

# 목차

|                                          |   |
|------------------------------------------|---|
| <b>Web Cache / Proxy</b> .....           | 3 |
| <b>Web Interface Proxy Manager</b> ..... | 3 |
| <b>구성</b> .....                          | 3 |
| Forward .....                            | 3 |
| Reverse .....                            | 3 |
| Transparent .....                        | 4 |
| <b>솔루션 종류</b> .....                      | 4 |
| <b>참조링크</b> .....                        | 5 |



- <https://geekflare.com/open-source-load-balancer/>
- <https://giljae.medium.com/load-balancer-%EB%B9%84%EA%B5%90-c537247b699f>

## Web Cache / Proxy

웹캐시/프록시는 웹 문서 (HTML Page, 이미지 등) 를 캐시 서버에 미리 저장(Caching)하여, 원본 서버 대신 전송 해주는 기술로서 사용자 응답 속도 향상, 서비스 안정성 증대, 원본 서버의 부하 감소, 네트워크 지연 감소 및 대역폭 절감을 목적으로 한다.

- [Apache Traffic Server](#)
- [Squid](#)
- [Varnish](#)
- [HAProxy](#)
- [Nginx Proxy Manager\(NPM\)](#)

## Web Interface Proxy Manager

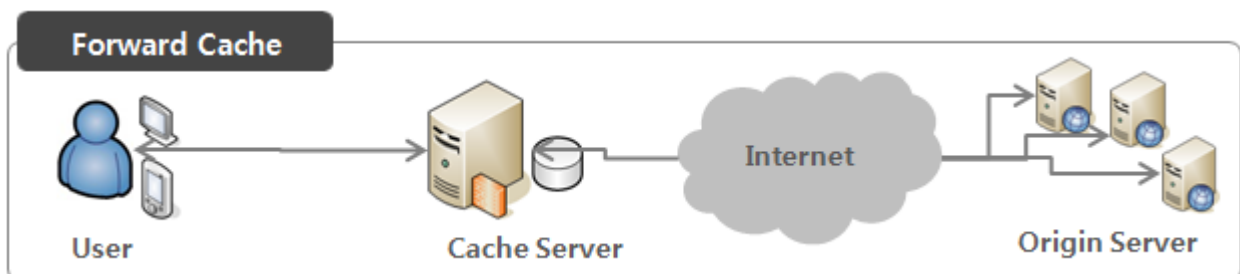
- [roxy-wi](#)
- [NginX Proxy Manager](#)
- [traefik](#)

## 구성

보통 웹 프록시 서버의 구성은 3가지 형태를 가진다.

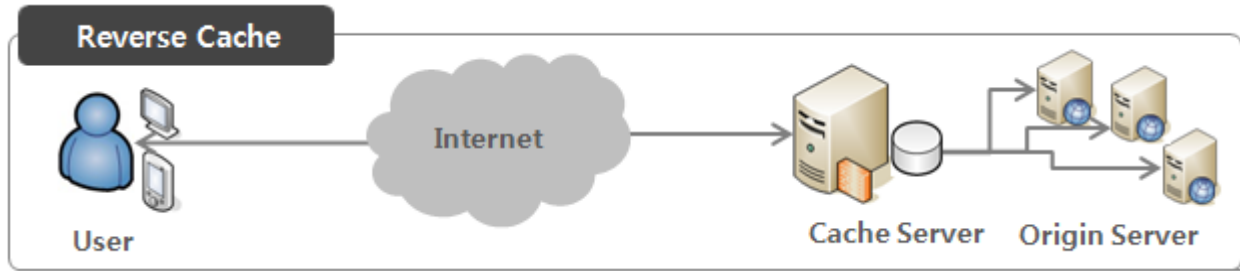
1. Forward
2. Reverse
3. Transparent

### Forward



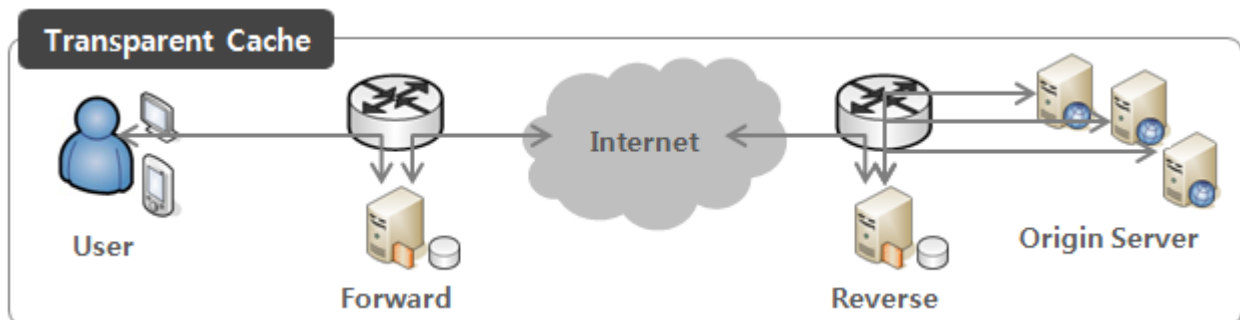
- 사용자의 인터넷 앞에 위치하여 인터넷 구간의 회선비용 절감
- 사용자의 클라이언트에서 캐시서버를 지정해야 함

### Reverse



- 원본 서버 앞에 위치하여 원본 서버 부하를 줄임
- 캐시서버가 원본 서버의 도메인을 가지고 원본 서버 성능을 높여준다고 하여 웹 가속기라고도 함

## Transparent



- 사용자는 캐시의 존재를 모름
- NW장비 연동(DPI redirection, 라우터 PBR 연동 등)으로 Forward나 Reverse 형태로 구성

## 솔루션 종류

- [Apache Traffic Server](#)
- [HAProxy](#)
- [Squid](#)
- [Varnish](#)

|                | ATS  | HAproxy | nginx | Squid | Varnish | mod_proxy |
|----------------|------|---------|-------|-------|---------|-----------|
| Worker Threads | Y    | N       | N     | N     | Y       | Y         |
| Multi-Process  | N    | Y       | Y     | N     | Y       | Y         |
| Event-driven   | Y    | Y       | Y     | Y     | N       | N?        |
| Plugin APIs    | Y    | N       | Y     | part  | Y       | Y         |
| Forward Proxy  | Y    | N       | N     | Y     | N       | Y         |
| Reverse Proxy  | Y    | Y       | Y     | Y     | Y       | Y         |
| Transp. Proxy  | Y    | Y       | N     | Y     | N       | N         |
| Load Balancer  | part | Y       | Y     | Y     | Y       | Y         |
| Cache          | Y    | N       | Y     | Y     | Y       | Y         |
| ESI            | soon | N       | N     | Y     | Y       | N         |
| ICP            | Y    | N       | N     | Y     | N       | N         |
| Keep-Alive     | Y    | N       | Y     | Y     | Y       | Y         |
| SSL            | Y    | N       | Y     | Y     | N       | Y         |

Table 4 - Comparison of Proxy Cache Solutions.

| Supported Features            | ATS | Nginx | Varnish | IIS with ARR | Squid |
|-------------------------------|-----|-------|---------|--------------|-------|
| Reverse Proxy                 | Yes | Yes   | Yes     | Yes          | Yes   |
| Forward Proxy                 | Yes | No    | No      | Yes          | Yes   |
| Transparent Proxy             | Yes | No    | No      | Yes          | Yes   |
| Plugin APIs                   | Yes | Yes   | Yes     | Yes          | Yes   |
| Cache                         | Yes | Yes   | Yes     | Yes          | Yes   |
| Edge Side Includes (ESI)      | Yes | No    | Partial | Yes          | Yes   |
| Internet Cache Protocol (ICP) | Yes | No    | No      | No           | Yes   |
| Secure Sockets Layer (SSL)    | Yes | Yes   | No      | Yes          | Yes   |
| SPDY                          | Yes | Yes   | No      | No           | No    |

## 참조링크

- <https://d2.naver.com/helloworld/352076> # Varnish
- <https://gongspace.tistory.com/16> # ATS
- <https://chhanz.github.io/linux/2019/11/13/configuration-squid-proxy/> # Squid

From:

<https://atl.kr/dokuwiki/> - **AllThatLinux!**

Permanent link:

[https://atl.kr/dokuwiki/doku.php/webcache\\_proxy](https://atl.kr/dokuwiki/doku.php/webcache_proxy)

Last update: **2024/10/22 11:22**

