/OSSII/oxool-community.git
cd oxool-community
git switch v4
```

編譯 OxOffice Online v4 社群版

```
debuild -b -uc -us
```

編譯好的 deb 軟體包會放在編譯用目錄的上層

```
cd ..
ls oxool_4.x.x-1.community_amd64.deb
```

此處的 DEB 軟體包即可用來安裝。

注意：若您在編譯過程中遇到 Node.js/npm 的記憶體不足問題，可在專案根目錄下依序執行以下指令：

```
cd loleaflet
npm install
cd archived-packages
npm install
cd ..
touch node_modules
```

如此即可避開此問題。

OxOffice Online 社群版編譯說明

在 Rocky Linux 9 環境下編譯 OxOffice Online v4 社群版

請先安裝 Rocky Linux 9，然後執行以下指令更新系統與安裝必要的軟體包（此處是以一般使用者身份登入為準，且編譯時不建議使用 root 身份）：

```
sudo dnf config-manager --set-enabled crb
sudo dnf upgrade -y
sudo dnf install vim net-tools unzip wget -y
sudo wget -O /etc/yum.repos.d/ooxml-community-v4-el8.repo
http://www.oxoffice.com.tw/rpm/el/oxool-community-v4-el8.repo
sudo dnf install epel-release -y
sudo dnf upgrade -y
sudo dnf install git yum-utils libpcap libcap-devel libgit2-devel poco-devel ccache python3-
devel python3-polib python3-lxml -y
sudo dnf install libtool gcc-c++ fontconfig-devel cppunit-devel pam-devel rpm-build -y
sudo dnf install oxoffice* poco* gumbo* -y
```

停用 SELinux

```
vim /etc/selinux/config # vim 可代換成任何您慣用的文字編輯器
# 尋找 SELinux=enforcing 此行並將 enforcing 改為 disabled 或 permissive 並存檔離開
```

安裝並設定 Node.js

```
sudo dnf module enable nodejs:18 -y
sudo dnf install nodejs -y
```

將 OxOffice Online v4 社群版的原始碼 clone 下來

```
git clone https://github.com/OSSII/oxool-community.git
cd oxool-community
git switch v4
```

編譯 OxOffice Online v4 社群版

```
./autogen.sh
./configure
make dist
rpmbuild -tb oxool-x.x.x.tar.gz
```

編譯完成的 RPM 軟體包會放在以下路徑

```
~/rpmbuild/RPMS/x86_64
```

此處的 RPM 軟體包即可用來安裝。

在 Rocky Linux 8 環境下編譯 OxOffice Online v5 社群版
請先安裝 Rocky Linux 8（可參照 Rocky Linux 8 基本安裝
），然後執行以下指令更新系統與安裝必要的軟體包（此處是以一般使用者身份登入為準，且編譯時不建議使
用 root 身份）

sudo dnf config-manager --set-enabled powertools
sudo dnf upgrade -y
sudo dnf install vim net-tools unzip wget -y
sudo wget -P /etc/yum.repos.d http://www.oxoffice.com.tw/rpm/el/oxool-community-v5-el8.repo sudo dnf install epel-release -y
sudo dnf upgrade -y
sudo dnf install make git yum-utils libpcap libcap-devel libgit2-devel libzstd-devel poco-devel ccache python3-devel python3-polib python3-lxml -y
sudo dnf install libtool gcc-c++ fontconfig-devel cppunit-devel pam-devel rpm-build -y
sudo dnf install oxoffice* poco* -y

停用 SELinux

vim /etc/selinux/config # vim 可代換成任何您慣用的文字編輯器
尋找 SELinux=enforcing 此行並將 enforcing 改為 disabled 或 permissive 並存檔離開

安裝並設定 Node.js

sudo dnf module enable nodejs:18 -y
sudo dnf install nodejs -y

將 OxOffice Online v5 社群版的原始碼 clone 下來

git clone https://github.com/OSSII/oxool-community.git
cd oxool-community
git switch v5

編譯 OxOffice Online v5 社群版

```
./autogen.sh
make dist
rpmbuild -tb oxool-x.x.x.tar.gz
```

□□□□□ RPM □□□□□□□□□□

```
~/rpmbuild/RPMS/x86_64
```

□□□ RPM □□□□□□□□□□

□ Ubuntu 24.04 □□□□ OxOffice Online v5 □□

□□□□ Ubuntu 24.04□□□□□□□□□□□□□□□□□□□□

```
sudo apt update
sudo apt upgrade -y
sudo apt install vim openssh-server net-tools curl git -y
# □□□□□□□□□□□□□□□□□□ yes □□
sudo curl -o /etc/apt/keyrings/OSSII.asc http://www.oxoffice.com.tw/deb/OSSII.key
sudo curl -o /etc/apt/sources.list.d/oxool-community-v5-noble.list
http://www.oxoffice.com.tw/deb/oxool-community-v5-noble.list
sudo apt update
sudo apt install oxoffice* libpoco-dev libgumbo-dev
```

□□□□□□□□□□□□□

```
sudo apt update
sudo apt install -y build-essential libsqlite3-dev libcurl4-openssl-dev libcppunit-dev libcap-
dev libcap2-bin libgit2-dev libtool libpng-dev automake m4 wget autoconf pkg-config openssl
libgumbo-dev ccache fontconfig libfontconfig1-dev libpam-dev
sudo apt install -y devscripts debhelper dh-exec
sudo apt install -y libpcre16-3 libpcre3-dev libpcre32-3 libpcrecpp0v5
sudo apt install -y hunspell
sudo apt install -y python3-polib python3-lxml
```

□□□□□ Node.js

```
cd ~
curl -fsSL https://deb.nodesource.com/setup_20.x -o nodesource_setup.sh
sudo -E bash nodesource_setup.sh
sudo apt install -y nodejs
```

□ OxOffice Online v5 □□□□□□□ clone □□

```
cd ~
git clone https://github.com/OSSII/oxool-community.git
cd oxool-community
git switch v5
```

編譯 OxOffice Online v5 社群版

```
./autogen.sh
debuild -b -uc -us
```

編譯好的 deb 軟體包會放在編譯用目錄的上層

```
cd ..
ls oxool*.deb
```

此處的 DEB 軟體包即可用來安裝。

注意：若您在編譯過程中遇到 Node.js/npm 的記憶體不足問題，可在專案根目錄下依序執行以下指令：

```
cd loleaflet
npm install
cd archived-packages
npm install
cd ..
touch node_modules
```

如此即可避開此問題。

OxOffice Online v4 □□□

□□□□□□□□□ OxOffice Online Module SDK □□ OxOffice Online □□

OxOOL v4 模組編譯手冊

一、環境準備

- 參考 技術手冊_____，先行編譯 oxool-community 以產生 oxool 開發相關套件。
  - 請留意: 此步驟編譯 oxool-community 時應使用 `autogen.sh`，不需再執行 `configure`。`autogen.sh` 預設會開啟 `--enable-debug` 並載入 ModuleTesting 模組供開發測試使用。
  - 透過 `rpmbuild` 或 `debuild` 產生 `oxool` 與 `oxool-dev` 安裝套件。
- 將產生出來之 oxool-dev 套件安裝於系統內。 oxool-dev 內含編譯模組所需之標頭檔、模組範本與 `oxool-module-maker` 程式。

二、使用 oxool-module-maker 產生模組基本檔案

- 執行 oxool-module-maker 以產生新模組的基本檔案。參數說明如下:
  - `--module-name=name`：模組名稱。必須符合正規表示式: `^[\w\@#]+$`，也就是所有的大小寫英文字母、數字 0-9、底線以及 # @ 三種符號的組合。注意減號 '-' 不能使用，而底線雖然可以使用但不建議。
  - `--serviceURI=serviceURI`：此模組所使用的網址路徑。若結尾有 / 表示此模組會處理一系列指令。例如 "/oxool/user/" 表示此模組可能會處理 /oxool/user/add 、 /oxool/user/del 、 /oxool/user/update 等一系列動作。若結尾沒有 / 則為固定的網址路徑。
  - `--version=version`：指定版號。預設為 "0.0.1"。
  - `--summary=summary`：指定 Summary, 用於產生出來的 rpm 檔。
  - `--description=Description`：指定 Description, 用於產生出來的 deb 檔。
  - `--author=author`：模組作者。預設會去找使用者全域 git 設定中的 user.name 與 user.email 兩個欄位。
  - `--license=license`：所使用的授權條款。預設為 MPLv2.0。
  - `--adminPrivilege=true/false`：此模組的網址路徑是否需要 admin 權限。預設為 false。
  - `--adminIcon=icon`：在後台管理介面中所使用的圖示（請參考 getbootstrap.com 。預設值為 "bug-fill"。
  - `--adminItem=text`：在後台管理介面中所使用的文字。
  - `--template-path=path`：
    指定模組範本路徑，也就是產生新模組時要從哪裡複製檔案過來。預設會從

/usr/share/oxool-devel/module-template 中複製模組檔案。
- - output-path=path :
指定產生的模組要放在哪個路徑。預設為使用者的家目錄。要留意不能用 ~
符號。最好使用完整路徑。
執行範例：

oxool-module-maker --module-name="samplemod" --serviceURI="/oxool/samplemod/" --summary="A sample moudle service on OxOOL" --description="A sample module service on OxOOL" --output-path="/home/oxool/git"

完成後在指定的 output-path 中會產生一個 git repository。裡面的內容是 template-path
參數指定的路徑裡的內容。

三、編譯與測試模組
用 autogen.sh 產生 configure 檔，然後用 configure
去確認該有的標頭檔、函式庫等是否都存在，並產生 Makefile 檔。
用 make 編譯。
測試模組前，先在 oxool-community 中執行 make run 啟動 oxool 服務。
啟動後，在模組目錄中執行 ./test.sh <模組的 XML 檔>。 test.sh 會呼叫 oxool 服務中的 ModuleTesting 模組，並將 XML 路徑傳送過去。 ModuleTesting 模組則會通知 oxool 載入模組的 . so 檔（模組編譯完成後可以在 .libs/ 目錄下找到此模組的 .so 檔）。此時就可以透過
serviceURI 指定的網址路徑執行模組。例如在瀏覽器中輸入：
http://127.0.0.1:9980/oxool/samplemod/。
模組修改後需重新編譯並重新執行 test.sh 以便把新的 .so 檔載入 oxool 服務中。

OxOOL 模組開發手冊
本手冊將簡單介紹 OxOOL 模組結構與開發說明。

OxOOL 預設模組檔案結構
這裡就以使用 oxool-module-maker 並以 oxool-dev
套件中預設之模組範本產生之新模組為例，重要的檔案與目錄重點說明如下：

ModuleConfiguration.md : 模組配置檔說明。
module.xml.in : 模組配置檔，用來在執行 configure 時產生模組 XML 檔。如果在 oxool-module-maker 所下的參數有誤或需要修改、需要調整 XML 檔中的內容或新增標籤等，則可根據
ModuleConfiguration.md 中的說明修改此檔並重新執行 autogen.sh 與 configure。
module.spec.in : 用來產生 rpm 軟體包的 spec 檔。
debian/* : 用來產生 deb 軟體包的檔案。
admin/* : 如果模組有後端管理介面，則放在此處。管理介面裡顯示的名稱定義在 module.xml.in 中的 <adminItem> 標籤，或是在執行 oxool-module-maker 時指定的 --adminItem 參數。可參考 README.md 中的說明。
src/* : 模組的 C++ 源碼檔，會編譯成 .so 檔並在 OxOOL 執行時載入。
html/* : 模組前端網頁介面。模組可以不必用 C++ 實作，而是直接透過網頁進行。預設的頁面為
index.html。
test.sh : 模組開發期間測試時，只要執行此檔，並將模組的 XML 檔傳給 OxOOL
即可進行測試。可參考 OxOOL模組編譯手冊中的說明。

OxOOL 模組中 C++ 的類別與方法
oxool-module-maker 產生之新模組，預設在 src 下會有 Module.cpp
檔，裡面定義此模組的基礎類別，以模組名稱為類別名稱。裡面除了建構子與解構子之外，定義了幾個方法：

getVersion() : 取得版本號。
initialize() : 初始化模組。
handleRequest() : 處理來自前端 Client 對模組的請求。根據模組需求去實作每個請求。 handleAdminRequest() : 處理來自後端管理介面對模組的請求。根據模組需求去實作每個請求。
handleAdminMessage() : 處理來自後端管理介面對模組傳送的 websocket 訊息。

OxOffice Online 設定檔說明
OxOOL 設定檔位於 /etc/oxool/oxoolwsd.xml以下介紹設定檔內容代表之意義與其可設定的值

<allowed_languages>
預設：de_DE en_GB en_US es_ES fr_FR it nl pt_BR pt_PT ru
說明：設定支援的拼字檢查器、語法檢查器、同義詞庫、連字號的語系，注意，若啟用太多會影響到啟動的效能。
</allowed_languages>

<sys_template_path>
預設：systemplate（系統編譯時已決定的變數）
說明：放置 LibreOfficeKit 呼叫使用的 LibreOffice 動態函式庫所需的執行時期環境，用以做為子行程 chroot jails 環境來源範本。 </sys_template_path>

<child_root_path>
預設：child_root_path（系統編譯時已決定的變數）
說明：用來建立子處理程序 chroot jails 環境的目錄路徑，應與 systemplate 在同一個檔案系統（分割區）上。 </child_root_path>

<mount_jail_tree>
預設：true
說明：控制是否要掛載 systemplate 與 lotemplate，這會比連結或複製每個檔案來的快。
</mount_jail_tree>

<server_name>
預設：空
說明：執行 oxoolwsd 主程式的主機名稱和連接埠號，如果設定為空，那麼會取決於連線需求。但 OxOOL 主機若在反向代理主機後，或是主機名稱無法直接可達的話則必須設定。 </server_name>

<client_port_number>
預設：9980
說明：OxOOL 要監聽的連接埠號。
</client_port_number>

<file_server_root_path>
預設：loleaflet/../
說明：指向檔案伺服器的 root 路徑，應包含 loleaflet 的目錄。

</file_server_root_path>

<hexify_embedded_urls>
預設：false
說明：保護編碼的 URL 不被中間的節點解碼。在 Azure 部署中較有用。 </hexify_embedded_urls>

<convert_to>
預設：false
說明：是否啟用檔案格式轉換。
</convert_to>

<memproportion>
預設：80.0
說明：設定 OxOffice 最大可消耗的系統記憶體百分比，達到後便會開始清理空閒的文件。 </memproportion>

<num_prespawn_children>
預設：1
說明：預設一開機就啟動，用以等待新連線的子處理程序數量。 </num_prespawn_children>

<per_document>
說明：關於文件的設定，包含 OxOffice 的核心設定。

<max_concurrency>
預設：8
說明：處理單一文件時，所使用的最大執行緒數量。
</max_concurrency>

<batch_priority>
預設：5
說明：針對批次處理程序（如 convert-to）降低優先程度，以避免搶走互動式處理程序的資源。 </batch_priority>

<document_signing_url>
預設：無
說明：簽署文件伺服器的端點 URL，若留空則停用簽署文件功能。 </document_signing_url>

<redlining_as_comments>
預設：true
說明：若設定為 true，則會在註解中顯示紅線。
</redlining_as_comments>

<idle_timeout_secs>
預設：3600
說明：設定閒置多久後連線中斷，單位為秒，預設為1小時。
</idle_timeout_secs>

<idlesave_duration_secs>
預設：1800
說明：設定若文件每經修改並閒置幾秒後就自動儲存一次檔案，預設為30秒，設定為0或負值即為停用 。 </idlesave_duration_secs>

<autosave_duration_secs>
預設：1800
說明：設定若文件每經修改後，經幾秒後就自動儲存一次檔案，預設為300秒 ，設定為0或負值即為停用。 </autosave_duration_secs>

<always_save_on_exit>
預設：false
說明：設定不管文件有沒有修改過，一旦關閉文件的編輯畫面就會強制儲存，預設為 false 。 </always_save_on_exit>

<limit_virt_mem_kb>
預設：0
說明：設定每個處理文件的行程可使用的最大虛擬記憶體，0 為未限制，至少要有 1700kb。 </limit_virt_mem_kb>

<limit_stack_mem_kb>
預設：8000
說明：設定每個處理文件的行程可使用的最大stack大小，0 為未限制。 </limit_stack_mem_kb>

<limit_file_size_mb>
預設：0
說明：設定每個處理文件的行程可開啟的最大檔案大小，0 為未限制。 </limit_file_size_mb>

<limit_num_open_files>
預設：0
說明：設定每個處理文件的行程可開啟的最多檔案數量，0 為未限制。 </limit_num_open_files>

<limit_load_secs>
預設：100
說明：設定開啟每個檔案的最大等待秒數，0 為未限制。
</limit_load_secs>

<limit_convert_secs>
預設：100
說明：設定轉換每個檔案的最大等待秒數，0 為未限制。
</limit_convert_secs>

<cleanup>
預設：false
說明：檢查異常消耗資源的文件並砍除對應的 LibreOfficeKit 處理程序，由以下的幾個選項決定其閾值。 <cleanup_interval_ms>
預設：10000
說明：兩次檢查間的間隔（單位為毫秒）。
</cleanup_interval_ms>
<bad_behavior_period_secs>
預設：60
說明：文件被認定是消耗資源並被砍除對應 LibreOfficeKit 處理程序的最小時間間隔。若此間隔內被認定為異常消耗資源的文件不符合條件則會重設間隔（單位為秒）。 </bad_behavior_period_secs>
<idle_time_secs>
預設：300
說明：文件被認定是異常消耗資源狀態候選的最小閒置時間（單位為秒）。
</idle_time_secs>
<limit_dirty_mem_mb>
預設：3072
說明：文件被認定是異常消耗資源狀態候選的最小記憶體使用量（單位為 MB）。 </limit_dirty_mem_mb>
<limit_cpu_per>
預設：85
說明：文件被認定是異常消耗資源狀態候選的最小 CPU 使用量（單位為百分比）。 </limit_cpu_per>
</cleanup>

</per_document>

<per_view>
說明：與開啟文件瀏覽文件相關的設定。
<out_of_focus_timeout_secs>
預設：60
說明：設定當在瀏覽器分頁在失去焦點後，最多在幾秒後就停止更新文件及調暗視窗。 </out_of_focus_timeout_secs>

<idle_timeout_secs>
預設：900
說明：設定當在使用者完全沒有動作時（即便目前焦點在該瀏覽器分頁上），最多在幾秒後就停止更新文件及調暗視窗。 </idle_timeout_secs>
</per_view>

<loleaflet_html>
預設：loleaflet.html
說明：OxOffice Online 的主網頁檔案名稱，請注意，變更完這個設定後，同時也要將以下目錄的網頁名稱變更為與設定檔一致： /usr/share/oxool/loleaflet/dist。
</loleaflet_html>

<logging>
說明：關於記錄檔的設定參數。
<color>
預設：true
說明：是否啟用記錄檔的顏色機制。
</color>

<level>
預設：warning
說明：記錄檔的level設定值，可以設定0-8個等級，或是可以設定為：none（關閉記錄），fatal，critical，error，warning，notice，information， debug，trace。
</level>

<protocol>
預設：false
說明：啟動時就開始記錄最小化的客戶端 JavaScript 協定紀錄（編譯時決定是否啟用）。
</protocol>

<lokit_sal_log>

預設：-INFO-WARN
說明：微調來自 LibreOfficeKit 的紀錄檔訊息。預設會停用來自 LibreOfficeKit 的紀錄檔訊息。 </lokit_sal_log>

<file>
說明：關於記錄檔的相關參數設定。
<property name=“path“>
預設：/var/log/oxoolwsd.log
說明：OxOffice Online 的記錄檔路徑。
<property name=“rotation“>
預設：daily
說明：設定檔案 rotation 機制的策略，可參考 Poco 的 FileChannel 機制。
<property name=“archive“>
預設：timestamp
說明：設定要在封存的紀錄檔名稱上附加時間戳 (timestamp) 或編號 (number)。
<property name=“times“>
預設：local
說明：指定要將紀錄檔 rotation 的時間視為本機時間 (local) 或 UTC 時間。有效的值為 local 與 utc。 <property name=“compress“>
預設：true
說明：設定是否要啟動紀錄檔壓縮功能。
<property name=“purgeAge“>
預設：30 days
說明：紀錄檔最多保存天數，可參考 Poco 的 FileChannel 機制。
<property name=“purgeCount“>
預設：100
說明：紀錄檔最大的保存數，設為 none 以停用清除紀錄檔的功能，可參考 Poco 的 FileChannel 機制。
<property name=“rotateOnOpen“>
預設：true
說明：設定是否在開啟時啟動紀錄檔 rotation 的機制。
<property name=“flush“>
預設：false
說明：設定是否要在記錄每行 log 時啟用 flush 機制，啟用時可能會影響效能，若設定不啟用時，來自不同處理程序的 log 可能不會依照時間順序記錄。 </file>
<anonymize>
<anonymize_user_data>
預設：false
說明：設定是否啟用對紀錄檔中的使用者資料進行匿名／模糊處理，預設為 false。
</anonymize_user_data>

<anonymization_salt>
預設：82589933
說明：設定對紀錄檔中的使用者資料進行匿名／模糊處理時要使用的 salt。使用 64 位元隨機數。 </anonymization_salt>
</anonymize>

</logging>

<loleaflet_logging>
預設：false
說明：設定是否將紀錄檔同時寫入瀏覽器的主控台。
</loleaflet_logging>

<trace_event>
預設：false
說明：控制是否開啟 Chrome Trace Event 檔案的產生流程。 <path>
預設：空
說明：輸出檔案的路徑設定。
</path>
</trace_event>

<trace>

預設：false
說明：將指令與通知紀錄下來。
<path>
預設：空
說明：輸出檔案的路徑設定。
</path>

<filter>
<message>
預設：空
說明：設定排除特定記錄以節省空間的正規表示式規則。
</message>
</filter>

<outgoing>
<record>
預設：false

說明：設定是否要記錄輸出的訊息。
</record>
</outgoing>
</trace>

<net>
說明：網路相關設定。
<proto>
預設：all
說明：設定開放使用 IPv4 或是 IPv6 或是全部的網路協定。 </proto>

<listen>
預設：any
說明：設定主程式可綁定的位置，可設為 any 或 loopback。 </listen>

<service_root>
預設：無
說明：設定所有頁面、websocket 等路徑的前綴字串。 </service_root>

<proxy_prefix>
預設：false
說明：設定是否啟用傳遞給重新導向請求的 ProxyPrefix。 </proxy_prefix>

<post_allow>
預設：true
說明：設定允許／拒絕開放 POST(REST) 的 IP 來源。 <host>
預設：
192\.168\.[0-9]{1,3}\.[0-9]{1,3} ::ffff:192\.168\.[0-9]{1,3}\.[0-9]{1,3} 127\.0\.0\.1
::ffff:127\.0\.0\.1
::1
172\.17\.[0-9]{1,3}\.[0-9]{1,3} ::ffff:172\.17\.[0-9]{1,3}\.[0-9]{1,3} </host>

</post_allow>

<frame_ancestors>
預設：無
說明：指定誰可以嵌入 OxOffice Online 的 iframe（oxoolwsd 與 WOPI host 總是允許），以空格分隔多個主機。 </frame_ancestors>

<connection_timeout_secs>
預設：30
說明：指定由 oxoolwsd 發起（如 WOPI 連線）的連線、傳送、接收等動作的逾時秒數。 </connection_timeout_secs>
</net>

<ssl>
說明：OxOffice Online 的 SSL 設定。
<enable>
預設：false
說明：設定是否要啟動 SSL 機制。
</enable>

<termination>
預設：true
說明：如果是透過代理伺服器設定連線，則此選項須設定為 true。 </termination>

<cert_file_path>
預設：/etc/oxool/cert.pem
說明：設定憑證檔的路徑，若是使用代理伺服器則不需變更，可使用代理伺服器的設定。 </cert_file_path>

<key_file_path>
預設：/etc/oxool/key.pem
說明：設定憑證金鑰的路徑，若是使用代理伺服器則不需變更，可使用代理伺服器的設定。 </key_file_path>

<ca_file_path>
預設：/etc/oxool/ca-chain.cert.pem
說明：設定憑證 ca 檔的路徑，若是使用代理伺服器則不需變更，可使用代理伺服器的設定。 </ca_file_path>

<cipher_list>
預設：ALL:!ADH:!LOW:!EXP:!MD5:@STRENGTH 說明：設定可接受的 OpenSSL 加密演算法清單。
</cipher_list>

<hpkp>
說明：啟用 HPKP。
<max_age>
預設：1000
說明：HPKP max-age 指令，以秒為單位，設定瀏覽器應該保留 pins 的時間。 </max_age>
<report_uri>
預設：空
說明：HPKP report-uri 指令，pin 的驗證錯誤回報 URL。 </report_uri>
<pins>
說明：要 pin 的金鑰的 Base64 編碼 SPKI 指紋。 <pin></pin>
</pins>
</hpkp>
</ssl>

<security>
說明：【警告】更改這些預設值可能會讓系統面臨重大風險。
<seccomp>
預設：true
說明：設定使用 seccomp 系統呼叫 filtering。 </seccomp>

<capabilities>
預設：true
說明：設定使用 chroot jails 的機制。
</capabilities>

<jwt_expiry_sec>
預設：1800
說明：管理介面的 JWT 權杖過期時間（以秒為單位）。
</jwt_expiry_sec>

<enable_macros_execution>

預設：false
說明：指定是否要執行巨集。這會啟用 Basic、Beanshell、JavaScript 與 Python 指令稿。若設定為 false，則下方的 macro_security_level 設定會被忽略。若設定為 true，則前述設定會用來指定巨集的安全層級。
</enable_macros_execution>

<macro_security_level>
預設：1
說明：巨集的安全層級。1（中等）必須在執行從未受信任的來源執行巨集前先進行確認。0（低，不建議）是所有巨集都可不經確認即執行。
</macro_security_level>

<enable_metrics_unauthenticated>
預設：false
說明：啟用時，/oxool/getMetrics 即不需要驗證。 </enable_metrics_unauthenticated>
</security>

<watermark>
說明：浮水印相關設定。
<editing>
預設：false
說明：若啟用，則會在編輯文件時顯示浮水印。
</editing>

<printing>
預設：false
說明：若啟用，則會在列印文件或將其匯出為 PDF 檔案時顯示浮水印。 </printing>

<opacity>
預設：0.2
說明：畫面上浮水印的不透明度，值可從 0.0 到 1.0。 </opacity>

<angle>
預設：45
說明：浮水印角度。
</angle>

預設：無

說明：字型名稱。

<color>
預設：#000000
說明：浮水印文字色彩。
</color>

<bold>
預設：false
說明：浮水印文字是否要為粗體。
</bold>

<italic>
預設：false
說明：浮水印文字是否要為義式斜體。
</italic>

<outline>
預設：false
說明：浮水印文字是否要變為中空。
</outline>

<shadow>
預設：false
說明：浮水印文字是否要帶有陰影。
</shadow>

<text>
預設：無
說明：輸入時要在畫面上顯示的浮水印文字。
</text>
</watermark>

<welcome>
說明：歡迎畫面的相關設定。
<enable>
預設：編譯時決定的參數
說明：控制是否要在新安裝及更新時對使用者顯示歡迎畫面。
</enable>

<enable_button>
預設：編譯時決定的參數
說明：決定歡迎畫面是否應該要有解釋性的按鈕而非關閉按鈕。 </enable_button>

<path>
預設：loleaflet/welcome
說明：歡迎畫面網頁的路徑。
</path>
</welcome>

<user_interface>
說明：編輯器使用者介面的相關設定。

<pdf_viewer_resolution_dpi>
預設：600
說明：檢視程式在繪製 PDF 內容時要使用的 DPI。
</pdf_viewer_resolution_dpi>

<use_integration_theme>
預設：false
說明：是否要使用來自 Nextcloud Office 等整合程式的佈景主題。 </use_integration_theme>
</user_interface>

<storage>
說明：不同後端儲存空間的連線設定。
<filesystem>
預設：false
說明：是否啟動讓 OxOffice Online 直接編輯本機檔案的機制，建議在正式佈署時確定更改為 false。

<wopi>
預設：true
說明：允許／拒絕 wopi storage，與 webdav 互斥。
<host>
預設：
true:
localhost
127\.0\.0\.1

10\.[0-9]{1,3}\.[0-9]{1,3}\.[0-9]{1,3} 172\.1[6789]\.[0-9]{1,3}\.[0-9]{1,3}
172\.2[0-9]\.[0-9]{1,3}\.[0-9]{1,3}
172\.3[01]\.[0-9]{1,3}\.[0-9]{1,3}
192\.168\.[0-9]{1,3}\.[0-9]{1,3}
false:
192\.168\.1\.1

說明：設定 wopi host 的主機名稱或是 IP 位置，如果主機名稱或 IP 中包含了「.」，則需加入「\」跳脫字元，例如：原本是 www.ossii.com.tw，那麼在 wopi host 內需設定成為「www\.ossii\.com\.tw」。
</host>

<max_file_size>
預設：0
說明：可載入的最大文件大小（以位元組為單位），0 代表無限制。
</max_file_size>

<reuse_cookies>
預設：false
說明：若啟用，則會捕捉來自瀏覽器的 cookies，並在 WOPI 請求中設定。 </reuse_cookies>

<locking>
說明：鎖定相關設定。
<refresh>
說明：設定應多常重新取得儲存伺服器的鎖，以秒為單位（預設為900秒，亦即15分鐘），設為 0 則不更新。 </refresh>
</locking>
</wopi>

<webdav>
預設：false
說明：允許／拒絕 webdav storage，與 wopi 互斥。
<host>
預設：false: localhost
說明：設定 webdav host的主機名稱或是 IP 位置，如果主機名稱或 IP 中間包含了「.」，則需加入「\」跳脫字元，例如：原本是 www.ossii.com.tw，那麼在 webdav host 內需設定成為「www\.ossii\.com\.tw」。
</host>
</webdav>
<ssl>

說明：OxOffice Online 的 SSL 設定。
<as_scheme>
預設：true
說明：設定後，將會啟用專門針對 WOPI URI 結構的 SSL 來供儲存空間使用。 </as_scheme>
<enable>
預設：false
說明：設定是否要啟動 SSL 機制。
</enable>

<cert_file_path>
預設：/etc/oxool/cert.pem
說明：設定憑證檔的路徑，若是使用代理伺服器則不需變更，可使用代理伺服器的設定。 </cert_file_path>

<key_file_path>
預設：/etc/oxool/key.pem
說明：設定憑證金鑰的路徑，若是使用代理伺服器則不需變更，可使用代理伺服器的設定。 </key_file_path>

<ca_file_path>
預設：/etc/oxool/ca-chain.cert.pem
說明：設定憑證 ca 檔的路徑，若是使用代理伺服器則不需變更，可使用代理伺服器的設定。 </ca_file_path>

<cipher_list>
預設：ALL:!ADH:!LOW:!EXP:!MD5:@STRENGTH
說明：設定可接受的 OpenSSL 加密演算法清單。
</cipher_list>
</ssl>
</storage>

<tile_cache_persistent>
預設：true
說明：設定是否在相同文件的不同編輯 session 保留 tile cache。 </tile_cache_persistent>

<admin_console>
說明：OxOffice Online 網頁端管理介面設定參數，預設登入的連線位置為： http(s)://youroxoolip:9980/loleaflet/dist/admin/admin.html。

<enable>

預設: true

說明: 啟用管理功能。

</enalbe>

<enable_pam>

預設: false

說明: 啟用 PAM 使用者認證機制。

</enalbe_pam>

<username>

預設: admin

說明: 設定管理介面的登入帳號，若啟用 PAM 機制則會忽略此設定。

</username>

<password>

預設: @administrator1

說明: 設定管理介面的登入密碼。在多數平台上已被棄用，建議使用 PAM 或 oxoolconfig 設定安全的密碼。

</password>

</admin_console>

<monitors>

預設: 空

說明: 設定連線的伺服器位置。

</monitors>

OxOffice Online

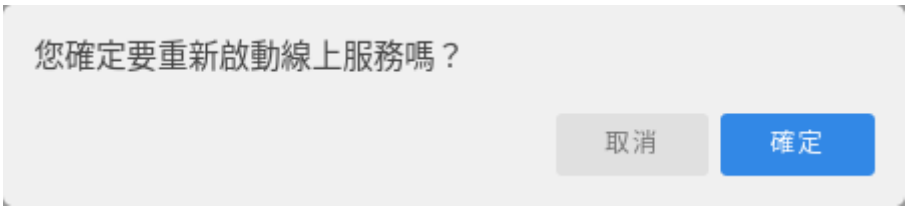
管理主控臺說明

頂部功能列



此部份需要注意的只有最右邊的「 系統 」按鈕，按下此按鈕後會出現包含「 重啟服務 」與「 變更帳號密碼」兩個按鈕的選單，其中：

重啟服務 ：重新啟動整個 OxOffice Online 服務，按下後會出現以下對話方塊：



在此對話方塊按下「取消」即會回到管理介面而不重新啟動服務，按下「確定」則會重新啟動服務，您必須等
待服務重新啟動完成才能繼續操作（不論是編輯器或管理介面皆如此）。

變更帳號密碼
：變更管理介面的帳號與密碼，您必須先輸入一次目前的管理介面帳號與密碼，按下「驗證」後再輸入新的帳
號與密碼。

側邊欄



左側邊欄可看見有數個功能分頁可供選擇，分別是「 概覽 」、「 分析 」、「 日誌 」、「 字型管理
」，以及下方會顯示目前所使用的 OxOffice Online 社群版與其使用的核心 OxOffice 版本。

概覽



線上使用者 ：目前正在使用 OxOfffice Online 的使用者數量 開啟的文件 ：目前由 OxOffice Online
開啟的文件數量 消耗的記憶體 ：目前 OxOffice Online 使用的記憶體空間 已傳送的流量 ：OxOffice Online 已傳送出去的資料量 已接收的流量 ：OxOffice Online 已從使用者處收到的資料量 服務時間 ：OxOffice Online 服務的已上線時間

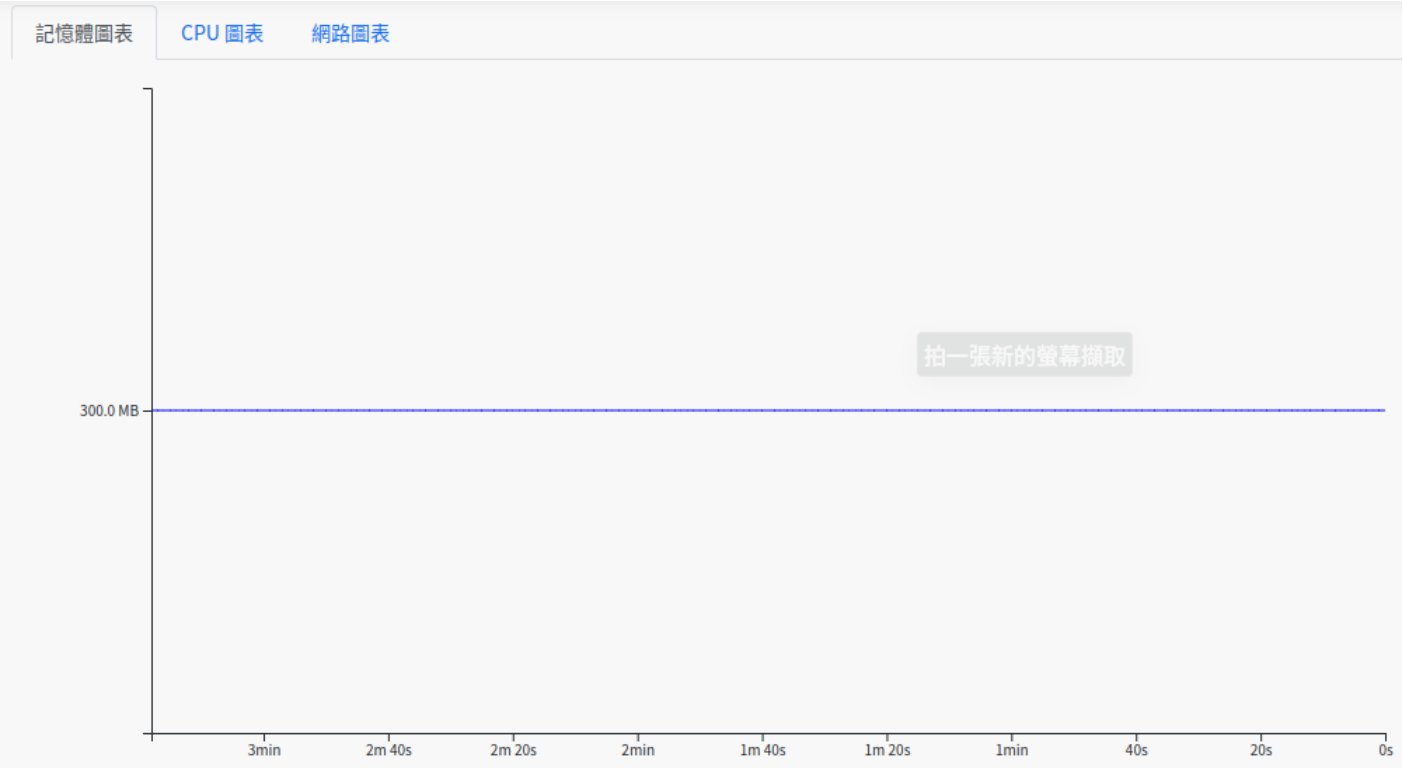
「開啟的文件」分頁

開啟者 ：開啟某個文件的使用者名稱 PID：開啟某個文件的處理程序 PID 文件 ：被開啟的文件 消耗的記憶體
：此處理程序所消耗的記憶體 經過的時間 ：此處理程序開啟後所經過的時間 閒置時間 ：使用者閒置的時間
最後修改 ：開啟的文件最後被修改的時間

「線上使用者」分頁

使用者名稱 ：目前在線上的使用者的名稱 文件數 ：特定的使用者所開啟的文件數

分析



「記憶體圖表」分頁

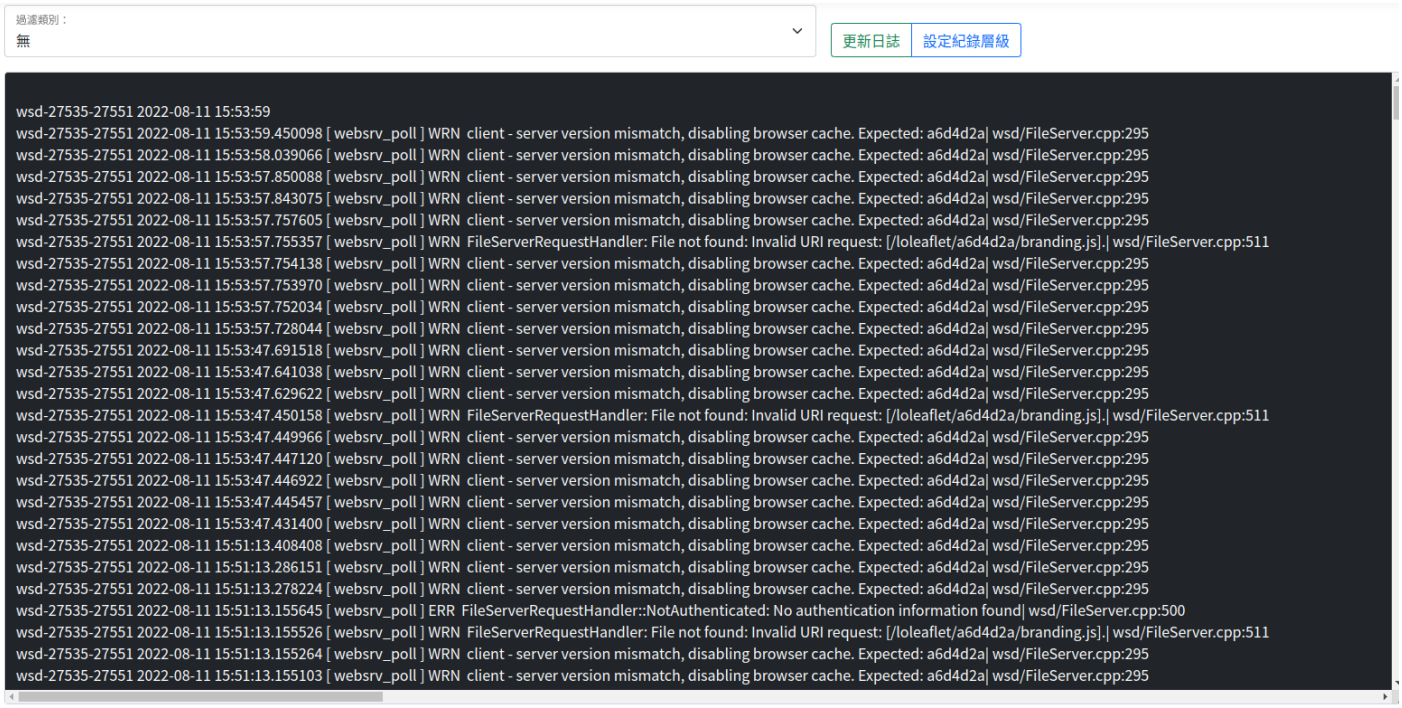
顯示消耗記憶體的圖表，橫軸由右至左為與目前相差的時間（最右邊的 0s
代表當下，向左則分別代表某個時間之前，如 1min即為 1分鐘前），縱軸則為消耗的記憶體大小

「CPU 圖表」分頁
顯示 CPU 使用率的圖表，橫軸代表意義與記憶體圖表的橫軸相同，而縱軸則為 CPU 使用率

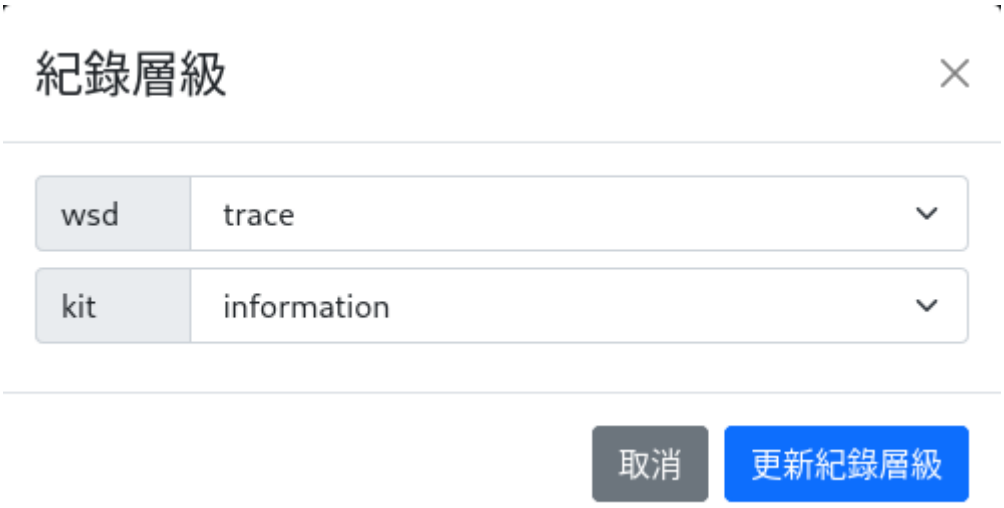
「網路圖表」分頁

顯示網路流量的圖表，橫軸代表意義與記憶體圖表的橫軸相同，而縱軸則為網路進出流量，可參考圖表旁的圖
例

日誌



□□□□ □□□□□□□□□□□□□□□□□□ □□□□ □□□□□□□□□□□□□□□□□□
□□□□□□ □□□□□□□□□□□□□□□□□



□□□□□ WebSocket Dameon □ OxOfficeKit □□□□□□□□□□□□□□□□□□□□ □□□□□□
□□□□□ □□□□ □□□ OxOffice Online □□□□□

□□□□

全選

5 個檔案 10 個字型

檔案名稱	字型名稱（預覽）	樣式	彩色字型	符號字型
kaiu.ttf	標楷體	Regular		
mingliu.ttc	MingLiU	Regular		
	PMingLiU	Regular		
	MingLiU_HKSCS	Regular		
msjh.ttc	Microsoft JhengHei	Regular		
	Microsoft JhengHei UI	Regular		
msjhbd.ttc	**Microsoft JhengHei**	Félkövér		
	Microsoft JhengHei UI	Félkövér		
msjhl.ttc	微軟正黑體 Light	Regular		
	Microsoft JhengHei UI Light	Regular		

清單部份會顯示目前可用的字型各項屬性，右下角的綠色圓形上傳按鈕則是用來從本機端上傳欲在 OxOffice Online 上提供使用的字型檔案