Panel: "The Multicians"

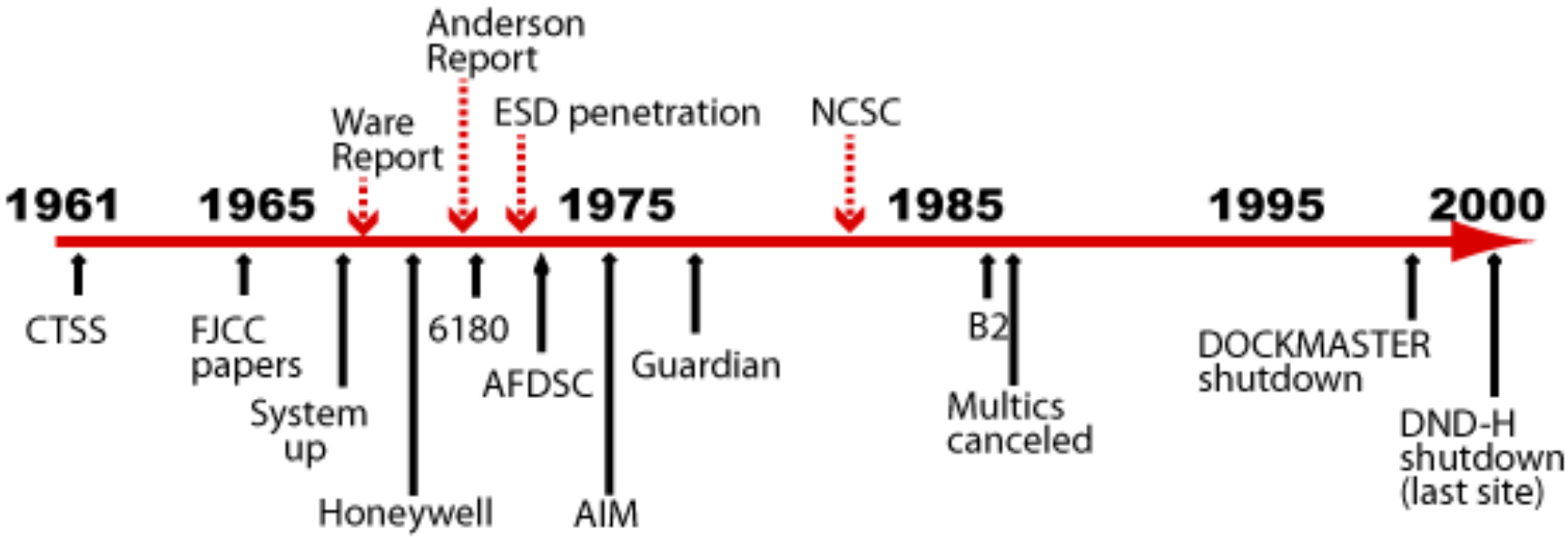
Moderator: Olin Sibert

Before Multics

Professor Roger R. Schell
University of Southern California

ACSAC 2014
New Orleans, Louisiana
December 10, 2014

Multics Security Activity Timeline



3 Levels of Security Consciousness

#1 There is no Problem

Deny the Problem

- Common security consciousness before Multics
  - Only air-gap had basis for trust
  - Many people unaware of the threat
- As an ACM presentation put it:

  "***Security is inherently different from other aspects of computing due to the presence of an adversary. As a result, identifying and addressing security vulnerabilities requires a different mindset from traditional engineering. Proper security engineering—or the lack of it!—affects everything . . . .***"
- Subversion is likely witted adversary attack of choice
  - Demonstrated in Karger's Multics security analysis

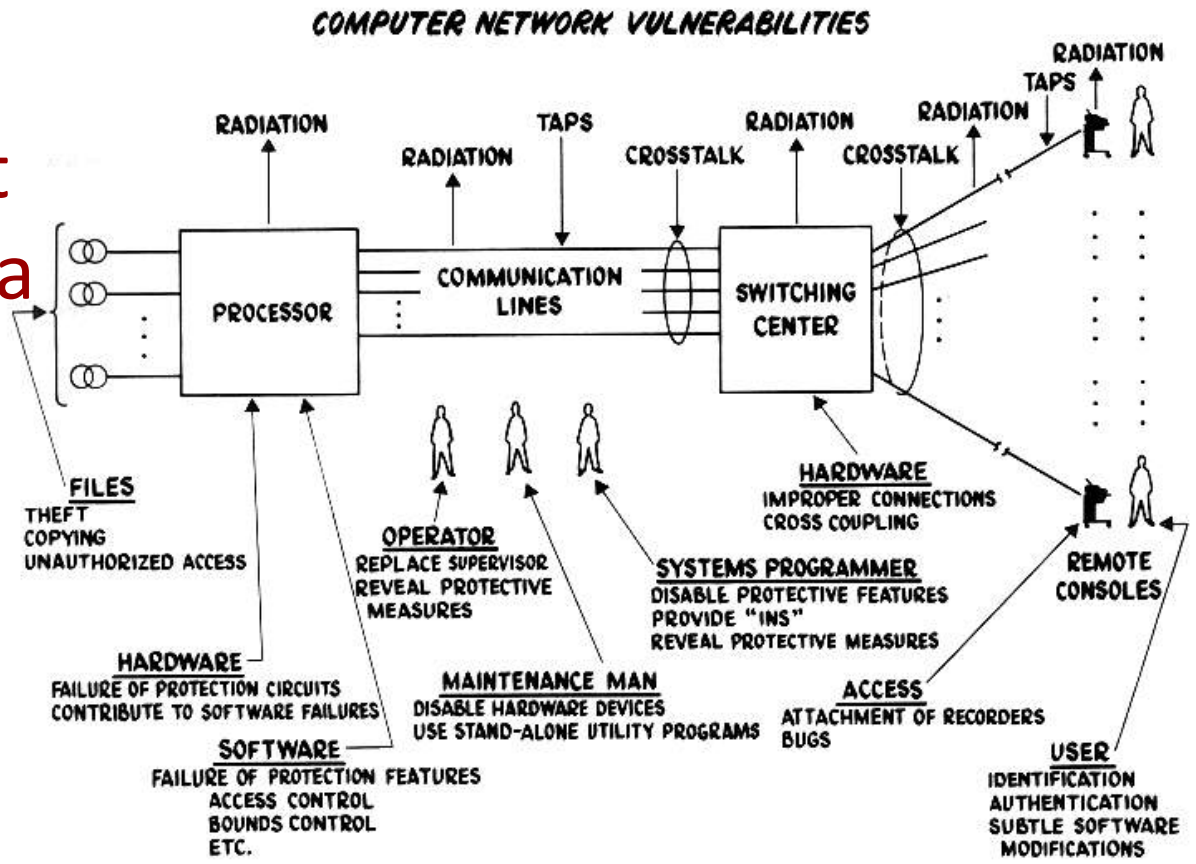
3 Levels of Security Consciousness

#1 There is no Problem
Ignore Threat (especial subversion)

#2 There is no Solution

Security Can Seem Overwhelming

- Willis Ware 1969 Report
- Recognized a witted adversary



3 Levels of Security Consciousness

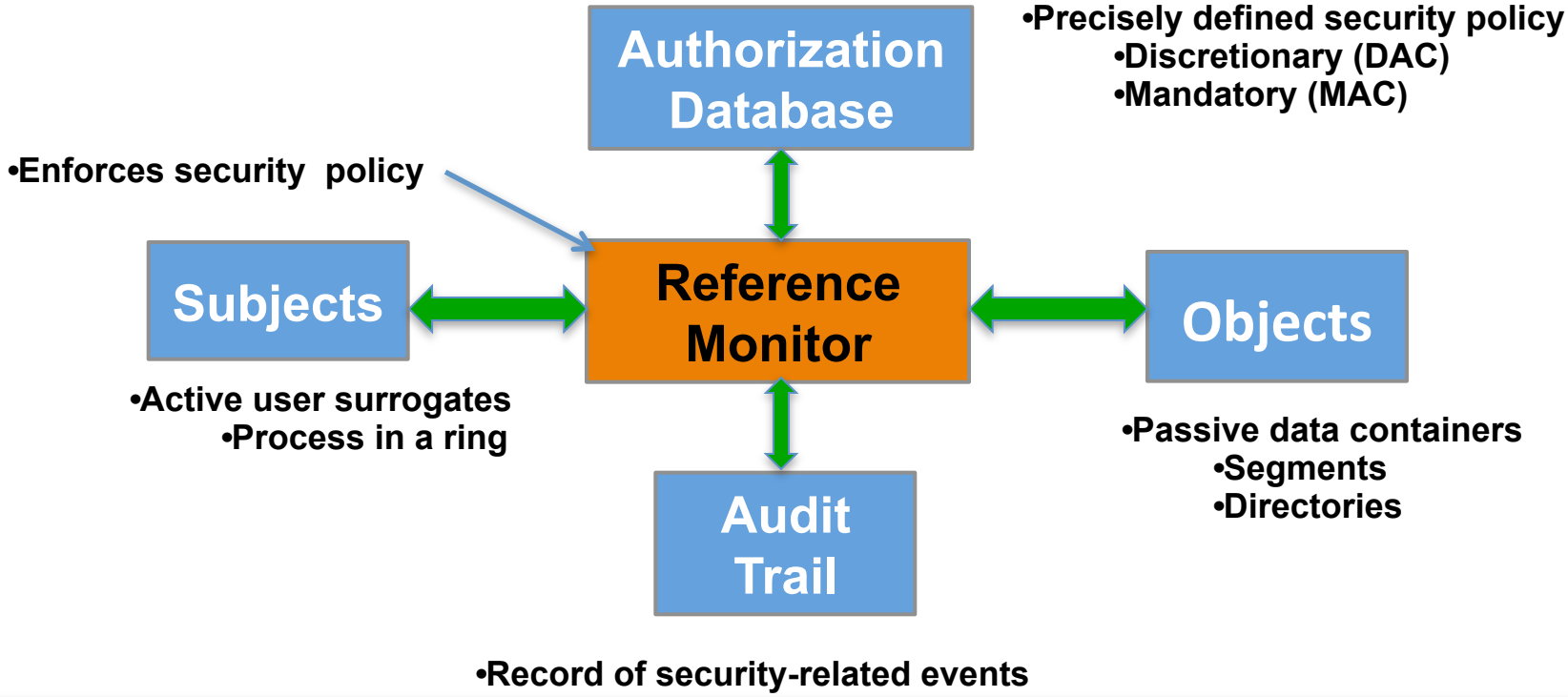
#1 There is no Problem
Ignore Threat (especial subversion)

#2 There is no Solution
Devastating impact of vulnerabilities

#3 There is no Free Lunch

Reference Monitor Abstraction

Anderson Report Directly stimulated by Multics



Summary of 3 Levels of Consciousness

#1 There is no Problem

Ignore Threat (especial subversion)

#2 There is no Solution

Devastating impact of vulnerabilities

#3 There is no Free Lunch

Systematic engineering to leverage Multics

Security Problems Illuminated by Multics

- Need for precisely defined and understood policy
  - MAC (lattice); DAC (matrix/ACL) ; Application policy
- Witted adversary malicious subversion
  - Trojan horse flow control; Class A1 to mitigate trap doors
- Security by obscurity – defense in depth
  - Abstract interface supporting general computer utility
- S/W quality Optimism – non-rigorous arguments
  - Logical internal design, e.g., 2-level scheduler, eventcounts
- Assume lazy attackers – "no one would ever do that"
  - "Complete", deterministic and repeatable behavior

So-called "Solutions" Exposed by Multics

- Lack critical hardware for security and performance
  - Segmentation is crucial enabler, rings, manage processes
- Penetration and patch, without life-cycle protection
  - Paradigm shift: no Class A1 security patches in years of use
- Non-rigorous mappings for user surrogates
  - Reference monitor "subjects" – process-domain (ring) pair
- Imprecise information container notions, e.g., "files"
  - RM "objects" – directly sharable, CPU addressable segments
- Security "features" in Monolithic operating systems
  - Evaluable, precisely defined, composable TCB "subsets"

Security Engineering Gaps

- Rigorous logical argument policy is enforced
  - Reference monitor, and implementation ("security kernel")
- How to prove the negative – never an insecure state
  - Bell and LaPadula model "lichpin", Multics interpretation
- Making highly secure system with MAC usable
  - 20 years experience – Pentagon, GM, Ford, NCSC
- Architectural longevity, e.g., user devices, embedded
  - SCOMP SPM retrofit; GEMSOS "mini-Multics" on Intel x.86
- Systematic software engineering to support security
  - HOL for OS, modularity, layering, abstraction, minimization

Summary of Security World Multics Faced

- #1 There is no Problem

  Witted adversary subversion is "inherently different"

- #2 There is no Solution

  "Best practice" and surveillance (back doors) can't solve

- #3 There is no Free Lunch

  "Mere mortals" can engineer high assurance systems

  BLACKER, Oracle MLS DBMS, Pentagon MLS access, UK guard

Panel: "The Multicians"

Moderator: Olin Sibert

Before Multics

Professor Roger R. Schell
University of Southern California

ACSAC 2014
New Orleans, Louisiana
December 10, 2014