

Load Balancer 연결 구성에서 Connection Reset 사례

May 2021

Contents

| | |
|----------------------|---|
| 1. 개요 | 1 |
| 2. 발생한 문제 | 1 |
| 3. TCPDUMP를 통한 현상 분석 | 1 |
| 4. TCPDUMP 상세 분석 결과 | 3 |
| 5. 해결 방안 | 4 |

1. 개요

웹 서비스의 접속자가 많을 경우 여러 대의 서버가 트래픽 및 부하량 등을 분산하여 처리 할 수 있도록 Load Balancer 를 자주 활용합니다.

이 문서에서는 Load Balancer 를 사용하는 환경에서 발생할 수 있는 여러 장애 사례 중 웹 서버 접속 시 간헐적으로 Connection Reset 이 발생하는 사례에 대해 살펴보려고 합니다.

2. 발생한 문제

웹 서버는 클라이언트가 Load Balancer 를 통해서 접속되는 구성이며, 특정 시간대에 웹 서버로 접속이 되지 않는 문제가 발생했습니다.

- 웹 서버의 Connection error 현상 발생
- 클라이언트는 웹 서버에 접속하는 다른 기능을 하는 서버

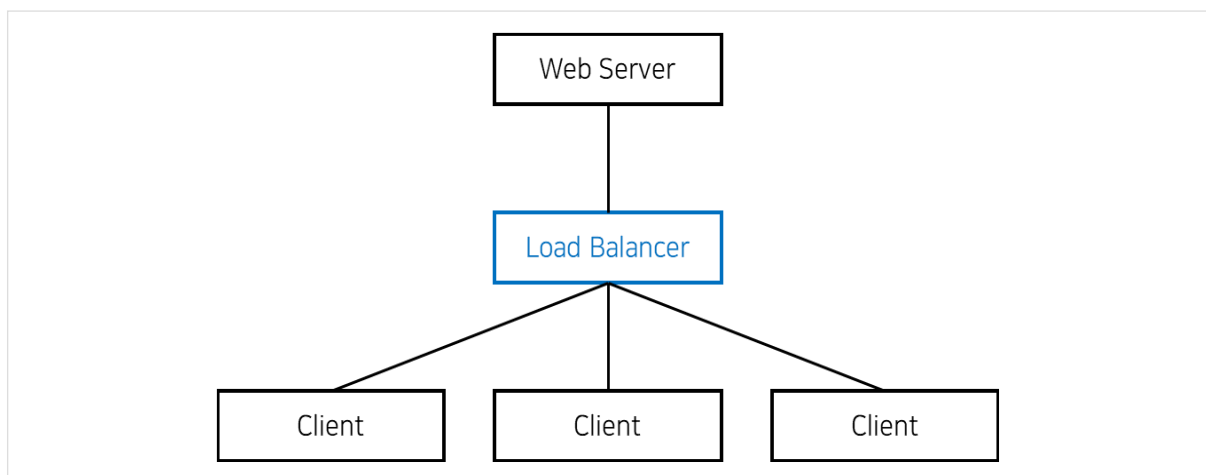


Figure 1. 웹 서버에 접속하는 시스템 구성

3. tcpdump를 통한 현상 분석

문제가 발생했을 상황에서 수집한 tcpdump 파일을 분석한 결과에 대해 정리해 보겠습니다. 일반적으로 정상적인 TCP 3-Way Handshaking 은 아래 그림과 같이 동작합니다.

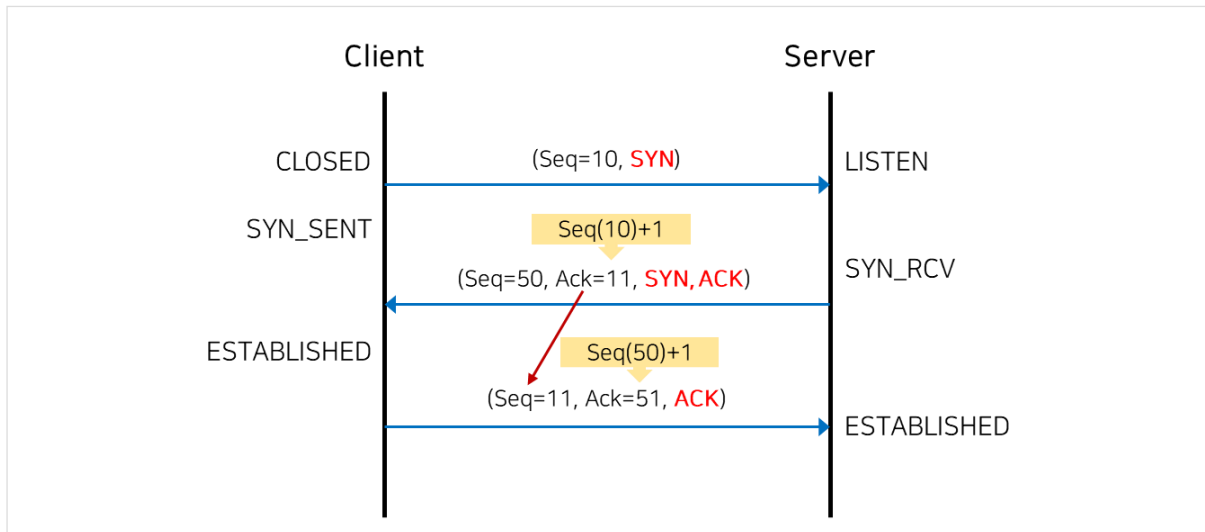


Figure 2. 정상적인 TCP 3-Way Handshaking

하지만, tcpdump 분석 결과 웹 서버의 Connection error 발생했을 당시에는 TCP 세션 정상 종료 후 동일 포트로 접속을 다시 시도할 때 클라이언트에서 RST 패킷을 보내는 현상을 확인했습니다.

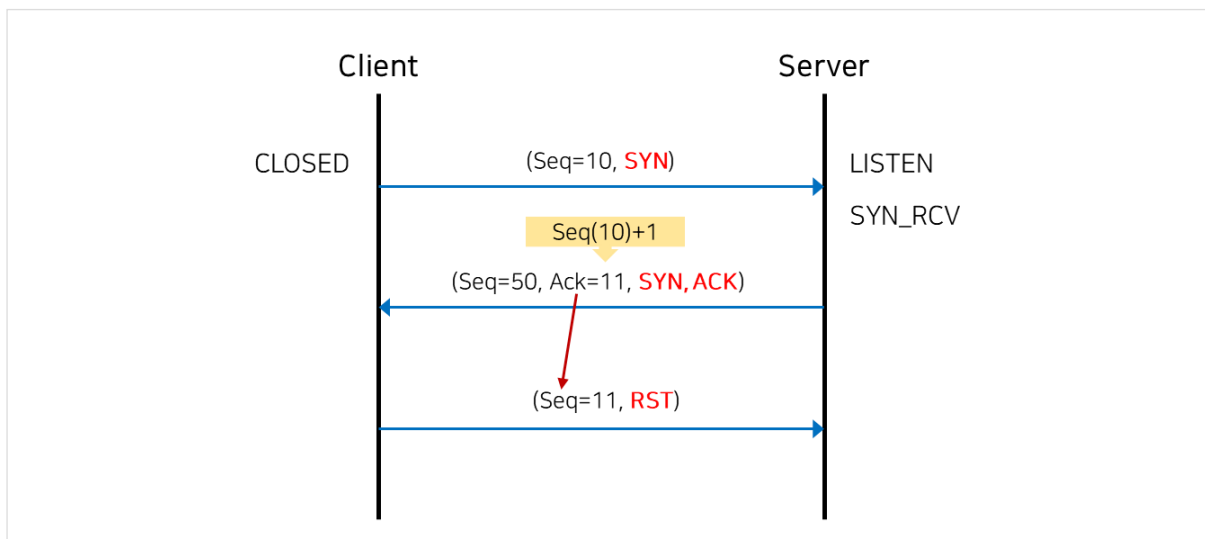


Figure 3. Connection error 발생 시 TCP Handshaking

4. tcpdump 상세 분석 결과

클라이언트와 웹 서버 간에 정상 상태인 TCP 세션 연결과 종료는 아래 그림과 같이 이뤄집니다.

그림에서 보듯이, 웹 서버가 클라이언트에게 보낸 마지막 시퀀스 번호는 2805574211 번입니다. 연결 종료 후 웹 서버는 이 마지막 시퀀스 번호를 기록해두고, TIME_WAIT 상태로 대기하다가 Linux의 TIME_WAIT 기본값인 60 초 이후 사라집니다.

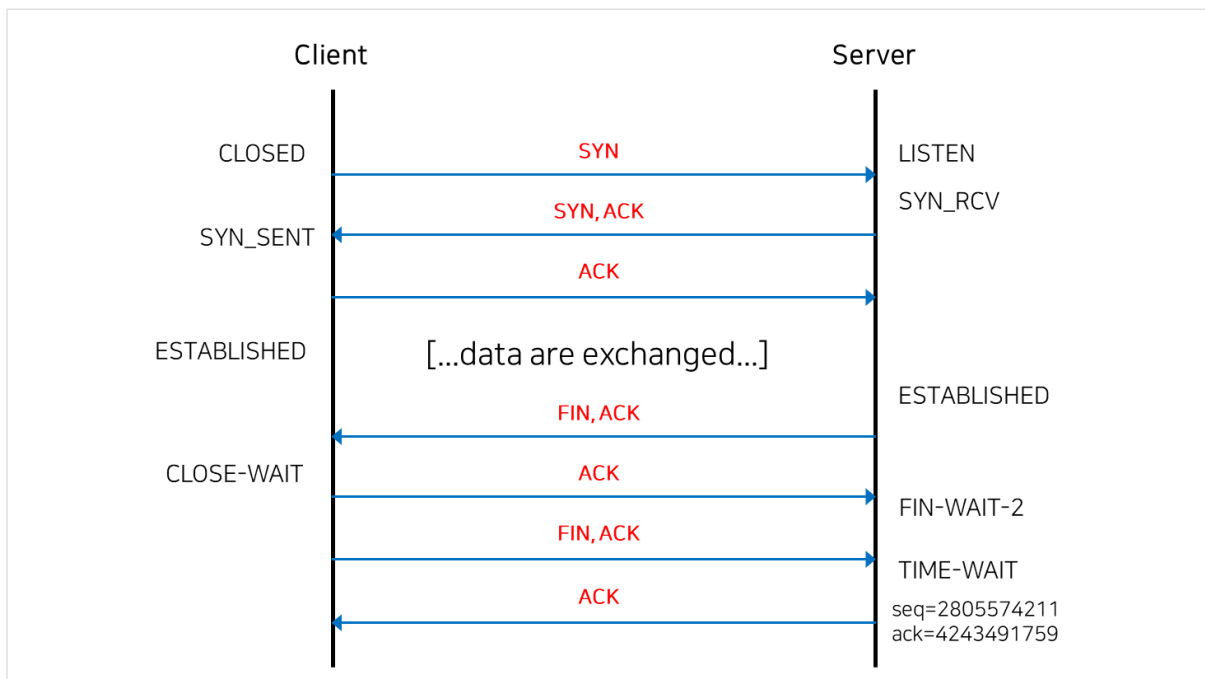


Figure 4. 클라이언트-서버 간 정상 상태의 TCP 세션 연결 및 종료

웹 서버의 Connection Reset 이슈가 발생할 때는 위 그림에서와 같이 웹 서버의 연결 종료된 세션이 TIME_WAIT 상태로 60 초간 대기하므로 클라이언트가 동일한 포트와 새 시퀀스 번호를 사용하여 TIME_WAIT 상태의 세션에 접속을 시도할 수 있습니다.

특히 Load Balancer 를 거쳐갈 때 Load Balancer IP 주소를 사용하므로, 세션이 증가하면 클라이언트 포트가 동일해질 확률이 높아질 수 있습니다.

웹 서버는 여전히 이전 연결에서 동기화되어 클라이언트가 마지막 패킷을 수신하지 않았다고 가정하기 때문에, 이전 연결과 동일한 시퀀스와 승인 번호를 다시 보내게 됩니다. 클라이언트는 그 패킷이 자신이 보낸 것과 일치하지 않는 것으로 인식하고 RST 패킷을 보내 연결을 재설정하게 됩니다. 이 때문에 클라이언트가 접속을 시도할 때 Connection error 가 발생합니다.

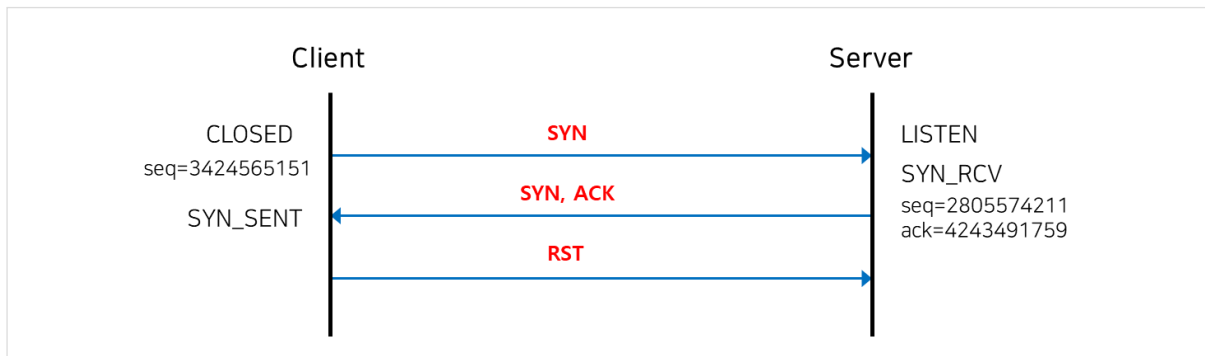


Figure 5. Connection error 발생 시 TCP Handshaking

5. 해결 방안

TCP timestamps 파라미터를 활성화하여 두 번째 새 시퀀스 번호를 사용한 접속 시도에서 SYN 을 인식하는 서버가 실제로 새로운 연결을 위한 것으로 인식하여 정상적으로 연결이 설정되도록 할 수 있습니다.

클라이언트와 서버의 timestamps 관련 커널 파라미터는 아래와 같이 기본값으로 사용 중이었습니다.

```
net.ipv4.tcp_timestamps = 1 (enable, default)
net.ipv4.tcp_tw_recycle = 0 (disable, default)
net.ipv4.tcp_tw_reuse = 0 (disable, default)
```

Figure 6. 커널 tcp 파라미터 설정

클라이언트와 웹 서버는 timestamps 파라미터가 활성화되어 있었으나, Load Balancer 에서 timestamps 파라미터가 비활성화되어 있어서 클라이언트와 웹 서버 모두 timestamps 파라미터가 비활성화되어 통신하는 상황이었습니다. timestamps 파라미터는 연결되는 모든 장비에서 활성화되어야 활성화된 값으로 통신하게 됩니다.

결국, Load Balancer 에서 timestamps 파라미터를 활성화하여 해당 이슈를 해결할 수 있습니다. Load Balancer 를 활용해 서버 ~ 클라이언트 간을 연동할 때 timestamps 파라미터의 설정도 점검하시기 바랍니다.