목차

find 명령어 사용법 3
명령어 사용법 3
사용 예제 3
용량이 0 인 비어있는 파일을 검색하는 것 3
특정 소유자의 파일 찾고자 할때 3
특정 사용자의 소유파일들을 검색 3
특정 디렉토리에서 특정 파일 검색 3
검색된 파일을 바로 삭제 3
파일용량이 큰 파일들을 모두 검색하여 처리 3
지정한 퍼미션을 가지고 있는 파일 검색 4
최근 몇분에 변경된 적이 있는 모든 파일 검색 4
최근 100 시간 이내에 사용된 적이 있는 파일(액세스된 파일) 검색 4
지정한 파일보다 더 최근에 사용된 적(액세스된 파일)이 있는 파일 검색 4
지정한 파일보다 더 최근에 변경된 적이 있는 파일 검색 4
실행가능한 파일을 검색 4
실행 가능하지만 쓰지 못하는 파일을 대상으로 검색 4
블록장치파일을 검색 4
문자장치파일을 검색 4
디렉토리를 검색 5
일반파일만을 검색 5
심볼릭 링크파일을 검색 5
root 소유의 SetUID, SetGID, StickyBit 파일을 검색 5
외부에서 제한없이 로그인할 수 있는 .rhosts 파일 검색 5
.bash_histroy 파일 검색 5
무적(소유주나 그룹이 없는)파일의 검색 5
특정 문자열을 포함하고 있는 파일을 검색 5
옵션 5
수행할작업(action) 6
연산자 6
참조링크 7

find 명령어 사용법

find는 검색기능과 여러가지 부가기능을 가진 명령어이다.

명령어 사용법

```
사용법 : find [검색대상위치] [옵션] [수행할작업]
```

사용 예제

용량이 0 인 비어있는 파일을 검색하는 것

```
find / -empty -exec ls -l {} \;
```

특정 소유자의 파일 찾고자 할때

```
find / -uid 500 -exec ls -l {} \;
```

특정 사용자의 소유파일들을 검색

```
find / -user nestgoer -print
```

'-print' 는 검색결과를 파일의 절대경로로 표시하는 반면 '-exec ls -l \;' 는 ls -l 과 같은 형식으로 검색결과를 표시한다.

특정 디렉토리에서 특정 파일 검색

```
find /home -name *.html -exec ls -l {} \;
```

검색된 파일을 바로 삭제

```
find ~nestgoer -name *.bak -exec rm -f {} \;
```

'-exec' 는 찾은 파일들을 대상으로 뒤에 나오는 명령어를 실행하라는 의미이며 {} 는 찾아진 파일들을 하나씩 대입됨을 의미한다. 따라서 검색된 파일의 개수만큼 -exec 뒤의 명령어가 실행된다.

파일용량이 큰 파일들을 모두 검색하여 처리

```
find / -size +100M -exec ls -lh {} \;
```

지정한 퍼미션을 가지고 있는 파일 검색

```
find / -perm 4755 -exec ls -l {} \;
```

최근 몇분에 변경된 적이 있는 모든 파일 검색

```
find /home/nestgoer -mmin 5 -exec ls -l {} \;
```

5 는 5분을 뜻하는데, 이때 +5 이면 5분 보다 더 이후(더 오래된 값), -5 이면 5분 보다 더 이전(덜 된 값)

최근 100 시간 이내에 사용된 적이 있는 파일(액세스된 파일) 검색

```
find /home/nestgoer -atime -100 -exec ls -l {} \;
```

지정한 파일보다 더 최근에 사용된 적(액세스된 파일)이 있는 파일 검색

```
find /home -anewer /home/nestgoer/begin -exec ls -l {} \;
```

지정한 파일보다 더 최근에 변경된 적이 있는 파일 검색

```
find /home -newer /home/nestgoer/begin -exec ls -l {} \;
```

실행가능한 파일을 검색

```
find /home -executable -print
```

실행 가능하지만 쓰지 못하는 파일을 대상으로 검색

```
find /home -executable \! -writable -print
```

블록장치파일을 검색

```
find /dev -type b -exec ls -l {} \;
```

문자장치파일을 검색

```
find /dev -type c -exec ls -l {} \;
```

디렉토리를 검색

```
find /etc -type d -print;
```

일반파일만을 검색

```
find /home/nestgoer -type f -exec ls -l {} \;
```

심볼릭 링크파일을 검색

```
find /etc -type l -exec ls -l {} \;
```

root 소유의 SetUID, SetGID, StickyBit 파일을 검색

```
find / -user root -perm 4755 -exec ls -l {} \;
```

외부에서 제한없이 로그인할 수 있는 .rhosts 파일 검색

```
find / -name .rhosts -exec ls -l {} \;
```

.bash_histroy 파일 검색

```
find / -name .bash_history -exec ls -lh {} \;
```

무적(소유주나 그룹이 없는)파일의 검색

```
find / -nouser -o -nogroup -exec ls -l {} \;
```

특정 문자열을 포함하고 있는 파일을 검색

```
find /home -type f -exec egrep -i "linux" /dev/null {} \;
```

옵션

- -empty : 비어 있는 파일, 단 일반파일 또는 일반 디렉토리에 한함
- -uid n : 지정된 n값의 UID를 갖는 파일
- -gid n : 지정된 n값의 GID를 갖는 파일

- -group 그룹명 : 지정한 그룹명에 소속된 파일
- -name : 지정된 형식의 패턴을 가지는 파일
- -newer 파일 : 지정한 파일보다 더 최근에 변경된 적이 있는 파일
- -used n : 파일이 변경된 이후 최근 n일 이후에 액세스된 적이 있는 파일
- -user uname : 지정한 uname의 소유로 된 모든 파일(UID로 지정가능)
- -amin n : 최근 n분 이전에 액세스된 적이 있는 파일
- -anewer 파일 : 지정한 파일보다 더 최근에 액세스된 적이 있는 파일
- -atime n : 최근 n*24 시간 전에 액세스된 적이 있는 파일
- -mmin n : 최근 n분 이전에 변경된 적이 있는 파일
- -mtime n : 최근 n*24 시간 이전에 변경된 적이 있는 파일
- -executable : 실행 가능한 파일
- -readable : 읽기 가능한 파일
- -writable : 쓰기 가능한 파일
- -fstype type : 지정한 파일시스템타입(type)에 존재하는 파일
- -nogroup : 소속그룹이 존재하지 않는 파일
- -nouser : 소속 사용자(ID)가 존재하지 않는 파일
- -path 패턴 : 파일명이 패턴에 매칭되는 파일
- -perm 모드 : 모드에서 지정된 퍼미션과 일치하는 퍼미션을 가진 파일만 해당됨.
- -perm -mode : 모드에서 지정된 퍼미션을 가진 모든 파일.
- -regex 패턴 : 패턴에서 지정한 정규표현식에 매칭되는 파일
- -size n[cwbkMG] : 지정된 n 단위의 디스크공간을 사용하는 파일. 디폴트는 b
- -type 타입 : 지정한 파일타입에 해당하는 파일. b,c,d,p,f,l,s 등이 있음

위의 n 값은 +n 은 n 보다 큰 값을 의미하고, -n 은 n 값보다 작은 값을 의미하며, n 은 정확하게 n 값만을 의미한다.

수행할작업(action)

- -delete : 검색된 파일들을 삭제
- -exec command ; : 지정한 command 를 실행
- -exec command {} + : 검색된 파일들을 차례대로 매칭하면서 지정된 command 를 수행
- -ls : 'ls - dils' 를 수행
- -ok command ; : -exec 와 같지만 실행하기 전에 먼저 사용자에게 표준입력으로 수행할 command 를 입력 받는다. 만약 입력이 없다면 지정된 command 를 수행한다.
- -print : 검색된 파일들에 대하여 전체파일 경로로 파일리스트를 표준출력으로 출력한다.
- -printf format : 검색된 파일들에 대하여 지정된 format 형식으로 표준출력으로 출력

연산자

- ( ) : 괄호( )안의 연산이 먼저 행해짐
- ! expr : expr 이 거짓일때 참, 참일때 거짓이 됨.
- -not expr : 위의 ! expr 과 동일함 (POSIX 호환용)
- expr1 expr2 : AND 의 의미. expr1 이 거짓일 때에는 expr2 는 평가되지 않음
- expr1 -a expr2 : 'expr1 expr2' 와 같음
- expr1 -and expr2 : 'expr1 expr2' 와 같음 (POSIX 호환용)
- expr1 -o expr2 : OR 의 의미. expr1 이 참일 경우 expr2 는 평가되지 않음.
- expr1 -or expr2 : 'expr1 -o expr2' 와 같은 의미

참조링크

- 출처: https://webdir.tistory.com/155

From:
https://atl.kr/dokuwiki/ - **AllThatLinux!**

Permanent link:
https://atl.kr/dokuwiki/doku.php/find_%EB%AA%85%EB%A0%B9%EC%96%B4_%EC%82%AC%EC%9A%A9%EB%B2%95

Last update: **2020/02/21 03:40**