

2. 리눅스 Server 구축 및 실무 운영

2.7. NFS Server와 AutoMount

네트워크 파일 시스템의 일종인 NFS(Network File System)는 유닉스 환경에서 네트워크를 통해 파일과 응용 프로그램을 호스트간 공유하게 해주는 서비스이다.

대형 서비스 환경 구축 시 공유 파일 서버를 사용하여 데이터의 일관성을 유지 하는 경우가 많은데 이때 주로 사용되어지는 기술이다.

여기서는 실무에서 주로 사용되어지는 NFS 서버 구축 방법과 효율적인 NFS 서비스 관리 방법들에 대해 알아보도록 하겠다.

2.7.1 NFS 서버 동작 원리

NFS 서버와 클라이언트간의 통신에는 RPC(Remote Procedure Call)라는 기법을 이용한다. RPC 는 표준화된 규약이기 때문에 이를 이용하여 클라이언트에서 서버측의 함수들을 로컬시스템 함수 처럼 호출할수도 있고 서버의 사용되어 지는 함수들을 검색할 수도 있다.

NFS 서버는 이런 RPC를 이용하여 자신의 존재와 상태를 클라이언트에게 알려주게 된다. 이때 사용되는 프로그램이 Portmapper 이다. 이 portmapper 가 RPC에 등록되어져서 portmapper 를 통해 서버와 클라이언트간의 상태를 공유하고 서비스를 제어하게 된다.

구체적으로 클라이언트에서 NFS서버의 자원을 Mount 하는 과정에 대해 알아보도록 하자.

1. 클라이언트의 Portmapper 에서 서버의 Portmapper 통해 현재 서버의 수행중인 NFS Mount 서비스의 Port 를 알아낸다.
2. 클라이언트에서 서버의 마운트 서비스에 접속하여 원하는 파티션에 마운트를 요청한다.
3. 서버의 /etc/export 에 설정된 권한에 맞는 요청일 경우 이에 허가를 내려 준다.
4. 클라이언트는 Portmapper 를 통해 다시 NFS 서비스 Port 를 알아낸다. (2049/tcp)

5. 해당 Port 로 접속하여 nfs mount 를 수행한다.

2.7.2 NFS 관련 커널 기능

NFS 는 리눅스 커널 2.2 대 부터 커널에 포함되어져서 배포가 되어지고 있다. 커널에 포함이 됨으로 이전에 비해 보다 안정하고 빠른 성능 향상을 가져 오게 되었다.

그렇기 때문에 NFS 기능을 사용하기 위해서는 커널에 있는 NFS 기능을 활성화 시켜야 한다. Redhat 같은 리눅스 배포판의 기본 커널에는 대부분 NFS 기능이 활성화 되어져 있지만, 최신 커널로 업그레이드 할때는 반드시 이를 체크하여 주어야 한다.

리눅스 커널에서 NFS 관련 기능은 다음 위치에 있다.

Network File Systems	---	>	x
x x<M>	NFS file system support		x x
x x[*]	Provide NFSv3 client support		x x
x x<M>	NFS server support		x x
x x[*]	Provide NFSv3 server support		x x
x x[]	Provide NFS server over TCP support (EXPERIMENTAL)		x x

원하는 기능을 커널에 포함시키거나 모듈로 포함시키고 커널 컴파일을 하게 되면 된다.

2.7.3 NFS 서버 설정 하기

NFS 서버 설정은 매우 단순하다. /etc/exports 란 설정 파일에서 서버에 관련된 모든 설정이 완료된다.

간단한 설정 예제를 보도록 하자.

```
# vi /etc/exports
```

```
# NFS 원격 설치 시 사용되는 경로
/data/clunix/teragon *(ro,no_root_squash)
# 사용자 Data 보관에 이용되는 경로
/data/users *(rw,no_root_squash)
-----
```

exports 설정 형식은 다음과 같다.

```
/directory_path client_hosts(permissions) client_hosts(permissions)...
```

/directory_path 는 실제 NFS 상에서 export 될 Directory 경로이다.

client_hosts 는 NFS 서버에 접근할수 있는 클라이언트 대역을 정의하는 곳이다.

(permissions)은 클라이언트 호스트에 대한 권한을 지정하는 곳이다.

permissions에 사용되는 주요 옵션으로는 다음과 같은 것이 있다.

```
-----
ro : 파티션에 읽기 권한만 부여
rw : 파티션에 읽기 쓰기 권한 부여
noaccess : 특정 Directory의 액세스를 거부 시킴
no_root_squash : NFS 상에서 기본적으로는 root 권한에서는 NFS mount 를 수용하지 않는다.
하지만 이 옵션을 이용하면 클라이언트에서 root로도 마운트가 가능하다.
```

이제 NFS 서버 Daemon을 시작 시켜 보자

```
# /etc/rc.d/init.d/portmap restart
# /etc/rc.d/init.d/nfs restart
```

2.7.4 NFS 클라이언트 설정하기

클라이언트에서도 NFS 서버에 접근이 가능하도록 Portmapper Daemon을 실행 시키다.

```
# /etc/rc.d/init.d/portmap restart
```

NFS mount 를 시켜 보자

```
# mount -t nfs arhfw:/data/clunix/teragon /mnt/teragon
# mount -t nfs arhfw:/data/users /mnt/users
```

이와 같이 방법으로 NFS 서비스를 사용할 수 있다.

NFS 서비스는 NFS 기능에 관해서는 서버측 설정 보다는 NFS 클라이언트측 설정에 더 많은 의존성을 가지게 된다. 위 예제에서 처럼 NFS 클라이언트는 NFS Mount 를 통해 서버에 접근을 하게 되는데 NFS Mount 옵션에 따라 많은 기능 차이가 있다.

- NFS Mount 상세 옵션 설명

NFS Clients 에서의 nfs.mountd option으로 soft,hard,timeo,rsiz,wsiz... 등등 많은 option 이 있는데 이중에서 가장 최적화된 옵션을 찾아 되는 것은 남겨진 숙제이다.

주요 옵션 설명이다.

rsiz=n : NFS 서버로부터 읽어들이는 바이트 수를 지정 한다.
기본값은 커널에 따라다른데 현재로서는1024 바이트이다.

wsiz=n : NFS 서버에 쓰기를 할 때 사용하는 바이트수 를 정한다.
기본값은 커널에 따라다른데 현재 로서는 1024 바이트이다.

timeo=n : RPC 타임아웃이 생기고 나서 첫번째 재전송 요 구를 보낼 때 사용되는 시간> 으로서 1/10 초 단 위이다. 기본값은 7 * 1/10 초이다.

첫번째 타임아웃이 생기고 나서는 타임아웃 시간이 최대 치인 60 초에 이르거나 너무 많은 재전송 요구 가 별> 어 질 때 까지 타임아웃 시간이 2 배로 변화 한다.
만약 파일 시스템이 hard (hard >옵션을 참고) 마운트 되어있다면 각각의 타임아웃 시간은 2 배로 증가하고 재전>송 시도 가 일어날 때도 2 배로 증가한다.
최대 타임아웃 시간은 60 초이다.

네트워크 속도가 느리거나 서버 자체가 느리 다든지 여러 개의 라우터와 게이트웨이를 거칠 때는 마운트 당시 타임아웃 시간을 늘려 주는 것이 좋다.

retrans=n : 주 타임아웃을 발생시키는 부 타임아웃과 재전 송 횟수를 정한다.
기본값은 3 번의 타임아웃이다. 주 타임 아웃이 일어나면 화일 작업이 중지 되거나
콘솔 상 에 "서버가 반응하지 않음 "server not responding""이라는 메시지가
출력 된다.

retry=n : 백그라운드에서 진행 중인 NFS 마운트 작업 이 포기하기 전까지
실행할 횟수 를정한다. 기본값 은 10000 번이다.

namlen=n : NFS 서버가 RPC 마운트 프로토콜의 버전 2 를 지원하지 않을 때
원격 파일 시스템에서 지원 되는 파일 명의 길이를 명시한다.
POSIX pathconf 함 수 를 지원 하기 위해서 사용된다.
기본값은 255 개의 문자이다.

port=n : NFS 서버와 연결할 수 있는 포트 번호를 정 한다.
만약 0 이라면(기본값) 원격 호스트의 포트매퍼(portmapper) 에게 질의 하여
알아내도록 한다. 만약 포트매퍼에 NFS Daemon이 등록되어 있지 않은 경우에는
2049 라는 표준 NFS 포트 번호가 사용된다.

mountport=n : mountd 포트 번호 지정하기.

mounthost=name : mountd 를 실행 중인 호스트 명을 정한다.

mountprog=n : 원격 호스트의 마운트 Daemon과 접속하기 위해 사용할 수 있는
별도의 RPC 프로그램번호를 정 한다.
만약 여러 개의 NFS 서버를 운영하고 있을 때 사용한다.
기본값은 표준 RPC 마운트 Daemon 프로그램 번호인 100005 이다.

bg : 만약 첫 번째 NFS 마운트 시도가 타임아웃 걸리 면 백그라운드에서
실행을 계속 된다. 기본 값은 백그라운드로 실행하지 않고 그냥 포기한다.

fg : 첫번째 NFS 마운트 시도에서 타임아웃이 걸 리 면 그 즉시 포기 해 버린다.
기본값이다.

soft : NFS 화일 작업에서 주 타임아웃이 걸리면 프로 그램에게 I/O 에러를 보고한다.

기본값은 무한히 NFS 화일 작업을 재시도 하는 것이다

hard : NFS 화일 작업에서 주 타임아웃이 걸리면 콘솔 상에 "server not responding",

"서버가 반응 하지 않음" 이라고 출력하고 무한히 재시도한다. 이것이 기본값이다.

intr : 주 타임아웃이 생기고 하드 마운트 된 상태라면 화 일 작업을 중지하도록
시그널 을 보내도록 허용하고 EINTR 시그널을 보낸다.

기본값은 화일 작업을 인터럽트 하지 않는 것이다.

tcp : NFS 화 일 시스템을 기본값인 UDP 가 아니라 TCP 프로토콜을

사용하여 마운트 하도록 한다. 많은 NFS 서버들이 오로지 UDP 만을 지원한다.

udp : NFS 화일 시스템을 UDP 프로토콜로 마운트 한다. 기본값.

2.7.5 NFS Automount 설정하기

automount 에는 am-utils (amd), autofs 두 가지의 패키지가 있다.

Redhat9 에서는 기본적으로 autofs 를 채택하고 있다.

간단한 설정에 대해 알아보자.

주요 설정 파일은 ..

/etc/auto.master

/etc/auto.misc

이 있다.

vi /etc/auto.master

```
=====
/home          /etc/auto.user01  --timeout=5
```

```
# vi /etc/auto.user01
```

```
=====
```

```
user01/data          -fstype=nfs,rw,soft,bg      master:/data/nfs/user01
```

```
# /etc/rc.d/init.d/portmap restart
```

```
# /etc/rc.d/init.d/autofs restart
```

/home/user01/data 로 이동하면 자동으로 master 의 /data/nfs/user01
로 nfs mount 가 된다.

automount 를 사용하게 되면 실제 사용자가 지정된 mount point 에 접근 시에만
자동으로 nfs mount 하고 일정 시간 사용을 하지 않으면 자동으로 umount 를 시
키기 때문에 nfs 서비스로 인한 resource 를 최대한 절약 할 수 있고, 여러명이
지속적으로 Nfs 를 사용할 경우 1개의 곳에서 lock 이 걸려도 전체 서비스에 문
제가 생길 가능성이 있는데 automount 로 상당 부분 해소할 수 있다.

autofs-4.x 버전에서는 multiple hostname 기능이 지원한다.

이는 먼저 master 로 Connection 요청을 해서 정상적인 요청이 이루어 지지
않으면 slave 에 연결이 된다. Redhat9 에서는 기본적으로 autofs-3.x 임으로
autofs-4.x 로 업그레이드를 해야 한다.

<http://www.kernel.org/pub/linux/daemons/autofs/>