Grid Computing 7700
Fall 2005
Lecture 16: Grid Security

Gabrielle Allen
allen@bit.csc.lsu.edu

http://www.cct.lsu.edu/~gallen
Required Reading

- Chapter 16 of The Grid (version 1), freely available for download on the web
- GSIGrid Security Infrastructure
  - http://www.globus.org/toolkit/docs/4.0/security/

Recommended:

- Chapter 21 of the Grid (version 2)
  - Different aspects brought in by considering web/grid services
GSI: Grid Security Infrastructure

- Security solution from Globus
  - [http://www.globus.org/toolkit/docs/4.0/security/](http://www.globus.org/toolkit/docs/4.0/security/)
- Based on public key cryptography (asymmetric cryptography)
- **Motivations:**
  - Secure communication
  - Security across organization boundaries
  - Single sign on and delegation of credentials
- Standards based
Terminology

- **Authentication:** Establishing who you are
- **Authorization:** Establishing what you are allowed to do
- **Assurance/accreditation:** Validating authority of a service provider
- **Accounting and auditing:** Tracking, limiting and charging for resources
- **Messages:**
  - Message integrity
  - Message confidentiality
- **Non-repudiation:** Proof that you got the message
- **Digital signature:** Assurance about the message
- **Certificate authority:** A body which issues and manages security credentials
- **Delegation:** Authority to act as someone else

Balance with impact on performance, implementation and administrative costs
TLS/SSL

- TLS: Transport Layer Security Protocol is the successor to SSL: Secure Socket Layer.
- Secured Sockets Layer is a protocol that transmits your communications over the Internet in an encrypted form. SSL ensures that the information is sent, unchanged, only to the server you intended to send it to.
- Lies above TCP/IP layer and below HTTP layer.
- Developed by Netscape for transmitting private documents via the Internet. SSL works by using a private key to encrypt data that's transferred over the SSL connection. Both Netscape Navigator and Internet Explorer support SSL, and many Web sites use the protocol to obtain confidential user information, such as credit card numbers. By convention, URLs that require an SSL connection start with https: instead of http:.

- Requires a direct transport layer between endpoints
Public Key Encryption

- Entity generates two keys, one is designated as the public key, one is the private key.
- The private key must be kept private!
- Public key is given out (eg in an X.509 certificate)
- If one key is used to encrypt a message, the other key must be used to decrypt it.
- Possession of private key (and ability to encrypt/decrypt challenge messages) proves ownership.
How B sends an encrypted message to A:

1. A sends public key $e$
2. B uses $e$ to encrypt message $m$, $c=E(e,m)$
3. B sends $c$ to A
4. A uses $d$ to decrypt $c$, $m=D(d,c)$
Public Key Encryption

- Encryption method is public knowledge so does not provide data integrity or authentication of data origin.
- Slower than other methods (not so good for bulk transfer or lots of small items).
- Based on belief that it is not possible to determine the decryption mechanism from the encryption mechanism.
- More secure than username/password (requires passphrase and possession of private key).
- Security relies on identifier establishment.
Public Key Authentication

1. Send public key
2. Send challenge encrypted with public key
3. Decode challenge with private key
4. Send encrypted answer back
5. Decrypt answer and verify
Non-Repudiation

- In general, nonrepudiation is the ability to ensure that a party to a contract or a communication cannot deny the authenticity of their signature on a document or the sending of a message that they originated.
- On the Internet, a digital signature is used not only to ensure that a message or document has been electronically signed by the person that purported to sign the document, but also, since a digital signature can only be created by one person, to ensure that a person cannot later deny that they furnished the signature.
Digital Signature

- An electronic signature that authenticates the identity of the sender of a message, the signer of a document, or ensures that the contents of a message are intact.
- Digital signatures are easily transportable, cannot be imitated by someone else, and can be automatically time-stamped.
- The ability to ensure that the original signed message arrived means that the sender cannot repudiate it later.
- A digital certificate contains the digital signature of the certificate-issuing authority so that anyone can verify that the certificate is real.
Digital Signature

- To sign a piece of information, compute its mathematical hash. (The algorithm used to compute this hash must be known to the recipient of the information, but it isn't a secret.)

- Using your private key, encrypt the hash, and attach it to the message. Make sure that the recipient has your public key. To verify that your signed message is authentic, the recipient of the message will compute the hash of the message using the same hashing algorithm you used, and then decrypt the encrypted hash. If the newly-computed hash and the decrypted hash match, it proves that you signed the message and it has not been changed.
Hashes

- Public key encryption is relatively slow, so using it for digital signing by encrypting messages is not efficient.
- Instead sign a much smaller (redundant) proxy (or digest or hash) for the message to guarantee origin (authenticity) and genuineness (integrity).
- Other names digital fingerprint, message fingerprint, cryptographic hash, cryptographic checksum.
- SHA-1: Secure Hash Algorithm compresses Microsoft Office to disk space used for “xxxxxxxxxxxxxxxxxxxxxxxxxx”
Digital Certificate

- Public documents which identifies (authenticates) users and services on a Grid.
- The signer of a digital certificate says something like “I attached G.Allen’s public key to this digital certificate and then signed it with my private key”
- Any user of G.Allens digital certificate must completely trust the competency and honesty of the person/organization who signed the certificate
- For anyone to confidently use G.Allens digital certificate they must also trust that they have a validated copy of the signers public key
- There is nothing secret about the contents of a digital certificate
- Has expiration date
- Analogy e.g. with driving license, issued by DMV and trusted by other countries and states, or my PhD certificate.
Managing Digital Certificates

- Digital certificate administrative frameworks are called “public key infrastructures” (PKIs).
- Two major ones (sometime interoperable)
  - X.509 (standardized by IETF)
  - Pretty Good Privacy (PGP)
Certificate Authority

- Centrally controlled system for managing digital certificates in X.509 talk is a “certificate authority”
- Trusted third party (CA) which manages digital certificate application, certification, issuance and revocation
- X.509 trust networks (e.g. Mississippi will trust driving licenses issued in LA)
- Each X.509 PKI implementation has a root CA, which produces a self signed or root certificate
Distinguished Name (DN)

- Unique identifier for the owner (and issuer) of a certificate (with respect to the CA)
  - Analogy: social security number seems to be the main identifier in US

- With GSI, the gridmap file is used to map DNs to local user names

- `/O=LSU/OU=CCT/OU=CSC7700/OU=cct.lsu.edu/CN=User05`
GSI Grid Certificates

- On the Grid, each user and service is identified via a GSI certificate, which includes
  - A subject name, which identifies the person or object that the certificate represents.
  - The public key belonging to the subject.
  - The identity of a Certificate Authority (CA) that has signed the certificate to certify that the public key and the identity both belong to the subject.
  - The digital signature of the named CA

- GSI certificates are encoded in the X.509 certificate format.
- GSI provides single-sign-on and users have identity certificates with private/public keys instead of using username/password.
My Alliance Certificate

Certificate:
Data:
  Version: 3 (0x2)
  Serial Number: 338 (0x152)
  Signature Algorithm: md5WithRSAEncryption
  Issuer: C=US, O=National Computational Science Alliance, OU=Certification Authority
  Validity
    Not After : Aug 30 10:16:51 2004 GMT
  Subject: C=US, O=National Computational Science Alliance, CN=Gabrielle Allen
  Subject Public Key Info:
    Public Key Algorithm: rsaEncryption
    RSA Public Key: (1024 bit)
      Modulus (1024 bit):
        00:b6:ad:2f:fc:20:f3:45:8e:a0:9c:e2:a8:a5:1d:
        ETC ETC ff:f4:b7:2a:ce:d4:f8:e3:cd
      Exponent: 65537 (0x10001)
My Alliance Certificate

X509v3 extensions:
  X509v3 Key Usage: critical
    Digital Signature, Non Repudiation, Key Encipherment, Data Encipherment
  X509v3 Authority Key Identifier:

X509v3 CRL Distribution Points:
  URI:https://ca.ncsa.edu/5aba75cb.r0

Signature Algorithm: md5WithRSAEncryption
    ETC ETC
  37:26:1c:f0
Certificate Authorities and Policies

For example:

- **DOE**  [http://www.doe grids.org/Docs/CP-CPS.pdf](http://www.doe grids.org/Docs/CP-CPS.pdf)
- **Alliance**  [http://archive.ncsa.uiuc.edu/SCD/Alliance/GridSecurity/Certificates/AllianceCP9.1.html](http://archive.ncsa.uiuc.edu/SCD/Alliance/GridSecurity/Certificates/AllianceCP9.1.html)
- **In Louisiana?**
Globus Grid Certificates

- `grid-cert-request` is usually used to request a certificate
- `grid-cert-request -ca`
- Certificate is usually stored in `.globus` directory: `usercert.pem`
- `userkey.pem` is private key
- Private key is encrypted with a passphrase.
GSI in Action
“Create Processes at A and B that Communicate & Access Files at C”

User Proxy

Remote process creation requests*

Authorize
Map to local id
Create process
Generate credentials

Remote file access request*

Site A
(Kerberos)

GSI-enabled GRAM server

Computer

Process
Kerberos ticket
Local id
Restricted proxy

Ditto

GSI-enabled GRAM server

Site B
(Unix)

Computer

Process
Local id
Restricted proxy

Site C
(Kerberos)

GSI-enabled FTP server

Storage system

* With mutual authentication

Single sign-on via “grid-id” & generation of proxy cred.
Or: retrieval of proxy cred. from online repository

* With mutual authentication
# Grid Security Requirements

## User View

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1)</td>
<td>Easy to use</td>
</tr>
<tr>
<td>2)</td>
<td>Single sign-on</td>
</tr>
<tr>
<td>3)</td>
<td>Run applications</td>
</tr>
<tr>
<td></td>
<td>ftp, ssh, MPI, Condor, Web, ...</td>
</tr>
<tr>
<td>4)</td>
<td>User based trust model</td>
</tr>
<tr>
<td>5)</td>
<td>Proxies/agents (delegation)</td>
</tr>
</tbody>
</table>

## Resource Owner View

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1)</td>
<td>Specify local access control</td>
</tr>
<tr>
<td>2)</td>
<td>Auditing, accounting, etc.</td>
</tr>
<tr>
<td>3)</td>
<td>Integration w/ local system</td>
</tr>
<tr>
<td></td>
<td>Kerberos, AFS, license mgr.</td>
</tr>
<tr>
<td>4)</td>
<td>Protection from compromised resources</td>
</tr>
</tbody>
</table>

## Developer View

- API/SDK with authentication, flexible message protection, flexible communication, delegation, ...  
  - Direct calls to various security functions (e.g. GSS-API)  
  - Or security integrated into higher-level SDKs:  
    - E.g. GlobusIO, Condor-G, MPICH-G2, HDF5, etc.
Candidate Standards

- **Kerberos 5**
  - Fails to meet requirements:
    > Integration with various local security solutions
    > User based trust model

- **Transport Layer Security (TLS/SSL)**
  - Fails to meet requirements:
    > Single sign-on
    > Delegation
Grid Aspects

- Single sign-on
- Delegation
- Firewalls
- Distributed systems (intermediate components): Message projection must be moved from transport layer to message layer
- Group authentication and authorisation (for dynamic Vos)
Grid Security Infrastructure (GSI)

- Extensions to standard protocols & APIs
  - Standards: SSL/TLS, X.509 & CA, GSS-API
  - Extensions for single sign-on and delegation

- Globus Toolkit reference implementation of GSI
  - SSLeay/OpenSSL + GSS-API + SSO/delegation
  - Tools and services to interface to local security
    > Simple ACLs; SSLK5/PKINIT for access to K5, AFS; ...
  - Tools for credential management
    > Login, logout, etc.
    > Smartcards
    > MyProxy: Web portal login and delegation
    > K5cert: Automatic X.509 certificate creation
GSS-API

- Generic Security Services Application Programming Interface.
- The GSS-API is a generic API for doing client-server authentication. (calls for authentication, confidentiality, integrity independent of underlying security systems)
- The motivation behind it is that every security system has its own API, and the effort involved with adding different security systems to applications is extremely difficult with the variance between security APIs. However, with a common API, application vendors could write to the generic API and it could work with any number of security systems.
GSI: Mutual Authentication

- Services mutually authenticate against each other on the Grid
- Trust relationships have to be set up beforehand ... which certificate authorities does LSU trust? (DOE, NCSA, GridLab, ....)
  - Admins and policy makers involved
  - Exchange of certificates and public keys
- Look in /etc/grid-security/certificates/
Mutual Authentication

- If entity X wants to invoke entity Y:
  - X provides certificate to Y
  - Y validates certificate
  - Y challenges X: send a message to X, X encrypts it with private key, and sends it back, Y decodes it with public key from certificate
  - Y provides certificate to X
  - Etc

- Finally they both trust each other ...
User Proxies

- Minimize exposure of user’s private key
- A temporary, X.509 proxy credential for use by our computations
  - We call this a user proxy certificate
  - Allows process to act on behalf of user
  - User-signed user proxy cert stored in local file
  - Created via “grid-proxy-init” command
- Proxy’s private key is not encrypted
  - Rely on file system security, proxy certificate file must be readable only by the owner
Delegation

- Remote creation of a user proxy
- Results in a new private key and X.509 proxy certificate, signed by the original key
- Allows remote process to act on behalf of the user
- Avoids sending passwords or private keys across the network