메일서버의 장애와 대처 방법

인터넷이 웹중심으로 발전하면서도 의사 소통 수단이나 마케팅 수단으로 여전히 중요도가 높아지고 있는 것이 바로 메일 서비스이다. 이번 호에서는 sendmail 과 imap 을 이용한 pop3d 서비스 제공시 발생 가능한 각종 메일 서비스의 장애에 대비하여 안정적인 메일 서비스를 제공하기 위한 방안에 대해 알아보도록 한다.

오늘과내일 넷센터 홍석범(antihong@tt.co.kr)

보내고 받는 메일의 양 제한하기

지난 11월호에서도 잠깐 언급한 것처럼 시스템의 제한 설정과 서비스의 안정성은 매우 깊은 연관성을 가지고 있다. 기본적으로 대부분의 서비스는 유저가 사용 가능한 시스템의 자원 제한이 거의 설정되어 있지 않은데, 메일 서비스도 마찬가지이다.
최근에는 메일의 이용율이 높아지고, 메일의 컨텐츠도 전통적인 텍스트 방식에서 음성,이미지등 각종 동영상이 주종을 이루면서 용량도 점점 커지고 있다. 물론 그만큼 하드웨어나 메일 서버의 소프트웨어적인 성능도 향상되고 있지만 용량이 큰 메일을 주고 받는다면 당연히 시스템의 부하가 올라가기 마련이고 이로 인하여 같은 서버내 다른 서비스에까지 영향을 미치게 된다. 따라서 시스탬에서 보내는 메일 서비스(SMTP)나 받는 메일 서비스(POP3)를 제공하고 있다면 용량이 큰 파일을 주고 받는 것을 적절히 제한할 필요가 있다.

sendmail 은 로컬의 메일을 외부로 발송하는 SMTP(보내는 메일서버) 기능도 있지만 외부에서 서버내 계정으로 전송되는 메일을 받아서 서버에 저장하는 기능도 있다. 이때 기본적으로는 보내거나 받는 메일의 양에 대한 제한이 전혀 없어 10메가 이상이 넘는 큰 사이즈의 메일이 송 수신 될 경우 서버에 과부하가 걸릴 수 있으므로 아래와 같이 각각의 설정(보내는 메일과 받는 메일의 양) 을 적절히 제한하여 설정하는 것이 좋다.

>> SMTP 서버에서 보내는 양 제한하는 법.

/etc/mail/sendmail.cf (또는 /etc/sendmail.cf. 이는 sendmail 의 패키징 방법에 따라 다르다.) 파일에서 다음과 같이 MaxMessageSize 부분의 주석을 제거하고 제한하고자 하는 적절한 값을 입력한다.

```
# maximum message size
O MaxMessageSize=5024000
```

위와 같이 설정하였을 경우 현재의 서버를 보내는 메일 서버로 이용시 첨부 파일이 5M 이상 초과하거나 웹에서 /usr/sbin/sendmail 을 실행하여 외부로 메일을 발송하는 메일링 리스트등의 프로그램에서도 메일 발송시 5 메가 이상의 메일은 보낼 수 없게 된다.
5024000 은 byte 단위이며 설정 변경 후 변경된 내용을 적용하려면 killall –HUP sendmail 로 sendmail 데몬을 Refresh 하면 된다.

>> 받는 메일 서버에서 받는 양 제한하는 법.

외부에서 서버로 들어오는 메일에 대해서 용량을 제한하고 싶다면 같은 파일(sendmail.cf)에서 "Local and Program Mailer specification" 부분을 설정해 주면 된다.

Mlocal, P=/usr/bin/procmail, F=lsDFMAw5:/|@qSPfhn9, S=10/30,
R=20/40, **M=5024000**, T=DNS/RFC822/X-Unix, A=procmail -Y -a $h -d $u

위와 같이 T=DNS/RFC822/X-Unix 앞부분에 M=5024000 부분을 추가해 주면 된다.
마찬가지로 5024000는 byte 단위이며 각자의 시스템 환경에 따라 원하는 용량만큼 적절히 설정해 주면 된다 역시 설정 변경 후 sendmail 을 refresh 하면 적용이 된다.
위의 경우 서버에서는 5메가 이상의 메일은 수신하지 않으며 5메가 이상의 메일을
보낸 이는

552 5.2.3 <antihong@tt.co.kr>... Message is too large; 5024000 bytes max
554 5.0.0 <antihong@tt.co.kr>... Service unavailable
와 같은 에러 메시지를 회신받게 된다.

아울러
maximum number of recipients per SMTP envelope
O MaxRecipientsPerMessage=20

와 같은 부분이 있는데, 이 부분은 한번에 메일 발송 시 동시 발송(참조 발송)이 가능한 메일 계정의 수를 뜻하는 것으로 SMTP 서비스를 제공한다면 이 설정을 적용하는 것이 좋다. 기본적으로 이 값에도 제한이 없으므로 먼저 주석을 삭제한 후 적절한 값을 설정해 주면 한번에 동시 발송 가능한 메일의 수도 제한할 수 있다.
(위의 경우에는 한번에 참조 발송이 가능한 메일 유저를 20명으로 제한)
설정이 끝난 후에는 killall –HUP sendmail 로 sendmail 을 재가동해주면 적용된다.

메일 용량 쿼터 설정하기

각 유저의 홈페이지 공간에 대한 쿼터 설정방법은 잘 알고 있는데, Sendmail 을 제공시 메일 용량 쿼터에 대한 설정은 잘 모르는 경우가 많이 있다. 매일 쿼터에 대한 설정은 다소 복잡하기는 하지만 설정은 가능하다. 기본적으로 각 유저의 메일은 /var/spool/mail/ 디렉토리에 자신의 계정 소유로 저장이 되게 되는데 바로 이 특성을 이용하여 쿼터 설정이을 하면 된다. 쿼터는 각 파일 시스템별로 각각 설정이 가능하므로 각 유저의 홈디렉토리외에 /var 파티션에도 추가적으로 쿼터를 설정하면 되는 것이다.
쿼터를 설정하는 방법은 일반적인 방법과 동일하다.
먼저 /etc/fstab 파일을 열어 /var 파티션이 별도로 설정되어 있다면 /var 파티션에, 별도로 없으면 / 파티션에 유저쿼터나 또는 그룹쿼터 설정을 하면 된다.

```
/dev/sda1               /home          ext2      defaults,usrquota=/home/.quota
/dev/sda8               /var           ext2      defaults,usrquota=/var/.mailquota
```

위에서는 /home 파티션에도 쿼터 설정을 하고 /var 파티션에도 쿼터 설정을 한 것을 볼 수 있다. 이후 touch /home/.quota 및 touch /var/.mailquota 로 사이즈가 0인 파일을 생성한 후 quotacheck –a 를 실행하면 파일 시스템을 스캔하여 디스크 사용량을 체크하여 해당 파일에 정보를 저장한다.

edquota user 를 실행하면

```
/dev/sda1: blocks in use: 0, limits (soft = 99980, hard = 99980)
        inodes in use: 0, limits (soft = 0, hard = 0)
/dev/sda8: blocks in use: 0, limits (soft = 29980, hard = 29980)
        inodes in use: 0, limits (soft = 0, hard = 0)
```

위와 같이 쿼터 설정이 나오는데, 여기에서 /dev/sda1 은 /home/ 디렉토리에 대한 쿼터 설정이고, /dev/sda8 은 /var/ 디렉토리에 대한 쿼터 설정이다. 위 설정으로 각각 /home 디렉토리에는 100메가로, 메일 용량은 30메가로 총 130메가를 할당하여 쿼터를 설정한 것을 알 수 있다. 만약 별도의 /var 파티션이 없이 / 파티션만 있는 상황에서 100 메가로 쿼터 설정을 했다면 이 용량은 홈페이지의 용량과 메일 용량을 합쳐서 100메가로 적용이 되므로 주의하기 바란다.

###################### 참고. Quota 의 설정에 대해
위와 같이 edquota 사용시 관련된 라인이 아래와 같이 보이는 부분이 있다. 이 중
"blocks in use:" 는 유저가 현재 파티션에서 사용중인 총 블럭의 수를 킬로바이트로,
"inodes in use:" 는 유저가 현재 파티션에서 사용중인 총 파일의 개수를 보여준다.
이 두개의 "blocks in use:" 와 "inodes in use:" 는 시스템에 의해 자동으로 설정되고 제어되므로 이 값을 임의로 변경할 필요는 없다.
그리고 quota 설정시 soft 제한(soft = 5000)은 유저가 사용할 수 있는 최대 용량을 뜻하며 (이 예제에서는 약 5M 이다.) hard 제한(hard = 6000)은 유저가 초과할 수 없는 절대적인 디스크 사용량을 뜻한다. "hard limit" 는 "grace period" 옵션이 설정되었을 때에만 적용된다.
grace period 는 쿼터가 설정된 유저나 그룹이 soft limit 을 초과한 이후에도 사용 가능한 시간의 한계이다. 예를 들어서 여러분이 관리하는 시스템에 "해당 유저의 홈디렉토리를 50MB 로 쿼터 제한하고 초과시 7일간의 유예기간을 준다"는 정책을 세울 수도 있다. 각자 유예 기간의 설정에 대해서는 나름대로 적당하다고 생각하는 기간을 정의할 수 있다. grace period 는 edquota –t 로 확인 및 설정할 수 있으며 아래의 경우에는 grace period 가 7일로 설정되어 있는 것을 알 수 있다.

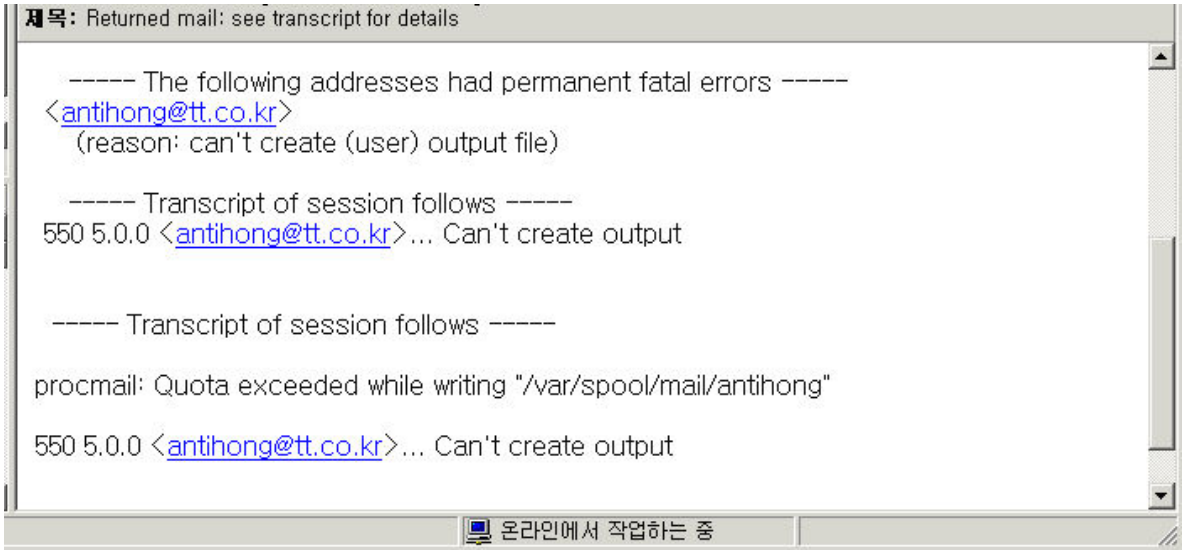
/dev/sd1: block grace period: 7 days, file grace period: 7 days
/dev/sda8: block grace period: 7 days, file grace period: 7 days

그리고 한 유저에게 적용된 쿼터 설정을 다른 유저에게도 그대로 적용하려면 –p 옵션을 사용하면 되는데, 아래와 같이 실행하면 edquota 프로그램은 /etc/passwd 에 정의된 유저중 UID 가 499 이후의 모든 유저에 대해 "user" 의 쿼터 설정을 그대로 복사하게 된다.

edquota -p user `awk -F: '$3 > 499 {print $1}' /etc/passwd`

##

만약 쿼터가 초과된 계정에 메일을 발송하게 되면 아래와 같은 에러 메시지가 나며 더 이상 메일을 수신하지 못하게 된다.



sendmail 이 정상적으로 작동하는지 여부를 아는 방법

sendmail 이 현재 작동중인지 확인하는 방법은 아래 두 가지 방법으로 가능하다.

(1) # ps auxw|grep sendmail 로 확인
위와 같이 확인시
root 0.0 0.0 2684 1460 S Aug24 sendmail: accepting connections on port 25

와 같이 sendmail: accepting connections on port 25 로 보이면 정상적으로 작동하는 것이다. 만약 sendmail 이 다운되어 작동하지 않을 때는 sendmail: rejecting connections 라는 메시지가 보이게 된다.

(2) sendmail 이 반응하는 25번 포트로 접속.

```
# telnet tt.co.kr 25
Trying 211.47.66.50...
Connected to tt.co.kr.
Escape character is '^ ]'.
220 www10.tt.co.kr ESMTP Today and Tomorrow (http://tt.co.kr/)
```

와 같이 sendmail 이 바인딩하는 25번 포트로 telnet 을 접속하면 sendmail 이 반응을

하게 되는데, 위와 같이 접속을 하여 응답이 있을 경우에는 접속을 받아들일 준비가 되어 있는 상태이며 반응하지 않을 때는

Trying tt.co.kr...

telnet: Unable to connect to remote host: Connection refused

와 같이 접근이 거부되었다는 것을 알 수 있다.

갑자기 sendmail 이 작동하지 않을 때

sendmail 이 작동하지 않는 경우는 주로 2가지이다.

첫번째는, 시스템의 부하율인 Load Average 가 높아져 sendmail 이 작동하지 않는 경우이고 두번째는 sendmail 에서 받는 메일이 저장되는 /var 파티션이 100%가 되었을 경우이다. Sendmail 은 기본적으로 시스템의 Load Average 수치가 12를 초과하였을 경우에는 자동으로 작동하지 않게 되는데 이는 sendmail이 서비스 거부 공격등으로 시스템의 부하가 높아졌을 때 sendmail 로 인하여 시스템이 다운되는 것을 막기 위한 조처이다.

이 값을 수정하려면 sendmail.cf의

load average at which we refuse connections

O RefuseLA=12

에서 수정한 후 killall –HUP sendmail 로 재실행해 주면 되고, 이 값은 각각의 시스템에 따라 적절히 조정하면 된다. 만약 현 시스템의 특성상 늘 부하가 높아 로드가 자주 12를 초과한다면 이 값을 각자의 시스템 환경에 맞게 적절히 조절하여야 외부에서 오는 메일을 받을 수 있게 된다.(서버에서 25번 포트로 바인딩하고 있어야 외부에서 오는 메일을 수신할 수 있다.) 그리고 메일이 저장되는 /var/spool/mail 파티션이 가득 찼을 경우(파티션 100%)에도 sendmail 이 작동하지 않으므로 파티션이 가득 찼을 경우에는 /var/log/ 등에서 불필요한 데이터를 삭제하여 /var/spool/mail 이 포함된 파티션이 100% 를 넘지 않도록 하여야 한다. 용량 정리를 하여 파티션이 100%가 넘지 않으면 sendmail 이 자동으로 살아나는 것을 알 수 있다.

또한 시스템의 Load Average 가 8을 넘으면 서버를 통해 메일을 발송해도 메일을 통해 바로 전송되지 않고 일단 서버의 메일 큐에 저장이 된 후에 발송이 되게 된다. 이 역시 같은 이유 때문인데 이 수치는 sendmail.cf 의

load average at which we just queue messages

O QueueLA=8

에서 적절히 설정하면 된다.

참고로 현재 시스템의 Load Average는 w 명령어를 이용하여 확인 가능하다.
w 를 이용시 시스템의 Load Average 는 0.25, 0.40, 0.43 와 같이 보이는데 이는 각각 현재를 기준으로 지난 1분, 5분, 15분간의 평균 시스템 부하율을 나타낸다.

sendmail 에서 보내는 메일(SMTP) 기능을 차단하고자 할 때

sendmail 에서 Relay 기능을 막아 두었다 하더라도 최근 버전에는 사용자 인증(SMTP AUTH) 기능이 있어 서버에 계정이 있으면 모든 유저가 메일 서버를 이용해 SMTP 기능을 이용하여 메일을 발송할 수 있다. 이를 막으려면 최신의 8.11.4 나 8.11.5 와 같이 최신 버전으로 업그레이드 후 /etc/mail/smtpauth 파일에 보내는 메일 기능을 허용할 유저를 입력해 주면 된다. (최근에 8.11.6 이전 버전에 심각한 보안 문제가 확인되었으므로 반드시 8.11.6 버전이나 8.12 버전으로 업그레이드하여야 한다.) 파일을 생성 후 아무런 유저도 입력하지 않으면 서버에 계정이 있다 하더라도 어느 누구도 메일을 발송할 수 없게 된다. 따라서 최신의 8.11.6 버전으로 업그레이드 할 것을 권장한다. 이외 여러 변형된 방법이 존재하는데, ipchains 나 iptables 를 이용해 패킷 필터링을 하는 방법도 있다.

커널 2.2.X 일 경우

```
ipchains -A output -p tcp -y -d 0/0 25 -j DENY
```

커널 2.4.X 일 경우

```
iptables -A OUTPUT -p tcp --syn --dport 25 -j DROP
```

위와 같이 설정시 목적지(Target) 포트가 25번 포트로 향하는 초기화(SYN) 패킷만을 차단하여 메일을 발송할 수 없도록 한다. 물론 초기화(SYN) 패킷에 대해서만 필터링을 하였으므로 외부에서 오는 메일을 받는 것은 관계 없다.

바이러스 메일 필터링 방법

최근에 Sircam 이나 Nimda 등 일정 주기마다 발생하는 바이러스 메일 때문에 서버 관리자들은 마음 고생이 이만저만이 아니다. Sendmail 에서는 이러한 바이러스 메일이나 스팸메일에 대해 룰셋(ruleset)을 이용하여 차단하는 기능이 있는데, 이를 사용하는 방법에 대해 알아보도록 하자.
Sendmail 에서는 제목이나 메일러 또는 첨부파일의 화일명등 각종 메일헤더 정보를 이용하여 필터링을 할 수 있는데, 먼저 발송되는 메일 제목(subject)으로 필터링을 해 보도록 하자. 아래는 메일 제목에 ILOVEYOU 로 발송하는 멜리사 바이러스를 차단하는 룰셋을 적용해

본 예이다.

먼저 sendmail.cf 파일을 열어 제일 하단에 아래의 내용을 추가한다.

```
HSubject: $>Check_Subject
D{WORMmsg}Access Denied - This message may contain a virus.

SCheck_Subject
RILOVEYOU                $#error $: 501 ${WORMmsg}
RRe: ILOVEYOU            $#error $: 501 ${WORMmsg}
RFW: ILOVEYOU             $#error $: 501 ${WORMmsg}
```

주의 : $#error 앞의 blank는 스페이스가 아니라 반드시 탭으로 띄워주어야 한다.

Sendmail.cf 의 설정 내용이 다소 어렵고 복잡하기는 한데, 위 설정의 의미를 간단히 살펴보도록 하자.

H -- 위 경우에는 헤더에서 Subject:라는 문자열을 찾아 이 헤더를 Check_Subject로 정의한다.
D -- WORMmsg 라는 매크로를 정의하여 해당 룰셋에 적용되는 제목을 확인시 발송한 유저에게 보낼 메시지를 정의한다.
S -- 헤더에서 check_subject로 정의한 부분을 룰셋으로 지정하는 부분이다.
R -- 해당 문자열이 포함된 메일을 발견시 앞에서 정의한 에러 메세지를 첨부하여 반송을 시킨다.

위와 같이 룰셋을 적용하였을 경우 "I LOVE YOU" 와 같이 공란이 있을 경우 적용되지 않으며 "ILOVEYOU from me" 와 같이 특정 단어가 추가시에도 적용되지 않으며 반드시 정확히 일치하여야 한다. 추가적으로 회신시 추가되는 Re: 와 전달(포워딩)시 추가되는 FW: 가 추가된 메일도 거부한다.

다음으로 얼마전 유행했던 Sircam 바이러스 메일을 필터링해 보도록 하자.
Sircam 바이러스의 헤더를 보면 정상적인 메일과는 달리 메일 헤더에
Content-Disposition: Multipart message 와 같은 부분이 추가되어 있으며 이 특징을 이용하여 필터링을 하면 된다.

Sendmail.cf 파일에 아래의 룰셋을 추가하면 된다.

HContent-Disposition: $>check_sircam
D{SIRCAM}"Warning: I Guess Sircam.worm Virus"
Scheck_sircam
RMultipart message $#error $: 550 ${SIRCAM}

주의 : $#error 앞의 blank는 스페이스가 아니라 탭으로 띄워주어야 한다.

sendmail.cf의 수정을 끝낸 후 바로 sendmail을 재 시작하지 말고
룰셋이 정상적으로 작동하고 있는지 아래와 같이 테스트를 하는 것이 좋다.

 # /usr/lib/sendmail –bt # 테스트 모드로 접속
ADDRESS TEST MODE (ruleset 3 NOT automatically invoked)
Enter
> check_sircam Multipart message # Sircam 룰셋 테스트
check_sircam input: Multipart message
check_sircam returns: $# error $: 550 553 Warning: I Guess Sircam.worm Virus
> ctrl-D # 테스트 종료

위와 같이 확인된 후 sendmail을 재시작(killall –HUP sendmail) 하면 바로 적용된다.
아래와 같이 tail –f /var/log/maillog 로 로그 파일을 지켜보면 아래와 같이 실제로 Sircam 바이러스가 필터링되고 있음을 확인할 수 있다.

Sep 27 15:09:51 www sendmail[21386]: f8369of21386: to=<antihong@tt.co.kr>, delay=00:00:01, pri=241584 Warning: I Guess Sircam.worm Virus.

마지막으로 최근에 가장 영향을 많이 주었던 변형된 Nimda Worm 을 필터링하는 방법에 대해 알아보자. Nimda Worm 은 정상적인 메일 메시지와 달리 헤더에
boundary="====_ABC1234567890DEF_====" 나
boundary="====_ABC123456j7890DEF_====" 라는 부분이 있는데, 이 부분으로 필터링을 할 수 있다. 즉 메일 헤더에 위와 같은 설정이 되어 있으면 Nimda Worm 으로 간주하고 필터링 하면 되는 것이다. Sircam 에서와 같은 방법으로 sendmail.cf 파일의 설정은 아래와 같다.

HContent-Type: $>check_ct

D{NIMDA}"I guess NIMDA.WORM!!!"

Scheck_ct
R$+ boundary="====_ABC1234567890DEF_====" $#error $: 550 ${NIMDA}
R$+ boundary="====_ABC123456j7890DEF_====" $#error $: 550 ${NIMDA}

이외 메일 필터링에 대한 더욱 구체적인 방법에 대해서는 http://certcc.or.kr/paper/tr2001/tr2001-03/email_security_by_procmail.html 나 http://quanta.khu.ac.kr/~dacapo/sendmail/rulesets/ 를 참고하기 바란다.
그리고 이외 관련하여 바이러스를 스캔하거나 필터링 할 수 있는 몇몇 프로그램이 있는데 이에 대해서는 http://www.rav.ro/ , http://www.amavis.org/ , http://www.sophos.com/ 등을 참고하기 바란다.

메일이 받아지지 않는 경우

아웃룩 익스프레스에서 "배달" 을 눌러 메일을 수신하려고 할 때 메일이 받아지지 않는 경우가 있다. 이러한 경우에는 아래와 같이 여러가지 이유가 있을 수 있으니 아래의 사항에 대해 하나씩 원인을 찾아보기 바란다.

(1) IMAP 패키지가 설치되지 않았을 경우
서버에 배달되어 있는 자신의 계정으로 온 메일을 클라이언트 PC에서 받으려면 pop3 데몬이 반응하게 된다. pop3d 는 IMAP 패키지안에 포함되어 있으므로, IMAP 패키지를 설치하여야 pop3 를 사용할 수 있다. Rpm 으로 설치했다면 rpm –q imap 으로 현재 시스템에 imap 패키지가 설치되어 있는지 확인한다. 또는 /usr/sbin/ipop3d 파일이 있는지 확인해 본다.

(2) Inetd 에 설정되어 있지 않을 경우
pop3d 는 inetd 또는 Xinetd 에서 작동하게 된다.
/etc/inetd.conf 또는 /etc/xinetd.conf 파일을 살펴보아 ipop3 가 주석처리 되어 있거나 pop3 가 disable = yes 로 되어 있지는 않은지 확인한다.

(3) TCP Wrapper 에 설정되었는지 여부 확인
/etc./hosts.deny 에 pop3d 접근이 차단되지는 않았는지 확인한다.

(4) 계정에 Lock 이 걸리지 않았는지 확인
메일을 받는 과정에서 갑자기 회선이 끊기거나 PC가 다운되는 등 비정상적으로 종료시 서버의 pop3d 프로세스가 죽지 않고 계속 남아 있는 경우가 있다.
이러한 경우 계정에 "Lock 이 걸렸다" 라고 하며 이러한 경우에는 해당 프로세스를 찾아 kill 을 하면 된다. 만약 계정에 Lock 이 걸린 상태에서 아웃룩 익스프레스에서 메

일을 수신하려고 하면 아래와 같은 에러가 나게 된다.
“메일 서버에 로그온하는 데 문제가 있습니다. 지정한 암호가 거부되었습니다.
계정: 'temazone.com', 서버: 'tt.co.kr', 프로토콜: POP3, 서버 응답:
'-ERR Can't get lock. Mailbox in use', 포트: 110, 보안(SSL): 아니오,
서버 오류: 0x800CCC90, 오류 번호: 0x800CCC92”

(5) Pop3 접속이 많은 경우

Pop3d 가 서비스되는 inetd는 기본적으로 60초동안 40회의 접속을 받아들이도록 (즉, maximum 40회 fork되도록) 설정되어 있다. 따라서 짧은 시간에 pop3d 요구가 많을 경우에는 메일로그에 pop3/tcp server failing (looping) 라는 메시지가 나면서 pop3d 데몬 자체가 다운되어 버리므로 동시에 처리 가능한 프로세스의 한계를 적절히 높여주어야 한다.

이를 위해서는 /etc/inetd.conf 를 열어 아래와 같이 수정하면 된다.

이전설정)

```
pop-3     stream     tcp     nowait     root     /usr/sbin/tcpd     ipop3d
```

변경 설정)

```
pop-3     stream     tcp     nowait.200     root     /usr/sbin/tcpd     ipop3d
```

(위의 경우 처리 가능한 프로세스를 200회로 늘려주었다.)
이후 killall -HUP inetd 를 하면 된다.

(6) 110 번 포트로 확인

아래와 같이 pop3d 포트인 110번 포트로 직접 접속하여 수작업으로 확인 가능하다.

```
# telnet   pop3.tt.co.kr 110              # 110번으로 직접 확인
Trying 210.17.6.5...
Connected to pop3.tt.co.kr.
Escape character is '^ ]'.
+ OK POP3 pop3.tt.co.kr v2001.76 server ready
user abc                                  # abc 라는 계정으로 접속
+ OK User name accepted, password please
pass xyz                                  # abc 의 암호 xyz 입력
+ OK Mailbox open, 10 messages
quit                                      # 접속을 끊음.
+ OK Sayonara
```

Connection closed by foreign host.

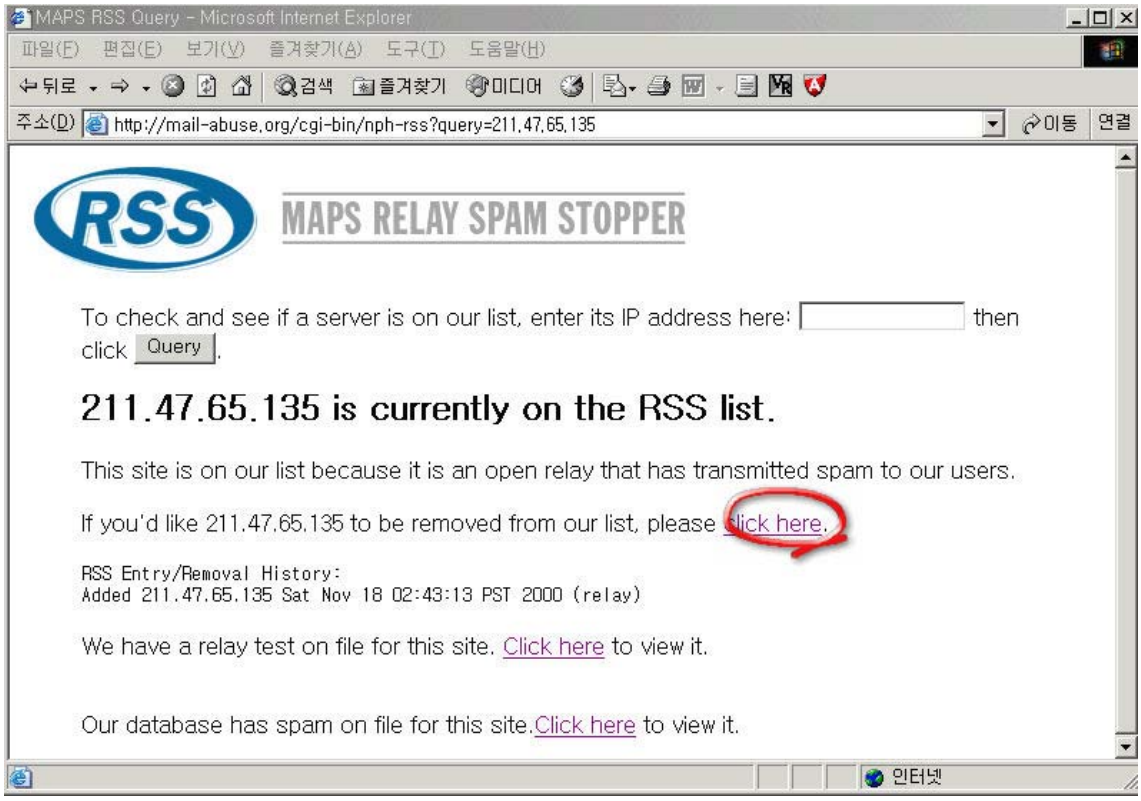
위의 경우는 정상적인 경우이며 에러가 있을 경우(만약 암호가 다르게 설정되었을 경우 - ERR Bad login 와 같은 메시지가 나게 된다.) 각각의 경우에 따라 에러 메시지를 각각 확인할 수 있다.

(7) mail –v 로 확인

타 서버에서 mail –v antihong@tt.co.kr 와 같이 메일을 발송하여 정상적으로 메일이 도착하는지를 확인해 본다. –v 옵션을 이용하여 메일 발송시에는 메일 전송의 경로 및 메일 서버간에 주고받는 메시지를 확인할 수 있으므로 문제의 원인을 찾는데 도움이 된다.

특정한 곳으로만 메일이 돌아올 때

다른 곳은 문제가 없는데, 해외등 특정한 곳으로만 메일이 전송되지 않고 리턴되는 경우가 있다. 이러한 경우라면 자신의 메일서버가 mail-abuse.org 의 블랙 리스트에 등록되어 있지는 않은지 확인해 볼 필요가 있다. 특히 회신된 메일에 "...refused by blackhole site relays.mail-abuse.org" 와 같은 메시지가 보인다면 반드시 여부를 확인해 보아야 한다. 적지 않은 메일 서버에서는 메일 수신시 실시간으로 이 데이터를 참조하므로 mail-abuse.org 에서 스팸 메일 서버로 등록되면 이 기관에 등록된 도메인으로 메일을 보낼 때 받는 쪽에서는 스팸 메일로 간주하고 수신을 거부하게 된다. 이를 확인하는 방법은 http://mail-abuse.org/cgi-bin/nph-rss 사이트에서 메일 서버의 IP 를 조회해 보면 된다. 아래는 위 사이트에서 한 IP 에 대해 조회해 본 결과 블랙 리스트에 등록되어 있는 것을 보여주고 있다. 이러한 경우라면 조회한 메일 서버의 Relay 가 허용되어 스팸 메일 서버로 사용된 적이 있거나 현재 사용되고 있다는 뜻이다. 만약 스팸메일 서버로 등록되어 있지 않다면 211.47.65.xxx is NOT currently on the RSS list 와 같이 보이게 된다.



자신의 메일 서버를 이 블랙리스트에서 제외하려면 먼저 자신의 메일서버에 Relay 가 허용되어 있는지 확인 후 메일 서버에서 Relay 를 거부 설정한 후 If you'd like 211.47.65.135 to be removed from our list, please click here. 를 따라 클릭하여 신청을 하면 된다. 이 링크를 클릭하면 신청폼이 나오는데, 이 곳에 입력하여 신청을 하면 바로 처리가 된다. Relay 거부 설정을 한 후 신청을 해야 처리가 되므로 반드시 사전에 Relay 거부 설정을 확인하기 바란다. 메일 서버의 Relay 여부를 조회하는 방법에 대해서는 본지 10 월호 “철벽 보안을 위한 모니터링 올가이드” 를 참고하기 바란다.

복수 MX 설정시 주의해야 할 점

DNS 서버에서 설정하는 MX 레코드는 해당 호스트로 수신되는 편지를 다른 호스트로 라우팅 하도록 한다. 특히 웹서버와 메일 서버를 분리하고자 할 경우 사용되는데, 원격 호스트에서 아래와 같이 설정된 도메인 tt.co.kr 로 편지를 송신할 경우에 Sendmail이 어떻게 동작하는지 알아보자.

```
tt.co.kr.          IN    MX      10      mail1.tt.co.kr.
```

```
IN    MX      20      mail2.tt.co.kr.
IN    MX      20      mail3.tt.co.kr.
```

다음은 메일이 수신되는 차례를 보여준다.

(1) Preference 값이 10으로 가장 낮은 mail1 로 먼저 배달을 시도한다.
(2) 만약 mail1.tt.co.kr 이 접근이 불가능하면 mail2 혹은 mail3 으로 배달을 시도한다.
(3) (2) 에서 시도한 메일서버로도 접근이 되지 않으면 (2)에서 접근 되지 않은 호스트로 배달을 시도한다. 즉 mail2 로 전송을 시도했다면 mail3 으로 배달을 시도한다.
(4) mail2 와 mail3 서버에 접근이 불가능하다면 자체 큐잉 후, 일정 기간동안 주기적으로 1-3의 과정을 반복한다.

흔히 MX 레코드에 대해 잘못 생각하는 것 중 하나는 만약 mail1 이 다운되어 mail2 로 편지가 배달되었을 때, 편지가 mail2 의 메일 박스에 저장 된다고 생각하는 것이다. 만약 이렇게 된다면 유저 입장에서는 메일 수신시 pop3 서버를 mail1.tt.co.kr 와 mail2.tt.co.kr 과 같이 여러 개 설정해야 하는 것처럼 보인다. 그러나 일반적으로 mail2.tt.co.kr 이나 mail3.tt.co.kr 처럼 Preference 가 높은(즉 우선도가 낮은) 값을 갖는 메일 서버는 큐잉 서버로 동작하도록 설정하기 때문에, 결국 메일은 하나의 호스트(mail1)로 모이게 되는 것이다. 위와 같이 mail2와 mail3 서버가 큐잉 메일 서버로 작동하려면 mail1 와 mail2의 sendmail 이 아래와 같이 설정되어야 한다.

(1) 해당 도메인(tt.co.kr)에 대한 인증을 갖지 않아야 한다.
(즉, mail2 나 mail3 메일 서버의 sendmail.cw 또는 local-host-names 파일에 tt.co.kr 이 설정되어 있으면 안 된다.)

(2)서버는 해당 호스트로의 메일 릴레이(Relay)를 허용하여야 한다.
(즉, /etc/mail/access 에서 아래와 같이 정의되어야 한다.)
mail1.tt.co.kr relay

인증을 갖지 않아야 한다는 것은 Sendmail의 w 클래스(sendmail.cw(local-host-names) 혹은 sendmail.cf의 Cw)에 tt.co.kr 도메인이 설정되지 않아야 하는 것을 의미하고, 메일 릴레이란 수신되는 편지의 최종 배달지가 자신이 아닐 경우, 즉 인증을 갖지 않을 경우 편지를 해당 호스트로 포워딩하는 것을 의미한다. 최근의 배포판에서는 기본적으로 sendmail 이 릴레이를 거부하도록 설정되어 있으므로 메일 큐잉 서버의 경우는 해당 호스트를 목적지로 하는 메일에 대해서는 릴레이를 허용하도록 설정하여야 한다는 것을 주의하기 바란다.
mail1 의 다운으로 인해 mail2 로 전달되는 메일은 메일큐에 저장되어 있으면서, 일정 기간

(Sendmail.cf에서 지정된 Timeout.queuereturn=5d 만큼)동안 주기적(Sendmail 구동시 지정된, 일반적으로 30분 -q30m)으로 mail1 로 배달이 재시도된다.

메일 서버의 버전을 숨기는 법

다른 데몬도 마찬가지이지만 메일 서버 역시 해당 포트로 원격 접속을 해 보면 메일 서버의 버전 정보등을 확인할 수 있다. 그러나 시스템 관리자 입장에서 보안상의 문제로 현재 운영중인 메일 서버의 버전등을 숨기거나 속이고 싶을 때가 있는데. 이러한 경우에는 아래의 방법을 이용하면 된다.

(1) sendmail 의 경우
sendmail.cf 파일을 보면 아래와 같은 설정이 있다.
SMTP initial login message (old $e macro)
O SmtpGreetingMessage=$j Sendmail $v/$Z; $b
이 부분을 적절히 삭제하거나 다른 정보로 입력후 sendmail 을 재가동하면 된다.
필자가 운영하는 메일서버의 경우
O SmtpGreetingMessage=$j Today and Tomorrow(http://tt.co.kr/) 와 같이 설정하였고 이때 25번 포트로 접속시 보이는 정보는 아래와 같다.

```
# telnet tt.co.kr 25
Trying 211.47.66.50...
Connected to tt.co.kr.
Escape character is '^]'.
220 www10.tt.co.kr ESMTP Today and Tomorrow(http://tt.co.kr/)
```

(2) pop3d 의 경우
pop3d 의 경우 소스에서 직접 수정하여야 하는데, 압축 해제한 디렉토리의 /src/ipopd 에 보면 ipop3d.c 파일이 있다. 이 파일을 살펴보면

```
char *version = "2001.75";   /* server version */
```
라는 부분이 있는데, 필자가 운영하는 pop3d 의 경우 소스에서
```
char *version = "xxxxxxxxxx";   /* server version */
```
와 같이 수정 후 컴파일 하였고 이때 110번 포트로 원격 접속시 보이는 정보는 아래와 같다.

```
# telnet tt.co.kr 110
```

```
Trying 211.47.66.50...
Connected to tt.co.kr.
Escape character is '^]'.
+OK POP3 www10.tt.co.kr vxxxxxxxxxx server ready
```

버전외 다른 각종 정보도 수정할 수 있으니 각자 상황에 맞게 적절히 설정하기 바란다.

sendmail 과 관련된 몇 가지 명령어

>> mailq

mailq 프로그램의 목적은 큐잉된(/var/spool/mqueue 에 저장된) mail 메시지의 요약된 정보를 보여준다. 네트워크 다운등 어떤 특정한 이유로 바로 발송되지 못한 메일은 일차적으로 /var/spool/mqueue 에 큐잉된 상태로 저장된 후 일정 시간마다 발송을 위해 재시도가 되는데, 현재 큐잉된 메일 메시지의 요약 정보를 보려면 아래와 같이 확인할 수 있다.

```
# mailq

/var/spool/mqueue/q1 (2 requests)
----Q-ID---- --Size-- -----Q-Time----- ------------Sender/Recipient------------
f7A84oV15068      1446 Fri Aug 10 17:04 nobody
                (Deferred: Connection timed out with kebi.net.)
                                      darling@kebi.net
f775ieF24893    521898 Tue Aug  7 14:44 <shlee@tt.co.kr>
                (Deferred: Connection timed out with mail.unitel.net.)
                                      <cf1318@unitel.net>
/var/spool/mqueue/q2 is empty
               /var/spool/mqueue/q3 (1 request)
----Q-ID---- --Size-- -----Q-Time----- ------------Sender/Recipient------------
f775nJF25249    230815 Tue Aug  7 14:49 <shlee@tt.co.kr>
                (Deferred: Connection timed out with hanmail.com)
                                      cuwww23@hanmail.com
```

위 메시지를 보면 어떠한 이유로 메일이 발송되지 못하고 있는지를 추측할 수 있다.
3 메시지 모두 수신자의 e-mail 주소를 잘못 기입했기 때문인데, 각각 kebi.com 인데, kebi.net 으로 unitel.co.kr 인데, unitel.net 으로 , hanmail.net 인데, hanmail.com 으로

도메인 주소를 잘못 기입하여 메일을 발송하여 서버에서 메일을 발송하지 못하고 큐에 저장되어 있는 것을 확인할 수 있다.
 여기에서 주의할 점은 mailq 명령어는 일반 유저로 실행하여 확인이 가능하므로 퍼미션을 700 등으로 조절하여 일반 유저들은 실행할 수 없도록 하는 것이 좋다.

>> mailstats
 mailstats 프로그램은 현재의 메일 송수신과 관련하여 통계를 보여준다.

  * 현재의 메일 통계를 보려면 아래와 같이 확인할 수 있다.

```
# mailstats
Statistics from Sat Aug 11 04:02:02 2001
 M   msgsfr  bytes_from   msgsto    bytes_to  msgsrej msgsdis  Mailer
 1        0          0K        3        317K        0       0  *file*
 4      690      596691K      824     137070K    68426       0  esmtp
 9       63       12212K        0          0K       27       0  local
=============================================================
 T      753      608903K      827     137387K    68453       0
 C      753                   827               68453
```

이를 적절히 이용하면 mrtg 를 이용해 일정 시간마다 발송되고 수신되는 메일의 개수를 통계로 내어 그래프로 볼 수 있다.(본지 10월호, 철벽보안을 위한 모니터링 올가이드 참조)

최근 sendmail 관련 버그에 대해

한동안 문제가 없었던 sendmail 에 최근 들어 몇 가지 보안 문제가 발견되었다.
이 버그는 매우 치명적인 문제인데, 아직 이를 모르고 그대로 사용중인 유저들이 많은 것 같다. 각자의 메일 서버에는 해당사항이 없는지 꼭 확인해 보기 바란다.

첫번째로, 8월말에 발표된 버그는 현재 대부분의 메일 서버 프로그램으로 사용중인 sendmail 8.11.6 이전 버전에 해당하는 보안버그로서 일반유저가 Local 에서 root 권한을 얻을 수 있는 매우 치명적인 버그인데, 이미 공격 소스가 여러 사이트에 공개되어 있다.
참고로 이 버그는 8.11.0부터 8.11.5 버전까지만 해당하므로 8.10.x 나 8.9.x 는 해당되지 않는다. 따라서 아래의 사이트를 참고로 sendmail 을 8.11.6 이나 8.12등 최신버전으로 업그레이드하기 바란다.

8.11.0부터 8.11.5 의 경우 8.11.6 으로 업그레이드하면 되고 8.12.0.Beta 의 경우 8.12.0.Beta19 이상으로 업그레이드하면 된다. 이에 대해서는
http://www.securityfocus.com/bid/3163 나 http://www.sendmail.org/8.11.html
를 참고하기 바란다.

두번째는, 10월초에 발견된 버그로서 모든 버전에 해당하는 문제인데, 이전에도 자주 나왔던 문제이다. 바로 shell 접근이 가능한 일반유저가 sendmail 에 -q 옵션을 사용하여 큐에 있는 메시지를 드롭할 수 있는 문제이다. 아래의 설명을 보기 바란다.

```
[user@net user]$ id
  uid=778(user) gid=778(user)
[user@net user]$ mailq
          Mail Queue (1 request)
--Q-ID-- --Size-- -----Q-Time----- ------------Sender/Recipient----------
--
NAA05248      11 Tue Oct  2 13:03 user1
          (Deferred: Connection refused by tt.co.kr.)
                    test@tt.co.kr

[system@net system]$ /usr/sbin/sendmail -q -h10000
Too many hops 10000 (25 max): from system via localhost, to test@tt.co.kr
Too many hops 10000 (25 max): from MAILER-DAEMON via localhost, to
postmaster
Too many hops 10000 (25 max): from MAILER-DAEMON via localhost, to
postmaster
MAILER-DAEMON... Saved message in /usr/tmp/dead.letter
[user@net user]$ mailq
Mail queue is empty
```

위와 같이 hop count 를 크게 설정함으로써 일반 유저가 현재 큐의 내용을 강제적으로 drop 시킬 수 있다.

세번째는 역시 모든 버전에 해당하는 문제로 일반 유저가 sendmail -q -d0-xxxx.xxx 와 같이 사용시 (xxx는 디버깅 레벨이다.) 일반 유저가 메일서버의 각종 설정 뿐만 아니라 큐에 저장되어 있는 내용, 메시지 경로나 제목, 메일 소프트웨어등의 정보를 볼 수 있는 문제이다.

두번째,세번째 문제는 sendmail.cf 에서

O PrivacyOptions=authwarnings,novrfy,noexpn,restrictqrun
와 같이 restrictqrun 를 추가함으로써 해결 가능하다.

기타 메일과 관련된 장애가 확인 시

지난달 아파치 웹서버의 장애에 대해 이야기하면서 문제나 장애가 발생시에는 웹서버의 error_log 메시지를 살펴보도록 이야기 했었다. 메일서버도 마찬가지이다. 메일서버 장애시는 문제의 원인을 찾기 위해 로그 파일을 살펴보는 습관을 들이는 것이 좋다.
메일 관련 로그는 /var/log/messages 나 /var/log/maillog 파일을 살펴보면 되며 로그파일을 보면 여기에서 언급하지 않은 문제가 발생했다 하더리도 어렵지 않게 원인을 찾을 수 있을 것이다. 다시 한번 강조하지만 모든 문제의 원인과 해결책은 로그에 있다는 것을 명심하기 바란다.