What to expect from this talk

• Collection of bits about GC I thought you might find interesting

• Mixed bag:
  • Systems engineering considerations
  • Exploration without bounds
What to expect from this talk

Shed some light on these questions:

• What is Grid Computing about?
• Why is it interesting to computer security folk?
• What is the status quo in Grid Computing, security-wise?
Fahrplan

• Introduction to Grid Computing
• Assorted Interesting Grid Issues
• Globus Toolkit & Issues
• Possible playgrounds
What is Grid Computing?

Grid Computing is

• Kind of Distributed Computing
.* Computing

Distributed Computing

Parallel Computing

Grid Computing
What is Grid Computing?

Grid Computing is

• Kind of Distributed Computing
• Direction of development in systems engineering
• Yet Another Virtualization Effort
• Pragmatic Virtualization
The problem of Grid Computing

• Pre-existing infrastructure
  • Hardware
  • Software
  • Organizations
  • Policies
• (Dynamically) Tie together resources
Heterogeneity

- Purpose of Unit
- Hardware
- Software
- Connectivity
- Trust
COs rule (literally)

- Important principle: COs always retain full control of their resources
- (resource owners always retain full control of that resource)
Virtual Organizations

- VOs – Virtual Organizations, COs – Conventional Organizations
- VOs: Created through overlay on COs
Virtual Organizations – Properties

• Lifetime: Arbitrary
• Highly dynamic
• User-driven
Purpose of VOs

• Reconciliation across COs:
  • different technologies
  • different policies

• Providing environment for cooperation across CO-boundaries
Old wine in larger bottles…

• Some problems are 30+ years old
• First occurred in larger multi-user computing environments
• Often see scaled up versions of these old problems in GC
Short Summary: GC Features

• Pragmatic Virtualization
  • Use what’s there
  • Share what’s there
  • Obey existing policies
Applications

Grid Computing applications are today found in:

• Science
  • Life sciences
  • Physics

• Medical environments
  • Hospitals

• Business
Grid Scenario I: LHC Computing Grid

- Large Hadron Collider Computing Grid
- Hosted mainly at CERN in Geneve
- Uses
  - Globus Toolkit Version 2
  - Condor
  - gLite
  - More middleware developed in DataGrid project

http://lcg.web.cern.ch/LCG/activities/middleware.html
“When it begins operations in 2007, it will produce roughly 15 Petabytes (15 million Gigabytes) of data annually, which thousands of scientists around the world will access and analyse.

The mission of the LHC Computing Project (LCG) is to build and maintain a data storage and analysis infrastructure for the entire high energy physics community that will use the LHC.”
Grid Scenario II: seti@home

- Large number of desktop PCs
- Different infrastructure, software, connectivity
- Resources “shared” in a rather unidirectional way
- Participants don’t know each other, no reason for trust
Grid Scenario III: Sun Grid

- Commercial service
- Users can rent time and storage in Sun Grid
Grid Scenario IV: MediGRID

• German Biosciences Research project
• Processing and sharing of biomedical data
• Protection requirements for some of the data regulated by law
Threats to Grid Systems

→ “Threat” as in “goal of an attacker”
Random compromise (e.g. for botnets)
  • Gateway to more interesting targets
  • Access to restricted databases
    • Research databases
    • Sensitive information, e.g. MediGRID
  • Harvesting computing power
  • Access to interesting devices
Attack vectors for Grid Systems

- Middleware software
- Custom software
- Identity/Authorization infrastructure
- Underlying software
  - OS
  - SSL implementation
  - ssh implementation
  - ...

Security Issues in Grid Computing

- Yes.
- Conceptual issues
- Issues in existing
  - Architecture
  - Implementation
- Unsolved security-relevant problems
Access Control

• Conventional:
  • MAC for
    • network boundary traversal
    • Resource access
  • DAC for user file access

• Grid Computing:
  • DAC for
    • User file access
    • Resource access
    • Network boundary traversal
“There are often firewalls and NATs between them, which are blocking packets. We believe it's going to be critical to the future of grid, not that you remove those things but that you make them another managed component of the grid that can be dynamically configured and reconfigured based on application needs. If I need to get two jobs talking to each other, but there's a firewall between them, I should be able to negotiate with that firewall to allow those two jobs to talk with each other, perhaps only for a limited period of time.“ – Steven Tuecke

Authorization

• Done via Identification per default:
  • Store lists of “$user is allowed to do $action” and “$user is not allowed to do $action” rules
  • Establish the identity of a requester
  • Derive from identity + list of rules whether requested action is authorized
Authorization and Identification

→ Shifts problem to identification. Trouble.

• Identification is a hard problem, esp. with only machines involved

→ Makes “Identification without Authorization” an interesting research topic
The Firewalling Mentality

- “Put a wall around the network. Inside: Good. Outside: Bad.”
The Firewalling Mentality

• Grid Computing ideal and Firewalling Mentality $\rightarrow$ clash

• But can’t we use DMZs…?
DMZs to the Rescue

→ Just put the particle accelerator in the DMZ?
Firewalling and DMZs

- Modelling around this with DMZs: not really
  - “Share resource with some” → “Put it in the DMZ”?
  - Dynamic sharing and DAC – remember?
Open Problems in GC security

- Authorization
- Users vs. Hosts: Minimize required level of trust
Globus Toolkit

• Globus Toolkit (GT) provides Grid middleware
  • e.g. GRAM, GSI, GridFTP
• Foundation for actual Grid applications
• Widely used
Globus Toolkit

- Written mainly in
  - Java
  - C/C++
- APIs for other languages available, e.g. pyGlobus
- Earlier versions: more C/C++, newer versions: more Java
- “Latest version” seemingly not that common…
Some Globus Toolkit Users

- TeraGrid
- LHCGrid (Large Hadron Collider Grid)
- Fraunhofer Resource Grid
- NASA Energy Grid
GT and the GSI

- Globus Security Infrastructure
- Collection of security-related protocols and services within the GT
  - CAS
  - MyProxy
  - SimpleCA
Identification in Globus

- X.509 end-entity certificates
- X.509 proxy certificates (RFC 3820)
  - Single sign-on
  - Delegation
- Identity mapping
  - X.509 vs Kerberos vs UNIX accounts
CAS

• Community Authorization Service
• Problem domain: Distributed policy definition and enforcement
• Implementation in Java
Local site policy

FunkyVO is granted full file access to
/share/funkyvo/*
/share/lhcraw/*

FunkyVO policy

User Bert is granted read access to
/share/funkyvo/bert/*
/share/lhcraw/*

Option 1: Store both policies locally

Option 2: Store site policy locally, VO policy somewhere else
FunkyVO is granted full file access to /share/funkyvo/*/share/lhcraw/*

Local site policy
User Bert is granted read access to /share/funkyvo/bert/*
/share/lhcraw*

FunkyVO policy
User Bert is granted read access to /share/funkyvo/bert/*
/share/lhcraw*
FunkyVO is granted full file access to /share/funkyvo/*
/share/lhcraw/*

Local site policy

FunkyVO is granted full file access to /share/funkyvo/*
/share/lhcraw/*

Local site policy

FunkyVO is granted full file access to /share/funkyvo/*
/share/lhcraw/*

Local site policy

User Bert is granted read access to /share/funkyvo/bert/*
/share/lhcraw/*

FunkyVO policy

CAS server
CAS

• User authenticates to CAS server
• CAS server sends signed capabilities to user
• User generates new keypair and proxy cert with capabilities embedded
• User authenticates to resource
• User presents the proxycert+capabilities to resource
• Resource authorizes the required action or rejects it
Find the CAS server

• CAS server is a SPF
• Even more of a juicy target:
  • Get information on structure of VO
    • Participating hosts
    • Participating people
    • Existing resources
  • Create your own VO access rules
  • Get information on valid proxy certs
gridftp

- Extension of FTP to support striped and remote-controlled transfer
- Part of Globus
- Implementation in Globus based on wuftpd
gridftpd

- CVE-2004-0185 – wuftpdp

```c
char *skey_challenge(char *name, struct passwd *pwd, int pwok)
{
    static char buf[128];
    ...
    if (pwd == NULL || skeychallenge(&skey, pwd->pw_name,
                                         sbuf))
        sprintf(buf, "Password required for %s.", name);
    else
        sprintf(buf, "%s %s for %s.", sbuf, pwok ? "allowed" : "required", name);
    return (buf);
}
```
char *skey_challenge(char *name, struct passwd *pwd, int pwok)
{
    static char buf[128];
    char sbuf[40];
    struct skey skey;

    /* Display s/key challenge where appropriate. */
    if (pwd == NULL || skeychallenge(&skey, pwd->pw_name, sbuf))
        sprintf(buf, "Password required for %s.", name);
    else
        sprintf(buf, "%s %s for %s.", sbuf, pwok ? "allowed" : "required", name);
    return (buf);
}
Code Quality Examples

• MyProxy
  • Sloppy malloc return code handling

• GRAM
  • More solid code
Things to do in long winter nights

- Proper code audit of Grid middleware
- Architecture review of Grid middleware
- Architecture & implementation review of Grid applications
- Puzzling over unsolved fundamental problems
- Investigate implications of “service-oriented architecture” fancy
References

• The Grid 2. Blueprint for a new Computing Infrastructure, Foster et al

• http://www.globus.org/

• Hacking the Grid, talk given at 5th Hope by Greg Newby