NFSv4 MAC Attribute Interoperability

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

- Multiple MAC models exist
  - MLS/Biba
  - Type Enforcement
- Multiple MAC policies exist
  - RHEL4/RHEL5/Fedora 9-11
  - RHEL MLS vs Trusted Extensions MLS
- Policy definitions must be flexible
- Accommodating everyone in one format is impossible
NFSv4 MAC Attribute

- Contains two components
  - Opaque label data
  - Some sort of policy/model identifier

- How do we use the opaque data section?
- How do we use the policy/model identifier?
The Old Idea (DOIs)

• A DOI is a unique 32 bit unsigned value
  – Identifies a MAC model and a specific policy

• Problems?
  – DOI space explodes quickly
  – Difficult to manage
  – Makes implementation a nightmare
The New Idea (LFS)

- **LFS** – Label Format Specifier
  - Identifies entry in Label Format Registry
  - Separates label format from label meaning

- **LFR** – Label Format Registry
  - Contains entries describing structure of the opaque field
  - Registry is managed by an external entity
  - Entry 0 reserved for keeping the field completely opaque
Label Format Registry

What is in an Entry?

- Unique identifier for each entry (unsigned int?)
- Description of the format
  - Colon separated string of strings
  - Description of binary encoding of label data
  - Comma separated key/value pairs
- Reference to document describing format

Each format is strongly recommended to contain a field identifying the specific MAC policy
Example

- Deployment uses CALIPSO style MLS with Labeled NFS
- Registers LFS 1 as a CALIPSO label format
  - Places CALIPSO draft as label description
- Format Contains DOI to specify policies
- Label now has two identifiers
  - LFS@<DOI + Binary Label Encoding>
Interop Prototype

- Two Labeled NFS implementations
  - SELinux
  - FreeBSD
- Each end is running a MLS policy
- Each end is running a translation daemon
  - Each agrees on CALIPSO style labels
  - Each has its own local label representation
Prototype Diagram

httpd

Translation Daemon

NFS Client

Send Request

Return SE Linux label

Translate CALIPSO Label

NFS Server

Get File and Attrs

File System

Request myTSfile.txt

Reply with CALIPSO encoded MAC attr

Request myTSfile.txt

Return CALIPSO label

Translate CALIPSO Label

Translate blp=TopSecret

User Space

Kernel Space

NFS Client

fallback_u:fallback_r:fallback_t:SystemHigh

blp=TopSecret
Questions?