CSci530: Computer Security Systems
Authorization
29 October 2003

Dr. Clifford Neuman

University of Southern California

Information Sciences Institute

Administrative

- Plan to have mid-terms available next Wednesday.
- Most proposals replied to. If you do not have a response by tomorrow morning, send a follow-up message to csci530@usc.edu.

Authorization

- Final goal of security
  - Determine whether to allow an operation.
- Depends upon
  - ▪ Policy
  - ▪ Possibly authentication
  - ▪ Other characteristics

The role of policy in security architecture

Policy – Defines what is allowed and how the system and security mechanisms should act.

Enforced By

Mechanism – Provides protection interprets/evaluates (firewalls, ID, access control, confidentiality, integrity)

Implemented as:

Software: which must be implemented correctly and according to sound software engineering principles.

Policy: Review – The Access Matrix

- Policy represented by an Access Matrix
  - Also called Access Control Matrix
  - One row per object
  - One column per subject
  - Tabulates permissions
  - But implemented by:
    - ▪ Row – Capability list
    - ▪ Column – Access Control List

Policy models: Bell-LaPadula

- Discretionary Policy
  - Based on Access Matrix
- Mandatory Policy
  - Top Secret, Secret, Confidential, Unclassified
  - * Property: S can write O if and only if Level S <= Level O
    - ▪ Write UP, Read DOWN
  - Categories treated as levels
    - ▪ Form a matrix

(more models later in the course)

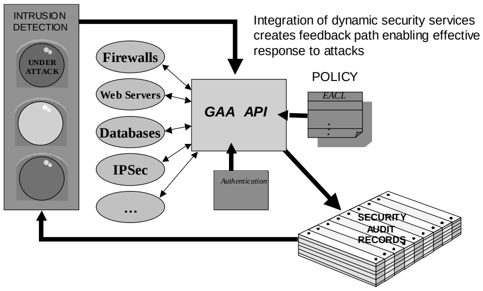
Security is more than mix of point solutions

- **Today's security tools work with no coordinated policy**
  - **Firewalls and Virtual Private Networks**
  - **Authentication and Public Key Infrastructure**
  - **Intrusion Detection and limited response**
- **We need better coordination**
  - **Intrusion response affected at firewalls, VPN's and Applications**
  - **Not just who can access what, but policy says what kind of encryption to use, when to notify ID systems.**
- **Tools should implement coordinated policies**
  - **Policies originate from multiple sources**
  - **Policies should adapt to dynamic threat conditions**
  - **Policies should adapt to dynamic policy changes triggered by activities like September 11th response.**

GAA-API: Integration through Authorization

- **Focus integration efforts on authorization and the management of policies used in the authorization decision.**
  - **Not really new - this is a reference monitor.**
  - **Applications shouldn't care about authentication or identity.**
    - **Separate policy from mechanism**
  - **Authorization may be easier to integrate with applications.**
  - **Hide the calls to individual security services**
    - **E.g. key management, authentication, encryption, audit**

Authorization and Integrated Security Services



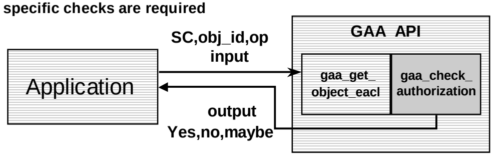
Integration of dynamic security services creates feedback path enabling effective response to attacks

Generic Authorization and Access-control API

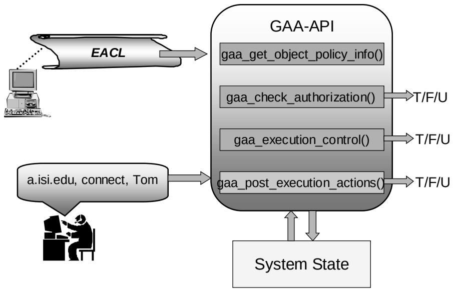
Allows applications to use the security infrastructure to implement security policies.

gaa_get_object_policy_info **function called before other GAA API routines which require a handle to object EACL to identify EACLs on which to operate. Can interpret existing policy databases.**

gaa_check_authorization **function tells application whether requested operation is authorized, or if additional application specific checks are required**



Three Phases of Condition Evaluation



GAA-API Policies originate from multiple sources

- **Discretionary policies associated with objects**
  - **Read from existing applications or EACLs**
- **Local system policies merged with object policies**
  - **Broadening or narrowing allowed access**
- **Policies imported from policy/state issuers**
  - **ID system issues state credentials, These credentials may embed policy as well.**
- **Policies embedded in credentials**
  - **These policies attach to user/process credentials and apply to access by only specific processes.**
- **Policies evaluated remotely**
  - **Credential issuers (e.g. authentication and authorization servers) evaluate policies to decide which credentials to issue.**

Communicating threat conditions

Threat Conditions and New Policies carried in signed certificates

- Added info in authentication credentials
- Threat condition credential signed by ID system

Base conditions require presentation or availability of credential

- Matching the condition brings in additional policy elements.

Integrating security services

The API calls must be made by applications.

- This is a major undertaking, but one which must be done no matter how one chooses to do authorization.

These calls are at the control points in the app

- They occur at auditable events, and this is where records should be generated for ID systems
- They occur at the places where one needs to consider dynamic network threat conditions.
- Adaptive policies use such information from ID systems.
- They occur at the right point for billable events.

Advances Needed in Policy

- Ability to merge & apply policies from many sources
  - Legislated policies
  - Organizational policies
  - Agreed upon constraints
- Integration of Policy Evaluation with Applications
  - So that policies can be uniformly enforced
- Support for Adaptive Policies is Critical
  - Allows response to attack or suspicion
- Policies must manage use of security services
  - What to encrypt, when to sign, what to audit.
  - Hide these details from the application developer.

GAA - Applications and other integration

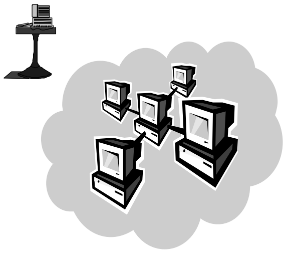
- Web servers - apache
- Grid services - globus
- Network control – IPsec and firewalls
- Remote login applications – ssh
- Trust management
  - Can call BYU code to negotiate credentials
  - Will eventually guide the negotiation steps

What dynamic policies enable

- Dynamic policy evaluation enables response to attacks:
  - Lockdown system if attack is detected
  - Establish quarantines by changing policy to establish isolated virtual networks dynamically.
  - Allow increased access between coalition members as new coalitions are formed or membership changes to respond to unexpected events.

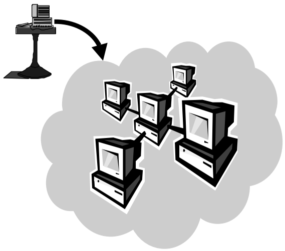
Demo Scenario - LockDown

- You have an isolated local area network with mixed access to web services (some clients authenticated, some not).



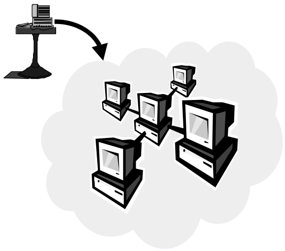
Demo Scenario - LockDown

- You have an isolated local area network with mixed access to web services (some clients authenticated, some not).
- You need to allow incoming authenticated SSH or IPSec connections.



Demo Scenario - LockDown

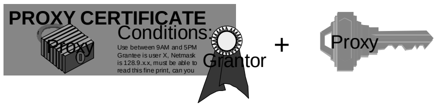
- You have an isolated local area network with mixed access to web services (some clients authenticated, some not).
- You need to allow incoming authenticated SSH or IPSec connections.
- When such connections are active, you want to lock down your servers and require stronger authentication and confidentiality protection on all accesses within the network.



Proxies

- A proxy allows a second principal to operate with the rights and privileges of the principal that issued the proxy
  - Existing authentication credentials
  - Too much privilege and too easily propagated
- Restricted Proxies
  - By placing conditions on the use of proxies, they form the basis of a flexible authorization mechanism

Restricted Proxies


PROXY CERTIFICATE
Proxy
Conditions:
Use between 9AM and 5PM
Grantee is user X, Netmask
is 128.9.x.x, must be able to
read this fine print, can you
Grantor
+
Proxy

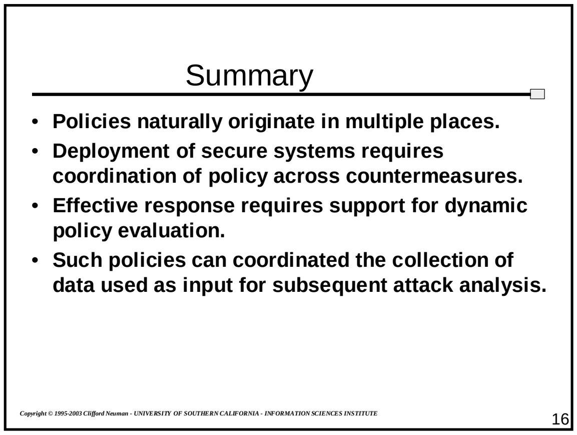
- Two Kinds of proxies
  - Proxy key needed to exercise bearer proxy
  - Restrictions limit use of a delegate proxy
- Restrictions limit authorized operations
  - Individual objects
  - Additional conditions

Policies

- HIPAA, other legislation
- Privacy statements
- Discretionary policies
- Mandatory policies (e.g. classification)
- Business policies

Mechanisms

- Access Matrix
  - Access Control List
  - Capability list
- Unix file system
- Andrew file system
- SSH authorized key files
- Restricted proxies, extended certificates
- Group membership
- Payment



Summary

- **Policies naturally originate in multiple places.**
- **Deployment of secure systems requires coordination of policy across countermeasures.**
- **Effective response requires support for dynamic policy evaluation.**
- **Such policies can coordinated the collection of data used as input for subsequent attack analysis.**