Collaborative Intrusion Detection and Response

Limitations of Monolithic ID
- Single point of failure
- Limited access to data sources
- Only one perspective on transactions
- Some attacks are inherently distributed
  - Smurf
  - DDoS
- Conclusion: "Complete solutions" aren't

Sharing Information
- Benefits
  - Increased robustness
  - More information for all components
  - Broader perspective on attacks
  - Capture distributed attacks
- Risks
  - Eavesdroppers, compromised components

Sharing Information
- Communication risks can be resolved cryptographically (at least in part)
- Defining appropriate level of expression
  - Efficiency
  - Expressivity
  - Specificity

CIDF
- Common Intrusion Detection Framework
  - Collaborative work of DARPA-funded projects in late 1990s
  - Task: Define language, protocols to exchange information about attacks and responses

CISL
- Common Intrusion Specification Language
  - Conveys information about attacks using ordinary English words
  - E.g., User joe obtains root access on demon.example.com at 2003 Jun 12 14:15 PDT
CISL
- Problem: Parsing English is hard
- S-expressions (Rivest)
  - Lisp-like grouping using parentheses
  - Simplest examples: (name value) pairs
    - (Username "Joe")
    - (Hostname "demon.example.com")
    - (Date "2003 Jun 12 14:15 PDT")
    - (Action obtainRootAccess)

CISL: Roles
- Clarifies roles identified by descriptors
  - (Attacker)
    - (Username "Joe")
    - (Hostname "carton.example.com")
    - (RoleID 501)
  - (Target)
    - (Hostname "demon.example.com")

CISL: Verbs
- Permit generic description of actions
  - (Compromise)
    - (Attacker "...")
    - (Observer)
      - (Date "2003 Jun 12 14:15 PDT")
      - (ProgramName "GRIDSDetector")
  - (Target "...")

CISL: Conjunctions
- Permit expression of compound events
  - HelpCause: Indicates partial causality
  - InOrder: Indicates sequencing
  - AsAwayOf: Indicates multiple views of the same attack

CISL: Open S-expressions
- Lambda calculus-like macros
  - (def CompromiseHost $1 $2 $3)
    - (Compromise)
      - (Attacker (Username "$1"))
      - (Target (Hostname "$2"))
      - (Observer (Date "$3"))
  - )}
CISL: Open S-expressions

- Originally defined to reduce payload
- Also usable for database queries
  - Look for all records matching 'Compromisehost'
  - Difficulty: Store expanded form or macro form in database?

Testing CISL

- CISL is expressive, leading to questions
  - Is it ambiguous?
  - Does a given GIDO have more than one interpretation?
  - Is it overbuilt?
  - Is there more than one GIDO that expresses the same thing (aside from reordering)?

Testing CISL

- GIDO Bake-offs
  - June 1999: Demonstration of simple corroboration
  - October 2000: Semantic testing
    - Group A: Devised scenarios/questions
    - Group B: Only knows scenarios, creates GIDOs
    - Group C: Only knows questions, receives GIDOs
    - Three levels: Easy, medium, gnarly

Lessons from CISL

- Lessons from testing, standardization efforts
  - Heavyweight
  - Not ambiguous, but too many ways to say the same thing
  - Mismatch between what CISL can say and what detectors/analysts can reliably know

Enter IDWG

- Intrusion Detection Working Group
  - WG of Internet Engineering Task Force
  - Chief product: IDMEF
  - Intrusion Detection Message Exchange Format
  - Driven by many CIDF participants

IDMEF

- XML-based; defines DTD for ID
- Reduced vocabulary
  - Roles reduced to analyzer (observer), source, target
  - Extra information for identifying exploits, buffer overflows
  - Provision for indicating that previous alerts are related
  - No provision for response prescriptions
IDWG Status

- IDMEF (and other IDWG drafts)
  - Submitted to IESG for advancement to IETF Draft Standard (as standards-track RFC)