GRID SECURITY

Tran Ngoc Cuong	11076009
Nguyen Huynh	11076030

OUTLINES

1. Introduction of basic security
2. Grid security
3. Some current security standard
4. Some implementations on Grid
5. References

INTRODUCTION OF BASIC SECURITY

What is security?

IT security is concerned with ensuring that critical information and the associated infrastructures are not compromised or put at risk by external agents.

Goals of security

- Prevention
- Detection
- Recovery

Security concerns for DATA

- Confidentiality
- Integrity
- Availability

Other security concerns

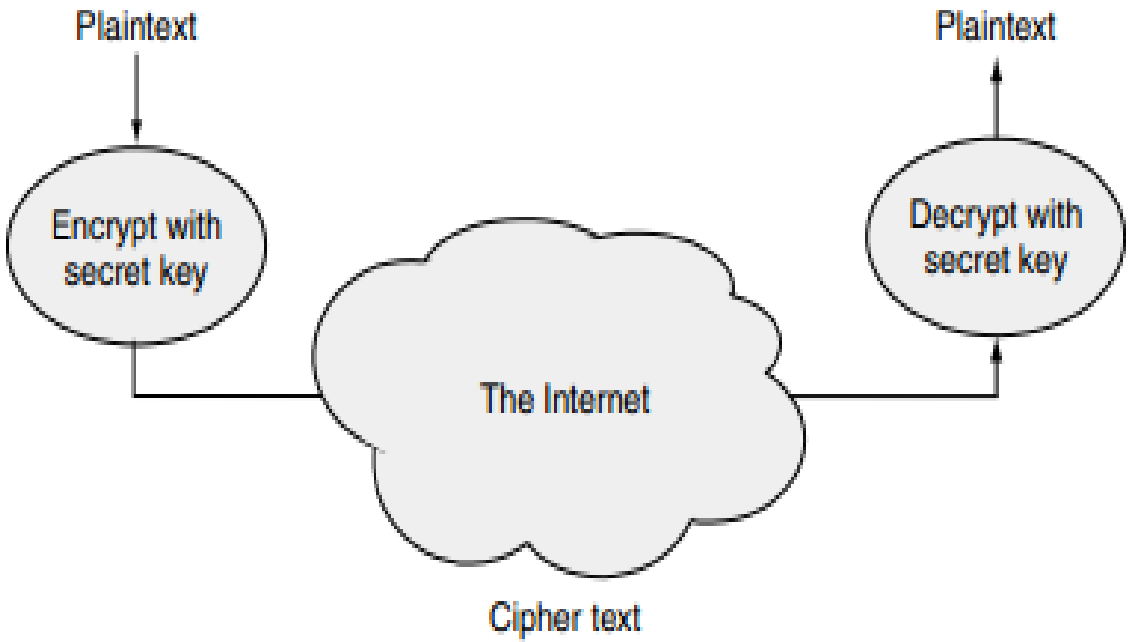
- Authentication
- Authorization
- Assurance
- Non-repudiation
- Auditability
- Trust
- Reliability
- Privacy

Cryptography

Cryptography is the most commonly used means of providing security, it can be used to address four goals:

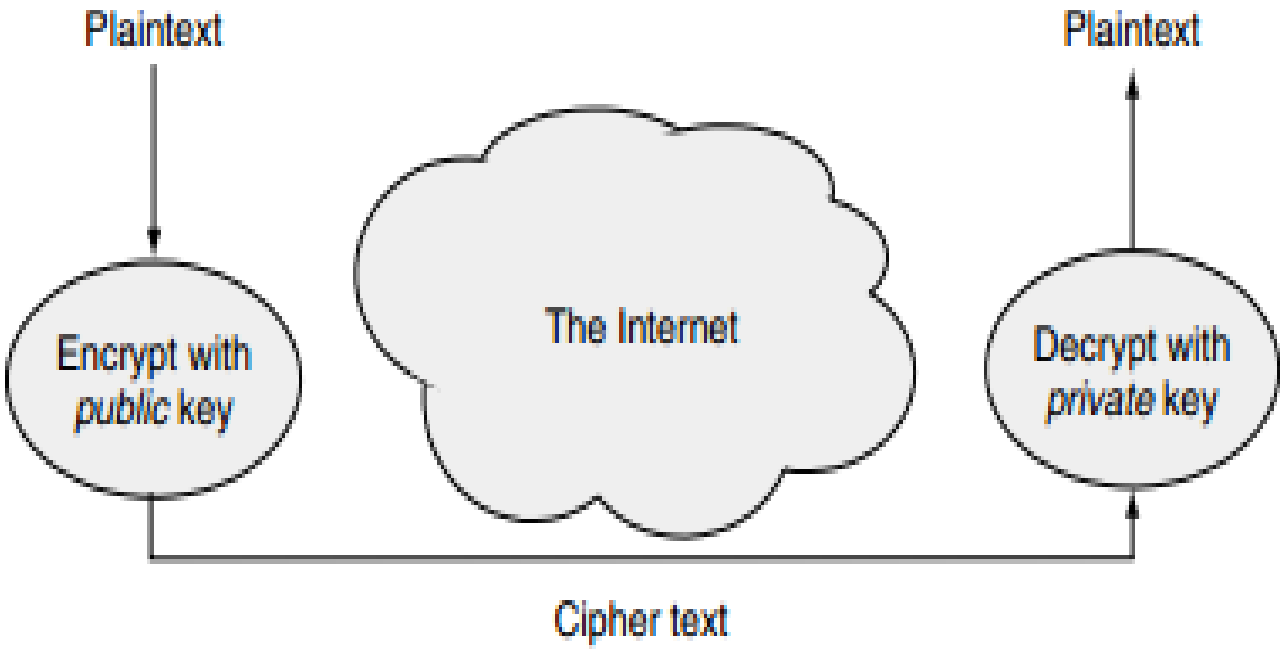
- Message confidentiality
- Message integrity
- Sender authentication
- Sender non-repudiation

SYMMETRIC CRYPTOSYSTEMS



Symmetric key cryptography

Asymmetric cryptosystems



Asymmetric key cryptography

Cryptography components (1/2)

- Digital signature
- Public-key certificate:
  - ITU-T X.509 format:
    - Subject
    - Subject's public key
    - Issuer's subject
    - Digital signature

Cryptography components (2/2)

- Certificate Authority (CA)
- Firewall

GRID SECURITY

Grid security requirements (1/5)

- The dissemination, processing, sharing, and virtualization of data, as well as the sharing and virtualization of compute resources, networks, and experiments, lead to challenging requirements for storage, network bandwidth, and compute power.
- The associated security requirements are equally challenging

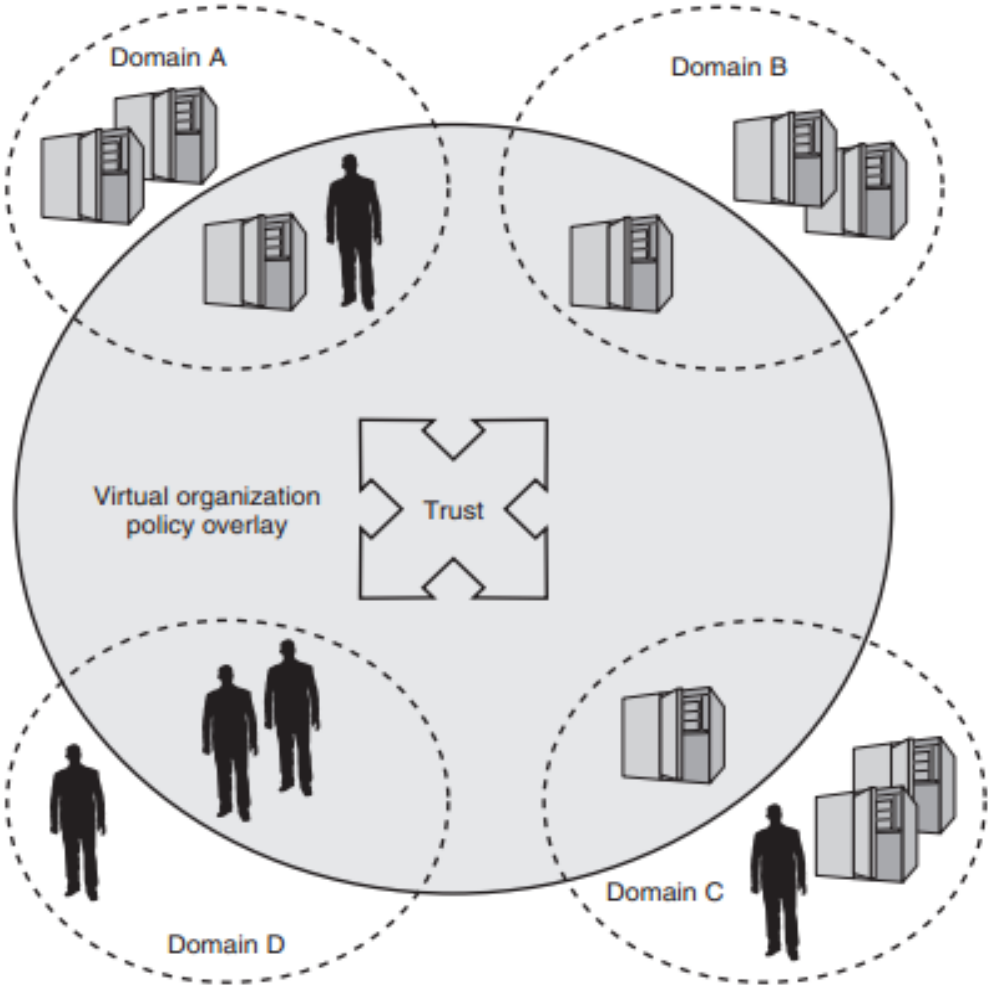
Grid security requirements (2/5)

- Data will move through, and be accessed from, many different centers in different countries with different security mechanisms and policies in place at each center
- The community requiring access to the data spans multiple organizations and countries. Thus, center administrators need the ability to enforce policy without knowing the individuals that access their resources

Grid security requirements (3/5)

- Trust must be established and expressed between different centers, from which remote access policies must be derived
- Data integrity and confidentiality can be crucial

Grid security requirements (4/5)

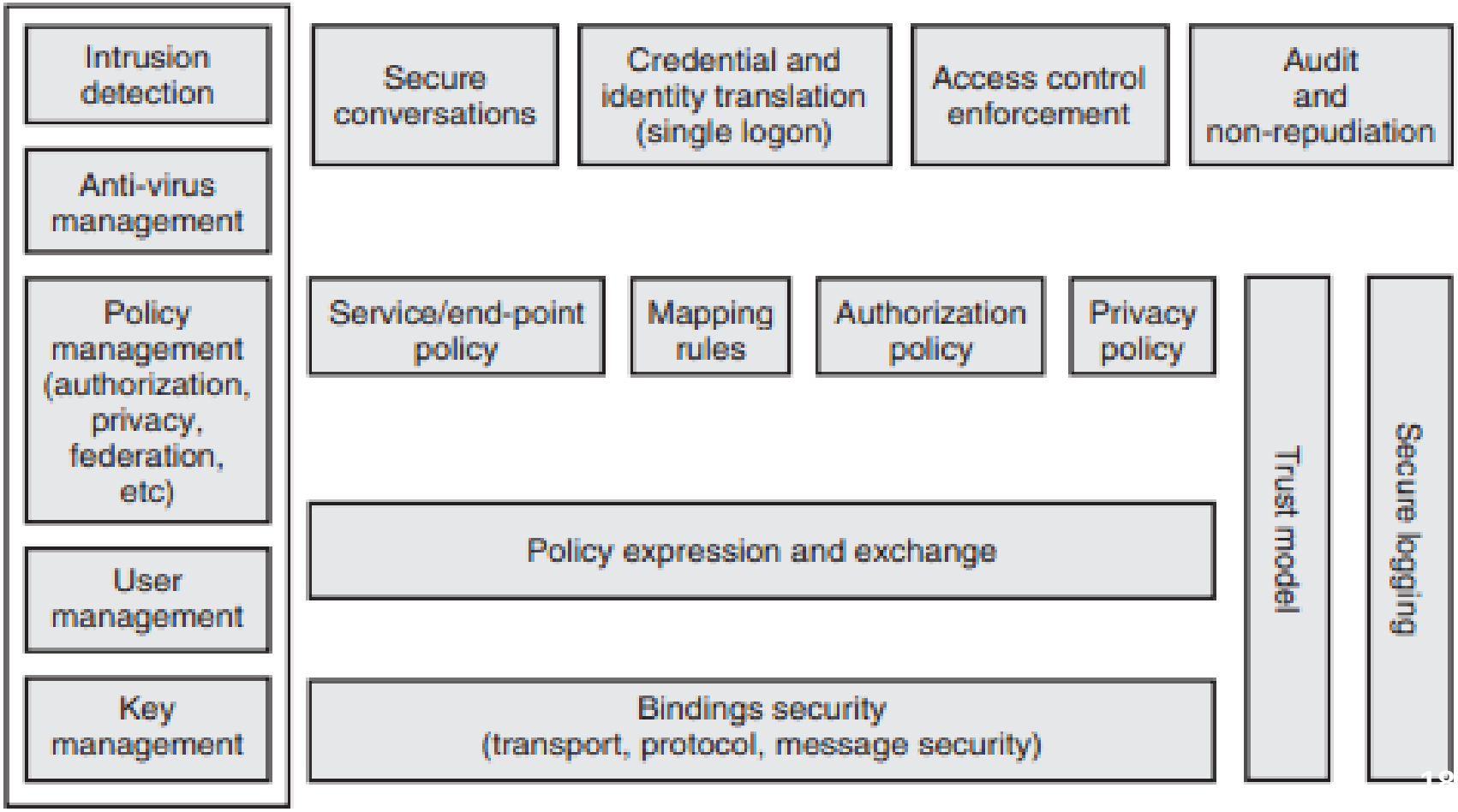


Grid security requirements (5/5)

3 key characteristics in grid security model:

- Enable integration and interoperability
- Enable creation and management of dynamic trust domain
- support dynamic creation of services

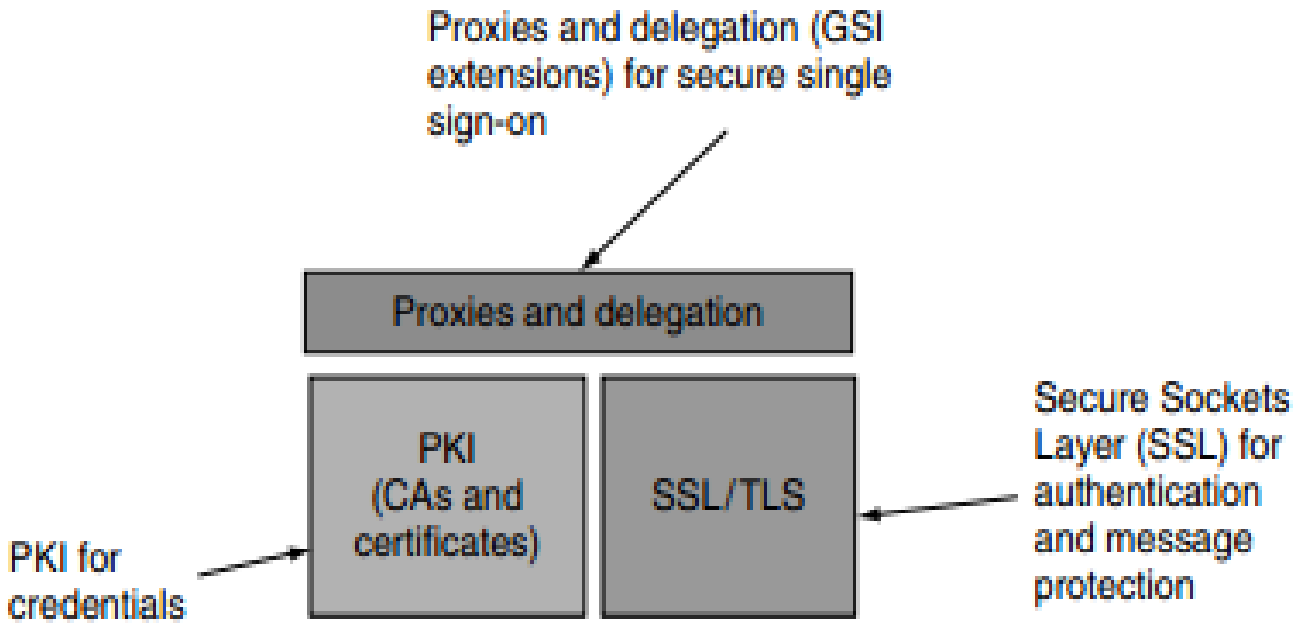
GRID SECURITY MODEL



Components of the Grid security model.

GRID SECURITY INFRASTRUCTURE (1/2)

GSI is an OGSA security reference implementation, and is included as part of Globus Toolkit Version 3



The Grid Security Infrastructure

Grid security infrastructure (2/2)

- A public-key system
- Mutual authentication through digital certificates
- Credential delegation and single sign-on

Authorization modes in GSI

Server-side authorization

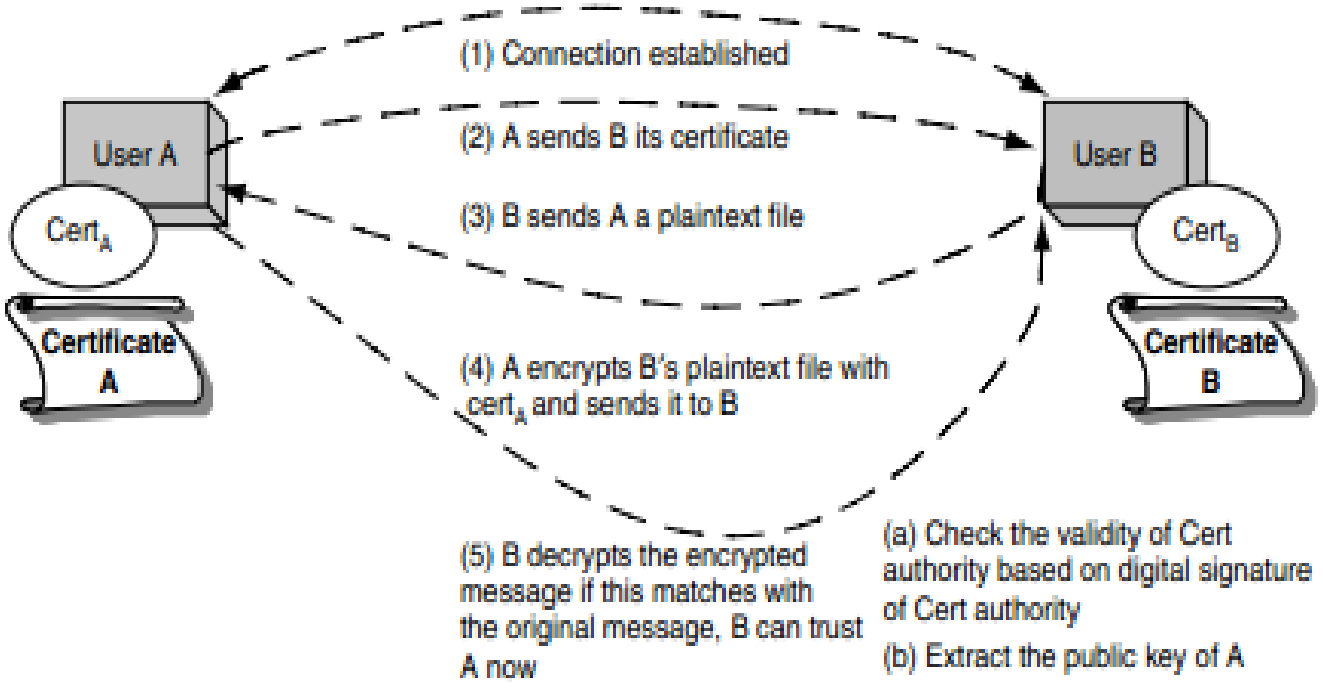
- None
- Self
- Gridmap

Client-side authorization

- None
- Self
- Host

GSI OPERATIONS (1/2)

- Requesting a certificate
- Mutual authorization



Mutual authentication

GSI OPERATIONS (2/2)

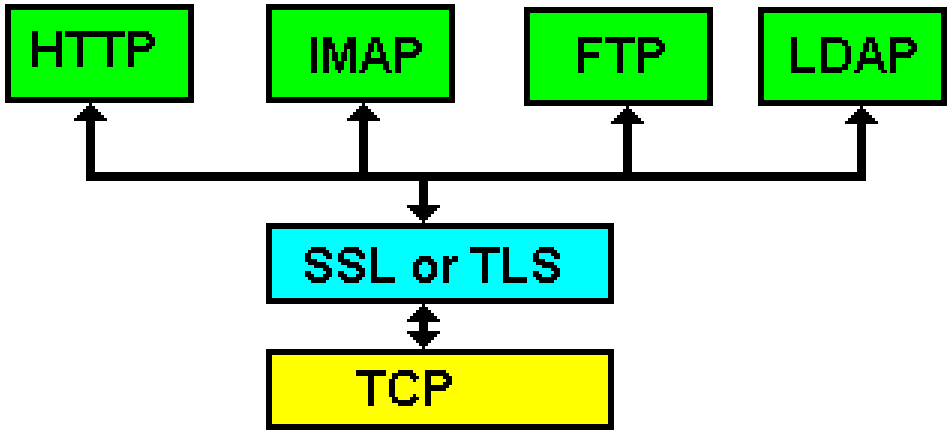
- Confidential communication
- Securing private keys
- Delegation and single sign-on

Some security standards

SSL/TLS

- The major use of SSL (X.509) certificates is with the SSL/TLS protocol.
- Secure Sockets Layer (SSL) is a Netscape protocol originally created in 1992.
- SSL v1: never publish
- SSL v2: 1995, contained a number of security flaws.
- SSL v3: 1996
- TLS 1.0 was first defined in January 1999 as an upgrade to SSL Version 3.0.
- TLS 1.1: in April 2006.
- TLS 1.2: in August 2008.

SSL/TLS



HTTP + SSL/TLS + TCP = HTTPS

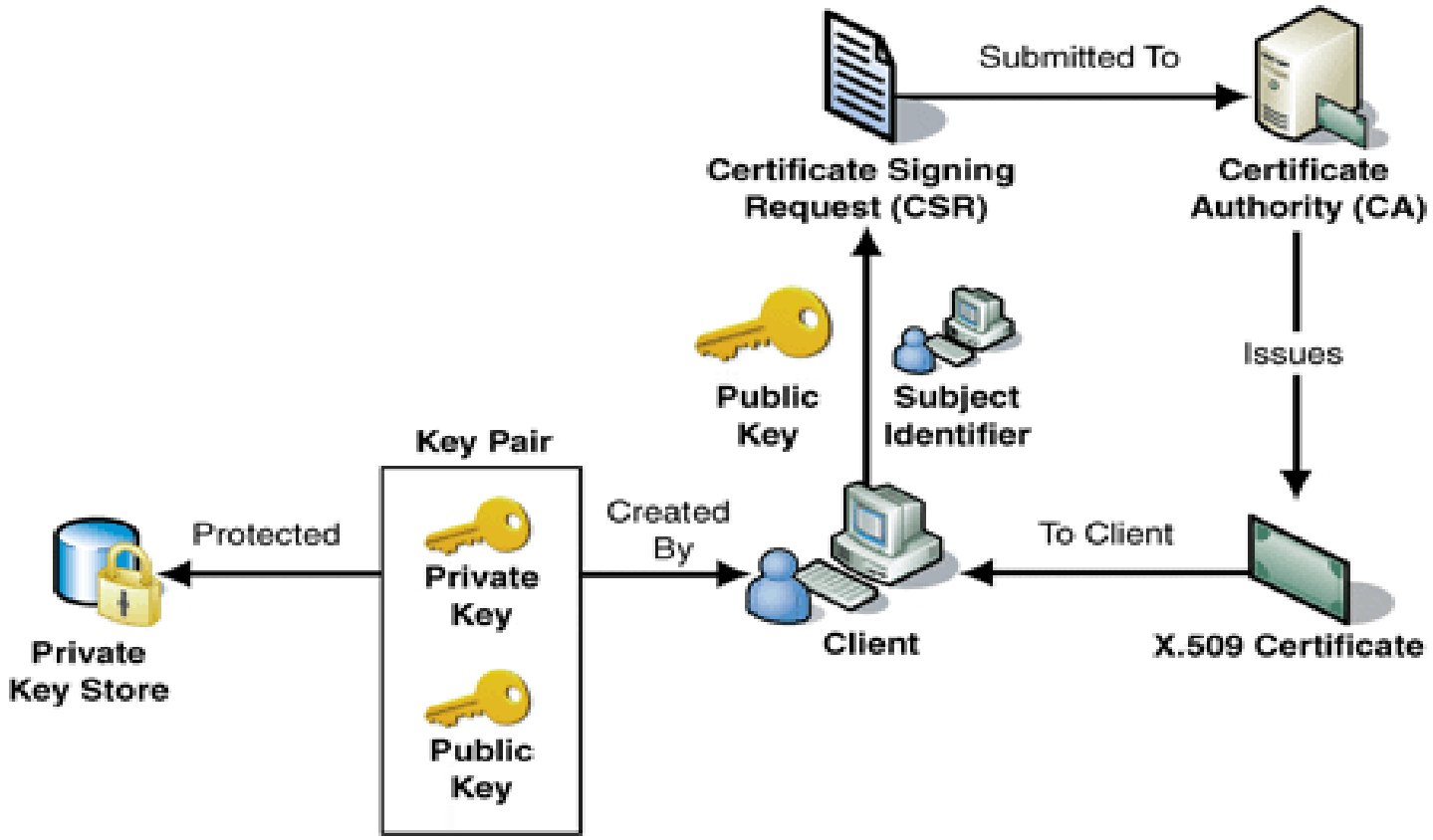
Some security standards

X.509 certificate:

- X.509 is a standard for a public key infrastructure (PKI) and Privilege Management Infrastructure (PMI).
- X.509 specifies standard formats for public key certificates, certificate revocation lists, attribute certificates, and a certification path validation algorithm.

Some security standards

X.509 certificate:



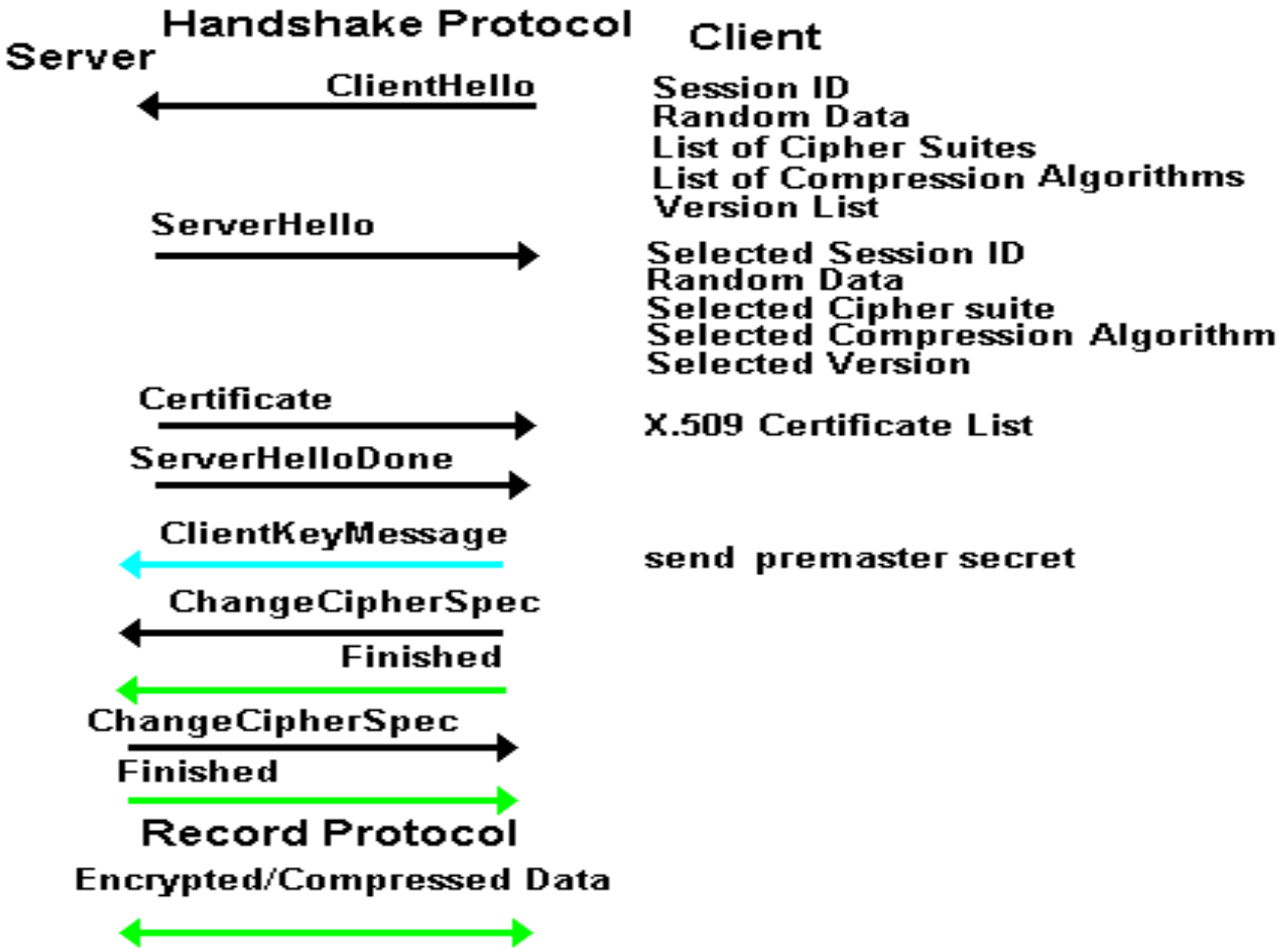
Some security standards

X.509 certificate vs SSL/TLS:

SSL sits on X.509.

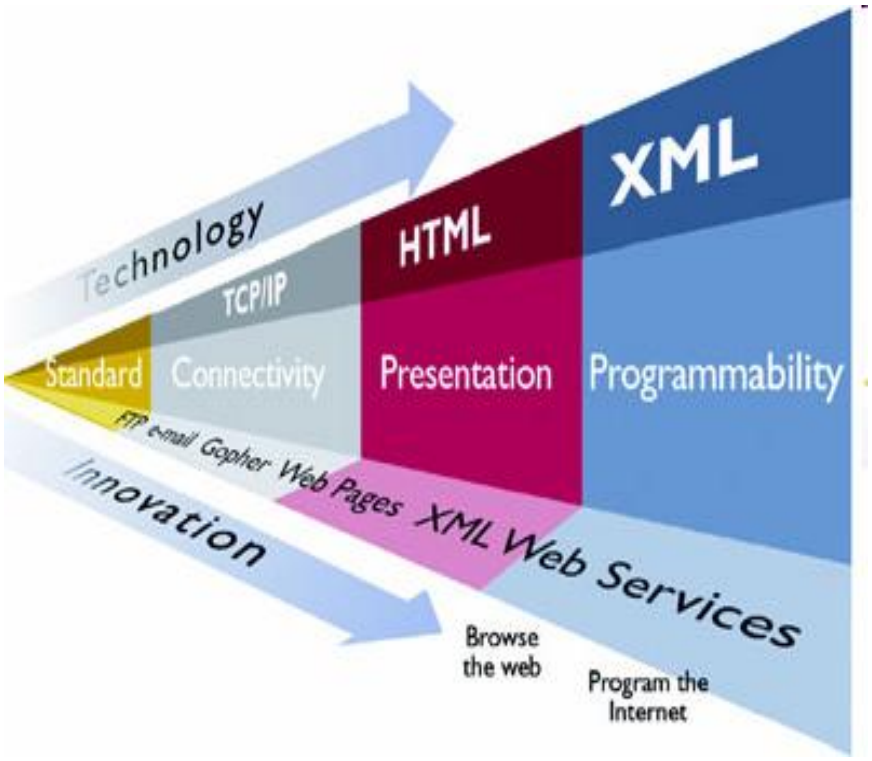
X.509 Specification: Complexity and lack of quality

SSL/TLS


Handshake Protocol
Server
Client
ClientHello
Session ID
Random Data
List of Cipher Suites
List of Compression Algorithms
Version List
ServerHello
Selected Session ID
Random Data
Selected Cipher suite
Selected Compression Algorithm
Selected Version
Certificate
X.509 Certificate List
ServerHelloDone
ClientKeyMessage
send premaster secret
ChangeCipherSpec
Finished
ChangeCipherSpec
Finished
Record Protocol
Encrypted/Compressed Data

Web and web service

- A Web service is a method of communication between two electronic devices over the web (internet).
- RPC, SOA and REST are three most commons style of Webservice.



SOAP, SAML, XML ENC, XML SIG are based on XML

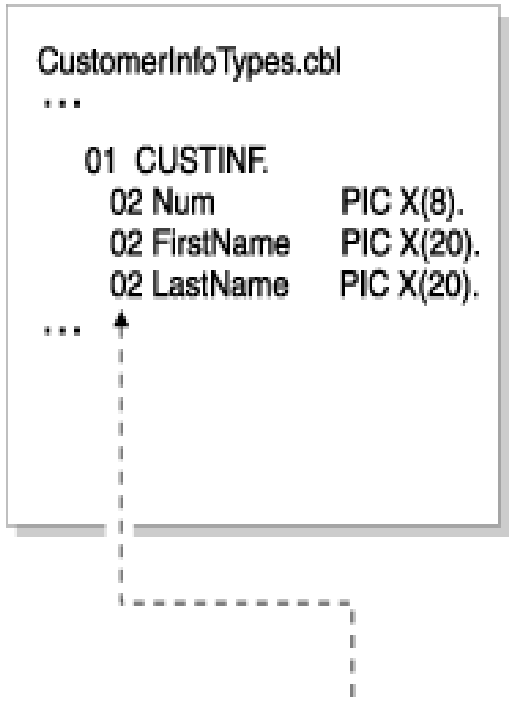
SOAP

```
POST /InStock HTTP/1.1
Host: www.example.org
Content-Type: application/soap+xml; charset=utf-8
Content-Length: 299
SOAPAction: "http://www.w3.org/2003/05/soap-envelope"

<?xml version="1.0"?>
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope">
  <soap:Header>
  soap:Header>
  <soap:Body>
    <m:GetStockPrice xmlns:m="http://www.example.org/stock">
      <m:StockName>IBM</m:StockName>
    </m:GetStockPrice>
  soap:Body>
soap:Envelope>
```

WSDL

```
<?xml version="1.0"?>
<definitionsname="CustomerInfo"
 <types>
  <xsd schema targetNamespace=
     "http://www.customercommandservice.com/CustomerCommand"
     xmlns="http://www.w3.org/1999/XMLSchema/"
    <xsd:complexType name="Customer">
     <xsd:element name="Num"type="xsd:string"/>
     . . .
    </xsd:complexType>
  </xsd schema>
 </types>
 <message name="GetCustomerInfoInput">
    <part name="Customer"type="Customer"/>
 </message>
  . . .
 <portType name="CustomerInfoPortType">
   <operation name="GetCustomerInfo">
     <input message="GetCustomerInfoInput"/>
     <output message="GetCustomerInfoOutput"/>
   </operation>
 </portType>
```



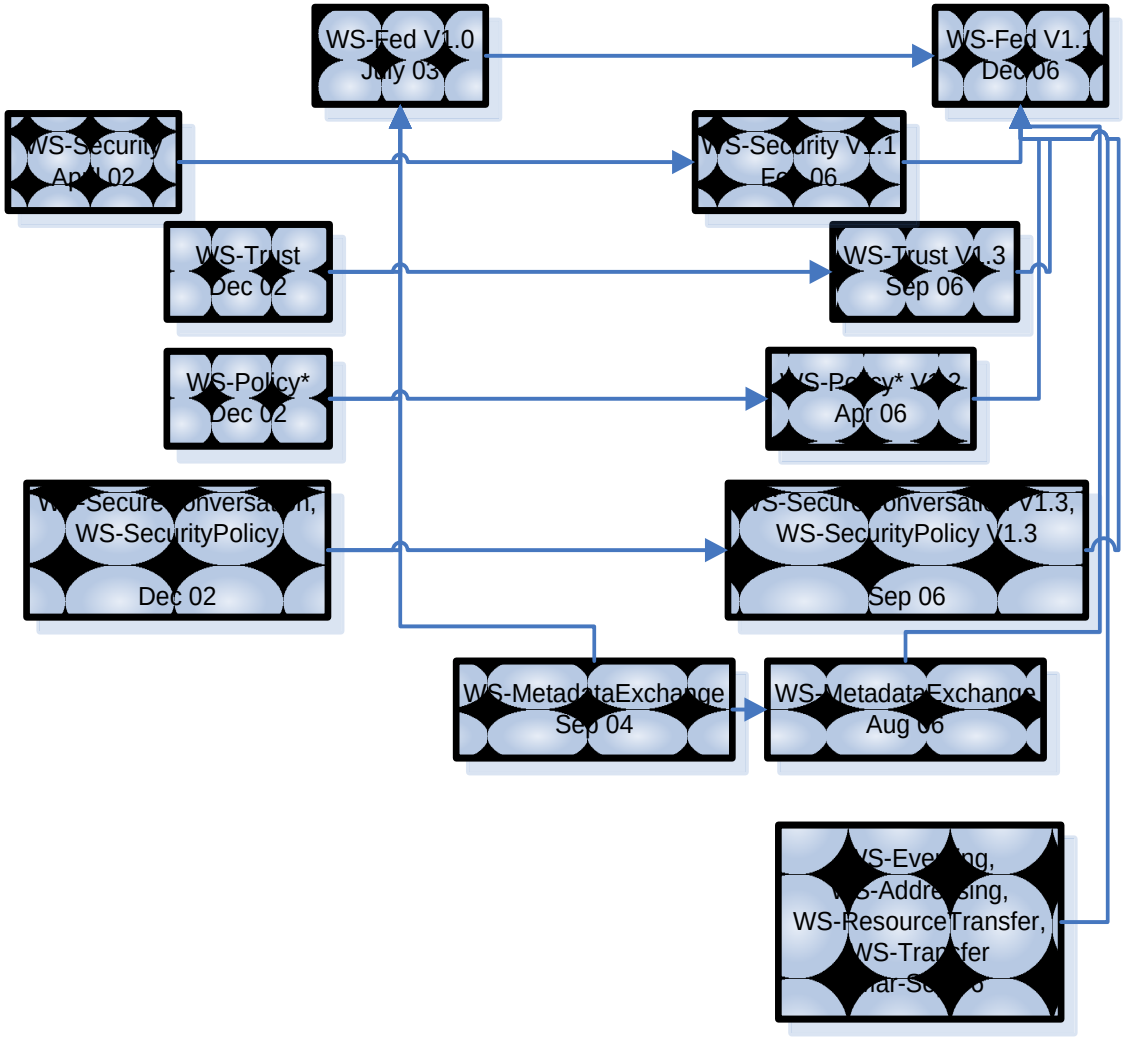
WSDL

```
<binding name="CustomerInfoConnectorBinding"type="CustomerInfoPortType">
    <format:typemapping style="COBOL"encoding="COBOL">
      <format:typemap typename="Customer"formattype="/CustomerInfo.ccp:CUSTINF"/>
    </format:typemapping>
  <operation name="GetCustomerInfo">
    <cics:operation functionName="GETCUST"/>
    <input>
      . . .
    </input>
    <output>
      . . .
    </output>
  </operation>
</binding>
<service name="CustomerServices">
    <port name='CICS_A'binding='CustomreInfoConnectorBinding'>
      <cics:address connectionURL=". . ."serverName="CICS_A"/>
    </port>
</service>
</definition>
```

Security on web service

- **WS-Security** (**Web Services Security, short WSS**) is a flexible and feature-rich extension to SOAP to apply security to web services.
- **WS-SecureConversation** is a Web Services specification, created by IBM and others, that works in conjunction with WS-Security, WS-Trust and WS-Policy to allow the creation and sharing of security contexts.
- And more…

Security on web service



Security on web service

- WS-Security adds significant overhead to SOAP processing due to the increased size of the message on the wire, XML and cryptographic processing, requiring faster CPUs and more memory and bandwidth.

Security Mechanism	Messages/second
WS-Security (X.509) XML Signature & Encryption	352
WS-SecureConversation XML Signature & Encryption	798
Transport Layer Security	2918

WEB SERVICE SECURITY VS SSL/TLS

SSL Provides In-Transit Security Only

Targeted Security

Faster Routing

- **Transport layer**

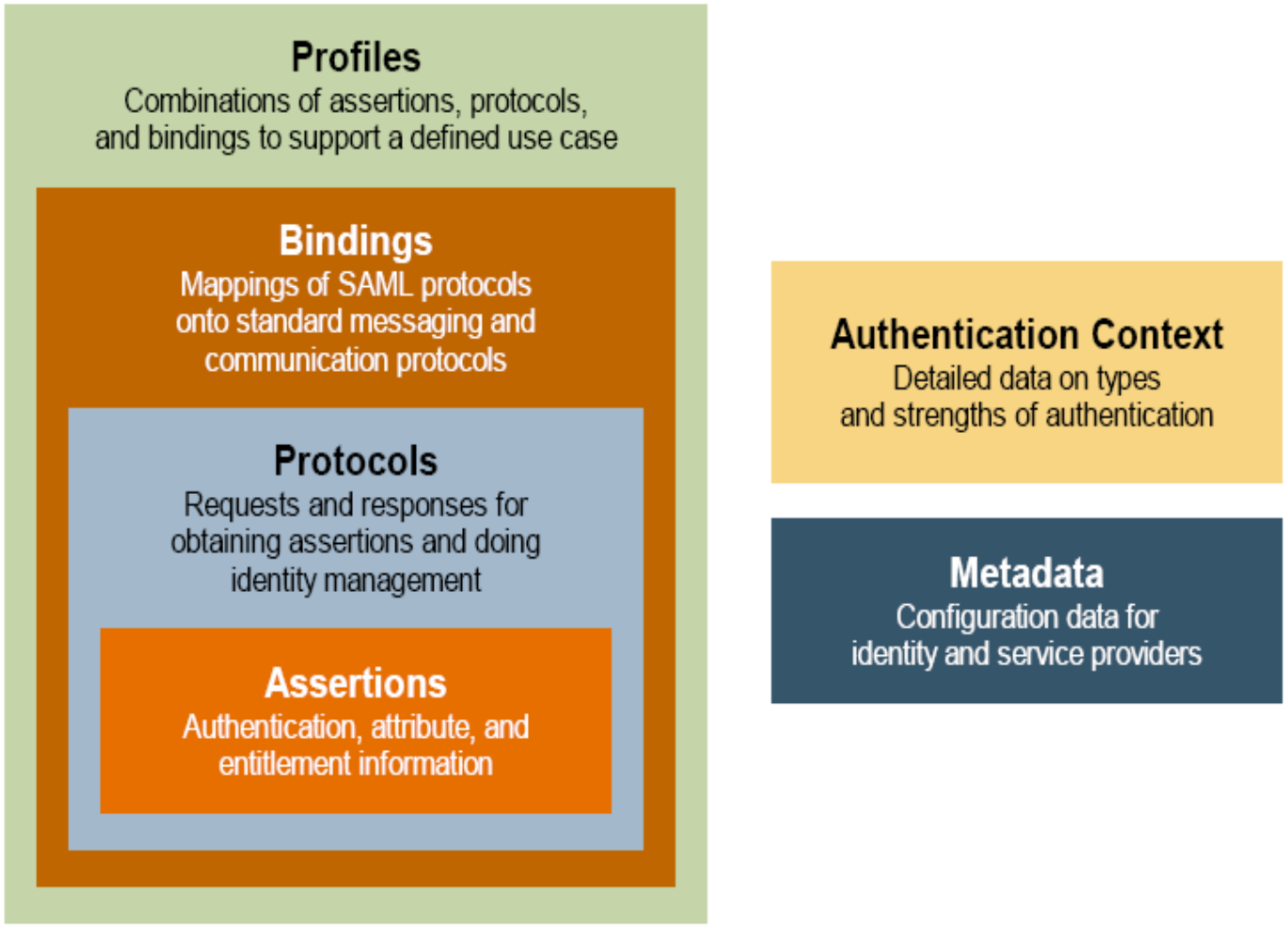
- **Message layer**

SAML

Security Assertion Markup Language (SAML) is an XML-based open standard for exchanging authentication and authorization data between security domains.

SAML is built upon a number of existing standards: XML, XML Schema, XML Signature, XML Encryption, HTTP, SOAP.

SAML


Profiles
Combinations of assertions, protocols, and bindings to support a defined use case
Bindings
Mappings of SAML protocols onto standard messaging and communication protocols
Protocols
Requests and responses for obtaining assertions and doing identity management
Assertions
Authentication, attribute, and entitlement information
Authentication Context
Detailed data on types and strengths of authentication
Metadata
Configuration data for identity and service providers

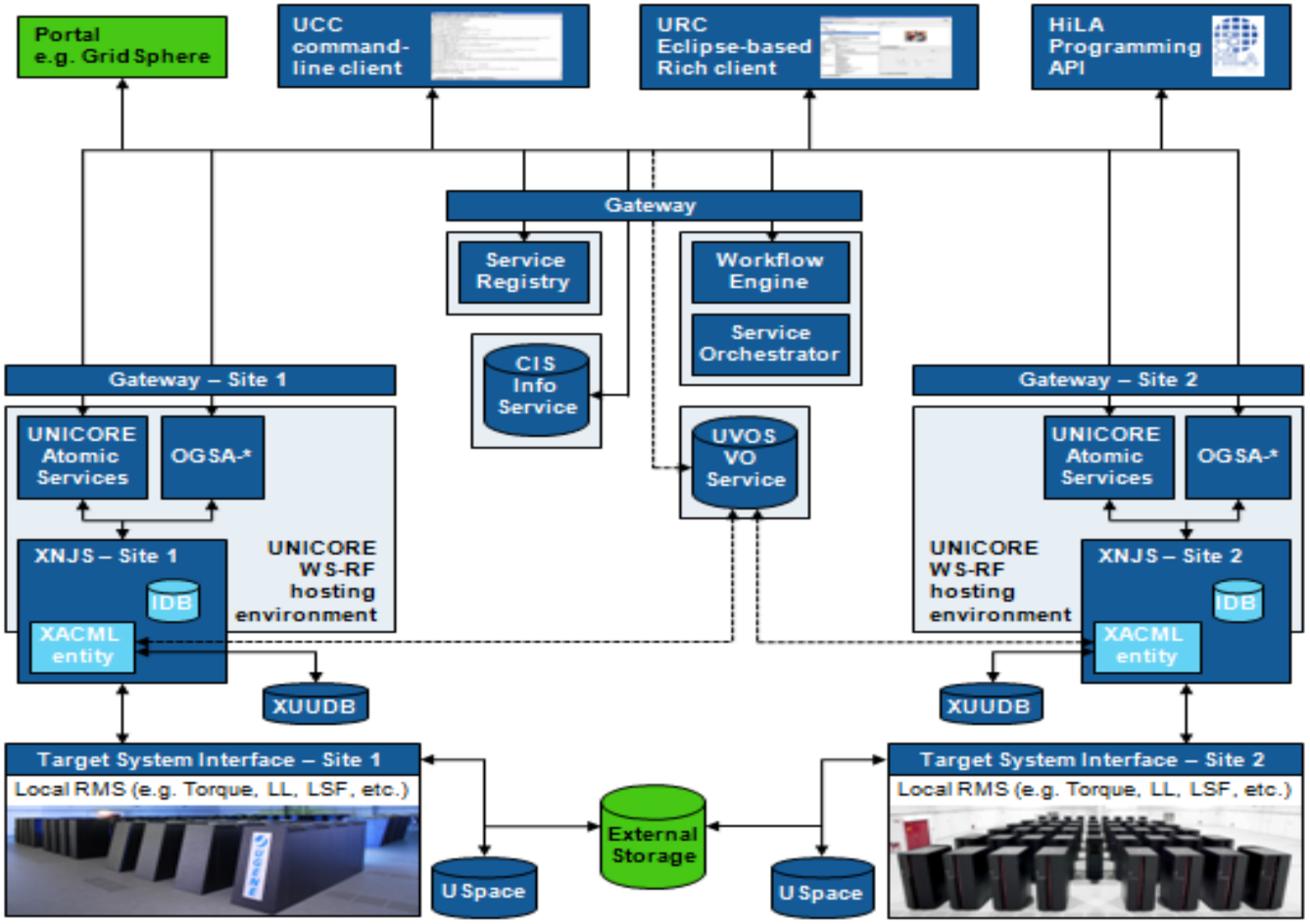
SAML vs WS-*

SAML: Simple, matural, good performance.

WS-*:

- It is too complex
- It is too immature
- Interoperability will be difficult
- It doesn't appear to solve anything that SAML 2.0 and ID-WSF can't already do

UNICORE v6


Portal
e.g. GridSphere
UCC
command-
line client
URC
Eclipse-based
Rich client
HiLA
Programming
API
Gateway
Service
Registry
Workflow
Engine
Service
Orchestrator
CIS
Info
Service
UVOS
VO
Service
Gateway – Site 1
UNICORE
Atomic
Services
OGSA-*
XNJS – Site 1
IDB
XACML
entity
UNICORE
WS-RF
hosting
environment
XUUDB
Gateway – Site 2
UNICORE
Atomic
Services
OGSA-*
XNJS – Site 2
IDB
XACML
entity
UNICORE
WS-RF
hosting
environment
XUUDB
Target System Interface – Site 1
Local RMS (e.g. Torque, LL, LSF, etc.)
Target System Interface – Site 2
Local RMS (e.g. Torque, LL, LSF, etc.)
External
Storage
U Space
U Space

UNICORE v6

Security overview

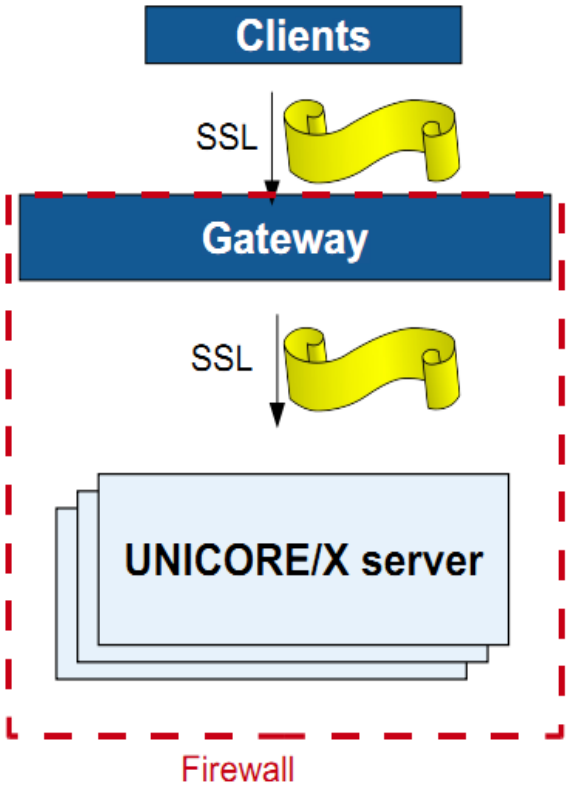
Security based on open standards, XML-based where possible

- X.509 certificates for clients and servers
- Client-authenticated SSL for all client-server and inter-component interaction
- Signed SAML assertions (Security assertion markup language)
  - XML-DSig, Web-services security, SAML v2.0
- Open and flexible security system
  - Authorisation attribute sources: VO server, LDAP, …
  - Optional, limited, proxy support
- Extensible clients

UNICORE v6

Sites are protected by firewalls.

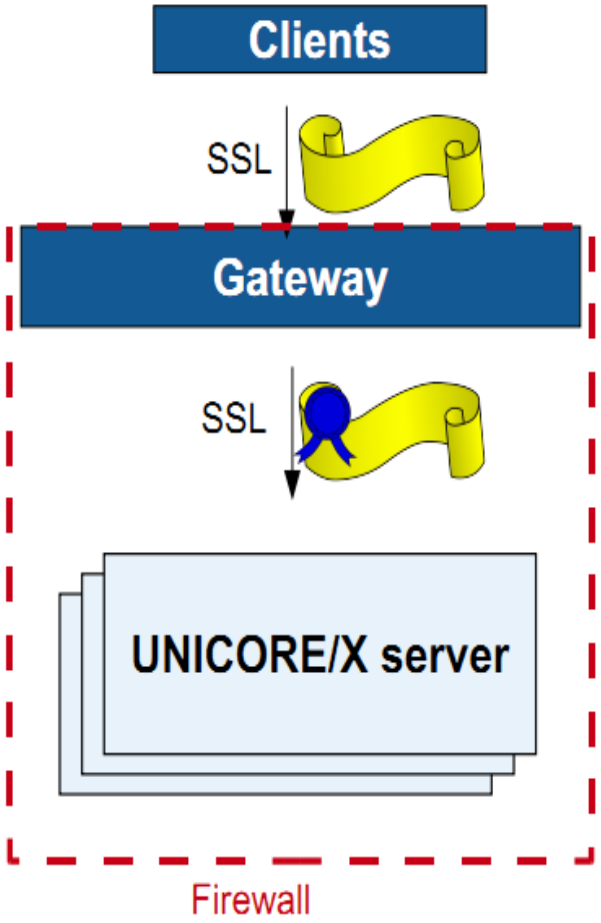
- Gateway provides single firewall entry point
- Client makes client-authenticated SSL connection to the gateway
- Gateway should forward request to the target site
- **How to preserve client certificate information?**
- Proxy based solution not acceptable



UNICORE v6

Solution using SAML assertion

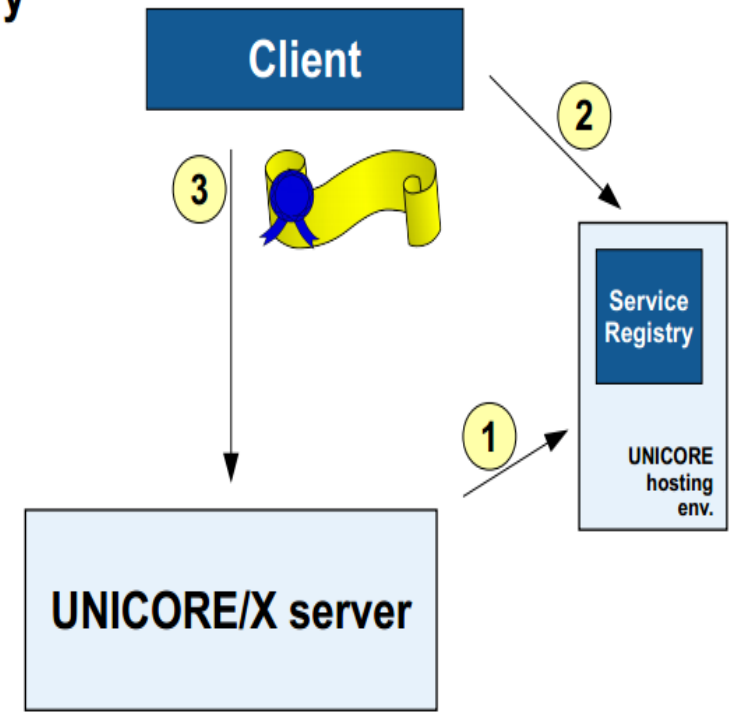
- Client makes client-authenticated SSL connection to the gateway
- Gateway issues a SAML assertion (optionally signed by the gateway) containing client certificate info
  → **Consignor assertion**
- Placed in SOAP header
- Gateway forwards request to target site, target site gets client information from the assertion



UNICORE v6

SAML assertions for trust delegation

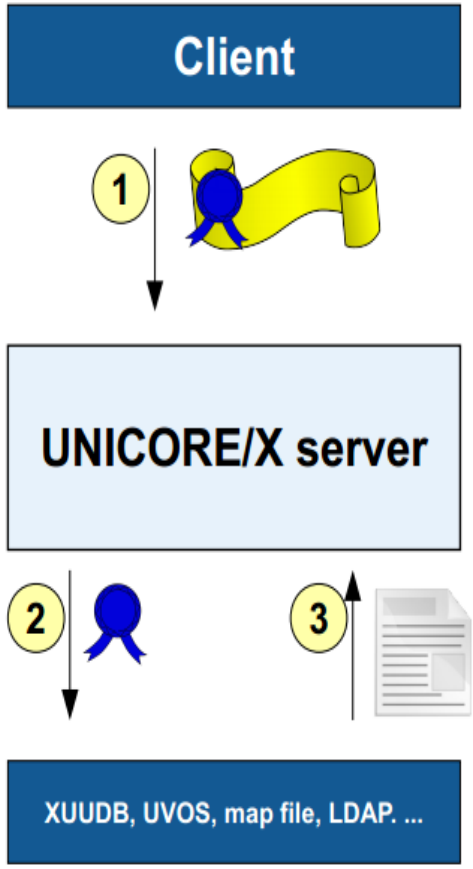
- Server publishes identity information (DN) to the **registry**
- Client gets identity info from the registry
- Client issues **Trust delegation assertion**
- Client sends request, and adds the TD to the SOAP header



UNICORE v6

Authorisation attributes

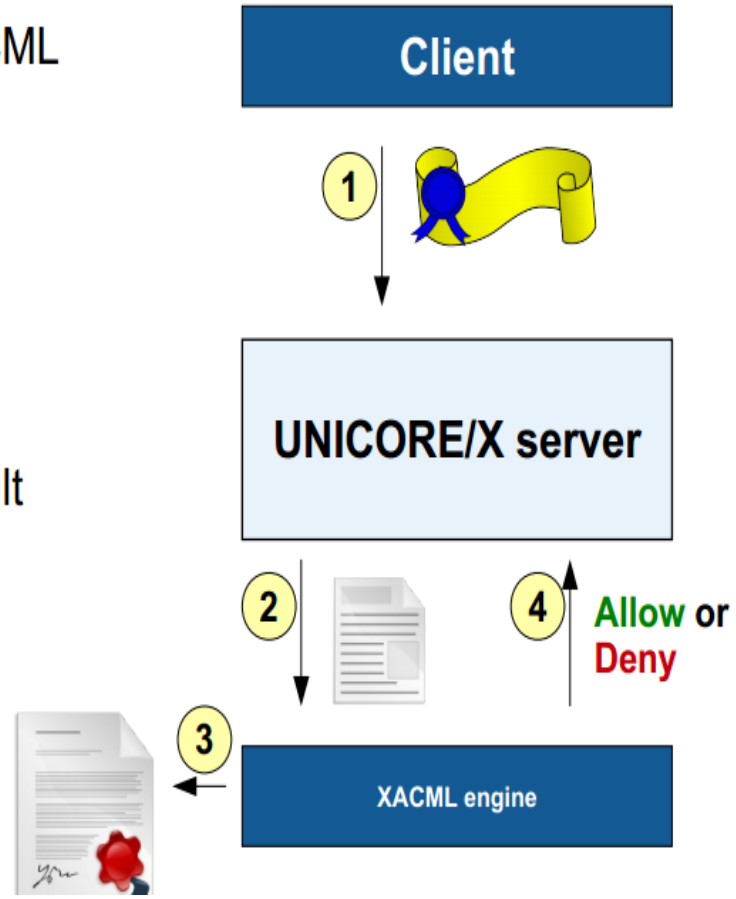
- Authorisation process occurs on the web-service level
- User identity (certificate or DN) is used by the UNICORE/X server to retrieve attributes
- Current sources:
  - XUUDB (default)
  - UVOS (or SAML VOMS)
  - Local map file
- Typical attributes
  - Local Unix login (xlogin)



UNICORE v6

Authorisation: XACML

- Attributes are used for an XACML callout
  (Default XACML 1.0 engine is built into UNICORE/X)
- XACML policies are checked
- Engine returns evaluation result
- UNICORE/X allows or denies the intended action (web service method invocation)



GLOBUS V4

Overviews

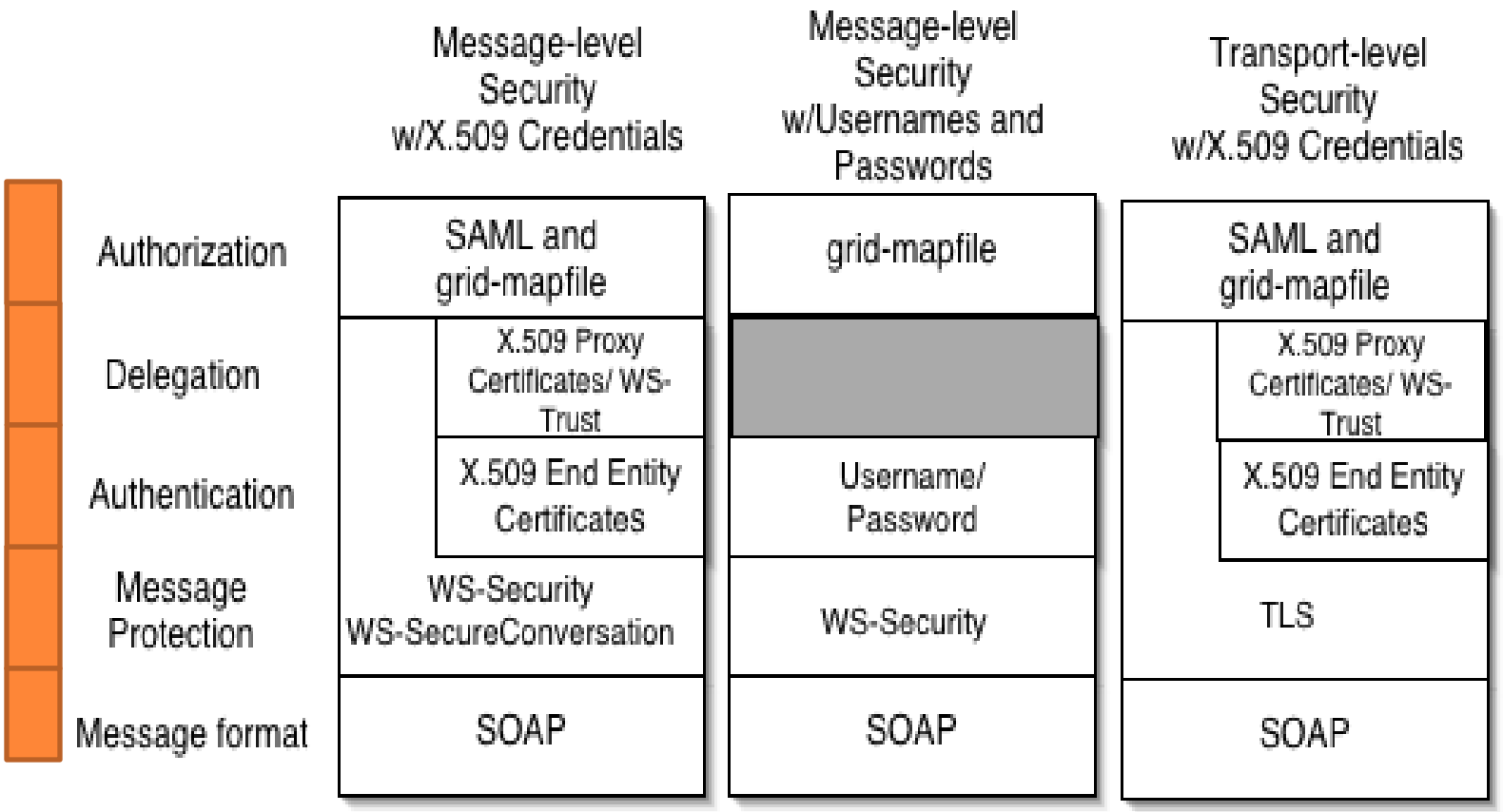
GT4.0 supports both message-level and transport-level security.

message-level security: Support for the WS-Security standard and the WS-SecureConversation.

transport-level security: Authentication via TLS with support for X.509 proxy certificates.

GLOBUS V4

GSI Functional Layers



GLOBUS V4

GSI Functional Layers (cont)

	GSI Secure Conversation	GSI Secure Message	GSI Transport
Technology	WS-SecureConversation	WS-Security	TLS
Privacy (Encrypted)	YES	YES	YES
Integrity (Signed)	YES	YES	YES
Anonymous authentication	YES	NO	YES
Delegation	YES	NO	NO
Performance	Good if sending many messages	Good if sending few messages	Best

GLOBUS V4

Message Protection:

The Web Services portions of GT4 use SOAP as their message protocol for communication.

GLOBUS V4

message-level security:

GSI implements the WS-Security standard and the WS-SecureConversation specification to provide message protection for SOAP messages.

WS-SecureConversation allows for a less computational overhead.

GLOBUS V4

Message Protection (cont)

Transport-level security:

Authentication via TLS and normally used in conjunction with X.509 proxy certificates. But can also be used without such certificate in “anonymous transport-level security.” mode.

GLOBUS V4

Authentication and Delegation

GSI use X.509 Certificates, Anonymous authentication or plain username and passwords for authentication and deledation.

GLOBUS V4

Authentication and Delegation (cont):

X.509 Credentials:

GSI uses X.509 end entity certificates (EECs) to identify persistent entities such as users and services.

GSI also supports delegation and single sign-on through the use of standard X.509 Proxy Certificates.

GLOBUS V4

Authentication and Delegation (cont):

Username and Password Authentication

GSI may use WS-Security with textual Usernames and Passwords as described in the WS-Security standard.

GLOBUS V4

Authentication and Delegation (cont):

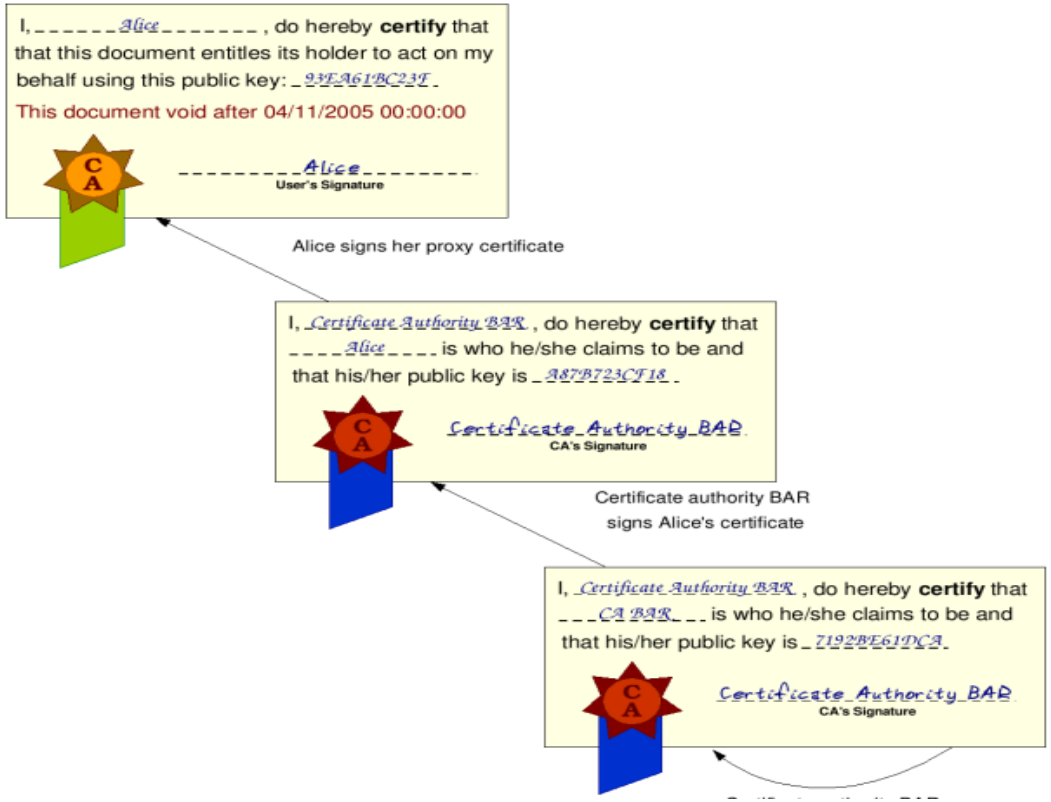
Delegation:

GT4 supports a delegation service that provides an interface to allow clients to delegate (and renew) X.509 proxy certificates to a service.

GLOBUS V4

Authentication and Delegation (cont):

X.509 Proxy Certificates:



GLOBUS V4

Authorization:

Server side authorization:

- **None**: No authorization will be performed.
- **Self**: compare the client's identity with the service's identity.
- **Gridmap**: A gridmap is a list of 'authorized users' akin to an ACL
- **Identity authorization**: compare the client's identity with a specified identity.
- **Host authorization**: Allow access if it presents a host credential that matches a specified hostname.
- **SAML Callout authorization**: delegate the authorization decision to an OGSA

GLOBUS V4

Authorization (cont):

Client-side authorization

- **None**: No authorization will be performed.
- **Self**: compare the client's identity with the service's identity.
- **Identity authorization**: compare the client's identity with a specified identity.
- **Host authorization**: Allow access if it presents a host credential that matches a specified hostname.

GLOBUS V4

Authorization (cont):

Custom authorization

- GSI provides an infrastructure to easily plug in our own authorization mechanisms.

References

I. Foster and C. Kesselman, *The Grid: Blueprint for a New Computing Infrastructure*. Morgab Kaufmann Publishers, 1999.

II. Maozhen Li, Mark Baker, The Grid Core Technologies, Wiley, 2005.

III. Wikimedia.com.

IV. Globus project tech page.

V. Unicore project tech page.