





# Apple TV (BTV) 설정

Unifi OS 8.x로 오면서 igmpproxy라는 옵션이 제공 됩니다. 기본적으로 igmpproxy를 활성화하면 IPTV 시청이 가능해지나 현재 기능이 잘 동작하지 않아 TV는 나오지만 지속적으로 끊어지는 문제가 발생합니다. (채널 변경 시 5분마다 멈춤)

이런 문제점 때문에 일단 이 기능을 비활성화하고, igmpproxy 데몬을 이용한 방법을 설명합니다.

이름 ▾	VLAN ID ▾	라우터 ▾	서브넷 ▾	인터넷 ▾	IP 임대 ▾
● Default	1	UDM Pro	192.168.1.0/24	Primary (WAN1)	 13
● DHCloud	2	UDM Pro	192.168.10.0/24	Primary (WAN1)	0
● Cam	3	UDM Pro	192.168.20.0/24	Primary (WAN1)	 1
● Room	4	UDM Pro	192.168.30.0/24	Primary (WAN1)	 2
● IPTV	5	UDM Pro	192.168.33.0/24	Primary (WAN1)	 3
<a href="#">새로운 가상 네트워크</a> <a href="#">VLAN 뷰어</a>					<a href="#">관리</a>

## 글로벌 네트워크 설정

IPv6 지원



멀티캐스트 DNS ⓘ



Default × DHCloud × Cam × Room ×  
IPTV ×

5개 항목 선택됨 [편집](#)

IGMP Snooping ⓘ



IPTV ×

1개 항목 선택됨 [편집](#)

IGMP 프록시 ⓘ



## 글로벌 스위치 설정

DHCP 스누핑 ⓘ



스패닝 트리 프로토콜 ⓘ



STP



RSTP



사용 안 함

정보 프레임 ⓘ



흐름 제어 ⓘ



802.1X 제어 ⓘ



스위치 제외 ⓘ



[장치 선택](#)

## 1. VLAN 생성

- "Network > 설정 > 네트워크"에서 "새로운 가상 네트워크"를 누릅니다. 네트워크 이름을 지정하고 사용할 IP 대역 정보를 넣어줍니다. 밑부분 IGMP Snooping, 멀티캐스트 DNS는 체크해 줍니다.

네트워크 이름: IPTV

라우터: UDM Pro

게이트웨이 IP/서브넷: 192.168.33.1, 24

게이트웨이 IP	브로드캐스트 IP	사용 가능한 IP 범위	서브넷 마스크
192.168.33.1	192.168.33.255	192.168.33.6 - 192.168.33.254	255.255.255.0

고급: 자동, 수동

VLAN ID: 5

격리: ☐ 네트워크

콘텐츠 필터링: ☒ 없음, ☐ 작업, ☐ 패밀리

☒ IGMP Snooping

☒ 멀티캐스트 DNS

## 2. 멀티캐스트 DNS 및 IGMP Snooping 설정 확인

- VLAN ID (5)에 IPTV가 생성 됐음을 확인하고, 아래와 같이 설정을 합니다.

이름 ▾	VLAN ID ▾	라우터 ▾	서브넷 ▾	인터넷 ▾	IP 임대 ▾
● Default	1	UDM Pro	192.168.1.0/24	Primary (WAN1)	13
● DHCPcloud	2	UDM Pro	192.168.10.0/24	Primary (WAN1)	1
● Cam	3	UDM Pro	192.168.20.0/24	Primary (WAN1)	1
● Room	4	UDM Pro	192.168.30.0/24	Primary (WAN1)	2
● IPTV	5	UDM Pro	192.168.33.0/24	Primary (WAN1)	0

새로운 가상 네트워크 VLAN 뷰어 관리

#### 글로벌 네트워크 설정

IPv6 지원 ☒

멀티캐스트 DNS ☒ Default × DHCPcloud × Cam × Room × **IPTV ×**  
5개 항목 선택됨 편집

IGMP Snooping ☒ Default × **IPTV ×**  
2개 항목 선택됨 편집

IGMP 프록시 ☐

#### 글로벌 스위치 설정

DHCP 스누핑 ☒

스패닝 트리 프로토콜 ☐ STP ☒ RSTP ☐ 사용 안 함

정보 프레임 ☐

흐름 제어 ☐

802.1X 제어 ☐


스위치 제외 장치 선택




### 3. 방화벽 규칙 설정 #1 (IPTV Multicast 허용)

"네트워크 > 보안 > 방화벽 규칙"으로 이동하여 항목 만들기를 눌러줍니다.




- 유형 : 인너넷 인
- 이름 : Allow IPTV Multicast
- 작업 : 수락
- 프로토콜 : TCP 및 UDP
- 소스 : "주소그룹" 옆 신규를 누른 후 프로필 이름을 "BTV"로 넣고 IPv4 주소.서브넷에 "192.168.0.0/16"를 넣고 만들기를 눌러줍니다. "주소그룹"에서 BTV를 선택 합니다.
- 목적지 : "주소그룹" 옆 신규를 누른 후 프로필 이름을 "IGMP"로 넣고 IPv4 주소.서브넷에 "224.0.0.0/4"를 넣고 만들기를 눌러줍니다. "주소그룹"에서 IGMP를 선택 합니다.
- 고급 : 일치상태는 전부 체크하고, IPSec은 "일치하지 않음"을 선택 후 "규칙 추가"를 눌러 줍니다.

## Allow IPTV Multicast

 트래픽 규칙은(는) 네트워크와 클라이언트 간의 트래픽을 차단하거나 허용하는 보다 효율적이고 직관적인 방법입니다.

유형 	<input type="text" value="인터넷 인"/>
이름	<input type="text" value="Allow IPTV Multicast"/>
작업 	<input checked="" type="button" value="수락"/> <input type="button" value="거부"/> <input type="button" value="드롭"/>
프로토콜	<input type="text" value="TCP 및 UDP"/> <input checked="" type="checkbox"/> 사전 정의 이전 
소스	
소스 유형	<input type="text" value="포트/IP 그룹"/>
주소 그룹	<input type="text" value="BTV"/> <span>신규</span>
포트 그룹	<input type="text" value="임의"/> <span>신규</span>
MAC 주소	<input type="text" value="MAC 주소 입력"/>
목적지	
목적지 유형	<input type="text" value="포트/IP 그룹"/>
주소 그룹	<input type="text" value="IGMP"/> <span>신규</span>
포트 그룹	<input type="text" value="임의"/> <span>신규</span>
고급	<input type="button" value="자동"/> <input checked="" type="button" value="수동"/>

---

일치 상태 	<input checked="" type="checkbox"/> 신규 <input checked="" type="checkbox"/> 유효하지 않음 <input checked="" type="checkbox"/> 확정 <input checked="" type="checkbox"/> 관련
IPSec 일치 	<input checked="" type="radio"/> 일치하지 않음 <input type="radio"/> IPsec <input type="radio"/> 비 IPsec
로깅 	<input type="checkbox"/>

## 4. 방화벽 규칙 설정 #2 (IGMP Traffic 허용)

다시 "네트워크 > 보안 > 방화벽 규칙"으로 이동하여 항목 만들기를 눌러줍니다.

- 유형 : 인터넷 로컬
- 이름 : Allow IGMP Traffic
- 작업 : 수락
- 프로토콜 : "사전 정의 이전" 체크를 풀고, IGMP를 선택해 줍니다.
- 소스 : 아무것도 건들지 않음
- 목적지 : 아무것도 건들지 않음

- 고급 : 일치상태는 전부 체크하고, IPSec은 "일치하지 않음"을 선택 후 "규칙 추가"를 눌러 줍니다.

**Allow IGMP Traffic**

---

**트래픽 규칙**은(는) 네트워크와 클라이언트 간의 트래픽을 차단하거나 허용하는 보다 효율적이고 직관적인 방법입니다.

**유형** 인터넷 로컬

**이름** Allow IGMP Traffic

**작업** ✓ 수락 거부 드롭

**프로토콜** IGMP 사전 정의 이전

**반대 일치** ☐

**소스**

**소스 유형** 포트/IP 그룹

**주소 그룹** 임의 신규

**포트 그룹** 임의 신규

**MAC 주소** MAC 주소 입력

**목적지**

**목적지 유형** 포트/IP 그룹

**주소 그룹** 임의 신규

**포트 그룹** 임의 신규

**고급** 자동 수동

---

**일치 상태** ☒ 신규 ☒ 유효하지 않음 ☒ 확정 ☒ 관련

**IPSec 일치** ☒ 일치하지 않음 ☐ IPsec ☐ 비 IPsec

**로깅** ☐

## 5. igmpproxy 데몬 설치

UDM Pro에 ssh 접속을 해야 합니다. ssh 접속은 UniFi OS의 "Console Settings"에서 Advanced에 있는 SSH를 체크 후 비밀번호를 생성해 주면 접속할 수 있습니다.

- 1) 터미널 프로그램에서 호스트 192.168.1.1 (포트는 기본 22), id는 root, password는 위에서 입력한 값을 넣어 ssh 접속을 합니다.
- 2) igmpproxy를 설치해 줍니다.

```
cd /mnt
mkdir data
mkdir data/igmpproxy
cd /mnt/data/igmpproxy
curl -Lo igmpproxy https://raw.githubusercontent.com/peacey/udm-telus/main/igmpproxy
curl -Lo igmpproxy.conf https://raw.githubusercontent.com/peacey/udm-telus/main/igmpproxy.conf
chmod +x igmpproxy
```

- 3) 먼저 각 포트별 값 확인을 위해 아래 명령을 실행 합니다.

```
ifconfig
```

- 여기서 brXX가 VLAN 식별자이고, ethXX는 LAN와 WAN 포트 번호 입니다.
- upstream과 downstream 포트를 찾아야 하는데 합니다.
- upstream의 경우 RJ45 WAN이면 eh8이고 SFP WAN이면 eh9 입니다.
- downstream의 경우 brXX 쪽에서 부여하신 IP 범위 값으로 해당 포트 값을 찾아 줍니다. 제 경우 192.168.33.1 이므로 br5가 해당 포트임을 확인 할 수 있습니다.

- 4) igmpproxy.conf를 수정해 줍니다. (손쉬운 수정을 위해 nano를 sudo apt install nano 명령어로 설치해 줍니다. 설치된 nano는 재부팅하면 자동 삭제 되며, 일회성으로만 사용 가능합니다.)

```
nano /mnt/data/igmpproxy/igmpproxy.conf
```

- nano를 설치하지 않을 시 vi로 수정하시면 됩니다. (사용 방법은 생략)
- upstream 부분을 **eh8**로 수정 합니다.
- downstream 부분을 **br5**로 수정합니다.
- 작성 후 ctrl + x를 눌러 빠져나와 y를 눌러 저장해 줍니다.
- nano를 사용하지 않고 기본 vi를 사용할 경우 "vi /mnt/data/igmpproxy/igmpproxy.conf" 실행 후 ESC를 눌러 명령 모드로 진입 > i를 눌러 수정 모드에서 내용을 수정 후 다시 ESC를 누른 후 :wq로 저장 후 종료하시면 됩니다.
- 작성된 예제는 아래와 같습니다.

```
quickleave

# upstream = modem interface
phyint eth8 upstream ratelimit 0 threshold 1
    altnet 0.0.0.0/0;

# lan interface of iptv device
phyint br5 downstream ratelimit 0 threshold 1
    altnet 0.0.0.0/0;

# disable all unused interfaces
phyint lo disabled
phyint eth9 disabled
```

- 작성한 문장이 잘 동작되는지 확인 합니다. 정상 작동을 확인하고 Ctrl + c로 종료 합니다.

```
./igmpproxy -ndv ./igmpproxy.conf
```

## 6. igmpproxy 데몬 실행

```
./igmpproxy ./igmpproxy.conf
```

- 데몬을 중지 시키는 방법은 아래와 같습니다.
  - 1) ps aux로 igmp.conf의 PID 값을 확인 합니다. (PID 값을 알고 있다면 ps aux | grep PID로 확인)
  - 2) kill -9 PID로 중지시킵니다. (-9는 즉시 중지 옵션입니다.)

## 7. Apple TV 네트워크 VLAN 설정

- "Network > 포트"에서 Apple TV를 찾아 들어갑니다.

포트		이름	작업	PoE 모드	속도	연결	프로필
1	⚡	USW-Lite-8-PoE	Port 1	PoE+	GbE	U6-Pro	-
2	⬆	USW-Lite-8-PoE	Port 2	PoE+	GbE	USW-Enterprise-8-PoE	-
3	■	USW-Lite-8-PoE	Port 3	꺼짐	GbE	AppleTV	-
4	⚡	USW-Lite-8-PoE	Port 4	PoE+	FE	G3 Flex	-
5	■	USW-Lite-8-PoE	Port 5	-	GbE	HPE iLO	-
8	■	USW-Lite-8-PoE	Port 8	-	GbE	ROCK	-

- 기본 VLAN/네트워크를 IPTV로 변경해 줍니다.



### 포트 3

이름	Port 3
포트	<input checked="" type="radio"/> 활성화 <input type="radio"/> 사용 안 함 <input type="radio"/> MAC ID로 제한됨
기본 VLAN/네트워크 ⓘ	<div style="border: 2px solid red; padding: 2px;"> <input style="width: 150px;" type="text" value="IPTV (5)"/> <span>192.168.40.0/24 ▼</span> </div>
태그가 지정된 VLAN 관리 ⓘ	<input checked="" type="radio"/> 모두 허용 <input type="radio"/> 모두 차단 <input type="radio"/> 사용자 지정
PoE	<input checked="" type="radio"/> 꺼짐 <input type="radio"/> PoE+
고급	<div> <div>자동</div> <div style="background-color: #007bff; color: white; padding: 2px 10px; border-radius: 4px;">수동</div> </div>

- Apple TV를 재기동하면 IPTV VLAN에서 DHCP IP를 받아 옵니다.

## 8. 자동 시작 스크립트

UDM Pro를 재부팅 시 igmpoxy 자동 시작 스트립트는 아래와 같습니다.

먼저 UDM Boot Script를 설치해야 합니다. (<https://github.com/unifi-utilities/unifios-utilities>)

1. 1.6.3 이상



```
curl -fsL "https://raw.githubusercontent.com/unifi-utilities/unifios-utilities/HEAD/on-boot-script/remote_install.sh" | /bin/sh
```

## 2.2.4.x 이상

```
curl -fsL "https://raw.githubusercontent.com/unifi-utilities/unifios-utilities/HEAD/on-boot-script-2.x/remote_install.sh" | /bin/bash
```

설치를 하고 `"/data/on_boot.d"`로 이동 후 자동시작 스크립트를 작성하시면 됩니다.

```
cd /data/on_boot.d
curl -Lo 99-run-igmpproxy.sh https://raw.githubusercontent.com/peacey/udm-telus/main/run-igmpproxy.sh
chmod +x 99-run-igmpproxy.sh
```

```
root@UDMPro:/mnt/data# curl -fsL "https://raw.githubusercontent.com/unifi-utilities/unifios-utilities/HEAD/on-boot-script/remote_install.sh" | /bin/sh
UDM [EX-01]
Execute any script when your udm system starts.

UniFi Dream Machine Pro version 3.2.9 was detected
Installing on-boot script...
Failed to disable unit: Unit file udm-boot.service does not exist.
Creating systemd service file
Enabling UDM boot...
Created symlink /etc/systemd/system/multi-user.target.wants/udm-boot.service → /etc/systemd/system/udm-boot.service.
UDM Boot Script installed

Downloading CNI plugins script...
CNI plugins script installed
Executing CNI plugins script...
Downloading https://github.com/containernetworking/plugins/releases/download/v1.4.0/cni-plugins-linux-arm64-v1.4.0.tgz.sha256
Downloading https://github.com/containernetworking/plugins/releases/download/v1.4.0/cni-plugins-linux-arm64-v1.4.0.tgz
Pouring /data/.cache/cni-plugins/cni-plugins-linux-arm64-v1.4.0.tgz

Downloading CNI bridge script...
CNI bridge script installed
Executing CNI bridge script...
/data/on_boot.d/06-cni-bridge.sh

On boot script installation finished

You can now place your scripts in '/data/on_boot.d'

root@UDMPro:/mnt/data# ls
igmpproxy/
root@UDMPro:/mnt/data# cd ..
root@UDMPro:/mnt# ls
data/
root@UDMPro:/mnt# cd ..
root@UDMPro:/# ls
2 bin/ boot/ data/ dev/ etc/ home/ lib/ lib64@ media/ mnt/ opt/ persistent/ proc/ root/ run//sbin/ srv@ sys/ tmp/ usr/ var/ volu
root@UDMPro:/# cd /
root@UDMPro:/# cd data
root@UDMPro:/data# ls
dpi-tracer/ on_boot.d/ podman/ postgresql/ udapi-config/ ui-db/ uid/ ulp-go/ unifi/ unifi-core/ unifi-innerspace/ unifi-protect/ wifiman/
root@UDMPro:/data# cd on_boot.d/
root@UDMPro:/data/on_boot.d# ls
05-install-cni-plugins.sh* 06-cni-bridge.sh*
root@UDMPro:/data/on_boot.d# curl -Lo 99-run-igmpproxy.sh https://raw.githubusercontent.com/peacey/udm-telus/main/run-igmpproxy.sh
% Total % Received % Xferd Average Speed Time Time Time Current
Dload Upload Total Spent Left Speed
100 75 100 75 0 0 204 0 --:--:-- --:--:-- --:--:-- 204
root@UDMPro:/data/on_boot.d# ls
05-install-cni-plugins.sh* 06-cni-bridge.sh* 99-run-igmpproxy.sh
root@UDMPro:/data/on_boot.d# chmod +x 99-run-igmpproxy.sh
root@UDMPro:/data/on_boot.d# ls
05-install-cni-plugins.sh* 06-cni-bridge.sh* 99-run-igmpproxy.sh*
root@UDMPro:/data/on_boot.d#
```

UnifiOS가 4.x로 올라가면서 더이상 위에 기술된 자동시작 스크립트가 동작하지 않습니다.

어쩔수 없이 다른 서버에서 Unifi의 상태값을 체크해 igmpproxy가 시행되지 않았을 시 시작시켜주는 스크립트를 작성하였습니다.

## 1. 가정

- Unifi 게이트웨이
  - 1) IP : 192.168.1.1
  - 2) ID : root
  - 3) PW : pass@me
- 명령을 내릴 서버
  - 1) IP : 192.168.10.200
  - 2) ID : myadmin
  - 3) PW : pass@me

2. "192.168.10.200"에서 "192.168.1.1"로 ssh명령을 내리기 위해서는 ssh 키생성 및 복제가 필요합니다. "192.168.10.200"에 공개 ssh 키를 생성합니다.

```
ssh-keygen -t rsa -b 2048
```

제 경우 ssh 키 생성 시 그냥 엔터를 눌러 공개키를 생성 했습니다.

이제 이 공개 키 값을 복사해야 합니다.

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

화면에 나오는 값을 복사해 줍니다.

이제 원격지에 복사된 ssh 키 값을 복사해줄 차례입니다.

```
ssh root@192.168.1.1
```

"root"는 예제이며 사용하는 ID를 넣고 접속하시고, "~/.ssh"로 이동합니다.

```
cd ~/.ssh
```

해당 디렉토리가 없을 경우 아래와 같이 생성해 주시면 됩니다.

```
mkdir -p ~/.ssh  
chmod 700 ~/.ssh
```

이제 공개 키를 추가해 줍니다.

```
echo "복사한 공개 키" >> ~/.ssh/authorized_keys  
chmod 600 ~/.ssh/authorized_keys
```

위 단계를 거치면 "192.168.10.200"에서 "192.168.1.1"로 비밀번호 없이 ssh 접속이 가능해 집니다.

"192.168.10.200" 서버에 아래 sh 스크립트를 만들고 저장해 줍니다.

```
#!/bin/bash

UNIFI_SERVER="192.168.1.1"
UNIFI_USER="root"

CHECK_PROCESS_CMD="sudo ps aux | grep 'igmpproxy.conf' | grep -v grep"
START_PROCESS_CMD="sudo nohup /mnt/data/igmpproxy/igmpproxy /mnt/data/igmpproxy/igmpproxy.conf
&"

ssh ${UNIFI_USER}@${UNIFI_SERVER} << EOF

if ${CHECK_PROCESS_CMD}; then
    echo "igmpproxy.conf 프로세스가 이미 실행 중입니다. 명령을 스킵합니다."
else
    echo "igmpproxy.conf 프로세스가 실행되지 않았습니다. 실행합니다."
    ${START_PROCESS_CMD}
fi

EOF
```

제 경우 시놀로지와 UnRAID에서 이 스크립트를 자동 스케줄러로 등록해 실행해 주었습니다.

### 1. igmpproxy가 이미 실행되고 있을때

```
Script igmpproxy Running

Script location: /tmp/user.scripts/tmpScripts/igmpproxy/script
Note that closing this window will abort the execution of this script

Linux UDMPro 4.19.152-ui-alpine #4.19.152 SMP Fri Jun 7 23:26:16 CST 2024 aarch64

Firmware version: v4.0.6

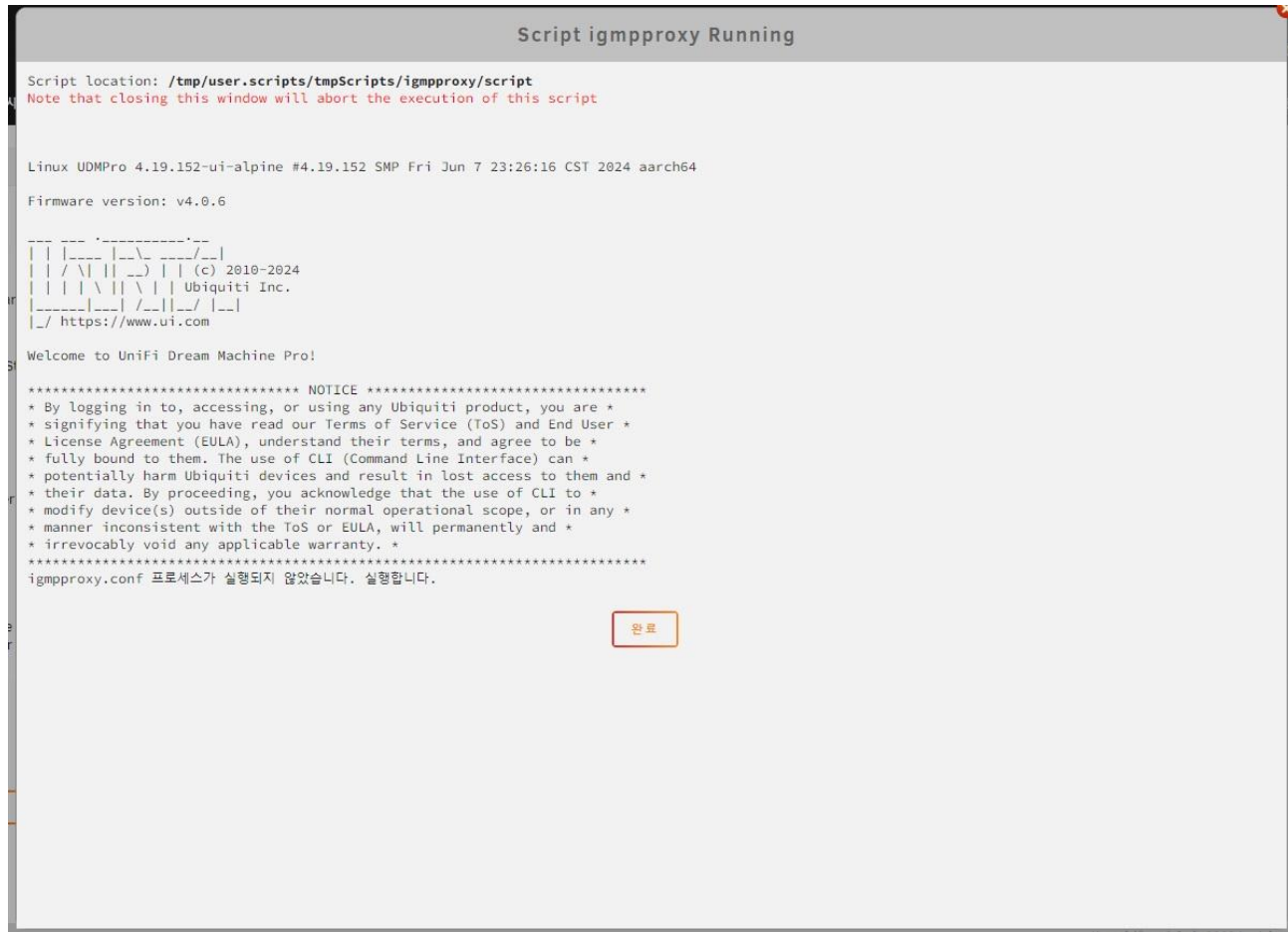
| | | _ _ _ _ _ |
| | | _ _ _ _ _ | _ _ \ _ _ _ _ _ |
| | | / \ | | _ _ ) | | (c) 2010-2024
| | | | \ | | \ | | Ubiquiti Inc.
| _ _ _ _ _ | _ _ \ _ _ _ _ _ |
| _ / https://www.ui.com

Welcome to UniFi Dream Machine Pro!

***** NOTICE *****
* By logging in to, accessing, or using any Ubiquiti product, you are *
* signifying that you have read our Terms of Service (ToS) and End User *
* License Agreement (EULA), understand their terms, and agree to be *
* fully bound to them. The use of CLI (Command Line Interface) can *
* potentially harm Ubiquiti devices and result in lost access to them and *
* their data. By proceeding, you acknowledge that the use of CLI to *
* modify device(s) outside of their normal operational scope, or in any *
* manner inconsistent with the ToS or EULA, will permanently and *
* irrevocably void any applicable warranty. *
*****
root 2132918 0.0 0.0 1960 1332 ? S 10:20 0:00 /mnt/data/igmpproxy/igmpproxy /mnt/data/igmpproxy/igmpproxy.conf
igmpproxy.conf 프로세스가 이미 실행 중입니다. 명령을 스킵합니다.
```

완료

## 2. igmpproxy가 실행되고 있지 않을때



ps aux로 실행한 결과 입니다. 잘 실행되고 있습니다.

root	2260850	0.0	0.0	0	0 ?	I	11:30	0:00	[kworker/2:2-mm_percpu_wq]
root	2263266	0.1	0.0	0	0 ?	I	11:31	0:00	[kworker/0:4-rcu_gp]
root	2263423	0.0	0.0	0	0 ?	I	11:31	0:00	[kworker/3:2-events]
root	2267498	0.0	0.0	0	0 ?	I	11:34	0:00	[kworker/1:0-mm_percpu_wq]
root	2267830	0.0	0.0	35088	728 ?	S<I	11:34	0:00	/usr/bin/dpinger -f -i eth8-mon6-142.250.206.238-google.com -B 211.204.2
root	2271837	0.7	0.0	0	0 ?	I	11:36	0:01	[kworker/u8:4-events_unbound]
root	2272623	0.0	0.0	0	0 ?	I	11:36	0:00	[kworker/2:1]
root	2272806	0.1	0.0	0	0 ?	D	11:36	0:00	[kworker/0:1+events]
postgres	2273001	0.0	0.5	212312	22248 ?	Ss	11:37	0:00	postgres: 14/protect: unifi-protect unifi-protect [local] idle
root	2273215	0.0	0.0	0	0 ?	I	11:37	0:00	[kworker/3:1-mm_percpu_wq]
root	2273533	0.5	0.0	0	0 ?	I	11:37	0:00	[kworker/u8:5-flush-135:0]
root	2273693	0.0	0.1	15256	7456 ?	Rs	11:37	0:00	sshd: root@pts/0
root	2273912	0.0	0.1	14928	7244 ?	Ss	11:37	0:00	sshd: root@notty
root	2273975	0.0	0.0	5356	3992 ?	Ss	11:37	0:00	/usr/lib/openssh/sftp-server
root	2274004	0.0	0.0	5832	3580 pts/0	Ss	11:37	0:00	-bash
postgres	2274867	0.0	0.3	211624	12308 ?	Ss	11:38	0:00	postgres: 14/main: postgres unifi-innerspace :1(55746) idle
root	2275143	0.0	0.0	1960	96 ?	S	11:38	0:00	/mnt/data/igmp/igmp/igmp /mnt/data/igmp/igmp/igmp.igmp.conf
postgres	2275403	0.0	0.3	211492	14920 ?	Ss	11:38	0:00	postgres: 14/protect: unifi-protect unifi-protect [local] idle
unifi	2275557	0.0	0.0	1952	436 ?	S	11:38	0:00	sleep 120
root	2275992	0.0	0.0	35088	732 ?	S<I	11:38	0:00	/usr/bin/dpinger -f -i eth8-mon5-23.53.37.224-www.microsoft.com -B 211.2
root	2276102	0.0	0.0	8328	2876 pts/0	R+	11:38	0:00	ps aux
postgres	2584811	0.0	0.3	212336	12328 ?	Ss	Jul28	0:00	postgres: 14/main: unifi-core unifi-core 127.0.0.1(44058) idle

🔄버전 #31

★생성 15 1월 2024 05:15:27, Admin

✎수정 3 12월 2024 07:28:13, Admin