MAC Labeling and Enforcement in NFSv4

David P. Quigley
dpquigl@tycho.nsa.gov
National Security Agency
National Information Assurance Research Laboratory
(NIARL)

Access Control Concepts

- Subjects - Active entities (e.g. executing programs).
- Objects - Passive entities (e.g. files, sockets).
- Reference Monitor
  - Mediates all accesses by subjects to objects.
  - Tamperproof, non-bypassable, verifiable

Discretionary Access Control (DAC)

NATIONAL INFORMATION ASSURANCE RESEARCH LABORATORY

- Typical form of access control.
- Decisions based on user identity/ownership.
- Users and their programs are free to change access rules (e.g. file modes, ACLs).
- No protection against malicious and flawed software.
- Coarse-grained privilege, prone to escalation.

Mandatory Access Control (MAC)

- Historically limited to separate "trusted" operating systems.
- Decisions based on security labels.
- Access rules defined by admin/organization.
- Can confine malicious and flawed software.
- Can enforce system-wide security requirements.

Flexible MAC

NATIONAL INFORMATION ASSURANCE RESEARCH LABORATORY

- Enabling MAC to address a full spectrum of security needs (confidentiality, integrity, least privilege, separation of duty, etc).
- Supports a wide range of security models (BLP, Biba, Type Enforcement, etc).
- Requires encapsulation of security labels/contexts and policy logic.

MAC entering the mainstream

- SELinux released as a proof of concept in December 2000, mainstreamed in Linux 2.6 since 2003.
- FreeBSD MAC framework and SEBSD module.
- Solaris trusted extensions and Solaris FMAC.
- All of these systems could benefit from NFSv4 MAC support.

security_attribute RA

- UTF-8 encoded string
- Per file object attribute
- RA format
  - Opaque data?
  - Structured string?
  - Combination of both? (<opaque>@doi)

Label Change Notification

NATIONAL INFORMATION ASSURANCE RESEARCH LABORATORY

- Label change callback
  - Fall back on cache timeout
  - Scaling problems?
- `OP_PUTFILELABEL`
  - Pass client's idea of label state
  - Server returns `ENFSRETRY` or `ENFSSTALE`
  - Client grabs new file handle.

Process Label Transport (OP_PUTCLIENTLABEL)

- Server needs to know client's process context
- Place PUTCLIENTLABEL call at start of each compound op
- Indicates process context for remaining operations
- Similar semantics to PUTFH

Label Translation

NATIONAL INFORMATION ASSURANCE RESEARCH LABORATORY

- Client and server may have different DOIs
  - different MAC models
  - different policy versions
  - different policy semantic
- Similar to ID -> {g,u}id mapping
- Administration issues?
- Similar model to DNS forwarding?
- Central DOI authority?

Dumb Server

NATIONAL INFORMATION ASSURANCE RESEARCH LABORATORY

- Truly Dumb Server
  - Server does not maintain a DOI
  - Stores label directly
  - Strip or leave the DOI?
- Semi-Dumb Server
  - Server maintains it's own DOI
  - Translates label into its own DOI
  - Appends it's DOI onto label when sending

Exports

- `seclabel` export option: enables exporting of file labels
- `filelabel=<label>` all files on this export given this label
- `clientlabel=<label>` client process label is always given this label

Smart Server

- Maintains a DOI
- Maintains local policy
- Uses client process label in
  - access decisions
  - file creation

Questions?

NATIONAL
INFORMATION
ASSURANCE
RESEARCH
LABORATORY