Risks in Anonymous Distributed Computing Systems

Michael J. Ciaraldi

David Finkel

Craig E. Wills

Worcester Polytechnic Institute

Worcester, MA 01609 USA

Presented at

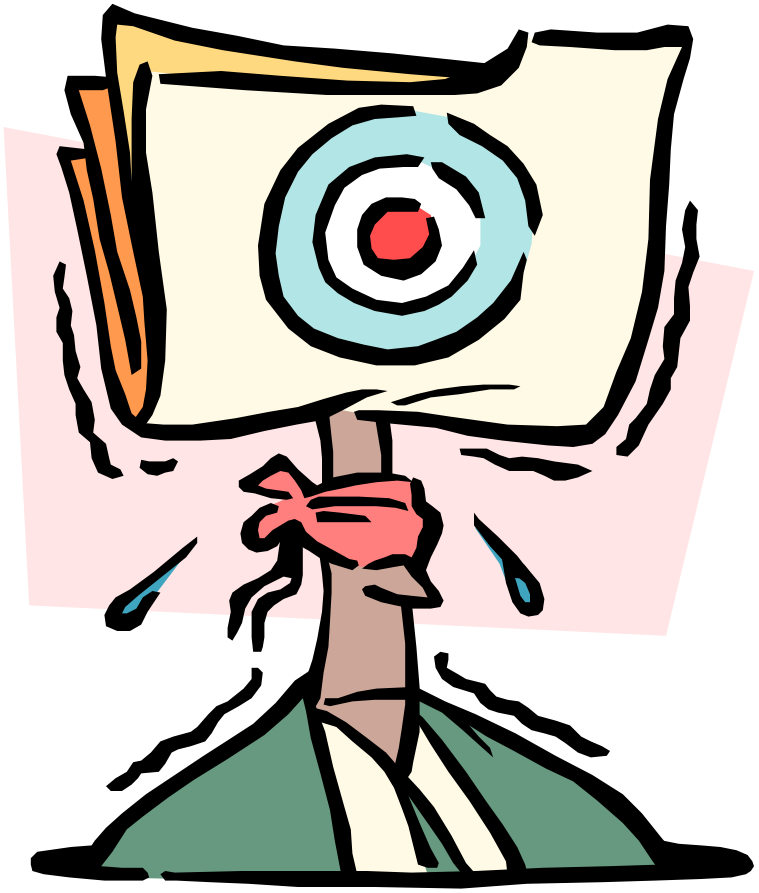
International Network Conference 2000

Plymouth, England

Overview

- Anonymous Distributed Computing Systems
  - What are they?
- What are the risks?
  - Most are well-known
  - ADCSs face some unique challenges.
- Which risks can be addressed, and how?

Anonymous Distributed Computing Systems



Distributed Computing Systems

- Traditional *vs.*
- Anonymous

Traditional Distributed Systems

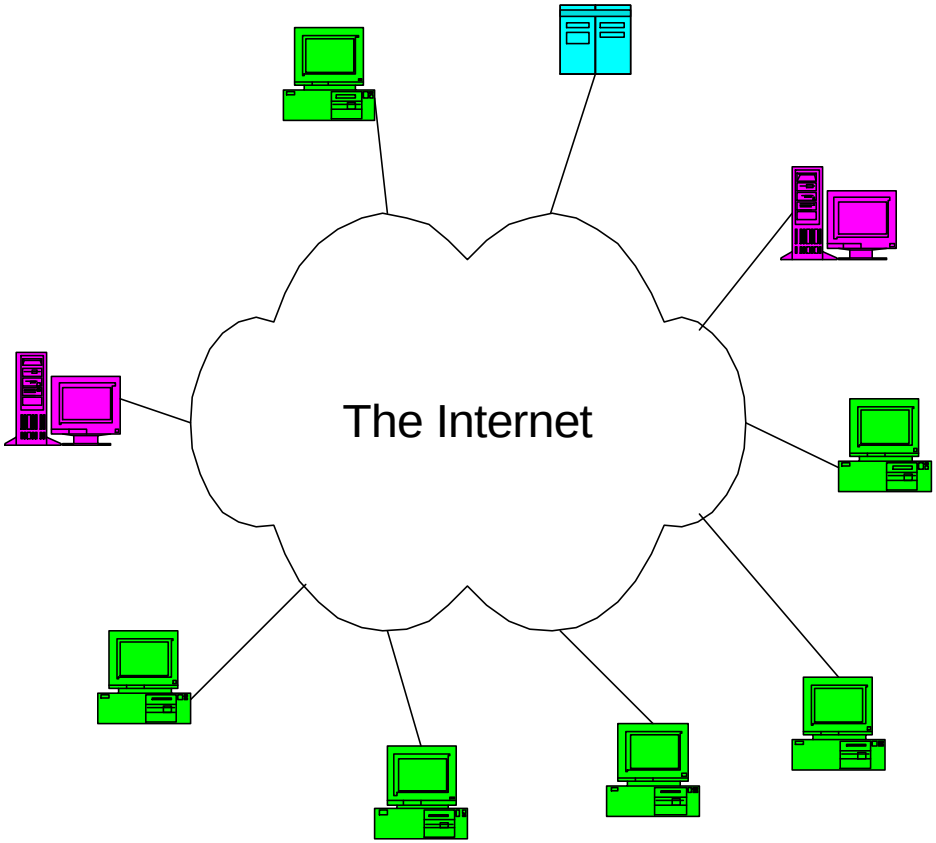
- Autonomous systems
  - Standalone machines
  - Explicit Services with explicit authorization
    - telnet, ftp
- Distributed operating systems
  - Appear as a single virtual machine
  - Single administrative domain
- Network file systems
  - Shared resources
  - Single administrative domain

Anonymous Distributed Computing Systems

- Types of Nodes
- Characteristics
- Approaches

Types of Nodes in ADCS

- Distributor nodes
  - Distribute pieces of a calculation.
- Client nodes
  - Execute pieces and report back to distributor.
- Portal nodes
  - Direct clients to distributors.



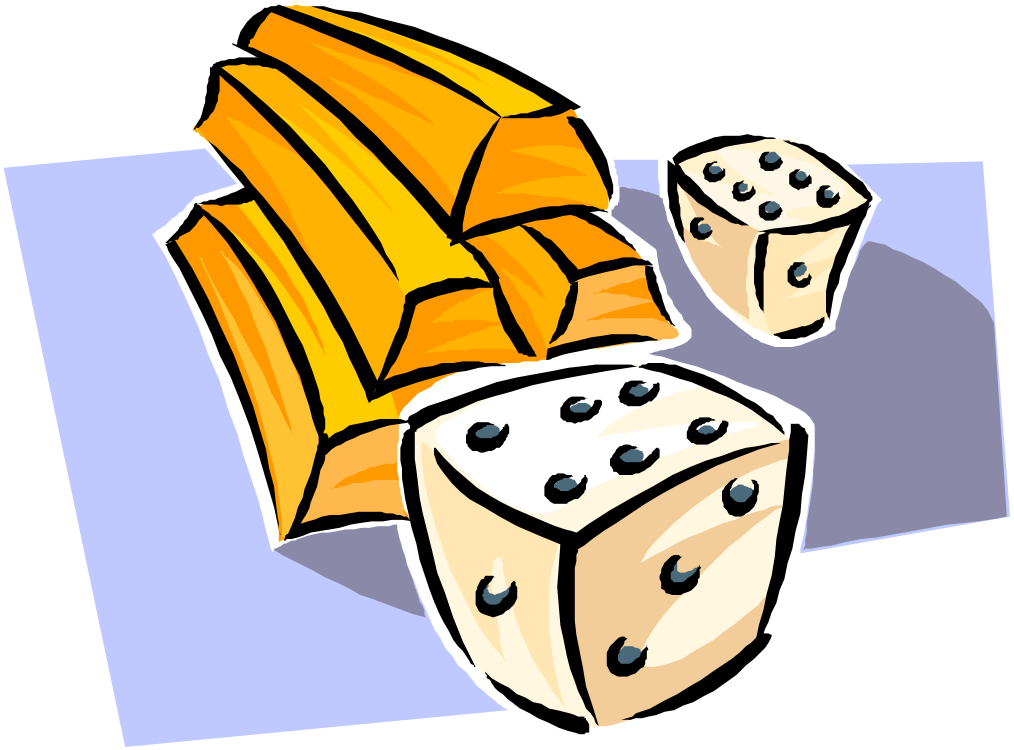
Characteristics of ADCS

- Potentially millions of nodes.
- Client nodes vary in power and architecture.
- Clients controlled by different administrative domains.
- Clients may be unaware of each other.
- Clients not always available for ADCS.
- Internet communications unreliable and at various speeds.
- Clients may crash or withdraw at any time.
- A client may be in several ADCSs.
- Clients may volunteer or be paid (micropayments).

Approaches in ADCS

- One-Time Download:
  - Just once, client downloads an executable program from a portal.
  - To participate, client program contacts portal.
  - Examples:
    - SETI@home, distributed.net
- Each-Time Download:
  - Client downloads Java applets or ActiveX controls each time.
  - Examples:
    - POPCORN, Charlotte, distriblets

Risks



Risks

- Where are they?
- What are they?
- Can they be reduced or eliminated?
  - By technology?
  - By human diligence?

Types of Risks and Where They Occur

- Internet Communication
  - Inherently unreliable
  - Passes through others' machines
    - Can be intercepted and/or altered.
  - Anonymous
    - What is the sender's true IP address?
    - Who is the sender, anyway?

Types of Risks and Where They Occur II

- Knowing identity of distributor
  - Recommended by others
  - Confidence that software is not harmful
    - To client
    - To others, e.g. DoS, cracking.
  - Accountability
- Knowing identity of client
  - Confidentiality
  - Payment
  - Invalid results

Dealing With Risks

Dealing With Risks

- Communication problems
- Malicious client code
  - Attacks the client or another machine.
- Counterfeit client code

Accidental Communication Problems

- Checksums guard against corruption.
- Timestamps guard against stale data.

Deliberate Communication Problems

- IPSec
  - Provides encryption and authentication end-to-end.
  - Guards against interception and/or modification *en route*.
  - Is only a protocol.

IPSec Is Not Enough

- ADCSs must use asymmetric (public key) encryption.
- This requires knowing the public key of the other party.
  - Or whoever the other party claims to be.
- To confirm the key, use a digital certificate from a Certification Authority (CA).

Problems with Certification Authorities

- Can the CA be trusted?
  - Could be run by an unethical organization.
  - Employees could be corrupt.
- Can the CA guarantee the identity of the entity?

Problems with Certification Authorities II

- Can the entity be trusted to be non-malicious and competent?
  - Can all its members?
- Certificates expire and are revoked
  - But not instantaneously.
- These are primarily human problems, not technological.

Malicious Client Code

- Mechanism:
  - Screen savers and ActiveX controls vs.
  - Java applets
- Examining source code

Screen Savers and ActiveX Controls

- Could be
  - One-time download (screen saver)
  - Each-time download (ActiveX)
- Privileges
  - Essentially unlimited in MS-Windows.
  - Can be limited by careful installation in Unix.

Java Applets

- Execute in a “sandbox” with limited privileges.
- Can still:
  - Open windows
  - Send email with your return address
  - Consume system resources.
- Can only open a network connection back to the download server.
  - Cannot directly participate in distributed attack.
  - Limits parallelism.

Examining Source Code

- Who is competent to examine it?
- You have to send the source code.
  - Confidentiality?
  - How to guard against counterfeit code?

Counterfeit Client Code: Why?

- Maliciousness
- Competition
- Denial of service
- Payment for services not rendered.

Counterfeit Client Code: Possible Defenses

- Possibilities suggested by Popcorn:
  - Send the same computation to several independent clients.
    - Widely applicable, but expensive.
  - Check the answers.
    - Less expensive, but not as applicable.
  - Are the resources spent on checking greater than those gained by parallelism?

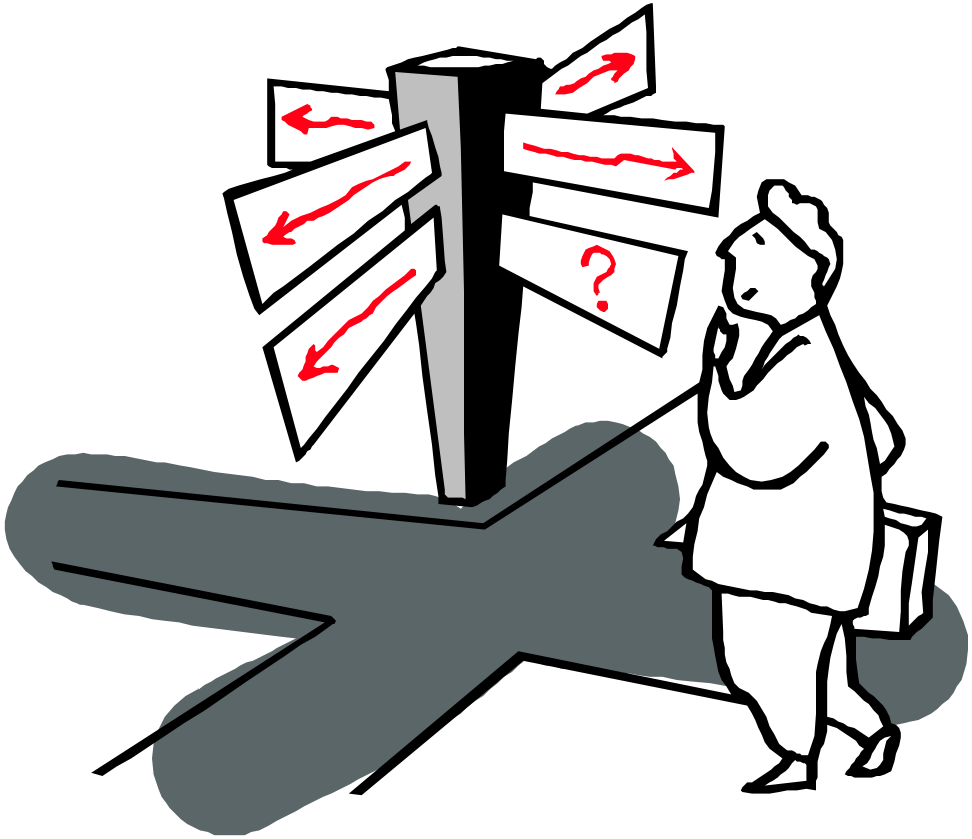
Counterfeit Client Code: Other Possible Defenses

- Challenge-response authentication.
  - Is it possible?
  - Reverse engineering?
  - Could a Trojan Horse later corrupt or replace the client code?
- Nonces
  - Cause authentication to expire.

Risks Facing Portals

- Connecting through a well-known central portal is no guarantee of safety.
  - Computations still come from third parties.
  - Portal operators can identify computation sources, but not their safety.
  - Portal operators cannot determine what all their clients will consider ethical.
  - Portal operators must exercise due diligence, but this may not protect them from liability.

In Conclusion



Summary

- ADCSs are attractive.
- They present many risks, some unique.
- Some of these risks:
  - Have technological solutions.
  - May have human solutions.
  - Have no currently-known solution.
- So, keep thinking!
- The ultimate test: will users be deterred?