Types and Type Attributes

Security Policy Development Primer for Security Enhanced Linux

(Module 6)

Overview of Types and Attributes

- Types: centerpiece of type enforcement
  - used in security contexts and TE rules
  - defined using 'type' statement
- Type Attributes: means of associating groups of types
  - used during policy "compile" process
    - not retained in run-time policy
  - may be used in place of types in TE rules
    - not in security contexts
  - defined using 'attribute' or 'type' statement
- Types and type attributes share same name space

Type Declarations

- Syntax

  type *type_name* [alias *alias_name(s)*] [, *attrib1*, …, *attribn*];

  - types, aliases, and attributes share a common name space
    - all must be unique
  - aliases and attributes are optional
  - aliases are synonymous with type name
    - can be used in running system
    - running system always returns primary type name
  - attributes associate the type with other types
    - all of which share the same attribute name
    - type may have zero or more attributes

Attribute Declarations

- Current version, declared via 'attribute' statement
  - syntax

    attribute *attrib_name*
  - once declared, can be used in type declarations
    - attributes are assigned to types via the type statement
    - the 'attribute' statement only reserves name (no implied policy semantics)
- In older versions, attributes were implicitly declared
  - first time an attribute was used in `type' it was declared
  - this method no longer works—attributes must be explicitly declared before use
- Attribute may be used in TE rules in places of types
  - to apply rule to groups of types

Type Declaration Examples

- types to support passwd program

type passwd_t, domain, privlog, auth, privowner;

type name — assigned attributes

type passwd_exec_t, file_type, sysadmfile, exec_type;

- attributes have no intrinsic meaning
  - their semantics are purely based on rules that use them

Common Type/Attribute Conventions

- Descriptive names
  - type identifiers end with "_t"
  - attribute identifiers are normal words describing a group
- Related types share common root name
  - user_t ordinary user domain
  - user_tmp_t type assigned to /tmp files created by user_t
- For program domains
  - executables that cause domain transition have same root name as domain but end with "exec_t"
    - passwd_t passwd program domain
    - passwd_exec_t executable file type

Common Type/Attribute Conventions

- Example common attributes in sample policy
  - domain
    - conventionally assigned to all domains
  - file_type
    - for all types intended for file related objects
  - user_domain
    - for all domain types intended for human users
  - privlog
    - allows domains to communicate to syslogd for logging

Types and Type Attributes

Exercise
Exploring Types and Attributes

QUESTIONS?