

<Revision 정보>

일자	VERSION	변경내역	작성자
2007. 07. 18	0.1	초기 작성	박준완

솔루션 개발지원

Hobak Webmail Solution Migration 2차

리눅스원(주)

한국소프트웨어진흥원

공개SW기술지원센터

목 차

1. 문서 개요	4
가. 문서의 목적	4
나. 본 문서의 사용방법	4
다. 참고 사항	4
2. Hobak Webmail Migration	5
가. 개요	5
나. Version	5
다. 변경할 사항	5
라. 개발 내용	6
마. 비교	13

< 그림 차례 >

1. Text-conv-1.2 파일 수정 내용	6
2. MIME-Base64-2.18 수정부분(1~4)	7-10
3. MANIFEST 수정부분	10
4. PHP와 MySQL의 테이블 생성 수정부분	11
5. Auth 관련 DB 수정부분(1~2)	12-13

< 표 차례 >

1. Version 비교	5
---------------------	---

1. 문서 개요

본 문서는 KIPA 공개S/W 기술지원 센터에서의 솔루션 개발지원의 결과를 보고하기 위해 제작되었다.

가. 문서의 목적

다음과 같은 세부적인 목적을 달성하기 위하여 작성되었다.

0 Hobak Webmail의 MTA인 Qmail을 Sendmail로 Migration

0 Migration을 통한 버전업 및 성능 향상

나. 본 문서의 사용방법

다음과 같은 방법으로 사용할 수 있다.

0 Qmail을 Sendmail로 Migration하는 작업에 대한 참조 문서로 사용한다.

0 개발 지원 결과 내역을 참조한다.

다. 참고사항

0 개발지원시 시스템 관련사항, 기술적 배경 등 참고사항들을 기술한다.

2. Hobak Webmail Migration

가. 개요

현재 리눅스원(주)의 웹메일 솔루션인 Hobak Webmail은 Qmail로 구현되어 있다. 그러나 버전이 낮아 보안 및 버그에 대해 취약하며, 대용량 메일 솔루션으로 적합하지 않아 SMTP를 Sendmail로 변경하면서 버전을 upgrade하고 보안 및 버그에 대한 업데이트를 통해 최적화된 솔루션으로 Migration하는 것이 목적이다. 이번 문서는 1차 개발 지원에 이은 2차 개발 지원서이다.

나. Version

목차	Before Migration	After Migration
SMTP	Qmail or Postfix	Sendmail
기능	<ul style="list-style-type: none"> ● 자체 라이브러리(libHBMail)제작 ● Hash 함수를 이용한 메일박스 관리 ● DRAC를 이용한 SMTP인증 ● 콘솔 기반의 인스톨러 ● 주소록 ● 웹폴더 ● 기타 	추가된 부분 <ul style="list-style-type: none"> ● libHBMail 수정 및 보완 ● 메일 박스의 유연한 확장 및 클러스터 기능 ● 메일 필터링 기능(procmail, sieve) ● 일정관리 ● 사용자관리 웹인터페이스 ● 공지사항, 게시판 기능 ● 기타
개발언어	C, Perl, PHP, SQL, HTML, JavaScript, CSS	
SMTP Version	qmail-1.0.3	Sendmail-8.13.8

<표 1> Version 비교

다. 변경할 사항

- I) Install
- II) packaging
- III) MTA Change - Daemon
 - Interface 연동
 - 기능 대체
- IV) UI Change
- V) Manual
- VI) MySQL 연동

라. 개발 내용

I) 2차 개발 지원 내용

2차 개발 지원에서는 1차 개발 지원에 완성하지 못한 여러 가지 부분들을 완성하였으며, 모듈간의 의존성이나, 기능에 대한 검증 등을 중심으로 안정성을 고려할 수 있는 부분들을 지원하였다.

```
package Text::Iconv;
# @(#) $Id: Iconv.pm,v 1.3 2001/08/11 10:10:04 mxp Exp $
# Copyright (c) 2000 Michael Piotrowski

use strict;
use vars qw($VERSION @ISA @EXPORT @EXPORT_OK);

require Exporter;
require DynLoader;
require AutoLoader;

@ISA = qw(Exporter AutoLoader DynLoader);
# Items to export into callers namespace by default. Note: do not export
# names by default without a very good reason. Use EXPORT_OK instead.
# Do not simply export all your public functions/methods/constants.
@EXPORT_OK = qw(
    convert
);
$VERSION = '1.2';

bootstrap Text::Iconv $VERSION;

# Preloaded methods go here.

# Autoload methods go after =cut, and are processed by the autosplit program.

1;
__END__
# Below is the documentation for the module.

=head1 NAME

Text::Iconv - Perl interface to iconv() codeset conversion function

=head1 SYNOPSIS

    use Text::Iconv;
    $converter = Text::Iconv->new("fromcode", "toencode");
    $converted = $converter->convert("Text to convert");

=head1 DESCRIPTION
```

<그림 1> Text-conv-1.2 파일 수정 내용


```

/* set up EOL from the second argument if present, default to "\n" */
if (items > 1 && SvOK(ST(1))) {
    eol = SvPV(ST(1), eollen);
} else {
    eol = "\n";
    eollen = 1;
}

/* calculate the length of the result */
rlen = (len+2) / 3 * 4; /* encoded bytes */
if (rlen) {
    /* add space for EOL */
    rlen += ((rlen-1) / MAX_LINE + 1) * eollen;
}

/* allocate a result buffer */
RETVAL = newSV(rlen ? rlen : 1);
SvPOK_on(RETVAL);
SvCUR_set(RETVAL, rlen);
r = SvPVX(RETVAL);

/* encode */
for (chunk=0; len > 0; len -= 3, chunk++) {
    if (chunk == (MAX_LINE/4)) {
        char *c = eol;
        char *e = eol + eollen;
        while (c < e)
            *r++ = *c++;
        chunk = 0;
    }
    c1 = *str++;
    c2 = *str++;
    *r++ = basis_64[(c1 >> 2)];
    *r++ = basis_64[((c1 & 0x3) << 4) | ((c2 & 0xf0) >> 4)];
    if (len > 2) {
        c3 = *str++;
        *r++ = basis_64[((c2 & 0xf) << 2) | ((c3 & 0xc0) >> 6)];
        *r++ = basis_64[c3 & 0x3f];
    } else if (len == 2) {
        *r++ = basis_64[(c2 & 0xf) << 2];
        *r++ = '=';
    } else { /* len == 1 */

```

<그림 2-3> MIME-Base64-2.18 수정부분

```

        *r++ = '=';
    } else { /* len == 1 */
        *r++ = '=';
        *r++ = '=';
    }
}

if (rlen) {
    /* append eol to the result string */
    char *c = eol;
    char *e = eol + eollen;
    while (c < e)
        *r++ = *c++;
}

*r = '\0'; /* every SV in perl should be NUL-terminated */

#line 200 "Base64.c"
ST(0) = RETVAL;
sv_2mortal(ST(0));
}
XSRETURN(1);
}

```

<그림 2-4> MIME-Base64-2.18 수정부분

base64.pm	Bootstrap and perl implementation of MIME::Base64
Base64.xs	Encoder and decoder written in C
Changes	What's new
MANIFEST	This file
Makefile.PL	Writes the Makefile
QuotedPrint.pm	MIME::QuotedPrint module, encoder and decoder
README	Please do!
encode-qp	Encode files as Quoted-Printable
decode-qp	Decode Quoted-Printable files
encode-base64	Encode files as Base64
decode-base64	Decode Quoted-Printable files
t/base64.t	Test that it works
t/quoted-print.t	
t/unicode.t	
t/warn.t	

<그림 3> MANIFEST 수정부분


```
--  
-- Table structure for table 'board1'  
--  
create database PHPBOARD;  
use PHPBOARD;  
  
CREATE TABLE board1 (  
  uid mediumint(9) unsigned NOT NULL auto_increment,  
  fid mediumint(9) unsigned NOT NULL default '0',  
  name varchar(12) NOT NULL default '',  
  email varchar(40) default NULL,  
  homepage varchar(60) default NULL,  
  subject varchar(60) NOT NULL default '',  
  comment text NOT NULL,  
  passwd varchar(30) NOT NULL default '',  
  signdate int(10) unsigned NOT NULL default '0',  
  ref smallint(5) unsigned NOT NULL default '0',  
  thread varchar(255) NOT NULL default '',  
  PRIMARY KEY (uid)  
) TYPE=MyISAM;
```

<그림 4> PHP와 MySQL의 테이블 생성 수정부분

```
-- Table structure for table 'tblUserInfo'  
--  
create database Auth;  
use Auth;  
  
CREATE TABLE tblUserInfo (  
  lpszUserID varchar(100) NOT NULL default '',  
  nUserID int(16) NOT NULL auto_increment,  
  rchPassword varchar(48) NOT NULL default '',  
  nUserGID int(16) NOT NULL default '0',  
  lpszUserName varchar(50) NOT NULL default '',  
  nUserQuota int(16) NOT NULL default '0',  
  nUserStatus char(1) NOT NULL default '',  
  lpszUserSuffix varchar(8) NOT NULL default '',  
  rchPassNoCrypt varchar(24) NOT NULL default '',  
  datJoinDate date NOT NULL default '0000-00-00',  
  PRIMARY KEY (lpszUserID),  
  UNIQUE KEY nUserID (nUserID)  
) TYPE=MyISAM;  
  
CREATE TABLE userdb (  
  id char(50) NOT NULL default '',  
  passwd char(50) NOT NULL default '',  
  name char(30) default NULL,  
  flag char(1) default NULL,  
  engname char(30) default NULL,  
  sex char(3) default NULL,  
  phone char(15) default NULL,  
  phone2 char(15) default NULL,  
  birth char(15) default NULL,  
  social char(15) default NULL,  
  zipcode char(7) default NULL,  
  addr char(100) default NULL,  
  job char(30) default NULL,  
  company char(30) default NULL,
```

<그림 5-1> Auth 관련 DB 수정부분

```
division char(30) default NULL,
college char(15) default NULL,
major char(30) default NULL,
hakbun char(15) default NULL,
grade1 char(2) default NULL,
grade2 char(10) default NULL,
assort char(8) default NULL,
email char(100) default NULL,
lpszUserSuffix char(8) NOT NULL default '',
joinDate date default NULL,
PRIMARY KEY (id)
) TYPE=MyISAM;
```

<그림 5-2> Auth 관련 DB 수정부분

마. 비교

1차 개발 지원 내용은 Hobak Webmail의 library와 연동이 가능한 헤더 파일 수정과 Sendmail과 DB를 연동 구현하기 위한 SendmailDB.sql 파일에 대한 개발 작업이 진행되었으며, 그림 첨부된 부분 이외에도 많은 부분에 대해 수정 작업이 이루어 졌다. 1차 개발 지원에서 중점적으로 진행 했던 부분은 Qmail을 참조하여 진행하는 여러 파일들과 library들을 Sendmail을 참조 할 수 있도록 변경 한 부분이며, Core에 접근하는 부분들에 대한 수정도 이루어 졌다. 1차까지 Migration 진행 상황은 약 15% 진행되었다. 2차 개발 지원 내용은 안정성을 중심으로 개발 지원을 진행하였다. 인증 부분을 포함한 DB와 PHP(Application) 간의 연동 및 테이블 생성에 대한 수정 부분이나, Architecture에 따른 변동 사항을 반영할 수 있는 부분등 Application의 안정성을 고려한 개발 지원을 진행하였으며, 현재 약 60% 진행이 완료되었다.

향후 개발 계획은 사용자 인터페이스 보완과 함께, Core 개발 마무리, 새로운 기능 추가 부분이 남아 있으며, 금년 말 완료를 목표로 개발을 진행하고 있다.