# SIMPLICITY AS A SECURITY PRINCIPLE

20th DOE CSG Meeting

28 April 1998

Lara H. Baker & David J. Bailey

# TOPICS

- Background
- Rationale
- Examples
- Conclusions
- Recommendations

# BACKGROUND

- “Of two possible solutions, try the simpler first”
  - Paraphrase of William of Occam. Often called “Occam’s Razor”.
- “Everything should be as simple as possible -- but no simpler”
  - Albert Einstein

# BACKGROUND

- “Complexity is troubling; hidden complexity is dangerous”
  - Many sources
- “What you don’t know CAN hurt you”
  - All flight instructors
- “Keep It Simple, Stupid”
  - All Sergeants

# WHY COMPLEXITY

- Creeping Feature-itis
- “Since you’re doing that anyway, why don’t you . . .”
- “The computer can do it, so why don’t we do it?”
- It seems easier to sell -- i.e., “More bang for the buck”

# COMPETING IDEAS

- “Trusted Systems”
  - Build a complex system that meets a complex set of rules, and prove that the rules and the implementation are correct.
- Sequestered Servers
  - Build a simple system that does a useful task, and prove it complete -- i.e., use an exhausting search to prove that it does nothing unexpected.

# SYNERGISTIC IDEAS

- Simplicity
- Layering
  - Simplification of interfaces

# SEQUESTERED SERVERS

- One machine -- one function
  - Hardware is free
- Separate Configuration and Operations
- Limit Interface(s) -- e.g., use NIU's
- Limit Operator Actions
- Exhaustively Test

# SEQUESTERED SERVERS

- Consider Layering Machines -- i.e., Use 2 or More Machines in Series
- Layered Internal Protections
  - No Extraneous Daemons Running
  - No Un-needed Software on System, Including Utilities
  - Use Firewall Rules to Preclude Connections
  - Immutable Files

# SIMPLE can equal MORE ROBUST

- Attacking a large, complex software system
  - e.g., sendmail
- Attacking the interfaces among software systems.
  - Time-of-check vs. time-of-use
- Cracking LTSS
  - “Is that all there is?”

# EXPERIENCE

An alert, aware user community is the best defense you'll ever have.

# APPLY SIMPLICITY

- System Design
- Implementation
- Verification and Accreditation
- Operation

# Reality Check

- “For every problem, there is a solution that is simple, obvious, and wrong.”
  - Murphy’s Laws, many sources

# LARA'S LAW OF LEAST LIARS

- In any situation with differing accounts from different sources, consider the smallest number of accounts one must ignore for all the rest of the accounts to be consistent.
- 1? 2? 3? 4+ No

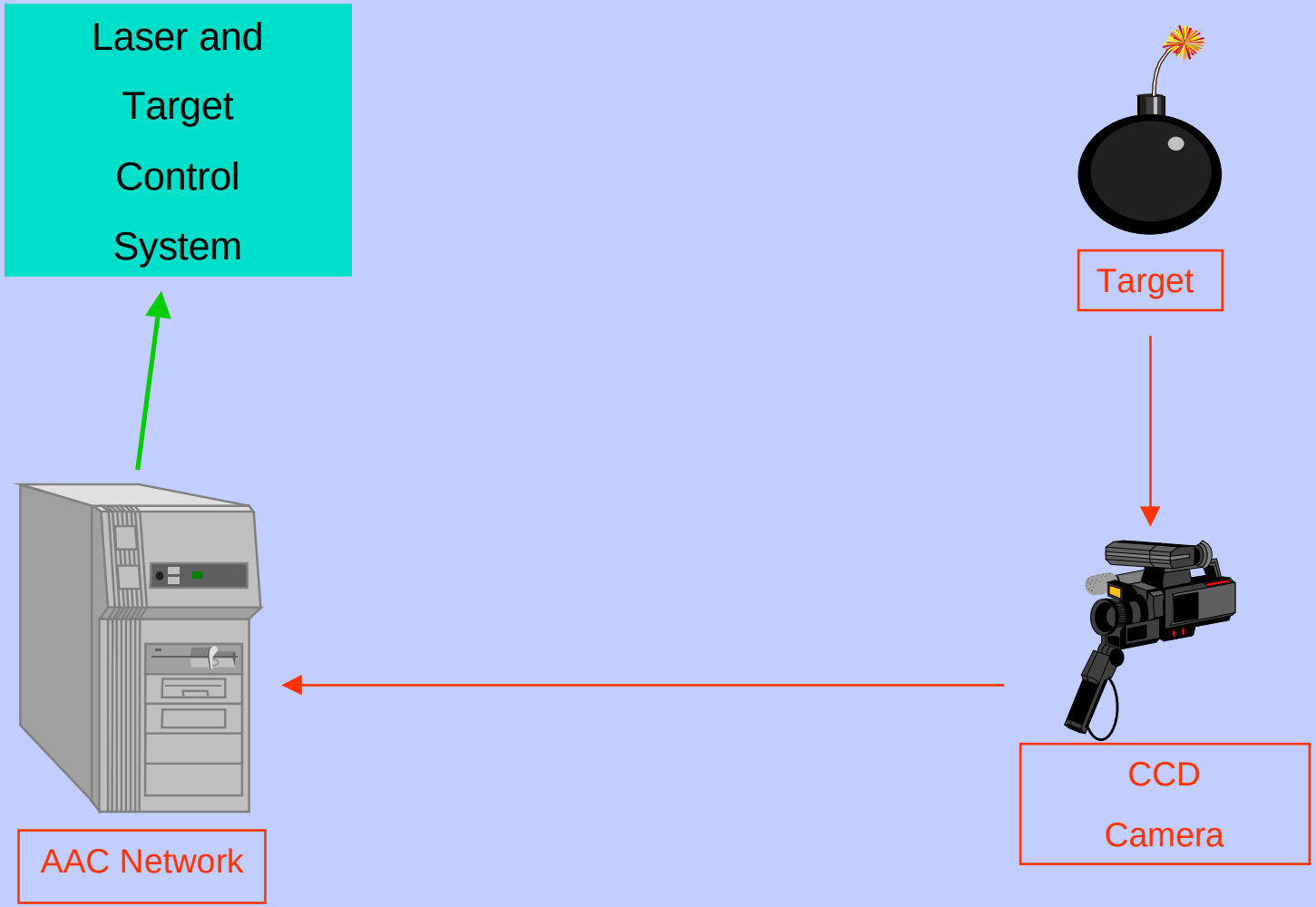
# DESIGN

- IV&V of design for moving data from Secret to Unclassified.
- Secure Xenix and Secure Oracle
- MLS
- Replacing Sneakernet
- Data Volume and Required Rate
- Simple Solution

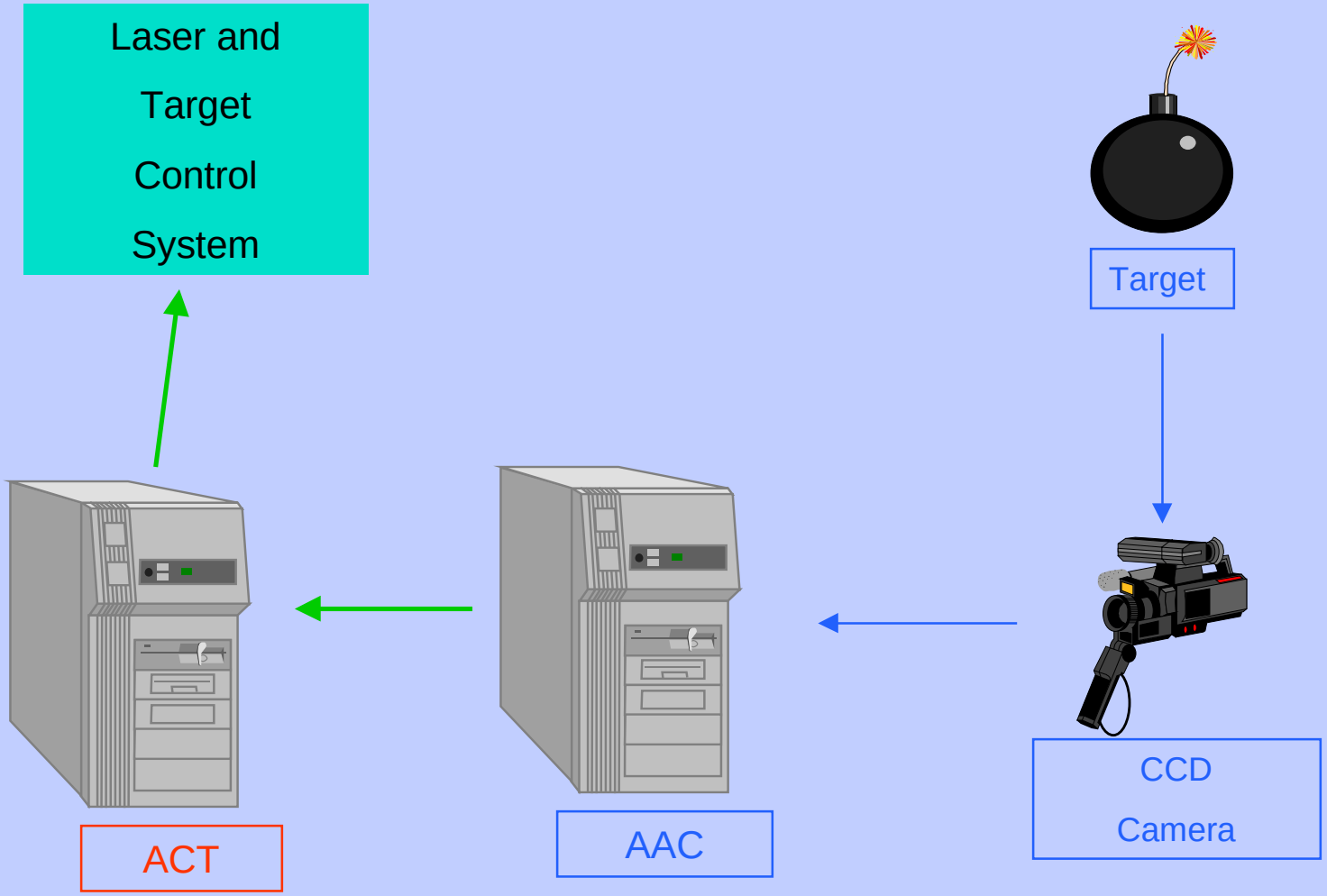
# IMPLEMENTATION

- LLNL Automatic Alignment System
- Simplification by Definition
- Bottom Line -- Changing a Multi-Level System is expensive and difficult. Avoid changes if possible; if changes are unavoidable, localize the changes as much as possible.

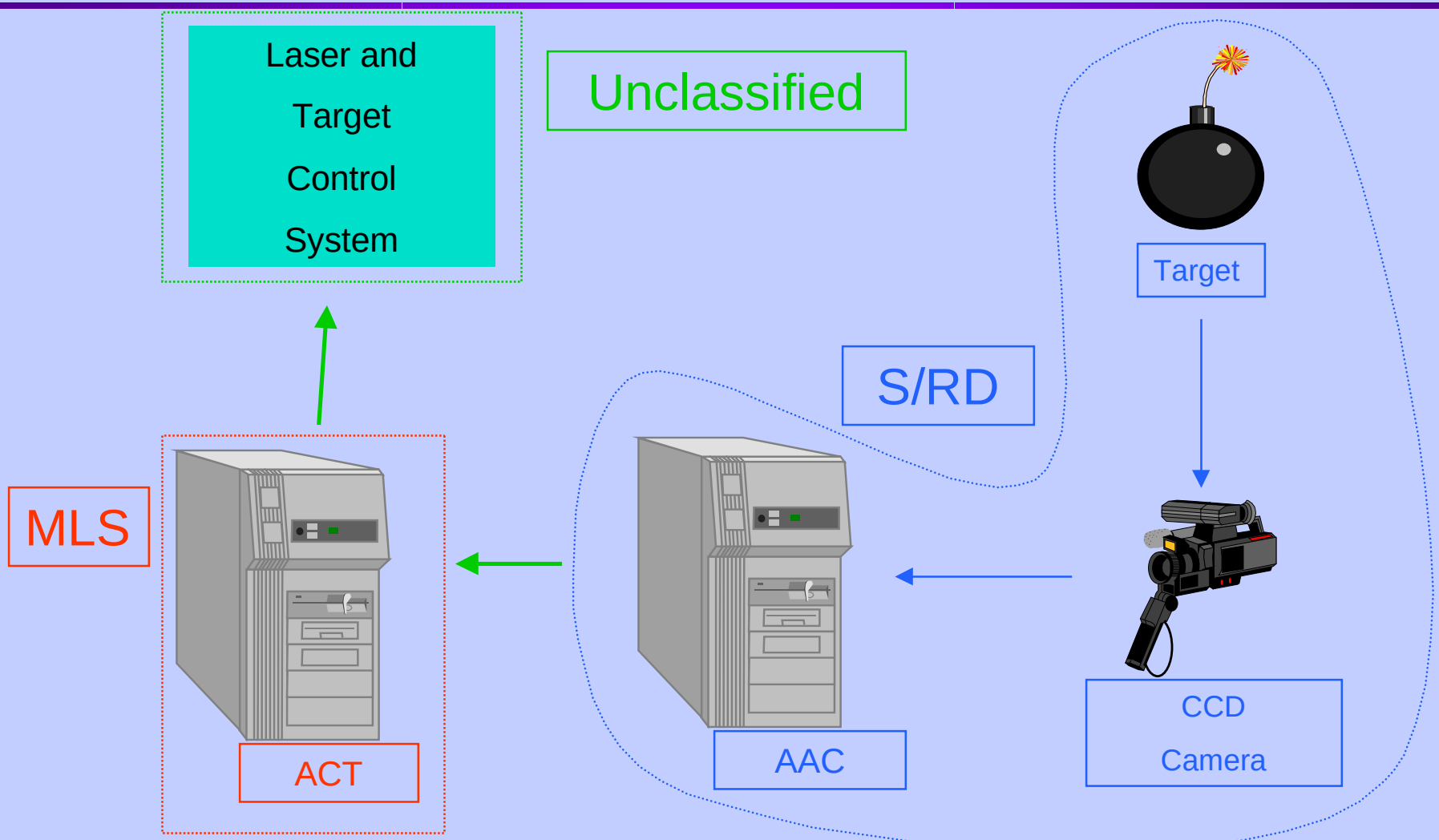
# Automatic Alignment Computer



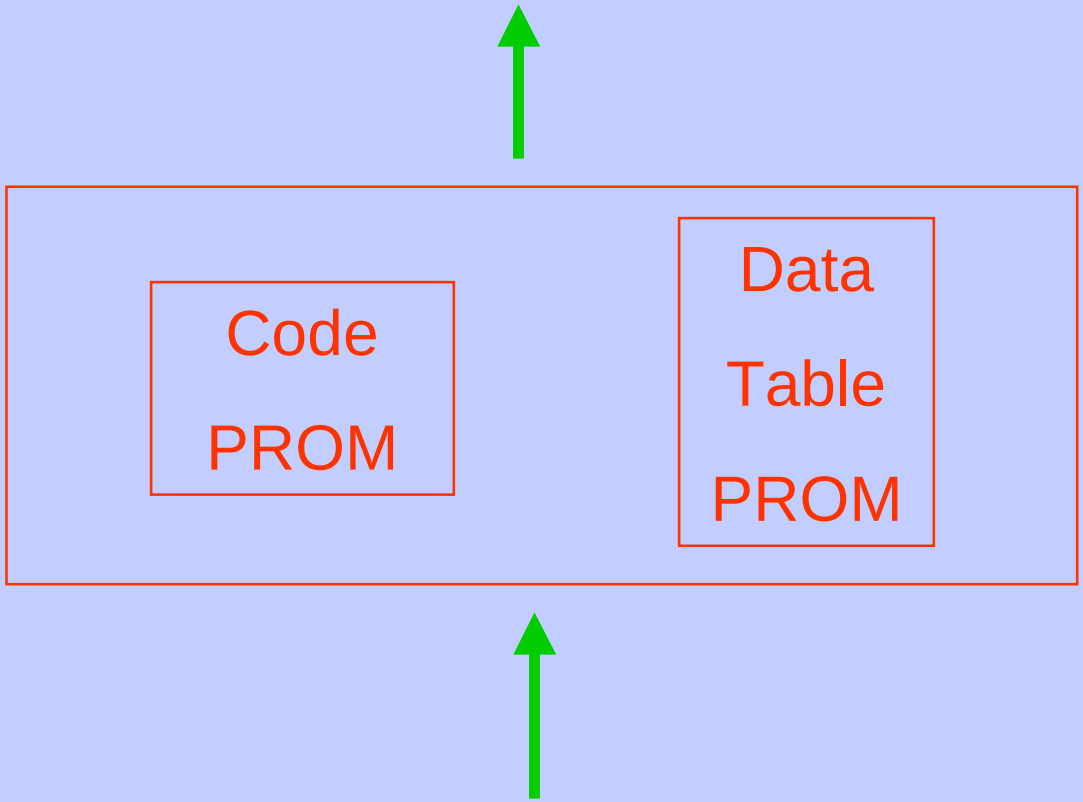
# Automatic Alignment Computer



# Automatic Alignment Computer



# Activator Control Translator



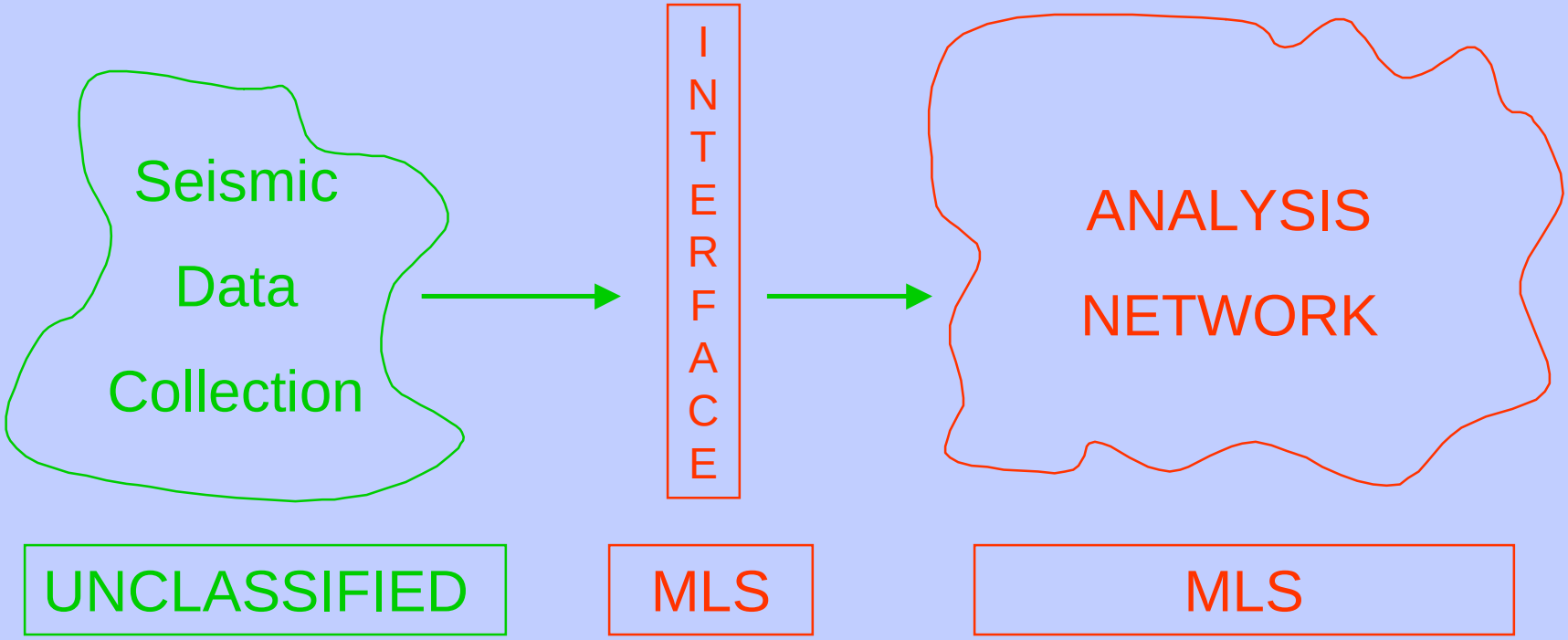
# AFTAC PROBLEM

- Large, world-wide seismic data collection system -- Unclassified
- Large, localized data analysis system -- Secret
- Automated, near-real-time transfer of data from the field to the analysis system -- ONE-WAY TRANSFER.

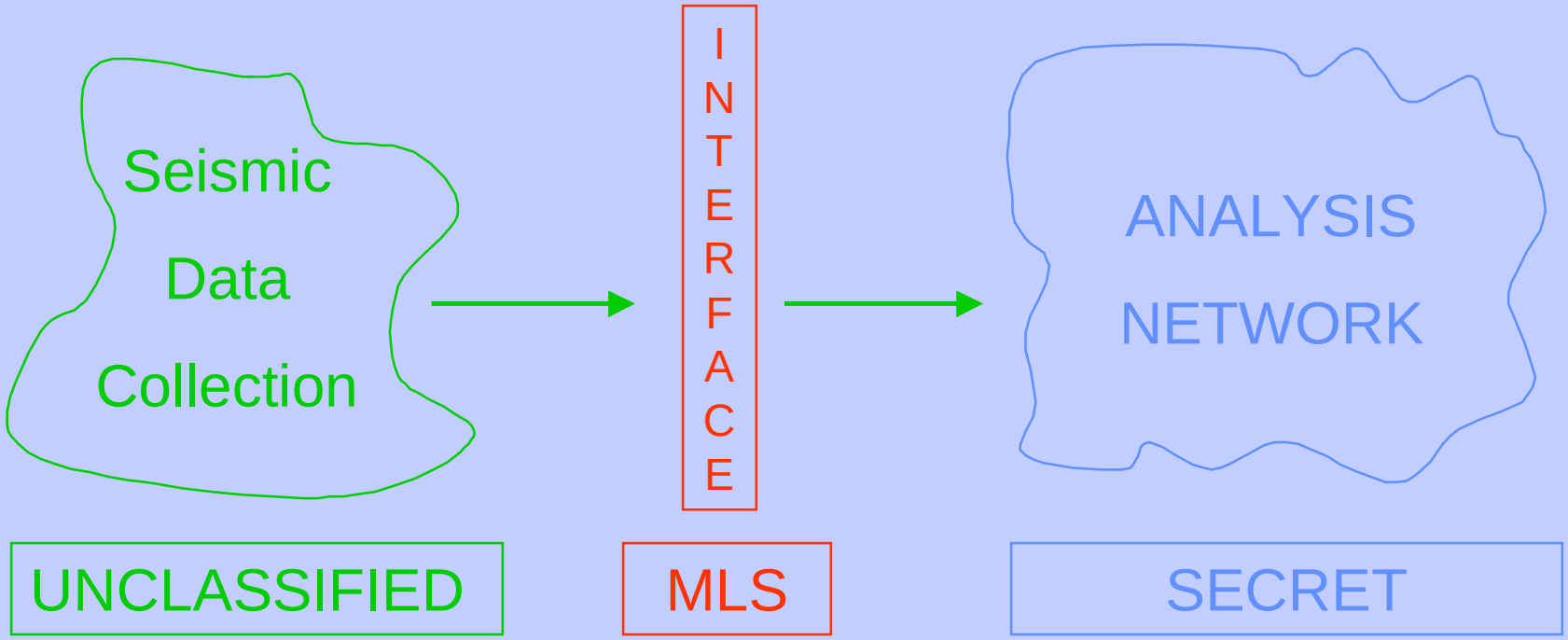
# DESIGN ASSUMPTIONS

- Source -- DOE Experience
- Any Machine That Runs User Code is **MALICIOUS** -- Accidents & Insiders
- Malicious Insiders DO Exist
- Machines Do Break
- Complexity is Troubling
- Hidden Complexity is Dangerous

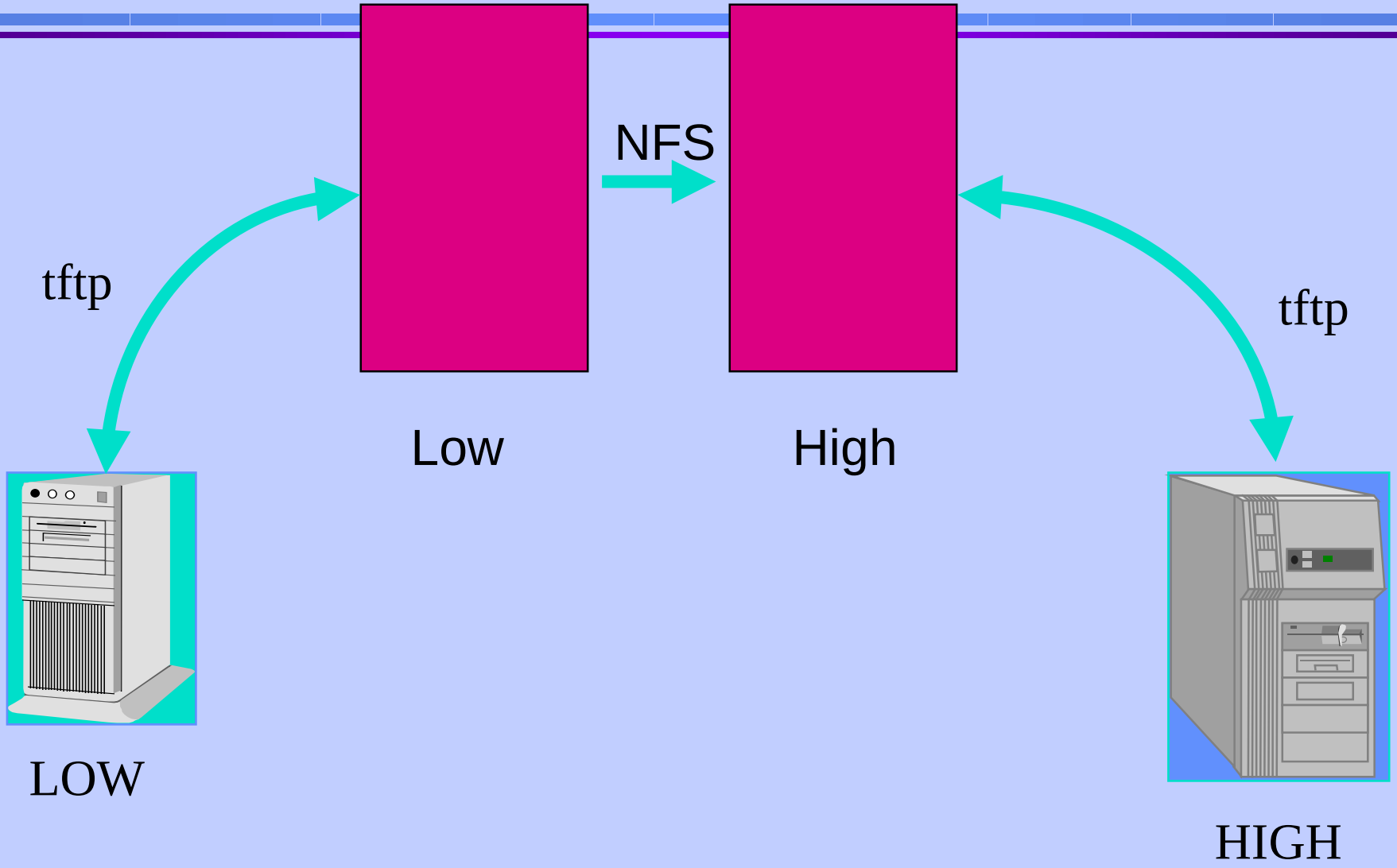
# AFTAC PROBLEM



# AFTAC PROBLEM



# INFORMATION DIODE STRUCTURE



# SYSTEM SECURITY POLICIES

- Most Systems
  - "That Which Is Not Expressly Forbidden, Is Permitted."
- Most "Firewalls"
  - "That Which Is Not Expressly Permitted Is Forbidden."

# THE DIODE's POLICY GOAL

- "That Which Is Not Expressly Permitted Is Impossible."

# OPERATIONS

- In general, Operators can/will make mistakes.
- Operators can’t make mistakes if they can’t do anything except what you let them do.
- Operator’s shell is */bin/false*

# CONCLUSIONS

- Security and Complexity are antagonistic
- Simplicity pays many dividends
  - » Evaluation, Certification, and Accreditation are much easier.
  - » Fewer Operational Surprises
  - » Easier to explain to management
- Simplicity works better!

# RECOMMENDATIONS

- Keeping It Simple is Smart
- Concentrate the “Multi-levelness” within as small a perimeter as feasible, then LEAVE THE SYSTEM ALONE.
- Don’t trust user machines
- Consider using Sequestered Servers

# SIMPLICITY AS A SECURITY PRINCIPLE

20th DOE CSG Meeting

Lara H. Baker & David J. Bailey
lara@gcsi.com,
daveb@gcsi.com