Grid Security Infrastructure

<u>On basis of works:</u>

An overview of the methods used to create a secure grid

- *Mike Jones (The University of Manchester)*

Security Implications of Typical Grid Computing Usage Scenarios

- Marty Humphrey (Computer Science Department, University of Virginia)
- Mary R. Thompson (Distributed Security Research Group Lawrence, Berkeley National Laboratory Berkeley, CA)

GridSec: Trusted Grid Computing

Kai Hwang, Yu-Kwong Kwok, Shanshan Song and others
(Internet and Grid Computing Laboratory, University of Southern California)

The Security Architecture for Open Grid Services

Nataraj Nagaratnam, Philippe Janson, John Dayka and others
(IBM Corporation; Department of Computer Science, University of Chicago)

CSE 720
Sergey Chernokozinskiy
November 30, 2006

Introduction (motivation)

- **Grid Security Infrastructure** - is a specification for secret, tamper proof, delegatable communication between software in the grid computing environment
- There are many ways to access the resources of a Computational Grid, BUT all of them should be more or less secure, because:
  - very attractive data resources and compute resources;
  - problems in one place can spread rapidly within a Grid;
  - stolen password/key can be used anywhere in a Grid;
- Microsoft has developed a prototype of a grid-oriented security language - Security Policy Assertion Language, or SecPAL. “Grids are becoming widely used in enterprises, as well as for sharing computing resources among academic research institutions. However, there is no single, widely used approach to dealing with grid security”.
- A comprehensive set of Grid usage scenarios is presented with regard to security requirements such as authentication, authorization, integrity, and confidentiality.

Preconditions to user Grid sessions

A Grid Session – is roughly defined as the activities that a particular user might perform during a single workday.

The following assumptions are made about the Grid Computing environment as a whole:

✓*Grid-wide Unique IDs*

✓*Some Resources Will Require Local IDs*

✓ *Multiple Authentication Sources*

This should take place prior to a particular user engaging in GS:

✓ *Allocation Requests on Per-Resource Basis*

✓ *Short-lived Credentials*

✓ *Per-Session Security Parameters*

Usage scenarios

1. **Immediate job execution**
2. **Job execution requiring advance scheduling**
3. **Job control**
4. **Accessing grid information services**
5. **Auditing use of Grid resources**

Grid user refers to the person who is attempting to access a resource;

Principal is used to mean either human or process that has an identity associated with it and wants to make use of or to provide resources;

Stakeholders are people or organizations who set the use policy for a resource;

Grid gateway is a process which accepts remote requests to use resources;

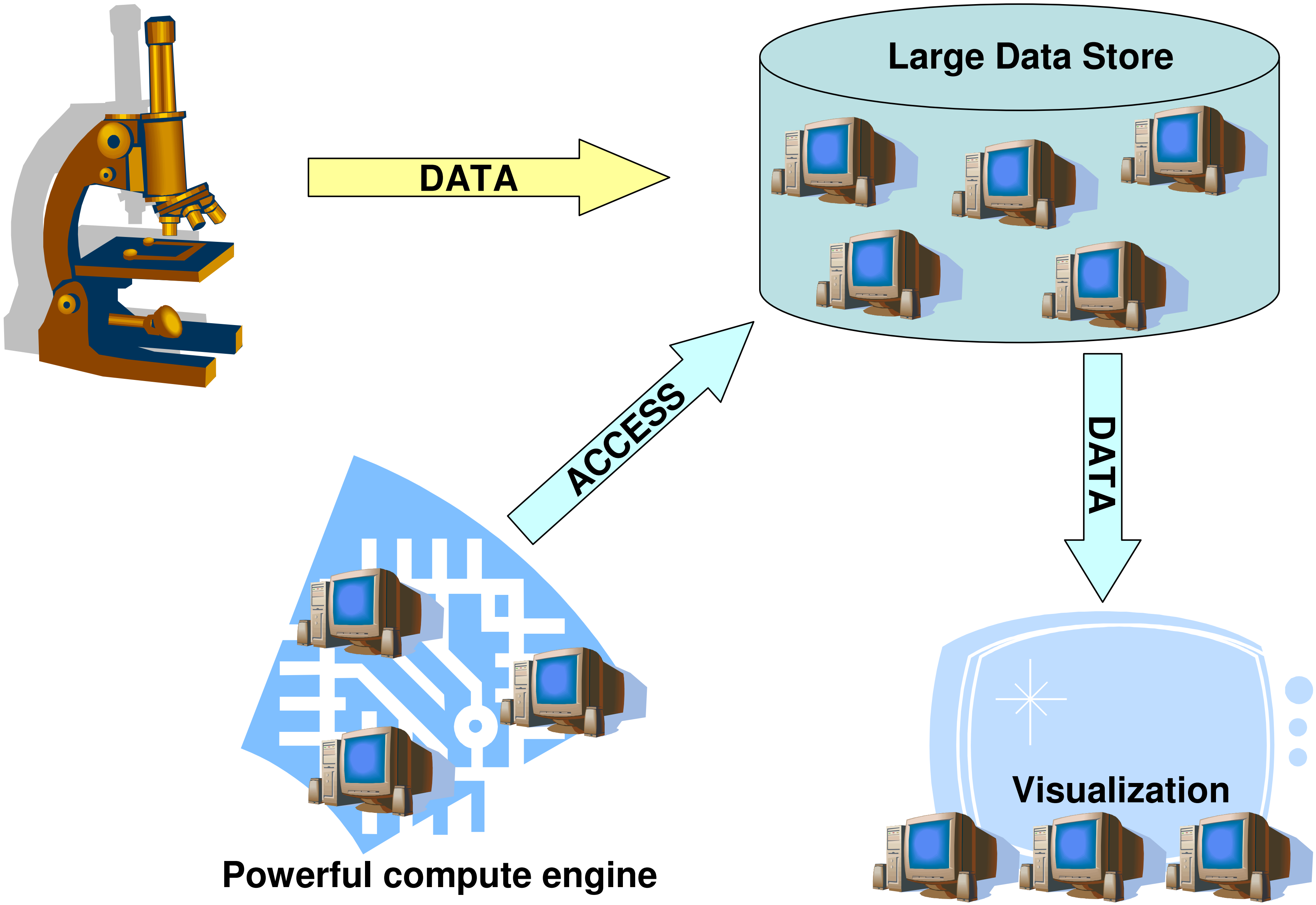
Grid resource gateway is the process that actually controls the use of the resource;

Grid administrator is a Grid-aware person with responsibility for the overall functioning of the Grid;

Usage scenarios

1. Immediate job execution
2. Job execution requiring advance scheduling
3. Job control
4. Accessing grid information services
5. Auditing use of Grid resources

A user wishes to combine resources from multiple sites into a single, coordinated job for immediate execution.



Immediate job execution
(The security requirements)

1. **If the set** of candidate hosts has not been identified by the user, the super scheduler will need to interact with the Information Services component(s) of the Grid to identify the set of possible hosts.

2. **The super scheduler** must determine if the target user is allowed to execute on each of the target Grid machines, and, if so, the remaining allocations of the user.

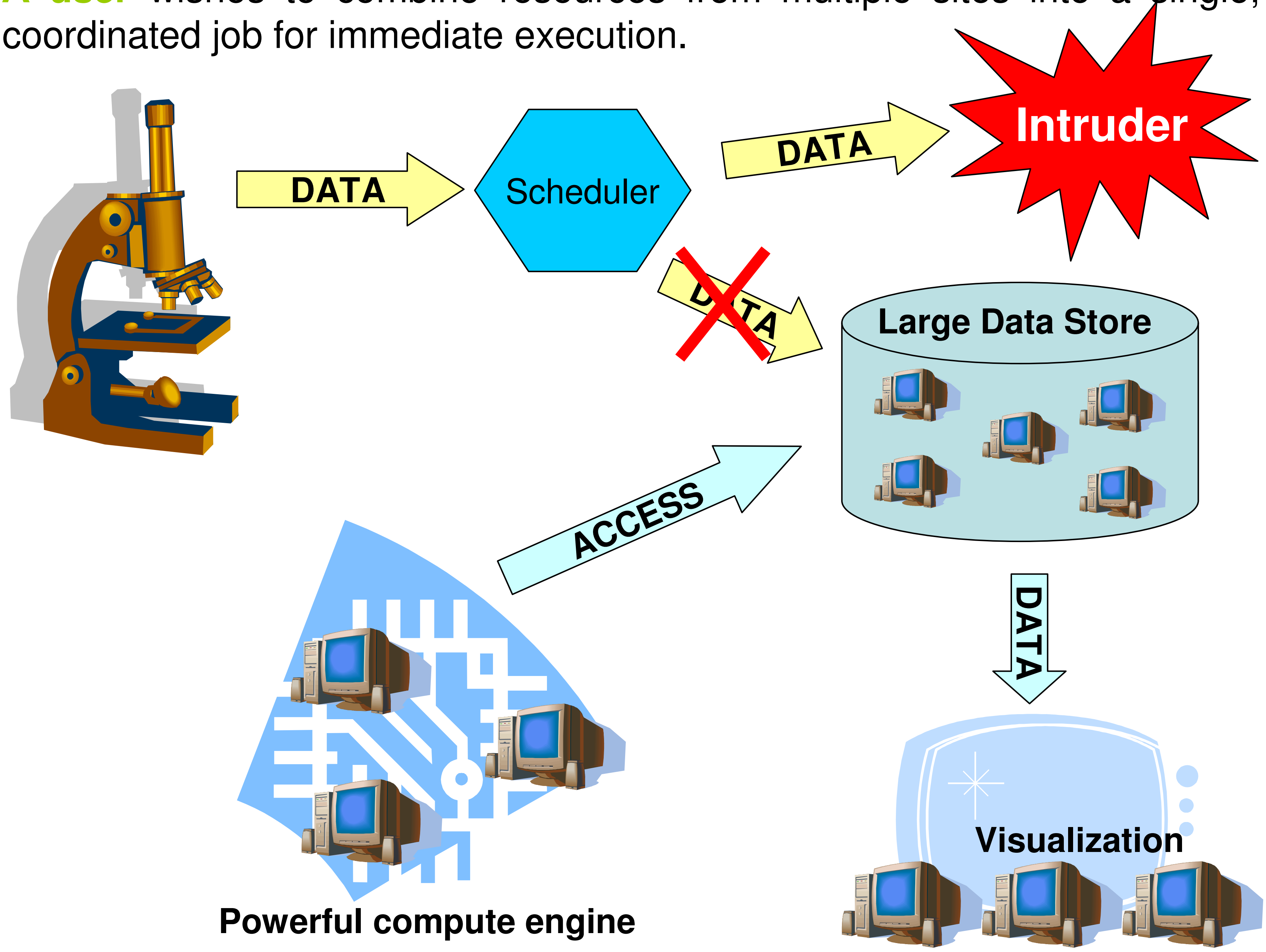
3. **A controlling agent** or each remote job in a sequence needs to request resources on behalf of the user, perhaps through subsequent calls to a super scheduler.

4. **Mutual authentication** of user and Grid gateway on specified host needs to be done before a piece of the job is run there.

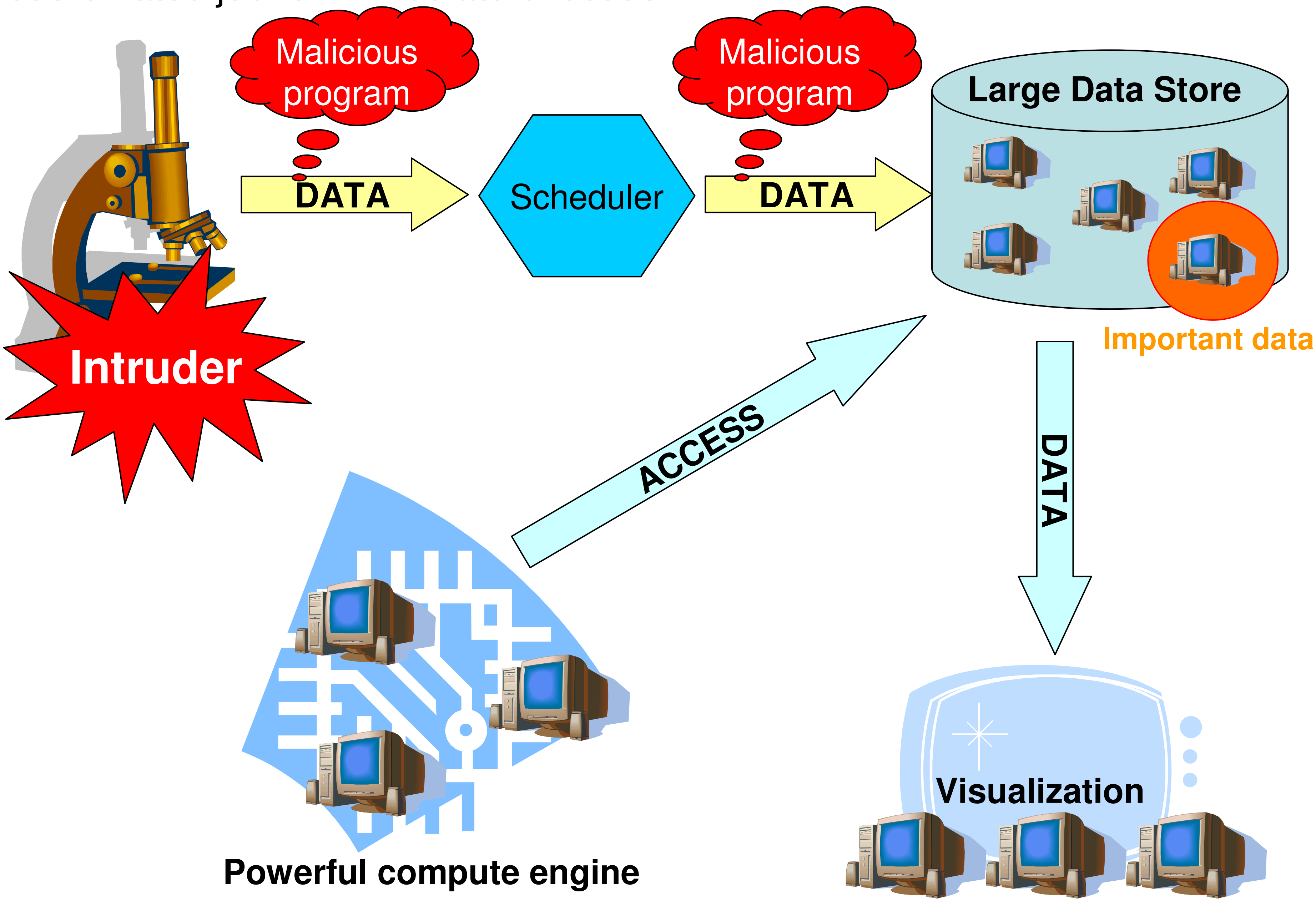
5. **The grid gateway** on a specified host must map the Grid ID to a local ID and submit the request to the resource gateway so that the job will run as the authorized local user.

6. **The executing jobs** may need to be given authorization to read and write remote files on behalf of the user.

A user wishes to combine resources from multiple sites into a single, coordinated job for immediate execution.



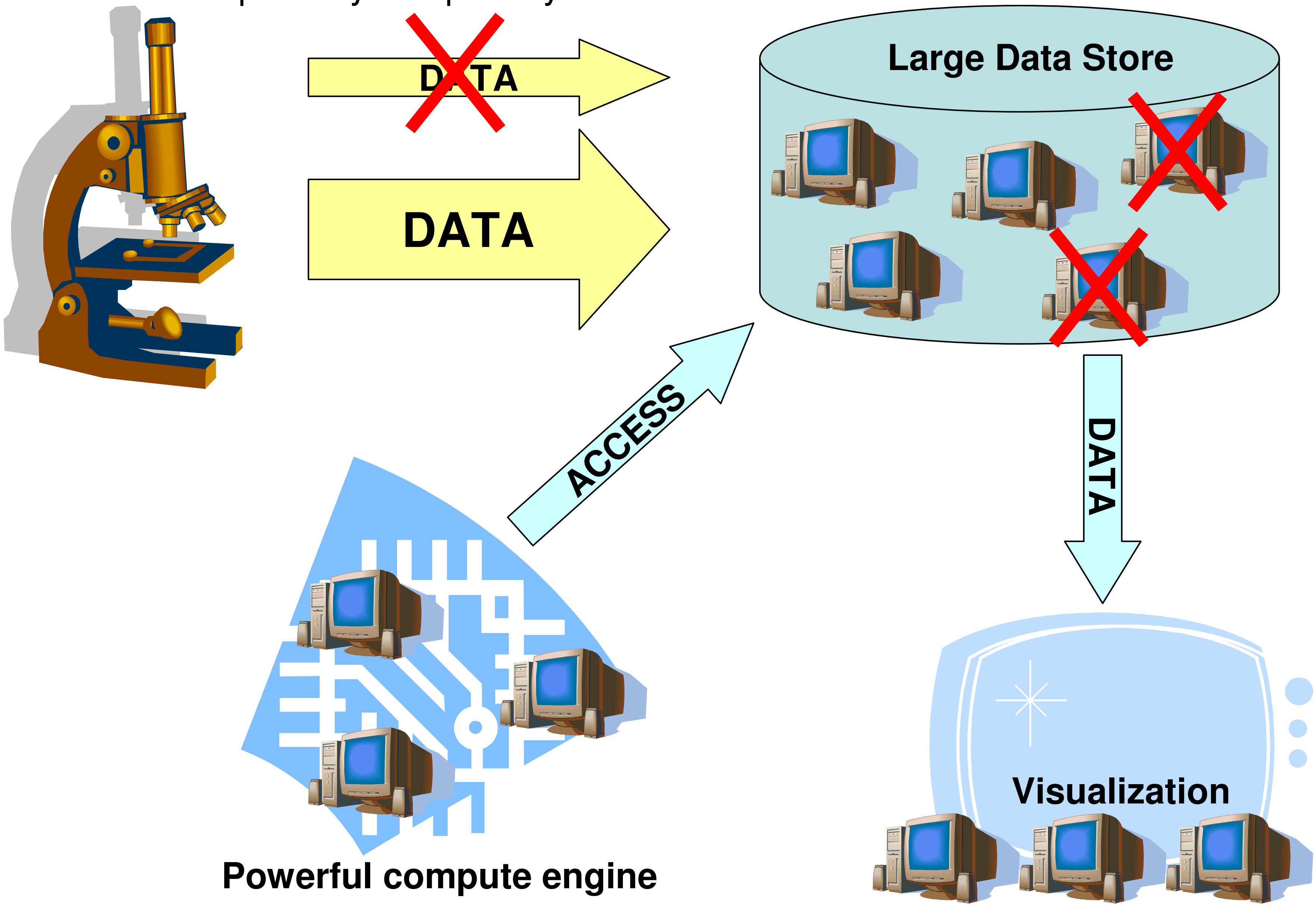
A user wishes to combine resources from multiple sites into a single, coordinated job for immediate execution.



Usage scenarios

1. Immediate job execution
2. Job execution requiring advance scheduling
3. Job control
4. Accessing grid information services
5. Auditing use of Grid resources

If the large data flow from an instrument must be processed in real time, it may require the advance reservation (or *co-scheduling)* of data storage, network bandwidth and possibly compute cycles.



Job execution requiring advance scheduling

Advance reservations require:

1. Delegation of the user’s rights to a super scheduler and bandwidth broker to make the reservations on behalf of the user.

2. Assurance that if a user has been granted a reservation for the future, he/she will have access at the time the reservation is claimed.

3. Bandwidth reservations usually require service agreements for priority bandwidth between ISP’s and compute sites. This implies that a bandwidth broker needs to know at reservation time that user’s connection will come from an authorized site.

A user directly interacts with the individual resource gateways to claim the reservation:

1. The user must be able to identify himself as the entity that made the reservation.

2. The user should still have access to all the resources that he has reserved, except in extreme cases.

3. In the case of a user losing access to a resource, a check should be made of advance reservations in his name.

Usage scenarios

1. Immediate job execution
2. Job execution requiring advance scheduling
3. Job control
4. Accessing grid information services
5. Auditing use of Grid resources

Job control

Job control (standard requirement of users with long-running remote jobs) - is the ability to disconnect from a job and then at a later time and possibly from a different location reattach to it.

Two types of job control:

- **Monitoring**
- **Steering (a user may control, who may connect to a job)**

Includes:

1. A user must be able to set the access policy to his own resources or jobs.
2. The potential collaborator must be able to authenticate to the computation itself.
3. In the case of a forced termination, the system administrator must detect the out-of-control process and trace its origin to a particular Grid user.
4. Alternatively, Grid monitoring software might detect the out-of-control process and notify the system administrator.
5. The system administrator should be able to inform the Grid Administrators that the process is about to be terminated.
6. The Grid Administrators must be able to terminate the individual components of the job directly or indirectly on each site.
7. The job owner must be notified by the Grid Administrator that his job has been terminated.

Usage scenarios

1. Immediate job execution
2. Job execution requiring advance scheduling
3. Job control
4. Accessing grid information services
5. Auditing use of Grid resources

Accessing grid information services

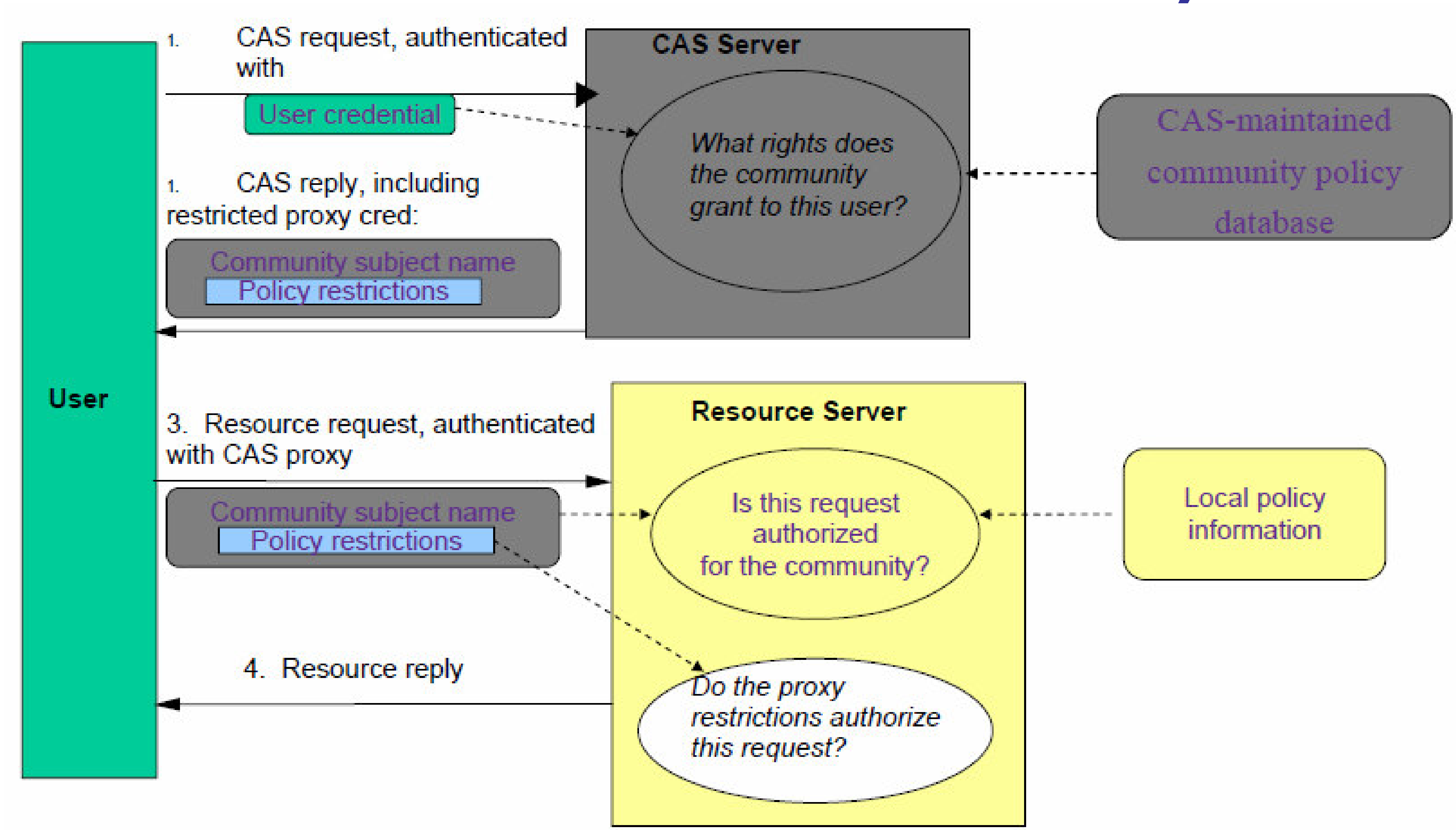
Many services require carefully controlled access to information regarding the services they provide, their current status, and who can use them.

1. Authentication should take place between the user and the information services.

2. The information services should implement the access control policy as desired by the service.

3. When publishing information, confidentiality or message integrity on the communication from the publisher to the information services could be required by the publisher.

Accessing grid information services with CAS (Community Authorization Service)


1. CAS request, authenticated with
User credential
CAS Server
What rights does the community grant to this user?
CAS-maintained community policy database
1. CAS reply, including restricted proxy cred:
Community subject name
Policy restrictions
User
3. Resource request, authenticated with CAS proxy
Community subject name
Policy restrictions
Resource Server
Is this request authorized for the community?
Local policy information
4. Resource reply
Do the proxy restrictions authorize this request?

Mike Jones' scheme from "An overview of the methods used to create a secure grid"

Usage scenarios

1. Immediate job execution
2. Job execution requiring advance scheduling
3. Job control
4. Accessing grid information services
5. Auditing use of Grid resources

Auditing use of Grid resources

1. The resource gateway server must keep an non-forgeable log of all access by unique user identification and time of access.
2. The format of the entries to this log must be negotiated between the system administrator and the resource gateway.
3. Access to this log should be carefully restricted, but stakeholders need to be able to see the entries for their resources.
4. There is a need to identify a stakeholder with a resource.
5. To accomplish real-time intrusion detection, the resource gateway needs recognize and signal especially troublesome resource access requests in additions to logging.

Main aspects of Grid Security

General Security:

– **Protection**

- Hackers – securing your system from the outside world
- Viruses – securing your system from rogue software
- Physical – securing your console access and hardware

– Mutual Identification

- Knowing the entities you are dealing with

– Authorisation

- Knowing/assigning the entitlements of these entities

– Accounting

- Knowing what authorised (and unauthorised) actions were done by whom

– **Privacy**

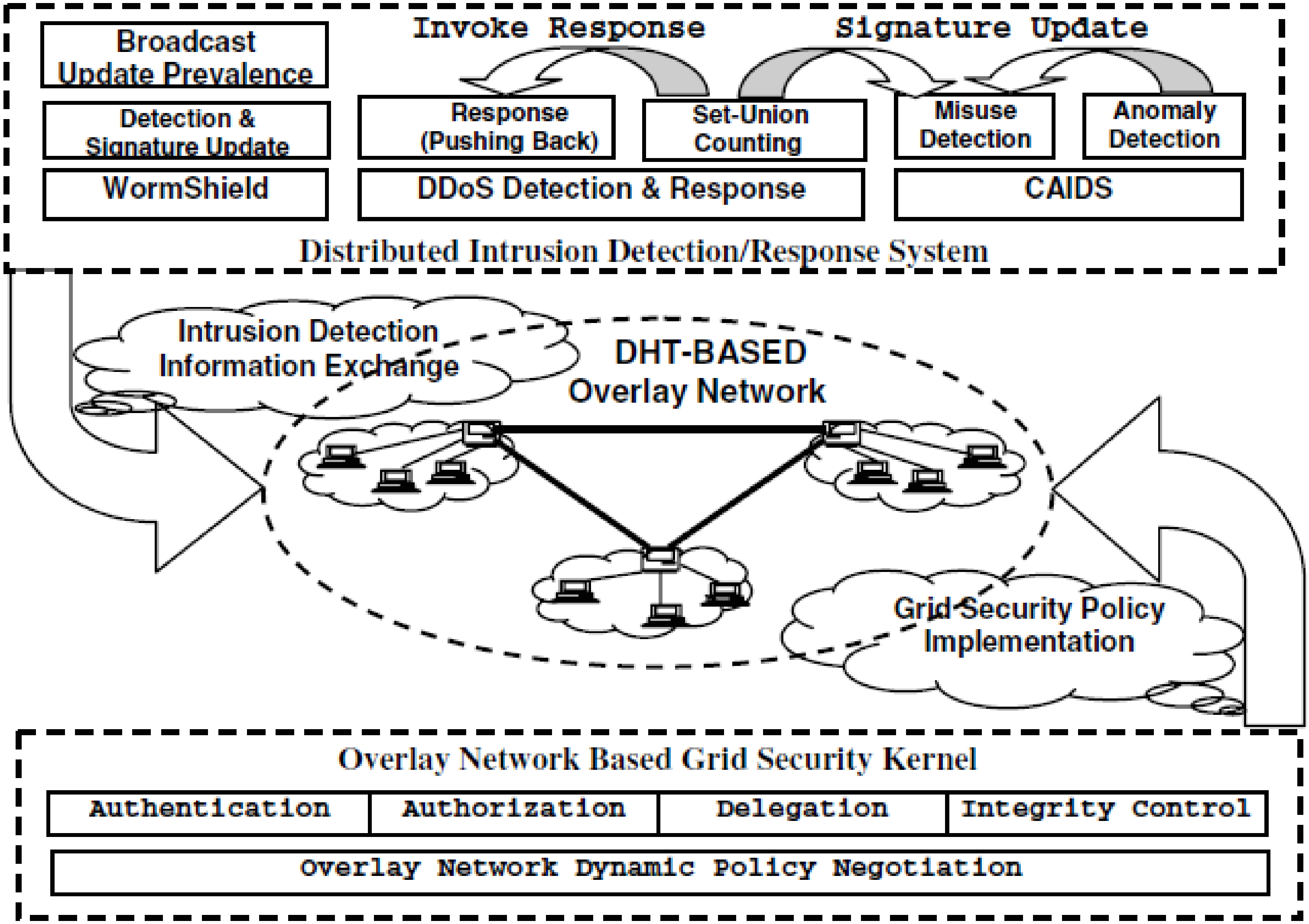
- Respecting and enforcing access rights to data
- Observing Licences,

– **Integrity**

- Data storage, Backups, Longevity

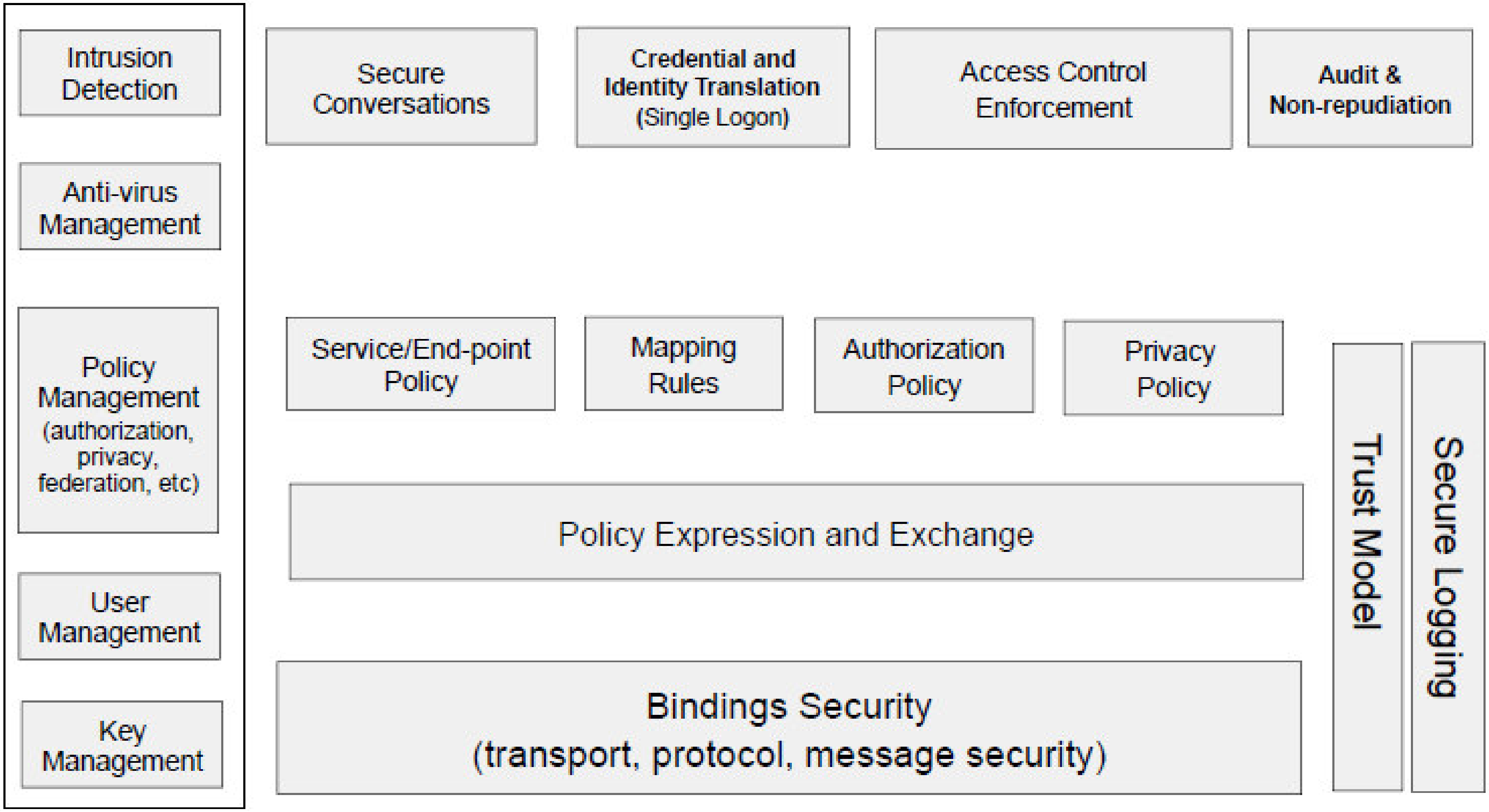
Based on Mike Jones
An overview of the methods used to create a secure grid

GridSec infrastructure for building self-defense capabilities to protect Grid sites


Broadcast Update Prevalence
Invoke Response
Signature Update
Detection & Signature Update
Response (Pushing Back)
Set-Union Counting
Misuse Detection
Anomaly Detection
WormShield
DDoS Detection & Response
CAIDS
Distributed Intrusion Detection/Response System
Intrusion Detection Information Exchange
DHT-BASED Overlay Network
Grid Security Policy Implementation
Overlay Network Based Grid Security Kernel
Authentication
Authorization
Delegation
Integrity Control
Overlay Network Dynamic Policy Negotiation

Kai Hwang, Yu-Kwong Kwok, Shanshan Song, Min Cai Yu Chen, Ying Chen, Runfang Zhou, and Xiaosong Lou

Components of Grid Security Model



Intrusion Detection

Anti-virus Management

Policy Management (authorization, privacy, federation, etc)

User Management

Key Management

Secure Conversations

Credential and Identity Translation (Single Logon)

Access Control Enforcement

Audit & Non-repudiation

Service/End-point Policy

Mapping Rules

Authorization Policy

Privacy Policy

Policy Expression and Exchange

Bindings Security (transport, protocol, message security)

Trust Model

Secure Logging

Nataraj Nagaratnam, Philippe Janson, John Dayka, Anthony Nadalin, Frank Siebenlist, Von Welch, Ian Foster, Steve Tuecke