Communities of Interest
Design and Operational Concepts

Kevin Foltz, Coimbatore Chandersekaran
June 14, 2005
Agenda

- Introduction to Computer Collaboration
- Models of Dynamic Collaboration
- The COI
- Creating a COI
- Policies
- Usage Model
- COIs vs. Domains
- Future Work
Intro to Computer Collaboration

- **Scenario:** Several people want to work together by sharing resources. (Documents, spreadsheets, databases, applications, etc.)
  - Trivial solution: email files they want to share back and forth.

- **Problem:** Limited shared resources, serial sharing, no version control.
  - **Solution:** Create servers to hold shared resources (on user machines or separate servers)

- **Problem:** How do people know how to find servers / shared resources?
  - **Solution:** Use a discovery service

- **Problem:** Can’t control who is able to access resources
  - **Solution:** Partition, divide world into “members”, “non-members”

- **Problem:** How do we know which members and objects are legitimate?
  - **Solution:** Use credential-based naming system, enforce during authentication
Computer Collaboration

- Problem: Not all members get equal access to shared resources
  - Solution: Use authorization system

- Problem: How do we set and enforce collaboration rules and prevent unauthorized access?
  - Solution: Define security policies

- Problem: How can the above be grouped into an independent entity?
  - Solution: Define an encapsulated set of users, machines, and resources—call it a domain.

- Problem: How is the domain managed and organized?
  - Solution: Designate domain administrators and create central servers to oversee domain (“domain controllers”)
A Typical Domain

Security Manager

Domain Controller

Discovery Server

Service Providers

Clients, Users, Requestors

Clients, Users, Requestors
Scenario

- Objective: Find terrorist leader known to be hiding in Eastern Europe
- Collaborators: Selected officials from law enforcement and intelligence agencies in U.S and select E. European countries (e.g. Poland, Ukraine, Belarus), as well as European-wide agencies such as Interpol

- Operation Requirements:
  - Collaborators gather data through human and electronic means (i.e. sensors) and combine to analyze information to pinpoint terrorist’s whereabouts.
  - Mission is time-sensitive—collaboration must begin quickly
  - Officials will continue to work for their original organizations

- Nature of Collaboration:
  - Intelligence agencies share information learned from regional agents
  - Law enforcement agencies make available portions of criminal database, enlist local law enforcement when needed
  - Electronic intelligence shared among all parties
  - Intelligence agencies given limited freedom in all participating nations
Scenario

- Security Requirements
  - Information provided is extremely sensitive
  - Interactions must be strongly authenticated
  - All actions in collaboration fully audited
  - All collected data is confined to the collaborator community
  - Parties are distrustful of one another
  - Collaborators may not share all data collected in pursuing objective
Trivial Solutions

- Sharing
  - Collaborators simply “share” the information/resources needed for the mission to other collaborating members (other nations’ law enforcement/intelligence)

  - This doesn’t work: No trusted authentication mechanism; impossible to implement security policies.

- Mutual Trust
  - Each organization involved in the collaboration set up authentication trust relationships to enable inter-domain access

  - Also won’t work:
    - Trust relationships can take months to implement
    - Mutually suspicious organizations may not want to create trust relationships,
    - Only parts of each organizations are in the collaboration.
A Slightly Better Solution—Lotus QuickPlace (QP)

- A “QuickPlace” is actually a webpage—any community member can create one easily with QP software
- Users are added with roles (Manager, Author and Reader)
- Quick creation, addition of members (via email)
- Resources are various individual files (extra support for MS Office)
- QP also provides “inner rooms” wherein objects can be stored
  - Rooms provide minimal confinement – list of who can enter
  - Once within room access rules apply in review of objects
- Entire community exists on one Domino server

- Disadvantages
  - Resource potential restricted
  - Fixed policies and simple credentials that are quite limited
  - Little isolation—domain administrator has full control
  - All software built as middleware—easy to compromise
Our Solution – Community of Interest (COI)

Domain-like” structures that incorporate users from multiple, unrelated, already-existing organizations.

- Built for a specific mission
- Able to be created in a short time
- Members can operate concurrently in the community and in their original organizations
- Resources include files (documents, folders), applications, databases, sensor data
- Relatively small in scale

- Security Features
  - Credential-based authentication / authorization/auditing
  - Unique naming / discovery services
  - Security policy model
  - Information confinement
  - Joint administration

(Remember that a domain is the basic form of encapsulated collaboration (users, resources, policies))
Community of Interest

- All members and resources located in a COI domain that operates and is managed separately from all contributing domains.

- Preferred model
  - Members given user authentication
  - Authorization by resource managers or attribute credentials
  - All shared resources are imported into central domain
  - COI confined from all outside domains
    - Special trust relationships possible
COI

Sec Mgr

Sec Manager

Discovery

Service Providers

Clients, Users, Requestors

Disc. Service

Sec Manager

Discovery

Service Providers

Clients, Users, Requestors
Creating a COI

- Some Creation Steps
  - 1. Install Network OS on a server to be domain controller
  - 2. Select name for computer and domain
  - 3. Install directory service
  - 4. Configure local, domain, domain controller policies
    - account rules—password policies, credential life, account duration
    - user rights—ability to modify/shutdown computer, remote access
    - security policies—encryption, digital signatures, access to hardware
  - 5. Set up a credential system (Kerberos or certificate-based)
  - 6. Create an auditing system
  - 7. Create and configure users and groups
    - built-in (administrative levels, credential managers)
    - custom (accountants, programmers)
  - 8. Configure group policies
  - 9. Set up additional domain controllers / replication policies

- Above steps are similar to those in creating a static domain
Creating a COI

- **Specialized COI Creation Steps**
  - Determine what resources each new member will import into the community
  - Setup multiple credentials (smartcard, biometric)
  - Determine methods of import/export
  - Create a mechanism for joint administration
    - Threshold cryptography
    - Consensus mechanisms
  - Set up group-based encryption keys
  - Additional policy configuration (joining/leaving community)
Policies

- Policies = “rules” governing operation of all entities in COI
- Hundreds of policies, some specific to COI
- Some interesting COI policies
  - Joint ownership (threshold keys, consensus models)
  - Group encryption keys
  - Information confinement
  - Multiple credentials
  - Delegation
  - Multiple administrators, reviewers
  - Auditing details
  - Domain services offered
  - Encryption (type, content)
  - Joining/leaving community
    - Mission-critical members might not be allowed to leave
    - Fate of resources contributed by members who later leave
    - Import/Export of resources
Available Tools

- Most domain creation steps are automated
  - Network OS handles basic installation steps well
    - Domain controller installation
    - Discovery services
    - Credential system
    - Naming

- Post-installation tools limited
  - Policy templates
  - New policies needed
  - Joint administration
  - Confinement
  - Group-based management
  - Usage model
Usage Model

- In practice, users belonging to a COI will want to operate in both the community and their original organization.

- Requirements
  - Easy access to resources in both initial organization and COI
  - Resource confinement in each domain
  - Separate identities, credentials
  - Scalability

- Trivial Solutions
  - Users have a single machine, connected to both domains simultaneously
    - Problems:
      - 1. No confinement
      - 2. Policy conflicts, machine resources/data subjected to two sets of policies
      - 3. May be prohibited by OS
  - To switch between a COI and original domain, user logs off one name and logs back on another
    - Problems:
      - 1. Still no policy resolution (ex: encrypted files)
      - 2. Highly secure domains might not prohibit this type of logon
  - User operates separate machines, one for original organization, one for COI
    - 1. Impractical
    - 2. Does not scale

- Issue not unique to the COI model
Usage Model

- One potential solution: Divide a computer into isolated virtual machines (VM), one for each dynamic community.
  - VM’s have separate credentials, domain membership
  - Accessing a COI involves simply switching to a VM
  - VM’s must be securely isolated from one another

- Current commercial software more capable
  - VMWare
    - Independent domain credentials
    - VM’s not independent of OS
    - VM’s not mutually confined
  - VirtualPC
    - Good isolation of domains
    - Potentially good solution to switch between domains
COI vs. Static Domain

- **Limited Mission**
  - Number/type of resources already known
  - Departure of mission-critical member could change COI’s functionality

- **Autonomous Operation**
  - Limited external trust relationships, quick creation
  - More flexibility in policies, credentials, naming, security

- **Relatively Small**
  - Low number of resources
  - Multiple review, group keys

- **Mutually Distrusting Membership**
  - Stringent internal security/confine ment policies
  - Joint administration and ownership
  - Some trust evaluation already done
Future Work

- Policies
  - Develop policy templates for COIs of various security levels
  - Further define policies necessary for COI function

- Clean up Design and Usage Model
  - Fully identify existing tools and new tools that are needed

- Performance Evaluation
  - How long will it actually take to create a COI
  - How dynamic is it?

- Longer-term
  - Identify threats through threat models
  - Validate model with customers
    - Enhance and massage model as necessary
    - Work to implement in requirements