



제4회 리눅스 공동체 세미나 강의를

B-1세션

웹-PHP-MySQL 연동(1)

PHP와 IMP를 이용한 웹메일 서비스의 구현

1. 시작하기 전에
2. 설치
3. 끝내며

● 정진호

loveme@linuxone.co.kr

● 게시판 주소

http://seminar.klug.or.kr/inform/teacher/index.php3?ses=b_1

B-1 세션 - 웹-PHP-MySQL 연동(1)

PHP와 IMP를 이용한 웹메일 서비스의 구현

1 시작하기 전에

1 목적

본 강의는 최근 강력한 성능과의 편리함으로 인기를 끌고 있는 스크립트 언어인 PHP, 가볍고 빠른 무료 공개형 데이터베이스 서버 MySQL, 최강의 웹서버인 아파치, IMAP 프로토콜을 사용하는 공개 웹메일 프로그램 IMP를 이용하여 간단한 기능의 웹메일 서비스 구현을 목표로 한다. PHP를 이용한 실제 소스의 코드 분석에는 큰 비중을 두지 않고, 각각의 프로그램을 유기적으로 결합해 최종적으로 웹메일 서비스를 구현하는 것에 중점을 둘 것이다.

2 사용 소프트웨어

각각의 프로그램 버전은 다음과 같고 2000년 3월 현재 구할 수 있는 최신 버전을 사용할 것이다.

프로그램	버전	출처
Apache	1.3.12	http://www.apache.org/dist
PHP	3.0.15	http://www.php.net
MySQL	3.22.32	http://www.mysql.org/download.html
IMAP	4.5-4	ftp://ftp.redhat.com/redhat/redhat-6.1/i386/RedHat/RPMS
IMP	2.0.11	http://www.horde.org/imp/download
Yamail	1999-04-08	http://www.alzzalinux.com/solutions/yamail/index.php3

3 웹메일 서비스를 위한 설치 순서

위의 프로그램들을 효율적으로 연결해 서비스하기 위해 다음과 같은 순서로 프로그램을 설치하게 된다.

- 1) 기존 RPM 방식으로 설치된 아파치 확인 / 제거
- 2) MySQL 한글 정렬 옵션 소스 컴파일 / 인스톨 / 자동 실행 설정
- 3) IMAP 인스톨 및 접속 확인
- 4) 아파치 소스 임시로 풀기
- 5) PHP 컴파일 / 인스톨
- 6) 아파치 컴파일 / 인스톨 / 설치확인 / 자동실행 설정
- 7) Yamail 설치 / 테스트
- 8) Horde / IMP 설치 / 테스트



4 설치 디렉토리

설치가 완료되면 각각 프로그램의 최종 디렉토리는 다음과 같이 된다.

- Apache : /usr/local/apache
- PHP : /usr/local/php
- MySQL : /usr/local/mysql
- IMAP : /usr/local/imap(/usr/lib , /usr/include)
- IMP(horde library) : /usr/local/apache/htdocs/horde/imp
- Yamil : /usr/local/yamil(alias as /yamil/ in httpd.conf)

5 사용된 서버

- 펜티엄 셀러론 433, 128M, 리바TNT2 m64 32, 8.4G 쿼텨, 40배속 CD-ROM
- 밀레니엄 알자 레드햇 6.1,(커널 2.2.12)

2

설치

설치에 앞서 IMAP에 대해 간략히 살펴보자.

● IMAP이란 무엇인가?

출처 : <http://dbfa.ce.pusan.ac.kr/dblab/member/yminpark/reading/Email/using6.html>

IMAP은 'Internet Message Access Protocol'의 약자로서 로컬 서버로부터 e-mail을 읽기 위한 표준 프로토콜을 말한다. 최신 버전은 IMAP4이며, POP와 달리 메일의 제목이나 보낸 사람만 보고 메일을 다운로드할 것인지 선택할 수 있습니다.

IMAP 서버를 사용하면 폴더나 메일박스를 서버에 만들 수 있으며, 메일박스의 메시지에 직접 접근하여 지우거나 검색을 할 수도 있다. IMAP은 리모트 파일 서버(메일 서버에 접속하여 메시지를 관리하는)에, POP는 '저장하고 전달하는' (다운로드만 하는) 메일 서비스에 비유할 수 있다.

IMAP은 이런 경우에 매우 효과적이다. 학교나 직장에서는 인터넷에 항상 접속된 컴퓨터를 사용하며 메일을 체크하고 집에서는 전화로 전자우편을 체크한다면 POP3는 불편할 때가 많다. 받았던 메일을 또 받아야 되고, 답신을 한 것인지 하지 않은 것인지 체크하기도 어렵기 때문이다. 또, POP3 서버에서 메일을 받은 후 서버에서 삭제해 버렸다면 메일박스나 개별 메일을 FTP 등으로 전송해야 하는 불편함도 있다.

IMAP은 이런 POP의 단점을 보완하기 위해 제안되었다. POP 대신 IMAP을 받는 메일 서버로 사용하면 계층적 폴더 관리, 원격 폴더 조작, 메일 폴더 공유가 가능하다. IMAP이 POP를 대체하는 인터넷의 새로운 전자우편 이용을 위한 프로토콜로 부상하고 있기는 하지만, 아직은 이를 지원하는 메일 서비스가 그리 많은 편은 아니다. 그럼에도 불구하고 최근에 나오는 대부분의 메일 클라이언트들은 IMAP 서비스를 지원하고 있다.

IMAP을 제대로 사용하려면 최신의 IMAP4를 서버에서 지원해야 하고, 메일 클라이언트 또한 이것을 잘 지원해야 한다. 현재 나와 있는 Outlook Express4.x의 경우, IMAP4를 지원하고 있기는 하지만 플래그(flag)나 메시지 삭제등을 제대로 지원하지 못하는 실정이다. 그러나 Outlook Express 5.0은 IMAP4 규약을 좀더 확실히 지원한다. Netscape 4.5PR2에 포함된 메일러인 Messenger의 경우는 최신의 IMAP4를 잘 지원하는 편이며, IMAP4 서버를 사용하면 다중 계정을 지원한다. 한편, 유닉스 호환 서버에서 대중적으로 사용하는 메일러인 Pine은 이전 버전부터 IMAP4를 잘 지원해 왔다. 게다가 최신 버전인 Pine 4.03의 경우는 IMAP4와 POP3 모두 리모트에서 접속할 수 있는 기능을 제공한다.

국내 ISP 중 받는 메일 서버로서 IMAP를 지원하는 곳으로는 Channel i, Netsgo 등이 있으며, 무료 E-메일 서비스인 Netian도 IMAP4를 지원한다. 만약 여러분이 학교나 연구소 혹은 회사에서 E-메일 서비스를 이용하고 있다면 메일 서버 관리자에게 IMAP 서비스를 제공하도록 요청해 보기 바란다. IMAP 서버를 설치하는 것은 그다지 어렵지 않으며, 여러분은 그로 인해 E-메일을 보다 편리하게 관리할 수 있을 것이다.

자신이 사용하는 메일 서버가 IMAP을 지원하는지 확인하는 가장 간단한 방법은 메일 서버 관리자에게 메일을 보내 물어 보는 것이 있다. 또, 텔넷으로 직접 IMAP 포트로 연결하여 확인하는 방법도 있다. IMAP은 대개 143번 포트를 사용하므로 계정이나 텔넷 에뮬레이터를 사용하여 143번 포트로 접속하여 응답이 오는지 확인해도 된다.



7 데이터베이스 중지

MySQL 중지 : `mysqladmin shutdown`

```
[root@roni mysql]# ./bin/mysqladmin shutdown
[root@roni mysql]# mysqld daemon ended

[1]+  Done                  ./bin/safe_mysqld
[root@roni mysql]#
```

8 부팅시 자동시작 설정

`/usr/local/mysql/share/mysql/mysql.server` 를 `/etc/rc.d/init.d/`에 복사한 후 실행 가능하도록 퍼미션을 변경한다.

```
[root@roni mysql]# chmod 700 /etc/rc.d/init.d/mysql.server
```

기동 :# `/etc/rc.d/init.d/mysql.server start`

중지 :# `/etc/rc.d/init.d/mysql.server stop`

`/etc/rc.d/rc.local` 마지막에 다음 내용을 추가하면 부팅시 MySQL 데몬이 자동으로 시작된다.

```
....
/etc/rc.d/init.d/mysql.server start
```

팁 한글 메시지 보기

오류 메시지나 기타 메시지를 한글로 보기 위해서는 데몬을 기동할 때 다음 옵션을 사용한다.

`-language=korean`

`mysql]# ./bin/safe_mysqld -language=korean &`

```
[root@roni mysql]# ./bin/safe_mysqld -language=korean &
[1] 19867
[root@roni mysql]# Starting mysqld daemon with databases
from /usr/local/mysql/var

[root@roni mysql]# ./bin/mysql mysql
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Welcome to the MySQL monitor.  Commands end with : or \g.
Your MySQL connection id is 1 to server version: 3.22.32

Type 'help' for help.
mysql> sleldfsdf;
ERROR 1064: 'SQL 구문에 오류가 있습니다.' 에러 같습니다. ('sleldfsdf' 명령어 라인 1)
mysql>
```

3 IMAP 인스톨 및 접속 확인

소스위치 : `ftp://ftp.cac.washington.edu/imap/imap.tar.Z`

최신버전 : 4.7b

① imap 서비스 확인(port 143)

```
(root@roni tmp)# telnet localhost 143
Trying 127.0.0.1...
telnet: Unable to connect to remote host: 연결이 거부됨
[root@roni tmp]#
```

② IMAP 소스 설치

```
# zcat imap.tar.Z | tar xvf -
# mv ./imap-4.7b/ /usr/local/imap
# cd /usr/local/imap
# make slx
```

③ 데몬 프로그램 복사

```
#cp imapd/imapd /usr/sbin
#cp ipopd/ipop[23]d /usr/sbin
```

④ 링크 생성

```
ln -s c-client include
ln -s c-client lib
```

⑤ 헤더와 라이브러리 복사

```
# cp ./include/*.h /usr/include/
# cp ./include/*.a /usr/lib/
```

⑥ /etc/service 확인 및 /etc/inetd.conf 설정 파일 수정

```
root@roni /tmp)# vi /etc/services // 포트확인
[root@roni /tmp]# vi /etc/inetd.conf // inetd 수퍼서버 설정확인
```

```
.....
pop-2  stream tcp    nowait root    /usr/sbin/tcpd  ipop2d
pop-3  stream tcp    nowait root    /usr/sbin/tcpd  ipop3d
imap   stream tcp    nowait root    /usr/sbin/tcpd  imapd
.....
```



2 MySQL 한글 정렬 옵션 소스 컴파일 / 인스톨 / 자동 실행 설정

단순하게 IMP만을 쓰기 위해서라면 굳이 MySQL을 이용하지 않아도 주소록과 같은 사용자정보를 단순 텍스트파일을 이용해 서비스할 수 있지만, MySQL이 빠진 PHP는 상상할 수도 없으므로 내친 김에 한글 정렬 옵션을 포함해 MySQL을 컴파일하고 설치하자.

MySQL 설치단계

- /tmp에서 압축 풀기
- configuration
- compile
- install
- 부팅시 자동실행 설정

① /tmp에서 압축 풀기

다운로드받은 최신 버전을 이용해 /tmp에서 설치를 시작한다.

```
[root@roni /tmp]# zcat mysql-3.22.32.tar.gz | tar xvf -
mysql-3.22.32/
mysql-3.22.32/Makefile.in
mysql-3.22.32/README
mysql-3.22.32/stamp-h.in
mysql-3.22.32/Makefile.am
mysql-3.22.32/acconfig.h
....
mysql-3.22.32/support-files/
mysql-3.22.32/support-files/Makefile.in
mysql-3.22.32/support-files/Makefile.am
mysql-3.22.32/support-files/mysql.spec.sh
mysql-3.22.32/support-files/my-example.cnf.sh
mysql-3.22.32/support-files/mysql-log-rotate.sh
mysql-3.22.32/support-files/mysql.server.sh
mysql-3.22.32/support-files/binary-configure.sh
[root@roni /tmp]#
```

② configuration

한글 정렬 옵션 및 설치 디렉토리를 지정한다.

```
# ./configure --prefix=/usr/local/mysql --with-charset=euc_kr
```

③ compile

```
[root@roni /tmp]# make
```

④ Install

```
[root@roni /tmp]# make install
```

⑤ 권한 테이블 생성

```
[root@roni mysql]# cd /usr/local/mysql/bin
[root@roni bin]# ./mysql_install_db
Creating db table
Creating host table
Creating user table
Creating func table
Creating tables_priv table
Creating columns_priv table
To start mysqld at boot time you have to copy support-files/mysql.server
to the right place for your system

PLEASE REMEMBER TO SET A PASSWORD FOR THE MySQL root USER !
This is done with:
/usr/local/mysql/bin/mysqladmin -u root password 'new-password'
See the manual for more instructions.

Please report any problems with the /usr/local/mysql/bin/mysqlbug script!

The latest information about MySQL is available on the web at http://www.mysql.com
Support MySQL by buying support/licenses at http://www.tcx.se/license.htm.
```

⑥ 데이터베이스 기동/ 확인

ps 혹은 pstree 명령을 이용해 MySQL 서버의 데몬이 실행중인지 확인한다.

MySQL 시작 : `./safe_mysqld &`

```
[root@roni bin]# ./safe_mysqld &
[1] 19605
[root@roni bin]# Starting mysqld daemon with databases from /usr/local/mysql/var
[root@roni bin]# pstree
....
-safe_mysqld--mysqld--mysqld--mysqld
....

[root@roni mysql]# ps -aux | grep mysql
root  19650  0.7  1.1 11228 1472 pts/1  SN  17:25  0:00 /usr/local/mysql/
root  19652  0.0  1.1 11228 1472 pts/1  SN  17:25  0:00 /usr/local/mysql/
root  19653  0.0  1.1 11228 1472 pts/1  SN  17:25  0:00 /usr/local/mysql/
root  19657  0.0  0.3 2232  488 pts/1  S   17:25  0:00 grep mysql
```




7 inetd 재시작

```
[root@roni /tmp]# killall -HUP inetd
```

8 imap 서비스 확인

```
# telnet localhost 143
Trying 127.0.0.1...
Connected to localhost.localdomain.
Escape character is '^]'.
* OK localhost.localdomain IMAP4rev1 v12.264 server ready
```

참고 레드햇에서 제공하는 imap rpm은 데몬 프로그램과 man 파일만 있다. 컴파일을 위해서는 헤더와 라이브러리가 필요하므로 반드시 소스를 설치해야 한다.

4 아파치 소스 임시로 풀기

아파치의 디폴트로 정해진 디렉토리는 /usr/local/apache이지만 바로 설치해 버리면 소스 디렉토리는 생략된 채 컴파일된 바이너리와 필요한 파일들만 복사 되므로 다음 단계인 PHP의 컴파일에서 아파치의 헤더와 라이브러리 등을 찾지 못해 컴파일을 진행 할 수 없다.

따라서 임시 디렉토리에 아파치를 잠시 풀어 놓고, PHP 컴파일에서 아파치 관련 파일들을 참고해 컴파일이 완료 되면, 이 파일들을 다시 아파치의 임시 디렉토리에 복사해서 모듈로 사용해 아파치가 컴파일되고 나면 비로소 원래의 디렉토리인 /usr/local/apache에 설치된다.

1 소스풀기

최신 버전의 아파치 배포본이 /tmp에 있다고 가정한다.

```
root@roni /tmp]# zcat apache_1.3.12.tar.gz | tar xvf -
[root@roni /tmp]# cd apache_1.3.12
```

2 두 번의 Configuration

이제 방금 압축을 풀 아파치 디렉토리로 이동해 자신의 하드웨어에 적절한 설정을 하기 위한 스크립트 Configure를 실행하자. 배포판의 압축을 풀 디렉토리에서 한 번 그리고 src 디렉토리에서 한 번, 모두 두 번을 해야 한다는 것을 명심해야 한다. 베이스 디렉토리에서는 아파치의 설치에 관한 설정을, src 디렉토리에서는 아파치 바이너리 httpd에 대한 설정을 하게 된다. src 디렉토리에서는 ./Configure를 실행한다.

```
[root@roni apache_1.3.12]# ./configure
Configuring for Apache, Version 1.3.12
+ Warning: Configuring Apache with default settings.
+ This is probably not what you really want.
+ Please read the README.configure and INSTALL files
```

- + first or at least run './configure --help' for
- + a compact summary of available options.
- + using installation path layout: Apache (config.layout)

Creating Makefile

Creating Configuration.apaci in src

Creating Makefile in src

- + configured for Linux platform
- + setting C compiler to gcc
- + setting C pre-processor to gcc -E
- + checking for system header files
- + adding selected modules
- + checking sizeof various data types
- + doing sanity check on compiler and options

Creating Makefile in src/support

Creating Makefile in src/regex

Creating Makefile in src/os/unix

Creating Makefile in src/ap

Creating Makefile in src/main

Creating Makefile in src/lib/expat-lite

Creating Makefile in src/modules/standard

[root@roni apache_1.3.12]#

[root@roni apache_1.3.12]# cd src/

[root@roni src]# ./Configure

Using config file: Configuration

Creating Makefile

- + configured for Linux platform
- + setting C compiler to gcc
- + setting C pre-processor to gcc -E
- + checking for system header files
- + adding selected modules
- + checking sizeof various data types
- + doing sanity check on compiler and options

Creating Makefile in support

Creating Makefile in regex

Creating Makefile in os/unix

Creating Makefile in ap

Creating Makefile in main

Creating Makefile in lib/expat-lite

Creating Makefile in modules/standard

[root@roni src]#

주의 PHP의 인스톨을 시작하기 전에 반드시 아파치의 configure 스크립트를 실행해 주어야 한다는 것을 명심하기 바란다. 그래야만 PHP 의 setup 스크립트가 아파치의 배포 디렉토리에서 필요한 파일들을 참조할 수 있다.



5 PHP 컴파일 / 인스톨

이번 단계에서는 아파치의 모듈로 사용될 PHP를 만들게 된다. 우선, 최신 버전의 PHP를 /tmp에 옮겨 놓고 작업을 시작하자.

여기서 사용할 파일은 php-3.0.15.tar.gz이지만 워낙 빠르게 버전업되므로 여러분은 좀더 상위버전을 사용하게 될 수도 있다.

① 압축 풀기

```
# zcat php-3.0.15.tar.gz | tar xvf -
```

② 디렉토리 이동

```
# mv ./php-3.0.15 /usr/local/php
```

③ 설정

```
# cd /usr/local/php
# ./setup
```

```
*****
*
* Welcome to the PHP 3.0 setup script. Use this script if you do not
* want or know how to use the configure program. See the INSTALL file
* for further installation instructions.
*
* You will now be asked a series of questions for your installation.
* For each question, your options will be shown in parantheses, and
* for some a default value is listed shown in brackets.
*
*****

.....
Build as an Apache module? (yes', no' or dir) [no] : y
Enter Apache base install directory [/usr/local/etc/httpd] : /tmp/apache_1.3.12
...

MySQL support? (yes', no' or dir) [no] : y
Enter MySQL install directory [/usr/local] : /usr/local/mysql
...

IMAP support? (yes', no' or dir) [no] : y
Enter IMAP base install directory [/usr/local/imap] : /usr/local/imap/include
...

```

```
checking for MySQL support... yes
checking for errmsg.h... yes
checking for mysql.h... yes
```

설정된 옵션들이 제대로 찾아졌는지 잘 확인해야 한다. 디렉토리를 잘못 입력해도 간단히 '찾지 못했다'는 메시지만 나오기 때문에 무심코 지나치면 나중에 unsupported function 오류가 나온다. 이 경우, 함수 이름을 잘못 쓴 경우가 아니라면 대부분 해당 기능을 지원하는 설정이 제대로 안 된 경우이다. 여기서는 MySQL, IMAP, 아파치 부분의 설정만을 확인하면 된다.

* ./setup의 결과, do-conf 파일이 만들어지며 이를 토대로 컴파일과 인스톨이 진행된다.

4 컴파일

```
[root@roni php]# make
```

5 인스톨

컴파일이 끝나면 이제 만들어진 모듈이 아파치의 소스 디렉토리로 복사되도록 한다.

```
[root@roni php]# make install
```

6 설치 확인

make install 스크립트의 실행결과는 간단하다. 컴파일된 아파치 모듈 형태의 PHP를 setup 스크립트에서 설정한 아파치의 소스 디렉토리로 복사하는 것이 전부이다. 정확히 말해, 아파치 디렉토리의 src/modules/ 밑에 php3라는 디렉토리가 만들어 지고 이곳에 모듈이 복사된다. 제대로 복사되었는지 확인해 보자.

파일이 제대로 복사된 것을 확인하였으면 이제 아파치를 컴파일하는 단계로 넘어가자.

```
[root@localhost php]# ls -al /tmp/apache_1.3.6/src/modules/php3/
total 6850
drwxr-xr-x  2 root  root    1024 Oct 24 18:51 .
drwxr-xr-x  8 161  games   1024 Oct 24 18:43 ..
-rw-r--r--  1 root  root     252 Oct 24 18:51 Makefile.libdir
-rw-r--r--  1 root  root    2449 Oct 24 18:51 Makefile.tmpl
-rw-r--r--  1 root  root 6806038 Oct 24 18:51 libmodphp3.a
-rw-r--r--  1 root  root    432 Oct 24 18:51 libphp3.module
-rw-r--r--  1 root  root   21943 Oct 24 18:51 mod_php3.c
-rw-r--r--  1 root  root    12 Oct 24 18:51 mod_php3.exp
-rw-r--r--  1 root  root   3716 Oct 24 18:51 mod_php3.h
-rw-r--r--  1 root  root    29 Oct 24 18:51 php_version.h
```



6 아파치 컴파일 / 인스톨 / 설치 확인 / 자동 실행 설정

① 디렉토리 이동

임시로 풀어 놓은 아파치 디렉토리의 src 디렉토리로 이동한다.

```
(root@localhost php)# cd /tmp/apache_1.3.9/src/  
(root@localhost src)#
```

② Configuration 파일을 수정(libphp3.a 추가)

우선 PHP3 모듈을 포함하도록 Configuration 파일을 수정한다.

편집해야 하는 파일은 Configure가 아니라 Configuration이다.

적지 않은 사람들이 실수하는 부분이다. Configuration 파일은 컴파일 전에 아파치의 설정을 저장하는 파일이고, Configure 스크립트는 Configuration 파일에 따라 적절한 Makefile 을 만드는 역할을 한다. 이것에 따라 컴파일이 이루어진다.

편집기로 Configuration 파일을 열어 적장한 곳에 다음을 추가한다.

```
...  
AddModule modules/php3/libphp3.a  
...
```

주의 modules/php3/libphp3.a 파일은 실제로 존재하지 않는다. 몇몇 사람들은 이것을 잘못된 것으로 착각하고 Configuration 파일의 AddModule modules/php3/libphp3.a 부분을 다른 파일이름으로 변경하는 경우가 있는데, libphp3.a 파일은 아파치의 APACI(APache Advanced configuration mechanism)에 의해 자동으로 생성된다.

③ Configure 실행

변경된 Configuration 파일이 적용되도록 Configure 스크립트를 다시 한 번 실행한다.

추가한 PHP3의 모듈을 인식했음을 확인해야 한다. 이제 아파치의 컴파일을 시작하자.

④ 컴파일

```
#make
```

오류 메시지 없이 프롬프트가 떨어지면 이제 정식으로 만들어진 아파치를 인스톨하자.

한 단계 상위 디렉토리로 이동해 make install 명령을 실행한다.

5 설치 (install)

```
# make install
```

주의 make install은 반드시 아파치 배포판을 풀어 놓은 메인 디렉토리에서 실행해야 한다. 만일 src 디렉토리에서 실행하면 다음과 같은 메시지가 나온다.

```
(root@localhost src)# make install
++ Sorry, no installation procedure available at this level.
++ Go to the parent directory for an 'install' target.
(root@localhost src)#
```

6 httpd.conf 설정 파일의 수정

우선, 아파치의 conf 디렉토리에 있는 httpd.conf 설정 파일을 수정한다.

httpd.conf 설정 파일에 대한 자세한 설정은 뒤에서 다시 설명하고 우선은 PHP3의 사용을 가능하게 하는 옵션만 추가한다

두 줄의 앞부분에 놓은 주석표시(#)를 지운다. 그리고 내친 김에 일반적인 확장자 *.html 과 *.htm , 이전 버전의 PHP/FI 2에서 사용되던 *.phtml과 다음 버전의 PHP4 Zend를 위한 확장자 *.php까지 등록시키자. 만일 *.inc 등과 같이 특정한 확장자의 인클루드 파일을 사용한다면 그 파일의 확장자까지 등록시키자.

```
...
#AddType application/x-httpd-php3 .php3
#AddType application/x-httpd-php3-source .phps
...
```

참고로, *.phps 는 PHP 스크립트의 소스를 나타내는 확장자이다. 자동으로 문자열을 컬러로 보여주며, php3.ini 파일 등을 통해 설정을 변경할 수 있다.

팁 스크립트 소스로 보여주기

웹상에서 PHP 스크립트의 소스를 보여주기 위해서는 보여주려는 파일 *.phps로 복사해서 보여줄 수도 있지만 원래 스크립트를 수정하는 경우에는 다시 복사 해야 한다는 번거로움이 있다. 이럴 때에는 심볼릭 링크를 이용하면 아주 쉽게 해결된다. 예를 들어, test.php3 파일의 소스를 나타내려면

```
# ln -s test.php3 test.p:ps
```

위와 같이 하면 항상 test.php3의 소스는 test.php를 통해 웹상에서 볼 수 있다.

7 apachectl 웹 서버 기동

```
(root@localhost bin)# cd /usr/local/apache/bin/
(root@localhost bin)# ./apachectl start
./apachectl start: httpd started
(root@localhost bin)#
```



8) 아파치 중지

```
[root@localhost bin]# ./apachectl stop
./apachectl stop: httpd stopped
[root@localhost bin]#
```

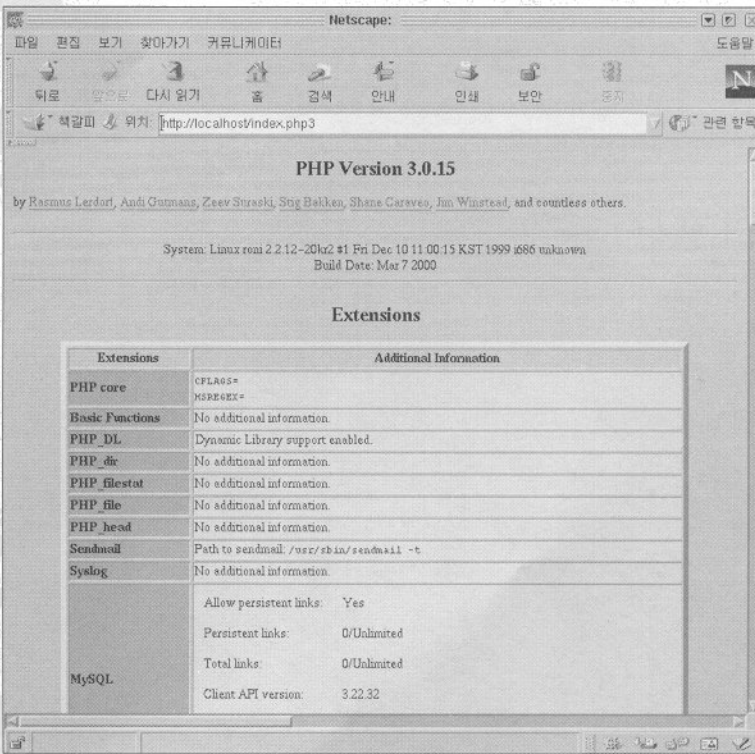
9) 아파치 서비스 확인

브라우저를 열어 URL 부분에 http://localhost를 입력한다

10) PHP 동작확인

아래와 같은 파일을 만들어 브라우저를 통해 확인한다.

```
?php
phpinfo();
?>
```



11) 아파치와 MySQL의 부팅시 자동으로 실행시키기

서버를 재부팅할 때마다 일일이 아파치와 MySQL을 실행시키지 않고 자동으로 실행하기 위해서는 /etc/rc.d/rc.local 파일을 열어 다음을 추가한다.

```
...
/usr/local/apache/bin/apachectl start
/etc/rc.d/init.d/mysql.server start
```

7 Horde / IMP 설치 / 테스트

① horde*.tar.gz와 imp*.tar.gz 파일을 다운받는다

② 적당한 디렉토리에 압축을 푼다. /tmp를 이용하겠다

```
# zcat horde-1.0.11.tar.gz | tar xvf -
```

③ Horde 모듈을 적당한 웹 디렉토리로 옮긴다

```
# mv /tmp/horde-1.0.11 /usr/local/apache/htdocs/horde
```

④ IMP의 압축을 풀고 역시 적당한 웹 디렉토리로 옮긴다

다음과 같은 디렉토리구조가 되도록 해야한다.

```
horde/
```

```
horde/imp
```

```
# zcat imp-2.0.11.tar.gz | tar xvf -
```

```
# mv imp-2.0.11 /usr/local/apache/htdocs/horde/imp
```

⑤ MySQL 데이터베이스의 생성

/usr/local/apache/htdocs/horde/imp/config/scripts/mysql_create.sql 스크립트를 다음과 같이 실행한다.

```
[root@roni scripts]# cd /usr/local/apache/htdocs/horde/imp/config/scripts
```

```
[root@roni scripts]# /usr/local/mysql/bin/mysql < mysql_create.sql
```

스크립트의 내용은 다음과 같다.

```
=====
=====
# mysql_create_all.sql
# All the ver1 mysql scripts rolled into one
# you can simply direct this file to mysql as STDIN:
# mysql (user/pass/host args) < mysql_create.sql
#

use mysql

INSERT INTO user ( host, user, password )
VALUES (
    'localhost',
    'impmgr',
    password( 'impmgr' )
);
```




```
INSERT INTO db (
  host, db, user,
  Select_priv, Insert_priv, Update_priv, Delete_priv,
  Create_priv, Drop_priv )
VALUES (
  'localhost',
  'imp',
  'impmgr',
  Y, Y, Y, Y,
  Y, Y
);
```

```
CREATE DATABASE imp;
use imp
```

```
# create the address table
CREATE TABLE imp_addr (
  user text,
  address text,
  nickname text,
  fullname text
);
```

```
# now create the pref table
CREATE TABLE imp_pref (
  user text,
  fullname text,
  replyto text,
  lang varchar(30),
  sig text
);
```

```
# done!
```

```
=====
```

⑥ MySQL 서버를 다시 시작한다

```
# /usr/local/mysql/bin/mysqladmin reload
```

다음과 같이 확인해 볼 수 있다.

```
# /usr/local/mysql/bin/mysql mysql
mysql> select * from user ;
mysql> select * from db ;
```

imp 데이터베이스, impmgr 사용자가 만들어 졌음을 알 수 있다.

⑦ 브라우저를 통한 12단계 설정

이제 브라우저를 통해 셋업을 하자. 먼저, setup.php3 파일에 실행권한을 주고 빈 설정 파일을 생성하는 스크립트를 실행한다.

```
[root@linuxer imp]# cd /usr/local/apache/htdocs/horde
[root@linuxer horde]# ./install.sh

Your blank configuration files have been created, please go to
the configuration utility at :
your install path url/setup.php3
[root@linuxer horde]#
```

그리고 나서 브라우저를 열어 setup.php3를 연다. 12단계의 설정화면이 나온다.

URL : <http://localhost/horde/setup.php3>

<12단계의 설정 내용>

1. Language Configuration
2. Server Specific Configuration
3. Web Server Configuration
4. Mail Server Configuration
5. Cookie Configuration
6. External Binaries Configuration
7. User abilities Configuration
8. New Mail Notification Configuration
9. Message Configuration
10. Database Configuration
11. Confirm File Write - This is what your defaults.php3 will look like.
12. Configuration Complete!

설정이 끝나면 설정한 내용을 파일에 저장하게 된다.

⑧ secure.sh 의 실행

이제 다시 setup.php3 이 브라우저를 통해 실행되지 않도록 secure.sh 을 실행한다.

```
[root@linuxer horde]# ./secure.sh
I have made your configuration files, and libraries mode 0555
which is read / execute for everyone.
And the setup.php3 is mode 0000 which is no access period.
[root@linuxer horde]#
```

이제 IMP를 사용해 볼수 있다.

로그인 페이지의 URL을 다음과 같다.

URL : <http://localhost/horde/imp/index.php3>



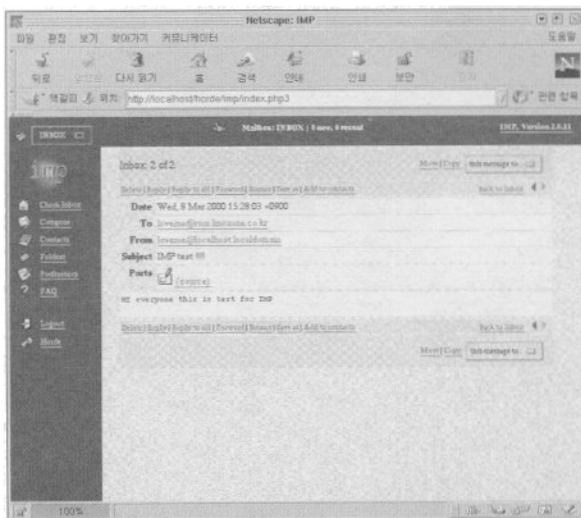
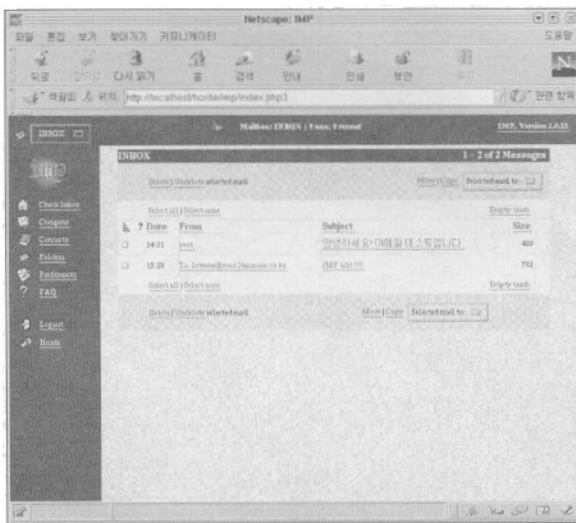
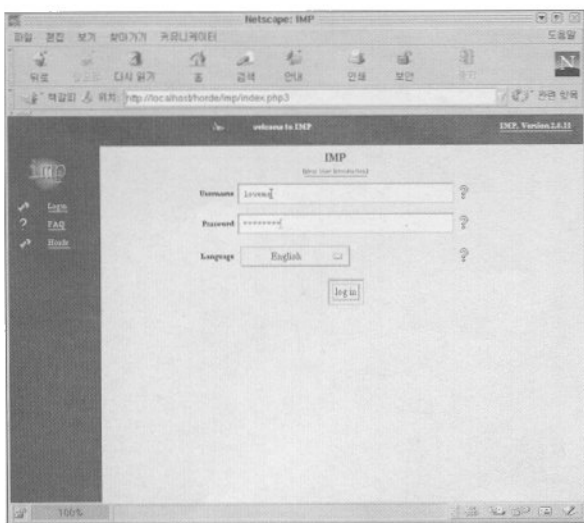
참고 index.php3 를 시작페이지로 만들기

index.html 이 없을 경우 디렉토리가 인덱싱되어 나오게 되므로 index.php3를 두번째 인덱스 페이지로 지정해 줄 수 있다. 아래의 아파치 설정 파일을 수정한다.

```

/usr/local/apache/conf/httpd.conf
...
<IfModule mod_dir.c>
    DirectoryIndex index.html
</IfModule>
...
    
```

이제 각 디렉토리에서 index.html 이 없을 경우 index.php3 가 실행될 것이다.



8 yamail 설치하기

/tmp 에서 작업한다고 가정한다.

① 압축 풀기

```
# zcat yamail-1999-04-08.tar.gz | tar xvf -
```

② 디렉토리 복사

```
# cp -r yamail-1999-04-08/yamail-1999-04-08 /usr/local
```

③ 디렉토리 이동

```
# cd /usr/local/
```

④ 링크 만들기

```
# ln -s yamail-1999-04-08 yamail
# vi /usr/local/yamail/config/defaults.phtml
```

⑤ 설정 파일 수정

필요에 따라 아래 파일들도 수정

```
/usr/local/yamail/config/department.phtml (optional)
/usr/local/yamail/config/sysmsg.phtml (optional)

{{{#cd /usr/local/yamail/lib/scripts
[root@roni scripts]# /usr/local/mysql/bin/mysql < mysql-common.sql
[root@roni scripts]# /usr/local/mysql/bin/mysql < mysql-3.22.sql
[root@roni scripts]# /usr/local/mysql/bin/mysqladmin reload
[root@roni scripts]#
—
```

⑥ MySQL에 테이블 생성

```
# cd /usr/local/
# chown -R nobody yamail
```

⑦ 디렉토리의 소유권 변경

```
#cd /usr/local/yamail/tool
#perl mk_Mail_dir.pl
```



⑧ User 홈 디렉토리에 메일 디렉토리 및 메일 박스 생성

```
....  
<Directory /usr/local/yamail>  
AllowOverride None  
Options Indexes FollowSymLinks  
</Directory>  
....  
Alias /yamail/ "/usr/local/yamail/"
```

⑨ 아파치 설정 파일 변경 - /yamail 디렉토리 Alias 설정

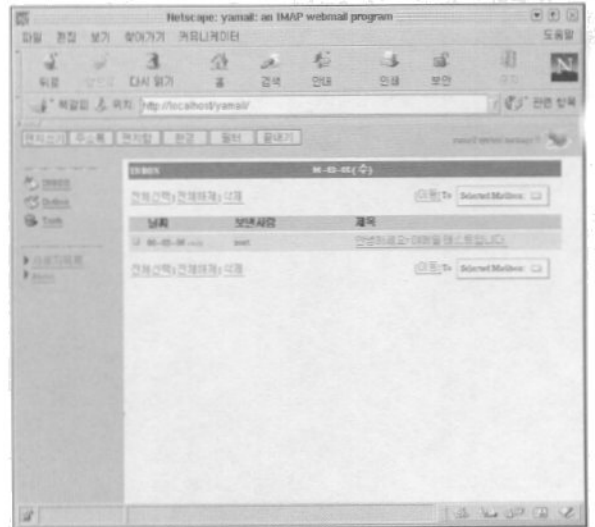
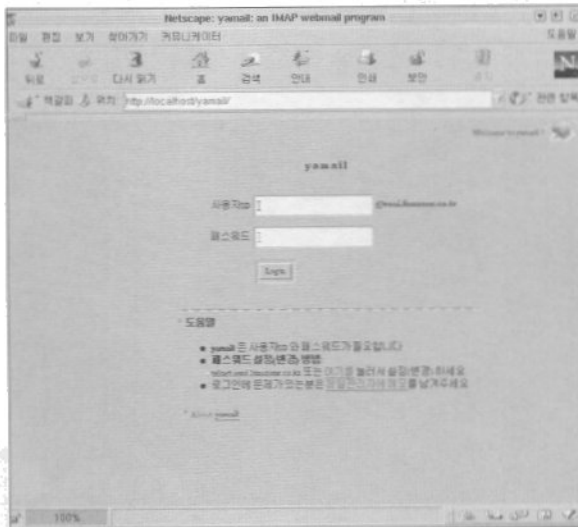
파일명 : /usr/local/apache/conf/httpd.conf

⑩ 아파치 데몬 재시작

```
# killall -HUP httpd 또는  
# /usr/local/apache/bin/apachectl restart
```

⑪ Yamail 로그인 및 테스트

http://localhost/yamail/ 반드시 (/) 를 붙인다.



3

▶ 끝내며

이상으로 간략하게 공개형 소스를 이용한 웹메일 서비스를 구현해 보았다. 아직까지 기능상으로 상용 웹메일 서비스에 뒤지는 감이 있으나 소스가 공개되어 있다는 사실 하나만으로도 충분한 가치가 있다고 생각된다.

실제 PHP는 수많은 IMAP 관련함 수를 지원하므로 관심있는 분은 한번쯤 어떤 것들이 있나 살펴보는 것도 큰 도움이 될 것이다.

개인적인 사정으로 보다 충실한 교재가 되지 못한 점 이 글을 읽는 모든 분들께 사과 드리며 조만간 국내에서도 오픈 소스로 진행되는 웹메일 프로젝트가 시작되기를 바라며 이만 두서없는 글을 마무리 지을까 한다.

여러분 부디 행복한 LINUXing 과 PHPing 하시기를...