

RED HAT :: SAN DIEGO :: 2007

SUMMIT



Creating SELinux Policy Modules

Dan Walsh

Wednesday May 9, 2007

SELinux Policy Modules

RHEL5 Introduces the concept of SELinux Policy Modules

- Allows users to easily customize policy
- Allows third parties to ship policy with their rpms
- Similar to kernel Modules
 - In RHEL4 you have to install entire policy source tree
 - Recompile and reload
 - In RHEL5 you can build a policy module without source tree



Requirements

- RPMS required to build SELinux Policy Modules
 - selinux-policy-devel
 - Existing policy “interface files”
 - /usr/share/selinux/devel/*
 - Replaces selinux-policy-TYPE-sources in RHEL5.
 - Policy sources still available in srpm.
 - checkpolicy
 - checkmodule (Policy module compiler)
 - policycoreutils
 - semodule, audit2allow, sepolgen
 - system-config-selinux
 - polgengui



Requirements

```
cat /etc/yum.repos.d/selinux.repo
```

```
[selinux]
```

```
name=selinux
```

```
baseurl=http://people.redhat.com/dwalsh/SELinux/Fedora
```

```
#baseurl=http://people.redhat.com/dwalsh/RHEL5/u1
```

```
enabled=1
```

```
gpgcheck=0
```



Policy Modules

- Three Components
 - Type Enforcement (TE) File
 - Contains all the rules used to confine your application
 - File Context (FC) File
 - Contains the regular expression mappings for on disk file contexts
 - Interface (if) Files
 - Contains the interfaces defined for other confined applications, to interact with your confined application
- Policy Package (pp)
 - Compiler/packager roles generates policy package to be installed on systems.



Building Policy Packages

- audit2allow
 - Examines /var/log/audit/audit.log and /var/log/messages for AVC messages
 - Searches Interface files for correct interface
 - If no interface found generates allow rules



Audit2allow

time->Thu Apr 12 05:12:01 2007

type=PATH msg=audit(1176369121.794:1514): item=0 name="/usr/games/vultureseye/vultureseye.#prelink#.m8SXxq"

inode=11960540 dev=fd:00 mode=040755 ouid=0 ogid=0 rdev=00:00 obj=system_u:object_r:usr_t:s0

type=CWD msg=audit(1176369121.794:1514): cwd="/"

type=SYSCALL msg=audit(1176369121.794:1514): arch=40000003 syscall=5 success=no exit=-13 a0=bfe58b40 a1=80c2 a2=180

a3=180 items=1 ppid=1443 pid=1452 auid=0 uid=0 gid=0 euid=0 suid=0 fsuid=0 egid=0 sgid=0 fsgid=0 tty=(none) comm="prelink"

exe="/usr/sbin/prelink" subj=user_u:system_r:prelink_t:s0 key=(null)

type=AVC msg=audit(1176369121.794:1514): avc: denied { add_name } for pid=1452 comm="prelink"

name="vultureseye.#prelink#.m8SXxq" scontext=user_u:system_r:prelink_t:s0 tcontext=system_u:object_r:usr_t:s0 tclass=dir

```
# audit2allow -i /var/log/audit/audit.log
```

```
allow prelink_t usr_t:dir add_name;
```

```
# audit2allow -R -i /var/log/audit/audit.log
```

```
require {
```

```
    type prelink_t;
```

```
}
```

```
files_rw_usr_dirs(prelink_t)
```



audit2allow

```
# audit2allow -M myprelink -R -i /var/log/audit/audit.log
```

```
***** IMPORTANT *****
```

To make this policy package active, execute:

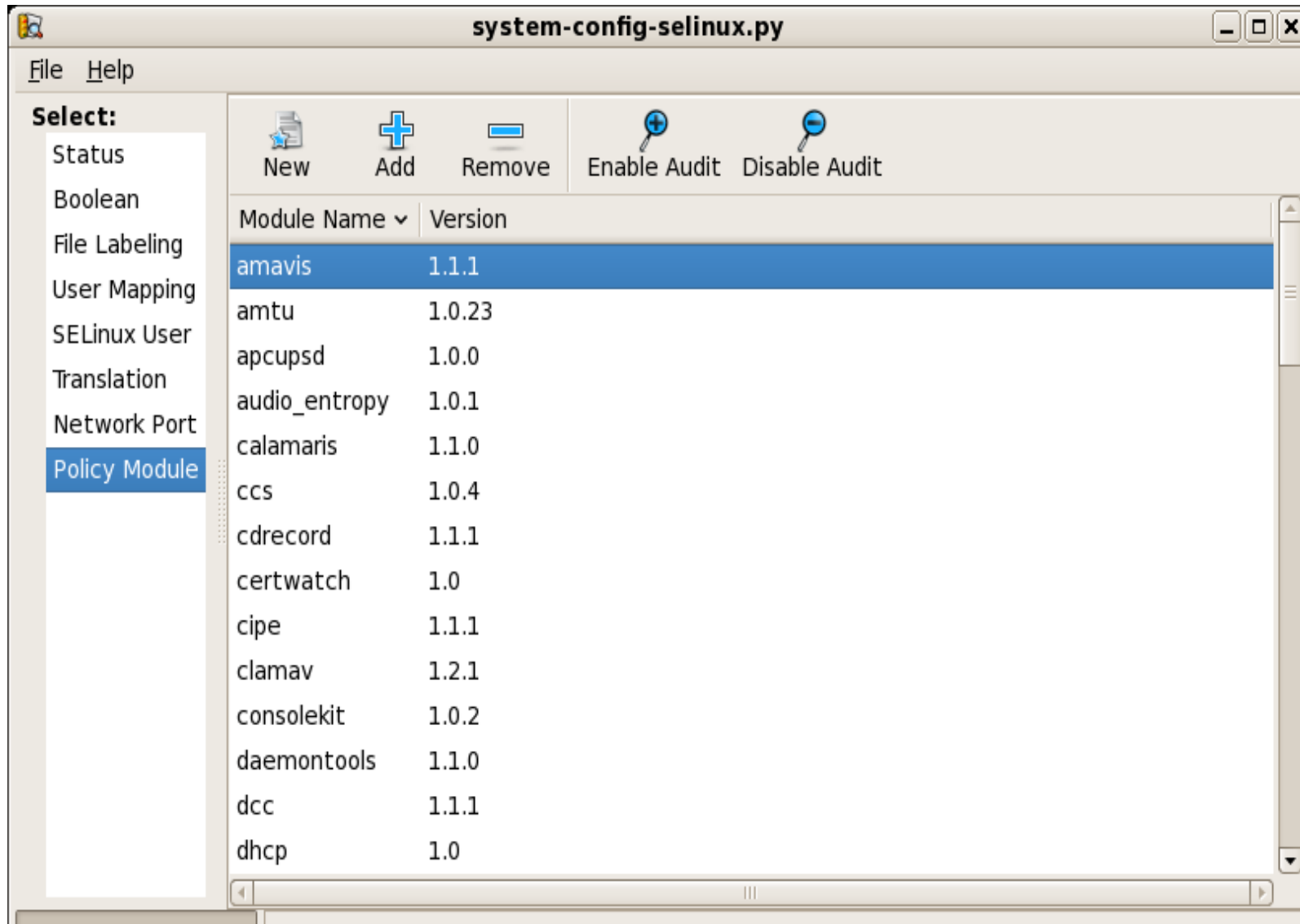
```
semodule -i myprelink.pp
```

```
# ls myprelink*
```

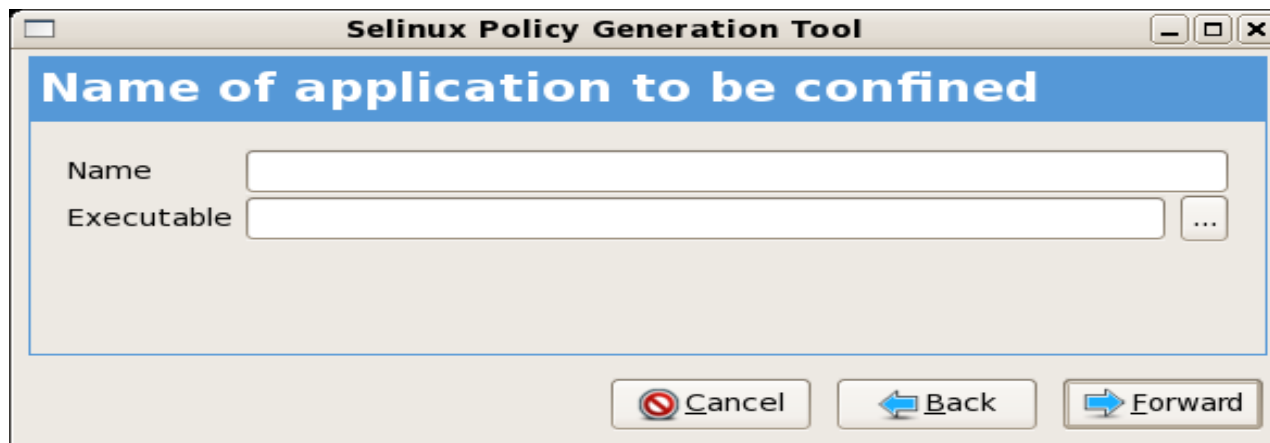
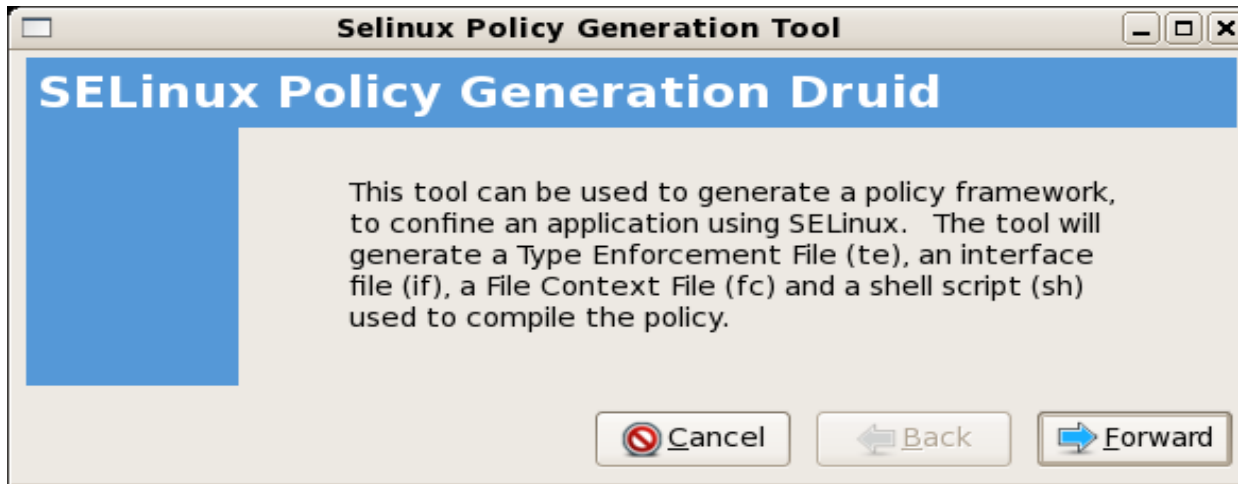
```
myprelink.fc myprelink.if myprelink.pp myprelink.te
```



system-config-selinux



polgengui



polgengui

- Generates 4 files
 - te, fc, if file
 - sh file used to compile/load/set file context
- Tool is not currently an editor.
- Required Fields
 - Name
 - Executable
 - Application Type
- Important Fields
 - What files/directories does application modify?



Lets Start Generating Policy

