IRODS Security

Wayne Schroeder
Data Intensive Cyber Environments Team (DICE)
Institute for Neural Computation (INC), University of California San Diego

irods.org, dice.unc.edu, diceresearch.org
Outline

- General Comments
- What Guarding Against
- Authentication
- Trust Model
- IRODS Counter-Measures
- Administrator Responsibilities
- Future
Overview

- Computer Software Never Completely Secure
  - Ease-of-use vs. security
  - Ease-of-Implementation, cost/benefit
  - Encryption time
  - Attacks/Counter-Measures

- Open Source Tends to be More Secure
  - Vulnerabilities must be Handled Responsibly
  - Needs to be Collaborative
What We Are Guarding Against

- IRODS Does What It Should
  - Users Are Who They Say
  - Access Controls Enforced (Read/Write)
  - Resist Denial-Of-Service Attacks
  - Resist SQL Injection Attacks
- Host OS Remains Secure
Protect OS

- Running as non-root helps
- Buffer Overflows Avoided
  - Rstrncpy, etc
- Open Source
Authentication

- IRODS Password/GSI/Kerberos Network Secure
  - Have to be

- Keys Can Be Stolen and Used
  - Host/NFS Needs to be Secure
  - GSI Credentials Time-Limited

- IRODS Credentials
  - Not Plain-text Credential (iinit)
  - But Source to Unscramble Is Open
  - NFS May Expose on Network
Trust

- Client Code Not Trusted
  - Can't be (Network Often Not Secure)

- Server Code Is Trusted
  - Has To Be

- Micro-Service Is Server Code
- IRODS Admins Are Trusted
- ICAT DB/Admins Are Trusted
Some iRODS Counter-Measures

- Buffer Overflow Checks Throughout
  - OSX 10.6 Noticed Some Inconsistencies;
  - Fixed in 2.3

- Client/Server Call (rc/rs) Privilege Levels
  - Some Admin-only (e.g. chSimpleQuery)

- Server/Agent Fork/Exec Mechanism
  - Planned Addition of Multi-Threading

- Use of Bind-Variables
  - DB Treats as Name; avoid SQL injection
IRODS Admin Responsibilities

- Keep Server Access Secure
  - Good passwords, OS Patches, etc
- Keep IRODS source code secure
  - Proper user-level access control
- Check Added Micro-Services
- Keep Passwords Secure
- Optionally:
  - Configure remoteZoneSID (man-in-middle)
  - User irodsServerDN if using GSI
Future Work

☐ Ongoing Security Analysis (UNC, Simon Spero)
☐ University Analysis U of Wisconsin (Barton Miller/James Kupsch)
  – Collaborative Project as done with SRB; Highly Effective
☐ Bug Fixes
☐ Continue On-Going; Balanced with Other Needs/Requirements
  – Enough For Most Instances
  – Without Placing Too Much Burden on Users/Admins/Developers