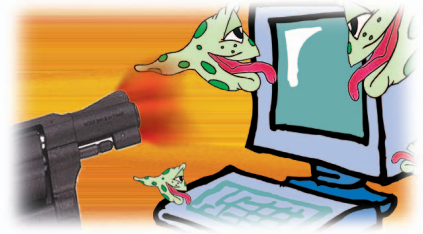


# Bug Kernel, Siapa Takut?

Linux dikembangkan oleh programmer. Dalam waktu singkat, Linux berkembang begitu pesat. Dalam perkembangan tersebut, ada kalanya terjadi bug. Sebagai pengguna, kita perlu menanggapinya dengan bijaksana.



**B**ug lagi, tidak hampir tidak ada habisnya. Salah satu bug yang berbahaya bagi pengguna Linux adalah bug kernel. Begitu berbahayanya sampai bug kernel masih merupakan topik yang cukup hangat untuk dibicarakan. Privasi perusahaan terancam hanya karena beberapa bug pada kernel yang dapat membuat kita menyesal, belum lagi data perusahaan kita yang sangat bergantung kepada sistem yang ada.

Bagi pengguna Linux, Linux memang sudah termasuk salah satu sistem operasi yang cukup stabil, tapi terkadang setiap *software* maupun sistem operasi pasti mempunyai lubang *security*. Sebelum lubang *security* itu menjadi masalah yang lebih besar, mungkin terlebih dahulu kita mengantisipasi dengan cara yang ada. Grsecurity merupakan salah satu alternatif *security* yang ada dari sekian banyak yang melakukan pengamanan terhadap sistem kita dari segi lokal. Sering bug yang ada pada kernel dapat di-*exploit* begitu saja dengan kode-kode program yang beredar bebas di Internet. Sebagai admin, Anda pasti was was. Daripada stress mending kita mencoba salah satu alternatif *security* yang ada yang dapat membantu kita untuk menghilangkan salah satu beban yang ada.

Grsecurity ini sudah cukup lama ada, Debian dan Gentoo merupakan distro yang didukung penuh dengan *access control list* yang sudah lengkap dan disesuaikan dengan distro Anda. Jika distro Anda mempunyai konfigurasi yang berbeda, Anda dapat menyesuakannya. Sistem admin dapat membuat sistem yang ada cukup *secure* dan superuser atau root harus dibuat "*mandul*". Untuk lebih jelasnya, marilah kita memulai percobaan kita. Anda yang sudah mempunyai konfigurasi kernel sebelumnya dapat memakai konfigurasi yang ada. Kernel yang dipakai di sini adalah kernel default yang

diberikan dari Gentoo, yaitu kernel Gentoo-sources dengan versi 2.4.22-r5. Caranya:

1. Download kernel 2.4.22 dari <http://www.kernel.org>.
2. Download patch untuk kernel dan gradm di situs Gentoo:
  - a. <http://www.zentek-international.com/mirrors/gentoo/distfiles/gradm-1.9.13.tar.gz>,
  - b. <http://gentoo.oregonstate.edu/distfiles/gradm-1.9.13.tar.gz>,
  - c. <http://dev.gentoo.org/~iggy/gentoo-sources-2.4.22-r5.patch.bz2>,
  - d. dan situs mirror yang lain.
3. Rule yang sudah dibuat oleh developer Gentoo
  - a. <http://www.zentek-international.com/mirrors/gentoo/distfiles/grsecurity-base-policy-20030614.tar.gz>,
  - b. <http://gentoo.oregonstate.edu/distfiles/grsecurity-base-policy-20030614.tar.gz>,
  - c. dan situs-situs mirror yang lain.

Alternatif lain yang lebih efisien yang sangat disarankan yaitu **emerge -f gentoo-sources gradm genkernel**. **emerge -f** ini maksudnya emerge with *fetch only* di mana package yang akan kita download hanya diambil dulu, sangat efisien tetapi pastikan komputer yang anda pakai sudah mempunyai koneksi Internet. Jangan sampai lupa. Dan mesti diingat juga jika Anda memilih langkah pertama Gentoo, maka langkah selanjutnya akan sama seperti langkah yang bukan menggunakan distro Gentoo, tetapi jika Anda memilih langkah ini maka Anda hanya mengikuti langkah yang khusus buat Gentoo.

Jika Anda bukan pengguna Gentoo.

1. Download kernel dengan versi apapun di <http://www.kernel.org>.
2. Download patch dari <http://grsecurity.net> dan sesuaikan dengan versi kernel Anda.
3. Dan terakhir download gradm dari <http://grsecurity.net>.

4. Untuk rule dari grsecurity sudah membuat *access control list* yang khusus buat pengguna Debian, jadi jika Anda pengguna Debian, Anda bisa berbesar hati.

Sebelum kita lanjutkan periksa dulu semua file apakah sudah lengkap, jangan lupa pastikan gradm Anda merupakan versi terakhir, versi terakhir gradm adalah 1.9.13.

Hmm, ternyata menyenangkan *download* banyak file, setelah itu dilanjutkan ke tahap di mana persiapan kompilasi kernel sudah mulai akan dilakukan.

Untuk pengguna Gentoo Anda hanya perlu mengetikkan beberapa sintak dan menunggu sebentar karena proses kompilasi membutuhkan waktu.

1. **SmallVille / # emerge -b genkernel**  
membuat paket untuk genkernel yang akan bertugas melakukan kompilasi kernel Anda secara otomatis.
2. **SmallVille / # emerge -b gentoo-sources**  
membuat paket kernel Anda dan instal.
3. **SmallVille / # emerge -b gradm**  
membuat paket gradm dan instal paket gradm.
4. **SmallVille / # ln -s /usr/src/linux-2.4.22-gentoo-r5 /usr/src/linux**  
pastikan /usr/src/linux sudah di-link ke sources Gentoo yang baru.

Pasti Anda bertanya-tanya apa lagi itu **emerge -b**, ini adalah langkah emerge adalah langkah untuk membuat GRP dari paket-paket dari program source yang sedang dikompilasi oleh komputer kita dan jika Anda adalah seorang administrator yang menangani banyak komputer yang sama. Maka ini merupakan pilihan yang sangat bijaksana,

yaitu membuat GRP dari program yang diinstal.

Jika Anda bukan pengguna gentoo langkah-langkah yang perlu dilakukan.

1. **SmallVille / # tar xjvf /home/dokaya/source/linux-2.4.24.tar.bz2 -C /usr/src** untuk melakukan ekstraksi source kernel kita ke directory /usr/src.
2. **SmallVille / # ln -s /usr/src/linux-2.4.24 /usr/src/linux**  
buat softlink kenapa ini wajib? Karena ada beberapa program yang depend ke kernel kita dan defaultnya mengacu ke /usr/src/linux.
3. Patch kernel Anda sesuai dengan kernel Anda, misalnya:
  - a. **SmallVille / # cd /usr/src/linux**
  - b. **SmallVille linux # patch -p1 < /home/dokaya/patch-kernel/grsec/grsecurity-1.9.13-2.4.24.patch** untuk melakukan patch terhadap kernel kita.
4. Ekstrak **gradm** ke /usr/src/, kemudian sesudah Anda mengekstrak semua paket yang ada dan gradm maka:
  - a. **SmallVille linux # tar xzvf /home/dokaya/source/gradm-1.9.13.tar.bz2 -C /usr/src** untuk melakukan ekstraksi program gradm.
  - b. **SmallVille linux # cd /usr/src/gradm** untuk masuk ke folder gradm.
  - c. **SmallVille gradm # make** lakukan kompilasi terhadap gradm.
  - d. **SmallVille gradm # make install** untuk melakukan instalasi gradm.

Ganti versi kernel yang ada supaya tidak tumpang tindih.

```
SmallVille / # vi /usr/src/linux/Makefile
VERSION = 2
PATCHLEVEL = 4
SUBLEVEL = 22
EXTRAVERSION = -gentoo-r5 misalnya
ganti ke EXTRAVERSION = -dokaya-r1
Untuk yang versi 2.4.24
VERSION = 2
PATCHLEVEL = 4
```

**SUBLEVEL = 24**

**EXTRAVERSION = -grsec misalnya ganti ke EXTRAVERSION = -dokaya-r1**

Ingat untuk version, patch level dan sublevel jangan diganti karena beberapa *source code* yang ada didalam kernel bergantung kepada versi kernel yang boleh diganti hanya Extraversion yang ada di dalam file **Makefile**.

Jika tahapan di atas sudah terselesaikan, Anda sudah dapat melanjutkan ke tahapan berikutnya yaitu kompilasi kernel, tapi sebelumnya copy dulu semua konfigurasi yang ada di /proc/config ke /usr/src/linux/.config (khusus buat yang bukan merupakan pengguna gentoo).

```
SmallVille / # cat /proc/config > /usr/src/linux/.config
```

Jika Anda pengguna gentoo, di mana sebelumnya konfigurasi Anda sudah dapat dipakai maka back-up dulu konfigurasi genkernel default yang ada dan copy konfigurasi kernel Anda yang sebelumnya ke dalam konfigurasi default

```
SmallVille / # cp /usr/share/genkernel/x86/kernel-config-2.4 /usr/share/genkernel/x86/kernel-config-2.4-backup.
```

```
SmallVille / # cat /proc/config > /usr/share/genkernel/x86/kernel-config-2.4.
```

Konfigurasi yang ada sudah dilakukan apalagi yang selanjutnya harus kita lakukan. Yang selanjutnya harus kita lakukan adalah kompilasi kernel, akhirnya.

Khusus pengguna Gentoo anda perlu melakukan kompilasi kernel dengan cara:

```
SmallVille / # genkernel --menuconfig all
```

**Catatan:** pastikan versi genkernel Anda minimal versi 3 karena ada perbedaan yang cukup jauh dengan versi genkernel yang lama.

Jika Anda bukan pengguna gentoo atau memilih cara tanpa genkernel, maka:

```
SmallVille / # cd /usr/src/linux
```

```
SmallVille linux # make oldconfig
```

```
SmallVille linux # make menuconfig &&
make bzImage modules modules_install
```

Setelah langkah di atas Anda lakukan pastilah Anda akan diberikan tampilan

menu dan submenu dari kompilasi kernel yang ada. Anggap konfigurasi kernel yang sebelum seharusnya sudah jalan jadi yang kita permasalahan hanya konfigurasi grsecurity yang ada.

Pertama masuk kedalam sub menu dari grsecurity. **Grsecurity** → **Grsecurity** aktifkan ini dulu baru bagian yang lain akan keluar. Setelah itu akan muncul bagian baru.

Ada 4 pilihan yang dapat Anda pilih:

1. Low Security Level
2. Medium Security Level
3. High Security Level
4. Customized Security Level

Low Security Level mempunyai konfigurasi default, yaitu:

- a. Linking restrictions
- b. Fifo restrictions
- c. Random pids
- d. Enforcing nproc on execve()
- e. Restricted dmesg
- f. Random ip ids
- g. Enforced chdir("/") on chroot

Medium Security Level mempunyai konfigurasi Low security Level dan:

- a. Random tcp source ports
- b. Altered ping ids
- c. Failed fork logging
- d. Signal logging
- e. Deny mounts in chroot
- f. Deny double chrooting
- g. Deny sysctl writes in chroot
- h. Deny mknod in chroot
- i. Deny access to abstract AF\_UNIX sockets out of chroot
- j. Deny pivot\_root in chroot
- k. Denied writes of /dev/kmem, /dev/mem, and /dev/port
- l. /proc restrictions with spesial gid set to 10 ( usually wheel )
- m. Address space layout randomization

High Security Level mempunyai konfigurasi dari Low dan Medium security level dan:

- a. Additional /proc restrictions
- b. Chmod restrictions in chroot
- c. No signals, ptrace, or viewing processes outside of chroot
- d. Capability restriction in chroot
- e. Deny fchdir out of chroot
- f. Priority restrictions in chroot

- g. Segmentation-based implementation of PaX
- h. Mprotect restrictions
- i. Removal of `/proc/<pid>/[maps|mem]`
- j. Kernel stack randomization
- k. Mount/unmount/remount logging
- l. Kernel symbol hiding

Sebelum Anda menggunakan semua jenis level yang ada terlebih dahulu baca dokumentasi yang ada dan pahami dulu maksud yang ada. Konfigurasi untuk low security level disarankan untuk Anda yang mempunyai banyak program yang kemungkinan bisa bentrok dengan kernel baru ini, konfigurasi kernel dengan medium security level dan disarankan pastikan service `identd` dijalankan sebagai group dari `wheel`. Sedangkan high security level sangat defensif sekali dan risiko yang ada beberapa program yang ada akan tidak dapat dijalankan, untuk itu Anda memerlukan **chpax** untuk program yang bermasalah.

Dan yang kita pilih sekarang High Security Level, Anda dapat mengantinya sesuai dengan keinginan. Sesuaikan dengan kebutuhan.

Setelah itu save konfigurasi kernel Anda dan keluar dari menu, silakan untuk menunggu proses kompilasi kernel Anda.

Proses compile kernel Anda sudah selesai? Kalau sudah Anda sudah boleh memulai *booting* dengan menggunakan kernel baru Anda. Ingat untuk memasukan kernel Anda ke dalam grub / lilo Anda, jangan lupa jika anda menggunakan lilo sebagai boot manager jangan lupa untuk menjalankan kembali lilo Anda. Jika Anda menggunakan grub sebagai boot manager, maka Anda hanya perlu mengganti konfigurasi `/boot/grub/grub.conf` dan merestart komputer Anda dan jalan kernel baru Anda, dan ingat jangan lupa untuk menyimpan kernel yang lama, sebagai cadangan kalau ada masalah baru.

Anda sudah *restart*? *Hmm*, ternyata kernel baru kita tidak ada masalah, jika Anda menemukan masalah, periksa kembali konfigurasi kernel Anda. Sekarang saatnya kita masuk ke bagian dari *Access Control List* biasanya file konfigurasi di simpan di `/etc/grsec/`.

Apa itu *Access Control List System*?

Access Control List adalah sekumpulan peraturan yang membatasi program-program dan user di dalam sebuah sistem.

Kenapa kita menggunakan Access Control List System? Karena kita ingin membatasi penggunaan files, *resources*, *capability*, dan *sockets* oleh semua pengguna termasuk super user/root. Patch *grsecurity* yang sudah kita implementasi kedalam sistem sudah membatasi user lokal untuk melakukan penyerangan dari dalam untuk mendapatkan account super user, ditambah dengan kemampuan Access Control List maka sebuah sistem sudah komplit.

Untuk lebih jelasnya akan dijelaskan penggunaan Access Control List secara lebih mendetail.

Struktur dari `acl` adalah seperti berikut:

```
<path of subject process> <optional
subject modes> {
  <file object> <optional object modes>
  [+][-] <capability>
  <resource name> <soft limit>
  <hard limit>
  connect {
    <ip>/<netmask>:<low port>-
    <high port> <type> <proto>
  }
  bind {
    <ip>/<netmask>:<low port>-
    <high port> <type> <proto>
  }
}
include </etc/grsec/folder-acl-lain>
```

Aturan-aturan yang ada:

- Semua file harus ditulis lengkap dengan path, cth `/usr/sbin/sshd` bukan `sshd`.
- Bisa menggunakan `include` seperti yang ditulis di atas atau `include` langsung mengacu ke nama-file cth:

```
/etc/grsec/acls/xfree
```

➔ untuk mengacu langsung ke file.

```
/etc/grsec/acls
```

➔ untuk mengacu langsung ke semua yang ada di dalam directory.

- Ingat mesti mempunyai `acl` default untuk path `/`, akan ada peringatan dari `gradm` jika tidak dimasukkan.

Konfigurasi **subject modes** yang ada:

- **h**, proses hidden dan hanya dapat dilihat

oleh proses yang mempunyai subject modes `v`.

- **v**, proses ini dapat melihat proses yang hidden.
- **p**, proses ini diproteksi dan hanya dapat di-kill oleh proses yang mempunyai subject modes `k`.
- **k**, proses ini dapat meng-kill proses yang diproteksi.
- **l**, mengaktifkan learning mode.
- **d**, melakukan proteksi terhadap `/proc/<pid>/fd` dan `/proc/<pid>/mem`.
- **b**, enable process accounting.
- **P**, disables `PAGEEXEC` di subject mode ini.
- **S**, disables `SEGMEXEC` di subject mode ini.
- **M**, disables `MPROTECT` di subject mode ini.
- **R**, disables `RANDMMAP` di subject mode ini.
- **G**, enables `EMUTRAMP` di subject mode ini.
- **X**, enables `RANDEXEC` di subject mode ini.
- **O**, menolak penambahan batasan `mmap()` dan `ptrace()` terhadap subject mode ini.
- **A**, memproteksi shared memory dari subject mode ini.
- **K**, ketika menerima pesan alert, proses ini langsung autokill.
- **C**, ketika menerima pesan alert, proses ini langsung autokill semua proses yang dimiliki penyerang dan proses ini.
- **T**, memastikan proses ini tidak menjalankan trojan.

Konfigurasi **object modes** yang ada:

- **r**, object ini bisa dibaca.
- **w**, object ini bisa ditulis dan ditambahkan.
- **x**, object ini dapat dijalankan.
- **a**, object ini dapat ditambahkan.
- **h**, object ini di-hide.
- **t**, object ini dapat di-*ptrace*, tetapi tidak dapat mengganti tugas yang jalan/read-only *ptrace*.
- **s**, log akan lebih ditekankan jika ada penolakan akses terhadap object ini.
- **i**, hanya mempengaruhi binary.
- **R**, audit successful reads untuk object ini.
- **W**, audit successful writes untuk object ini.

- **X**, audit successful execs untuk object ini.
- **A**, audit successful appends untuk object ini.
- **F**, audit successful finds untuk object ini.
- **I**, audit successful ACL inherits untuk object ini.

## Inheritance/penurunan sifat ACL

Cth konfigurasi dari acl Anda adalah seperti ini.

```
/ {
  /etc/shadow h
  /etc/shadow- h
  /etc/passwd h
  /etc/grsec h
  /sbin rx
  /home rwx
  /tmp rwx
}
/usr/sbin/sshd {
  /etc/shadow r
  /etc/passwd r
}
```

Konfigurasi yang ada setelah adanya sifat penurunan ACL menjadi:

```
/ {
  /etc/shadow h
  /etc/shadow- h
  /etc/passwd h
  /etc/grsec h
  /sbin rx
  /home rwx
  /tmp rwx
}
/usr/sbin/sshd {
  /etc/shadow r
  /etc/shadow- h
  /etc/passwd r
  /etc/grsec h
  /sbin rx
  /home rwx
  /tmp rwx
}
```

Sekarang yang ada adalah `/usr/sbin/sshd` mempunyai semua sifat dari `/` dan meng-*override* aturan yang ada dari `/` yaitu `/etc/shadow` yang sebelumnya `h` menjadi `r` dan `/etc/passwd` yang sebelum `h` menjadi `r`.

Algoritma yang diimplementasi oleh gradm ini tidak meng-*inheritance* hanya dari

induk yang paling luar tapi melalui semua induk yang ada, misalnya:

- `/usr/bin/sshd`, akan mempunyai sifat yang sama dengan `/`, `/usr`, `/usr/bin` ( jika ada ACLnya ).
- `/bin/mount`, akan mempunyai sifat yang sama dengan `/`, `/bin` ( jika ada ACLnya ).

Contoh konfigurasi default untuk `/etc/grsec/acl`:

```
/ {
  / r
  /opt rx
  /home rwx
  /mnt rw
  /dev
  /dev/urandom r
  /dev/randomr
  /dev/zerorw
  /dev/input rw
  /dev/psauxrw
  /dev/null rw
  /dev/tty? rw
  /dev/console rw
  /dev/tty rw
  /dev/tty? rw
  /dev/pts rw
  /dev/ptmx rw
  /dev/dsp rw
  /dev/mixer rw
  /dev/fd0 r
  /dev/cdrom r
  /dev/mem h
  /dev/kmemh
  /dev/port h
  /bin rx
  /sbin rx
  /lib rx
  /usr rx
  /etc rx
  /etc/ssh h
  /proc rwx
  /proc/kcore h
  /proc/sysr
  /root r
  /tmp rw
  /var rwx
  /var/tmp rw
  /var/log r
  /boot h
  /etc/grsec h
  -CAP_SYS_TTY_CONFIG
  -CAP_LINUX_IMMUTABLE
```

# IKLAN

```

-CAP_NET_RAW
-CAP_MKNOD
-CAP_SYS_ADMIN
-CAP_SYS_RAWIO
-CAP_SYS_MODULE
-CAP_SYS_PTRACE
-CAP_NET_ADMIN
-CAP_NET_BIND_SERVICE
-CAP_SYS_CHROOT
}

/sbin/init {
/dev/initctl rw
}

/sbin/syslogd {
/dev/log rw
/var/log w
}

/sbin/klogd {
/dev/log rw
}

/usr/sbin/cron {
/dev/log rw
}

/usr/sbin/crond {
/dev/log rw
}

/usr/sbin/xinetd p {
/dev/log rw
}

/usr/sbin/inetd p {
/dev/log rw
}

/usr/sbin/anacron {
/dev/log rw
}

/bin/login {
/dev/log rw
}

/usr/sbin/sshd dp {
/etc/ssh r
/dev/log rw
+ CAP_SYS_TTY_CONFIG
+ CAP_SYS_CHROOT
}

```

```

/usr/sbin/tcpd {
/dev/log rw
}

```

ACL di atas hanya sample yang diberikan default dari gradm, untuk pengguna debian, sudah ada beberapa konfigurasi yang ada yang diberikan untuk debian anda tinggal menambahkan ACL di konfigurasi default ACL :

```
include </etc/grsec/debian_secure_acls>
```

Untuk pengguna gentoo Anda dapat men-download konfigurasi yang ada dari gentoo atau Anda dapat membuat sendiri

tergantung dari minat Anda.

Terakhir implementasi atau tidaknya ACL tergantung kebutuhan. Untuk sistem yang tidak terlalu strict mungkin ACL ini tidak diperlukan, tapi jika data dan informasi yang ada di server Anda sangat dijaga rahasianya, maka Anda memerlukan ini. Konfigurasi default dari grsec dengan high security sudah sangat secure, tanpa acl saja sudah sangat membantu sekali. Tetapi ACL itu merupakan sebuah kebutuhan tambahan dimana data Anda sangat dijaga kerahasiaannya.

Sampai di sini saja semoga artikel ini bermanfaat bagi Anda semua.🙏

Dody Wijaya (lovedokaya@yahoo.com)

## Tips

- Jika Anda cukup iseng, maka saya dengan sangat menyarankan Anda untuk mengubah konfigurasi dan *coding* yang ada dari gradm. Dikarenakan cukup banyak orang yang sudah memahami cara kerja ACL ini, maka Anda dapat mengubah konfigurasi ACL dan penyimpanan *password* Anda */etc/grsec* menjadi sesuai kebutuhan Anda, meskipun sudah di-hide, karena kita hanya menjaga-jaga alternatif kemungkinan yang ada. Contoh: */etc/superstar* (ingat ini hanya contoh Anda bisa mengubah sesuai keinginan Anda). Perubahan ini dapat dilakukan dengan mengubah **Makefile** dari gradm, dan semua nama file gradm rubah menjadi superstar, dan jangan lupa manual dari gradm juga Anda ubah sesuai dengan keinginan. *Hmm*, sangat merepotkan, tapi Anda akan merasa lega jika berhasil mengimplementasikan.
- Ubah coding dan ubah nama binary dari **gradm** menjadi nama yang lain. Contoh, **superstard**.
- Ingat masukkan konfigurasi Anda ke dalam script booting Anda, tapi sebelum itu pastikan semua proses sudah berhasil dijalankan, baru Anda menjalankan program **gradm -E / superstard - E** (jika Anda sudah mengubahnya).
- Setelah Anda mengubah nama **gradm** menjadi **superstard**, dan */etc/grsec* menjadi */etc/superstar*, jangan lupa konfigurasi default acl Anda yang semula */etc/grsec h*, diubah menjadi */etc/superstar h*.
- Usahakan Anda bisa melakukan perubahan ACL sesuai dengan kebutuhan atau memakai modus *learning mode*.
- Ikut *mailing list* yang ada dari grsecurity dan ikut forum yang ada yaitu:  
→ [sympa@grsecurity.net](mailto:sympa@grsecurity.net) dengan body message "subscribe grsecurity" tanpa tanda kutip.  
→ <http://forum.grsecurity.net> ini untuk forum.
- Rajin-rajin untuk mengikuti informasi security yang ada (<http://www.securityfocus.com/archive/1>).
- Pastikan password untuk administrasi gradm tidak sama dengan password root anda, biasa disamakan saya sarankan untuk membedakannya.
- Jangan sering-sering men-*disable* gradm jika Anda sudah mengimplemen-tasikannya. Jika perlu administrasi, Anda hanya perlu masuk seperti user biasa, kemudian **su** untuk menjadi root, kemudian jalankan perintah **superstard -a**, ini hanya akan memberikan Anda hak user root di console yang sedang dijalankan. Tapi di tempat lain ACL untuk gradm masih berlaku.
- Jika terjadi perubahan ACL, lakukan perubahan dengan **superstard -R** untuk menjalankan ulang ACL Anda.