

# SELinux Policy Analysis Concepts and Techniques

---

2005 SELinux Symposium

David Caplan <dac@tresys.com>

Tresys Technology

<http://www.tresys.com>

# Type Enforcement

---

- Flexible mandatory access control
- No intrinsic security model
  - unlike MLS and other MAC systems
- Almost any security goal can be modeled
  - isolation
  - information flow
  - confidentiality
  - integrity and self-protection
  - least privilege (assurance)

# Policy Complexity in SELinux

---

- Flexible Linux MAC comes with a price
  - Linux is a rich general purpose operating system
  - Desire for fine granularity and least privilege
    - requires more rules and types
- Example: Fedora Core 3 strict policy
  - > 1.2K types
  - 39 kernel object classes & 197 unique permissions
  - > 40K type enforcement rules in policy source
  - > 360K type enforcement rules in policy binary

# TE Policy Analysis Methodology

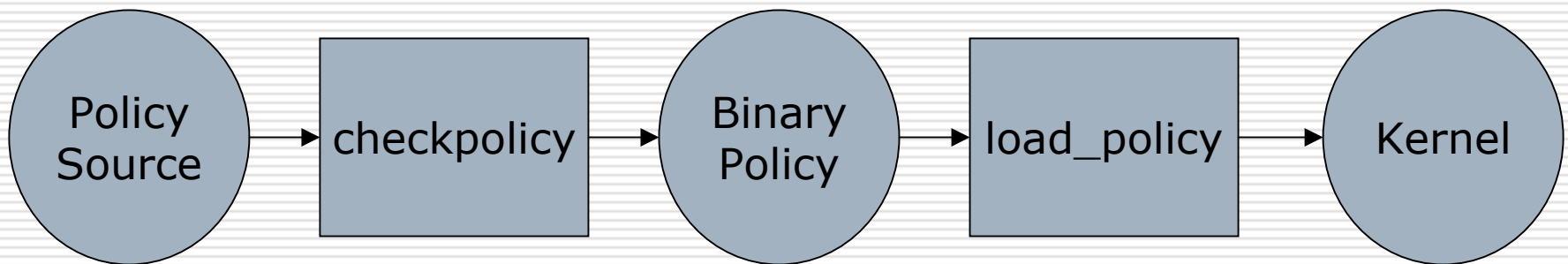
---

1. Identify security goals
2. Map security goals to type architecture
  - formulate hypotheses of type usage
3. Policy analysis
  - examine access rules
  - against type hypotheses and security goals
  - several possible analysis techniques
4. System-to-Policy analysis

# 1. Security Goals

---

- What are the security goals of the system?
  - e.g., integrity of SELinux policy files and policy loading



- Which security goals can be enforced by SELinux? And which can not?

# 1. Security Goals Example

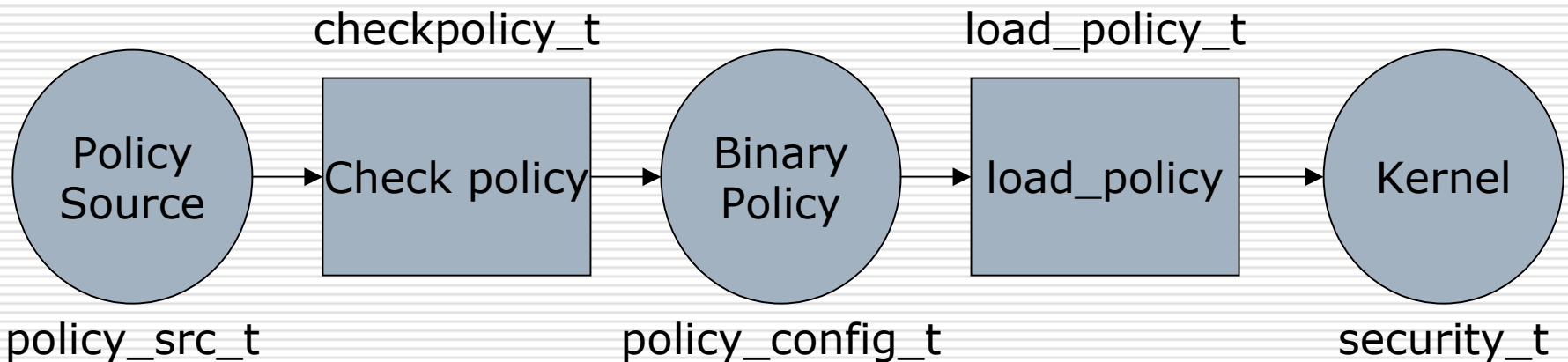
---

- Checkpolicy
  - Correctly compile binary policies
  - Only use appropriate policy source files
  - Only program that can create binary policy files
- Load\_policy
  - Correctly load binary policy into the kernel
  - Only use appropriate binary policy files
  - Only program that can load policies to kernel

## 2. Mapping to Type Architecture

---

- Look at only types and attributes
- Determine which types are intended to address relevant security goals
- Formulate independent hypotheses about policy treatment of types
- Become familiar with policy



# 3. Analyze Policy Against Security Goals

---

- Now look at access rules for types
- Determine if rules are consistent with hypotheses
  - If not iterate or note issue
  - Issues can be fixed or accepted as risks
- Some policy analysis techniques
  - ad hoc analysis
  - re-label analysis
  - information flow analysis



# Ad hoc Policy Analysis

---

- Search rules for type access
  - types as source and targets
  - resolve attributes and implied access
  - Multiple rules for same access
- Valuable analysis
  - spot check
  - regression testing
  - greater sense of the policy
- Least rigorous
- Analysis always done

# Ad hoc Policy Analysis

The screenshot displays the SE Linux Policy Analysis tool interface. The main window title is "SE Linux Policy Analysis - /tmp/etc/selinux/strict/src/policy/policy.conf". The interface includes a menu bar (File, Search, Query, Advanced, Help) and a tabbed interface with "Policy Components", "Policy Rules", "File Contexts", "Analysis", and "policy.conf".

Under the "Policy Rules" tab, there are sub-tabs for "TE Rules", "Conditional Expressions", and "RBAC Rules". The "Rule Selection" section is highlighted with a red circle and contains the following options:

- allow
- neverallow
- auditallow
- dontaudit
- type\_trans
- type\_change

A "Select All" button is located to the right of these options. Below this, the "Search Options" section includes:

- Only search for enabled rules
- Mark enabled conditional rules
- Mark disabled conditional rules
- Enable Regular Expressions

The "Type Enforcement Rules Display" section is also visible. The "Types/Attributes" section is highlighted with a red circle and contains:

- Use Source Type/Attrib
- Include Indirect Matches
- As source (selected) / Any
- Types
- Attribs
- Search field: ^checkpolicy\_t\$

A second, larger view of the "Types/Attributes" section is shown in a separate window, also highlighted with a red circle. It contains the same options as the main window, with the search field value ^checkpolicy\_t\$ circled in red.

At the bottom of the interface, a status bar displays: "Classes: 54 Perms: 197 Types: 1254 Attribs: 83 TE rules: 40661 Roles: 6 Users: 4 v.18 (source)".

# Ad hoc Policy Analysis

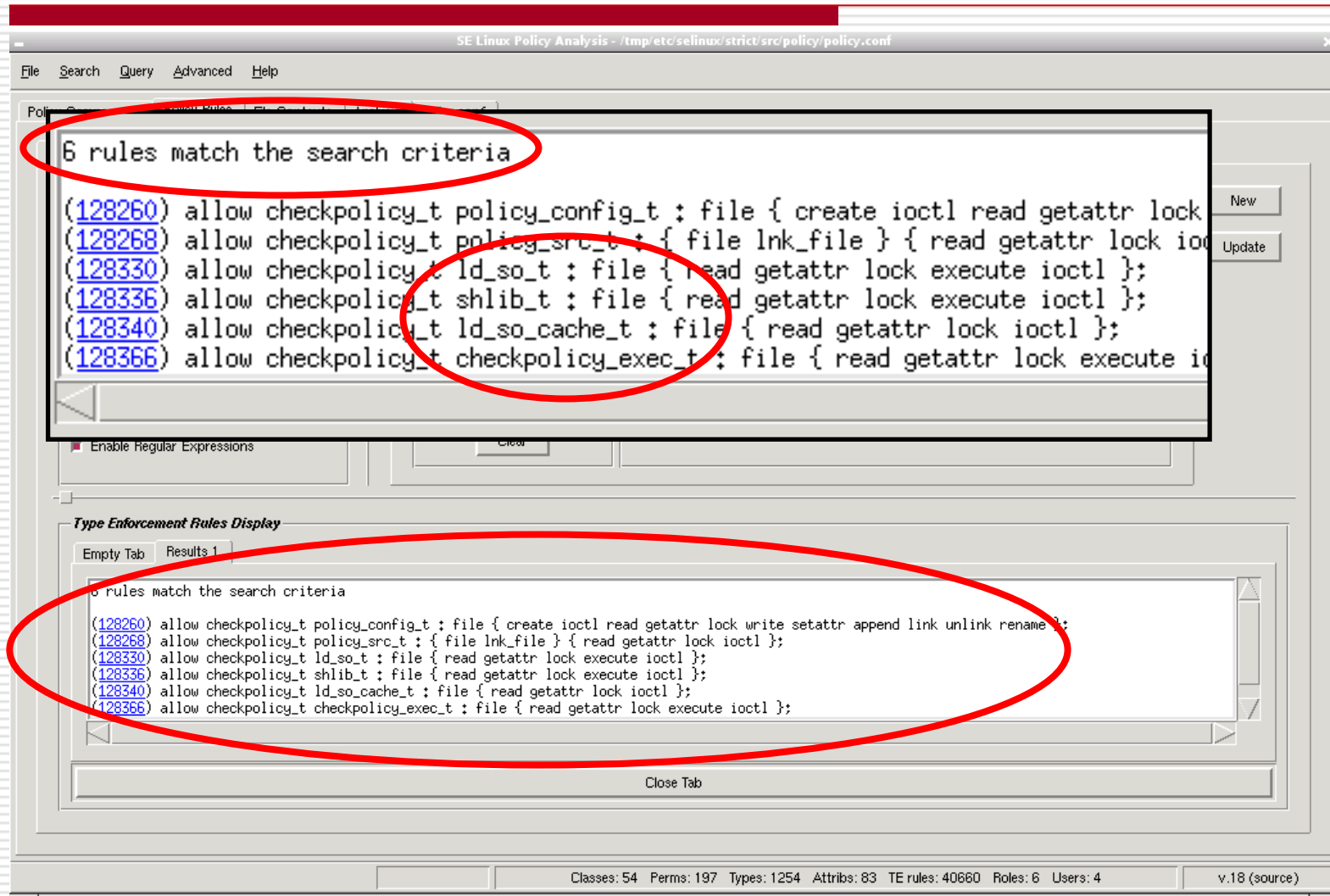
The screenshot displays the SE Linux Policy Analysis tool interface. The main window title is "SE Linux Policy Analysis - /tmp/etc/selinux/strict/src/policy/policy.conf". The interface includes a menu bar (File, Search, Query, Advanced, Help) and a tabbed interface with "Policy Components", "Policy Rules", "File Contexts", "Analysis", and "policy.conf".

Key components of the interface include:

- Rule Selection:** A section with checkboxes for "allow", "neverallow", "auditallow", "type\_trans", and "type\_change", along with a "Select All" button.
- Object Classes:** A list of object classes including "dir", "drawable", "fd", "file\_file", "file", "filesystem", "font", "gc", and "ipc". The "file" class is highlighted with a red circle.
- Classes/Permissions:** A list of permissions including "mounton", "quotaon", "read", "relabelfrom", and "relabelto". The "read" permission is highlighted with a red circle.
- Allow and Audit Rule Permissions:** A section with options for "Show all permissions" and "Only show permissions for selected object classes", along with radio buttons for "Union" and "Intersection".
- Buttons:** "New", "Update", "Clear", and "Reverse" buttons are visible.
- Status Bar:** At the bottom, it shows "Classes: 54 Perms: 197 Types: 1254 Attribs: 83 TE rules: 40661 Roles: 6 Users: 4" and "v.18 (source)".

Red circles highlight the "Classes/Permissions" label, the "file" class, and the "read" permission.

# Ad hoc Policy Analysis



The screenshot displays the SE Linux Policy Analysis interface. The main window shows a search results pane with the following text:

```
6 rules match the search criteria  
(128260) allow checkpolicy_t policy_config_t : file { create ioctl read getattr lock  
(128268) allow checkpolicy_t policy_src_t : { file lnk_file } { read getattr lock ioctl  
(128330) allow checkpolicy_t ld_so_t : file { read getattr lock execute ioctl };  
(128336) allow checkpolicy_t shlib_t : file { read getattr lock execute ioctl };  
(128340) allow checkpolicy_t ld_so_cache_t : file { read getattr lock ioctl };  
(128366) allow checkpolicy_t checkpolicy_exec_t : file { read getattr lock execute i
```

Below this, the 'Type Enforcement Rules Display' section shows a tabbed interface with 'Results 1' selected, displaying the same search results:

```
6 rules match the search criteria  
(128260) allow checkpolicy_t policy_config_t : file { create ioctl read getattr lock write setattr append link unlink rename };  
(128268) allow checkpolicy_t policy_src_t : { file lnk_file } { read getattr lock ioctl };  
(128330) allow checkpolicy_t ld_so_t : file { read getattr lock execute ioctl };  
(128336) allow checkpolicy_t shlib_t : file { read getattr lock execute ioctl };  
(128340) allow checkpolicy_t ld_so_cache_t : file { read getattr lock ioctl };  
(128366) allow checkpolicy_t checkpolicy_exec_t : file { read getattr lock execute ioctl };
```

The status bar at the bottom indicates: Classes: 54 Perms: 197 Types: 1254 Attribs: 83 TE rules: 40660 Roles: 6 Users: 4 v.18 (source)

# Re-labeling Policy Analysis

---

- Re-labeling is a change of an object's type
  - requires `relabelto` and `relabelfrom` permissions
  - As effective as write access
- Identify domains with from-to access for two types
- Can point to other areas of the policy to analyze

# Re-labeling Policy Analysis

The screenshot displays the SE Linux Policy Analysis interface. The main window shows the 'Analysis' tab for 'policy.conf'. The 'Analysis Type' list on the left includes 'Direct File Relabel', which is highlighted with a red circle. The 'Analysis Options' section shows 'Object Mode' selected, also circled in red. The 'Required parameters' section has 'Ending type: policy\_config\_t' entered in the text box, which is also circled in red. The 'Optional result filters' section is empty. A zoomed-in inset at the bottom right shows the 'Analysis Options' and 'Required parameters' sections in more detail, with 'Object Mode', 'From', and 'Ending type: policy\_config\_t' circled in red. The status bar at the bottom indicates 'Classes: 54 Perms: 197 Types: 1254 Attribs: 83 TE rules: 40668 Roles: 6 Users: 4' and 'v.18 (source)'.

# Re-labeling Policy Analysis

The screenshot displays the SELinux Policy Analysis tool interface. The window title is "SE Linux Policy Analysis - /tmp/etc/selinux/strict/src/policy/policy.conf".

**Left Panel: Type policy\_config\_t can be relabeled from:**

- user\_home\_dir\_t
- user\_home\_fingerlog\_t
- user\_home\_irc\_t
- user\_home\_screen\_t
- user\_home\_spamassassin\_t
- user\_home\_ssh\_t
- user\_home\_t**
- user\_home\_tvtime\_t
- user\_home\_xauth\_t
- user\_irc\_exec\_t
- user\_lockdev\_lock\_t
- user\_lpr\_tmp\_t
- user\_mail\_tmp\_t
- user\_mount\_tmp\_t
- user\_mozilla\_ro\_t
- user\_mozilla\_rw\_t

**Right Panel: File Relabeling Results**

**policy\_config\_t can be relabeled:**

- from user\_home\_t by restorecon\_t**
  - (231597) allow restorecon\_t file\_type : { dir file lnk\_file sock\_file fifo\_file } { getattr setattr }
  - (231597) allow restorecon\_t file\_type : { dir file lnk\_file sock\_file fifo\_file } { getattr setattr }
- from user\_home\_t by rpm\_t**
  - (239262) allow rpm\_t { file\_type -shadow\_t } : { file lnk\_file dir fifo\_file sock\_file }
  - (239262) allow rpm\_t { file\_type -shadow\_t } : { file lnk\_file dir fifo\_file sock\_file }
- from user\_home\_t by setfiles\_t**
  - (248290) allow setfiles\_t { file\_type unlabeled\_t device\_type } : { dir file lnk\_file sock\_file fifo\_file }
  - (248291) allow setfiles\_t file\_type : { dir file lnk\_file sock\_file fifo\_file } relabelto;
  - (248290) allow setfiles\_t { file\_type unlabeled\_t device\_type } : { dir file lnk\_file sock\_file fifo\_file }
  - (248291) allow setfiles\_t file\_type : { dir file lnk\_file sock\_file fifo\_file } relabelto;

**Bottom Panel (Zoomed):**

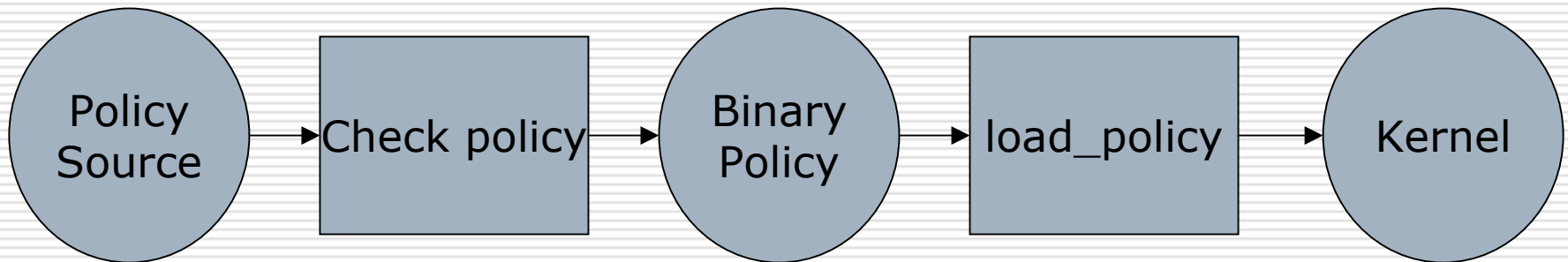
- (239262) allow rpm\_t { file\_type -shadow\_t } : { file lnk\_file dir fifo\_file sock\_file } { relabelfrom relabelto }
- (239262) allow rpm\_t { file\_type -shadow\_t } : { file lnk\_file dir fifo\_file sock\_file } { relabelfrom relabelto }
- from user\_home\_t by setfiles\_t**
  - (248290) allow setfiles\_t { file\_type unlabeled\_t device\_type } : { dir file lnk\_file sock\_file fifo\_file }
  - (248291) allow setfiles\_t file\_type : { dir file lnk\_file sock\_file fifo\_file } relabelto;
  - (248290) allow setfiles\_t { file\_type unlabeled\_t device\_type } : { dir file lnk\_file sock\_file fifo\_file }
  - (248291) allow setfiles\_t file\_type : { dir file lnk\_file sock\_file fifo\_file } relabelto;

Classes: 54 Perms: 197 Types: 1254 Attribs: 83 TE rules: 40661 Roles: 6 Users: 4 v.18 (source)

# Information Flow Policy Analysis

---

- Most comprehensive analysis
- Most difficult and complex analysis
- Iterative process





# Information Flow Policy Analysis

The screenshot displays the SE Linux Policy Analysis tool interface. The main window title is "SE Linux Policy Analysis - /tmp/etc/selinux/strict/src/policy/policy.conf". The interface is divided into several sections:

- Policy Components:** Policy Rules, File Contexts, Analysis (selected), policy.conf
- Analysis Type:** A list of analysis types including Domain Transition, Direct Information Flow, Transitive Information Flow (highlighted with a red oval), Direct File Relabel, and Types Relationship Summary.
- Analysis Options:** A section for configuring the analysis, including:
  - Required parameters:** Starting type: policy\_conf\_t (highlighted with a red oval). A checkbox for "Filter starting types to select using attribute:" is present but unchecked.
  - Flow direction:** Radio buttons for "Flow to" (selected and highlighted with a red oval) and "Flow from".
  - Optional result filters:** A checkbox for "Find end types using regular expression:" is present but unchecked.

At the bottom of the window, a status bar shows: "Classes: 54 Perms: 197 Types: 1254 Attribs: 83 TE rules: 40660 Roles: 6 Users: 4 v.18 (source)".

# Information Flow Policy Analysis

The screenshot displays the SE Linux Policy Analysis tool interface. The main window is titled "SE Linux Policy Analysis - /tmp/etc/selinux/strict/src/policy/policy.conf". The "Analysis Type" is set to "Transitive Information Flow". The "Starting type" is "policy\_config\_t". The "Optional result filters" section has "Find end types using regular expression:" checked. The "Transitive Information Flow Tree" on the left shows a list of process types, with "user\_tmp\_t" highlighted. The "Transitive Information Flow Data" pane on the right shows the following information:

Information flows to **policy\_config\_t** from **user\_tmp\_t** ([Find more flows](#))

**Apol** found the following number of information flows: **4**

**Flow 1** requires **2** step(s).

**Step 1: from anaconda\_t to policy\_config\_t**  
file  
[110194] allow anaconda\_t file\_type : { dir file lnk\_file sock\_file fifo\_file chr\_file blk\_file } \* ;

**Step 2: from user\_tmp\_t to anaconda\_t**  
file  
[110194] allow anaconda\_t file\_type : { dir file lnk\_file sock\_file fifo\_file chr\_file blk\_file } \* ;

The bottom of the window shows summary statistics: Classes: 54 Perms: 197 Types: 1254 Attribs: 83 TE rules: 40660 Roles: 6 Users: 4 v.18 (source).

# 4. System-to-Policy Analysis

---

- Map abstract policy to actual system
- Primarily involves understanding system type labeling
- Does the real system meet expectations
- Two example analyses
  - entry point file
  - real resources for key types

# Entry Point File Analysis

The screenshot displays the SE Linux Policy Analysis tool interface. The main window title is "SE Linux Policy Analysis - /tmp/etc/selinux/strict/src/policy/policy.conf". The interface includes a menu bar (File, Search, Query, Advanced, Help) and a tabbed interface with "Policy Components", "Policy Rules", "File Contexts", and "Analysis" (selected). The "Analysis" tab is active, showing "Analysis Type" and "Analysis Options" sections. The "Analysis Type" list includes "Domain Transition", "Direct Information Flow", "Transitive Information Flow", "Direct File Relabel", and "Types Relationship Summary". The "Analysis Options" section includes "Select direction:" with "Forward" and "Reverse" radio buttons, and "Select target domain:" with a dropdown menu set to "load\_policy\_t". A "Close Tab" button is visible at the bottom. A status bar at the bottom shows "Classes: 54 Perms: 197 Types: 1254 Attribs: 83 TE rules: 40661 Roles: 6 Users: 4" and "v.18 (source)".

Red circles highlight the "Domain Transition" option in the "Analysis Type" list, the "Reverse" radio button in the "Select direction:" section, and the "load\_policy\_t" dropdown in the "Select target domain:" section. A black box highlights the "Reverse" radio button and the "load\_policy\_t" dropdown in a zoomed-in view.

# Entry Point File Analysis

The screenshot displays the SE Linux Policy Analysis tool interface. The main window shows the 'Reverse Domain Transition Information' for the transition from `sysadm_t` to `load_policy_t`. The interface includes a tree view on the left and a main content area on the right. Red circles highlight the domain transition text and the 'Entry Point File Types' section in both the top and bottom views.

**Reverse Domain Transition Information**

Domain transition from `sysadm_t` to `load_policy_t`

**Process Transition Rules: 1**  
(185574) allow sysadm\_t load\_policy\_t : process transition;

**Entry Point File Types: 1**  
load\_policy\_exec\_t

**File Entrypoint Rules: 2**  
(185668) allow load\_policy\_t load\_policy\_exec\_t : file entrypoint;  
(240888) allow load\_policy\_t load\_policy\_exec\_t : file entrypoint;

**File Execute Rules: 5**  
(4882) allow sysadm\_t exec\_type : file { read getattr lock execute ioctl execute\_no\_trans };  
(33788) allow sysadm\_t { bin\_t shin\_t exec\_type } : file { read getattr lock execute ioctl };

**Reverse Domain Transition Information**

Domain transition from `sysadm_t` to `load_policy_t`

**Process Transition Rules: 1**  
(185574) allow sysadm\_t load\_policy\_t : process transition;

**Entry Point File Types: 1**  
load\_policy\_exec\_t

**File Entrypoint Rules: 2**  
(185668) allow load\_policy\_t load\_policy\_exec\_t : file entrypoint;  
(240888) allow load\_policy\_t load\_policy\_exec\_t : file entrypoint;

**File Execute Rules: 5**  
(4882) allow sysadm\_t exec\_type : file { read getattr lock execute ioctl execute\_no\_trans };  
(33788) allow sysadm\_t { bin\_t shin\_t exec\_type } : file { read getattr lock execute ioctl };

Classes: 54 Perms: 197 Types: 1254 Attribs: 83 TE rules: 40661 Roles: 6 Users: 4 v.18 (source)

# Entry Point File Analysis

The screenshot displays the SE Linux Policy Analysis tool interface. The window title is "SE Linux Policy Analysis - /tmp/etc/selinux/strict/src/policy/policy.conf". The "File Contexts" tab is selected and circled in red. Below the tabs, the "File Context Index" section includes "Create and Load" and "Load" buttons, with the "Loaded Index" set to "/tmp/index".

The "Search Options" section on the left has checkboxes for "Show context" and "Show object class". The "Search Criteria" section contains several search options, with "Search Using Object Class" and "Search Using Type" circled in red. The "Search Using Object Class" dropdown is set to "file", and the "Search Using Type" dropdown is set to "load\_policy\_exec\_t". There are also checkboxes for "Enable regular expressions" for each search criterion.

The "Matching Files" section is currently empty. At the bottom of the window, a status bar displays the following statistics: "Classes: 54 Perms: 197 Types: 1254 Attrs: 83 TE rules: 40660 Roles: 6 Users: 4" and the version "v.18 (source)".

# Entry Point File Analysis

The screenshot shows the SE Linux Policy Analysis tool interface. The window title is "SE Linux Policy Analysis - /tmp/etc/selinux/strict/src/policy/policy.conf". The menu bar includes "File", "Search", "Query", "Advanced", and "Help". The main area has tabs for "Policy Components", "Policy Rules", "File Contexts", "Analysis", and "policy.conf".

**File Context Index:** Buttons for "Create and Load" and "Load". Loaded Index: /tmp/index.

**Search Options:**  Show context,  Show object class.

**Search Criteria:**  Search Using User,  Search Using Object Class (file),  Search Using Type (load\_policy\_exec\_t),  Search Using File Path.  Enable regular expressions.

**Matching Files:** FILES FOUND (1):  
system\_u;object\_r;load\_policy\_exec\_t      file      /usr/sbin/load\_policy

**Matching Files (Zoomed):** FILES FOUND (1):  
system\_u;object\_r;load\_policy\_exec\_t      file      /usr/sbin/load\_policy

Classes: 54    Perms: 197    Types: 1254    Attribs: 83    TE rules: 40660    Roles: 6    Users: 4    v.18 (source)

# Real Resources for Key Types

The screenshot shows the SE Linux Policy Analysis tool interface. The main window is titled "SE Linux Policy Analysis - /tmp/etc/selinux/strict/src/policy/policy.conf". The "Search Using Object Class" dropdown is set to "file", and the "Search Using Type" dropdown is set to "shlib\_t". The "Enable regular expressions" checkbox is checked. A "Matching Files" dialog box is open, displaying a list of files found. The dialog box title is "Matching Files" and the content is "FILES FOUND (1980):". The list of files is as follows:

Object Class	Type	File Path
system_u:object_r:shlib_t	file	/lib/libSegFault.so
system_u:object_r:shlib_t	file	/lib/libblkid.so
system_u:object_r:shlib_t	file	/lib/libblkid.so.1
system_u:object_r:shlib_t	file	/lib/libblkid.so.1.0
system_u:object_r:shlib_t	file	/lib/libc-2.3.4.so
system_u:object_r:shlib_t	file	/lib/libc.so.6
system_u:object_r:shlib_t	file	/lib/libcom_err.so
system_u:object_r:shlib_t	file	/lib/libcom_err.so.2
system_u:object_r:shlib_t	file	/lib/libcom_err.so.2.1
system_u:object_r:shlib_t	file	/lib/libcrack.so
system_u:object_r:shlib_t	file	/lib/libcrack.so.2
system_u:object_r:shlib_t	file	/lib/libcrack.so.2.7
system_u:object_r:shlib_t	file	/lib/libcrypt-2.3.4.so
system_u:object_r:shlib_t	file	/lib/libcrypt.so.1
system_u:object_r:shlib_t	file	/lib/libcurses.so
system_u:object_r:shlib_t	file	/lib/libe2p.so
system_u:object_r:shlib_t	file	/lib/libe2p.so.2
system_u:object_r:shlib_t	file	/lib/libe2p.so.2.3
system_u:object_r:shlib_t	file	/lib/libext2fs.so
system_u:object_r:shlib_t	file	/lib/libext2fs.so.2
system_u:object_r:shlib_t	file	/lib/libext2fs.so.2.4



# Conclusions

---

- TE Analysis is challenging and difficult
  - type enforcement is worth the effort
- Analysis tools continue to improve
- Although challenging
  - Can be done
  - Has been done

---

# QUESTIONS?