SNE Security & Privacy

Guido van 't Noordende
guido <at> science.uva.nl

room f2.44

Grids

- Networks faster and bigger / more bandwidth
- Easier and cheaper to tie multiple computers together
- Connect multiple clusters by fast networks and you have a Grid
- Useful for running compute and data intensive (distributed) applications
- More compute power available than ever
- Applications: physics, biology, medical / imaging, ...

However...

- Grids are (widely) distributed over multiple administrative domains
- Grids/clusters are heterogenous
  - Multiple OS-es, multiple middleware installations/versions, different configurations, multiple administrators, ...
- Are grids secure?

Basics

- Example: UNIX Security model
  - UNIX: r/w/x bits, user/group/other
  - R is for *confidentiality*: who may read? -> privacy
  - W is for *integrity*: who may write/change?
  - Simple to understand
  - sufficient for most cases
  - UNIX kernel is reference monitor
  - Assume single system & administrator trusted
  - (relatively small) TCB

Grid (security) evolution

- Distributed resources (disks, clusters, ...)
  - Distributed management/owners/admins
- (meta)schedulers dispatch jobs to clusters
  - e.g., WMS, ...
- Public key cryptography / PKI: GSI
  - Hosts can be authenticated using 'host certificates'
  - Jobs can be authenticated using proxy certificates (signed by 'user key')
  - Strong PKI backbone: Grid CAs, ...
- Not so small TCB

Anything wrong?

- What do we *trust?*
- Host certificate says nothing about the host
  - Nor its administration..
  - Nor its configuration..
  - Nor its (past) users..
  - Nor its physical safety..
  - Nor its vulnerabilities..
- Job - proxy certificate binding flawed
  - Are we sure this certificate belongs to this job?
  - Jobs can be hijacked / modified..

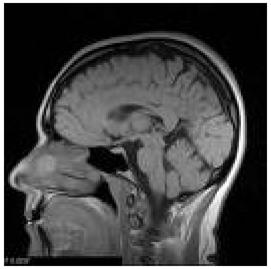
Privacy (confidentiality)

- Needs the *system* (host) to enforce confidentiality
- What is 'the' system in a Grid?
  - Who owns/manages storage?
  - Where is the job?
  - Do we have any control as a data owner?
- Risk assessment
  - privacy/security requirements
  - Can a data owner trust the system?

Privacy Sensitive Applications

- Industrial apps, (bio)medical apps
  - (DNA, imaging, ...)
- Medical requirements (personal information)
  - Data protection regulations (95/46/EC)
  - Purpose binding / necessity / minimality
  - Consent for medical research data: purpose bound
  - Physician *legally responsible* for ensuring an *appropriate level* of security to protect data
- Similar laws / regulations in non-EU countries
  - e.g., U.S. HIPAA, PIPEDA, ...

Medical apps



- Can we anonymize data?
  - DICOM header / strip names;
  - MRI data – images
  - DNA data/sequences (BioBanking)
- Can we control distribution?
  - Replication policies (TSRB: storage system constrains which clusters can obtain data)
- Can we control/avoid copying?
- Can we control/avoid data leakage at all?

Trusted Storage System: TSRB

- ACL: what system can a job access data from
- Systems trusted by data owner:
  - Administrator/domain trusted (ACL)
  - Has safe configuration (HPL)
    - up-to-date config, /temp cleaning, encrypted swap, etc
  - job/certificate binding verification
  - Defining ACL/HPL manual task...
  - Microcontracts for auditing
- Assumes that if job owners are trusted -> jobs are trusted

Can we trust jobs?

- Most jobs are binary programs
- Can access local FS, invoke GridFTP, set up sockets to outside world,...
- Can contain back-doors (trojan horse), may even be unknown to researcher who submitted the job
- Risk assessment exercise: *if you were a hospital director, would you trust the jobs that medical researchers submit to the Grid?*

Can we trust jobs?

- Most jobs are binary programs
- Can access local FS, invoke GridFTP, set up sockets to outside world,...
- Can contain back-doors (trojan horse), may even be unknown to researcher who submitted the job
- Risk assessment exercise: *if you were a hospital director, would you trust the jobs that medical researchers submit to the Grid?*
  - What if you get sued?

Can we trust jobs?

- Most jobs are binary programs
- Can access local FS, invoke GridFTP, set up sockets to outside world,...
- Can contain back-doors (trojan horse), may even be unknown to researcher who submitted the job
- Risk assessment exercise: *if you were a hospital director, would you trust the jobs that medical researchers submit to the Grid?*
  - What if you get sued?
  - Where does this leave TSRB?

Good news

- Good news is: perhaps we *can* assume submitted programs to be trustworthy
- We could require jobs signed by their authors
  - e.g., Sun, Microsoft, Linus, ...
- We can *jail* a job to avoid all obvious outbound channels
  - Control all actions by job at syscall entry point
  - disallow connects, only write to a temporary directory,
  - do something smart at job exit time
    - e.g., encrypt all written data using job owner's public key

Wrap-up for Grids

- We can harden Grids or the Grid's systems
- Ensure enforcement of policies and auditing such that Grid security gets tractable
- Implement mechanisms (e.g., confinement) that limit chances of abuse by unknown parties somewhere in the 'food chain'
  - e.g., software coders, compiler writers,
  - OS vendors / distributors, system administrators
  - The regular bunch of mistakes or vulnerabilities

But..

- Security does not come for free:
  - Sacrifice usability, cause inconvenience
  - Less performance than without security
  - Require sometimes difficult configuration
  - Manual work, risk assessments, ...
  - Require up-to-date systems and alert administrators
  - Solutions may not suit all applications
  - Current protocols need to be changed

Conclusion

- Yes, we can harden (distributed) systems
- But: still a lot of work to do..

Related (privacy) work

- Calculate / measure privacy or anonymity
  - In networks / overlays (TOR, anonymous remailers)
  - Entropy / information theory based calculations
  - K-Anonymity / re-identifyability of data sets
- Research existing systems/security
  - EPD, OV-chipcard (security + privacy)
  - Composition of systems containing data (UK: “database state”, how about NL?)

Summary

- SNE: research on networks, security, privacy
- Network security solved to some extent (SSL)
- Higher layers are more complex, security in distributed systems hard to achieve,
- need to trust many many components in many many places
- Better protocols and solutions still needed
- More end-user control, auditing, ...
- For privacy, also composition / properties of combined systems to consider

Contact

- Guido van 't Noordende
- Guido <at> science.uva.nl
- f2.44