High Level to Low Level Security Policies

James Carter
Trusted Systems Research
National Security Agency

Background

- Organization responsible for the creation of SELinux, the Flask XEN Security Module, and SE for Android.
- These provide:
  - Flexible Mandatory Access Control (MAC)
  - Fine-grained control of the system
  - Specification of allowed behavior through a security policy

Simple Rules, Complex Policy

- Individual rules are simple:

```
allow user_t file_t: file { read };
```

- But systems are large and complex,
- And control is fine-grained, so
- **The resulting policy is large, complex, and hard to understand**
  - It is hard to even develop the policy in the first place

Unfortunate Results

- No one writes policy from scratch
- Few customize policy to meet their security goals
- The security mechanisms are underutilized
- **Systems are not as secure as they could be**

How do we make policy writing easier?

- Layers of policy
  - Minimize what is needed to be known about lower level policy.
  - Write high-level policies that are translated to low-level policies.
- Better policy abstractions
  - Express more with less

Current Work

- Common Intermediate Language (CIL)
- Policy Driven Systems

Common Intermediate Language (CIL)

- Provide a good target for high-level policies to write SELinux policy
- Encourage experimentation in higher level languages
- Convert SELinux policy toolchain to use CIL

Policy Driven Systems

- Create general toolkit to:
    - Create abstract representations of the system at a given level of granularity
    - Determine the relevant security policy for the representation
    - Allow the operator to understand the impact of changes in the security goals and security policy for the system

The Ideal Solution

- Business logic and information handling requirements automatically translated into low-level policy
- Different information owners can set the policy for their information
- Works for a heterogeneous collection of systems using different low-level security mechanisms.

Questions?