

# 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
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????????

Revision #2

Created 3 April 2024 16:56:58 by manualmaton

Updated 3 April 2024 17:05:59 by manualmaton