



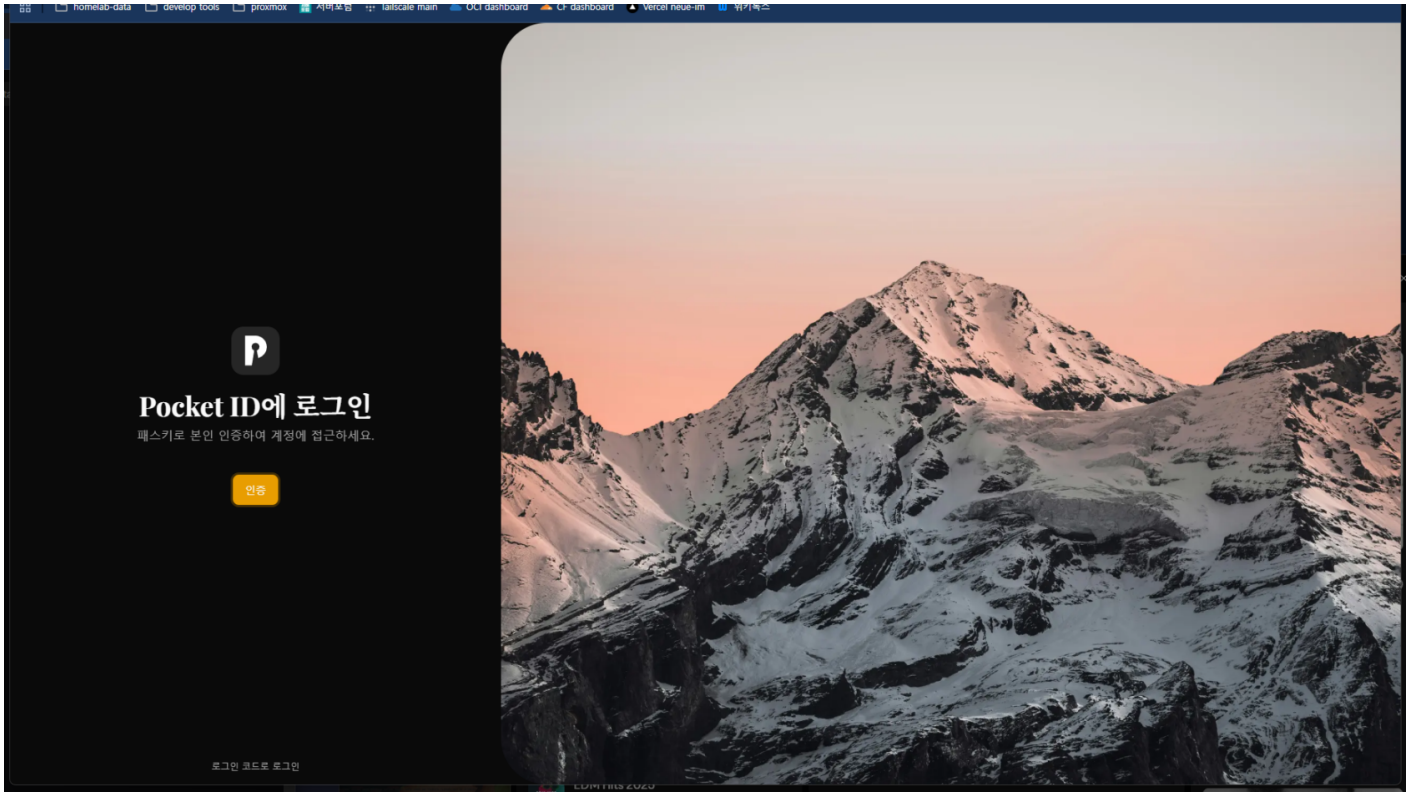
□□□ □□□□ □□□□□ , self-hosted □ □ □□

- [pocket-id](#)
- [traefik](#)
- [pangolin](#)
- [Obsidian](#)
- [uptime kuma](#)

# pocket-id

이 프로젝트는 OIDC (open id connect) provider를 제공하는 passkey 기반의 인증 솔루션입니다. 이 프로젝트는 오픈 소스이며, MIT 라이선스에 따라 배포됩니다.

- site : <https://pocket-id.org/>
- git : <https://github.com/pocket-id/pocket-id>



이 프로젝트는 passkey 기반의 인증 솔루션을 제공하는 오픈 소스 인증 솔루션입니다. 이 프로젝트는 오픈 소스이며, MIT 라이선스에 따라 배포됩니다.



```

volumes:
  - "./data:/app/data"
# Optional healthcheck
labels:
  - "traefik.enable=true"
  - "traefik.http.routers.pocket-id.rule=Host(`YOUR_OIDC_DOMAIN`)"
  - "traefik.http.routers.pocket-id.entrypoints=websecure"
  - "traefik.http.routers.pocket-id.tls=true"
  - "traefik.http.routers.pocket-id.tls.certresolver=letsencrypt"
  - "traefik.http.services.pocket-id.loadbalancer.server.port=1411"
networks:
  - proxy
healthcheck:
  test: [ "CMD", "/app/pocket-id", "healthcheck" ]
  interval: 1m30s
  timeout: 5s
  retries: 2
  start_period: 10s

networks:
  proxy:
    external: true

```

.env `TRUST_PROXY` `MAXMIND_LICENSE_KEY` `PGID` `PUID` `ENCRYPTION_KEY` `APP_URL` .

```

# See the documentation for more information: https://pocket-
id.org/docs/configuration/environment-variables

# These variables must be configured for your deployment:
APP_URL=https://auth.your.domain

# Encryption key (choose one method):
# Method 1: Direct key (simple but less secure)
# Generate with: openssl rand -base64 32
ENCRYPTION_KEY=...
# Method 2: File-based key (recommended)
# Put the base64 key in a file and point to it here.
# ENCRYPTION_KEY_FILE=/path/to/encryption_key

# These variables are optional but recommended to review:
TRUST_PROXY=true
MAXMIND_LICENSE_KEY=...
PUID=1000
PGID=1000

```

`TRUST_PROXY` `MAXMIND_LICENSE_KEY` `PGID` `PUID` `ENCRYPTION_KEY` `APP_URL` `docker` `audit log` `IP` `ip` .

`audit log` `IP` `ip` , pocket-id docs <https://pocket-id.org/docs/client-examples>



# pangolin

UI + traefik + newt VPN . cloudflare pangolin

- VPN
- SSO , OIDC provider
- CrowdSec

pangolin traefik wrapper traefik + authentic UI

Traefik

- traefik k8s, docker label dynamic config (file provider)
- pangolin / config , (traefik dynamic config ) UI

VM dynamic config UI

newt vpn tailscale

UI

- 
- IdP
- API

homelab supporter key

pangolin



# Obsidian

Obsidian은 markdown 파일을 기반으로 한 개인 지식 관리 도구입니다. . Obsidian은 다양한 플랫폼에서 사용할 수 있습니다.

- Obsidian은 markdown, cross platform
- Obsidian (Obsidian sync, publish 등)
- [Obsidian Sync](#) - Obsidian을 클라우드에 동기화하는 서비스
- Obsidian은 다양한 플랫폼에서 사용할 수 있습니다.
- Obsidian은 다양한 플랫폼에서 사용할 수 있습니다. (4\$)

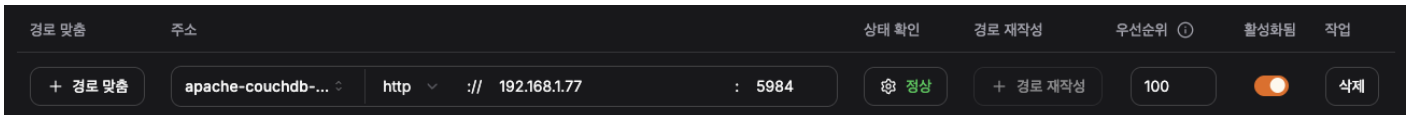
Obsidian은 다양한 플랫폼에서 사용할 수 있습니다. . Obsidian은 다양한 플랫폼에서 사용할 수 있습니다.

Obsidian은 self-hosted로 사용할 수 있습니다. . [self-hosted livesync](#) db, apache couchDB 등 다양한 데이터베이스를 사용할 수 있습니다.

## 설치

couchDB LXC (on proxmox). proxmox script [couchDB LXC](#) (IP 127.0.0.1:5984)

pangolin reverse proxy ( <https://couchdb-sample.your.domain> -> 127.0.0.1:5984)



Obsidian은 다양한 플랫폼에서 사용할 수 있습니다. . Obsidian은 다양한 플랫폼에서 사용할 수 있습니다. healthcheck url (/up)

### 상태 확인 설정

192.168.1.77:5984에 대한 상태 모니터링 설정

**상태 확인 활성화**  
 이 대상을 모니터링하여 건강 상태를 확인하세요. 필요에 따라 대상과 다른 엔드포인트를 모니터링할 수 있습니다.

방법	IP / 호스트	포트	경로
HTTP	192.168.1.77	5984	/_up

HTTP 메소드

livesync plugin DB require\_valid\_user: true healthcheck url require\_valid\_user\_except\_for\_up: true . health check url

chttpd	bind_address	0.0.0.0	
	enable_cors	true	
	max_http_request_size	4294967296	
	port	5984	
	require_valid_user	true	
	require_valid_user_except_for_up	true	

json http login dialog

[https://www.reddit.com/r/selfhosted/comments/1eo7knj/guide\\_obsidian\\_with\\_free\\_selfhosted\\_instance\\_sync/?tl=ko](https://www.reddit.com/r/selfhosted/comments/1eo7knj/guide_obsidian_with_free_selfhosted_instance_sync/?tl=ko) db obsidian app

# uptime kuma

Uptime Kuma is a self-hosted monitoring tool. It is a Prometheus + Grafana dashboard for monitoring uptime of various services. It is a simple and easy-to-use monitoring tool. It is a self-hosted monitoring tool. It is a Prometheus + Grafana dashboard for monitoring uptime of various services. It is a simple and easy-to-use monitoring tool. It is a self-hosted monitoring tool. It is a Prometheus + Grafana dashboard for monitoring uptime of various services. It is a simple and easy-to-use monitoring tool.

<https://github.com/louislam/uptime-kuma>

- It is a self-hosted monitoring tool. It is a Prometheus + Grafana dashboard for monitoring uptime of various services. It is a simple and easy-to-use monitoring tool.
- It is a self-hosted monitoring tool. It is a Prometheus + Grafana dashboard for monitoring uptime of various services. It is a simple and easy-to-use monitoring tool.
- It is a self-hosted monitoring tool. It is a Prometheus + Grafana dashboard for monitoring uptime of various services. It is a simple and easy-to-use monitoring tool.
- It is a self-hosted monitoring tool. It is a Prometheus + Grafana dashboard for monitoring uptime of various services. It is a simple and easy-to-use monitoring tool.
- It is a self-hosted monitoring tool. It is a Prometheus + Grafana dashboard for monitoring uptime of various services. It is a simple and easy-to-use monitoring tool.
- It is a self-hosted monitoring tool. It is a Prometheus + Grafana dashboard for monitoring uptime of various services. It is a simple and easy-to-use monitoring tool.
- It is a self-hosted monitoring tool. It is a Prometheus + Grafana dashboard for monitoring uptime of various services. It is a simple and easy-to-use monitoring tool.

Uptime Kuma can be installed using Docker Compose. (Uptime Kuma uses Traefik as a reverse proxy.)

```
services:
  uptime-kuma:
    container_name: uptime-kuma
    image: louislam/uptime-kuma:2
    restart: unless-stopped
    volumes:
      - ./data:/app/data
    ports:
      - "3001:3001"
    environment:
      - PUID=1000
      - PGID=1000
    #labels:
    # - "traefik.enable=true"
    # - "traefik.http.routers.uptimekuma.rule=Host(`monitor.neue.nz`)"
    # - "traefik.http.routers.uptimekuma.entrypoints=websecure"
    # - "traefik.http.routers.uptimekuma.tls=true"
    # - "traefik.http.routers.uptimekuma.tls.certresolver=letsencrypt"
    # - "traefik.http.services.uptimekuma.loadbalancer.server.port=3001"
    #networks:
    # - proxy

#networks:
# proxy:
#   external: true
```

**Uptime Kuma** 상태 페이지 대시보드

**요약**

[+ 새 모니터 추가](#)

선택 검색... 상태 활성 태그

온라인	오프라인	점검	알 수 없음	정지
14	0	0	0	2

[모든 이벤트 지우기](#)

이름	상태	날짜	메시지
immich (photo)	온라인	2026-01-11 04:55:33	200 - OK
external	온라인	2026-01-11 04:54:39	All children up and running
ai (ollama-openwebui)	온라인	2026-01-11 04:54:38	200 - OK
external	오프라인	2026-01-11 04:30:39	Child inaccessible
ai (ollama-openwebui)	오프라인	2026-01-11 04:30:38	Request failed with status code 503
immich (photo)	오프라인	2026-01-11 04:30:33	Request failed with status code 503
external	온라인	2026-01-10 04:54:38	All children up and running
ai (ollama-openwebui)	온라인	2026-01-10 04:54:37	200 - OK
immich (photo)	온라인	2026-01-10 04:54:32	200 - OK
external	오프라인	2026-01-10 04:30:38	Child inaccessible
ai (ollama-openwebui)	오프라인	2026-01-10 04:30:37	Request failed with status code 503
immich (photo)	오프라인	2026-01-10 04:30:32	Request failed with status code 503
external	온라인	2026-01-09 23:37:38	All children up and running
infisical (secret-manager)	온라인	2026-01-09 23:37:36	200 - OK
forgejo (git)	온라인	2026-01-09 23:37:35	200 - OK
pocket-id (auth)	온라인	2026-01-09 23:37:34	200 - OK
freshrss (reader)	온라인	2026-01-09 23:37:33	200 - OK
immich (photo)	온라인	2026-01-09 23:37:32	200 - OK
vaultwarden (passwords)	온라인	2026-01-09 23:37:31	200 - OK
karaoke (bookmark)	온라인	2026-01-09 23:37:31	200 - OK

100% external  
100% homepage (home)  
100% immich (photo)  
100% infisical (secret-manage)  
100% metube (youtube-dl)  
100% n8n (automate)  
100% pangolin (proxy)  
100% pocket-id (auth)  
100% torrent (qbittorrent)  
100% vaultwarden (passwords)  
0% terminal (termix)  
0% wikajs (wiki)

### some service is down

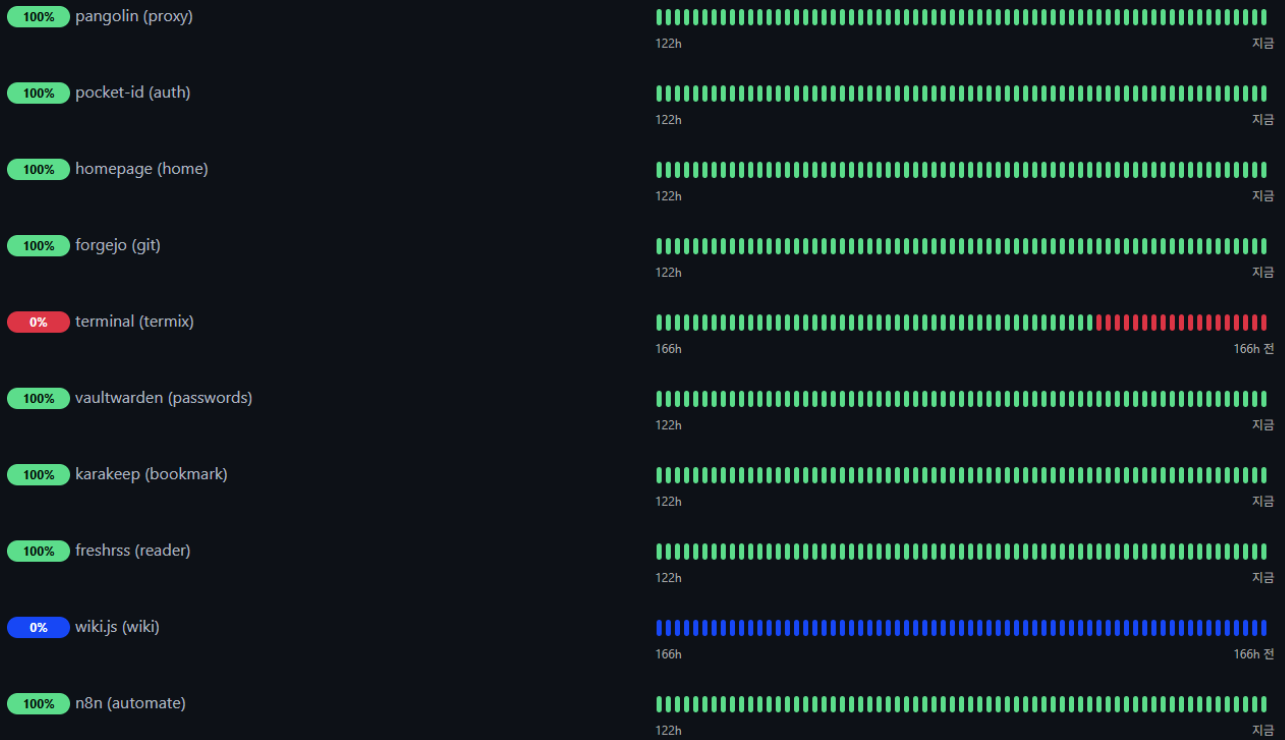
wiki, termix is down

생성된 날짜: 2026-01-09 16:36:20 (7 days ago)

점검 중

I'm not a kiwi :)

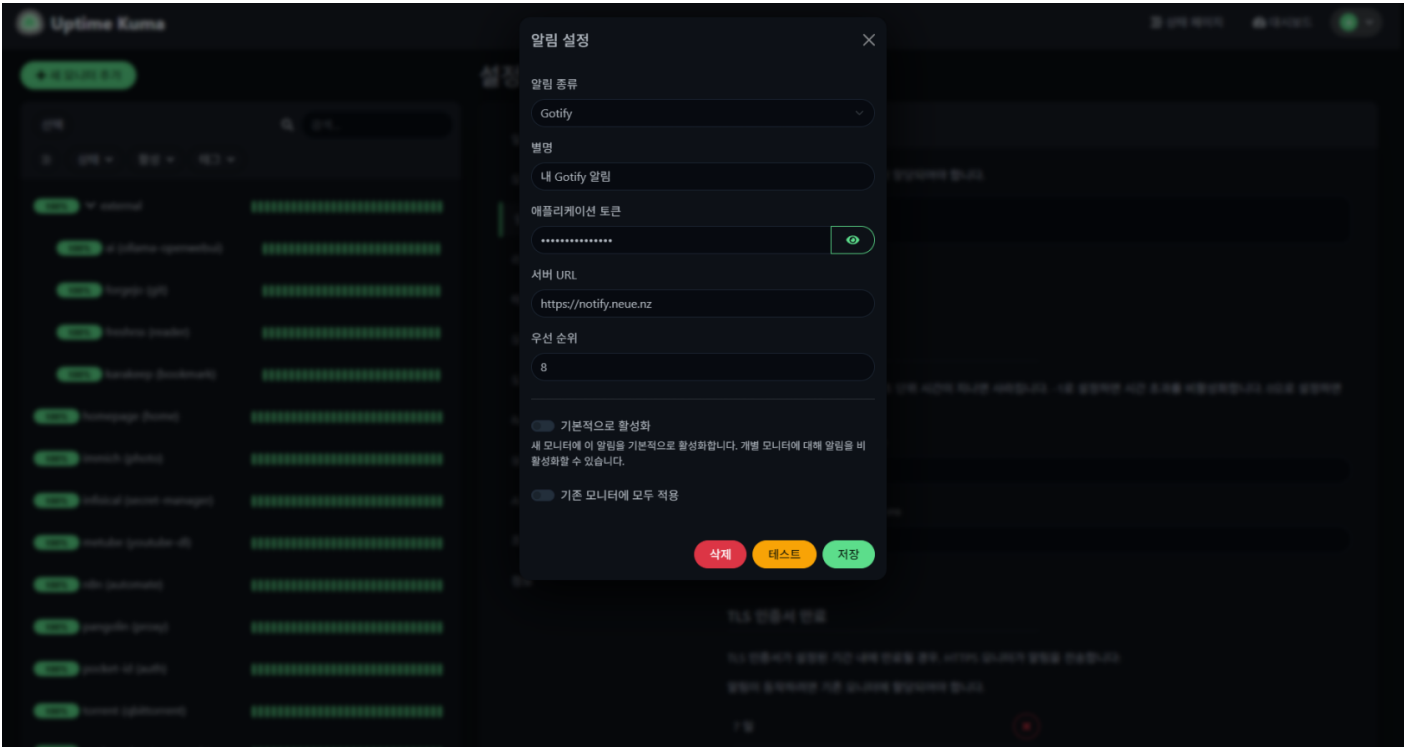
## 서비스



□□ □□□□□ □□ □□□ □□□ ( □□□ □□□□ □□ □□□□□□ traefik □□ OIDC middleware □  
□□□□ □□□□ □□□□□□ □□ ) □□□□□ □□ □□□□ status □□□□ □ □ □□ □□□ □□ □□

.





이 글에서는 Uptime Kuma의 알림 설정 방법을 소개합니다. 알림 설정은 Uptime Kuma의 중요한 기능 중 하나입니다.

- Uptime Kuma는 다양한 알림 채널을 지원합니다.
- Uptime Kuma는 Docker Compose, Config 등 다양한 방법으로 설치할 수 있습니다.
- Uptime Kuma는 Prometheus Exporter와 Grafana와도 연동할 수 있습니다.

이 글에서는 Uptime Kuma의 알림 설정 방법을 소개합니다. 알림 설정은 Uptime Kuma의 중요한 기능 중 하나입니다.