SELinux: Bitte nicht deaktivieren ...

Chemnitzer Linux-Tage 2014, Chemnitz

Robert Scheck

fedora™

Robert Scheck

Fedora Package Maintainer und Provenpackager

Fedora Ambassador und Ambassador Mentor

Unterstützung der Website-/Übersetzungsteams

Open Source Contributor und Software-Entwickler

Mail: robert@fedoraproject.org
Web: http://fedoraproject.org/wiki/RobertScheck

Was ist SELinux?

- Security-Enhanced Linux
- Implementation des FLASK-Konzepts (Flux Advanced Security Kernel)

- Zugriffskontrolle auf Ressourcen im Sinne von Mandatory Access Control (MAC)
- Maßgebliche Entwicklung: NSA und Red Hat
- Lizenziert unter GNU General Public License

fedora

Linux-Zugriffskontrolle

- Linux-Zugriffskontrolle kümmert sich um
  - Kernel-Überwachung
  - Prozesse (laufende Programme) und Zugriff auf
  - Ressourcen (Dateien, Verzeichnisse, Sockets, ...)
- Zum Beispiel:
  - Webserver-Prozesse können Webseiten lesen,
  - aber nicht `/etc/shadow`
- Wie werden diese Entscheidungen getroffen?

Standard-Zugriffskontrolle

- Prozesse und Dateien haben Attribute
  - Prozesse: Benutzer/Gruppe (real und effektiv)
  - Ressourcen: Benutzer/Gruppe und Zugriffsbits
    - Lesen, schreiben, ausführen für Benutzer, Gruppe, andere
- Richtlinie ist im Kernel hardcoded
- Beispiel: Kann Firefox meinen privaten SSH-Schlüssel lesen?
  - `robert 3127  1 5 10:00 ?  00:00:29 firefox`
  - `-rw------- 1 robert users 993  6. Feb 2005  id_rsa`

Gängige Sicherheitsprobleme

- Zugriff basiert auf Benutzerrechten
- Beispiel: Firefox kann SSH-Schlüssel lesen
  - Ist zwar normalerweise nicht der Fall, aber:
  - Wenn kompromittiert – potentiell desaströs
- Grundlegendes Problem:
  - Sicherheitsattribute nicht spezifisch genug
  - Kernel kann nicht zwischen Anwendung und Benutzer unterscheiden

Gängige Sicherheitsprobleme

- Prozesse können Sicherheitsattribute ändern
- Beispiel: E-Mail-Dateien nur für mich lesbar
  - Evolution kann diese für alle lesbar machen
- Grundlegendes Problem:
  - Benutzerbestimmbare Zugriffskontrolle, auch genannt Discretionary Access Control (DAC)
  - Prozesse können Sicherheitsrichtlinien anpassen bzw. ignorieren

Gängige Sicherheitsprobleme

- Zwei Berechtigungsstufen: Benutzer und Root
- Beispiel: Rechteausweitung im Apache
  - Apache-Fehler ermöglicht Root-Rechte
  - Gesamtes Linux-System ist kompromittiert
- Grundlegendes Problem:
  - Zu einfache Sicherheitsrichtlinie
  - Keine Möglichkeit minimale Berechtigungen zu erzwingen

Lösung: SELinux

- Zusätzliche Zugriffskontrollen durch SELinux
  - Neue Sicherheitsattribute für Prozesse/Ressourcen
  - Flexible Sicherheitsrichtlinie, die anpassbar ist
- Erzwingung durch Kernel und Anwendungen
- Adressiert Sicherheitsprobleme von Haus aus
  - Zwingend erforderlich (Mandatory Access Control, „MAC“), fein granuliert, minimale Berechtigungen
  - Kein allmächtiger Root-Benutzer
- Transparent für Anwendungen

SELinux-Zugriffskontrolle

- SELinux hat 3 Formen der Zugriffskontrolle
  - Type Enforcement (TE), primärer Mechanismus
  - Rollenbasierte Zugriffskontrolle (RBAC)
  - Multi-Level-Sicherheit (MLS)
- Konfigurierbar über Richtliniensprache
  - Zentrale Konfigurationsdateien für alle Zugriffe
  - Verschiedene Richtlinien (targeted, mls, minimum)
- Jeder Zugriff wird standardmäßig verweigert

SELinux-Sicherheitsattribute

- Prozesse und Dateien mit Sicherheitskontext
  - `robert_u:staff_r:firefox_t:s0`
  - `robert_u:object_r:user_home_t:s0`
  - Benutzer:Rolle:Typ:Level
- Schlüsselfeld ist der Typ
  - Verwendet für Einführung von Type Enforcement
- Restliche Felder für RBAC und MLS
  - Hierzu später mehr

SELinux-Sicherheitskontext

- Verschiedene Tools für SELinux modifiziert
- „-Z“-Option üblicherweise für Kontextanzeige
- Beispiele:
  - `ps auxZ` (zeigt Kontexte von Prozessen)
  - `ls -laZ` (zeigt Kontexte von Dateien/Verzeichnissen)
- Ausgabebeispiele von „ls -Z“:
  - `----------. system_u:object_r:shadow_t:s0    /etc/shadow`
  - `-rwxr-xr-x. system_u:object_r:udev_exec_t:s0 /sbin/udevd`

Einführung: Type Enforcement

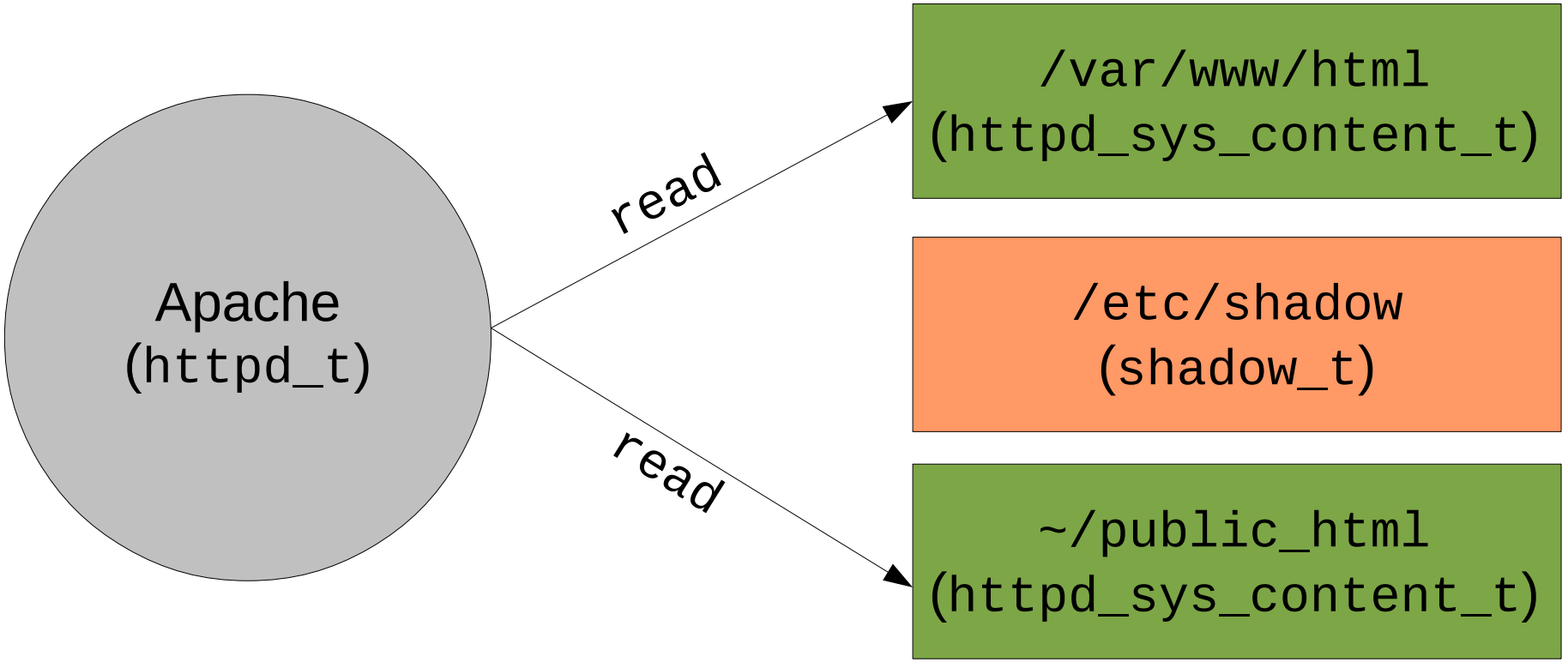
- Basiert auf einzelnem Sicherheitsattribut: Typ
  - Wird auf alle Prozesse & Ressourcen angewendet
  - Repräsentiert sicherheitsrelevante Informationen
- Typ wird Prozessen & Ressourcen zugewiesen
  - Apache-Prozesse → `httpd_t`
  - /var/www/html/index.html → `httpd_sys_content_t`
- Zugriff zwischen Typen wird erlaubt,
  - z.B. `httpd_t` kann `httpd_sys_content_t` lesen

Einführung: Objektklassen

- Objektklassen spezifizieren die Zugriffsdetails
- Ressourcen werden in Klassen unterteilt,
  - z.B. `file`, `lnk_file`, `dir`, `socket`, `process`
- Jede Klasse hat Berechtigungen,
  - z.B. für `file`: `read`, `write`, `execute`, `getattr`
- Lesezugriff über Type Enforcement:
  - `allow httpd_t httpd_sys_content_t:file read;`

Übersicht: Type Enforcement

```
allow httpd_t httpd_sys_content_t:file read;
```



Konzept: Type Enforcement

- Zugriff wird ausschließlich über Typ erlaubt
  - Viele Prozesse & Ressourcen haben gleichen Typ
    - Vereinfacht die Richtlinien durch Gruppierung
  - Prozesse mit gleichem Typ haben gleiche Rechte
    - Gleiches gilt für Ressourcen (Dateien)
- Prozesstypen werden auch „Domains“ genannt
  - Manchmal für Ressourcen verwendet, z.B. Sockets
- Unterschiedliche Ressourcen können gleiche Typen haben

Initiale Typenzuweisung

- Dateien und Verzeichnisse:
  - Konfigurationsdatei definiert Standardkontexte
    - Sogenannte „file contexts“ (`*.fc`) – Dateikontexte
    - Reguläre Ausdrücke, `/usr/(.*/)?bin(/.*)? → bin_t`
  - Vererbung vom Elternverzeichnis zur Laufzeit
- Anwendungen können Kontext explizit setzen
  - `chcon` – Tool für Kontextanpassungen (`→ chown`)
  - `passwd` – Verwaltet den Kontext von `/etc/shadow`

Zuweisung von Prozesstypen

- Prozesstypen werden
  - (standardmäßig) vom Elternprozess vererbt
  - durch Richtlinie gesetzt (Regel für Type Transition)
  - von der Anwendung gesetzt (z.B. `login`)
- Beispiele:
  - `bash` (`user_t`) → `ls` (`user_t`)
  - `init` (`init_t`) → httpd-Initskript (`initrc_t`) → `httpd` (`httpd_t`)
  - `login` (`login_t`) → `bash` (`user_t`)

Regeln für Type Transition

- Type Transition-Regeln setzen Prozesstypen:
  - Anhand Elternprozesstyp und Dateityp des Diensts
  - Ähnlich wie setuid()
- Beispiel: Nameserver starten
  - Richtlinien-Regel:
    - `domain_auto_trans(initrc_t, named_exec_t, named_t)`
  - Elternprozess (`initrc_t`)
  - Dateityp des Diensts (`named_exec_t`)
  - Ergebnis: `named_t`

Hinweise zur Type Transition

- Hauptgründe für das Setzen von Prozesstypen
  - Stellt sicher: Anwendung läuft in korrekter Domain
  - Benötigt keine Anpassung der Anwendung
- Muss durch Richtlinie erlaubt sein
  - z.B. Apache kann keine Prozesse in `init_t` starten
  - Verhindert Rechteausweitung für Anwendungen
- Bindet bestimmte ausführbare Dateien an eine bestimmte Domain
  - z.B. nur `/usr/bin/passwd` darf als `passwd_t` laufen

Benutzerfeld im Kontext

- `robert_u:staff_r:firefox_t:s0`
- Muss nicht dem Linux-Benutzer entsprechen
- Endet oft mit „_u“: `system_u`, `user_u`
- Keine Verwendung in der „targeted“-Richtlinie
- Dateien und Verzeichnisse:
  - Benutzer wird vom Prozess geerbt
  - Systemprozesse erzeugen Dateien mit dem Dateikontext `system_u`

Rollenfeld im Kontext

- `robert_u:staff_r:firefox_t:s0`
- Für rollenbasierte Zugriffskontrolle (RBAC)
  - Rolle schränkt Type Transitions weiter ein
  - Zusammen mit Type Enforcement (`user_r`/`user_t`)
- Endet üblicherweise mit „_r“
- Ressourcen erhalten standardmäßig `object_r`
- Verwendung in „mls“-Richtlinie
  - `user_r`, `staff_r`, `secadmin_r`

Details zum MCS-Levelfeld

- `robert_u:staff_r:firefox_t:s0`
- Verwendet für Multilevel-Sicherheit, kurz: MLS (oder für Multikategorie-Sicherheit, kurz: MCS)
- Oft unsichtbar in „targeted“-Richtlinie
- Gibt ein Level oder einen Bereich an
  - Einzelnes Level: `s0`
  - Bereich: `s0-s15:c0.c1023`
- Üblicherweise mit Labels übersetzt
  - `s15:c0.c1023` → „SystemHigh“

SELinux-Sicherheitsvorteile

- Typen für wichtige Sicherheitsinformationen
  - Zugriff basiert auf Benutzer- *und* Anwendungs-funktion bzw. -berechtigung
  - Transitions kümmern sich um Prozessketten
- Prozesse mit minimalen Berechtigungen
  - nur was für den Typ erlaubt ist
  - z.B. `httpd_t` kann nur Webseiten lesen
- Rechteausweitung streng kontrolliert
  - Kompromittierung des Apache durch Richtlinie begrenzt

Die „mls“-Richtlinie

- Richtlinie mit Bell-LaPadula-Unterstützung
  - Modell: Vertrauliche Informationen sollen nicht an nicht vertrauenswürdige Personen weitergegeben werden können
- Nur für Server-Betriebssysteme gedacht
  - Keine Unterstützung von X
  - Nur für bestimmte Pakete/Dienste
- Zertifizierung von Red Hat Enterprise Linux in 2007 (mit IBM) gegen LSPP, RBACPP & CAPP auf EAL 4+

Die „targeted“-Richtlinie

- Prozesse sind standardmäßig uneingeschränkt
  - Nur „targeted“ Prozesse sind eingeschränkt
- Uneingeschränkte Domains
  - Benutzerprozesse standardmäßig in `unconfined_t`
  - Systemprozesse in `initrc_t`
  - Uneingeschränkte Prozesse mit gleichem Zugriff als würden sie ohne SELinux laufen
- Dienste mit Richtlinie haben eine Transition von `unconfined_t` nach z.B. `httpd_t` (begrenzter Zugriff)

Konfigurationsdateien

- SELinux-Konfiguration in `/etc/selinux`

```
-rw-r--r--. 1 root root  458 26. Aug 2010  config
-rw-r--r--. 1 root root 2271 22. Jul 2010  semanage.conf
drwxr-xr-x. 5 root root 4096  7. Jun 01:53 mls
drwxr-xr-x. 5 root root 4096  7. Jun 01:53 targeted
```

- `/etc/selinux/config` – Richtlinie und Modus

```
# This file controls the state of SELinux on the system.
# SELINUX= can take one of these three values:
#     enforcing - SELinux security policy is enforced.
#     permissive - SELinux prints warnings instead of enforcing.
#     disabled - No SELinux policy is loaded.
SELINUX=enforcing
# SELINUXTYPE= can take one of these two values:
#     targeted - Targeted processes are protected,
#     mls - Multi Level Security protection.
SELINUXTYPE=targeted
```

Konfigurationsdateien

- contexts: Standardkontexte für das System
- modules: Module zum Bauen der Richtlinie
- policy: Kompilierte SELinux-Richtlinie
- setrans.conf: MLS/MCS-Übersetzungen
- seusers: Mapping Linux-/SELinux-Benutzer

```
$ ls -l /etc/selinux/targeted/
drwxr-xr-x. 4 root root 4096  7. Jun 01:53 contexts
drwxr-xr-x. 3 root root 4096  7. Jun 01:53 modules
drwxr-xr-x. 2 root root 4096  7. Jun 01:53 policy
-rw-r--r--. 1 root root  607 27. Mai 15:44 setrans.conf
-rw-r--r--. 1 root root  176  7. Jun 01:53 seusers
$
```

Boot-Parameter für den Kernel

- Kernel-Parameter übersteuern Einstellungen in `/etc/selinux/config`
- `selinux=0`
  - Startet den Kernel mit deaktiviertem SELinux
  - Alle neu erstellen Dateien haben keinen Kontext
  - Spätere SELinux-Nutzung erfordert Relabeling
- `enforcing=0`
  - Startet den Kernel im „permissive“ Modus
  - Eventuell abweichende Fehlermeldungen zu „enforced“

„man pages“ für „targeted“

```
httpd_selinux(8)       httpd Selinux Policy documentation      httpd_selinux(8)

NAME
       httpd_selinux - Security Enhanced Linux Policy for the httpd daemon

DESCRIPTION
       Security-Enhanced  Linux secures the httpd server via flexible mandatory
       access control.

FILE_CONTEXTS
       SELinux requires files to have an extended attribute to define the  file
       type.   Policy  governs the access daemons have to these files.  SELinux
       httpd policy is very flexible allowing users to setup their web services
       in as secure a method as possible.

       The following file contexts types are defined for httpd:

       httpd_sys_content_t
       - Set files with httpd_sys_content_t if you want httpd_sys_script_exec_t
       scripts and the daemon to read the file,  and  disallow  other  non  sys
       scripts from access.

       httpd_sys_script_exec_t
       - Set cgi scripts with httpd_sys_script_exec_t to allow them to run with
```

Modifizierte Systemwerkzeuge

- „Z“ ist die Antwort für SELinux
  - `ls -Z`
  - `id -Z`
  - `ps auxZ`
  - `lsof -Z`
  - `netstat -Z`
  - `find / -context=`

Modifizierte Systemwerkzeuge

- `cp`
  - Erbt Kontext vom Elternverzeichnis oder setzt den Kontext basierend auf dem System-Standard
  - Option „-a“ behält den ursprünglichen Kontext bei
- `mv`
  - Behält weiterhin den ursprünglichen Kontext bei
- `install`
  - Setzt Kontext basierend auf dem System-Standard
- Ausnahmen durch `restorecond`

SELinux-Pakete & -Werkzeuge

- libselinux mit Bibliothek für Anwendungen
- libselinux-utils
  - `getenforce`: Meldet enforcing/permissive/disabled
  - `setenforce 0/1`: Setzt permissive/enforcing
  - `selinuxenabled`: SELinux-Status für Skripte
  - `matchpathcon`: Zeigt standardmäßigen Kontext
  - `avcstat`: Zeigt SELinux AVC-Statistiken
- libselinux-python und libselinux-ruby
  - API-Schnittstellen

Policycoreutils

- genhomedircon, fixfiles, setfiles, chcat, restorecon, restorecond
- audit2allow, audit2why
  - SELinux AVC-Meldungen anzeigen/verstehen
- secon
  - Zeigt Kontext von Ressourcen und Programmen
- semodule, semodule_deps, semodule_link, semodule_expand, semodule_package
  - Verwaltung von Modulen

SELinux-Meldungen verstehen

- Access Vector Cache (AVC)
  - `/var/log/messages` (ohne auditd)
  - `/var/log/audit/audit.log` (mit auditd)

```
type=AVC msg=audit(1140184056.443:78): avc: denied { use } for ↩
  pid=2185 comm="mingetty" name="ptmx" dev=tmpfs ino=699 ↩
  scontext=system_u:system_r:getty_t:s0 ↩
  tcontext=system_u:system_r:kernel_t:s0 tclass=fd

type=AVC msg=audit(1166017682.366:876): avc: denied { getattr } for ↩
  pid=23768 comm="httpd" name="index.html" dev=dm0 ino=7996439 ↩
  scontext=user_u:system_r:httpd_t:s0 ↩
  tcontext=user_u:object_r:user_home_t:s0 tclass=file
```

SELinux-Meldungen verstehen

- AVC-Meldungen entstehen aus verschiedenen Gründen
  - Datei mit falschem Kontext
  - Prozess läuft unter falschem Kontext
  - Fehler in der SELinux-Richtlinie
    - Noch nie getestete Funktionalität oder Kombination wurde erstmals genutzt
  - Ein Einbruchsversuch

SELinux-Meldungen verstehen

- audit2allow
  - Werkzeug zum Erzeugen von „allow“-Regeln anhand der nicht gestatteten Zugriffe der Logs
  - `audit2allow -i /var/log/audit/audit.log`
    - `allow httpd_t user_home_t:file getattr;`
- audit2why
  - Liefert eine Beschreibung für AVC-Meldungen, warum der Zugriff nicht gestattet wurde
  - Begrenzt hilfreich für Einsteiger, eher benutzt von Richtlinien-Entwicklern

AVC-Meldungen analysieren

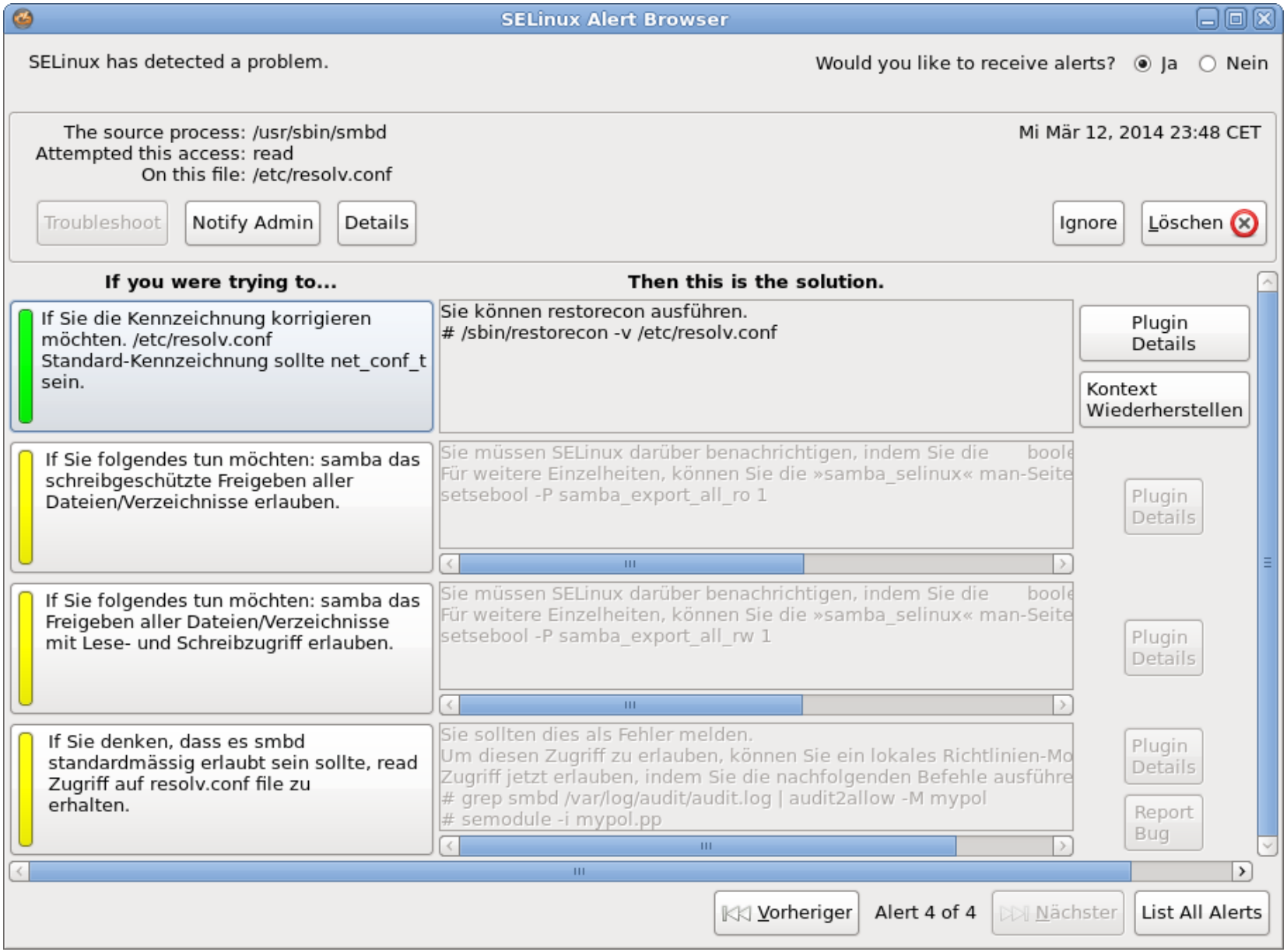
- AVC-Meldungen für Dateien mit `*:file_t`
  - Schwerwiegendes Kontextproblem
  - Datei wurde während `selinux=0` angelegt
  - Relabeling des Dateisystems durchführen
    - `touch /.autorelabel; reboot`
  - Neue Festplatte? `restorecon -R -v /<mnt>`
- AVC-Meldungen, die `default_t` enthalten
  - Vermutlich ein Labeling-Problem
  - Relabeling mit `chcon` oder s.o.

AVC-Meldungen analysieren

- Viele ähnliche Nachrichten zur gleichen Datei
  - Typischerweise ein Labeling-Problem
  - Beispiel:
    - Datei `/home/robert/resolv.conf` anlegen
    - `mv /home/robert/resolv.conf /etc/`
    - `ls -lZ /etc/resolv.conf`
    - Eingeschränkte Domains melden Zugriffsfehler auf `user_home_t`
    - `restorecon /etc/resolv.conf`

SELinux-Troubleshoot-Tool

- setroubleshoot
  - Dienst wartet AVC-Meldungen des Audit-Diensts
  - Plugin-Datenbank für bekannte Probleme
    - `/usr/share/setroubleshoot/plugins/`
  - Zeigt mögliche Abhilfemöglichkeiten/Lösungen
  - `sealert` startet Browser bzw. Protokollanalyse
  - E-Mail-Benachrichtigung durch Konfiguration
    - `/etc/setroubleshoot/setroubleshoot.conf`


SELinux Alert Browser
SELinux has detected a problem.
Would you like to receive alerts? Ja Nein
The source process: /usr/sbin/smbd
Attempted this access: read
On this file: /etc/resolv.conf
Mi Mär 12, 2014 23:48 CET
Troubleshoot
Notify Admin
Details
Ignore
Löschen
If you were trying to...
Then this is the solution.
If Sie die Kennzeichnung korrigieren möchten. /etc/resolv.conf Standard-Kennzeichnung sollte net_conf_t sein.
Sie können restorecon ausführen.
/sbin/restorecon -v /etc/resolv.conf
Plugin Details
Kontext Wiederherstellen
If Sie folgendes tun möchten: samba das schreibgeschützte Freigeben aller Dateien/Verzeichnisse erlauben.
Sie müssen SELinux darüber benachrichtigen, indem Sie die boole
Für weitere Einzelheiten, können Sie die »samba_selinux« man-Seite
setsebool -P samba_export_all_ro 1
Plugin Details
If Sie folgendes tun möchten: samba das Freigeben aller Dateien/Verzeichnisse mit Lese- und Schreibzugriff erlauben.
Sie müssen SELinux darüber benachrichtigen, indem Sie die boole
Für weitere Einzelheiten, können Sie die »samba_selinux« man-Seite
setsebool -P samba_export_all_rw 1
Plugin Details
If Sie denken, dass es smbd standardmässig erlaubt sein sollte, read Zugriff auf resolv.conf file zu erhalten.
Sie sollten dies als Fehler melden.
Um diesen Zugriff zu erlauben, können Sie ein lokales Richtlinien-Mo
Zugriff jetzt erlauben, indem Sie die nachfolgenden Befehle ausführe
grep smbd /var/log/audit/audit.log | audit2allow -M mypol
semodule -i mypol.pp
Plugin Details
Report Bug
Vorheriger
Alert 4 of 4
Nächster
List All Alerts

Fehlende AVC-Meldungen

- Keine AVC-Meldung bei Anwendungsfehler
  - Mit `setenforce 0` versuchen – funktioniert es?
- `dontaudit`-Regeln verhindern AVC-Meldungen
- Fedora 14+ und Red Hat Enterprise Linux 6
  - `semodule -DB  # --disable_dontaudit --build`
- Red Hat Enterprise Linux 5
  - `semodule -b /usr/share/selinux/targeted/enableaudit.pp`
  - `semodule -b /usr/share/selinux/targeted/base.pp`

Dateikontexte verwalten

- `chcon`
  - Grundlegendes Werkzeug für Kontextänderungen
    - `chcon -R -t httpd_sys_script_rw_t \ /var/www/myapp/data`
    - `chcon -t httpd_sys_script_t \ /var/www/cgi-bin/myapp`
  - Aufbau an chmod angelehnt
  - Anpassbare Typen: Kein Relabeling
    - `/etc/selinux/targeted/contexts/customizable_types`
- `touch /.autorelabel; reboot`
  - Vollständiges Relabeling

Dateikontexte verwalten

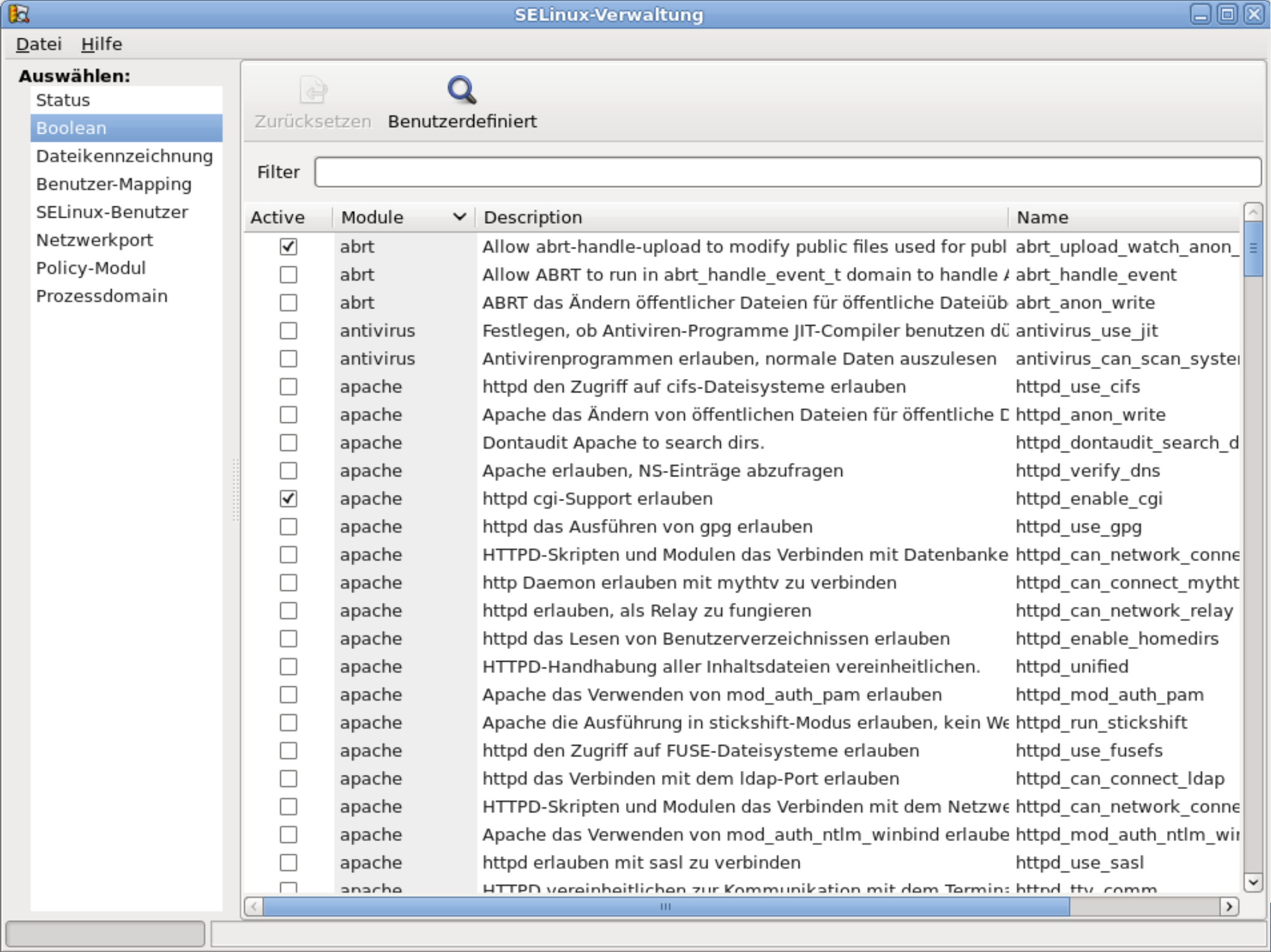
- `restorecon`
  - Setzt eine Datei auf den Standardkontext zurück
  - Arbeitet auf Verzeichnis-/Dateiebene
- `setfiles`
  - Für System-Initialisierung, auf Dateisystemebene
  - Erwartet `file_contexts`-Datei als Spezifikation
- `fixfiles`
  - Shellskript-Wrapper um `setfiles` und `restorecon`
  - RPM-Name als Argument für Relabeling der Dateien im Paket

Dateikontexte verwalten

- `matchpathcon`
  - Zeigt Standardkontext der Ressource
- `semanage`
  - Standardkontext für Ressourcen anzeigen/ändern
  - Verwendet reguläre Ausdrücke für Pfadangaben
  - Viele weitere Funktionalitäten
- `system-config-selinux`
  - Grafisches Frontend für viele CLI-Werkzeuge
  - Etwa `semanage`-Funktionalität

SELinux-Booleans

- Booleans für if/else-Anweisungen in Richtlinie
- Konfiguration ohne Richtline zu bearbeiten
- `getsebool`
  - `getsebool -a`
- `setsebool`
  - `setsebool -P -allow=[1|0]`
- `system-config-selinux`
- Aktiviert/deaktiviert Teile der Richtlinie
  - `setsebool -P virt_use_usb 1`

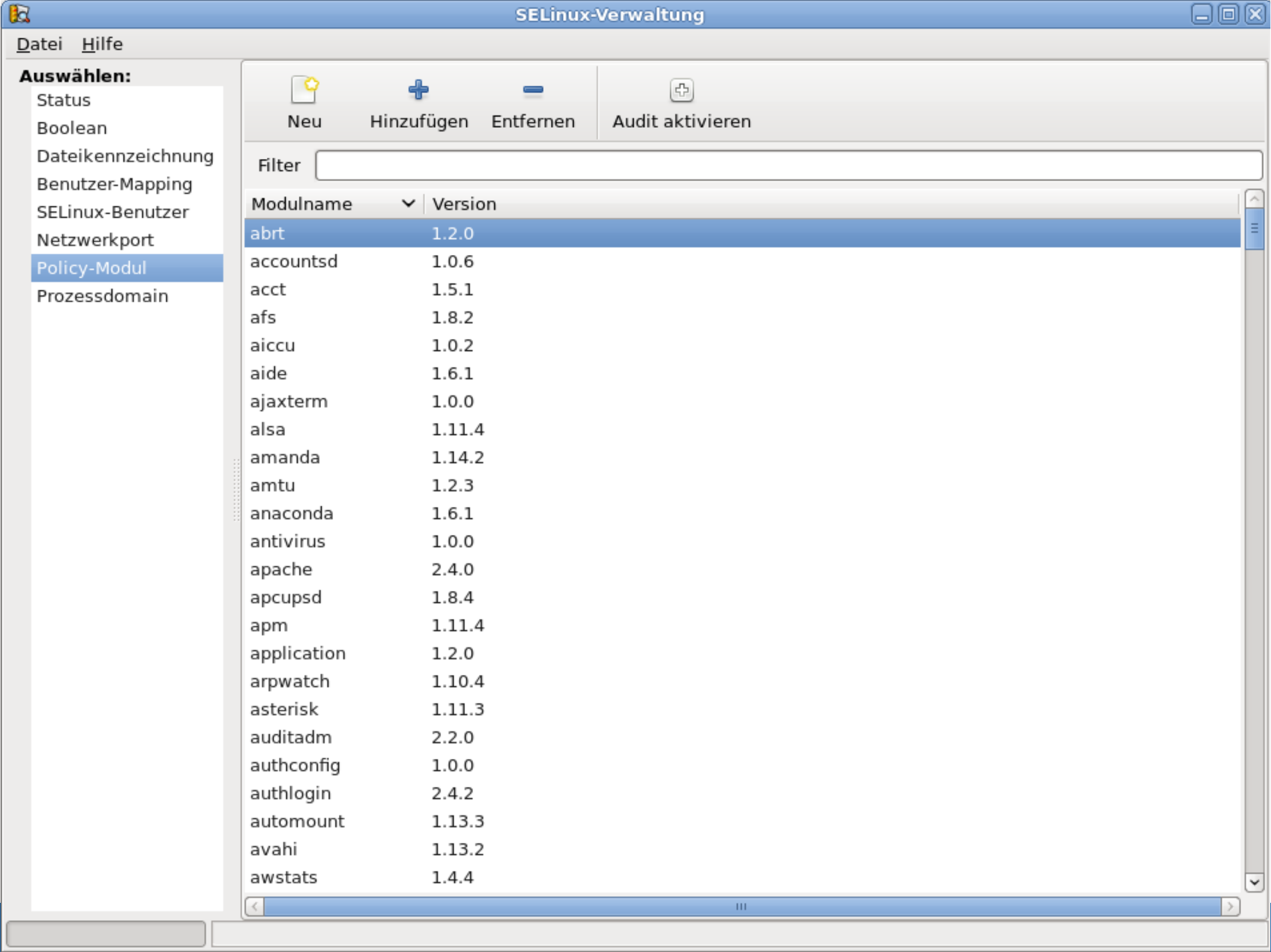

SELinux-Verwaltung
Datei Hilfe
Auswählen:
Status
Boolean
Dateikennzeichnung
Benutzer-Mapping
SELinux-Benutzer
Netzwerkport
Policy-Modul
Prozessdomain
Zurücksetzen
Benutzerdefiniert
Filter
Active Module Description Name
abrt Allow abrt-handle-upload to modify public files used for publ abrt_upload_watch_anon_
abrt Allow ABRT to run in abrt_handle_event_t domain to handle A abrt_handle_event
abrt ABRT das Ändern öffentlicher Dateien für öffentliche Dateiüb abrt_anon_write
antivirus Festlegen, ob Antiviren-Programme JIT-Compiler benutzen dü antivirus_use_jit
antivirus Antivirenprogrammen erlauben, normale Daten auszulesen antivirus_can_scan_syste
apache httpd den Zugriff auf cifs-Dateisysteme erlauben httpd_use_cifs
apache Apache das Ändern von öffentlichen Dateien für öffentliche D httpd_anon_write
apache Dontaudit Apache to search dirs. httpd_dontaudit_search_d
apache Apache erlauben, NS-Einträge abzufragen httpd_verify_dns
apache httpd cgi-Support erlauben httpd_enable_cgi
apache httpd das Ausführen von gpg erlauben httpd_use_gpg
apache HTTPD-Skripten und Modulen das Verbinden mit Datenbanke httpd_can_network_conne
apache http Daemon erlauben mit mythtv zu verbinden httpd_can_connect_mytht
apache httpd erlauben, als Relay zu fungieren httpd_can_network_relay
apache httpd das Lesen von Benutzerverzeichnissen erlauben httpd_enable_homedirs
apache HTTPD-Handhabung aller Inhaltsdateien vereinheitlichen. httpd_unified
apache Apache das Verwenden von mod_auth_pam erlauben httpd_mod_auth_pam
apache Apache die Ausführung in stickshift-Modus erlauben, kein We httpd_run_stickshift
apache httpd den Zugriff auf FUSE-Dateisysteme erlauben httpd_use_fusefs
apache httpd das Verbinden mit dem ldap-Port erlauben httpd_can_connect_ldap
apache HTTPD-Skripten und Modulen das Verbinden mit dem Netzwe httpd_can_network_conne
apache Apache das Verwenden von mod_auth_ntlm_winbind erlaube httpd_mod_auth_ntlm_wi
apache httpd erlauben mit sasl zu verbinden httpd_use_sasl
apache HTTPD vereinheitlichen zur Kommunikation mit dem Termin: httpd_tty_comm

SELinux-Module

- Modulare Richtlinie
  - Modulkonzept seit Fedora Core 5
- `semodule`-Befehl:
  - Kopiert „Policy Package“ (`*.pp`) ins Verzeichnis `/etc/selinux/targeted/modules/active/modules`
  - Kompiliert alle installierten `*.pp`-Dateien als Datei `/etc/selinux/targeted/policy/policy.24`
  - Erzeugt die Dateien `file_context` und `file_context.homedirs` neu
  - Lädt eine neue Richtlinie

SELinux-Module

- `semodule`-Befehl:
  - `semodule -l`
    - Zeigt alle zur Zeit geladenen SELinux-Module
  - `semodule -i /usr/share/selinux/targeted/gpg.pp`
  - `semodule -i meinmodul.pp`
    - Lädt (installiert) ein „Policy Package“
  - `semodule -r meinmodul`
    - Entlädt (entfernt) ein „Policy Package“


SELinux-Verwaltung
Datei
Hilfe
Auswählen:
Status
Boolean
Dateikennzeichnung
Benutzer-Mapping
SELinux-Benutzer
Netzwerkport
Policy-Modul
Prozessdomain
Neu
Hinzufügen
Entfernen
Audit aktivieren
Filter
Modulname
Version
abrt 1.2.0
accountsd 1.0.6
acct 1.5.1
afs 1.8.2
aiccu 1.0.2
aide 1.6.1
ajaxterm 1.0.0
alsa 1.11.4
amanda 1.14.2
amtu 1.2.3
anaconda 1.6.1
antivirus 1.0.0
apache 2.4.0
apcupsd 1.8.4
apm 1.11.4
application 1.2.0
arpwatch 1.10.4
asterisk 1.11.3
auditadm 2.2.0
authconfig 1.0.0
authlogin 2.4.2
automount 1.13.3
avahi 1.13.2
awstats 1.4.4

Richtlinien-Module erstellen

- Richtlinien-Modul besteht aus 3 Dateien
  - Type Enforcement-Datei (`*.te`)
    - Enthält `allow`-Regeln und Interface-Aufrufe, die mit der eingeschränkten Domain zusammenhängen
  - File Context-Datei (`*.fc`)
    - Enthält alle Ressourcen-Labels des Moduls
  - Interface-Datei (`*.if`)
    - Enthält alle Schnittstellen, für andere Domains zur Interaktion mit dieser eingeschränkten Domain
    - `DOMAIN_domtrans`, `DOMAIN_read_config`

Richtlinie mit audit2allow

- Für kleine Anpassungen an der Richtlinie
- `grep httpd_t /var/log/audit/audit.log \ | audit2allow -M mypolicy`
  - Erstellt eine `*.te`-Datei und kompiliert diese in eine `*.pp`-Binärdatei
- `semodule -i mypolicy.pp`

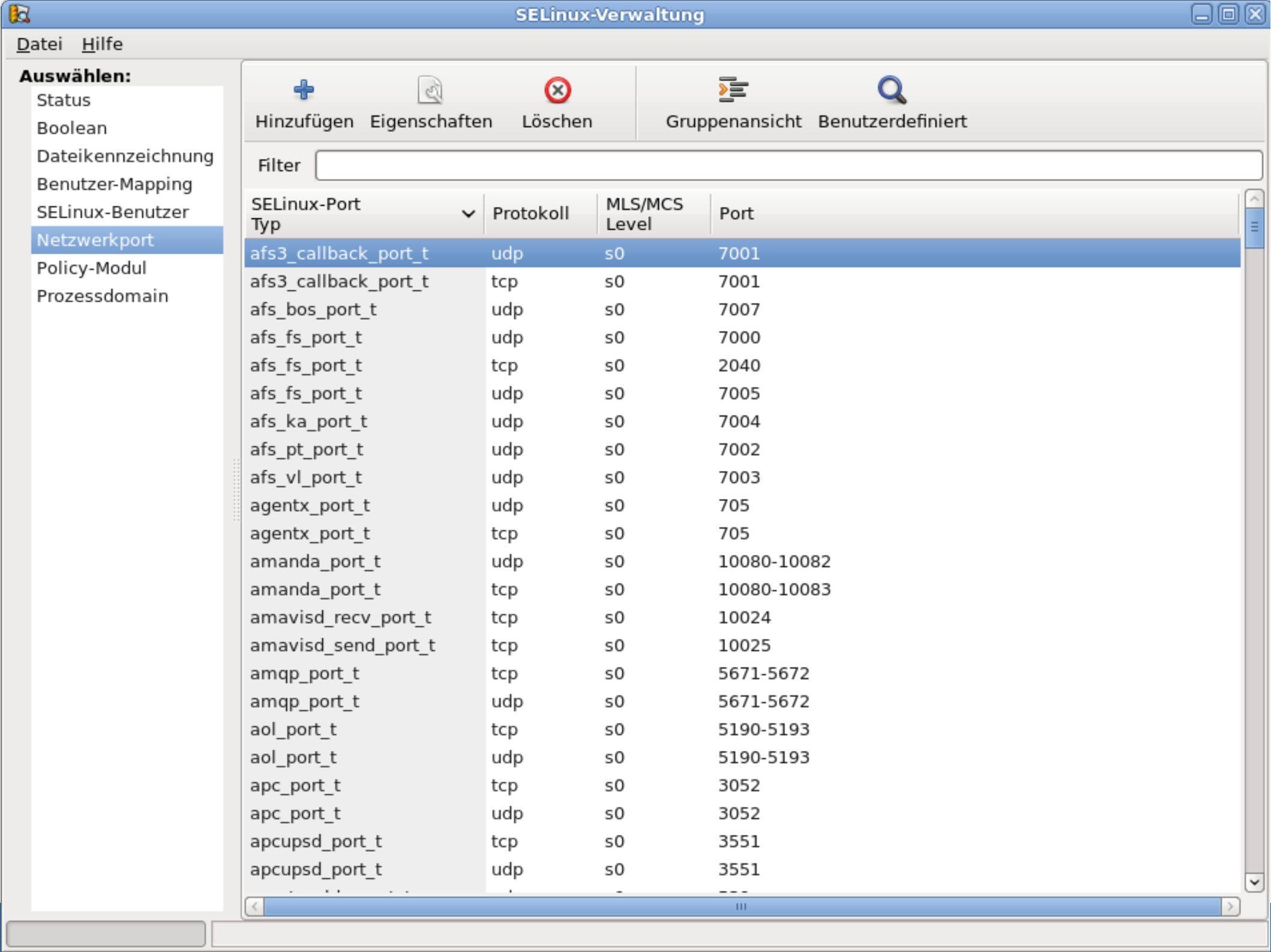
SELinux-Systeme verwalten

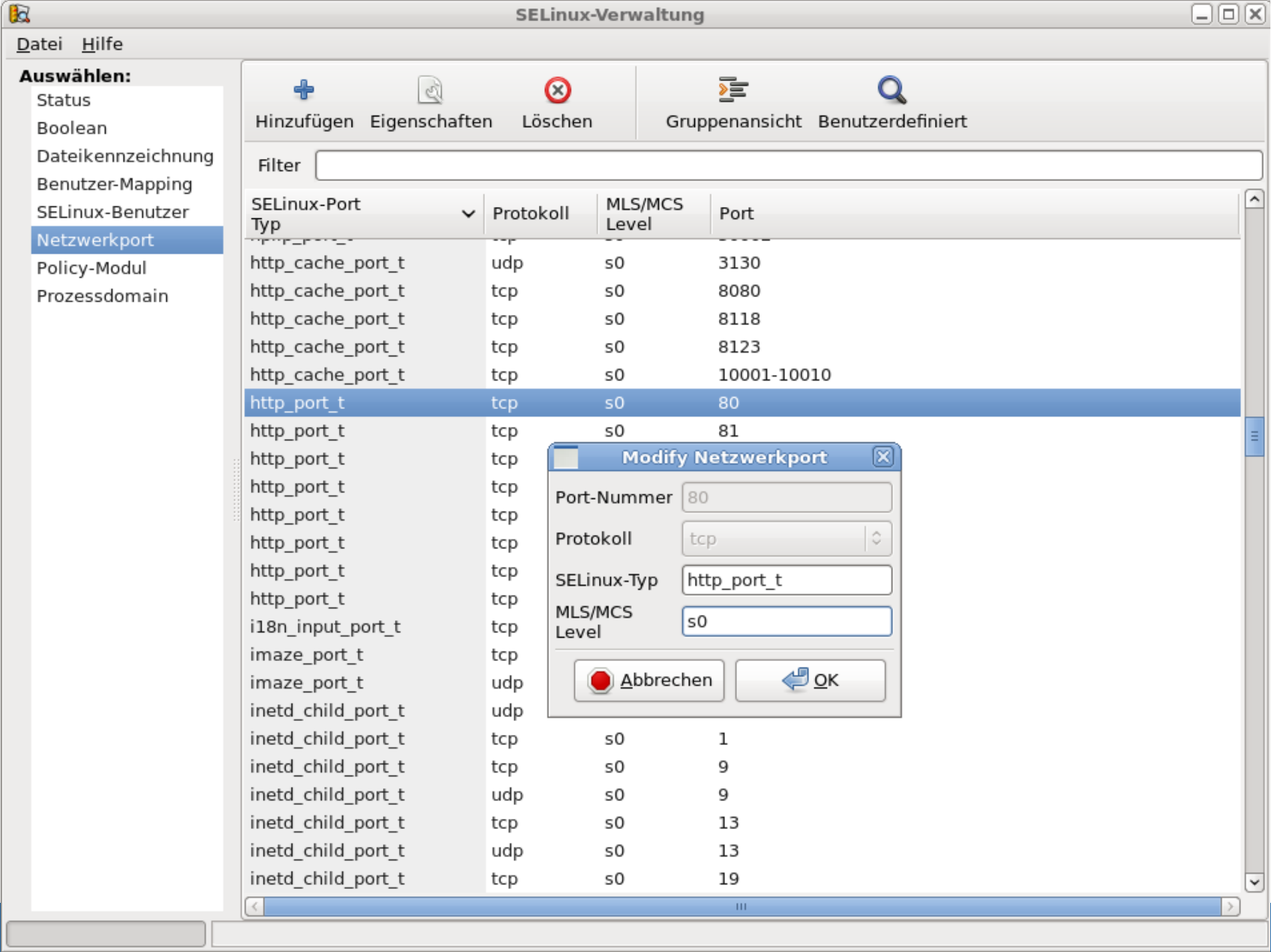
- `semanage`-Framework seit Fedora Core 5
- Vermeidet viele eigene Richtlinien/Module
- Beispiel:
  - Ohne `semanage`-Framework:
    - Apache erlauben sich an Port 81 zu binden
    - Benötigt eine eigene Richtlinie/Modul
  - Mit `semanage`-Framework:
    - `semanage port -a -t http_port_t -P tcp 81`

semanage-Befehle/-Parameter

- SELinux-Benutzer
  - `semanage user -l`
  - `semanage user -a guest_u`
- Linux- auf SELinux-Benutzer-Zuordnung
  - `semanage login -a -s guest_u robert`
- Dateikontexte
  - ```
    semanage fcontext -a -t \
    httpd_bugzilla_script_exec_t \
    '/usr/share/bugzilla/cgi(/.*)?'
    ```


    ```


SELinux-Verwaltung
Datei
Hilfe
Auswählen:
Status
Boolean
Dateikennzeichnung
Benutzer-Mapping
SELinux-Benutzer
Netzwerkport
Policy-Modul
Prozessdomain
Hinzufügen
Eigenschaften
Löschen
Gruppenansicht
Benutzerdefiniert
Filter
SELinux-Port Typ
Protokoll
MLS/MCS Level
Port
afs3_callback_port_t udp s0 7001
afs3_callback_port_t tcp s0 7001
afs_bos_port_t udp s0 7007
afs_fs_port_t udp s0 7000
afs_fs_port_t tcp s0 2040
afs_fs_port_t udp s0 7005
afs_ka_port_t udp s0 7004
afs_pt_port_t udp s0 7002
afs_vl_port_t udp s0 7003
agentx_port_t udp s0 705
agentx_port_t tcp s0 705
amanda_port_t udp s0 10080-10082
amanda_port_t tcp s0 10080-10083
amavisd_recv_port_t tcp s0 10024
amavisd_send_port_t tcp s0 10025
amqp_port_t tcp s0 5671-5672
amqp_port_t udp s0 5671-5672
aol_port_t tcp s0 5190-5193
aol_port_t udp s0 5190-5193
apc_port_t tcp s0 3052
apc_port_t udp s0 3052
apcupsd_port_t tcp s0 3551
apcupsd_port_t udp s0 3551


SELinux-Verwaltung
Datei Hilfe
Auswählen:
Status
Boolean
Dateikennzeichnung
Benutzer-Mapping
SELinux-Benutzer
Netzwerkport
Policy-Modul
Prozessdomain
Hinzufügen
Eigenschaften
Löschen
Gruppenansicht
Benutzerdefiniert
Filter
SELinux-Port Typ
Protokoll
MLS/MCS Level
Port
http_cache_port_t udp s0 3130
http_cache_port_t tcp s0 8080
http_cache_port_t tcp s0 8118
http_cache_port_t tcp s0 8123
http_cache_port_t tcp s0 10001-10010
http_port_t tcp s0 80
http_port_t tcp s0 81
Modify Netzwerkport
Port-Nummer 80
Protokoll tcp
SELinux-Typ http_port_t
MLS/MCS Level s0
Abbrechen
OK
i18n_input_port_t tcp
imaze_port_t tcp
imaze_port_t udp
inetd_child_port_t udp
inetd_child_port_t tcp s0 1
inetd_child_port_t tcp s0 9
inetd_child_port_t udp s0 9
inetd_child_port_t tcp s0 13
inetd_child_port_t udp s0 13
inetd_child_port_t tcp s0 19

Auditierung

- Audit-System empfängt SELinux-Ereignisse
- Kein auditd-Prozess
  - AVCs in `/var/log/messages` und `dmesg`
- Mit auditd-Prozess
  - AVCs in `/var/log/audit/audit.log`
- Vollständige Auditierung mit Kernel-Parameter
  - `audit=1`

LSPP, CAPP & RBAC: EAL 4+

- Labeled Security Protection Profile (LSPP)
  - Schutzprofil mit MLS/MCS und MAC (→ B1)
- Controlled Access Protection Profile (CAPP)
  - Schutzprofil mit Benutzern/Authentifikation (→ C1)
- Role-Based Access Control (RBACPP)
  - Schutzprofil mit rollenbasierter Zugriffskontrolle
- Evaluation Assurance Level (EAL 4+)
  - Level der Tests und Dokumentation
  - Methodisch entwickelt, getestet und durchgesehen

aureport

- Zusammenfassung aus Audit-Protokollen
  - `-a` – Bericht über AVC-Meldungen
  - `-i` – Nummerische Felder menschenlesbar darstellen
  - `-ts` „Startzeit“ `-te` „Endzeit“
    - `aureport -a -ts 1:00:00`
- `--success` / `--failed` – Standard ist beides
- `--summary` – Gesamtergebnis der Ereignisse

ausearch

- Protokolle des Audit-Diensts durchsuchen
  - `-m avc` – Ereignistyp, z.B. AVC-Meldungen
  - `-ts` – Startzeit der Suche
  - `-x` – Ausführbare Datei
    - `ausearch -m avc -ts 1:00:00 -x named`

Fazit

- SELinux
  - einfach benutzen
  - bitte nicht deaktivieren
  - schützt wirklich vor Systemeinbrüchen
  - ist kostenlose Sicherheit

fedora

Weiterführende Ressourcen

- Informationen
  - http://www.nsa.gov/research/selinux
  - http://docs.fedoraproject.org/en-US/ ↩ Fedora/13/html/Security-Enhanced_Linux/
  - http://fedoraproject.org/wiki/SELinux
- Mailinglisten
  - selinux@tycho.nsa.gov
  - selinux@lists.fedoraproject.org

Fragen?

fedora™

Vielen Dank!