Ubuntu Linux

- □□□□
- □□□□
- SSL
- Docker
- clamav
- SSH□□□□□
- bash
- VPS□□
- Ubuntu 22.04 VNC
- MySQL
- □□ Fail2Ban □□□□□□
- □□□□ htop Glances
- □□ Monit □□□□
- □□ log
- pm2□□

□□□□

□□□ root□□

```
sudo su
```

> □□ vps□□□□□ root□□□□□□□□□□□□□ root □□□□□□□□□□□□□□□□□□□□□□□□□□□□□□ XD

□□□□

```
dpkg-reconfigure tzdata
```

□□□□

```
vi /etc/locale.gen
□ zh_TW.UTF-8 mark□□
locale-gen
```

□□□□□□□□□□

```
apt update
apt dist-upgrade
reboot
```

□□□□□□□□□□ root □□

```
sudo su
```

□□□□

```
apt install nginx php-fpm php-gd php-mbstring php-gd php-xml php-mysql mysql-server mysql-client certbot python3-certbot-nginx
```

> Drupal[illegible] WordPress[illegible] php[illegible]

> [illegible] apt-install php[illegible] apache2

[illegible] MySQL

[illegible] mysql[illegible]

[illegible] root[illegible] mysql

```
mysql -u root
```

[illegible]

```
ALTER USER 'root'@'localhost' IDENTIFIED WITH caching_sha2_password BY 'yourpassword'; exit;
```

> yourpassword [illegible]

> [illegible] root[illegible]

[illegible] conf[illegible]

```
[client]
default-character-set = utf8mb4

[mysqld]
character-set-server = utf8mb4
max_connections = 600
innodb_read_io_threads = 24
innodb_write_io_threads = 24
expire_logs_days = 7
max_binlog_size = 1G
```

□□□□□□ server□ binlog_size□□□□□ log□□□□□□□□□□□

□ ~/.my.cnf □□□□

```
[client]
password = yourpassword
```

MySQL□□□□□□□

□□□□□

```
mysqladmin -u root create site
```

□□□□□

```
mysqladmin -u root drop site
```

□□□□□

```
mysqldump -u root site > site.sql
```

□□□□□

```
mysql -u root site < site.sql
```

□□□□□□□□□□□□□□□□ phpMyAdmin XD

□□□□□□□□□□□

```
ufw enable
ufw allow from xxx.xxx.xxx.xxx (□□□□□□□ IP)
ufw allow 80/tcp
ufw allow 'Nginx HTTP'
ufw allow 'Nginx HTTPS'
```

□□□□□□□ **php□□**

```
add-apt-repository ppa:ondrej/php
aptitude update
```

[illegible]

```
ls
```

[illegible]

```
ls -la
```

[illegible]

> Linux[illegible] .

```
ls ~/
```

[illegible]

> [illegible] /home/ ; [illegible] /root/

```
mkdir test
```

[illegible]

```
mkdir -p test/a
```

[illegible]

```
cp a.php b.php
```

[illegible]

> [illegible] copy nginx [illegible] cp /etc/nginx/sites-enabled/site1 /etc/nginx/sites-enabled/site2

```
cp -r ~/abc /var/www/
```

□□□□□□□□□□□ abc □□ /var/www/

> □□□□□□□□□□□ tab□□□□□□□□□□□□□□□□□ /var/w □□□ tab □□□□□□ /var/www/

```
rm xxx.txt
```

□□□□

```
rm -r xxx
```

□□□□

> □□□ linux□□□□□□□□□□□□□□□□□□ linux□□□□□□□□□□□□□□□

> □□□□□□□□□□□□□□□□□□ GG□ rm -rf /

```
mv ~/abc /var/www/
```

□□□□□□□□□□□□

```
tar zcf abc.tar.gz abc
```

□□□□□□□□ abc□□□ abc.tar.gz

```
tar zxvf abc.tar.gz
```

□□□□□□□ abc.tar.gz □□

```
unzip abc.zip
```

□□□□□ zip□□□□□□

```
cat /var/log/nginx/access.log
```

□□□□□□□□□□□□□□□□□□□□□□□□ log□

```
cat /var/log/nginx/access.log | more
```

□□□□□□□□□□□□□□ | more□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□ Ctrl + C □□□□

```
cat /var/log/nginx/access.log | less
```

□□□ more□□□□□□□□□□

```
zcat /var/log/nginx/access.log.10.gz
```

□□□ log□□□□□□□□□□□ zcat□□□□□□□ .gz□□□

```
htop
```

□□□□□□□□ CPU□□□□□□□□□□□

```
vi xxx.txt
nano xxx.txt
```

□□□□□□□□□□ vi□ nano□□□□

```
ssh name@ip
```

□□ ssh□□□□□

```
scp -r /var/www/ name@ip:/var/www/
```

□□ ssh□□□□□□□□□□

```
rsync -av --delete /var/www/wp/wp-content/uploads/ root@ip:/var/www/wp/zfun/wp-content/uploads/
rsync -av --delete /var/www/wp/wp-content/plugins/ root@ip:/var/www/wp/zfun/wp-content/plugins/
```

□□ rsync□□□□

> □□□□□□□□□□□□□□□
> □□□□□ ssh-keygen -t rsa
> cat .ssh/id_rsa.pub
> □□□□□□ .ssh/authorized_keys □□ cat□□□□□
> □□□□□□□□□□□□□□□□□□

```
crontab -e
```

□□□□□□□□□□□□□□□□□□□□□□□□□□

```
/etc/init.d/nginx reload
```

□□□□ nginx□□□□□□□□□□□□□□□□□□ conf□□□□□□ nginx -t □□□□□□□□

```
nginx: the configuration file /etc/nginx/nginx.conf syntax is ok
nginx: configuration file /etc/nginx/nginx.conf test is successful
```

□□□□□□□□□□□□□□□□□□ reload

```
/etc/init.d/nginx restart
```

□□□□□□

> □□□□□□□□□□□□□□□□□□ reload, restart, stop, start □□□□□□□
> Nginx: /etc/init.d/nginx
> PHP: /etc/init.d/php7.4-fpm
> MySQL: /etc/init.d/mysql
> □□□□□□□□□□□ cron□□ restart□□□□□□□□□

□□ shell script□□

> □□□□□□□□□□□□□□□□□□□□□□□ 2023□ AI
> □□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□

SSL

□□□□□□□ DNS□□ Cloudflare□□□□□□ Cloudflare□□ SSL

SSL/TLS □□□□ □□ □□□□
□□□ □□□□□□□ □□□□ □ cert.pem □ key.pem □□□ /etc/ssl/ □ □□ □□□□□□□

```
 listen 443 ssl;
    include /etc/letsencrypt/options-ssl-nginx.conf;
    ssl_dhparam /etc/letsencrypt/ssl-dhparams.pem;
    ssl_certificate         /etc/ssl/cert.pem;
    ssl_certificate_key     /etc/ssl/key.pem;
    ssl_client_certificate /etc/ssl/cloudflare.crt;
    ssl_verify_client on;
}
server {
    if ($host = site.com) {
        return 301 https://$host$request_uri;
    }



    server_name site.com;
    listen 80;
    return 404;
```

□□□□ conf□□ server □□□□□ "}" □□□□

□ Nginx□□ SSL□□□□□□□□
https://www.digitalocean.com/community/tutorials/how-to-host-a-website-using-cloudflare-and-nginx-on-ubuntu-20-04

□□□□□□□□□□ Certbot

```
apt instll certbot
```

□□

```
certbot -d site.com
```

[illegible]

> [illegible] /etc/nginx/sites-enabled/ [illegible] site.com [illegible]
> /etc/nginx/nginx.conf

```
certbot delete --cert-name site.com
```

[illegible] SSL

> [illegible] Ubuntu 22.04
> [illegible] cron[illegible] certbot renew

Cloudflare[illegible] SSL[illegible] A[illegible] SSL/TLS [illegible] **TLS** [illegible] **1.2** [illegible] A[illegible] **HTTP** [illegible] **(HSTS)**

> [illegible] SSL[illegible] subdomain[illegible] SSL[illegible] Mac [illegible] subdomain
> [illegible] Cloudflare
> [illegible] Certbot

Docker

```
apt install docker docker-compose
```

> Docker-compose □□□□□□□□□□□□□□□□□
> https://hackmd.io/@jimmy801/docker_compose_install

□□ docker□□□

```
docker pull portainer/portainer-ce:latest
docker run -d -p 9000:9000 --restart always -v /var/run/docker.sock:/var/run/docker.sock portainer/portainer-ce:latest
```

□□ web□□□□

```
server {
    server_name docker.site.com;

    location ~/ {
    # prevents 502 bad gateway error
    proxy_buffers 8 32k;
    proxy_buffer_size 64k;

    client_max_body_size 75M;

    # redirect all HTTP traffic to localhost:9000;
    proxy_pass http://localhost:9000;
    proxy_set_header X-Real-IP $remote_addr;
    proxy_set_header Host $http_host;
    proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
    #proxy_set_header X-NginX-Proxy true;

    # enables WS support
    proxy_http_version 1.1;
    proxy_set_header Upgrade $http_upgrade;
```

```
    proxy_set_header Connection "upgrade";

    proxy_read_timeout 999999999;
  }



}
```

> □□□□□ Docker□□□ Docker□□□□□□□□□□□□ IP + port □□
> □□□□ VPS□□ 9000□ port□□□□□□ Nginx Proxy □□□□□ Docker□□□□□□□ port

□□ WordPress

□□□□□□□□□□□□□ WordPress□□□□□□□□□□ Stacks

```
version: '3.1'

services:

  wordpress:
    image: wordpress:6.0.1-php7.4-apache
    restart: always
    ports:
      - 8001:80
    environment:
      WORDPRESS_DB_HOST: db
      WORDPRESS_DB_USER: exampleuser
      WORDPRESS_DB_PASSWORD: examplepass
      WORDPRESS_DB_NAME: exampledb
    volumes:
      - wordpress:/var/www/html

  db:
    image: mysql:5.7
    restart: always
    environment:
      MYSQL_DATABASE: exampledb
      MYSQL_USER: exampleuser
      MYSQL_PASSWORD: examplepass
```

```
      MYSQL_RANDOM_ROOT_PASSWORD: '1'
    volumes:
      - db:/var/lib/mysql

volumes:
  wordpress:
  db:
```

□□□□□□□□ Nginx□□□

```
server {
    server_name demo1.site.com;

    location ~/ {
    # prevents 502 bad gateway error
    proxy_buffers 8 32k;
    proxy_buffer_size 64k;

    client_max_body_size 75M;

    # redirect all HTTP traffic to localhost:8088;
    proxy_pass http://localhost:8001;
    proxy_set_header X-Real-IP $remote_addr;
    proxy_set_header Host $http_host;
    proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
    #proxy_set_header X-NginX-Proxy true;

    # enables WS support
    proxy_http_version 1.1;
    proxy_set_header Upgrade $http_upgrade;
    proxy_set_header Connection "upgrade";

    proxy_read_timeout 999999999;
}



}
```

> □□□□□□□ https://demo1.site.com □□□□□ Docker□□□□□□□

```
docker exec -i -t 8001-wordpress-1 bash
```

[illegible] WordPress[illegible]

```
echo -e "file_uploads = On\nmemory_limit = 500M\nupload_max_filesize = 500M\npost_max_size =
500M\nmax_execution_time = 600" | tee -a /usr/local/etc/php/conf.d/uploads.ini;/etc/init.d/apache2 restart
```

[illegible] docker[illegible]

```
docker cp /4TB/conoha2/www/wp/zfun/wp-content/uploads demo38-wordpress-1:/var/www/html/wp-content/
docker cp /4TB/conoha2/www/wp/site2/wp-content/themes demo38-wordpress-1:/var/www/html/wp-content/
docker cp /4TB/conoha2/www/wp/site2/wp-content/uploads demo38-wordpress-1:/var/www/html/wp-content/
```

Docker[illegible] WordPress[illegible] [illegible] WordPress[illegible] Apache2[illegible]

[illegible] WordPress[illegible] Docker[illegible] XD [illegible] WordPress[illegible]

clamav

□□□□

```
apt install clamav clamav-daemon
```

□□□□□□□□□□ up_clamav.sh

```
service clamav-freshclam stop
freshclam
service clamav-freshclam start
```

□□□□ □□□□□□□ clamav_scan.sh

```
clamscan -r --bell -i /var/www/
```

SSH□□□□

□□□□□□□□□□

```
ssh-keygen -t rsa
```

□□□

```
cat .ssh/id_rsa.pub
```

□□□□ □□□□□□

```
vi .ssh/authorized_keys
```

□□□□□□□□□□ ssh@ip □□□□□

□□□□ GCP□□□□□□□□□□□□□□□□□□□□□□ ssh□□

bash

□□□□□ mysql□□□

```
#!/bin/bash

BACKUP_DIR="/root/backup/db"

if [ ! -d "$BACKUP_DIR" ]; then
    mkdir -p $BACKUP_DIR
fi

# □□□□□□ .gz □□
rm -f $BACKUP_DIR/*.gz

DATABASES=$(mysql -u root -e "SHOW DATABASES;" | grep -Ev
"(Database|information_schema|performance_schema|mysql)")

for DB in $DATABASES; do
    BACKUP_FILE="$BACKUP_DIR/${DB}_backup_$(date +'%Y%m%d').sql"

    MYSQLDUMP_CMD="mysqldump -u root $DB > $BACKUP_FILE"

    echo "□□□□□□□ $DB..."
    eval $MYSQLDUMP_CMD

    if [ $? -eq 0 ]; then
        echo "□□□ $DB □□□□ : $BACKUP_FILE"
        gzip $BACKUP_FILE
        echo "□□□□□□ : $BACKUP_FILE.gz"
        # □□□□□□□□□ MB
        FILE_SIZE=$(du -m "$BACKUP_FILE.gz" | cut -f1)
        echo "□□□□□□□ : $FILE_SIZE MB"
    else
        echo "□□□ $DB □□□□ "
    fi
```

```
done
```

□□□□

```
#!/bin/bash
rsync -avz --delete --progress --exclude='cache/' --exclude='*demo*' root@ip:/etc/nginx/sites-enabled/
/4TB/conoha1/nginx/
```

□□□□□□□

```
#!/bin/bash

# □□□□□□□□□□□
SRC_DIR="/var/www/"
DEST_DIR="/root/gcp/www/"

# □□□□
DIR_LIST=("web1" "web2" "web3")

# □□□□□□□□□
for DIR_NAME in "${DIR_LIST[@]}"; do
  FILE_NAME="${DEST_DIR}${DIR_NAME}.tar.gz"
  tar zcf "$FILE_NAME" -C "$SRC_DIR" "$DIR_NAME"
  echo "□□□□        $DIR_NAME □  $FILE_NAME □□    "
done

echo "□□□□□□□□              "
```

VPS[illegible]

[illegible] VPS[illegible] XD

[illegible]

[illegible] ssh[illegible]

Ubuntu 22.04 VNC

□□□□□□

https://jackfrisht.medium.com/ubuntu-20-04-vnc%E5%AE%89%E8%A3%9D-b8ce6e7dec4a

□□□□□ (□□)

```
sudo apt-get update
sudo apt-get upgrade
```

Ubuntu 22.04 □□□□ gdm3□□□ VNC □□□□ (□□□□□□□□□□□□□□□□□□)□□□□□ lightdm

Ubuntu 16.04 □□□□ lightdm □□□□□□□□□

```
sudo apt install lightdm  ## ubuntu 22.04 □□□
sudo dpkg-reconfigure lightdm    ## □□□□□□ lightdm
```

□□ x11VNC□

```
sudo apt-get install x11vnc -y
```

□□□□□□□□□□□□□□□□□ error□□□□□□

```
sudo apt-get update
```

□□□□□ **VNC□□**

```
sudo x11vnc -storepasswd /etc/x11vnc.pass
```

□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□

```
sudo chmod 755 /etc/x11vnc.pass
```

`rfbport` □□□□□□□□□□ `-forever` □□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□

```
sudo x11vnc -auth guess -rfbauth /etc/x11vnc.pass -rfbport 5900 -forever -display :0
```

□□□□□□□□ **VNC □□**

```
sudo gedit /etc/systemd/system/x11vnc.service[Unit]
Description=Start x11vnc at startup.
```

```
After=multi-user.target
[Service]
Type=simple
ExecStart=/usr/bin/x11vnc -auth guess -forever -loop -noxdamage -repeat -rfbauth /etc/x11vnc.pass -rfbport 5900
[Install]
WantedBy=multi-user.target
```

□□□ systemd □□□□

```
sudo systemctl daemon-reload

sudo systemctl enable x11vnc

sudo systemctl start x11vnc
```

Step 2. □□□□□□

□□□□□□ Nvidia driver □□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□

Server □□□□□□□□□□□ x-session □□□□□□□□□□□□□□ xorg.conf □□□□□□□□□

ubuntu □□□□□ /etc/X11/xorg.conf□□□□□□ Xorg -configure □□□□□□□□□□□□□□□□□□□□ (□□□□□□□□□ 1920x1080□□□□□□□□□□□ Virtual □□)

turn to /etc/X11

```
cd /etc/X11sudo gedit xorg.conf## VNC-virsual montior
Section "Device"
	Identifier "Configured Video Device"
EndSectionSection "Monitor"
	Identifier "Configured Monitor"
EndSectionSection "Screen"
	Identifier "Default Screen"
	Monitor "Configured Monitor"
	Device "Configured Video Device"
	SubSection "Display"
		Depth 24
		Virtual 1920 1080
	EndSubSection
EndSection
```

Restart System

```
sudo reboot
```

□□□□□□□□

```
systemctl status sleep.target  sleep.target - Sleep
```

□□□□□□

```
systemctl mask sleep.target suspend.target hibernate.target hybrid-sleep.target
```

□□□□□□□□

```
systemctl status sleep.target
```

MySQL

htop □□ mysqld □ 100%

```
mysql -u root
```

□□□□□

```
SHOW FULL PROCESSLIST;
```

□□□□□□□□□□□□□□□□□□□

```
| 32892 | root | localhost | NULL | Query   |    0 | starting     | SHOW FULL PROCESSLIST
```

□□□□

```
KILL 32892;
```

□□ Fail2Ban □□□□□□

```
vi /etc/fail2ban/jail.local
```

□□

```
[nginx-http-auth]
enabled  = true
filter   = nginx-http-auth
action   = iptables[name=HTTP, port=http, protocol=tcp]
logpath  = /var/log/nginx/error.log
bantime  = 3600
findtime = 600
maxretry = 5
```

□□□□□□

```
systemctl restart fail2ban
```

□□□□□□□ IP

```
fail2ban-client status
```

□□□□□

```
[nginx-primary-script]
enabled = true
filter = nginx-primary-script
action = iptables[name=PrimaryScript, port=http, protocol=tcp]
logpath = /var/log/nginx/error.log
maxretry = 3
bantime = 3600
```

□□□□□□

```
[Definition]
failregex = .*FastCGI sent in stderr: "Primary script unknown".*request: ".*installer\.php.*
            .*FastCGI sent in stderr: "Primary script unknown".*request: ".*WordPress/installer\.php.*
```

```
        .*FastCGI sent in stderr: "Primary script unknown".*request: ".*phpinfo\.php.*
        .*FastCGI sent in stderr: "Primary script unknown".*request: ".*info\.php.*

# □□□□□□□□
        .*GET /wp-admin.*                    # WordPress □□□□□
        .*GET /wp-login\.php.*               # WordPress □□□
        .*GET /xmlrpc\.php.*                 # WordPress XML-RPC □□
        .*GET /phpMyAdmin/.*                 # phpMyAdmin □□
        .*GET /pma/.*                        # phpMyAdmin □□□
        .*GET /myadmin/.*                    # □□□□□□□
        .*GET /config\.php.*                 # □□□□□□□
        .*GET /setup\.php.*                  # □□□
        .*GET /install\.php.*                # □□□
        .*GET /adminer.*                     # Adminer □□

# □□□□□□□
        .*GET /shell\.php.*                  # □□□
        .*GET /cmd\.php.*                    # □□□□□
        .*GET /console\.php.*                # □□□□□
        .*GET /backdoor\.php.*               # □□□□□
        .*GET /wp-content/uploads/shell.*    # WordPress □□□
        .*GET /eval-stdin.*                  # eval □□□

# □□□□□
        .*GET /\.env.*                       # Laravel □□□□□□□
        .*GET /\.git/config.*                # Git □□□
        .*GET /backup.*                      # □□□□□
        .*GET /dump.*                        # □□□□□□□
        .*GET /debug.*                       # □□□
        .*GET /error_log.*                   # □□□

# □□□□□
        .*GET /HNAP1/.*                      # HNAP □□□    (□□□□□    )
        .*GET /boaform/admin/formLogin.*     # IOT □□□
        .*GET /invoker/JMXInvokerServlet.*   # JBoss □□
        .*GET /webdav.*                      # WebDAV □□
        .*GET /manager/html.*                # Tomcat □□□□□
```

□□□□□ IP

```
fail2ban-client set nginx-primary-script banip xxx.xxx.xxx.xxx
```

□□□□□□□

```
fail2ban-client set nginx-primary-script unbanip 192.168.1.100
```

□□□ □□ /etc/fail2ban/jail.local

```
[DEFAULT]
ignoreip = 127.0.0.1/8 ::1 192.168.1.100
```

□□□□□□□□□□□□

```
systemctl restart fail2ban
```

□□□□ htop Glances

□□□□

```
pip install glances fastapi[all] uvicorn
```

□□ glances -w □□

□□ which glances □□ /usr/local/bin/glances□□□ Glances □□□□□□ /usr/local/bin □□□□□□□□□□□□□□

□□□□ /usr/local/bin□□□□□□□□□□□□□

```
export PATH=$PATH:/usr/local/bin
```

□□□□□□□□ .bashrc □ .zshrc □□□□

```
echo 'export PATH=$PATH:/usr/local/bin' >> ~/.bashrc
source ~/.bashrc
```

□□ Monit □□□

□□

```
apt install monit
```

□□

```
cp /etc/monit/monitrc /etc/monit/monitrc.bak
```

vi /etc/monit/monitrc

```
set httpd port 2812
    use address 0.0.0.0  # □□□□□
    allow localhost      # □□    localhost □□
    allow yourip         # □□    □□   IP □□
    allow admin:monit

# NGINX
check process nginx with pidfile /var/run/nginx.pid
  start program = "/bin/systemctl start nginx"
  stop program  = "/bin/systemctl stop nginx"
    if failed port 443 then restart
    if 5 restarts within 5 cycles then timeout

# PHP 7.4-FPM
check process php7.4-fpm with pidfile /run/php/php7.4-fpm.pid
    start program = "/etc/init.d/php7.4-fpm start"
    stop program = "/etc/init.d/php7.4-fpm stop"
    if failed unixsocket /var/run/php/php7.4-fpm.sock then restart
    if 5 restarts within 5 cycles then timeout

# MySQL
check process mysql with pidfile /var/lib/mysql/zfun-server.pid
    start program = "/etc/init.d/mysql start"
    stop program = "/etc/init.d/mysql stop"
    if failed host localhost port 3306 protocol mysql then restart
    if 5 restarts within 5 cycles then timeout
```

http://yourip:2812 [illegible]

[illegible] vi /etc/monit/monitrc

```
set mailserver smtp.your-email-provider.com port 587
    username "your-email@example.com" password "your-password"
    using tlsv1
    with timeout 30 seconds

set alert your-email@example.com
```

> [illegible]

□□ log

□□□□□□□□□□□□□ 2024□ 11□ 20□□□ 2□

```
grep '20/Nov/2024:14:' /var/log/nginx/access.log | grep 'https://yoursite.com' | awk '{print $1}' | sort | uniq | wc -l
zgrep '20/Nov/2024:14:' /var/log/nginx/access.log.2.gz | grep 'https://yoursite.com' | awk '{print $1}' | sort | uniq | wc -l
```

pm2□□

□□□□□□□□□ pm2 □□□□□□□□□□□□□□□□□□□□

---

1. □□□□□□□□□□

(1) □□□□□□

□□□□□□□□□□□□□□□□□□

```
tar -czf projects_backup.tar.gz /path/to/your/project1 /path/to/your/project2
```

(2) □□ pm2 □□

□□ pm2 □□□□□□□□□□

```
pm2 save
```

□□□□□□□□□□□□ ~/.pm2/dump.pm2 □□□□□□□□□□

```
tar -czf pm2_backup.tar.gz ~/.pm2
```

(3) □□□□□□ (□□)

- □□□□□□□□ Node.js□□□□□□□□□□
  `node -v`
- □□□□□□ .env □□□□□□□□□□□□□□□□□□

---

2. □□□□□□□

(1) □□ Node.js □ npm

□□□□□□□ Node.js□□□□□□□□□□□□□□□□□□□□□□□□

```
curl -fsSL https://deb.nodesource.com/setup_18.x | sudo -E bash -
sudo apt install -y nodejs
```

(□□□□□□□□□□□□□)

(2) □□ pm2

□□ `pm2` □

```
sudo npm install -g pm2
```

---

3. □□□□

(1) □□□□□□

□□□□□□□□□□□□□□□□□□□□□□□□

```
scp projects_backup.tar.gz user@new-server:/path/to/
ssh user@new-server
tar -xzf projects_backup.tar.gz -C /desired/path/
```

(2) □□ `pm2` □□

□ `pm2_backup.tar.gz` □□□□□□□□□□□

```
scp pm2_backup.tar.gz user@new-server:/path/to/
ssh user@new-server
tar -xzf pm2_backup.tar.gz -C ~/
```

□□□□□□□□□

```
pm2 resurrect
```

---

4. □□□□□

(1) □□□□□□

□□□□□□□□□□□□□□□□□□

```
pm2 list
```

□□□□□□□□□□□□□□□

```
pm2 logs
```

(2) □□□□□□

□□ `pm2` □□□□□□□□□□□

```
pm2 startup
pm2 save
```

---

5. □□ DNS □□□□□□

□□□□□□□□□□□ DNS □□□□□□□□□□□□□□□□ IP□

□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□ pm2 □□□□□