

Linux???????

- [ufw????IP????????????????](#)
- [MySQL????????????????](#)

ufw????IP?????????????????????

?

??

1. ?????????????????????????????????IP?????????????????????????
2. ufw?????????????
3. ????????/24?/16?????????????????
4. ??????????????????????
5. ???ufw?????????????????

???????

- ufw_deny.sh

```
#!/bin/bash

while true; do
    # IP
    read -p "[ ]IP[ ]:" ip_address

    # 
    read -p "IP[ ]0/24[ ]1/16[ ]2[ ]0: " block_option

    # [ ]IP[ ]
    block_option=${block_option:-0}

    # 
    case $block_option in
        1) block_cidr="/24" ;;
        2) block_cidr="/16" ;;
        *) block_cidr="" ;;
    esac

    # 
    if [[ -n "$block_cidr" ]]; then
        if [[ $block_option == "1" ]]; then
            # /24
        fi
    fi
done
```

```

network_address=$(echo $ip_address | cut -d'.' -f1-3)
# CIDR[ ] ( /24)
cidr_notation="$network_address.0$block_cidr"
elif [[ $block_option == "2" ]]; then
# /16[ ]
network_address=$(echo $ip_address | cut -d'.' -f1-2)
# CIDR[ ] ( /16)
cidr_notation="$network_address.0.0$block_cidr"
fi
else
cidr_notation="$ip_address"
fi

# [ ]
echo "[ ]:"
echo "IP[ ]: $cidr_notation"

read -p "[ ]y/n: " confirm

# [ ]
if [[ $confirm != "y" && $confirm != "Y" ]]; then
echo "[ ]"
exit 0
fi

# [ ]ufw[ ]
sudo ufw deny from "$cidr_notation"

# [ ]
if [[ $? -ne 0 ]]; then
echo "ERROR: [ ]"
exit 1
fi

echo "[ ]"

# [ ]IP[ ]
read -p "[ ]IP[ ]y/n: " continue_blocking
if [[ $continue_blocking != "y" && $continue_blocking != "Y" ]]; then
break

```

```

fi
done

# ufw[ ]
sudo ufw reload

# [ ]
read -p "[ ]y/n: " check_status
if [[ $check_status == "y" || $check_status == "Y" ]]; then
    sudo ufw status numbered
fi

# [ ]
echo "IP[ ]"

```

???

```
sudo bash ufw_deny.sh
```

1. ??????IP????????????
2. IPv4????????
3. ?????(0)/24(1)????????/16(2)????????0/1/2????
4. ??y/n????????
5. y??ufw????IP????NW????
6. ??IP????y/n????
7. ?????y/n????y??ufw????
8. n??ufw????

MySQL????????????????????

??

??

????????MySQL????????????????????????????????

????????DB????????????

????????

- ?????DB????????????????????
- ?????????????????????????
- cron????????????????????

????????

- Ubuntu 24.04
- MySQL 8.0.39

???

????????

???????? db_account.txt ?????DB????????????????

```
[client]
user = db_user
password = "password"
```

- ?????400????(chmod 400 db_account.txt)
- ?????????

????DB???

1. ?????
2. ?????

????????????????(wasabi)????????

????????

/home/hoge/dbrestore_password ?????????????????????????????????

????????????????????/??????????

?????

??????

- ????????(?)
 - mysql_db_backup.sh

????????????????????

```
#!/bin/bash

##  ##

# $HOME
HOME_DIR="/home/hoge"

# SQL( )
backup_dir="/path/to/backup/directory"

# 
keep_days=7

# / /
current_date=$(date +%Y%m%d)
backup_name="backup_mysql_${current_date}"
zip_file="backup_mysql.${current_date}.zip"

# 
credentials_file="${HOME_DIR}/script/config/db_account/db_account.txt"

# 
password_dir="${HOME_DIR}/dbrestore_password"
password_file="${password_dir}/db-restore.${current_date}.txt"

# redmine
database_name=database

# 
options="--defaults-extra-file=$credentials_file --no-tablespaces --single-transaction"

# 
backup_file_pattern="backup_mysql.*.zip"

# 
password_file_pattern="*restore*.txt"

##  ##

##  ##
```

```

# 1. [redacted]400[redacted]
# 400[redacted]
permissions=$(stat -c "%a" "$credentials_file")
if [ "$permissions" != "400" ]; then
echo "[redacted]400[redacted]"
exit 1
fi

# 2. [redacted]
mkdir "${backup_dir}/${backup_name}"

# 3. mysqldump [redacted]
mysqldump $options -h localhost $database_name > "${backup_dir}/${backup_name}/${backup_name}.sql"

# 4. [redacted]
password=$(openssl rand -base64 12)
cd "${backup_dir}/${backup_name}"
zip -r "${backup_dir}/${zip_file}" -P "$password" .
cd -

# 5. [redacted]
rm -rf "${backup_dir}/${backup_name}"

# 6. [redacted]
echo $password > $password_file

# 7. [redacted]600[redacted]
chmod 600 $password_file

# 8. [redacted]
find "$backup_dir" -name "$backup_file_pattern" ! -type f -newermt "${keep_days} days ago" -delete
find "$password_dir" -name "$password_file_pattern" ! -type f -newermt "${keep_days} days ago" -delete

## [redacted] ##

```

???

```
chmod +x mysql_db_backup.sh
```

??????????

??????????

????????

```
./mysql_db_backup.sh
```

????????

1. ?????????????????mysql_dump????????????
2. ?????????????????zip????????????
3. ???
4. ???

??????DB??

```
unzip backup_mysql.yyyymmdd.zip
```

????????????????

```
db-restore.yyyymmdd.txt ?????????????????
```

???

```
unzip -P $(cat /path/to/password_file.txt) /path/to/zip_file.zip -d /path/to/output_directory
```

????????????????

cron??

????????

```
crontab -e -u hoge
```

?cron??????

```
0 2 * * * /home/hoge/script/directory/mysql_db_backup.sh
```

????????????DB????????????