

Mengamankan NFS Server

NFS Server telah digunakan secara luas sebagai file server. Pada kali ini kita akan membahas bagaimana mengamankan NFS Server.

Menjamin akses read-only untuk direktori yang di eksport

Untuk menjamin akses read-only ke direktori atau filesystem yang di eksport dari NFS server ke client, kita bisa gunakan akses *ro* seperti pada contoh berikut :

```
/apps devpc.nitec.com(ro)
```

Client *devpc.nitec.com* hanya memiliki akses read-only ke direktori */apps*. Disarankan menggunakan opsi *ro* bila mengeksport direktori yang menyimpan aplikasi binari.

Mencegah akses ke direktori tertentu

Ketika kita mengeksport seluruh filesystem direktori maka secara otomatis subdirektori nya akan dieksport dengan opsi akses yang sama. Dan kadang hal ini tidak di inginkan terjadi, anda mungkin ingin mengeskport direktori */pub* namun tidak direktori */pub/staff-only*. Pada kasus ini kita bisa gunakan opsi akses *noaccess* seperti contoh di bawah ini :

```
/pub weblab-??nitec.com(ro)  
/pub/staff-onlyweblab-??nitec.com(noaccess)
```

Seluruh host *weblab-??nitec.com* (dimana ?? adalah dua karakter bebas) yang memiliki akses read-only ke direktori */pub*, namun host tersebut tidak dapat mengakses direktori */pub/staff-only*.

Pemetaan pemakai antara NFS Server dan Client

Setelah kita meng'setup NFS server, satu hal yang [erlu diperhatikan ialah bagaimana menjaga pemakai dari pemetaan antara NFS server dan client. Sebagai contoh, misalkan anda mengeksport direktori */www* yang dimiliki oleh user *webguru* dan group *webdev*. NFS client tentunya harus memiliki hak sebagai user *webguru* dan group *webdev* untuk mengakses direktori tersebut. Dan jangan pernah memberi NFS client hak root account pada direktori NFS. Ini lah sebabnya NFS server secara

default menolak hak tersebut dengan opsi *root_squash*. Ini akan memetakan root user (UID = 0) dan group root (GID = 0) ke user nobody pada client. Anda juga dapat meng'disable opsi tersebut dengan menambahkan opsi *no_root_squash*.

Untuk memetakan root UID/GID ke UID/GID tertentu, kita dapat menggunakan akses opsi *anonuid* dan *anongid* seperti terlihat di contoh berikut :

```
/proj *nitec.com (anonuid=500 anongid=666)
```

Di contoh berikut *anonuid* dan *anongid* akan mempunyai UID 500 dan GID 666. Jika anda ingin mendaftar dari UID dan GID yang akan digunakan untuk anonymous UID/GID, anda dapat menggunakan opsi *squash_uids* dan *squash_gids* seperti contoh berikut :

```
/proj *nitec.com (anonuid=500 anongid=666 squash_uids=0-100 squash_gids=0-100)
```

Seluruh UID dan GID direktori dengan range 0-100 akan dipetakan menjadi UID 500 dan GID 666 anonymous.

Eksternal file map dapat digunakan untuk menentukan UID/GID NFS client, kali ini kita gunakan opsi *map_static* seperti contoh berikut :

```
/proj *nitec.com (map_static=/etc/nfs.map)
```

Direktori */proj* di eksport ke seluruh host nitec.com, namun semua UID/GID NFS client di petakan menggunakan file */etc/nfs.map*. Contoh isi *nfs.map* :

```
uid 0-100 - # squash seluruh remote uid pada range 0-100
gid 0-100 - # squash seluruh remote gid pada range 0-100
uid 500 666 # map remove uid 500 ke uid lokal 666
gid 500 777 # map remove gid 500 ke gid lokal 777
```

Kini anda telah banyak mengetahui opsi-opsiyang sering digunakan pada file */etc/export*. Agar daemon NFS mengetahui telah terjadi perubahan pada file */etc/export*, sebuah script *exportfs* harus di restart. Misalnya :

```
/usr/bin/exportfs
```

Sekarang, untuk meyakinkan *rpc.mountd* dan *rpc.nfsd* telah berjalan dengan baik, jalankan program *rpcinfo* seperti dibawah ini :

```
rpcinfo -p
```

Keluarannya akan terlihat seperti ini :

<i>program</i>	<i>vers</i>	<i>proto</i>	<i>port</i>	
100000	2	tcp	111	portmapper
100000	2	udp	111	portmapper
100021	1	udp	1024	nlockmgr
100021	3	udp	1024	nlockmgr
100024	1	udp	1025	status
100024	1	tcp	1024	status
100011	1	udp	728	rquotad
100011	2	udp	728	rquotad
100005	1	udp	1026	mountd
100005	1	tcp	1025	mountd
100005	2	udp	1026	mountd
100005	2	tcp	1025	mountd
100003	2	udp	2049	nfs

Terlihat *mountd* dan *nfsd* terlihat telah berjalan dan bekerja dengan baik.

Referensi : RedHat Linux Security and Optimization - O'reilly
Semoga Bermanfaat.

Faiz