Security Model in the Ambient Computational Environment

James Mauro

May 24, 2004

Committee: Dr. Minden, Dr. Agah, Dr. Alexander

Thanks

- Dr. Minden
- Dr. Agah and Dr. Alexander
- Leon Searl
- Eric Akers, Renzo Hayashi and Olaf Landsiedel
- All of the audience members for attending

Overview

- Related Work
- Ambient Computational Environment
- Enhanced RMI
- Encryption and Authentication
- Keynote Trust-Management
- Future Work

Related Work

- Remote Method Invocation (RMI)
  - Remote Access to a service
  - Stubs
- JINI
  - Discovery
  - Join
  - Lookup
- Ninja
  - Similar model to ACE
  - NinjaRMI
  - Secure Directory Service

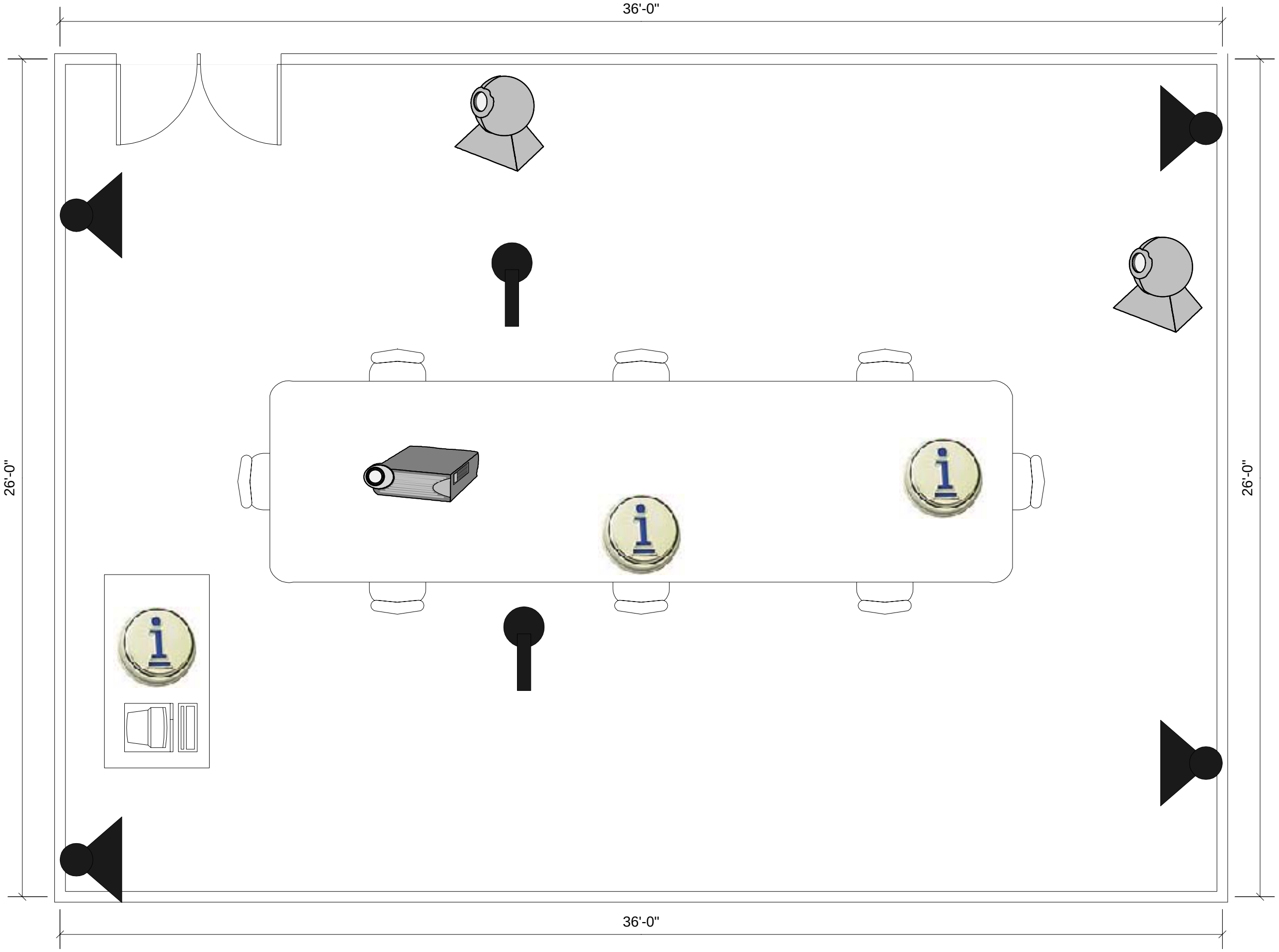
Ambient Computational Environment

- Computational Resources are available throughout the environment
- Users can co-opt services in their vicinity
- Computational Sessions are long lived and not tied to any one room
- Computational environment reacts to audio and visual cues from the user

Service Architecture

- Services are the core components
- Services designed to be simple and perform only one function
- More complex services can be formed by federating services
- A number of “core services” exist for users and services to learn about the environment

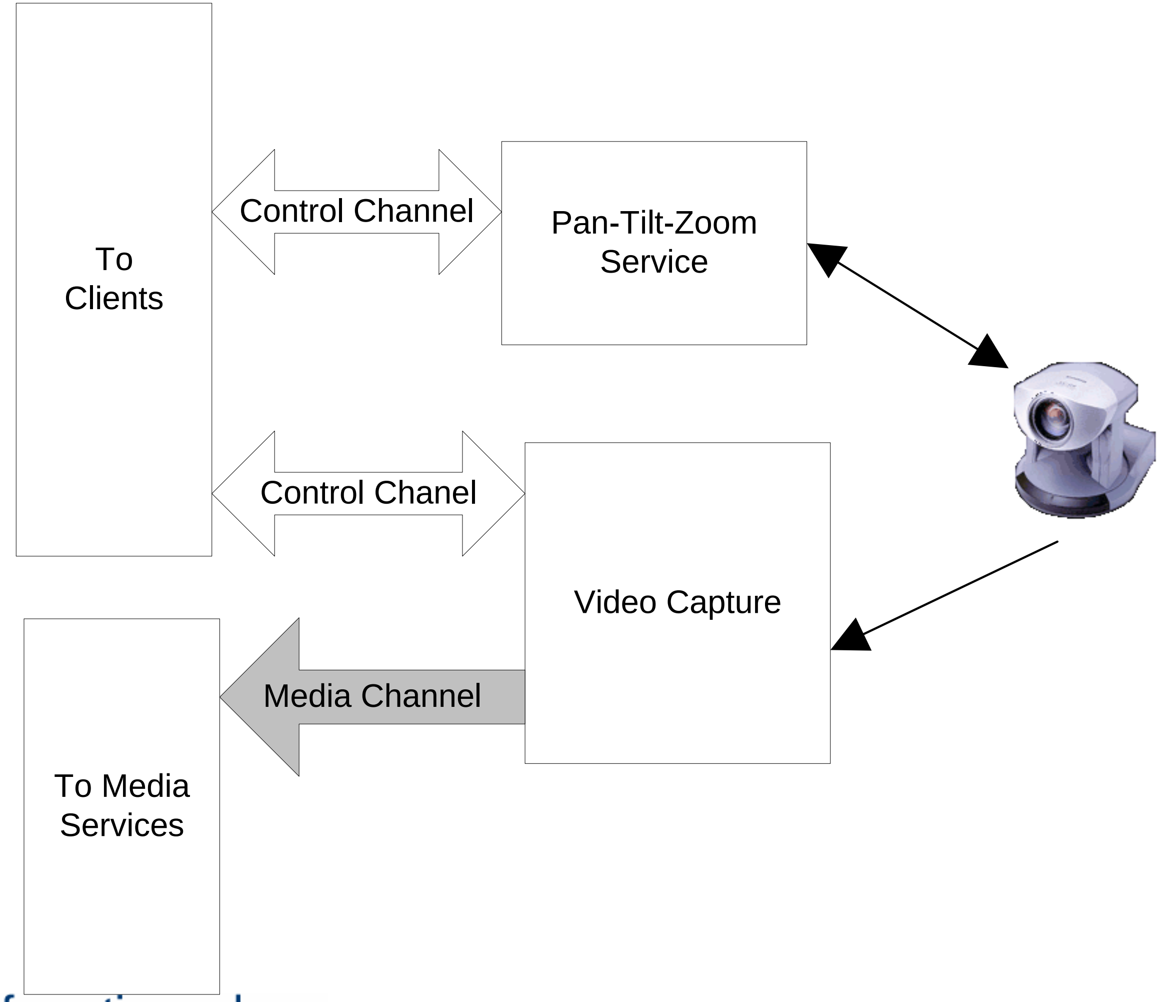
Room View



Service Communications

- Two main communications channels
- Control Channel
  - Reliable
  - In-order delivery
  - Bi-directional
- Media Channel
  - Unreliable
  - Unidirectional
  - Timeliness

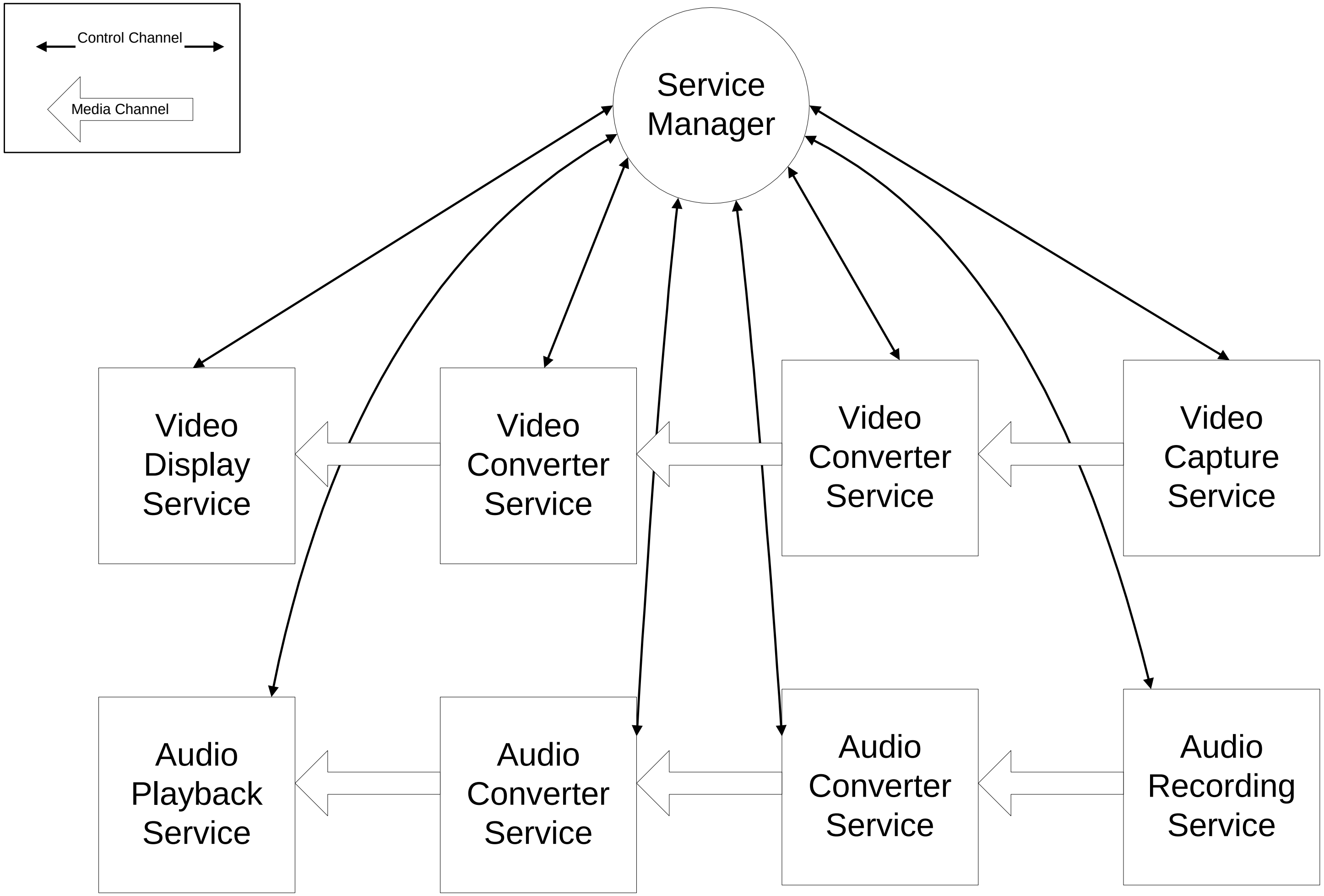
Service Architecture



Service Federations

- Simple Services
  - One function
  - Cannot directly use most other services
- Complex Services can be formed by creating federations
- Federations are managed by a client called a manager
- Federation exists while the manager wants it to
- After a federation ends, the services can join other federations

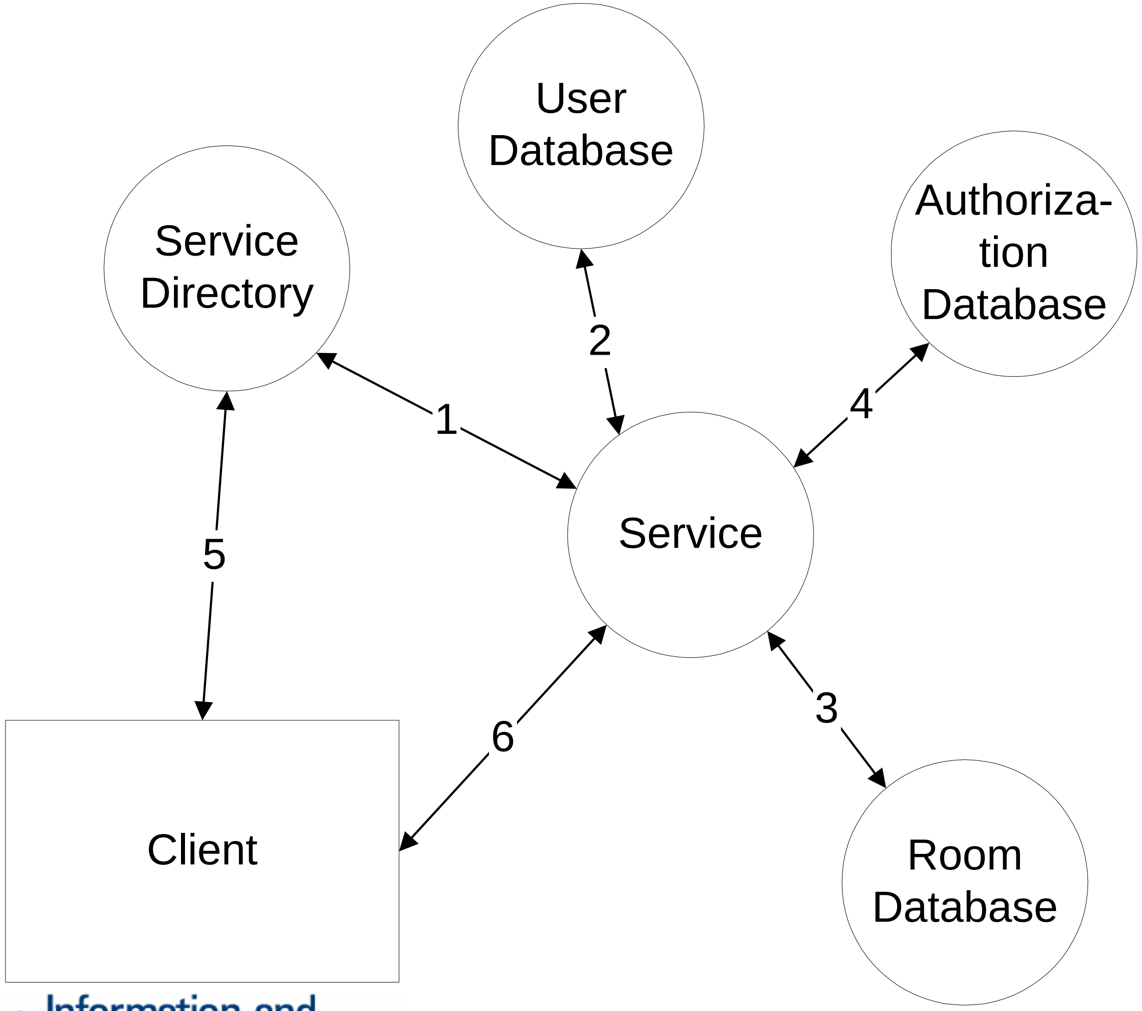
Service Federations



Core Services

- Service Directory
  - Stores where the services are in the network and environment
- User Database
  - “passwd” equivalent
- Room Database
  - Describes buildings, machines, and rooms
- Authentication Database
  - Stores Keynote assertions for authentication

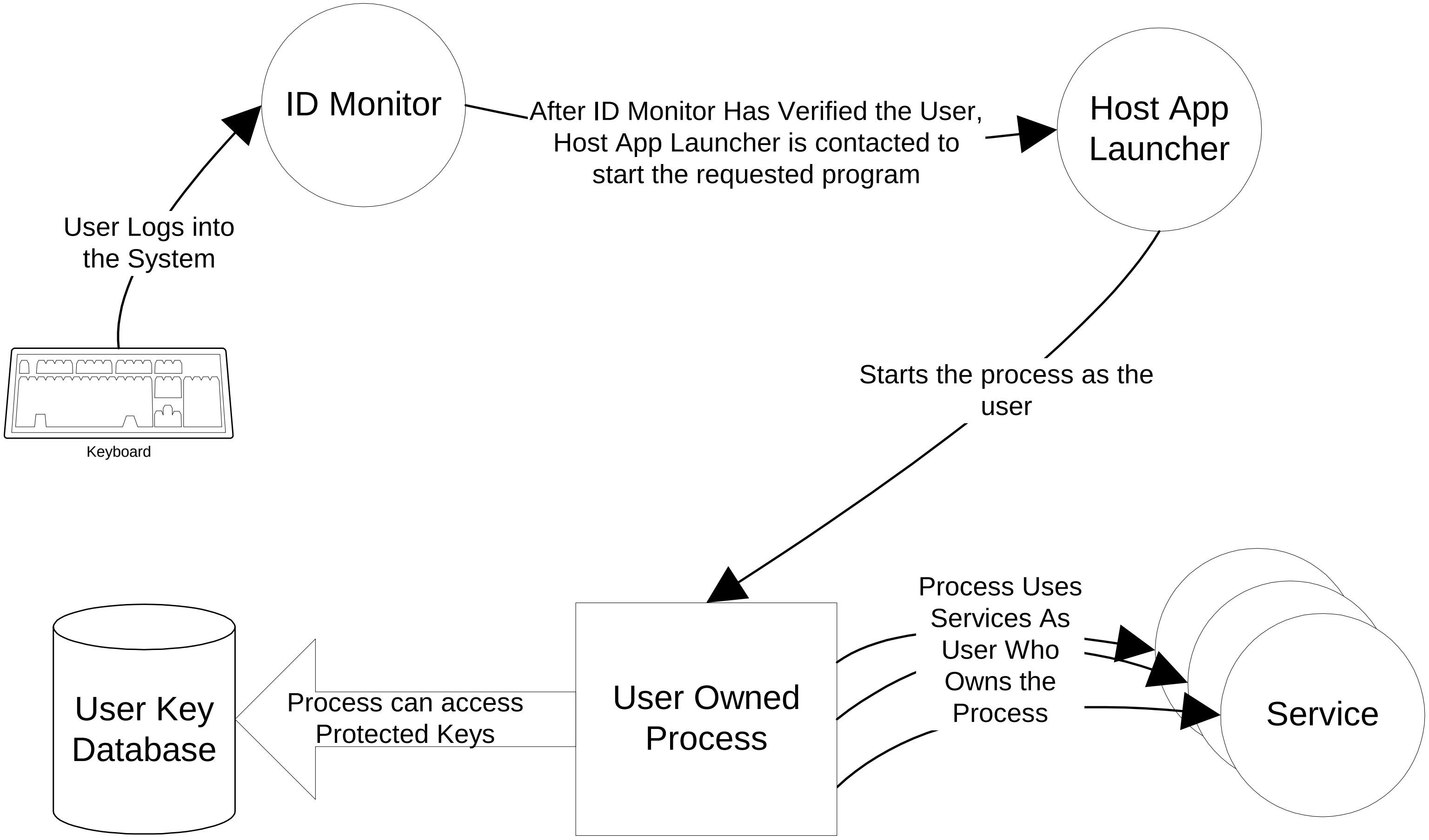
Core Services



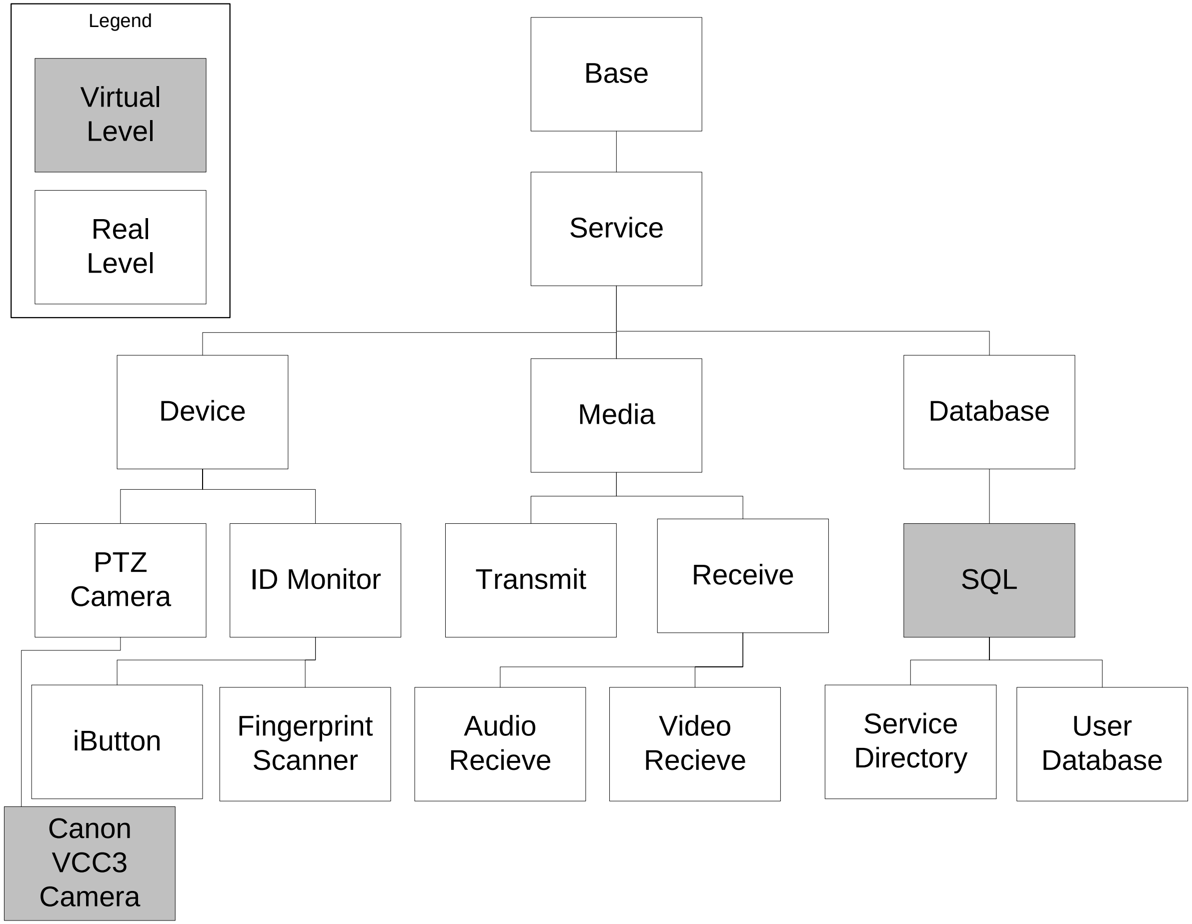
User Identification

- Users are identified by their public keys
- Certificate Authority (CA) for Key validation
  - Keys can be revoked using RCL functionality
- Keys are x509 certificates containing the public key and the CA’s signature
  - Remote users could have the same login, but key signed by two different CA
- Keys are used for two purposes
  - Identifying the user for TLS protocol
  - Authorizing the user under Keynote
- Private Keys protected by operating system mechanisms
  - Users are logged into the system using ID Monitors
- Global Users known as “ace” who functions as the root

ID Monitor


ID Monitor
After ID Monitor Has Verified the User, Host App Launcher is contacted to start the requested program
Host App Launcher
User Logs into the System
Keyboard
Starts the process as the user
User Owned Process
Process Uses Services As User Who Owns the Process
Service
User Key Database
Process can access Protected Keys

Service Hierarchy



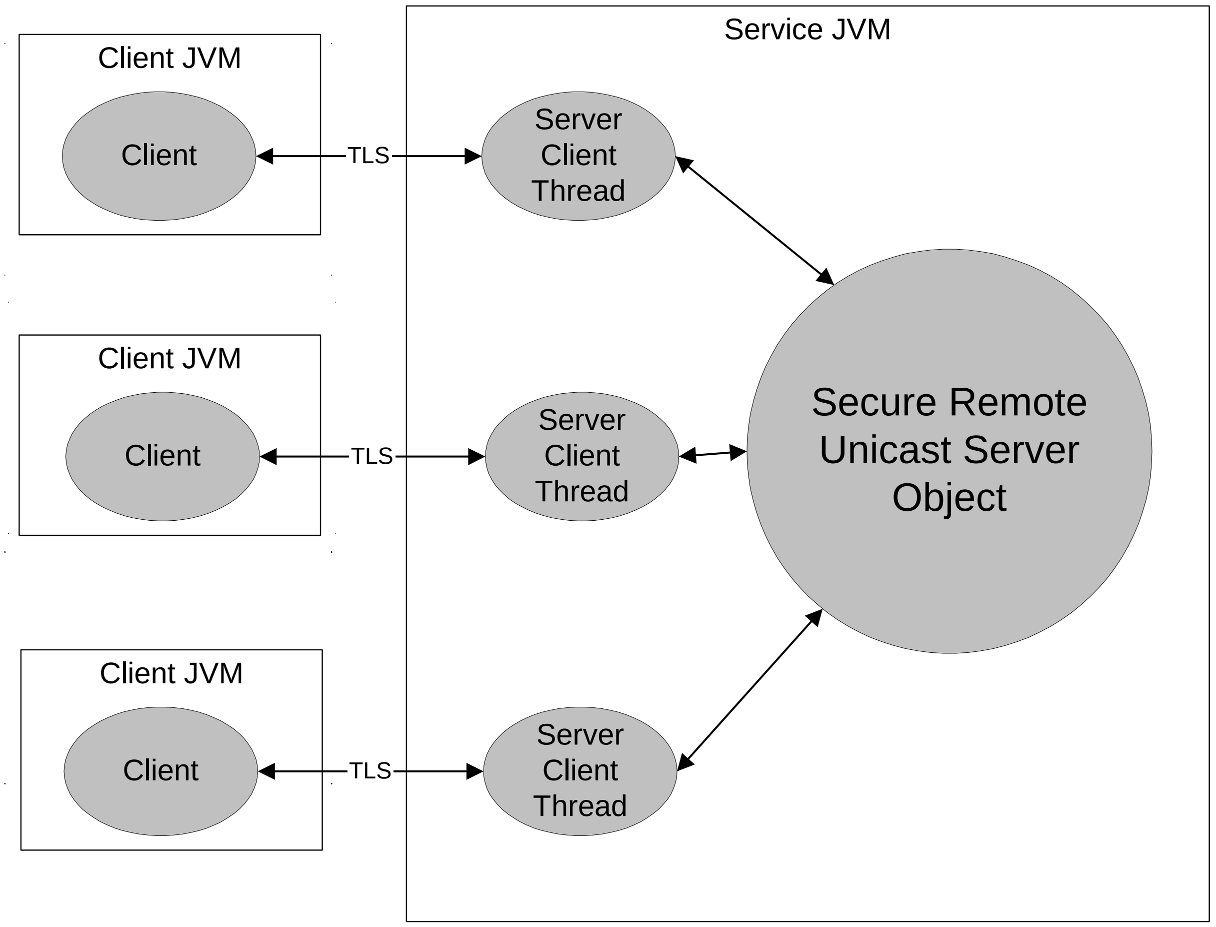
Enhanced RMI

- Used to implement the Control Channels
- Improvement over standard RMI
  - Uses standard RMI tools like rmiregistry
- Stateful connections
- Per-user/per-method security
- Transport Layer Security (TLS)
  - Transmit protection
  - Authentication
- Keynote for Authorization

Enhanced RMI Objects

- SecureRemoteUnicastObject
  - Replaces java.rmi.server.RemoteUnicastServer
  - Extended by other classes to implement Enhanced RMI
  - Thread handles new connections
- ClientServerThreads
  - One thread per client
  - Holds the reference to the current user and keynote session
- java.rmi.Remote
  - Interface that the RemoteObjects extend
- Stub
  - Generated from the implementation by StubGenerator
  - Passed to the client to access the remote service
- Communicates via Messages
  - Serialized objects containing method signature, arguments and/or returns

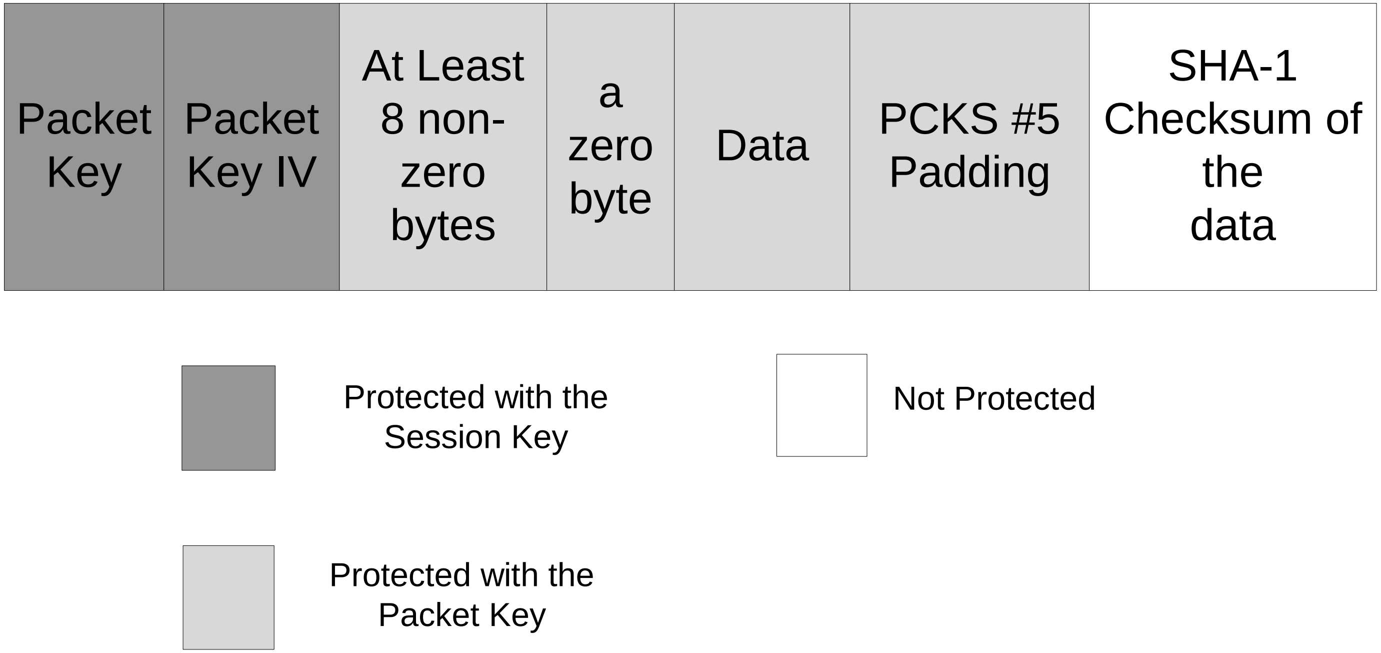
Enhanced RMI



Encryption

- Control Channels use TLS
  - Authenticates the remote user
  - Encrypts the data in transit
- Media Channels use a symmetric AES cipher and SHA-1 to test for data errors
- Keyed with a session key that is used to decrypt the packet key

Media Packet



Packet Key	Packet Key IV	At Least 8 non-zero bytes	a zero byte	Data	PCKS #5 Padding	SHA-1 Checksum of the data

▓ Protected with the Session Key

□ Not Protected

░ Protected with the Packet Key

Keynote Trust-Management

- Trust-Management contains
  - Language for describing actions
  - System for identifying principles
  - A language for describing which actions the principles can perform
  - A method for the principles to pass their authorizations to other principles.
  - A compliance checker for the above requirements
- Keynote uses the same language
  - Describing actions
  - Describing which actions a user can perform
  - Passing authorizations to other users
- Used the Keynote Implementation provided by Univ. of Pennsylvania

Keynote Compliance Checker

- Policy Assertions
  - Base of the tree
  - Identified by the policy authorizer
- Credential Assertions
  - Signed assertions
  - Can be added at any time
  - Allow for permissions to be passed from one service to another
- Conditions
  - Variable/Value pairs
  - Checked by Equality, Simple Math, or regex expressions
- Compliance Checker
  - Breadth first search for highest permission level
  - Starts from Policy assertions

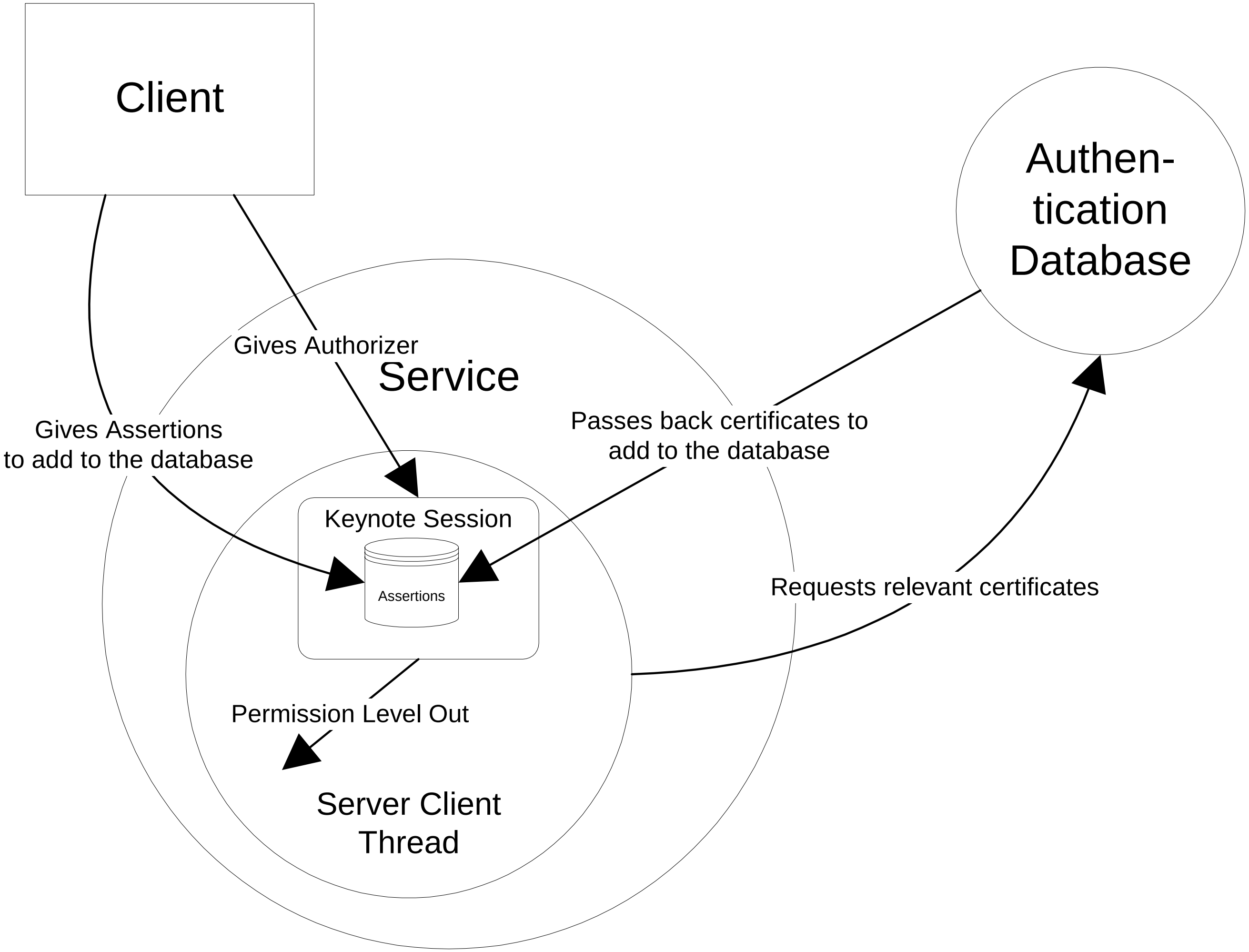
Keynote Assertion

```
keynote-version: 2
authorizer: "x509-base64:MIIEZzCCA9CgAw...LCSG0N2ICh"
local-constants: KEY1 = "x509-base64:MIIE...ighRT4523k"
licensees: KEY1
conditions: ((APP_DOMAIN == "ACE") &&
   (time >= 1082390980610) &&
   (time <= 1082390980628)) -> "write";
       ((APP_DOMAIN == "ACE") -> "read";
signature: "sig-rsa-sha1-base64:Nt4+XIP...soP+mgjjTXWA=="
```

Permission Levels and Conditions

- Levels
  - no_access
  - read
  - write
  - administrator
- Default Conditions
  - Time
  - Method Name
  - Service Type
  - Room
  - Other conditions can be added by the services as needed.

Keynote Operations



Future Work

- XML-RPC for the messages
- Secure Network File System
  - Access to keys
  - Secure Long Term Storage
- Visual and Audio Logins
  - How to tell one user from another
  - How to tell when a user has "logged out"