A Framework for Managing Inter-Site Storage Area Networks using Grid Technologies

Fritz McCall - UMIACS

Ben Kobler - NASA GSFC

Mike Smorul - UMIACS

Overview

- NASA Goddard Space Flight Center and the University of Maryland Institute for Advanced Computer Studies are maintaining a shared test-bed in order to evaluate new IP-SAN technologies over metropolitan and wide-are networks.
- We are currently developing a management framework for inter-site Storage Area Networks that span multiple collaborating Institutions using Storage Area Network Routers.

Motivations

- SAN extensions enable us to share large and exponentially growing data sets between geographically distributed data storage systems
- Dynamically establishing SAN extensions on an as-needed basis and then removing them when the application is complete can derive additional security and resource allocation benefits.

Design Overview

- Dynamic allocation of SAN extensions requires a framework for distributed administration.
- We need mechanisms that manage trust, limit risk, and delegate authority.
- Sites must retain full control of their local resources while delegating specific administrative functions to peers at remote sites.

Challenges

- Administration of SAN extensions requires a lot of coordination.
- No cross-site authentication mechanism
- Limited authorization levels: typically just read-only and read/write.
- No repository of administrative information between sites.

Our Approach

- Apply Grid and Web Service technologies to help demonstrate:
  - methods for strong authentication and fine-grained authorization
  - a software environment in which local sites can retain control of their own resources even while delegating some administrative tasks to remote collaborators,
  - tools to reduce the level of inter-site coordination and administrative intervention required to manage an IP SAN that operates between collaborating but independent organizations.

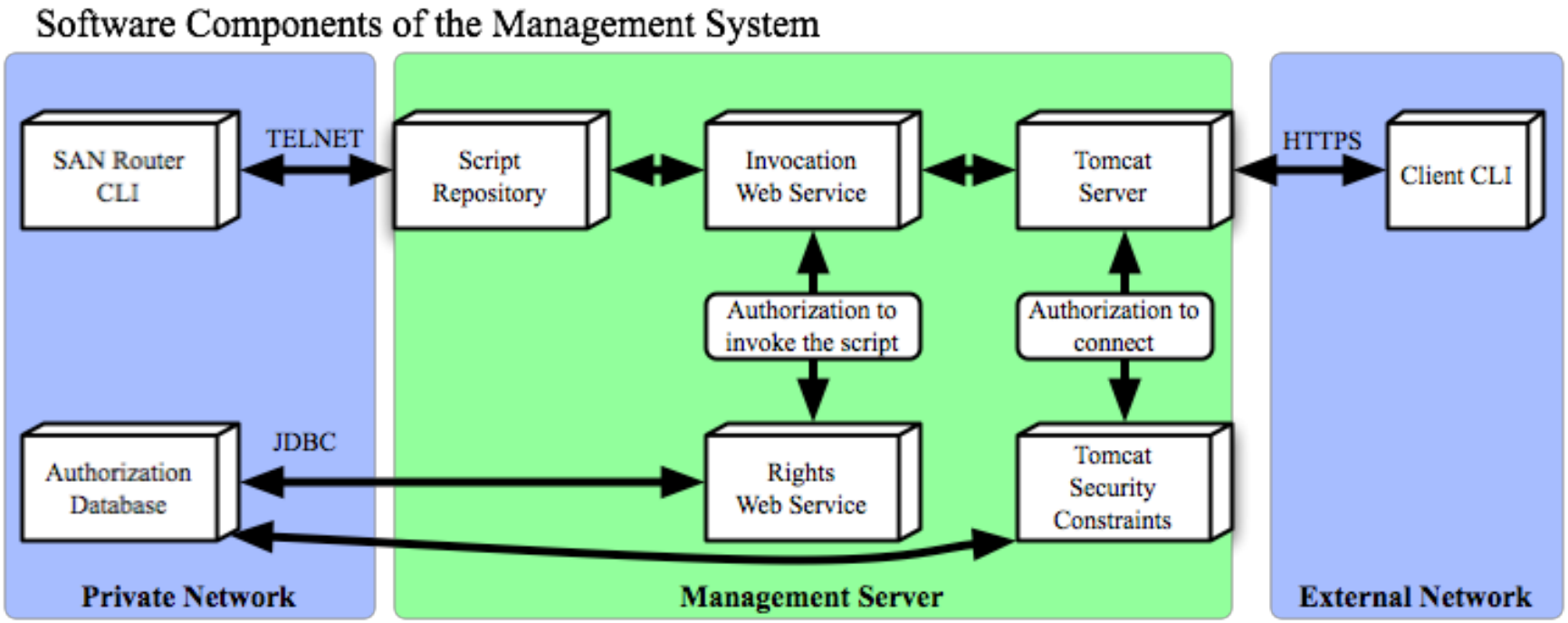
Prototype Architecture

- Each site runs a management server that hosts two secure web services:
  - An invocation service that allows authenticated users to invoke trusted administrative scripts for the purpose of setting up or tearing down a SAN extension.
  - A rights service that applies fine-grained authorization policies to allow or deny authenticated user requests.
- These web services make up the distributed administration interface.

Prototype Architecture (cont.)

- The management server's remote procedure calls are secured as follows:
  - All client-server communications are encrypted using the Secure Sockets Layer.
  - Both Web Services are secured according to the OASIS Web Security Specification
  - A Public Key Infrastructure identifies participating users, sites, and services with X.509 ceritficates.

Prototype Architecture (cont.)



Functionality

- Our system allows users to remotely execute shared administrative scripts through a command-line interface for the purpose of configuring an inter-site IP SAN.
- All requests are authenticated through a PKI and authorized through the Rights Web Service.

Functionality (cont.)

- Administrative scripts can be implemented in any language supported by the management server's operating system.
- We provide an interface that allows each site to independently define what administrative scripts they would like to share with specific remote users.

Management Server

- The management server is the core of our system:
  - handling all of the inter-site management requests through its secure web services,
  - managing the authorization databases,
  - Executing administrative scripts on behalf of remote users

Management Server (cont.)

- The management server is
  - Built on an SSL-enabled Apache Tomcat Server with Java 1.5 and the Unlimited Strength Cryptography Extensions from Sun Microsystems.
  - Configured to support mutually authenticated SSL connections through HTTPS.
  - Configured with security constraints that limit access to trusted users with valid certificates and entries in the site's authorization database.

Management Server (cont.)

- We use two add-on software packages to support the secure web services:
  - Apache Axis: a SOAP implementation that serves as the messaging layer for our web services.
  - Apache WSS4J: a WS-Security implementation that provides message level security for our web services.

Management Server (cont.)

- The management server hosts a MySQL RDBMS as an authorization database containing tables of:
  - Authorized users represented as X.509 Subject Identifiers
  - Permissions that grant or deny those users access to specific administrative scripts
- These tables are used by both Tomcat and the Rights Web Service.

Invocation Web Service

- The Invocation Service provides
  - a programmatic interface for our system's distributed management capabilities.
  - Functions for browsing and invoking available administrative scripts.
  - Managerial Functions that allow each site to register scripts that they wish to share.

Invocation Service (cont.)

- The Invocation Service processes requests as follows:
  - Receives the request as a SOAP message over an SSL connection
  - Ensures message integrity with WSS4J
  - Authorizes the request through the rights service
  - Check the user-specified arguments and inputs for disallowed characters or malicious input
  - Executes the scripts, returning output and error to the remote user.

Script Repository

- Our prototype administrative scripts are written in Expect, which is:
  - An extension of tcl that is frequently used by systems administrators to automate complex configuration tasks through terminal interfaces.
  - A convenient method of interfacing with the McDATA SAN router's command line interface through the telnet protocol.

Script Repository (cont.)

- Careful development of administrative scripts is an important part of the system's security:
  - Particularly vulnerable to input injection attacks if they are not carefully written
  - Granted administrative privilege on the SAN Routers
  - Important that each site review and customize the scripts.

Rights Service

- Determine which remote users are authorized to access each administrative script. It processes requests as follows:
  - Receives the request as a SOAP message over an SSL connection and validates the message
  - Checks that the client certificate is not listed in the certificate revocation list
  - Checks that the client certificate's Common Name is listed as a trusted user in our authorization database
  - Ensures that the users is authorized to invoke the requested script.

Classads Registry

- We are also developing a central registry of administrative information about the distributed SAN based on the Classified Advertisements (classads) library developed at the University of Wisconsin.
- They can allow administrators at the invocation site to accurately describe their shared resources and the policy requirements that govern their use.

Preparing Local SAN Environments

- Partition shared SAN devices from private SAN devices through physical reconfiguration, Zoning, or LUN masking so that SAN-attached devices can never be accessed by remote sites unless they are manually configured onto the shared SAN zone.

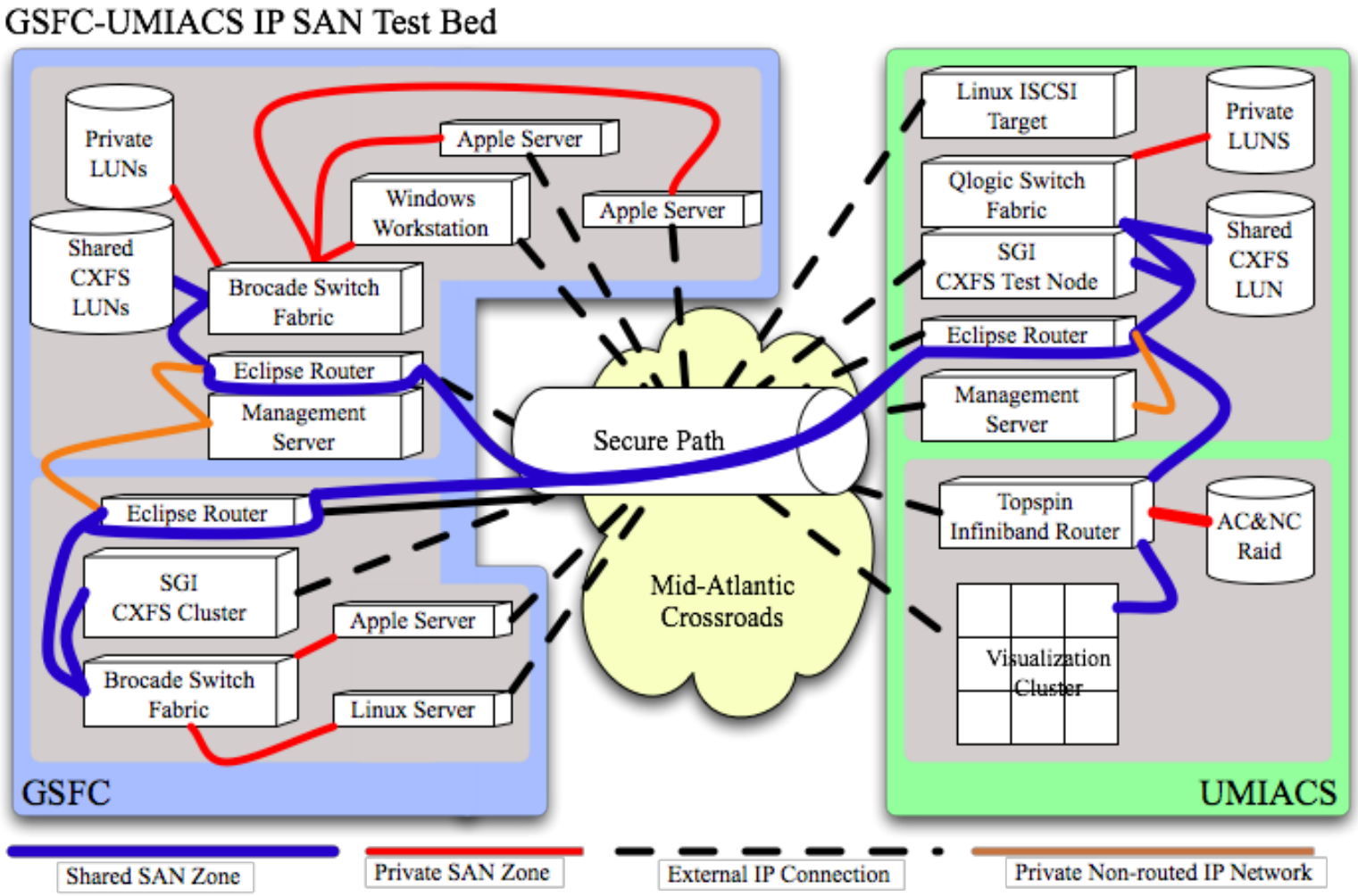
Configuring the SAN Routers

- The SAN Routers need to be configured for basic connectivity and functionality with:
  - A public Gigabit Ethernet connection that will communicate with other storage routers to move data across the inter-site SAN
  - A private Fast Ethernet that needs to be isolated and protected from untrusted networks so that we can use it to safely administer the SAN Router using unencrypted protocols like Telnet or SNMP
  - A unique mSAN identifier for each Storage Router and a unique Zone identifier for each shared SAN zone

Configuring the Management Server

- The management server needs to be configured with:
  - a public Ethernet interface that remote users can access
  - a connection to the SAN router's private management network so that it can access the SAN router's command-line interface
  - the router's login username and password

Current Test-bed



Extensibility

- The need for distributed administration arises in a number of applications that our framework could support because the underlying scripts are intended to be highly flexible.
- Our use of web services makes it very easy to develop and customize a variety of client interfaces.

Future Directions

- Migrate to GT4 to get advantages of credential delegation and SAML assertions
- Refine the script environment to provide greater control of execution, inputs, and outputs.
- Develop the management server as a Virtual Machine (Xen or VMware GSX) in order to make it very easy to install.

Acknowledgements

- The Mid-Atlantic Crossroads which is instrumental to our collaboration and our programs.
- All of the engineers at NASA GSFC and UMD who make these evaluations possible, especially Hoot Thompson, Gary Jackson, Pat Gary, Paul Lang, Bill Fink, Mike Van Opstal, and Steve Willett.
- All of you for attending MSST06.