Workshop on Measuring Assurance in Cyber Space

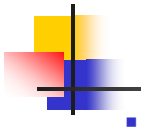
A few comments about today's sessions

Paulo Veríssimo

- take manageable bites out of the problem space
- *** would suggest an overarching fault model to structure these reductions, i.e. how vulnerabilities, attacks, intrusions, develop, and their inter-relationship

- Taxonomy:
  - attack-centric - identify attacks
  - defence-centric - manifestations, consequences
- *** as a matter of fact, i would say cause-centric and consequence-centric
- *** then, a structured fault model would help here, because what you are looking for is the last fault in the chain and/or the resulting error (not the primordial vulnerabilities, or the subsequent attacks)
- *** most taxonomies are devoted to faults and fault diagnosis, and maybe today we should be looking at taxonomies devoted to errors and error detection, or at least at ``ultimate fault'' diagnosis, if not at real error detection

- security properties under a formal methods perspective describe them as invariants
- *** yes, as (logical) safety properties, but also as (timeliness) safety, and liveness properties



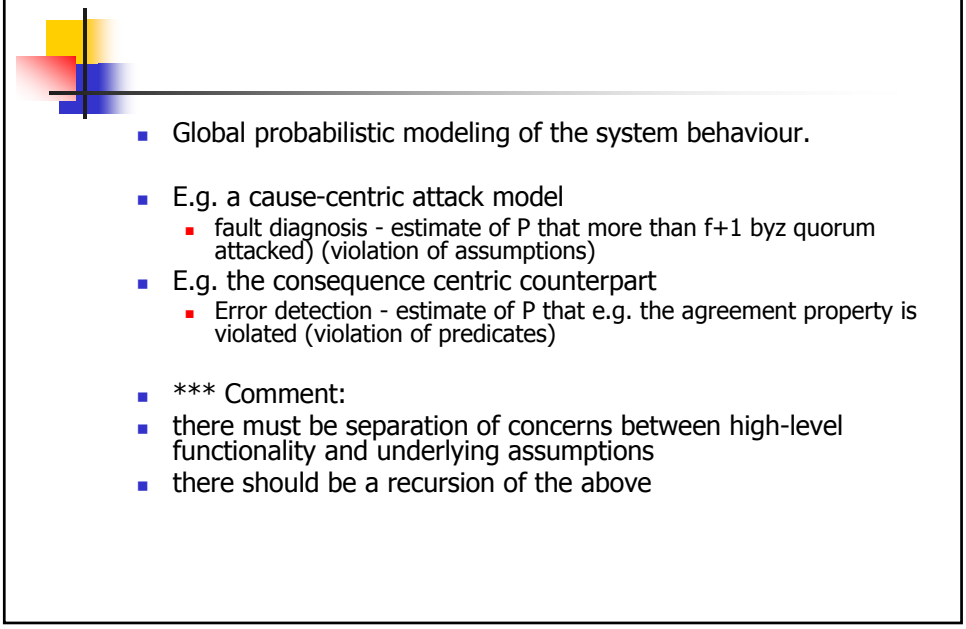
- Adversary Work Factor - maximize the AWF to impart a system (cost of breaking in)
- Critical Security Rating (CSR) - variables to give a numerical score to systems resilience to attack

- *** merits of the objective:
- may allow to establish a contract between the user and the provider of the system (Netscape SSL cost-of-breaking statement)

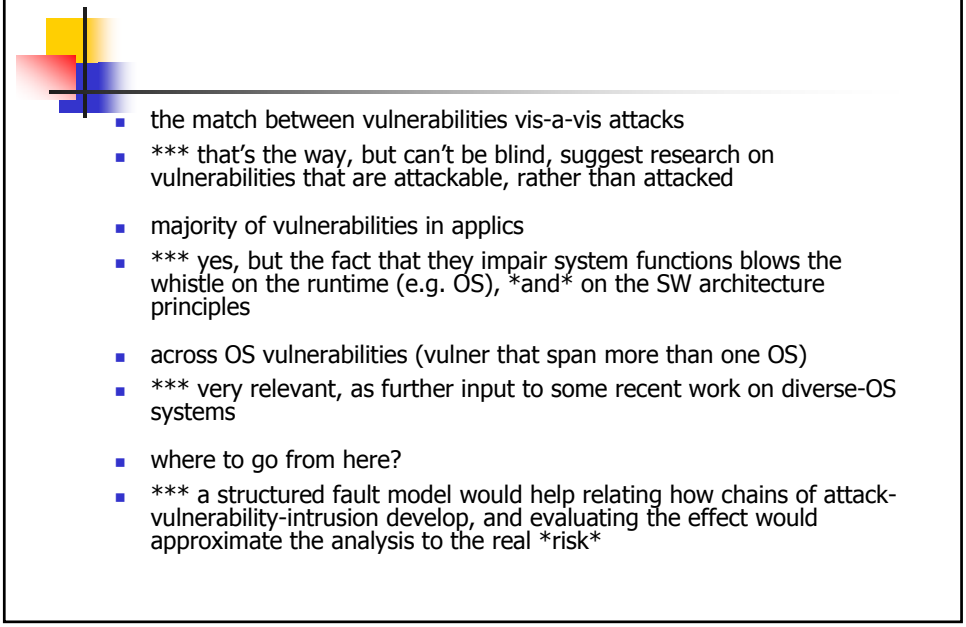
- *** problems with the method:
- intrusiveness of the intruder's operation analysis
- faithfulness of red-teaming: completeness (you get existence proofs, you don't get non-existence proofs); truthfulness (the red team may not want to uncover some problems)
- cause-centric

- Even with diversity, independence is not believable. What you can hope is to quantify how reliable a diverse system will be.

- *** on diversity in ID sensors, there is a difference between:
- anomaly and misuse detection sensors
  - (the detection target is complementary)
- sensors that measure different things
  - (only improve fault coverage in extension, not coverage of fault diagnosis/error detection itself)
- diverse implementations of same class sensors
  - (improve coverage of fault diagnosis/error detection itself)



- Global probabilistic modeling of the system behaviour.
- E.g. a cause-centric attack model
  - fault diagnosis - estimate of P that more than f+1 byz quorum attacked) (violation of assumptions)
- E.g. the consequence centric counterpart
  - Error detection - estimate of P that e.g. the agreement property is violated (violation of predicates)
- *** Comment:
- there must be separation of concerns between high-level functionality and underlying assumptions
- there should be a recursion of the above



- the match between vulnerabilities vis-a-vis attacks
- *** that's the way, but can't be blind, suggest research on vulnerabilities that are attackable, rather than attacked
- majority of vulnerabilities in applics
- *** yes, but the fact that they impair system functions blows the whistle on the runtime (e.g. OS), *and* on the SW architecture principles
- across OS vulnerabilities (vulner that span more than one OS)
- *** very relevant, as further input to some recent work on diverse-OS systems
- where to go from here?
- *** a structured fault model would help relating how chains of attack-vulnerability-intrusion develop, and evaluating the effect would approximate the analysis to the real *risk*

- fault prevention in OS
- *** cool, always a pleasure to see "good" OSs, but how effective?
  - this must be matched by an architectural approach to designing with trusted subsystems, otherwise it does not work, because appls will screw up anyway

Conclusions

- missing link is often between precise and accurate specification and proof of properties, and (necessarily) imprecise and inaccurate specification and proof or estimation of the underlying environment assumptions
- *** guidelines:
- precision/accuracy of spec/proof of props
  - handle expressiveness and complexity (new properties; composition theorems, etc.)
- underlying environment
  - structured fault models, consequence-oriented (e.g. attack-vulner-intrusion lattices); adequate quantitative metrics
- link between high-level assertions and low-level runtime
  - separation of concerns;
  - trustworthiness(dependability)-aware assertions;
  - continuous (rather than discrete) notions of trust

How might it be done?

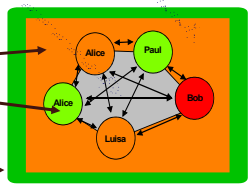
- There will be several ways, certainly
- Our own thoughts in two slides

- Paper ref. can be found at
- www.navigators.di.fc.ul.pt/it

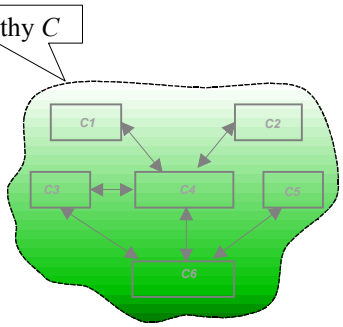
On coverage and separation of concerns

- predicate P holds with a coverage Pr
  - we say that we are confident that P has a probability Pr of holding
- environmental assumption coverage (Pr_e)
  - set of assumptions (H) about the environment where system will run
  - $Pr_e = Pr(H \mid f)$ *f- any fault*
- operational assumption coverage (Pr_o)
  - the assumptions about how the system/algorithm/mechanism proper (A) will run, under a given set of environmental assumptions
  - $Pr_o = Pr(A \mid H)$

$Pr(A) = Pro \times Pre = Pr(A \mid H) \times Pr(H \mid f)$

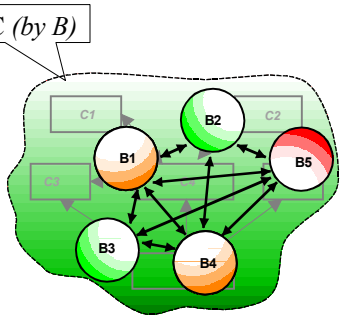


A robust design approach



- Architectural hybridization:
  - failure assumptions enforced by architecture and construction, thus substantiated
  - combined/recursive use of attack/vulnerability/intrusion prevention/removal/tolerance
- Trusted (trustworthy) components:
  - components or subsystems with justified coverage, used in the construction of fault-tolerant protocols under architectural hybrid failure assumptions

A robust design approach



- Architectural hybridization:
  - failure assumptions enforced by architecture and construction, thus substantiated
  - combined/recursive use of attack/vulnerability/intrusion prevention/removal/tolerance
- Trusted (trustworthy) components:
  - components or subsystems with justified coverage, used in the construction of fault-tolerant protocols under architectural hybrid failure assumptions