압축된 TCP/IP세션을 SSH 같은 도구들로 이용하기

Sebastian Schreiber

Schreib (at) SySS.de

도운이

김민수

minsu (at) kldp.org

2.2.2000

고친 과정
고침 1.02001-08-21고친이 김민수
번역
고침 1.02001-08-21고친이 minsu
최초 작성

1. 소개

과거엔, 우리는 압축 파일을 사용해 디스크에 공간을 아끼며 저장을 했다. 하지만 오늘날, 디스크는 값싸다. 그러나 네트웍 대역폭은 한정되어 있다. 당신은 두가지 목표를 이루어야 한다.

1) 당신은 대역폭/이동량을 아껴야 한다.(만약 당신이 네트웍 사용량에 따라 돈을 지불한다면 중요하다.- IDC센터가 그러하죠.역자 첨가)

2) 속도를 낮은 대역폭에서도 높인다.(모뎀,GSM,ISDN)

이 하우투 문서는 어떻게 대역폭과 접속시간 두가지를 SSH1, SSH2, OpenSSH 나 LSH로 아낄수 있는지 설명한다.

2. HTTP/FTP 압축,...

나의 사무실은 64KBit ISDN 으로 인터넷에 연결되어 있다, 그래서 최대 전송율은 약 7K/초이다. 당신은 압축을 하여 연결 속도를 높일수 있다. 내가 파일을 다운로드 받을때, 네스케이프는 초당 40K로 전송하는걸 보여준다. (로그파일은 15배로 압축이 가능하다.) SSH는 안전하지 않은 네트웍으로 안전한 연결을 만드는 도구로 디자인된 주된 툴이다. 게다가, SSH는 압축

연결과 포트 포워딩을 할수 있다. (rinetd나 redir같은). 그래서 압축한 간단한 TCP/IP 연결엔 적당한 툴이다. '간단' 의 의미는 , 오직 하나의 TCP-연결을 연다는 것이다. FTP-연결들이나 MS-아웃룩과 MS-익스체인지는 몇개의 연결을 만들기 때문에 간단하지 않다. SSH는 LempleZiv (LZ77) 압축 알고리즘을 사용한다. - 그래서 당신은 winzip/pkzip 와 같은 높은 압축율을 이룰수 있다. 나의 인트라넷에서 인터넷으로 모든 HTTP-연결들을 압축하기 위해, 나는 단지 나의 전화접속 컴퓨터에 하나의 명령만 실행시키면 된다.

```
ssh -l <login ID> <hostname> -C -L8080:<proxy_at_ISP>:80 -f sleep 10000
```

`<hostname>` = 호스트는 내 ISP에 위치한다. SSH-접근은 필수입니다.

`<로긴 ID>` = 나의 로긴-ID `<hostname>`

`<proxy_at_ISP>` =나의 ISP의 웹프록시

나의 웹 프라우저는 `localhost:8080` 프록시를 사용하도록 설정되어 있다. 나의 랩탑은 같은 소켓으로 접속한다. 그 접속은 SSH로 실제 프록시로 포워드 되며 압축된다. 그 구조는 다음과 같다.

```
                 64KBit ISDN
내 PC--------------------------------A PC (Unix/Linux/Win-NT) 나의 ISP
SSH-클라이언트        압축               SSH-서버, Port 22
Port 8080                               |
 |                                      |
 |                                      |
 |                                      |
 |10MBit 이더넷                        |100MBit
 |압축 안됨                            | 압축 안됨
 |                                      |
 |                                      |
내 두번째 PC                          ISP의 WWW-프록시
의 네스케이프,...                    포트 80
(Laptop)
```

3. 메일 압축

3.1. 이메일 들어옴 (POP3, IMAP4)

대부분의 사람은 그들의 이메일을 POP3를 통하여 메일서버로부터 가져온다. POP3는 많은 불편을 갖고 있는 프로토콜이다.

1. POP3는 패스워드를 깨끗한 텍스트문자로 전송한다.(POP/IMAP 의 SSL-이행과 요구/ 응답 인증이 있다. RFC-2095/2195에 정의되어있다.)

2. POP3의 많은 프로토콜 부하의 원인: 첫째로 클라이언트가 메세지를 요청하든지 서버가 메세지를 보낸다. 그후 클라이언트가 전송된 메일은 삭제해달라고 요청하고, 서버는 삭제를 확인한다. 그후 서버는 다음 처리를 준비한다. 그래서 각각의 이메일에 대해 4번의 처리가 필요하다.
3. POP3는 이메일이 높은 압축이 가능해도 압축없이 전달한다.

당신은 포워딩 로컬호스트 포트 110 통하여 당신의 ISP의 POP3-소켓으로 연결을 압축하여 할수 있었다. 그후 당신은 당신의 메일 클라이언트에게 로컬호스트:110에 안전하고 속도 높인 연결로 메일을 다운로드 받으라고 말해야 한다 -- 그러나 다운로드 시간은 아직까지 POP3-원래 프로토콜 부하로부터 손해를 입는다.

그것은 보다 효과적인 프로토콜로 POP3의 대체하게 만든다. 그 생각은 완전한 메일 박스에 프로토콜 부하없이 다운로드이다. 게다가 압축 연결을 한다. 적당한 도구로 두 특징을 가지는 SCP 가 제공된다. 당신은 당신의 메일-파일을 이렇게 다운로드 할수 있다.

```
scp -C -l loginId:/var/spool/mail/loginid /tmp/newmail
```

그러나 문제가 있다: 만약 새로운 메일을 당신의 메일박스 다운로드 중에 도착한다면 무슨 일이 있어날것인가? 새 메일은 잃어버린다. 그러므로 좀 더 나은 방법은 이렇게 해야한다.

```
ssh -l loginid mailserver -f mv /var/spool/mail/loginid
/tmp/loginid_fetchme
```

```
scp -C -l loginid:/tmp/my_new_mail /tmp/loginid_fetchme
```

mv 는 기본 동작이다. 그래서 만약 당신이 새로운 메일을 받을때 명령이 실행되어 걱정을 하지 않아도 될것이다. 그러나 만약 메일서버 디렉토리가 /tmp/ 와 /var/sp ool/mail 이 같은 디스크가 아니라면 문제가 있을지도 모른다. 해결책은 lockfile을 당신이 mv를 실행시키기전에 만드는 것이다:`touch /var/spool/mail/loginid.lock`. 그후에 옮기면 된다. 더 좋은 해결책은 loginid파일을 같은 디렉토리에 옮기는 것이다.

```
ssh -l loginid mailserver -f mv /var/spool/mail/loginid
/var/spool/mail/loginid_fetchme
```

당신은 /tmp/newmail 안에 맞는 폴더로 필터 하는 procmail대신에 formail을 사용할수 있다.:`formail -s procmail < /tmp/newmail`

3.2. 밖으로 나가는 이메일 (SMTP)

당신은 이메일을 압축과 암호화하여 SSH 연결로 보낼수 있다,이렇게

- 네트웍 트래픽 아끼기
- 보안 연결 (좋은건 아닌데, 나중에 만약 메일이 믿지못할 네트웍을 넘어서 온다면)
- 발신자 인증. 많은 메일서버들은 메일 중계를 악용을 예방하기 위해 거부한다. 만약 당신이 SSH- 연결로 이메일을 보낸다면 원격 메일서버(센드메일이나 MS-exchange)는 로컬로 연결되었다고 생각할것이다.

당신이 메일서버에 SSH-접근권한을 가지고 있다면, 당신은 다음 명령이 필요하다.

```
ssh -C -l loginid mailserver -L2525:mailserver:25
```

당신이 메일서버에 SSH-접근권한을 가지고 있지 않다해도 당신의 메일서버는 중계를 허락할것이다. 다음 명령으로:

```
ssh -C -l loginid other_server -L2525:mailserver:25
```

그 후 당신이 메일 클라이언트(이거나 메일서버: "smarthost"를 보라)를 로컬호스트 2525 포트로 메일이 나가도록 설정 할수 있다.

4. 성능에 대해서 생각하기

물론 압축/암호화는 CPU시간을 소모한다. 오래된 펜티엄-133에서 시간당 1기가를 암호화하고 압축할수 있었다. -- 아주 많은편인데. 만약 당신이 SSH 컴파일을 "--with-none"옵션을 주고 한다면 당신은 SSH에 암호화를 안하고 사용할수 있다. 이건 약간의 성능을 향상시켜 준다. 여기 몇가지 다운로드 방식으로 나타낸것이 있다. (테스트 하는 동안, 5메가의 파일을 압축 안하고 133Mhz의 펜티엄으로부터 233Mhz의 펜티엄2의 랩탑으로 10메가비트 이더넷으로 다른 로드 없이 전송시켰다.)

	FTP	암호화	압축화	압축+암호화
걸린 시간	7.6초	26초	9초	23초
처리량	790K/초	232K/초	320K/초	264K/초
압축율	1	1	3.8	3.8

5. 마치며

Harald Kenig <koenig@tat.physik.uni-tuebingen.de>, 에게 감사한다. 그는 메일박스들을 완벽하게 다운로드하기 위하여 rcp를 사용해왔다. 이 하우투의 마지막 버전은 `http://www.syss.de/howto.` 에 있다.