

【Mac】未安裝軟體

mac install for ctf

<https://medium.com/@seitzmanuel/how-to-get-your-mac-osx-ready-for-playing-ctfs-hacking-6b6801250d1e>

```
# homebrew
/bin/bash -c "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/HEAD/install.sh)"
brew update
```

```
curl https://bootstrap.pypa.io/get-pip.py > get-pip.py
python3 get-pip.py
python3 -m pip install pipx
ln -s <path_to_your_python_versions>/3.8/bin/pipx /usr/local/bin/pipx # for example
/Library/Frameworks/Python.framework/Versions/3.8/bin/pipx
```

```
brew install pyenv
brew install wget
brew install openssl
brew install pipx
brew install burp-suite
brew install wireshark
brew install samba
brew install swaks
brew install exploitdb
brew install john
brew install nmap
brew install gobuster
brew install metasploit
brew install sqlmap
brew install hashcat
brew install samba
brew install wpscanteam/tap/wpscan
brew install hydra
# 掃描工具
brew install nikto
or
brew install pyenv wget openssl burp-suite wireshark samba swaks exploitdb john nmap gobuster metasploit sqlmap hashcat samba
wpscanteam/tap/wpscan hydra nikto

# 文件掃描
brew install binwalk
# 無線網路密碼破解
brew install aircrack-ng
brew install owasp-zap
brew install ghidra
brew install exiftool
```

```
# NetExec
brew install pipx
pipx install git+https://github.com/Pennyw0rth/NetExec
```

```
# smbmap
git clone https://github.com/ShawnDEvans/smbmap.git /usr/local/Cellar/smbmap && python3 -m pip install -r
/usr/local/Cellar/smbmap/requirements.txt && ln -s /usr/local/Cellar/smbmap/smbmap.py /usr/local/bin/smbmap
# enum4linux
git clone https://github.com/CiscoCXSecurity/enum4linux.git /usr/local/Cellar/enum4linux && ln -s
/usr/local/Cellar/enum4linux/enum4linux.pl /usr/local/bin/enum4linux
```

```
pipx install crackmapexec
pipx install git+https://github.com/calebstewart/pwncat.git
```

```
# seclists
```

```
git clone https://github.com/3ndG4me/KaliLists.git /usr/local/share/wordlists && gzip -d /usr/local/share/wordlists/rockyou.txt.gz
wget -c https://github.com/danielmiessler/SecLists/archive/master.zip -O /tmp/master.zip ; unzip /tmp/master.zip -d /tmp ; mv
/tmp/SecLists-master /tmp/seclists ; mv /tmp/seclists /usr/local/share/
```

```
# chisel
wget https://github.com/jpillora/chisel/releases/download/v1.7.6/chisel_1.7.6_darwin_amd64.gz -O chisel_osx.gz && gunzip -c
chisel_osx.gz > linux/chisel_osx && rm chisel_osx.gz && chmod +x linux/chisel_osx
wget https://github.com/jpillora/chisel/releases/download/v1.7.6/chisel_1.7.6_linux_amd64.gz -O chisel_linux_64.gz && gunzip -c
chisel_linux_64.gz > linux/chisel_linux_64 && rm chisel_linux_64.gz
wget https://github.com/jpillora/chisel/releases/download/v1.7.6/chisel_1.7.6_linux_386.gz -O chisel_linux_386.gz && gunzip -c
chisel_linux_386.gz > linux/chisel_linux_386 && rm chisel_linux_386.gz
# PEASS-ng
wget https://raw.githubusercontent.com/carlospolop/PEASS-ng/master/linux/linpeas.sh -O linux/linpeas.sh
wget https://raw.githubusercontent.com/carlospolop/PEASS-ng/master/windows/winpeas.bat -O windows/winpeas.bat
wget https://github.com/carlospolop/PEASS-ng/raw/master/windows/winPEASx64/binaries/Release/winPEASany.exe -O
windows/winpeas.exe
wget https://github.com/carlospolop/PEASS-ng/raw/master/windows/winPEASx64/binaries/Obfuscated%20Releases/winPEASany.exe -O
windows/winpeas_obfuscated.exe
# linenum
wget https://raw.githubusercontent.com/rebootuser/LinEnum/master/LinEnum.sh -O linux/linenum.sh
# linux exploit suggester
wget https://raw.githubusercontent.com/mzet-/linux-exploit-suggester/master/linux-exploit-suggester.sh -O linux/linux-exploit-
suggester.sh
# lse
wget https://github.com/diego-treitos/linux-smart-enumeration/blob/master/lse.sh -O linux/lse.sh
# pspy
wget https://github.com/DominicBreuker/pspy/releases/download/v1.2.0/pspy64 -O linux/pspy64
wget https://github.com/DominicBreuker/pspy/releases/download/v1.2.0/pspy32 -O linux/pspy32
# powerup
wget https://raw.githubusercontent.com/PowerShellEmpire/PowerTools/master/PowerUp/PowerUp.ps1 -O windows/powerup.ps1
# jaws
wget https://raw.githubusercontent.com/411Hall/JAWS/master/jaws-enum.ps1 -O windows/jaws-enum.ps1
# print spoofer
wget https://github.com/itm4n/PrintSpoofer/releases/download/v1.0/PrintSpoofer32.exe -O windows/printspoofer.exe
# powershells revs
wget https://raw.githubusercontent.com/samratashok/nishang/master/Shell/Invoke-PowerShellTcp.ps1 -O reverse_shells/invoke-
powershelltcp.ps1
# php rev shell
wget https://raw.githubusercontent.com/pentestmonkey/php-reverse-shell/master/php-reverse-shell.php -O reverse_shells/php-rev-
shell.php
```

```
# TokenBreaker
wget https://raw.githubusercontent.com/cyberblackhole/TokenBreaker/master/RsaToHmac.py -O jwt/RsaToHmac.py && python3 -m pip
install -r https://raw.githubusercontent.com/cyberblackhole/TokenBreaker/master/requirements.txt
chmod +x jwt/RsaToHmac.py
wget https://raw.githubusercontent.com/cyberblackhole/TokenBreaker/master/TheNone.py -O jwt/TheNone.py
chmod +x jwt/TheNone.py
# jwt cracker
mkdir -p jwt/jwt-cracker
wget https://raw.githubusercontent.com/brendan-rius/c-jwt-cracker/master/Makefile -O jwt/jwt-cracker/Makefile
wget https://raw.githubusercontent.com/brendan-rius/c-jwt-cracker/master/base64.c -O jwt/jwt-cracker/base64.c
wget https://raw.githubusercontent.com/brendan-rius/c-jwt-cracker/master/base64.h -O jwt/jwt-cracker/base64.h
wget https://raw.githubusercontent.com/brendan-rius/c-jwt-cracker/master/main.c -O jwt/jwt-cracker/main.c
cd jwt/jwt-cracker && make OPENSSL=/usr/local/opt/openssl/include OPENSSL_LIB=-L/usr/local/opt/openssl/lib && cd ../..
# hash identifier
wget https://raw.githubusercontent.com/blackploit/hash-identifier/master/hash-id.py -O misc/hash-id.py
chmod +x misc/hash-id.py
# linkfinder
git clone https://github.com/GerbenJavado/LinkFinder.git misc/linkfinder
cd misc/linkfinder
python3 -m pip install -r requirements.txt
python3 setup.py install
chmod +x linkfinder.py
cd ../..
# Pentest Scripts
wget https://raw.githubusercontent.com/chikko80/Pen-Scripts/master/basic_scanner.py -O misc/basic_scanner.py
wget https://raw.githubusercontent.com/chikko80/Pen-Scripts/master/hydra_builder.py -O misc/hydra_builder.py
wget https://raw.githubusercontent.com/chikko80/Pen-Scripts/master/string_finder.py -O misc/string_finder.py
```

```
python3 -m pip install -r https://raw.githubusercontent.com/chikko80/Pen-Scripts/master/requirements.txt
chmod +x misc/*
```

```
#kali
/usr/bin/unix-privesc-check
#./unix-privesc-check standard > output.txt
```

Install Metasploit on OS X

<https://gist.github.com/xl7dev/a19da077792c5894529f>

```
# XCode Command Line Tools

>xcode-select --install

# Install Homebrew

>ruby -e "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/master/install)"
>echo PATH=/usr/local/bin:/usr/local/sbin:$PATH >> ~/.bash_profile
>source ~/.bash_profile
>brew tap homebrew/versions

# Install nmap

>brew install nmap

# Install libxml2

>brew install libxml2

# Install PostgreSQL

#>brew install postgresql --without-ossdp-uuid
>brew install postgresql

# ruby 2.1.X

#brew install homebrew/versions/ruby21
>brew install ruby
#ruby is keg-only, which means it was not symlinked into /usr/local,
#because macOS already provides this software and installing another version in
#parallel can cause all kinds of trouble.

#If you need to have ruby first in your PATH, run:
# echo 'export PATH="/usr/local/opt/ruby/bin:$PATH"' >> ~/.zshrc

#For compilers to find ruby you may need to set:
# export LDFLAGS="-L/usr/local/opt/ruby/lib"
# export CPPFLAGS="-I/usr/local/opt/ruby/include"

#For pkg-config to find ruby you may need to set:
# export PKG_CONFIG_PATH="/usr/local/opt/ruby/lib/pkgconfig"

# Initialize the database

>initdb /usr/local/var/postgres

#Success. You can now start the database server using:
# pg_ctl -D /usr/local/var/postgres -l logfile start

>mkdir -p ~/Library/LaunchAgents
#>cp /usr/local/Cellar/postgresql/9.4.0/homebrew.mxcl.postgresql.plist ~/Library/LaunchAgents/
>cp /usr/local/Cellar/postgresql@14/14.12/homebrew.mxcl.postgresql@14.plist ~/Library/LaunchAgents/
#>launchctl load -w ~/Library/LaunchAgents/homebrew.mxcl.postgresql.plist
>launchctl load -w ~/Library/LaunchAgents/homebrew.mxcl.postgresql@14.plist
>echo "alias pg_start='pg_ctl -D /usr/local/var/postgres -l /usr/local/var/postgres/server.log start'"
>echo "alias pg_stop='pg_ctl -D /usr/local/var/postgres stop'"

# Create the db for the metasploit framework
```

```
>createuser msf -P -h localhost
>createdb -O msf msf -h localhost

# Clone the Git Metasploit

>git clone https://github.com/rapid7/metasploit-framework.git /usr/local/share/metasploit-framework

# settings Env

>echo 'alias msfconsole="/usr/local/share/metasploit-framework && ./msfconsole && cd -"' >> ~/.zshrc
>echo 'alias msfbinscan="/usr/local/share/metasploit-framework && ./msfbinscan && cd -"' >> ~/.zshrc
>echo 'alias msfd="/usr/local/share/metasploit-framework && ./msfd && cd -"' >> ~/.zshrc
>echo 'alias msfelfscan="/usr/local/share/metasploit-framework && ./msfelfscan && cd -"' >> ~/.zshrc
>echo 'alias msfmachscan="/usr/local/share/metasploit-framework && ./msfmachscan && cd -"' >> ~/.zshrc
>echo 'alias msfpescan="/usr/local/share/metasploit-framework && ./msfpescan && cd -"' >> ~/.zshrc
>echo 'alias msfrop="/usr/local/share/metasploit-framework && ./msfrop && cd -"' >> ~/.zshrc
>echo 'alias msfrpc="/usr/local/share/metasploit-framework && ./msfrpc && cd -"' >> ~/.zshrc
>echo 'alias msfrpcd="/usr/local/share/metasploit-framework && ./msfrpcd && cd -"' >> ~/.zshrc
>echo 'alias msfupdate="/usr/local/share/metasploit-framework && ./msfupdate && cd -"' >> ~/.zshrc
>echo 'alias msfvenom="/usr/local/share/metasploit-framework && ./msfvenom && cd -"' >> ~/.zshrc
>sudo chmod go+w /etc/profile
>sudo echo export MSF_DATABASE_CONFIG=/usr/local/share/metasploit-framework/config/database.yml >> /etc/profile
>cd /usr/local/share/metasploit-framework
>sudo gem install bundler:2.1.4
>sudo gem install activesupport -v '6.1.4.1'
>bundle install

# Create the Database Configuration

>vim /usr/local/share/metasploit-framework/config/database.yml
Paste the following text:

production:
  adapter: postgresql
  database: msf
  username: msf
  password: <your password>
  host: 127.0.0.1
  port: 5432
  pool: 75
  timeout: 5

# update your environment

>source /etc/profile
>source ~/.bash_profile

> msfconsole
```

🕒修訂版本 #15

★由 treeman 建立於 10 🕒C🕒🕒 2024 19:56:47

🔧由 treeman 更新於 15 🕒C🕒🕒 2024 15:56:32