

본 자료는 슈퍼유저코리아에서 출판한 “리눅스 실무명령어 바이블 (전 2 권)”의 본문중 일부  
분이며 책 구입희망자분들께서 공개요청하셨던 부분중 요구횟수가 가장 많았던 부분만을 발  
췌하여 공개한 것입니다. 비록 일부분이지만 유용하게 사용하시기 바랍니다.

# 리눅스 실무명령어 바이블 1, 2 편

## 53 장. crontab 편

### 53 장. crontab 목 차 (책 실제페이지 : 260 페이지)

- 실무명령어# 53-1 절 : 주기적이고 반복적인 cron 설정작업을 할 수 있는 crontab 개론
- 실무명령어# 53-2 절: 시스템 크론데몬(cron)의 기본 동작원리 분석
- 실무명령어# 53-3 절: 일반사용자의 crontab 명령어 사용허가 또는 제한하는 설정
- 실무명령어# 53-4 절: “분, 시, 일, 월, 요일, 실행명령” 순으로 cron 설정하는 방법
- 실무명령어# 53-5 절: crontab 으로 설정한 개별사용자의 cron 설정, 파일로 확인하기
- 실무명령어# 53-6 절: root 권한으로 root 의 cron 설정하는 방법
- 실무명령어# 53-7 절: root 권한으로 일반 사용자의 cron 설정하는 방법
- 실무명령어# 53-8 절: root 권한으로 root 의 cron 설정내용 확인하는 방법
- 실무명령어# 53-9 절: root 권한으로 일반 사용자의 cron 설정내용 확인하는 방법
- 실무명령어# 53-10 절: 일반사용자의 자기자신의 cron 설정하는 방법
- 실무명령어# 53-11 절: 일반사용자의 cron 설정 확인하는 방법
- 실무명령어# 53-12 절: root 권한으로 root 의 cron 설정 삭제하기
- 실무명령어# 53-13 절: 일반계정사용자의 자기 자신의 cron 설정 삭제하기
- 실무명령어# 53-14 절: root 권한으로 일반계정사용자의 cron 설정 삭제하기

- 본 자료의 저작권은 “슈퍼유저코리아” ([www.superuser.co.kr](http://www.superuser.co.kr))에 있습니다 .
- 본자료는 “슈퍼유저코리아”에서 출판한 『리눅스 실무명령어 바이블(1 편, 2 편)』의 실제  
내용중 일부분으로서 책의 본문내용을 확인하고자하시는 분들의 요청에 의해 요청횟수가  
많았던 부분만을 일부 공개한 것입니다.
- 본 책자의 자세한 모든 정보는 <http://www.superuser.co.kr/linuxcommandbible/> 에서  
확인하시기 바랍니다.
- 본 자료의 수정 및 편집은 허용되지않습니다.
- 본 책은 권수 : 2 권 (1 편, 2 편), 페이지수 : 총 1,859 페이지, 장(chapter): 382 장,  
절: 1,900 개의 절로 구성되어있습니다.
- 저작권 및 문의전화(사무실) : TEL : 051-583-8128, FAX : 051-583-8130

### 실무명령어# 53-1 : 주기적이고 반복적인 cron 설정작업을 할 수 있는 crontab 개론

리눅스 서버관리를 하는 서버관리자에게 있어서 crontab 명령어 만큼 고마운 명령어는 없을  
것입니다. 왜냐하면 직접 작업을 하지않아도 작업설정을 예약해두고 주기적으로 반복실행할  
수 있기 때문입니다. 즉, 백업이나 서버점검등과 같은 주기적이고 반복적인 작업을 crontab  
명령어로 설정해두면 정해진 시간에 주기적으로 자동실행하기 때문에 우리 같은 서버관리자  
들에게는 얼마나 큰 위안과 힘이 되는지 모릅니다.

첨언한다면 crontab 의 설정방법과 함께 셸프로그래밍의 기본제작법을 익힌다면 여러분들은  
바로 고급관리자로 인정받을 수 있을 것입니다. 이 책에서는 셸프프로그램에 대해서도 각각  
의 명령어편에서 다루고 있으므로 참고하시고 이번 장에서는 시스템의 크론설정에 대해서  
실무적인 사례를 들어 그 설정법을 배워보도록 하겠습니다.

즉, 필자는 주기적인 작업을 할 수 있는 크론(cron)설정에 대해서 많은 질문을 받아왔는데  
이번 기회에 시스템의 주기적이고 반복적인 작업설정을 하는 크론(cron)설정법의 실무적인



본 자료는 슈퍼유저코리아에서 출판한 “리눅스 실무명령어 바이블 (전 2 권)”의 본문중 일부  
분이며 책 구입희망자분들께서 공개요청하였던 부분중 요구횟수가 가장 많았던 부분만을 발  
췌하여 공개한 것입니다. 비록 일부분이지만 유용하게 사용하시기 바랍니다.

# 리눅스 실무명령어 바이블 1,11 편

방법들과 사례를 위주로 살펴보도록 하겠습니다.

간략히 표현하면 crontab 은 시스템의 주기적인 cron 작업의 설정, 수정등을 하는 명령어입  
니다. 시스템에서 가장 중요한 데몬(daemon)중의 하나인 crond 와 직접적인 관련이 있는 명  
령어로서 슈퍼유저(root) 또는 일반사용자들의 주기적이고 반복적인 특정 작업의 자동화를  
설정하는 명령어입니다.

이 명령어로 설정한 작업이 주기적으로 실행되려면 crond 데몬이 실행상태에 있어야합니다.  
그리고 crond 데몬은 /etc/rc.d/init.d/crond 스크립트에 의해 시작, 종료, 재시작될 수 있  
습니다.

따라서 crontab 이라는 명령어를 정확하게 이해하고 사용하려면 다음과 같은 데몬과 파일들  
을 정확하게 이해하셔야 합니다.

- crond 데몬
- /etc/crontab 파일
- /etc/rc.d/init.d/crond 스크립트
- /var/spool/cron 디렉토리내의 크론설정파일들

명령어위치 : /usr/bin/crontab

사용형식

```
crontab [ -u 사용자 ID ] 파일  
crontab [ -u 사용자 ID ] { -l | -r | -e }
```

이번 장에서는 root 와 개별 사용자들의 cron 설정방법과 수정,삭제, 확인하는 방법, 그리고  
위의 파일들이 유기적으로 상호작용하여 작동하는 시스템 cron 설정의 전반적인 내용들에  
대해서 알아볼 것입니다.

## 실무명령어# 53-2 : 시스템 크론데몬(crond)의 기본 동작원리 분석

### 시스템 크론데몬 crond 실행 확인하기

crontab 명령어의 사용법을 설명하기에 앞서 시스템에서 cron 이 어떻게 동작하는가를 알아  
야합니다. 시스템에서 cron 이 제대로 동작하기 위해서는 crond 라는 크론데몬(cron daemon)  
이 동작하고 있어야만 합니다.

먼저 ps 명령어로 crond 가 동작중인가를 확인한 예입니다.

아래의 결과를 보시면 crond 라는 데몬이 root 권한으로 실행되어 있음을 알 수가 있습니다.

```
[root@file init.d]# ps -ef | grep crond  
root      1805      1   0 01:33 ?          00:00:00 crond  
root      17746    1956   0 04:49 pts/0    00:00:00 grep crond  
[root@file init.d]#
```

본 자료는 슈퍼유저코리아에서 출판한 “리눅스 실무명령어 바이블 (전 2 권)”의 본문중 일부  
분이며 책 구입희망자분들께서 공개요청하였던 부분중 요구횟수가 가장 많았던 부분만을 발  
췌하여 공개한 것입니다. 비록 일부분이지만 유용하게 사용하시기 바랍니다.

## 리눅스 실무명령어 바이블 1,11 편

이렇게 crond가 동작중이면 시스템에 기본적으로 설정된 cron 설정과 root 의 cron 설정, 그  
리고 개별사용자의 cron 설정들이 정상적으로 동작하게 됩니다.

### 시스템 크론데몬 crond 실행, 중지, 재시작하기

시스템 크론데몬인 crond를 실행,중지,재시작하려면 /etc/rc.d/init.d/crond 스크립트를 이  
용합니다.

먼저 crond 를 시작하기 위해서는 “/etc/rc.d/init.d/crond start” 라고 하시면 됩니다.  
거의 대부분 리눅스 시스템이 부팅이 되면서 ntsysv 설정항목에 의해 crond 가 시작되도록  
설정되어 있으므로 crond는 부팅과 함께 자동 실행될 것입니다.(이책의 ntsysv 명령어편 참  
조바람)

```
[root@file init.d]# /etc/rc.d/init.d/crond start
crond (을)를 시작합니다: [ 확인 ]
[root@file init.d]#
[root@file init.d]#
[root@file init.d]# /etc/rc.d/init.d/crond restart
crond 를 중지함: [ 확인 ]
crond (을)를 시작합니다: [ 확인 ]
[root@file init.d]#
[root@file init.d]#
[root@file init.d]#
[root@file init.d]# /etc/rc.d/init.d/crond stop
crond 를 중지함: [ 확인 ]
[root@file init.d]#
```

그리고 crond 를 재시작하려면 위의 예와 같이 “/etc/rc.d/init.d/crond restart” 라고 하  
시면 됩니다. 또한 crond 의 실행을 중지하려면 위의 예와 같이 “/etc/rc.d/init.d/crond  
stop” 이라고 하시면 됩니다.

### 시스템의 기본 cron 설정파일 /etc/crontab 의 실행내용 분석

시스템 크론데몬인 crond 는 /etc/crontab 파일의 설정을 읽어들이어서 설정되어 있는대로 실행  
을 합니다. 아래의 /etc/crontab 파일은 리눅스를 설치하면 기본적으로 설정되어 있는  
내용으로서 시스템에 기본적으로 필요한 사항들을 주기적으로 자동실행하기 위한 설정사항  
입니다.

즉, /etc/crontab 파일은 시스템 크론데몬인 crond 에 의해 실행될 시스템관리에 꼭 필요한  
주기적인 자동실행내용들이 저장되어 있는 파일입니다. 이 파일의 기본설정내용을 살펴보면  
다음과 같습니다.

본 자료는 슈퍼유저코리아에서 출판한 “리눅스 실무명령어 바이블 (전 2 권)”의 본문중 일부 분이며 책 구입희망자분들께서 공개요청하셨던 부분중 요구횟수가 가장 많았던 부분만을 발췌하여 공개한 것입니다. 비록 일부분이지만 유용하게 사용하시기 바랍니다.

## 리눅스 실무명령어 바이블 1, II 편

```
[root@file root]# cat /etc/crontab
SHELL=/bin/bash
PATH=/sbin:/bin:/usr/sbin:/usr/bin
MAILTO=root
HOME=/

# run-parts
01 * * * * root run-parts /etc/cron.hourly
02 4 * * * root run-parts /etc/cron.daily
22 4 * * 0 root run-parts /etc/cron.weekly
42 4 1 * * root run-parts /etc/cron.monthly
[root@file root]#
```

즉, 리눅스 시스템은 /etc/crontab 파일의 설정에 의해 시스템관리를 위하여 다음과 같이 주기적인 실행을 하게 됩니다. 이 파일의 주된 설정내용을 살펴보면 다음과 같습니다.

매시 1 회 자동실행하기위한 시스템 크론설정

```
01 * * * * root run-parts /etc/cron.hourly
```

/etc/crontab 파일내에 존재하는 이 내용은 매일 매시 01 분마다 /etc/cron.hourly 디렉토리 내에 존재하는 파일들을 실행하게 됩니다.

즉, /etc/cron.hourly 디렉토리에서 실행되는 파일들은 다음과 같습니다. 즉, 매시 01 분마다 실행될 내용이 /etc/cron.hourly 디렉토리에 저장되어 있는 것입니다. 아래 파일들은 여러분들께서 직접 그 내용을 확인해 보시기 바랍니다.

```
[root@file root]# ls -l /etc/cron.hourly/
합계 8
-rwxr-xr-x 1 root root 113 1월 27 2003 inn-cron-rntpsend
-rwxr-xr-x 1 root root 113 1월 27 2003 inn-cron-meus
[root@file root]#
```

만약 여러분들께서 매시 01 분마다 실행시키고자 하는 내용이 있다면 /etc/cron.hourly 디렉토리에 스크립트파일로 만들어서 넣어두시면 됩니다.

매일 1 회 자동실행하기위한 시스템 크론설정

```
02 4 * * * root run-parts /etc/cron.daily
```

/etc/crontab 파일내에 존재하는 위의 내용은 매일 새벽 4 시 02 분마다 /etc/cron.daily 디렉토리내에 존재하는 파일들을 실행하게 됩니다. 즉, 매일 한번씩 실행될 내용이 저장될 디렉토리이며 이 디렉토리에서 실행되는 파일들은 다음과 같습니다.

본 자료는 슈퍼유저코리아에서 출판한 “리눅스 실무명령어 바이블 (전 2 권)”의 본문중 일부  
분이며 책 구입희망자분들께서 공개요청하였던 부분중 요구횟수가 가장 많았던 부분만을 발  
췌하여 공개한 것입니다. 비록 일부분이지만 유용하게 사용하시기 바랍니다.

## 리눅스 실무명령어 바이블 1,11 편

```
[root@file root]# ls -l /etc/cron.daily/  
합계 36  
-rwxrwxrwx    1 root    root          28 10월 28 01:38 00-logwatch -> ../log  
-rwxr-xr-x    1 root    root         135 1월 26 2003 00webalizer  
-rwxr-xr-x    1 root    root         276 1월 25 2003 0anacron  
-rwxr-xr-x    1 root    root         123 1월 27 2003 inn-cron-expire  
-rwxr-xr-x    1 root    root          51 1월 25 2003 logrotate  
-rwxr-xr-x    1 root    root         418 2월 11 2003 makewhatiscron  
-rwxr-xr-x    1 root    root         104 2월 28 2003 rpm  
-rwxr-xr-x    1 root    root         132 2월 20 2003 slocate.cron  
-rwxr-xr-x    1 root    root         100 2월 19 2003 tetex.cron  
-rwxr-xr-x    1 root    root         193 2월 11 2003 tlpwatch  
[root@file root]#
```

만약 여러분들께서 매일 한번씩 실행시키고자 하는 내용이 있다면 /etc/cron.daily 디렉토  
리에 스크립트파일로 만들어서 넣어두시면 됩니다.

매주 1 회 자동실행하기위한 시스템 크론설정

```
22 4 * * 0 root run-parts /etc/cron.weekly
```

/etc/crontab 파일내에 존재하는 위의 내용은 매주 일요일 새벽 4 시 22 분마다  
/etc/cron.weekly 디렉토리내에 존재하는 파일들을 실행하게 됩니다. 즉, 매주 한번씩 실행  
될 내용이 저장된 디렉토리이며 이 디렉토리에서 실행되는 파일들은 다음과 같습니다.

```
[root@file root]# ls -l /etc/cron.weekly/  
합계 8  
-rwxr-xr-x    1 root    root         277 1월 25 2003 0anacron  
-rwxr-xr-x    1 root    root         414 2월 11 2003 makewhatiscron  
[root@file root]#
```

만약 여러분들께서 매주 한번씩 실행시키고자 하는 내용이 있다면 /etc/cron.weekly 디렉토  
리에 스크립트파일로 만들어서 넣어두시면 됩니다.

매월 1 회 자동실행하기위한 시스템 크론설정

```
42 4 1 * * root run-parts /etc/cron.monthly
```

/etc/crontab 파일내에 존재하는 위의 내용은 매월 1 일 새벽 4 시 42 분마다  
/etc/cron.monthly 디렉토리내에 존재하는 파일들을 실행하게 됩니다. 즉, 매월 한번씩 실행  
될 내용이 저장된 디렉토리이며 이 디렉토리에서 실행되는 파일들은 다음과 같습니다.

```
[root@file root]# ls -l /etc/cron.monthly/  
합계 4  
-rwxr-xr-x    1 root    root         278 1월 25 2003 0anacron  
[root@file root]#
```

만약 여러분들께서 매월 한번씩 실행시키고자 하는 내용이 있다면 /etc/cron.monthly 디렉  
토리에 스크립트파일로 만들어서 넣어두시면 됩니다.

이렇게 리눅스는 시스템에 꼭 필요한 주기적인 실행내용들을 /etc/crontab 파일에 설정해  
두고서 시스템 크론데몬인 crond 에 의해 실행시키고 있습니다. 이 설정내용을 잘 활용한다  
면 시스템관리에 매우 효율적으로 활용할 수 있습니다.



본 자료는 슈퍼유저코리아에서 출판한 “리눅스 실무명령어 바이블 (전 2 권)”의 본문중 일부  
분이며 책 구입희망자분들께서 공개요청하였던 부분중 요구횟수가 가장 많았던 부분만을 발  
췌하여 공개한 것입니다. 비록 일부분이지만 유용하게 사용하시기 바랍니다.

# 리눅스 실무명령어 바이블 1,11 편

즉, 지금까지의 크론(cron)에 대한 설명은 이미 시스템에 설정되어있는 크론설정  
(/etc/crontab 과 관련디렉토리들)을 이용하는 방법이었습니다. 이어지는 설명과 예들은  
crontab 명령어를 이용하여 직접 크론(cron)을 설정하는 방법에 대한 설명입니다.

## 실무명령어# 53-3 : 일반사용자의 crontab 명령어 사용허가 또는 제한하는 설정

/etc/ 디렉토리에는 cron.allow 파일과 cron.deny 파일이 존재할 수 있습니다. 리눅스 초기  
설정에는 이 두파일은 존재하지 않기 때문에 root 만이 crontab 명령어를 사용할 수 있습니  
다.

즉, /etc/ 디렉토리에 cron.allow 파일과 cron.deny 파일이 존재하지 않는다면 root 만이  
crontab 명령어를 이용하여 cron 설정을 할 수 있다는 의미가 됩니다.

만약 일반사용자에게도 crontab 명령어를 이용할 수 있도록 하기 위해서는 /etc/cron.allow  
파일에 사용자의 ID 를 등록해 주어야 합니다. 반대로 일반사용자의 crontab 명령어사용을  
제한하고자 한다면 /etc/cron.deny 파일에 사용자의 ID 를 등록해 주시면 됩니다.

리눅스 설치시에 이 두 파일은 기본생성되어 있는 파일이 아니므로 슈퍼유저(root)가 필요  
하다고 판단할 때에는 직접 만들어 주시면 됩니다.

이제 예를 위하여 필자가 설정한 두 파일의 예를 보도록 하겠습니다. 아래의 예를 보시면  
/etc/cron.allow 파일의 내용을 확인하고 있습니다.

```
[root@file root]# cat /etc/cron.allow
root
bible
locli
[root@file root]#
```

위의 /etc/cron.allow 파일에 등록되어 있는 root, bible, locli 계정 사용자는 crontab 명  
령어를 이용하여 개인적인 cron 설정을 할 수가 있습니다.

아래는 crontab 사용이 허용된 두 사용자 중 bible 이라는 사용자로 로그인하여 crontab 명  
령어를 사용하여 cron 설정을 하기위해 “crontab -e” 명령어를 사용하고 있는 예를 보인  
것입니다.

```
[bible@file bible]# id
uid=500(bible) gid=500(bible) groups=500(bible),5(tty),10(wheel)
[bible@file bible]#
[bible@file bible]# crontab -e

00 01 * * * su - root -c "/home/bible/backup.sh" >& /dev/null
~
~
~
~
~
```

즉, crontab 명령어를 사용가능토록 허용하고자 한다면 /etc/cron.allow 파일에 사용자의 ID

본 자료는 슈퍼유저코리아에서 출판한 “리눅스 실무명령어 바이블 (전 2 권)”의 본문중 일부  
분이며 책 구입희망자분들께서 공개요청하였던 부분중 요구횟수가 가장 많았던 부분만을 발  
췌하여 공개한 것입니다. 비록 일부분이지만 유용하게 사용하시기 바랍니다.

## 리눅스 실무명령어 바이블 1,11 편

를 위와 같이 등록하면 된다는 것을 설명하고 있는 것입니다. 참고로 /etc/cron.allow 파일  
이 존재한다면 root 라 하더라도 이 파일에 root 를 등록해야만 crontab 설정이 가능합니다.

위의 경우와는 반대로 crontab 명령어의 사용이 허용되어 있지 않는 sspark 이라는 사용자가  
crontab 명령어를 실행했을 경우에는 “You (sspark) are not allowed to use this program  
(crontab)” 이라는 메시지와 함께 crontab 명령어가 허용되지 않음을 알 수가 있습니다.

```
[sspark@file sspark]# id
uid=501(sspark) gid=501(sspark) groups=501(sspark),5(tty)
[sspark@file sspark]#
[sspark@file sspark]# crontab -e
You (sspark) are not allowed to use this program (crontab)
See crontab(1) for more information
[sspark@file sspark]#
```

앞서도 잠깐 언급하였지만 /etc/cron.allow 파일과 /etc/cron.deny 파일이 모두 존재하지 않  
는다면 오직 슈퍼유저(root)만이 crontab 명령어의 사용이 가능하다는 것을 기억해 두시기  
바랍니다.

### 실무명령어# 53-4 : “분, 시, 일, 월, 요일, 실행명령” 순으로 cron 설정하는 방법

crontab 명령어로 root 나 일반사용자들의 개별적인 cron 설정을 하는 구체적인 방법에 대해  
서 알아보도록 하겠습니다. cron 설정을 실행하는 명령어에 대해서는 다음 내용에 구체적인  
사례별로 설명되어 있으므로 참고하시고 여기서는 crontab 명령어로 설정되는 내용들의 설  
정방법에 대해서 알아보도록 하겠습니다.

먼저 root 의 cron 설정예를 보도록 하겠습니다.

뒤에서 설명하겠지만 root 의 cron 설정을 확인하려면 root 계정으로 “crontab -l” 이라고  
하시면 됩니다.

```
[root@file root]# id
uid=0(root) gid=0(root) groups=0(root),1(bin),2(daemon),3(sys),4(adm),6(disk),
[root@file root]#
[root@file root]# crontab -l
# DO NOT EDIT THIS FILE - edit the master and reinstall.
# (/tmp/crontab,10120 installed on Sun Dec 14 03:50:22 2003)
# (Cron version -- $Id: crontab.c,v 2.13 1994/01/17 03:20:37 vixie Exp #)
00 01 * * * su - root /usr/bin/rdate -s time.bora.net && /sbin/clock -w
00 03 * * * su - root -c "/root/backup.sh" >& /dev/null
00 08 * * * su - root -c "/root/system_check" > /root/syscheck/syscheck_logs
[root@file root]#
```

위의 root 계정의 cron 설정을 보면 각각의 행마다 숫자와 \*, 그리고 명령어들로 구성되어  
있는 것을 볼 수 있습니다. 위의 예에서 보신 바와 같이 root 를 포함한 각 계정들의 cron  
설정내용은 다음과 같이 모두 6 필드로 구성되어 있습니다.

설명을 위하여 위의 root 의 cron 설정행 중 다음행을 참고로 하여 그 의미를 설명하도록  
하겠습니다.



본 자료는 슈퍼유저코리아에서 출판한 “리눅스 실무명령어 바이블 (전 2 권)”의 본문중 일부  
 분이며 책 구입희망자분들께서 공개요청하셨던 부분중 요구횟수가 가장 많았던 부분만을 발  
 채하여 공개한 것입니다. 비록 일부분이지만 유용하게 사용하시기 바랍니다.

# 리눅스 실무명령어 바이블 1,11 편

```
00 03 * * * su - root -c '/root/backup.sh' >& /dev/null
```

맨 앞의 “00” 은 0 분을 의미합니다.  
 그 뒤의 “03” 은 03 시를 의미합니다.  
 그 뒤의 “\*” 는 매일을 의미합니다.  
 그 뒤의 “\*” 는 매월을 의미합니다.  
 그 뒤의 “\*” 는 매주를 의미합니다.

그 뒤의 “su - root -c '/root/backup.sh' >& /dev/null” 는 root 계정으로  
 “/root/backup.sh >& /dev/null” 을 실행하라는 의미입니다.

즉, 위의 행 내용을 종합적으로 해석해 보면 매월, 매주, 매일 03 시 00 분에 root 계정으로  
 “su - root -c '/root/backup.sh' >& /dev/null” 을 실행하라는 의미입니다. 참고로  
 /root/backup.sh 파일은 매일 실행되는 시스템의 백업스크립트가 저장된 파일로서 일백업  
 (daily backup)을 하기 위하여 필자가 만들어 놓은 스크립트파일입니다.

자, 그럼 설정내용의 예에 대한 구체적인 사례를 보았으므로 cron 으로 설정되는 각행들의  
 필드내용을 좀 더 구체적으로 설명해 보도록 하겠습니다.

첫번째 필드	“분(minutes)” 을 의미. 0,1,2,3,,,59 분까지 원하는 대로 설정가능.
두번째 필드	“시(hour)” 를 의미. 0,1,2,3,,,23 시까지 원하는 대로 설정가능.
세번째 필드	“일(day)” 을 의미. 1,2,3,4,,,31 일까지 원하는 대로 설정가능.
네번째 필드	“월(month)” 을 의미. 1,2,3,4,,,12 월까지 원하는 대로 설정가능.
다섯번째 필드	“요일(weekday)” 을 의미. 1,2,3,4,5,6,7 중 하나로 설정가능. (1:월요일,2:화요일,3:수요일,4:목요일,5:금요일,6:토요일,7:일요일을 각각 의미함. 0 도 일요일을 의미함.)
여섯번째 필드	실행될 명령어 내용

## 주의사항 1 : “\*” 문자

각 필드에는 “\*” 문자가 올 수 있으며, 이는 각 필드에 해당하는 모든 숫자를 의미 합니다.  
 예를 들어 “일(day)” 필드자리에 “\*” 로 설정되었다면 설정된 명령어가 매일 실행이 된다  
 는 의미입니다.

## 주의사항 2 : 하이픈(“-”)문자

각 필드자리에 하이픈(“-”)문자가 올 수 있습니다. 예를 들어 일(day)필드자리에 “11-  
 15” 로 설정되어 있다면 11 일, 12 일, 13 일, 14 일, 15 일을 의미합니다. 그리고 시간  
 (hour)필드자리에 “1-5” 로 설정되었다면 1 시, 2 시, 3 시, 4 시, 5 시를 의미합니다.

## 주의사항 3 : 콤마(,)문자

각 필드자리에는 콤마(,)문자가 올 수 있습니다. 예를 들어 일(day)필드자리에 “1,11,21”  
 로 설정되었다면 1 일, 11 일, 21 일을 각각 의미합니다. 그리고 시간(hour)필드자리에  
 “3,6,9” 로 설정되었다면 3 시, 6 시, 9 시를 각각 의미합니다.

본 자료는 슈퍼유저코리아에서 출판한 “리눅스 실무명령어 바이블 (전 2 권)”의 본문중 일부  
분이며 책 구입희망자분들께서 공개요청하셨던 부분중 요구횟수가 가장 많았던 부분만을 발  
췌하여 공개한 것입니다. 비록 일부분이지만 유용하게 사용하시기 바랍니다.

# 리눅스 실무명령어 바이블 1,11 편

## 실무명령어# 53-5 : crontab 으로 설정한 개별사용자의 cron 설정, 파일로 확인하기

root 를 비롯한 개별 사용자들의 cron 설정만을 모아둔 디렉토리가 존재합니다.  
/var/spool/cron 디렉토리에 root 와 개별사용자들의 cron 설정이 계정명과 동일한 파일명  
으로 보관되어 있습니다. 이 디렉토리에 존재하는 파일의 내용은 각 계정사용자들이 자기의  
cron 설정을 확인하기 위하여 사용하는 “crontab -l” 명령어의 결과와 동일한 것입니다.

다음은 현재 필자가 사용하고 있는 리눅스 서버의 /var/spool/cron 디렉토리내의 파일들입  
니다.

```
[root@file cron]# pwd
/var/spool/cron
[root@file cron]#
[root@file cron]# ls -l
합계 8
-rw-----  1 root    bible    253 12월 14 02:41 bible
-rw-----  1 root    root     396 12월 14 02:44 root
[root@file cron]#
```

위의 예를 보시면 /var/spool/cron 디렉토리에 bible, 그리고 root 라는 파일이 존재하고  
있습니다. 즉, bible 이라는 계정사용자의 cron 설정파일이 bible 이며 root 의 cron 설정파  
일이 root 라는 것을 알 수 있습니다.

이 두 파일의 내용을 살펴보도록 하겠습니다.

먼저 /var/spool/cron 디렉토리내의 bible 이라는 파일의 내용입니다.

이 파일의 내용은 bible 계정으로 로그인하여 “crontab -l” 의 결과와 동일하다는 것을 알  
아 두시기 바랍니다.

```
[root@file cron]# cat bible
# DO NOT EDIT THIS FILE - edit the master and reinstall.
# (/tmp/crontab.2600 installed on Sun Dec 14 02:41:06 2003)
# (Cron version -- $Id: crontab.c,v 2.13 1994/01/17 03:20:37 vixie Exp #)
00 01 * * * su - root -c "/home/bible/backup.sh" >& /dev/null
[root@file cron]#
```

다음은 /var/spool/cron 디렉토리내의 root 라는 파일의 내용입니다.

이 파일의 내용은 root 계정으로 로그인하여 “crontab -l” 의 결과와 동일하다는 것을 알  
아 두시기 바랍니다.

```
[root@file cron]# cat root
# DO NOT EDIT THIS FILE - edit the master and reinstall.
# (/tmp/crontab.8529 installed on Sun Dec 14 02:44:45 2003)
# (Cron version -- $Id: crontab.c,v 2.13 1994/01/17 03:20:37 vixie Exp #)
00 01 * * * su - root /usr/bin/rdate -s time.bora.net && /sbin/clock -w
00 03 * * * su - root -c "/root/backup.sh" >& /dev/null
00 08 * * * su - root -c "/root/system_check" > /root/syscheck/syscheck_logs
[root@file cron]#
```

위의 예들로 미루어 보면 “crontab -l” 이란 명령어의 결과는 /var/spool/cron/ 디렉토리  
에서 해당계정사용자의 파일을 가져와서 보여주는 것임을 알 수 있습니다.

본 자료는 슈퍼유저코리아에서 출판한 “리눅스 실무명령어 바이블 (전 2 권)”의 본문중 일부  
분이며 책 구입희망자분들께서 공개요청하였던 부분중 요구횟수가 가장 많았던 부분만을 발  
췌하여 공개한 것입니다. 비록 일부분이지만 유용하게 사용하시기 바랍니다.

# 리눅스 실무명령어 바이블 1,11 편

이렇듯 서버관리자는 특정명령어의 실행결과가 어떤 파일을 이용하여 출력되는 것인가를 알  
아들 필요가 있습니다. 훌륭한 서버관리자가 되기 위해서는 서버내부의 구석구석에 설정된  
모든 설정들을 손바닥에 올려놓은 듯 훤히 알고 있어야 합니다.

## 실무명령어# 53-6 : root 권한으로 root 의 cron 설정하는 방법

시스템관리자인 root 는 시스템의 모든 cron 설정을 수정, 변경, 삭제, 확인할 수 있습니다.  
먼저, 시스템관리자인 슈퍼유저(root) 자신의 cron 설정하는 방법에 대한 설명입니다.

아래의 예는 root 자신의 cron 설정을 하기 위하여 root 로 로그인 한 후에 “crontab -e”  
라는 명령어를 사용하여 cron 설정을 하는 내용입니다. “crontab -e” 을 실행하면  
/var/spool/cron/root 파일의 내용을 vi 로 불러와서 열어줍니다. 여기에서 vi 모드와 동일  
한 방법으로 cron 설정을 할 수 있습니다.

```
[root@file root]# id
uid=0(root) gid=0(root) groups=0(root),1(bin),2(daemon),3(sys),4(adm),6(disk),...
[root@file root]#
[root@file root]# crontab -e

00 01 * * * su - root /usr/bin/rdate -s time.bora.net && /sbin/clock -w
00 03 * * * su - root -c "/root/backup.sh" >& /dev/null
00 08 * * * su - root -c "/root/system_check" > /root/syscheck/syscheck_logs
~
~
~
```

root 의 cron 설정을 마무리하고 빠져 나가려면 vi 에서와 마찬가지로 “wq!” 라고 하면 됩  
니다. 이렇게 수정, 변경되어 저장되는 내용은 /var/spool/cron/root 파일에 저장됩니다.

여기서 설명하고자 하는 내용은 root 권한으로 root 자신의 크론(cron)을 설정, 변경하는  
“crontab -e” 라는 명령어의 사용법입니다.

## 실무명령어# 53-7 : root 권한으로 일반 사용자의 cron 설정하는 방법

시스템관리자인 root 는 root 자신의 cron 설정뿐만아니라 일반 계정사용자의 cron 설정까지도  
수정, 설정, 삭제를 할 수 있습니다.

시스템관리자인 root 권한으로 일반사용자의 cron 을 설정하고자 한다면 “crontab -u 계정  
명 -e” 라고 하시면 됩니다.

아래의 예는 root 권한으로 일반사용자인 bible 의 cron 을 설정, 변경하는 방법에 대한 예입  
니다.



본 자료는 슈퍼유저코리아에서 출판한 “리눅스 실무명령어 바이블 (전 2 권)”의 본문중 일부  
분이며 책 구입희망자분들께서 공개요청하셨던 부분중 요구횟수가 가장 많았던 부분만을 발  
췌하여 공개한 것입니다. 비록 일부분이지만 유용하게 사용하시기 바랍니다.

## 리눅스 실무명령어 바이블 1,11 편

```
[root@file root]# id
uid=0(root) gid=0(root) groups=0(root),1(bin),2(daemon),3(sys),4(adm),6(disk),
[root@file root]#
[root@file root]# crontab -u bible -e

00 01 * * * su - root -c "/home/bible/backup.sh" >& /dev/null
~
~
~
~
```

root 권한으로 일반계정사용자(bible)의 cron 설정을 마무리하고 빠져 나가려면 vi 에서와  
마찬가지로 “wq!” 라고 하면 됩니다. 이렇게 수정,변경되어 저장되는 내용은  
/var/spool/cron/bible 파일에 저장됩니다.

### 실무명령어# 53-8 : root 권한으로 root 의 cron 설정내용 확인하는 방법

이번에는 root 권한으로 root 자신의 cron 설정내용을 확인하는 방법에 대한 설명입니다.  
root 뿐아니라 일반계정사용자도 자기자신의 cron 설정을 확인하고자 한다면 “crontab -l”  
이라고 하시면 됩니다.

아래의 예는 root 로 로그인 후에 root 자신의 cron 설정을 확인하기 위하여 “crontab -l”  
이라는 명령어를 사용한 예입니다.

```
[root@file root]# id
uid=0(root) gid=0(root) groups=0(root),1(bin),2(daemon),3(sys),4(adm),6(disk),
[root@file root]#
[root@file root]# crontab -l
# DO NOT EDIT THIS FILE - edit the master and reinstall.
# (/tmp/crontab.8529 installed on Sun Dec 14 02:44:45 2003)
# (Cron version -- $Id: crontab.c,v 2.13 1994/01/17 03:20:37 vixie Exp $)
00 01 * * * su - root /usr/bin/rdate -s time.bora.net %& /sbin/clock -w
00 03 * * * su - root -c "/root/backup.sh" >& /dev/null
00 08 * * * su - root -c "/root/system_check" > /root/syscheck/syscheck_logs
[root@file root]#
```

위의 내용은 /var/spool/cron/root 파일내용과 동일한 내용으로서 root 계정으로 “crontab  
-l” 이라고 하면 /var/spool/cron/root 파일의 내용을 불러와 그대로 보여지게 됩니다.

### 실무명령어# 53-9 : root 권한으로 일반 사용자의 cron 설정내용 확인하는 방법

앞서 설명드렸듯이 시스템관리자인 root 는 자신의 cron 설정뿐 아니라 일반계정 사용자의  
cron 설정내용까지도 확인이 가능합니다.

root 권한으로 일반계정사용자의 cron 설정을 확인하고자 한다면 “crontab -u 계정명 -l”



본 자료는 슈퍼유저코리아에서 출판한 “리눅스 실무명령어 바이블 (전 2 권)”의 본문중 일부  
분이며 책 구입희망자분들께서 공개요청하셨던 부분중 요구횟수가 가장 많았던 부분만을 발  
췌하여 공개한 것입니다. 비록 일부분이지만 유용하게 사용하시기 바랍니다.

## 리눅스 실무명령어 바이블 1,11 편

이라고 하시면 됩니다.

아래의 예는 root 권한으로 일반계정사용자인 bible 사용자의 cron 설정내용을 확인하고 있  
는 것입니다.

```
[root@file root]# id
uid=0(root) gid=0(root) groups=0(root),1(bin),2(daemon),3(sys),4(adm),6(disk),
[root@file root]#
[root@file root]# crontab -u bible -l
# DO NOT EDIT THIS FILE - edit the master and reinstall.
# (/tmp/crontab.2600 installed on Sun Dec 14 02:41:06 2003)
# (Cron version -- $Id: crontab,c,v 2.13 1994/01/17 03:20:37 vixie Exp #)
00 01 * * * su - root -c "/home/bible/backup.sh" >& /dev/null
[root@file root]#
```

이렇게 보여지는 내용 또한 /var/spool/cron/bible 이라는 파일을 불러와서 보여준 결과입  
니다.

### 실무명령어# 53-10 : 일반사용자의 자기자신의 cron 설정하는 방법

일반계정사용자는 자기자신의 cron 만을 설정할 수 있습니다. 앞서 설명드렸듯이 일반 사용  
자가 cron 설정을 하려면 /etc/cron.allow 파일에 계정명으로 등록이 되어 있어야만 cron 설  
정이 가능합니다.

아래의 예는 bible 이라는 계정사용자가 자기자신의 cron 설정을 하기 위하여 bible 이라는  
계정으로 로그인 한 후에 “crontab -e” 라는 명령어를 사용한 것입니다.

```
[bible@file bible]# id
uid=500(bible) gid=500(bible) groups=500(bible),5(tty),10(wheel)
[bible@file bible]#
[bible@file bible]# crontab -e

00 01 * * * su - root -c "/home/bible/backup.sh" >& /dev/null
~
~
~
```

위의 내용은 /var/spool/cron/bible 이라는 파일을 불러와 vi 로 열어준 결과입니다. 새로  
설정하거나 변경된 내용으로 저장하려면 “:wq!” 로 저장한 후에 빠져나올 수 있습니다. 이  
렇게 설정 또는 변경된 내용은 /var/spool/cron/bible 이라는 파일에 저장이 됩니다.

### 실무명령어# 53-11 : 일반사용자의 cron 설정 확인하는 방법

일반계정사용자는 자기자신의 cron 설정만을 확인할 수 있습니다. 일반 계정사용자가 자기  
자신의 cron 설정을 확인하고자 한다면 로그인 한 후에 “crontab -l” 이라는 명령어를 사  
용하면 됩니다.



본 자료는 슈퍼유저코리아에서 출판한 “리눅스 실무명령어 바이블 (전 2 권)”의 본문중 일부 분이며 책 구입희망자분들께서 공개요청하였던 부분중 요구횟수가 가장 많았던 부분만을 발췌하여 공개한 것입니다. 비록 일부분이지만 유용하게 사용하시기 바랍니다.

## 리눅스 실무명령어 바이블 1,11 편

아래의 예는 bible 이라는 계정사용자가 bible 계정으로 로그인 한 후에 자기자신의 cron 설정을 확인하기 위하여 “crontab -l” 이라는 명령어를 사용한 결과입니다.

```
[bible@file bible]$ id
uid=500(bible) gid=500(bible) groups=500(bible),5(tty),10(wheel)
[bible@file bible]$
[bible@file bible]$ crontab -l
# DO NOT EDIT THIS FILE - edit the master and reinstall.
# (/tmp/crontab.2600 installed on Sun Dec 14 02:41:06 2003)
# (Cron version -- $Id: crontab.c,v 2.13 1994/01/17 03:20:37 vixie Exp $)
00 01 * * * su - root -c "/home/bible/backup.sh" >& /dev/null
[bible@file bible]$
```

이렇게 출력된 내용은 /var/spool/cron/bible 파일의 내용과 동일한 것입니다. 즉, /var/spool/cron/bible 파일의 내용을 불러와 그대로 보여준 것입니다.

### 실무명령어# 53-12 : root 권한으로 root 의 cron 설정 삭제하기

시스템관리자인 root 는 자기자신의 cron 설정뿐 아니라 일반계정사용자의 cron 설정까지도 삭제할 수 있습니다. 시스템관리자인 root 가 자기자신의 cron 설정을 삭제하고자 한다면 “crontab -r” 이라고 하면 됩니다.

아래는 root 가 자기자신의 cron 설정을 삭제하는 예입니다. 먼저 id 로 현재 사용자가 root 임을 확인하고 “crontab -l” 로 현재 설정된 root 의 cron 설정을 확인한 것입니다.

```
[root@file root]$ id
uid=0(root) gid=0(root) groups=0(root),1(bin),2(daemon),3(sys),4(adm),6(disk),
[root@file root]$
[root@file root]$ crontab -l
# DO NOT EDIT THIS FILE - edit the master and reinstall.
# (/tmp/crontab.10120 installed on Sun Dec 14 03:50:22 2003)
# (Cron version -- $Id: crontab.c,v 2.13 1994/01/17 03:20:37 vixie Exp $)
00 01 * * * su - root /usr/bin/rdate -s time.bora.net && /sbin/clock -w
00 03 * * * su - root -c "/root/backup.sh" >& /dev/null
00 08 * * * su - root -c "/root/system_check" > /root/syscheck/syscheck_logs
[root@file root]$
```

그리고 다음은 “crontab -r” 로 root 자신의 cron 설정을 삭제 하였습니다. 그리고 다시 “crontab -l” 로 설정내용을 확인해 보았으나 “crontab -r” 명령으로 이미 cron 설정이 삭제되었으므로 아무런 설정도 되어있지않다는 것을 확인한 것입니다.

```
[root@file root]$ crontab -r
[root@file root]$
[root@file root]$ crontab -l
no crontab for root
[root@file root]$
```

“crontab -r” 로 root 의 cron 설정이 삭제 되었다는 것은 /var/spool/cron/root 파일이 삭제되었다는 것을 의미합니다. 따라서 아래의 예는 “crontab -r” 의 결과



본 자료는 슈퍼유저코리아에서 출판한 “리눅스 실무명령어 바이블 (전 2 권)”의 본문중 일부 분이며 책 구입희망자분들께서 공개요청하였던 부분중 요구횟수가 가장 많았던 부분만을 발췌하여 공개한 것입니다. 비록 일부분이지만 유용하게 사용하시기 바랍니다.

## 리눅스 실무명령어 바이블 1,11 편

/var/spool/cron/root 파일이 삭제되었음을 확인한 것입니다.

```
[root@file root]#  
[root@file root]# ls -l /var/spool/cron/root  
ls: /var/spool/cron/root: 그런 파일이나 디렉토리가 없음  
[root@file root]#
```

### 실무명령어# 53-13 : 일반계정사용자의 자기 자신의 cron 설정 삭제하기

일반계정 사용자는 자기 자신의 cron 설정만을 삭제할 수 있습니다. 일반 계정사용자가 자기 자신의 cron 설정을 삭제하려면 자기 자신의 계정으로 로그인 한 후에 “crontab -r” 이라고 하면 됩니다.

아래의 예는 bible 이라는 계정사용자가 bible 계정으로 로그인 한 후에 자기 자신의 cron 설정을 삭제한 예입니다.

먼저, id로 현재 계정사용자가 bible 임을 확인한 후에 “crontab -l” 로 현재 설정되어 있는 bible 사용자의 cron 설정을 확인 하였습니다.

```
[bible@file bible]# id  
uid=500(bible) gid=500(bible) groups=500(bible),5(tty),10(wheel)  
[bible@file bible]#  
[bible@file bible]# crontab -l  
# DO NOT EDIT THIS FILE - edit the master and reinstall,  
# (/tmp/crontab.2600 installed on Sun Dec 14 02:41:06 2003)  
# (Cron version -- #Id: crontab.c,v 2.13 1994/01/17 03:20:37 vixie Exp #)  
00 01 * * * su - root -c "/home/bible/backup.sh" >& /dev/null  
[bible@file bible]#
```

그리고 다음은 bible 사용자가 자기 자신의 cron 설정을 삭제하기 위하여 “crontab -r” 명령어를 사용한 것입니다. 그런 다음 “crontab -l” 로 설정내용을 확인하였으나 이미 삭제되었으므로 아무런 내용도 설정되어 있지않다는 “no crontab for bible” 이라는 메시지를 보여주고 있는 것입니다.

```
[bible@file bible]# crontab -r  
[bible@file bible]#  
[bible@file bible]# crontab -l  
no crontab for bible  
[bible@file bible]#
```

이렇게 bible 계정사용자가 자기 자신의 cron 설정을 삭제하게 되면 /var/spool/cron/bible 파일이 삭제됩니다.

### 실무명령어# 53-14 : root 권한으로 일반계정사용자의 cron 설정 삭제하기

이번에는 root 권한으로 일반계정사용자의 cron 설정을 삭제하는 방법을 설명 드리겠습니다.



본 자료는 슈퍼유저코리아에서 출판한 “리눅스 실무명령어 바이블 (전 2 권)”의 본문중 일부  
분이며 책 구입희망자분들께서 공개요청하셨던 부분중 요구횟수가 가장 많았던 부분만을 발  
췌하여 공개한 것입니다. 비록 일부분이지만 유용하게 사용하시기 바랍니다.

## 리눅스 실무명령어 바이블 1,11 편

시스템관리자 root 의 권한으로 일반계정사용자의 cron 을 삭제하려면 “crontab -u 계정명 -r” 이라고 하시면 됩니다.

아래의 예는 root 사용자가 bible 사용자의 cron 설정을 삭제하는 예입니다.

먼저, id 로 root 임을 확인한 후에 앞서 설명드린 방법과 마찬가지로 bible 의 cron 설정을  
확인하기 위하여 “crontab -u bible -l” 이라는 명령어를 실행한 것입니다. 결과 bible 의  
cron 설정을 확인 하였습니다.

```
[root@file root]# id
uid=0(root) gid=0(root) groups=0(root),1(bin),2(daemon),3(sys),4(adm),6(disk),
[root@file root]#
[root@file root]# crontab -u bible -l
# DO NOT EDIT THIS FILE - edit the master and reinstall.
# (/tmp/crontab,10432 installed on Sun Dec 14 04:11:15 2003)
# (Cron version -- $Id: crontab,c,v 2.13 1994/01/17 03:20:37 vixie Exp #)
00 01 * * * su - root -c "/home/bible/backup.sh" >& /dev/null
[root@file root]#
```

그런 다음 root 계정으로 bible 의 cron 을 삭제하기 위하여 “crontab -u bible -r” 이라는  
명령어를 실행 하였습니다. 그리고 삭제를 확인하기 위하여 좀 전에 실행했던 “crontab -  
u bible -l” 을 다시한번 실행한 것입니다. 실행결과 bible 의 cron 설정이 삭제되었으므로  
“no crontab for bible” 이라는 메시지만 출력된 것입니다.

```
[root@file root]# crontab -u bible -r
[root@file root]#
[root@file root]# crontab -u bible -l
no crontab for bible
[root@file root]#
```

그리고 아래의 예는 bible 의 cron 설정을 삭제하였다는 의미는 /var/spool/cron/bible 파일  
이 삭제되었다는 의미이므로 이를 확인한 것입니다.

```
[root@file root]# ls -l /var/spool/cron/bible
ls: /var/spool/cron/bible: 그런 파일이나 디렉토리가 없음
[root@file root]#
```

이상과 같이 시스템의 cron 실행체계와 crontab 명령어로 설정되는 root 와 각 계정들의  
cron 설정방법, 설정내용 확인,삭제등에 대해서 전반적으로 알아보았습니다.

만약 여러분들께서 시스템관리자라고 한다면 crontab 명령어편에서 설명드린 내용을 참조하  
여 직접 관리하시는 서버에 적용해 보시기 바랍니다.

crontab 명령어의 서두에서도 말씀드린 바와 같이 쉘프로그램작성법과 함께 crontab 의 실무  
설정법을 완벽히 숙지하신다면 여러분들은 이미 고급 서버관리자가 되신 것입니다.

필자의 이런 말씀이 남의 얘기가 아닌 여러분들 자신의 얘기가 될 수 있기를 바라면서  
crontab 에 대한 필자의 의무를 다하였으므로 이만 물러갑니다.

- 본 자료의 저작권은 “슈퍼유저코리아” ([www.superuser.co.kr](http://www.superuser.co.kr))에 있습니다 .
- 본자료는 “슈퍼유저코리아”에서 출판한 『리눅스 실무명령어 바이블(1편,2편)』의 실제  
내용중 일부분으로서 책의 본문내용을 확인하고자하시는 분들의 요청에 의해 요청횟수가



본 자료는 슈퍼유저코리아에서 출판한 “리눅스 실무명령어 바이블 (전 2 권)”의 본문중 일부  
분이며 책 구입희망자분들께서 공개요청하였던 부분중 요구횟수가 가장 많았던 부분만을 발  
췌하여 공개한 것입니다. 비록 일부분이지만 유용하게 사용하시기 바랍니다.

## 리눅스 실무명령어 바이블 1, 2 편

많았던 부분만을 일부 공개한 것입니다.

○ 본 책자의 자세한 모든 정보는 <http://www.superuser.co.kr/linuxcommandbible/> 에서  
확인하시기 바랍니다.

○ 본 자료의 수정 및 편집은 허용되지않습니다.

○ 본 책은 권수 : 2 권 (1 편, 2 편), 페이지수 : 총 1,859 페이지, 장(chapter): 382 장,  
절: 1,900 개의 절로 구성되어있습니다.

○ 저작권 및 문의전화(사무실) : TEL : 051-583-8128, FAX : 051-583-8130

