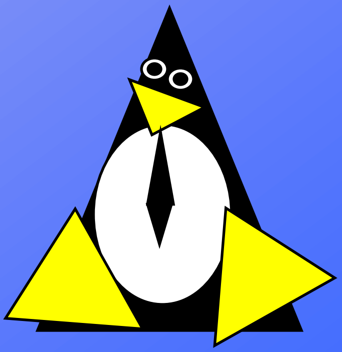
The Sanctions Project

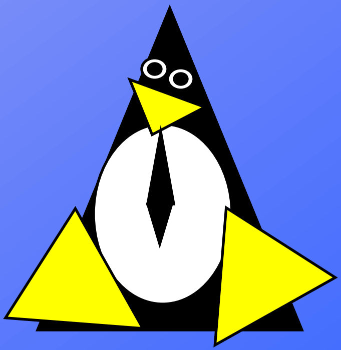
Casey Schaufler

July 2008



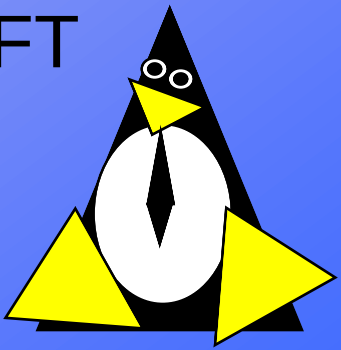
Today's Talk

- Access Controls
- Other Privileged Operations
- Granularity
- Sanctions
- Conclusions



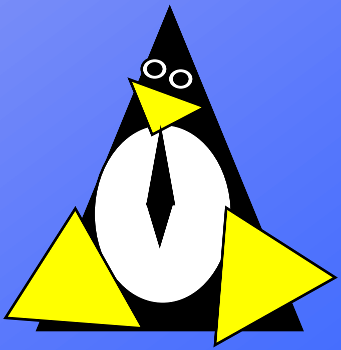
Access Controls

- Subjects Accessing Objects
  - Subject is an active entity
  - Object is a passive entity
  - Access is an operation preformed on an object by a subject
  - Covered by POSIX P1003.1e DRAFT
    - As permitted by P1003



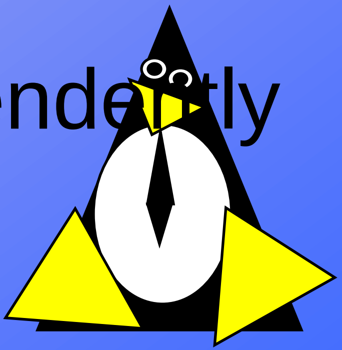
Other Privileged Operations

- Everything Else
  - – Device specific ioctl()s
  - – Privileged ports
  - – System configuration
  - – Tuning



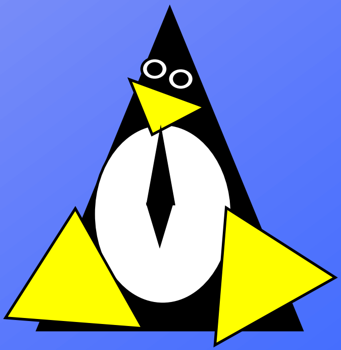
Granularity

- None
  - – Always grant or never grant
- Root or not
- Capabilities
  - – Based on “POSIX objects”
- Each decision considered independently
  - – SELinux does this for its policies



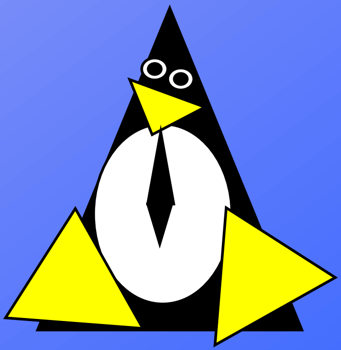
Sanctions

- Each decision considered independently
- `capable(CAP_FOO)`
- `sanction(credential, subject_type, object_type, access_type)`
- Lots of each type



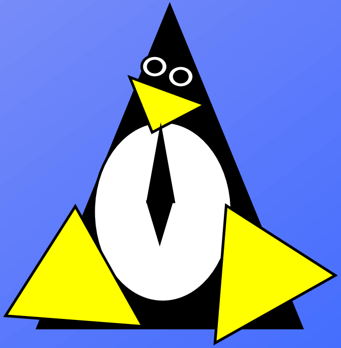
Do We Need Sanctions?

- Detangle capabilities from LSM
  - – SELinux direct capability changes
    - Does it’s own, sort of
  - – Smack already does direct capabilities
  - – Current situation isn’t so bad



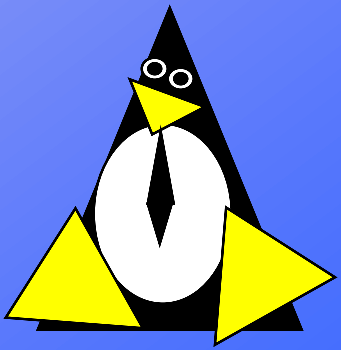
Sanctions Project

- Waiting on active projects
  - Credentials
  - SELinux direct capability calls
- Waiting on a cause
  - Finer granularity than capabilities for access control, not just CAP_SYS_ADMIN



What Have You Learned?

- Sanctions provide arbitrary granularity
- They will be very hard to maintain
- They are waiting on a purpose



Special Thank You

- POSIX P1003.1e/2c Working Group

Contact Information

- http://schaufler-ca.com
- casey@schaufler-ca.com
- rancidfat@yahoo.com