Quantitative Approaches to Software Security & Information Privacy
Rachel Greenstadt, David J. Malan, Stuart E. Schechter, Michael D. Smith
http://www.eecs.harvard.edu/securitas/

Core Proposition: Address the threat that malicious individuals pose to the security of software systems and personal data not only as a technical problem but also as a human, and specifically an economic, problem.

Economics of Software Security
The Security Risk faced by a system in running a software package is a function of:
1) the number of potential adversaries,
2) the adversaries’ incentive to attack,
3) the risk posed to the adversary of attacking the system,
4) the time, effort, and other resources required in a successful attack.

To increase (4) is to increase the Security Strength of the software.

By determining the expected cost to find a vulnerability in a software system, we can begin to measure Security Strength.

By offering rewards for vulnerability discovery, software vendors can measure the security strength of their software using markets.

Avoiding False Positives in Behavior-Based Worm Detection

Previous Approaches
Network-based: Detect scanning worms by counting the number of connection failures. Trigger alarm when connection failures exceed threshold set high enough to avoid false positives (e.g., 100).

Host-based: Detect fast-spreading worms by comparing a host’s current actions against its prior actions.

Advances
Detect scanning worms by measuring success-to-failure ratio of outgoing connections to new hosts. Use sequential hypothesis testing to trigger alarms based on strength of the evidence. Alarms triggered in as few as ten outgoing connections with very few false positives.

Detect fast-spreading worms by comparing current actions against peers’ current actions. We find that two peers, upon exchanging snapshots of their internal behavior, can decide that they are, more likely than not, both executing the same worm between 76% and 97% of the time.

A Framework for Comparing Models of Information Privacy

Our framework evaluates privacy models based on:
1) Decision-making – deciding what information is worth protecting and controlling;
2) Negotiation – reaching agreements about the use of the protected information;
3) Enforcement – assuring that all parties abide by the negotiated rights.

Example insight: No existing model adequately enforces privacy rules and audits privacy practices, but solving this issue is not sufficient.