Session 3 – Intrusion Tolerance

- R&D Directions in Intrusion Tolerant Systems
  - Carl Landwehr, NSF, USA
- The Design and Deployment of *COCA*
  - Fred Schneider, Cornell University, Ithaca, NY, USA
- Authorization Schemes and Intrusion Tolerance for Internet Applications
  - Yves Deswarte, LAAS-CNRS, Toulouse, France
- An Adaptive Intrusion-Tolerant Server Architecture
  - Victoria Stavridou, SRI, Menlo Park, CA, USA

COCA

- Intrusion-tolerant certification authority (a TTP) based on masking
- Two idempotent operations supported: query/update
- Non-assumptions (assumptions are Achilles heal of ITS): window of vulnerability; fair links; asynchrony
- Dissemination Byzantine quorum system: $n \geq 3t+1$; quorums $\geq 2t+1$
- Pb: mobile virus attack => pro-active secret sharing
  - rejuvenation at the extreme: shares, keys, state all refreshed
  - local clock at some server initiates refresh
- Key management
  - service public key never changes, it is only reshared
  - server public key not known to clients: no scaling problem, but clients cannot authenticate server responses
  - clients cannot determine whether a request has been processed by a quorum => t+1 delegates collect responses
- DOS defenses
  - increase cost of making a bogus request (eg. sign all requests)
  - decrease cost/impact of processing a bogus request

Authorization

- Authorization contributes to protection:
  - error detection/confinement
  - intrusion prevention/confinement
- Focus:
  - "principle of least privilege" to counter "abuse of privilege" attacks
  - "need to know" disclosure of personal information for privacy
  - generalized multi-party interactions, rather than just client/server
- Two-level authorization scheme:
  - composite operations authorized by IT authorization server (a TTP) (masking)
  - elementary operations checked by local reference monitors
    - local dispatcher (on untrusted host)
    - local security kernel (on tamperproof JavaCard)
- Permissions
  - for object methods: using capabilities and vouchers (signed permission list)
  - for composite operations: using tokens (COP equivalent of a method capability)

Intrusion-Tolerant Servers

- Intrusion detection hard; intrusion tolerance (masking) easier
- Reusable, specializable architecture
- IDS to determine alert level, not as prerequisite to recovery
- Principle: proxies placed between clients and servers
  - proxy leader accepts and filters requests
  - forwards requests to servers depending on agreement regime
  - proxies check results
  - if no agreement is reached, agreement regime is adjusted
  - auxiliary proxies monitor leader
- Implementation using diverse COTS, network IDSes + specific proxy code
- Challenge-response protocol (preemptive error detection)
- Runtime-verification monitors to self-check proxies
- Validation
  - diversity of mechanisms employed (IDS on int/ext networks, c-r protocol…)
  - performance (measurements)
  - resistance to attacks (test against known exploits, formal verif., red teaming)
  - information assurance case assembling claims, evidence (product/process/codesign) and arguments

ITS Workshop at DSN 2002

- 5 ITSes => red team critique; green team response
- ☹ Complexity, new vulnerabilities
- ☹ Reliance on good IDS / firewalls ; Vulnerability to DoS attacks
- 😐 Diversity:
  - ☺ temporal, crypto, spatial, defense mechanisms
  - ☺ shorter time to market than high-assurance non-diverse approach?
  - ☹ quantification? difficult to administrate? cost/benefit?
  - ☹ problem of exact comparison? (less of problem for *infrastructure* diversity)
- ☺ Separation of control/data channels (cf. Bob Gleichauf's comments)
- ☺ Randomization, camouflage
- 😐 Assurance
  - explicit assurance arguments rare
  - modeling attacks? what distributions?
  - resistance to automated (high-speed) attacks? resistance to stealth attacks?

Research Issues

- Appropriate assumptions? (cf. Achilles heel).
- Security policies: particular business process / application? Expression?
- Architecture:
  - intrusion masking (vs. detection)?
  - appropriate responses (fault treatment)?: shut down, isolate, reboot… but what about adaptation?
  - relaxation of ACID properties in ITDB and similar architectures?
  - coherent, analyzable intrusion tolerant system architectures?
- Assurance
  - limits of IT approach vs. "high grade" security?
  - modeling attackers/attacks?
  - quantification of benefits vs. cost (cf. diversity)
  - assurance arguments for ITS vs. those for safety-critical systems?
  - survivability of critical "business process" as a whole?
- Way forward: large scale system demonstration and red-teaming?