



# AG Security Models and Virtual Venues Server Plans

Robert Olson  
Futures Laboratory  
Access Grid Retreat  
March 5, 2002



# Project goals

- New architecture intended to **enable** ..
  - Development of new technology by the community
  - Integration of a reliable security model
  - Exposing Venues scoping mechanisms to new applications



# Requirements

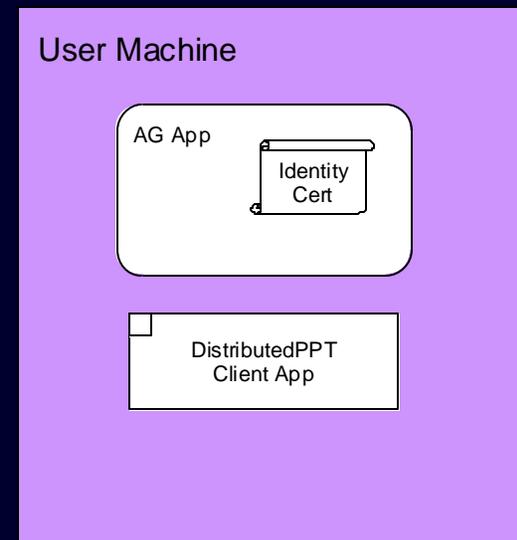
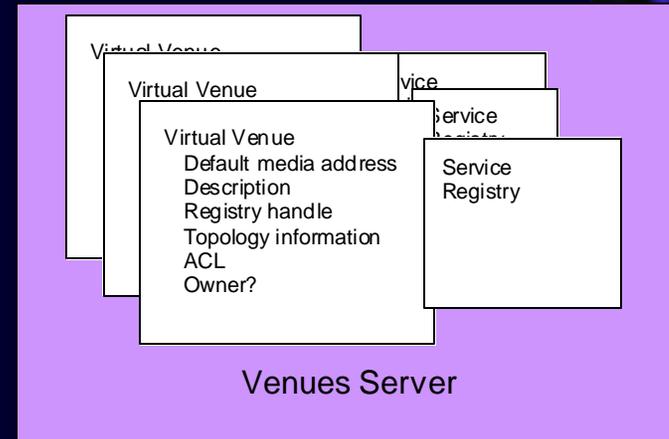
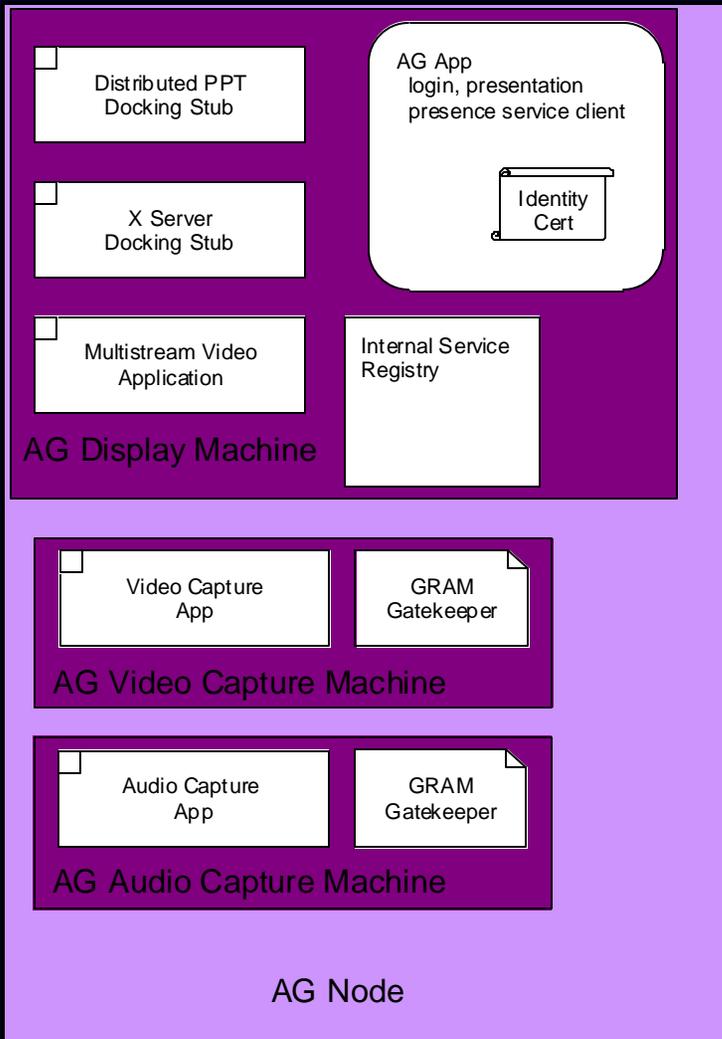
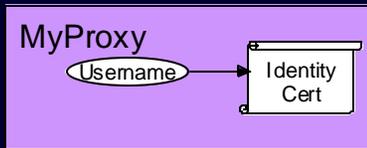
- Community development of new tech
  - Published APIs
  - Integration mechanisms
- Reliable security model
  - Security APIs
  - PKI mechanisms
  - Key management tools
- Exposure of scoping mechanisms
  - Published APIs
  - Tools for leveraging scoping
- ad hoc usage



## How?

- Peer to peer service-based model
- Security built in as a foundation technology

# Architecture Overview





# Strategy

- Framework for implementation of web service model
  - Discovery
  - Service description
  - Secure messaging
  - Data exchange
- (Business world driving web services, but not necessarily in a direction precisely appropriate for us)
- Basic services as part of standard software set
  - Venues
  - Presence
  - Security
- Leverage extended services from the community and other projects



# Design Considerations

- Scalability
  - Target range: small groups,  $O(100)$
  - Still require central servers, but for minimal data (presence)
  - Eliminate servers with secure group communication (InterGroup, ...)
- Security
- Ease of distribution
- Leverage existing technology wherever possible
  - XML / XML Schemas
  - XMLRPC
  - HTTP
  - Apache
  - SSL
  - Globus infrastructure
  - Akenti? CAS?



# Basic services

- Scoping
- Presence
- Security
- Scheduling
  - What do scheduler-writers need?
- Persistence
  - What are the operations on a persistence service?
- Navigation



# What is a Service?

- Named entity performing some function
- Discovery
  - Venues mechanisms provide scoping
- Description
  - XML schema
- Communications
  - XMLRPC APIs



## Okay, so what's in a Venue?

- Ordinary things
  - Name
  - Description
  - Media channels
  - Topology
- Service registry
  - “Service Handles” of logged-in nodes & users
- Discovery?
  - Query nodes & users for desired services
- All wrapped up in an XML description



# Backward Compatibility

- Possible to accommodate current AG nodes
- Translation from XML to HTML
- Support those capabilities with a direct mapping older tech
- Secured spaces?
- Do we need it?

The image features a background of a stone wall with a repeating pattern of rectangular stones in shades of brown and tan. A solid orange horizontal banner is centered across the middle of the image, containing the text "On to Security" in a black, sans-serif font.

On to Security



# Goals of the Security Services Architecture

- Provide a concrete implementation of the things we know we want
  - Identity
  - Basic services for obtaining and managing identity
  - Secure control communications
  - Access control for venues
  - Privacy of media streams
- API for use throughout the system
- Provide hooks / APIs / Protocols for future extensibility
  - “Correct” solutions not yet clear
- Single Sign-on



# Identity

- X.509 Identity Certificates
- Problems
  - Key management
  - Semantics of identity
  - Establishing trust
  - Casual / one-time users
  - Host Certificates
- Initial implementation: Globus identity certificates
  - Globus Project runs a CA
  - Other entities can run CAs as desired (trust)
  - Enough to bootstrap the project



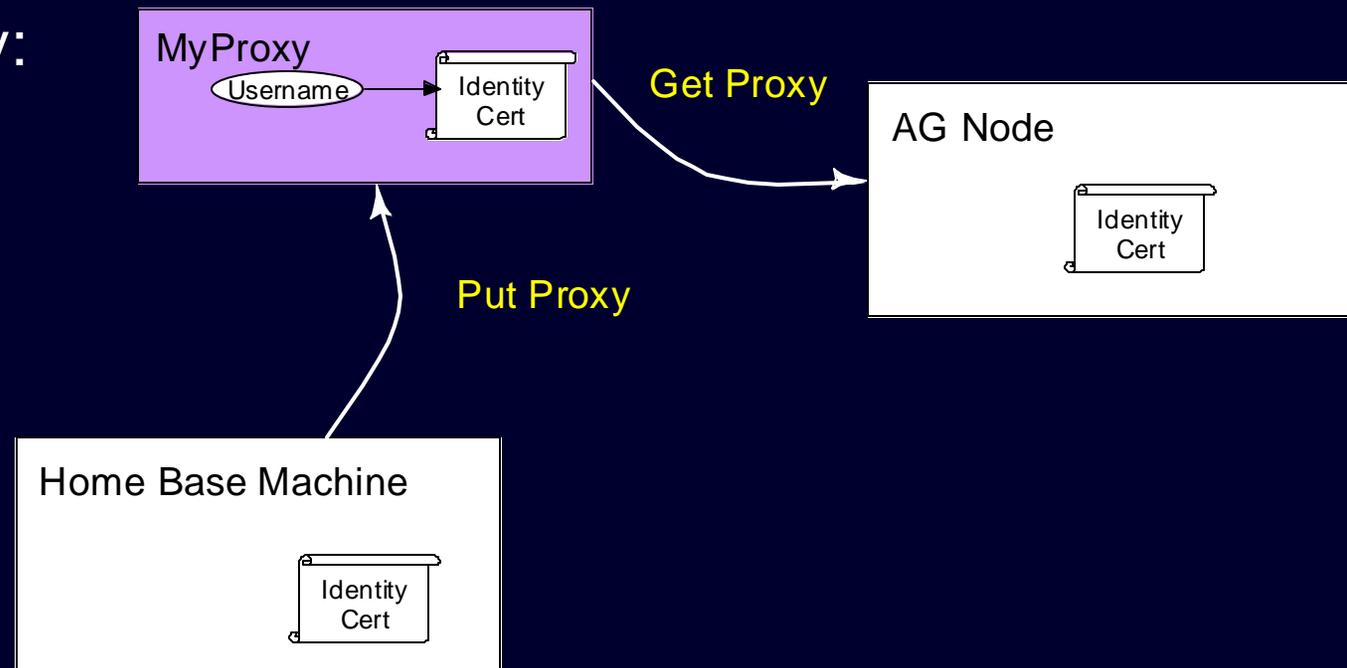
# Proxy Certificates

- Mechanism to support single sign-on
- Create short-lived proxy identity certificates from long-lived certificate
- Why?
  - Proxies kept without passphrases
  - Delegation mechanism used in Globus for information access, process startup, etc.
  - Restricted proxies



# Key Management

- Private key lives on disk in one location
- But I want to use my identity anywhere
- MyProxy:





# Key Management

- Possibly not ideal
  - MyProxy server possible single point of failure
  - Paranoia factor: Do I want a proxy held by someone else?
- But limited lifetimes and restricted proxies help
- Other solutions
  - Online CAs where keys retrievable at any time
  - “Username/Password” registration ? certificate
  - ???
- Answers here provide for single sign-on



# Secure communications

- Authentication
  - Ensure both sides have certificates
  - Verification rules (trusted CAs, etc)
- SSL / GSI
- XMLRPC over HTTPS



# Access Control

- Hard problem: dynamic groups, dynamic resources
- Multiple mechanisms
  - Simple ACLs
  - Directory-based group authorizations (mod\_ldap\_auth)
  - Globus Community Authorization Services
  - Akenti
  - Capability Certificates
- Initial choices...
  - Likely simple ACLs or LDAP solutions
  - Still to be decided
- ...may depend on context



# Stream Security

- Current vic / rat support AES/Rijndael encryption
- Key distribution via venues services mechanisms
- Per RFC1889
- Vague worries...
  - Are keys recoverable (in face of many gigabytes of encrypted data)
  - Rekeying intervals?
- IETF Secure RTP draft (draft-ietf-avt-srtp-02)
  - Implementations?
  - Who's interested?
- However...



# How much do we care?

- What is the level of paranoia?
- What is the acceptable level of inconvenience for security?
- Do we want military level cryptographic protections, or just to keep the demo folks out of our group meeting?
- Auditing?
- Interested in user perspectives
- GGF ACE-RG draft Informational Document on Security Scenarios
- Possibility of spinning up GGF ACE Security WG



# Firewalls

- How paranoid are the firewall admins?
- Current solutions
  - Put AG outside the firewall
  - Burn holes through the firewall
- Interested in usage scenarios, acceptable practices from firewall admins
- Future solutions
  - AG media / control proxies on firewall?
  - Mutual authentication agreements between firewall and AG infrastructure
  - ???