6장

Permission 이해 및 구성하기

전체 내용

파일 Permission 확인하기

파일 Permission 변경하기

기본 접근 권한 확인하기

1 – 파일 Permission 확인하기

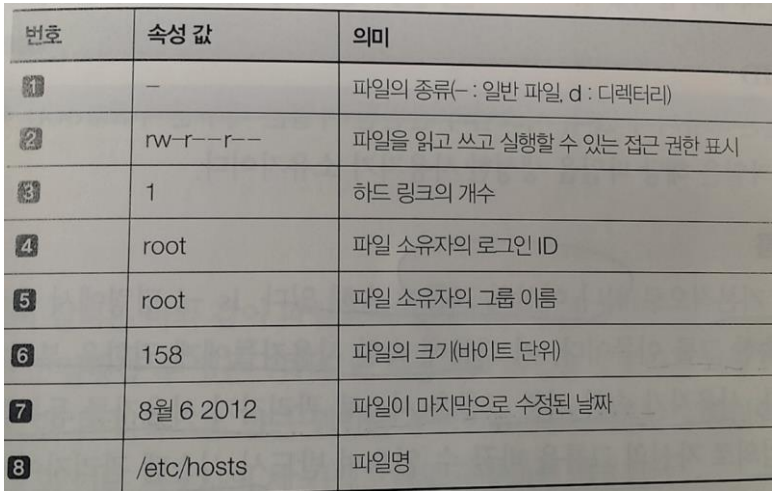
파일 속성 이해하기

파일의 접근 권한

1 – 파일 Permission 확인하기

- 파일 속성 이해하기
  - 개요
    - Linux 시스템은 여러 사람이 동시에 접속하여 작업을 하기 때문에 UserA가 만든 파일을 UserB가 삭제할 수 있다
    - 이렇기 때문에 파일에 대한 접근 권한을 설정하여 파일을 보호할 필요가 있다
    - 이런 것을 접근 제어, Permission 설정하기라고 한다
  - 파일 속성
    - **ls -l /etc/hosts**

      -rw-r--r-- 1 root root 246 6월 2 21:42 /etc/hosts



번호	속성 값	의미
1	–	파일의 종류(– : 일반 파일, d : 디렉터리)
2	rw-r--r--	파일을 읽고 쓰고 실행할 수 있는 접근 권한 표시
3	1	하드 링크의 개수
4	root	파일 소유자의 로그인 ID
5	root	파일 소유자의 그룹 이름
6	158	파일의 크기(바이트 단위)
7	8월 6 2012	파일이 마지막으로 수정된 날짜
8	/etc/hosts	파일명

1 – 파일 Permission 확인하기

- 파일 속성 이해하기
  - 파일 접근 권한
    - -rw-r--r-- 1 root root 246 6월 2 21:42 /etc/hosts
    - 이 파일의 **소유자**는 **사용자 root**의 접근 권한은 [읽기,쓰기]
    - 이 파일을 소유한 그룹(**소유 그룹**)인 **그룹 root**의 접근 권한은 [읽기]
    - **그 외 사용자들**의 접근 권한은 [읽기]
  - 특정한 사용자가 소속된 **[기본 그룹]** 및 **[2차 그룹]들** 확인하기
    - **groups** suser1
      suser1 : SalesGroup PublicGroup
    - **id** suser1
      uid=515(suser1) gid=515(SalesGroup) groups=515(SalesGroup),517(PublicGroup)
  - 리눅스 시스템에 저장된 **모든 사용자 계정** 확인하기
    - tail **/etc/passwd**
  - 리눅스 시스템에 저장된 **모든 그룹 계정** 확인하기
    - tail **/etc/group**

1 – 파일 Permission 확인하기

- 파일의 접근 권한 예제
  - Directory 생성하기(/public/SalesData, /public/EngData)
    - Root 계정으로 작업
    - **mkdir -p** /public/SalesData /public/EngData
  - 생성한 디렉터리에 파일 생성하기
    - **echo "This is sales data." > /public/SalesData/sales.txt**
    - echo "This is eng data." > /public/EngData/eng.txt
    - echo "This is public data." > /public/public.txt
  - 생성한 파일 내용 보기
    - **cat /public/SalesData/sales.txt**
    - cat /public/EngData/eng.txt
    - cat /public/public.txt
  - 생성한 파일의 사용 권한 확인하기
    - **ls –l /public/SalesData/sales.txt**
      -rw-r--r--. 1 root root 18 2015-10-13 01:30 sales.txt

1 – 파일 Permission 확인하기

- 파일의 접근 권한 예제(계속)
  - 생성한 2개의 파일의 Permission 수정하기
    - **chmod 640** /public/**SalesData/sales.txt**
    - **chmod 640** /public/**EngData/eng.txt**
    - **chmod 640** /public/**public.txt**
  - Adminuser로 전환하여 해당 파일 내용 보기Directory
    - **su adminuser**
    - **cat /public/SalesData/sales.txt**
    - ** 접근 실패

1 – 파일 Permission 확인하기

- 파일의 접근 권한 예제
  - Group 생성하기: groupadd 그룹이름
    - **groupadd SalesGroup**
      groupadd **EngGroup**
      groupadd **PublicGroup**
    - tail /etc/group
  - User 생성하기
    - useradd -m -d /home/suser1 suser1
      passwd suser1
    - useradd -m -d /home/suser2 suser2
      passwd suser2
    - useradd -m -d /home/euser1 euser1
      passwd euser1
    - useradd -m -d /home/euser2 euser2
      passwd euser2

1 – 파일 Permission 확인하기

- 파일의 접근 권한 예제(계속)
  - 사용자 계정을 그룹의 구성원으로 추가하기(usermod –G 그룹 사용자)
    - SalesGroup에는 suser1, suser2, EngGroup에는 euser1, euser2, PublicGroup에는 suser1, suser2, euser1, euser4를 추가한다
    - 사용자 계정의 [2차 그룹]의 구성원을 추가하는 작업이다
    - **usermod –G** SalesGroup, PublicGroup suser1
      usermod –G SalesGroup, PublicGroup suser2
    - **gpasswd –a** euser1 EngGroup
      gpasswd –a euser2 EngGroup
      gpasswd –a euser1 PublicGroup
      gpasswd –a euser2 PublicGroup
  - SalesGroup, EngGroup, PublicGroup의 구성원들(사용자들) 확인하기
    - cat /etc/group **| grep SalesGroup**
      SalesGroup:x:515:suser1,suser2
    - cat /etc/group **| grep EngGroup**
      EngGroup:x:516:euser1,euser2
    - cat /etc/group **| grep PublicGroup**
      PublicGroup:x:517:suser1,suser2,euser1,euser2

1 – 파일 Permission 확인하기

- 파일의 접근 권한 예제(계속)
  - 사용자 계정의 2차 그룹 바꿔치기(Overwrite)
    - 사용자를 생성하면 사용자와 이것이 소속된 [기본 그룹]과 [2차 그룹]이 모두 동일하다
      useradd imsiuser1
      id imsiuser1
      uid=510(imsiuser1) gid=512(imsiuser1) groups=512(imsiuser1)
    - SalesGroup에 imsiuser1을 **추가하기**
      gpasswd –a imsiuser1 SalesGroup
      id imsiuser1
      *uid=510(imsiuser1) gid=512(imsiuser1) groups=512(imsiuser1),**503(SalesGroup)***
    - EngGroup에 imsiuser1을 **추가하기**
      gpasswd –a imsiuser1 EngGroup
      id imsiuser1
      *gid=512(imsiuser1) groups=512(imsiuser1),503(SalesGroup),**504(EngGroup)***
    - imsiuser1의 2차 그룹을 adminuser 그룹으로 **바꿔치기**
      usermod –G adminuser imsiuser1
      id imsiuser1
      *uid=510(imsiuser1) gid=512(imsiuser1) groups=512(imsiuser1),**501(adminuser)***

1 – 파일 Permission 확인하기

- 파일의 접근 권한 예제(계속)
  - 사용자 계정의 그룹 구성원을 초기화하기
    - **usermod –G imsiuser1 imsiuser1**
      id imsiuser1
      uid=510(imsiuser1) gid=512(imsiuser1) groups=512(imsiuser1)

1 – 파일 Permission 확인하기

- 파일의 접근 권한 예제(계속)
  - 사용자 계정이 소속된 [기본 그룹]의 역할 이해하기
    - 사용자 계정을 생성하면 기본적으로 **한 개의 [기본 그룹]** 및 **여러 개의 [2차 그룹]**에 소속된다
      id suser1
      uid=503(suser1) gid=505(suser1) groups=505(suser1),503(SalesGroup)
    - 여기서 suser1의 기본 그룹은 suser1이고, 2차 그룹은 suser1,SalesGroup이다
    - [기본 그룹]은 사용자가 파일을 만들 때 그 파일의 소유 그룹이 되고, 사용자의 기본 그룹은 하나만 될 수 있다
    - [2차 그룹/보충 그룹]은 사용자가 속할 수 있는 그룹들이다. 사용자가 파일에 액세스할 때 그 파일의 소유 그룹에 접속하는 사용자의 2차 그룹이 포함되면 자원에 액세스할 수 있게 된다.

1 – 파일 Permission 확인하기

- 파일의 접근 권한 예제(계속)
  - 사용자 계정의 [기본 그룹] 변경하기: usermod –g 그룹 사용자
    - -g는 gid를 말한다. 즉, 1차 그룹을 변경하겠다는 뜻이다
    - **usermod -g SalesGroup suser1**
    - **usermod -g SalesGroup suser2**
    - **usermod -g EngGroup euser1**
    - **usermod -g EngGroup euser2**

1 – 파일 Permission 확인하기

- 파일의 접근 권한 예제(계속)
  - 특정한 사용자의 세부 정보 확인하기: 기본 그룹 확인
    - **id suser1**
      uid=515(suser1) ***gid=515(SalesGroup)***
      groups=515(SalesGroup),517(PublicGroup)
    - **su - suser1**
      touch suser1imsi.txt
      ls -l suser1imsi.txt
      -rw-r--r--. 1 **suser1 SalesGroup** 0 Oct 14 14:28 suser1imsi.txt
    - 사용자의 1차 그룹(gid)는 파일을 생성할 때 소유 그룹으로 사용된다

1 – 파일 Permission 확인하기

- 파일의 접근 권한 예제(계속)
  - 사용자가 파일/디렉터리에 접근할 때의 사용 권한 적용 여부 확인
    - 사용자가 디렉터리 및 파일에 접근할 때는 파일/디렉터리의 소유자 권한을 먼저 보고, 아니면 그 다음에 소유 그룹 권한을 검사한다. 이 때 사용자가 소속된 **[2차 그룹]**이 **소유 그룹과 동일하면** 이 권한이 적용된다
    - 접근하는 사용자가 파일의 소유자가 아니고, 접속하는 사용자가 소속된 [2차 그룹]이 파일의 소유그룹이 아닌 경우에는 최종적으로는 파일의 Others 권한이 사용자에게 적용된다
  - Public, SalesData, EngData 디렉터리의 [소유 그룹] 변경하기
    - 각 디렉터리의 사용권한 변경하기
      chgrp –R PublicGroup /public/
      chgrp –R SalesGroup /public/SalesData/
      chgrp –R EngGroup /public/EngData/
    - 현재 디렉터리의 사용 권한 확인하기
      ls -l /public
      drwxr-xr-x. 2 root EngGroup 4096 Oct 13 01:23 EngData
      -rw-r--r--. 1 root PublicGroup 21 Oct 14 14:48 public.txt
      drwxr-xr-x. 2 root SalesGroup 4096 Oct 13 01:24 SalesData

1 – 파일 Permission 확인하기

- 파일의 접근 권한 예제(계속)
  - Root 사용자가 생성한 파일을 SalesGroup 및 EngGroup의 구성원 사용자로 접근하여 파일 읽기
    - **su suser1**
      cat /public/public.txt (## 성공)
      cat /public/SalesData/sales.txt (## 성공)
      cat /public/EngData/eng.txt (## 실패)
    - **Su euser2**
      cat /public/public.txt (## 성공)
      cat /public/EngData/eng.txt (## 성공)
      cat /public/SalesData/sales.txt (## 실패)
  - suser1의 2차 그룹은 PublicGroup, SalesGroup이기 때문에 public.txt와 sales.txt에 모두 액세스 가능한 것이다.
    - 사용자가 파일에 액세스할 때는 사용자가 속한 [2차 그룹]이 그 파일의 [소유 그룹]과 같으면 그것이 적용되고, 다르면 Others 권한이 적용된다.

1 – 파일 Permission 확인하기

- 파일의 접근 권한
  - 읽기 권한
    - 파일 내용을 변경할 수 없지만 파일 내용은 볼 수 있는 권한
    - 파일 내용을 공개하면 안 될 사용자 및 그룹에게 읽기 권한을 주지 않는다
  - 쓰기 권한
    - 파일 내용을 볼 수 있을 뿐 아니라, 수정하거나 삭제할 수 있다
    - 함부로 다른 사용자 및 그룹 구성원들에게 쓰기 권한을 주지 않는다
  - 실행 권한
    - 파일 실행 권한이 없으면 명령어도 실행할 수 없다
    - 디렉터리에 실행 권한이 없으면 change directory로 이동할 수 없다
    - 디렉터리에 실행 권한만 있고 읽기 및 쓰기 권한이 없으면 ls –la 명령도 원하는 결과를 불러 오지 못한다
    - 디렉터리에 쓰기 권한이 있어야만 파일을 저장할 수 있다
    - 디렉터리에 읽기 권한이 있어야만 ls –la 가 실행된다

1 – 파일 Permission 확인하기

- 파일의 접근 권한
  - 파일과 디렉터리 접근 권한

권한	파일	디렉터리
읽기	파일을 읽거나 복사할 수 있다.	ls 명령으로 디렉터리 목록을 볼 수 있다(ls 명령의 옵션은 실행 권한이 있어야 사용할 수 있다).
쓰기	파일을 수정, 이동, 삭제할 수 있다(디렉터리에 쓰기 권한이 있어야 한다).	파일을 생성하거나 삭제할 수 있다.
실행	파일을 실행할 수 있다(셸 스크립트나 실행 파일의 경우).	cd 명령을 사용할 수 있다. 파일을 디렉터리로 이동하거나 복사할 수 있다.

  - 다양한 접근 권한 조합의 예

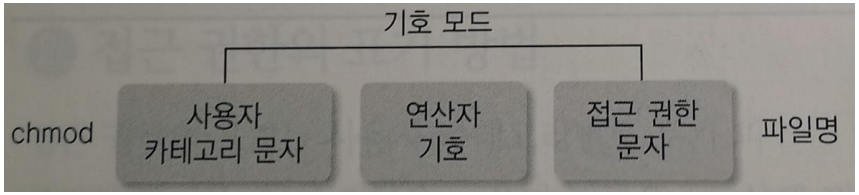
접근 권한	의미
rwxr-xr-x	소유자는 읽기, 쓰기, 실행 권한을 모두 가지고 있고 그룹과 기타 사용자는 읽기와 실행 권한만 가지고 있다.
r-xr-xr-x	소유자, 그룹, 기타 사용자 모두 읽기와 실행 권한만 가지고 있다.
rw-------	소유자만 읽기, 쓰기 권한을 가지고 있고 그룹과 기타 사용자는 아무 권한이 없다.
rw-rw-rw-	소유자, 그룹, 기타 사용자 모두 읽기와 쓰기 권한을 가지고 있다.
rwxrwxrwx	소유자, 그룹, 기타 사용자 모두 읽기, 쓰기, 실행 권한을 가지고 있다.
rwx------	소유자만 읽기, 쓰기, 실행 권한을 가지고 있고 그룹과 기타 사용자는 아무 권한이 없다.
r--------	소유자만 읽기 권한을 가지고 있다.

2 – 파일 Permission 변경하기

- 기호를 이용한 파일 접근 권한 변경하기
- 숫자를 이용한 파일 접근 권한 변경하기

2 – 파일 Permission 변경하기

- 기호를 이용한 파일 접근 권한 변경하기
  - 접근 권한을 변경하는 명령어는 chmod이다
    - 하위 디렉터리까지 변경하려면 –r 옵션을 사용한다



구분	문자/기호	의미
사용자 카테고리 문자	u	파일 소유자
	g	소유자가 속한 그룹
	o	소유자와 그룹 이외의 기타 사용자
	a	전체 사용자
연산자 기호	+	권한 부여
	–	권한 제거
	=	접근 권한 설정
접근 권한 문자	r	읽기 권한
	w	쓰기 권한
	x	실행 권한

2 – 파일 Permission 변경하기

- 기호를 이용한 파일 접근 권한 변경하기
  - 기호 모드를 사용한 접근 권한 설정의 예

권한 표기	의미
u+w	소유자(u)에게 쓰기(w) 권한 부여(+)
u-x	소유자(u)의 실행(x) 권한 제거(-)
g+w	그룹(g)에 쓰기(w) 권한 부여(+)
o-r	기타 사용자(o)의 읽기(r) 권한 제거(-)
g+wx	그룹(g)에 쓰기(w)와 실행(x) 권한 부여(+)
+wx	모든 사용자에게 쓰기(w)와 실행(x) 권한 부여(+)
a+rwx	모든 사용자에게 읽기(r), 쓰기(w), 실행(x) 권한 부여(+)
u=rwx	소유자(u)에게 읽기(r), 쓰기(w), 실행(x) 권한 부여(=)
go+w	그룹(g)과 기타 사용자(o)에게 쓰기(w) 권한 부여(+)
u+x,go+w	소유자(u)에게 실행(x) 권한을 부여하고(+) 그룹(g)과 기타 사용자(o)에게 쓰기(w) 권한 부여(+)

2 – 파일 Permission 변경하기

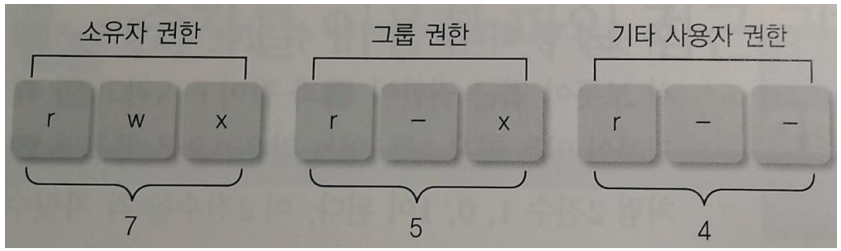
- 기호를 이용한 파일 접근 권한 변경하기
  - 소유자에게 쓰기 권한 제거하기
    - **mkdir mod6**
      cd mod6/
      cp /etc/hosts test.txt
    - **ls -l test.txt**
      -rw-r--r-- 1 adminuser adminuser 246 6월 5 00:29 test.txt
    - **chmod u-w test.txt**
      ls -l test.txt
      -r--r--r-- 1 adminuser adminuser 246 6월 5 00:29 test.txt
    - **cat > test.txt**
      bash: test.txt: Permission denied
  - **모든 사람에게** 실행 권한 부여하기
    - **chmod a+x test.txt**
      ls -l test.txt
      -r-xr-xr-x 1 adminuser adminuser 246 6월 5 00:29 test.txt

2 – 파일 Permission 변경하기

- 기호를 이용한 파일 접근 권한 변경하기
  - 소유자에게 쓰기 권한 부여, 그룹에게 쓰기 권한 제거하기
    - **chmod u+w,g-w test.txt**
      ls -l test.txt
      -rwxr-xr-x 1 adminuser adminuser 246 6월 5 00:29 test.txt
  - 사용자가 소속된 그룹(g)과 기타 사용자(o)는 test.txt 파일에 어떤 권한도 부여하지 못하도록 설정하기(=)
    - **chmod go= test.txt**
      ls -la test.txt
      -rwx-------. 1 root root 0 2015-06-06 17:05 test.txt
    - **= 옵션**: 설정을 해주지 말라는 것. 즉, group, others에는 접근 권한을 부여하지 말 것

2 – 파일 Permission 변경하기

- 숫자를 이용한 파일 접근 권한 변경하기
  - 소유자별로, 그룹별로, 다른 사람별로 개개별로 접근 권한을 부여할 때는 기호모드가 편리하다
  - 하지만 전체적으로 접근 권한을 조정할 때는 숫자 모드가 편리하다
  - 숫자로 환산하기



  - 접근 권한과 숫자의 대응 관계

접근 권한	환산	숫자	의미
rwx	111 → 4+2+1	7	읽기, 쓰기, 실행
rw–	110 → 4+2+0	6	읽기, 쓰기
r–x	101 → 4+0+1	5	읽기, 실행
r––	100 → 4+0+0	4	읽기
–wx	011 → 0+2+1	3	쓰기, 실행
–w–	010 → 0+2+0	2	쓰기
––x	001 → 0+0+1	1	실행
–––	000 → 0+0+0	0	권한이 없음

2 – 파일 Permission 변경하기

- 숫자를 이용한 파일 접근 권한 변경하기
  - 숫자로 설정할 때는 파일/디렉터리의 소유자, 소유 그룹, 그 외 사용자들을 한꺼번에 설정해야 한다. 개별적으로 설정은 안된다.
  - 소유자의 쓰기 권한을 제거하기
    - **ls -l test.txt**
      -rwxr-xr-x 1 adminuser adminuser 246 6월 5 00:29 test.txt
    - **chmod 555 test.txt**
      ls -l test.txt
      -r-xr-xr-x 1 adminuser adminuser 246 6월 5 00:29 test.txt
  - 그룹에게 쓰기 권한을 추가하기
    - **ls -l test.txt**
      -r-xr-xr-x 1 adminuser adminuser 246 6월 5 00:29 test.txt
    - **chmod 575 test.txt**
      ls -l test.txt
      -r-xrwxr-x 1 adminuser adminuser 246 6월 5 00:29 test.txt

3 – 기본 접근 권한 확인하기

- 기본 접근 권한 확인하고 변경하기
  - 파일을 생성할 때 기본 접근 권한
    - 소유자: rw-, 소유그룹: r--, 기타: r--
  - 디렉터리를 생성할 때 기본 접근 권한
    - 소유자: rwx, 소유그룹: r-x, 기타: r-x
    - ** 이렇게 설정되는 이유는 umask 값 때문이다
  - umask 사용하기
    - umask를 사용하여 파일이 생성될 때 기본 접근 권한을 설정한다
    - umask 명령어는 파일/디렉터리를 생성할 때 **부여하지 않을 권한을 지정하는 것이다**
    - Directory에 대한 기본 접근 권한 확인하기
      umask -S
      u=rwx,g=rx,o=rx
    - File에 대한 기본 접근 권한 확인하기
      umask
      0022
      rw-r--r--

3 – 기본 접근 권한 확인하기

- 기본 접근 권한 확인하고 변경하기
  - umask가 0022인 기본 상황에서 파일을 생성한다
    - touch umaskfile.txt
      ls -l umaskfile.txt
      -rw-r--r--. 1 root root 0 2015-06-06 20:00 umaskfile.txt
  - rwxrwxrx-로 나올 것으로 예상했는데 다르게 나온다
    - 그 이유는 기본적으로 셸이 파일을 만들 땐 777 mask를 가지지 않고 0666 이라는 마스크를 가지기 때문이다.
    - 이 때문에 다음과 같은 공식이 성립한다
      0666 - 0022 = 0644 = rw-r--r--
    - 즉, 보통 디렉터리에 파일을 만들면 파일의 소유자만 읽기, 쓰기가 되고, 나머지 사용자는 읽기만 가능하다

3 – 기본 접근 권한 확인하기

- 기본 접근 권한 확인하고 변경하기
  - 생성되는 파일에 대하여 소유자처럼 "소유그룹"에도 읽기, 쓰기 권한을 부여하고자 한다
    - 우리가 설정하고자 하는 것은 **rw-rw----** 이다. 즉, 소유자는 rw-, 소유그룹은 rw-, 나머지는 ---로 설정하고자 한다
    - rw-rw----은 660이므로 0666-0660=0006이다
    - umask를 새롭게 설정하기
      - **umask 0006**
      - **umask**
        0006
    - Directory를 생성할 때 기본 접근 권한 확인하기
      - **umask -S**
        u=rwx,g=rwx,o=x
    - Directory 생성하기
      - **mkdir /yslee**
        ls -ld /yslee
        drwxrwx--x. 2 root root 6 Sep 18 15:13 /yslee

3 – 기본 접근 권한 확인하기

- 기본 접근 권한 확인하고 변경하기
  - 생성되는 파일에 대하여 소유자처럼 "소유그룹"에도 읽기, 쓰기 권한을 부여하고자 한다
    - 우리가 설정하고자 하는 것은 **rw-rw----** 이다. 즉, 소유자는 rw-, 소유그룹은 rw-, 나머지는 ---로 설정하고자 한다
    - rw-rw----은 660이므로 0666-0660=0006이다
    - File을 생성할 때 기본 접근 권한 확인하기
      - **umask**
        0006
      - **touch /yslee/sample.txt**
      - **ls -l /yslee/sample.txt**
        -rw-rw----. 1 root root 0 Sep 18 15:16 /yslee/sample.txt