SeoulTech

제11장 리눅스 시스템 조사

박 종 혁 교수

UCS Lab

Tel: 02-970-6702

Email: jhpark1@seoultech.ac.kr

개요

- **학습목표**
  - 오픈 소스 운영체제인 리눅스의 특성과 각종 환경 설정, 로그 파일 분석에 대해 알아본다.
  - 이 장에서는 리눅스의 여러 배포판 중 우분투를 기준으로 설명한다.

- **학습 내용**
  - 리눅스 시스템의 특징
  - 리눅스 환경 설정 파일
  - 자동 실행 프로그램
  - 시스템 로그 파일
  - /proc 파일 시스템
  - 활성 시스템 조사

목 차

리눅스 시스템의 특성

- **명령어 기반**
  - 모든 입·출력은 문자를 기반으로 수행
  - 운영체제의 핵심인 커널이 명령어 기반의 입·출력을 하도록 작성
  - 리눅스 GUI 환경 : X 윈도우(GNOME, KDE)
    - 그래픽 등에 소모되는 시스템의 자원을 절약하기 위해 X 윈도우를 쓰지 않는 경우가 많다.
- **디렉토리**
  - 드라이브 문자가 없음, 필요한 파일 시스템을 마운트하여 사용
  - 디렉토리 구분자로는 "/"를 사용
  - 디렉토리 이름의 대·소문자를 구분, 최대 255글자까지 사용
  - 최상위 디렉토리로 루트 디렉토리가 존재하며 "/"로 표시
  - "." : 현재 디렉토리, ".." : 상위 디렉토리
  - "."으로 시작하는 디렉토리명 : 숨겨진 디렉토리
    - 별도의 옵션을 사용하지 않는 경우 사용자에게 보이지 않음

리눅스 시스템의 특성

/ : 루트 디렉토리라
/bin : ls, cp와 같은 기본 명령어가 저장
/boot : 시스템 부팅시 필요한 파일 저장
/dev : 시스템의 장치 파일 저장
/etc : 시스템 설정 파일 저장
/home : 사용자의 기본 디렉토리 저장
/lib : 시스템의 공유 라이브러리 저장

```
n0fate@ubuntu:/$ ls
bin   dev  home  lost+found  mnt  proc  sbin     srv  usr
boot  etc  lib   media       opt  root  selinux  sys  var
```

/lost+found : fsck 명령어가 사용하는 디렉토리, 특정 파일의 위치를 결정할 수 없는 경우 이 디렉토리에 보관

- fsck : 파일 시스템의 이상 여부를 진단하고 복구

/mnt : 다른 장치를 마운트하기 위한 디렉토리
/proc : 가상 파일시스템, 실제로 디스크에 저장되지 않음, 프로세스에 대한 정보를 저장
/usr : 사용자가 설치한 응용 프로그램, 소스코드, 도움말 파일 등이 저장
/root : 슈퍼 유저인 root의 홈 디렉토리
/sbin : 슈퍼 유저가 사용하는 명령어가 저장
/var : 로그 파일, 프린터 스풀 파일, 시스템 동작 중에 변화하는 파일 저장

리눅스 시스템의 특성

- **파일 특성**
  - 실행 파일의 확장자 제한 없음
  - 리눅스 시스템은 파일의 권한을 통해 실행 여부를 결정
  - 대・소문자를 구분, 최대 255글자까지 사용
  - "."으로 시작하는 파일 : 숨김 속성
  - 특수 파일 : 심볼릭 링크 파일, 하드 링크 파일
  - 심볼릭 링크(Symbolic Link)
    - 윈도우 시스템의 바로가기 파일과 유사

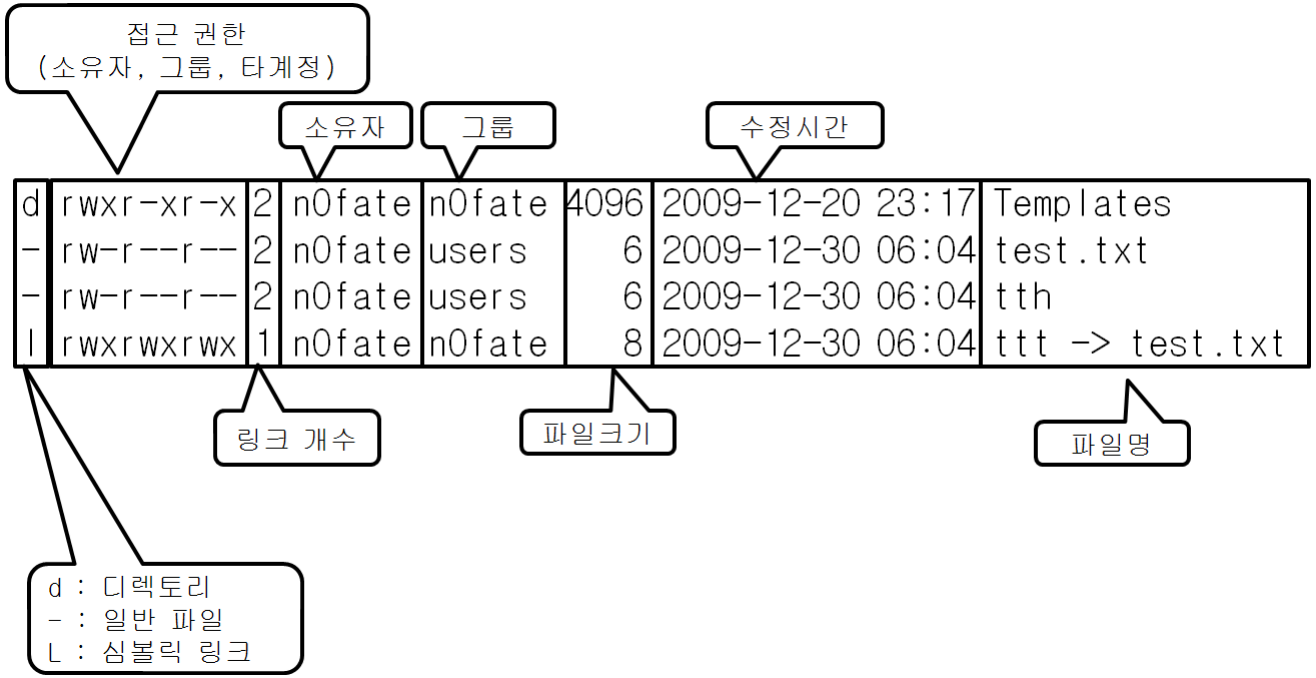
```
n0fate@ubuntu:~$ ls -l
lrwxrwxrwx 1 n0fate n0fate 8 2009-12-30 06:04 forensic -> /root/forensic.txt
```

    - "forensic" 파일은 심볼릭 링크로써 "/root/forensic.txt"를 가리키고 있다.
    - 심볼릭 링크는 원본 파일이 삭제되면 함께 지워진다.

리눅스 시스템의 특성

- 하드 링크(Hard Link)
  - 하드 링크는 원본을 복사하여 사본을 생성
  - 하드 링크 파일 수정시 원본과 사본 모두 영향을 받음
  - 원본 파일 삭제시 하드 링크 파일 삭제되지 않음
  - 여러 시스템에서 파일을 공유하면서 안전하게 보관하고자 할 때 이용

리눅스 시스템의 특성



- 첫 바이트
  - "d" : 디렉토리, "-" : 일반 파일, "l" : 심볼릭 링크
- 두 번째 바이트
  - 소유자, 그룹, 타계정에 대한 권한
  - "r" : 읽기, "w" : 쓰기, "x" : 실행
- Ex)"test.txt" 파일
  - 소유자 : 읽기 및 쓰기 권한(rw-) ,그룹과 타계정 : 읽기 권한(r--)

리눅스 시스템의 특성

- **SUID와 SGID**
  - 특정 파일을 실행시 해당 파일의 소유자 또는 그룹의 권한으로 실행

```
-rwsr-xr-x 1 root root 34696 2009-05-11 14:04 ping
```

    - 파일의 소유자는 "root", 소유자의 권한은 "rws"
    - "s"가 SUID가 설정되어 있음

- **데몬(Daemon)**
  - 백그라운드로 동작하는 서버 프로그램, 윈도우의 서비스와 유사
  - 일반적으로 데몬은 시스템이 시작할 때 동작

리눅스 환경 설정 파일

- **사용자 및 그룹 설정**
  - 시스템에서 가장 중요한 정보로써 유출되면 공격자가 시스템에 접속할 수 있는 시발점이 된다.
  - 침해 사고를 조사할 때에는 불필요한 계정의 유・무, 잘못된 그룹 설정, 빈 패스워드를 사용하는 계정 등이 있는지 확인해야 한다.
  - /etc/passwd
    - 시스템의 모든 계정 정의
    - 계정명, 그룹, 홈 디렉토리 등의 정보를 저장
    - 콜론(:)을 통해 각 항목을 구분

리눅스 환경 설정 파일

- username:password:uid:gid:gecos:homedir:program
- username : 계정명, 로그인시 사용되는 이름
- password : 패스워드가 암호화되어 저장
  - 사용자가 패스워드를 변경시"passwd" 명령 사용
  - 쉐도우(shadow) 패스워드 : "x"나 "*"로 표시
- uid : 사용자 ID로 각 사용자에게 시스템이 부여하는 고유한 정수값
  - 시스템에서 uid와 username는 1:1 대응
- gid : 사용자가 속해 있는 기본 그룹의 ID
  - 그룹의 ID에 따른 이름은 /etc/group에 정의
- gecos : 사용자의 실제 이름이나 사무실, 연락처 등의 부가 정보
  - finger 명령이 사용자를 구별하기 위해 이 항목을 이용
- homedir : 각 사용자의 홈 디렉토리
  - 홈 디렉토리 : 사용자가 로그인시 최초로 접근하는 디렉토리
- program : 사용자가 로그인한 후에 실행되는 프로그램
  - 사용자가 지정한 쉘을 실행

리눅스 환경 설정 파일

- **/etc/shadow**
  - 암호화된 패스워드를 별도로 보관하는 파일
  - ①:②:③:④:⑤:⑥:⑦:⑧:⑨
    - ① : 계정명, /etc/passwd 파일의 계정명과 동일
    - ② : 암호화된 패스워드, 공란일 경우 로그인시 패스워드 불필요, "*" 일 경우 사용하지 않는 계정
    - ③ : 패스워드를 마지막으로 바꾼 날, UTC 기준일부터 며칠 째인지
    - ④ : 며칠이 지난 후 패스워드를 변경 가능한지, 0이면 언제든 패스워드의 변경이 가능
    - ⑤ : 패스워드를 바꿔야 하는 날부터 며칠이 지났는지
    - ⑥ : 패스워드 만료기간이 되었음을 얼마 동안 사용자에게 알릴지
    - ⑦ : 패스워드가 만료되어 사용할 수 없게 된 후 며칠이 지났는지
    - ⑧ : 계정 사용을 할 수 없게 된 것이 UTC 기준일부터 며칠 째인지
    - ⑨ : 사용하지 않는다.

리눅스 환경 설정 파일

- **/etc/group**
  - 그룹과 그룹에 속해있는 사용자에 대해 저장
  - 모든 사용자는 최소한 하나의 그룹에 속해 있음
  - groupname:password:gid:member1,member2, member3 …
    - groupname : 그룹의 이름
    - password : 그룹에 지정된 패스워드,
      - 패스워드가 지정된 경우, 그룹의 멤버가 아니어도, 패스워드를 알고 있는 사용자는 그룹의 권한을 획득 가능
    - gid : 그룹의 ID로 groupname과 대응
    - member1, member2 … : 그룹에 속해있는 사용자 목록

리눅스 환경 설정 파일

- **파일 시스템 설정**
  - 설정파일 : /etc/fstab, /etc/mtab에 저장
  - 파일시스템이 로컬 시스템에 한정되지 않음
  - NFS(Network File System)과 같이 원격지 시스템에 대한 마운트 정보 역시 파일 시스템 설정에 기록 가능
  - 로컬 파일 시스템에 대한 분석을 수행하기 전에 분석 대상 시스템에 설정되어 있는 파일 시스템 설정을 확인
  - /etc/fstab
    - 시스템을 부팅하면서 각 장치를 자동으로 마운트할 때 사용
    - "mount" 명령에서 "-a" 옵션을 사용할 경우 사용

```
FileSystem MountPoint Type Option1,Option2… Dump Pass
```

리눅스 환경 설정 파일

- FileSystem : 마운트되는 장치의 이름
  - NFS를 마운트하는 경우에는 원격지의 경로
- MountPoint : 마운트 지점
- Type : 파일시스템의 타입
- Option
  - ro/rw : 읽기 전용/읽기 쓰기 가능
  - suid/nosuid : SUID 사용 가능/불가
  - sgid/nosgid : SGID 사용 가능/불가
  - exec/noexec : 실행 권한을 가진 파일의 실행 가능/불가
  - quota/noquota : 사용자별 사용 용량 제한/제한하지 않음
  - auto/noauto : 부팅할 때 자동 마운트 수행/수행하지 않음
  - user/nouser : 일반 사용자도 마운트 가능/불가
- Dump : "dump"라는 도구를 이용하여 파일 시스템 백업 가능 여부
- Pass : 마운트시 파일 시스템 검사 여부
  - 0 : 검사하지 않음, 1,2 : 파일 시스템 검사. Pass가 1 부터 검사

리눅스 환경 설정 파일

- **/etc/fstab의 예제 파일**

```
proc /proc proc defaults 0 0
UUID=42682e56-77f3-49b6-a2b3-94413bd74a0f / ext4 errors=remount-ro 0 1
UUID=6e6c1a88-8c7b-44ea-80e0-0560af0d3cb9 none swap sw 0 0
/dev/scd1 /media/cdrom0 udf,iso9660 user,noauto,exec,utf8 0 0
/dev/scd0 /media/cdrom1 udf,iso9660 user,noauto,exec,utf8 0 0
/dev/fd0 /media/floppy0 auto rw,user,noauto,exec,utf8 0 0
```

리눅스 환경 설정 파일

- **/etc/mtab**
  - 현재 마운트되어 있는 정보 저장
  - “mount" 명령어로 확인
  - "mount", "umount" 명령에 의해 기록 및 삭제
  - "mount" 명령어에 ”-n" 옵션 : /etc/mtab에 내용을 기록하지 않음

```
/dev/sda1 / ext4 rw,errors=remount-ro 0 0
proc /proc proc rw 0 0
none /sys sysfs rw,noexec,nosuid,nodev 0 0
```

리눅스 환경 설정 파일

- **네임 서버 설정**
  - 도메인 주소 또는 "WebServer"와 같이 특정 이름을 사용하는 경우 IP 주소로 바꾸는 역할
    - 로컬의 네임 서비스 정보를 이용하는 호스트 설정
    - 원격의 DNS 서버를 통해 IP 주소를 알아내는 DNS 서비스
  - 호스트 설정
    - 호스트 설정 파일은 로컬에서 운용되는 네임 서비스
    - DNS 서버를 거치지 않고 지정된 IP 주소를 사용
    - IP 스푸핑(spoofing) 방지, 특정 호스트에 대한 접근 제어
    - 설정 파일 : /etc/hosts.conf, /etc/hosts, /etc/hosts.allow, /etc/hosts.deny
    - 잘못된 호스트 파일의 설정은 시스템 사용자가 인식하지 못하는 상태에서 공격자의 사이트로 유도될 수 있다.

리눅스 환경 설정 파일

- /etc/hosts.conf
  - 네임 서비스를 어떻게 동작시킬 것인지를 결정
  - order : 네임 서비스가 동작하는 순서를 결정, hosts는 로컬에 저장된 /etc/hosts 파일 참조를, bind는 DNS 서버를 통한 네임 서비스를, nis는 네트워크 정보 서비스(NIS) 프로토콜
  - alert : on/off로 설정, IP 스푸핑이 발생하면 syslog에 기록
  - multi : on/off로 설정, 동일한 호스트에 여러 IP 주소를 부여
  - nospoof : on/off로 설정
    - 주소를 이용하여 IP를 얻어내는 것과 IP를 통해 주소를 얻는 두가지 방법을 모두 수행하여 두 개가 일치할 경우에만 IP 주소를 전달한다.
  - trim : 도메인 주소를 파라메터로 설정, hosts 파일을 참조할 때, 도메인을 입력하지 않아도 자동으로 찾을 수 있게 한다.

리눅스 환경 설정 파일

- /etc/hosts
  - 호스트의 이름과 IP 주소의 쌍으로 구성

```
127.0.0.1 localhost

127.0.1.1 ubuntu

# The following lines are desirable for IPv6 capable hosts

::1 localhost ip6-localhost ip6-loopback

fe00::0 ip6-localnet

ff00::0 ip6-mcastprefix

ff02::1 ip6-allnodes

ff02::2 ip6-allrouters
```

리눅스 환경 설정 파일

- /etc/hosts.allow, /etc/hosts.deny
  - 외부의 접근을 허용 또는 차단하기 위해 사용

```
ALL: 192.168.0.

ALL: 192.168.1.

ALL: 192.168.3. EXCEPT 192.168.3.100

ALL: .korea.ac.kr EXCEPT malware.korea.ac.kr
```

- DNS 설정
  - DNS 설정 파일은 /etc/resolv.conf

```
nameserver 192.168.80.2

nameserver 192.168.80.3
```

리눅스 환경 설정 파일

- 네트워크 인터페이스 카드(NIC) 설정
  - /etc/network/interfaces에 저장
  - auto는 부팅시 자동으로 인터페이스를 활성화
  - 고정 IP 주소 : "static"로 선언, IP 주소, 넷마스크, 네크워크 주소, 브로드캐스트 주소, 게이트 웨이 등의 항목을 설정
  - 동적 IP 주소 : "static" 대신에 "dhcp"로 지정

```
auto lo
iface lo inet loopback
auto eth0
iface eth0 inet static
address 192.168.80.100
netmask 255.255.255.0
network 192.168.80.0
broadcast 192.168.80.255
gateway 192.168.80.1
```

자동실행 프로그램

- **자동 실행 프로그램**
  - 다양한 방법으로 프로그램이 자동으로 실행 가능
  - 실행 순서는 일반적으로 스크립트에 의존
  - 용도와 스크립트의 위치도 다름
    - 시스템이 부팅하거나 종료할 때의 자동 실행,
    - 로그인 등으로 인해 쉘이 시작할 때의 자동 실행
    - 데몬에 의해 주기적인 자동 실행 등
- **시스템의 시작 • 종료시의 자동 실행 파일**
  - 실행레벨(RunLevel)에 의존
  - 실행레벨은 0~6
  - 리눅스 배포판에 따라 실행레벨의 정의 다름

자동실행 프로그램

- **"페도라 코어 8"에 정의되어 있는 실행레벨**
  - 0 - 시스템 중지(halt), 임의로 지정하면 안됨
  - 1 - 단일 사용자 모드
  - 2 - NFS를 제외한 다중 사용자 모드(네트워크와 연결되어 있지 않으면 3과 동일)
  - 3 - 모든 기능이 동작하는 다중 사용자 모드(default)
  - 4 - 사용안함
  - 5 - X 윈도우 이용
  - 6 - 재부팅, 임의로 지정하면 안됨

자동실행 프로그램

- **우분투 데스크톱 버전에 정의되어 있는 실행레벨**
  - 0 - 시스템 중지(halt), 임의로 지정하면 안됨
  - 1 - 단일 사용자 모드
  - 2 - 모든 기능이 동작하는 다중 사용자 모드(default)
  - 3 - 모든 기능이 동작하는 다중 사용자 모드(default)
  - 4 - 모든 기능이 동작하는 다중 사용자 모드(default)
  - 5 - 모든 기능이 동작하는 다중 사용자 모드(default)
  - 6 - 재부팅, 임의로 지정하면 안됨
  - /etc 디렉토리
    - 실행레벨에 따라 실행할 파일을 정의한 파일
    - 실행파일(또는 실행파일에 대한 심볼릭 링크)
    - 각 실행파일의 환경설정 파일이 정의
    - 페도라 스크립트 파일 저장위치 : /etc/rc.d/rc0~/etc/rc.d/rc6
    - 우분투 스크립트 파일 저장위치 : /etc/rc0.d~/etc/rc6.d

자동실행 프로그램

- 스크립트 파일의 이름과 실제 링크의 위치
  - S16ssh -> ../init.d/ssh
  - S20kerneloops -> ../init.d/kerneloops
  - S20vsftpd -> ../init.d/vsftpd
  - S25bluetooth -> ../init.d/bluetooth
  - K20kerneloops -> ../init.d/kerneloops
  - K20vsftpd -> ../init.d/vsftpd
  - K50alsa-utils -> ../init.d/alsa-utils
  - K74bluetooth -> ../init.d/bluetooth
- 자동 실행 스크립트의 이름
  - SnnName, KnnName(nn은 00~99의 숫자)로 지정
  - S로 시작하는 스크립트는 시스템이 시작할 때 자동으로 실행
  - K로 시작하는 스크립트는 시스템이 종료할 때 자동으로 실행
  - nn은 디렉토리 내 스크립트의 실행 순서, 숫자가 작은 것부터 먼저 실행

자동실행 프로그램

- **".bashrc", ".bash_profile", ".bash_logout"**
  - bash 쉘은 동작할 때 자동으로 실행되는 스크립트
  - ".bashrc" bash 쉘이 실행할 때 마다 실행
  - ".bash_profile" bash 쉘이 로그인 쉘로 이용되었을 때만 동작 ".bash_logout" bash 로그인 쉘이 종료
  - /etc/profile, /etc/bash.bashrc 등
  - /etc 디렉토리의 파일 : 사용자의 홈 디렉토리에 있는 스크립트가 동작할 때 해당 파일의 유・무를 검사하여 있으면 실행

- **".cshrc", ".login"**
  - c 쉘이나 tcsh가 실행할 때 자동으로 실행되는 스크립트 파일
  - "./login"은 bash 쉘의 ".bash_profile"과 같이 로그인 할 때만 동작

자동실행 프로그램

- **“.kshrc"**
  - 콘(Korn) 쉘이 실행할 때 자동으로 실행되는 스크립트 파일
- **".vimrc"**
  - 텍스트 편집기인 vi, vim 에디터를 이용할 경우에 자동으로 실행
- **".xinitrc"**
  - X 윈도우가 구동될 때 자동으로 실행된다.

자동실행 프로그램

- **작업 스케줄링(crontab)**
  - crontab은 리눅스 시스템에서 사용하는 작업 스케줄링 데몬
  - 작업 스케줄링 정보는 /etc/crontab 파일 내에 정의

```
Min Hour Day Month DayOfWeek Command
```

  - Min : 0~59분
  - Hour : 0~23시
  - Day : 1~31일
  - Month : 1~12 혹은 Jan~Dec
  - DayOfWeek : 0 또는 7은 일요일, 1=월요일, 2=화요일… 혹은 Sun~Sat
  - Command : 실행하고자 하는 명령어 또는 스크립트

시스템 로그 파일

- **syslog 데몬 : 시스템의 로그를 기록**
  - 운영체제의 동작과정에서 발생하는 로그를 기록
  - 다른 데몬 프로그램이 발생시키는 로그 역시 기록
  - /var/log 디렉토리에 기록
- **bash 쉘에 의해 기록되는 명령어 히스토리가 존재**
- **utm p 파일, wtmp 파일 : 사용자 접속 기록**
- **시스템에서 발생된 사건에 대해 많은 정보를 제공**

시스템 로그 파일

- 사용자가 입력한 명령어의 히스토리는 각 사용자의 홈 디렉토리에 저장
- bash 쉘 : “.bash_history" 파일에 입력한 명령어를 기록
- $HISTSIZE : 명령어 히스토리 파일에 기록되는 명령어의 수 지정

```
n0fate@ubuntu:~$ echo $HISTSIZE
100
```

시스템 로그 파일

- **/var/log/wtmp와 /var/run/utmp, /var/log/lastlog**
  - /var/log/wtmp 파일
    - 로그인 시간과 사용자가 시스템에 연결한 기간과 재부팅할 때의 부팅 기록을 바이너리 파일로 저장
    - last 명령이 /var/log/wtmp 파일을 사용
      - last : 사용자의 로그인 목록을 생성
    - 파일을 보기 위해서 별도의 뷰어 또는 파서가 필요

– / var/log/wtmp 파일

```
n0fate@ubuntu:/var/log$ last
n0fate pts/0 :0 Wed Dec 30 03:25 still logged in
n0fate :0 Wed Dec 30 03:25 still logged in
reboot system boot 2.6.31-14-generi Wed Dec 30 03:23 - 13:32 (10:08)
n0fate pts/4 192.168.109.1 Mon Dec 21 18:15 - 18:19 (00:04)
n0fate pts/3 192.168.109.1 Sun Dec 20 23:44 - down (6+23:33)
n0fate pts/0 :0 Sun Dec 20 23:18 - down (6+23:59)
n0fate :0 Sun Dec 20 23:17 - 23:17 (7+00:00)
reboot system boot 2.6.31-14-generi Sun Dec 20 23:15 - 23:17 (7+00:01)
reboot system boot 2.6.31-14-generi Sun Dec 20 23:14 - 23:15 (00:00)
reboot system boot 2.6.31-14-generi Sun Dec 20 23:07 - 23:12 (00:04)
reboot system boot 2.6.31-14-generi Wed Dec 2 19:20 - 19:42 (00:22)
wtmp begins Wed Dec 2 19:20:18 2009
```

시스템 로그 파일

- /var/run/utmp 파일
  - 시스템에 현재 로그인되어 있는 사용자의 정보를 저장
  - who, w, finger 등의 명령어 사용

```
n0fate@ubuntu:/var/log$ w
13:39:35 up 10:15, 3 users, load average: 0.11, 0.06, 0.02
USER TTY FROM LOGIN@ IDLE JCPU PCPU WHAT
n0fate :0 - 03:25 ?xdm? 1:44 0.03s /bin/sh /usr/bin/x-session-manager
n0fate pts/0 :0 03:25 10:13m 0.00s 0.02s /usr/bin/kwrited
```

- /var/log/lastlog 파일
  - 각 계정이 마지막으로 로그인한 시간을 저장
  - lastlog 명령어를 사용

```
n0fate@ubuntu:/var/log$ lastlog
n0fate :0 Wed Dec 30 03:25:29 -0800 2009
gdfriend pts/1 Wed Dec 30 13:38:42 -0800 2009
```

시스템 로그 파일

- /var/log/dmesg
  - 시스템이 부팅할 때 출력하는 메시지를 기록하는 로그 파일
  - 시스템의 하드웨어 설정 및 로드된 디바이스 드라이버 등을 확인

```
[4.020939] udev: starting version 147
[4.022419] EXT4-fs (sda1): internal journal on sda1:8
[4.364226] VMCI: Major device number is: 251
[4.364295] Probing for vmci/PCI.
[4.364363] alloc irq_desc for 16 on node -1
[4.364406] alloc kstat_irqs on node -1
[4.364475] vmci 0000:00:07.7: PCI INT A -> GSI 16 (level, low) -> IRQ 16
[4.364544] Found vmci/PCI at 0x1080, irq 16.
[4.364613] VMCIUtil: Host capability check: PASSED
[4.364682] Registered vmci device.
[4.416411] shpchp: Standard Hot Plug PCI Controller Driver version: 0.4
```

시스템 로그 파일

- /var/log/messages
  - 시스템에서 발생하는 각종 로그를 기록한 파일
  - 운영체제가 동작하며 발생하는 하드웨어 및 소프트웨어의 이벤트와 에러를 기록

```
Dec 30 03:23:37 ubuntu kernel: [0.756396] TCP reno registered
Dec 30 03:23:37 ubuntu kernel: [0.756473] NET: Registered protocol family 1
Dec 30 03:23:37 ubuntu kernel: [0.760091] Trying to unpack rootfs image as initramfs...
Dec 30 03:23:37 ubuntu kernel: [1.016091] Freeing initrd memory: 7481k freed
Dec 30 03:23:37 ubuntu kernel: [1.016168] Simple Boot Flag at 0x36 set to 0x1
Dec 30 03:23:37 ubuntu kernel: [1.016245] cpufreq-nforce2: No nForce2 chipset.
Dec 30 03:23:37 ubuntu kernel: [1.016322] Scanning for low memory corruption every 60 seconds
Dec 30 03:23:37 ubuntu kernel: [1.016447] audit: initializing netlink socket (disabled)
Dec 30 03:23:37 ubuntu kernel: [1.016524] type=2000 audit(1262172212.016:1): initialized
Dec 30 03:23:37 ubuntu kernel: [1.020745] HugeTLB registered 4 MB page size, pre-allocated 0 pages
Dec 30 03:23:37 ubuntu kernel: [1.021001] VFS: Disk quotas dquot_6.5.2
Dec 30 03:23:37 ubuntu kernel: [1.021078] Dquot-cache hash table entries: 1024 (order 0, 4096 bytes)
Dec 30 03:23:37 ubuntu kernel: [1.021344] fuse init (API version 7.12)
Dec 30 03:23:37 ubuntu kernel: [1.021421] msgmni has been set to 993
Dec 30 03:23:37 ubuntu kernel: [1.022133] alg: No test for stdrng (krng)
Dec 30 03:23:37 ubuntu kernel: [1.022247] io scheduler noop registered
```

시스템 로그 파일

- /var/log/auth.log
  - 관리자 권한으로 실행된 명령에 대해 시간, 계정명, 시도한 작업, 실패 여부 등을 저장
  - sudo 명령을 통해 관리자 권한으로 동작한 것 뿐만 아니라 passwd와 같이 SUID, SGID가 설정되어 관리자 권한으로 실행된 명령어도 기록

```
Dec 30 08:15:38 ubuntu groupadd[4844]: new group: name=gdfriend, GID=1001
Dec 30 08:15:38 ubuntu useradd[4848]: new user: name=gdfriend, UID=1001, GID=100
1, home=/home/gdfriend, shell=/bin/bash
Dec 30 08:15:42 ubuntu passwd[4855]: pam_unix(passwd:chauthtok): password change
d for gdfriend
Dec 30 08:16:19 ubuntu sudo: n0fate : TTY=pts/5 ; PWD=/etc ; USER=root ; COMMAN
D=/usr/sbin/adduser users gdfriend
Dec 30 08:16:45 ubuntu sudo: n0fate : TTY=pts/5 ; PWD=/etc ; USER=root ; COMMAN
D=/usr/sbin/adduser gdfriend users
```

/proc 파일 시스템

- **/proc 파일 시스템**
  - 리눅스 시스템이 동작하기 위한 커널 정보를 메모리에 계층 구조로 구현한 것
  - 메모리에만 존재하며 일반 사용자 및 관리자가 커널의 데이터를 쉽게 접근할 수 있도록 한다.
  - 현재 동작 중인 프로세스 정보, CPU 사용, 인터럽트, I/O 포트 정보, 장치정보 등을 쉽게 가져올 수 있음

/proc 파일 시스템

- /proc 파일 시스템의 내용은 시스템 정보와 프로세스 정보로 구분
- 숫자로 구성된 디렉토리 : 현재 동작하거나 시스템에서 동작했었던 프로세스 ID
- 각 디렉토리 : 프로세스의 동작과 관련된 파일
- 그 외 : 시스템에 관련된 정보

```
n0fate@ubuntu:/proc$ ls
1      1591  18    39    886        driver          net
10     16    1808  4     888        execdomains     pagetypeinfo
1035   1633  187   40    889        fb              partitions
1038   1636  188   4064  892        filesystems     sched_debug
1039   1637  19    41    9          fs              schedstat
1059   1679  1925  414   902        interrupts      scsi
1063   1680  2     418   903        iomem           self
1068   1682  20    42    910        ioports         slabinfo
1070   17    21    43    9634       irq             softirqs
11     1714  22    5     9638       kallsyms        stat
1153   1717  23    6     986        kcore           swaps
11601  1718  24    604   987        key-users       sys
1163   1720  25    628   989        kmsg            sysrq-trigger
1164   1723  26    667   acpi       kpagecount      sysvipc
12     1725  27    7     asound     kpageflags      timer_list
1217   1731  277   753   binder     latency_stats   timer_stats
1233   1734  28    758   buddyinfo  loadavg         tty
13     1742  29    8     bus        locks           uptime
1325   1745  3     800   cgroups    mdstat          version
1365   1750  32    830   cmdline    meminfo         version_signature
14     1760  34    835   cpuinfo    misc            vmallocinfo
1416   1775  35    836   crypto     modules         vmmemctl
1431   1777  358   877   devices    mounts          vmstat
15     1782  37    880   diskstats  mpt             zoneinfo
1569   1794  38    885   dma        mtrr
```

/proc 파일 시스템

- cmdline
  - 부팅할 때 사용했던 커널 파라메터를 출력

```
n0fate@ubuntu:/proc$ cat cmdline
BOOT_IMAGE=/boot/vmlinuz-2.6.31-14-generic root=/dev/sda1 ro quiet splash
```

- 부트 이미지로 /boot/vmlinuz-2.6.31-14-generic 파일을 이용했으며 시스템의 루트는 /dev/sda1으로 부팅할 때 읽기 전용(ro)으로 마운트 했음을 알 수 있다. 부팅 중에 부팅 내용을 콘솔로 출력하지 않았으며(quiet), 부팅할 때 고해상도의 화면으로 출력(splash)했음을 알 수 있다.

/proc 파일 시스템

- cpuinfo
  - 시스템에 장착되어 있는 모든 프로세서에 대한 정보를 보관
  - 멀티 프로세서 환경에서 동작하는 도구를 사용하거나 엔디안 문제가 발생할 가능성이 있는 경우 프로세서의 내용을 확인해야 한다.

```
n0fate@ubuntu:/proc$ cat cpuinfo
processor : 0
vendor_id : GenuineIntel
cpu family : 6
model : 15
model name :
ntel(R) Core(TM)2 Quad CPU Q6600 @ 2.40GHz
stepping : 11
cpu MHz : 2394.060
cache size : 4096 KB
fdiv_bug : no
hlt_bug : no
f00f_bug : no
coma_bug : no
fpu : yes
fpu_exception : yes
cpuid level : 10
wp : yes
```

```
flags : fpu vme de pse tsc msr pae mce cx8
apic mtrr pge mca cmov pat pse36 clflush dts
acpi mmx fxsr sse sse2 ss nx constant_tsc up
arch_perfmon pebs bts tsc_reliable pni ssse3
hypervisor
bogomips : 4788.12
clflush size : 64
power management:
```

/proc 파일 시스템

- diskstats
  - 디스크 I/O에 대한 통계
  - 굵게 표시한 부분에서 앞부분은 읽은 섹터의 개수, 뒷부분은 쓰여진 섹터의 개수

```
8 0 sda 21657 9635 824657 141032 9031 58465 538352 97436 0 83384 238468
8 1 sda1 21464 7889 815337 140336 8744 56605 522792 96920 0 82632 237256
8 2 sda2 3 0 6 20 0 0 0 0 0 20 20
8 5 sda5 171 1685 8674 632 85 1860 15560 516 0 864 1148
```

/proc 파일 시스템

- driver/rtc
  - 하드웨어적인 시스템 클럭인 RTC(Real Time Clock)의 값

```
n0fate@ubuntu:/proc$ cat driver/rtc
rtc_time : 23:11:43
rtc_date : 2009-12-30
alrm_time : 00:00:00
alrm_date : ****-**-**
alarm_IRQ : no
alrm_pending : no
24hr : yes
periodic_IRQ : no
update_IRQ : no
HPET_emulated : no
DST_enable : no
periodic_freq : 1024
batt_status : okay
```

/proc 파일 시스템

- filesystems
  - 커널에서 지원하는 파일 시스템
  - NODEV로 나타나는 부분은 별도의 물리적 장치가 필요없음

```
0fate@ubuntu:/proc$ cat filesystems
nodev sysfs
nodev rootfs
nodev proc
nodev usbfs
nodev pipefs
ext3
ext2
ext4
nodev ramfs
nodev vmblock
```

/proc 파일 시스템

- /kallsyms(커널 2.6), /ksyms(커널 2.4)
  - 커널에서 사용되는 전역 변수, 함수 등의 심볼을 보관

```
n0fate@ubuntu:/proc$ more kallsyms
c0100000 T startup_32
c0100000 T _text
c0100079 t bad_subarch
c0100079 W lguest_entry
c0100079 W xen_entry
c010007b t default_entry
c0101000 T wakeup_pmode_return
c010104c t bogus_magic
c010104e t save_registers
c010109d t restore_registers
```

/proc 파일 시스템

- kcore
  - 프로그램에서 크래쉬가 발생했을 때 생성되는 "core" 파일과 동일한 형식의 파일
  - 커널의 물리적 메모리를 나타내며 GDB(GNU Debugger)를 통해 디버깅할 수 있는 형태로 제공
- modules
  - 커널에 로드된 모든 모듈의 이름과 커널 주소

```
n0fate@ubuntu:/proc$ cat modules
vmblock 12444 1 - Live 0xe0a21000
vsock 39616 0 - Live 0xe09cc000
vmmemctl 8604 0 - Live 0xe08dd000
vmhgfs 51592 0 - Live 0xe0932000
pvscsi 14276 0 - Live 0xe087b000
acpiphp 22480 0 - Live 0xe08b1000
snd_ens1371 22016 2 - Live 0xe0884000
gameport 11368 1 snd_ens1371, Live 0xe0850000
```

/proc 파일 시스템

- mounts
  - NFS를 포함하여 현재 마운트 되어 있는 모든 장치

```
n0fate@ubuntu:/proc$ cat mounts
rootfs / rootfs rw 0 0
none /sys sysfs rw,nosuid,nodev,noexec,relatime 0 0
none /proc proc rw,nosuid,nodev,noexec,relatime 0 0
udev /dev tmpfs rw,relatime,mode=755 0 0
/dev/sda1 / ext4 rw,relatime,errors=remount-ro,barrier=1,data=ordered 0 0
none /sys/kernel/security securityfs rw,relatime 0 0
none /sys/fs/fuse/connections fusectl rw,relatime 0 0
none /sys/kernel/debug debugfs rw,relatime 0 0
none /dev/pts devpts rw,nosuid,noexec,relatime,gid=5,mode=620,ptmxmode=000 0 0
none /dev/shm tmpfs rw,nosuid,nodev,relatime 0 0
none /var/run tmpfs rw,nosuid,relatime,mode=755 0 0
none /var/lock tmpfs rw,nosuid,nodev,noexec,relatime 0 0
none /lib/init/rw tmpfs rw,nosuid,relatime,mode=755 0 0
none /proc/fs/vmblock/mountPoint vmblock rw,relatime 0 0
```

/proc 파일 시스템

- partitions
  - 시스템의 각 파티션에 대한 정보

```
n0fate@ubuntu:/proc$ cat partitions
major minor #blocks name
8 0 15728640 sda
8 1 15020743 sda1
8 2 1 sda2
8 5 706828 sda5
```

/proc 파일 시스템

- sys 디렉토리
  - 커널 내부에서 사용하는 변수에 대한 정보를 파일로 보관
  - 각 장치와 파일 시스템, 모듈 등에서 사용되는 내부 값을 열람 및 수정 가능

```
n0fate@ubuntu:/proc$ ls /sys -l
total 0
drwxr-xr-x 2 root root 0 2009-12-30 04:18 block
drwxr-xr-x 21 root root 0 2009-12-30 03:23 bus
drwxr-xr-x 43 root root 0 2009-12-30 03:23 class
drwxr-xr-x 4 root root 0 2009-12-30 04:18 dev
drwxr-xr-x 10 root root 0 2009-12-30 03:23 devices
drwxr-xr-x 4 root root 0 2009-12-30 04:18 firmware
drwxr-xr-x 5 root root 0 2009-12-30 03:23 fs
drwxr-xr-x 6 root root 0 2009-12-30 03:23 kernel
drwxr-xr-x 97 root root 0 2009-12-30 03:24 module
drwxr-xr-x 2 root root 0 2009-12-30 04:18 power
```

/proc 파일 시스템

- uptime
  - 시스템이 동작한 시간을 나타내는 값과 그 중 유휴(Idle) 시간이 얼마나 되는지에 대한 두 개의 시간 값을 가지고 있다. 각 시간은 초단위로 기록된다.

```
n0fate@ubuntu:/proc$ cat uptime
43853.44 40830.35
```

- version
  - gcc 버전과 리눅스 커널의 버전을 가지고 있다.

```
n0fate@ubuntu:/proc$ cat version
Linux version 2.6.31-14-generic (buildd@rothera) (gcc version 4.4.1 (Ubu
ntu 4.4.1-4ubuntu8) ) #48-Ubuntu SMP Fri Oct 16 14:04:26 UTC 2009
```

/proc 파일 시스템

- meminfo
  - 메모리의 사용량 및 상태에 대한 통계 값

```
n0fate@ubuntu:/proc$ cat meminfo
MemTotal: 509336 kB
MemFree: 35616 kB
Buffers: 30524 kB
Cached: 235596 kB
SwapCached: 3012 kB
Active: 221300 kB
Inactive: 216264 kB
Active(anon): 56224 kB
Inactive(anon): 115544 kB
Active(file): 165076 kB
Inactive(file): 100720 kB
 . . .
```

/proc 파일 시스템

- 프로세스 정보
  - 각 프로세스 ID의 디렉토리는 하나의 프로세스가 동작하기 위해 필요한 실행 파일, 연결된 파일, 네트워크 등의 정보를 저장
- 각 프로세스 ID 디렉토리는 기존에 실행되었던 프로세스도 남아있는 경우가 있어 침해 사고 분석에 매우 유용
- exe
  - 실제 실행 파일의 이미지에 대한 심볼릭 링크이다.

```
n0fate@ubuntu:/proc/1745$ ls exe -la

lrwxrwxrwx 1 n0fate n0fate 0 2009-12-30 15:43 exe -> /usr/bin/kdeinit4
```

- cmdline
  - 프로세스가 시작할 때, 입력한 실행 파일명과 파라메터를 포함하는 명령어 구문

```
n0fate@ubuntu:/proc/1745$ cat cmdline

kdeinit4: krunner [kdeinit]
```

/proc 파일 시스템

- cwd
  - 프로세스의 작업 디렉토리의 심볼릭 링크

```
lrwxrwxrwx 1 n0fate n0fate 0 2009-12-30 15:43 cwd -> /home/n0fate/Documents
```

- environ
  - 프로세스를 위한 환경 변수를 보관하고 있다.

```
n0fate@ubuntu:/proc/1745$ cat environ | xargs -0 -n 1
]
.
FULL_SESSION=true
GS_LIB=/home/n0fate/.fonts
DM_CONTROL=/var/run/xdmctl
KDEDIRS=/usr/share/kubuntu-netbook-default-settings/:/usr/share/kubuntu-default-set
tings/kde4-profile/default/
USER=n0fate
SSH_AGENT_PID=1633
SHLVL=0
HOME=/home/n0fate
 . . .
```

/proc 파일 시스템

- fd 디렉토리
  - 프로세스와 관련된 모든 파일의 디스크립터에 대한 정보를 저장
  - 0, 1, 2는 표준 입·출력과 표준 에러

```
n0fate@ubuntu:/proc/1745$ ls fd
0  1  10  11  12  13  14  15  2  3  4  5  6  7  8  9
```

- loginuid
  - 시스템에 접속한 UID
  - 만약 공격자가 일반 사용자 계정으로 로그인 후, su 또는 sudo 등의 명령어를 이용하여 root 권한으로 프로그램을 실행시켜도 loginuid에는 최초 접속했던 계정의 UID가 기록
  - 이 값을 사용하기 전에 커널에서 이 값을 사용할 수 있도록 감사 기능이 설정되어야 한다.

활성 시스템 조사

- 리눅스 활성 데이터 수집을 위한 쉘 스크립트
- 수행 과정
  - 현재 시간 및 부팅 이후 사용 시간, CPU, 운영체제, 실행 중인 프로세스, 연결된 장치, 네트워크 인터페이스 카드, 네트워크 연결, 파일시스템, 사용자 접속 기록, 명령어 history 등의 정보를 information.evi 파일에 저장
  - 저장된 파일을 netcat을 이용하여 외부의 수집 서버(IP: 123.456.777.890, Port: 7777)로 전송
  - 해당 파일을 삭제하는 과정이 수행

활성 시스템 조사

```
#!/bin/sh
# 현재시각, uptime
date >> ./infomation.evi
uptime >> ./infomation.evi
# cpu 정보
cat /proc/cpuinfo >> ./infomation.evi
# 운영체제 정보 (커널 버전)
cat /proc/version >> ./infomation.evi
# 실행중인 프로세스
ps aux >> ./infomation.txt
```

활성 시스템 조사

연결된 장치 정보 : IDE, NIC, PCI 등

cat /proc/scsi/scsi >> ./infomation.evi

cat /proc/ioports >> ./infomation.evi

dmesg | grep NIC >> ./infomation.evi

dmesg | grep PCI >> ./infomation.evi

네트워크 인터페이스 카드 정보

ifconfig -a >> ./infomation.evi

네트워크 연결 정보

netstat -a >> ./infomation.evi

활성 시스템 조사

```
# 파티션 및 파일시스템

df >> ./infomation.txt

cat /proc/partitions >> ./infomation.evi

cat /proc/filesystems >> ./infomation.evi

# 콘솔, 네트워크로 연결된 사용자 접속 기록

last |  grep tty >> ./infomation.evi

last |grep pts >> ./infomation.evi

# 현재 접속 중인 사용자 목록

who >> ./infomation.evi
```

활성 시스템 조사

명령어 History

history >> ./infomation.evi

netcat을 이용하여 수집된 데이터를 외부 서버로 전송

cat ./infomation.evi | nc 123.456.777.890 7777

수집 데이터 삭제

rm -rf ./infomation.evi

Q & A