

Real-Time Embedded Avionics System Security and COTS Operating Systems

Open Group Real-Time Forum

Robert Allen – Boeing Phantom Works

July 18, 2001

Real-Time Embedded Avionics System Security and COTS Operating Systems

- **Problem:**

- Security Has Been a “Show Stopper” For Using COTS Real-time Operating Systems (RTOSes)

- **Reasons:**

- Most Older Embedded Systems Were Stand-alone and Not Interconnected
- Until Recently, Most COTS RTOSes Were Single Address Space
- Security Is Often Project Specific, Giving Rise To Point Solutions; It Was Unclear How To Generalize For Customer Base
- Perceived As a Niche Market By Vendors

Real-Time Embedded Avionics System Security and COTS Operating Systems

- **Conclusion By RTOS Vendors:**
 - Rate of Return Did Not Justify Required Levels of Investment
- **Exceptions:**
 - Raytheon's RT-Secure
 - Green Hill's Integrity

Real-Time Embedded Avionics System Security and COTS Operating Systems

- **What Do I Mean By Security**

- For Years “Security” Levels Were Defined by the “Orange Book”
- The Orange Book Is the U.S. Department of Defense *Trusted Computer System Evaluation Criteria*
- Common Criteria “Supercedes” the Orange Book, However Protection Profiles Exist That Are Inherited From the Orange Book Definitions

- **Reference URLs**

- www.radium.ncsc.mil/tpep/library/rainbow/5200.28-STD.html
- www.radium.ncsc.mil/tpep/library/rainbow/
- www.dynamoo.com/orange/summary.htm

Real-Time Embedded Avionics System Security and COTS Operating Systems

- **Common Criteria**

- Defines The Process for the Evaluation of a System's Security Properties
- Does NOT Directly Specify System Security Requirements. Protection Profiles Contain System Security Requirements and Are Developed Independently for Each System. New Profiles Can Be Derived From Existing Profiles

- **Reference URLs**

- <http://www.commoncriteria.org/>
- <http://csrc.nist.gov/cc/>
- <http://www.radium.ncsc.mil/tpep/library/ccitse/>

Real-Time Embedded Avionics System Security and COTS Operating Systems

- **Orange Book Security Levels:**
 - **D – Minimal Protection**
 - Any System That Does Not Comply With Any Other Category, or Has Failed to Receive a Higher Classification; D-level Certification Is Rare
 - **C – Discretionary Protection**
 - Discretionary Protection Applies to Trusted Computer Bases (TCBs) With Optional Object (i.e., File, Directory, Devices, etc.) Protection

Real-Time Embedded Avionics System Security and COTS Operating Systems

- **Orange Book Security Levels:**
 - **C1 – Discretionary Security Protection**
 - Discretionary Access Control, for Example Access Control Lists (ACIs), User/Group/World Protection
 - Usually for Users Who Are All on the Same Security Level
 - Username and Password Protection and Secure Authorizations Database (ADB)
 - Protected Operating System and System Operations Mode
 - C1 Certification Is Rare. Example Systems Are Earlier Versions of Unix, IBM RACF

Real-Time Embedded Avionics System Security and COTS Operating Systems

- **Orange Book Security Levels:**
 - **C2 – Controlled Access Protection**
 - *C1, Plus*
 - Object Protection Can Be on a Single-user Basis, e.g. Through an ACL or Trustee Database
 - Authorization for Access May Only Be Assigned by Authorized Users
 - Object Reuse Protection (i.e. To Avoid Reallocation of Secure Deleted Objects)
 - Mandatory Identification and Authorization Procedures for Users, e.g. Username/Password
 - One of the Most Common Certifications; Example Operating Systems Are: VMS, IBM OS/400, Windows NT, Novell Netware 4.11, Oracle 7, DG AOS/VS II

Real-Time Embedded Avionics System Security and COTS Operating Systems

- **Orange Book Security Levels:**
 - **B – Mandatory Protection**
 - Division B Specifies That the TCB Protection Systems Should Be Mandatory, Not Discretionary
 - **B1 - Labeled Security Protection**
 - *C2, Plus*
 - Mandatory Security and Access Labeling of All Objects, e.g. Files, Processes, Devices, etc.
 - Label Integrity Checking (e.g. Maintenance of Sensitivity Labels When Data Is Exported)
 - Auditing of Labeled Objects
 - Mandatory Access Control for All Operations
 - Example Operating Systems Are: HP-UX BLS, Cray Research Trusted Unicos 8.0, Digital SEVMS, Harris CS/SX, SGI Trusted IRIX

Real-Time Embedded Avionics System Security and COTS Operating Systems

- **Orange Book Security Levels:**
 - **B2 - Structured Protection**
 - *B1, Plus*
 - Hierarchical Device Labels
 - Mandatory Access Over All Objects and Devices
 - Trusted Path Communications Between User and System
 - Tracking Down of Covert Storage Channels
 - Formal Models of TCB
 - Example Systems Are: Honeywell Multics, Cryptek VSLAN, Trusted XENIX

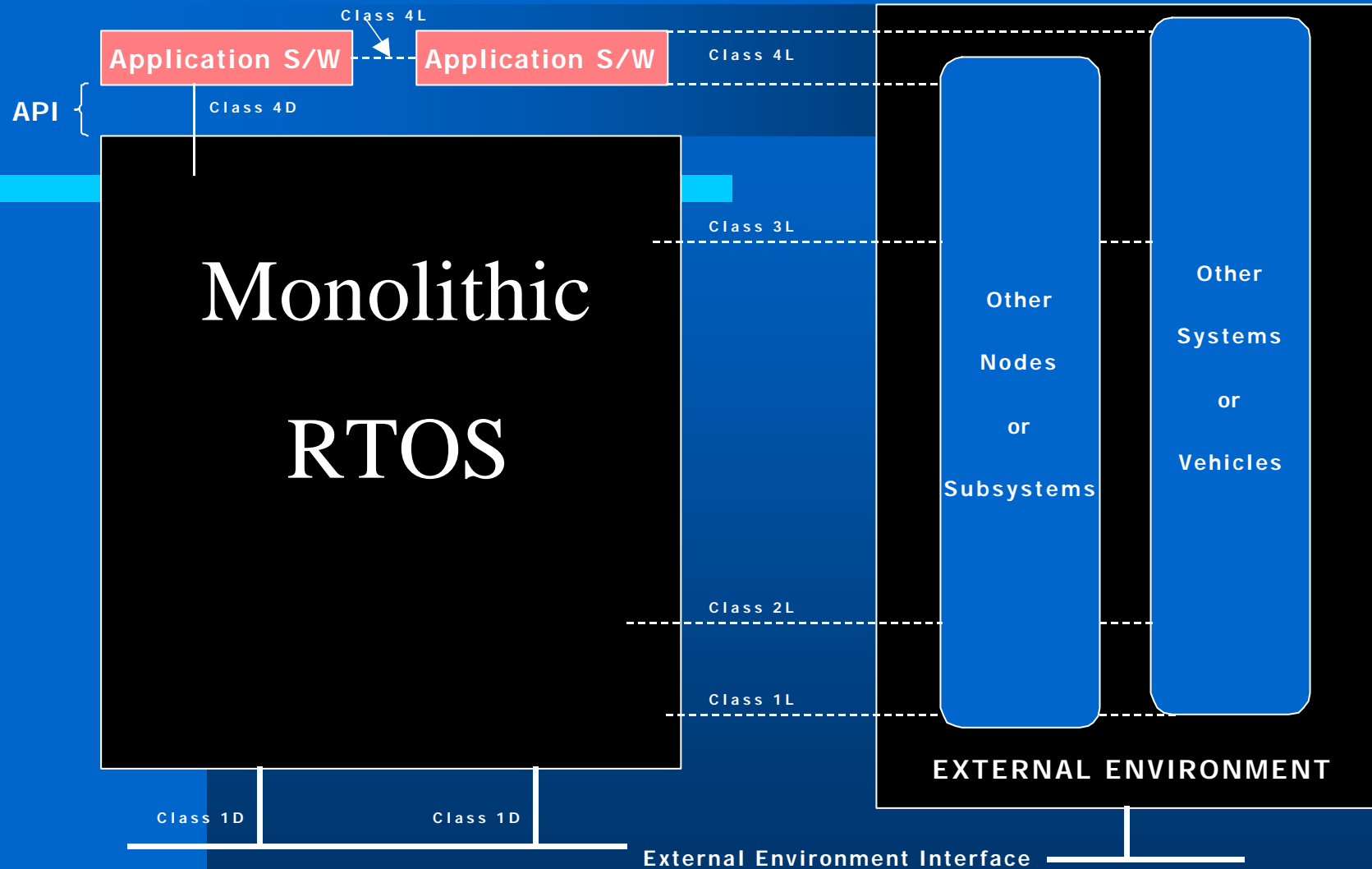
Real-Time Embedded Avionics System Security and COTS Operating Systems

- **Orange Book Security Levels:**
 - **B3 - Security Domains**
 - *B2, Plus*
 - ACLs Additionally Based on Groups and Identifiers
 - TCB Models More Formal
 - Trusted Recovery After System Down
 - Zero Design Flaws in TCB, and Minimum Implementation Flaws
 - The Only B3-certified OS Is Getronics/Wang Federal XTS-300

Real-Time Embedded Avionics System Security and COTS Operating Systems

- **Orange Book Security Levels:**
 - **A - Verified Protection**
 - Division A Is the Highest Security Division
 - **A1 - Verified Protection**
 - *B3, Plus*
 - Formal Methods and Proof of Integrity of TCB
 - Few A1-certified Systems: Boeing MLS LAN, Gemini Trusted Network Processor, Honeywell SCOMP
 - **A2 and Above**
 - Provision Is Made for Security Levels Higher Than A2, Although These Have Not Yet Been Formally Defined. No OS Rated Above A1

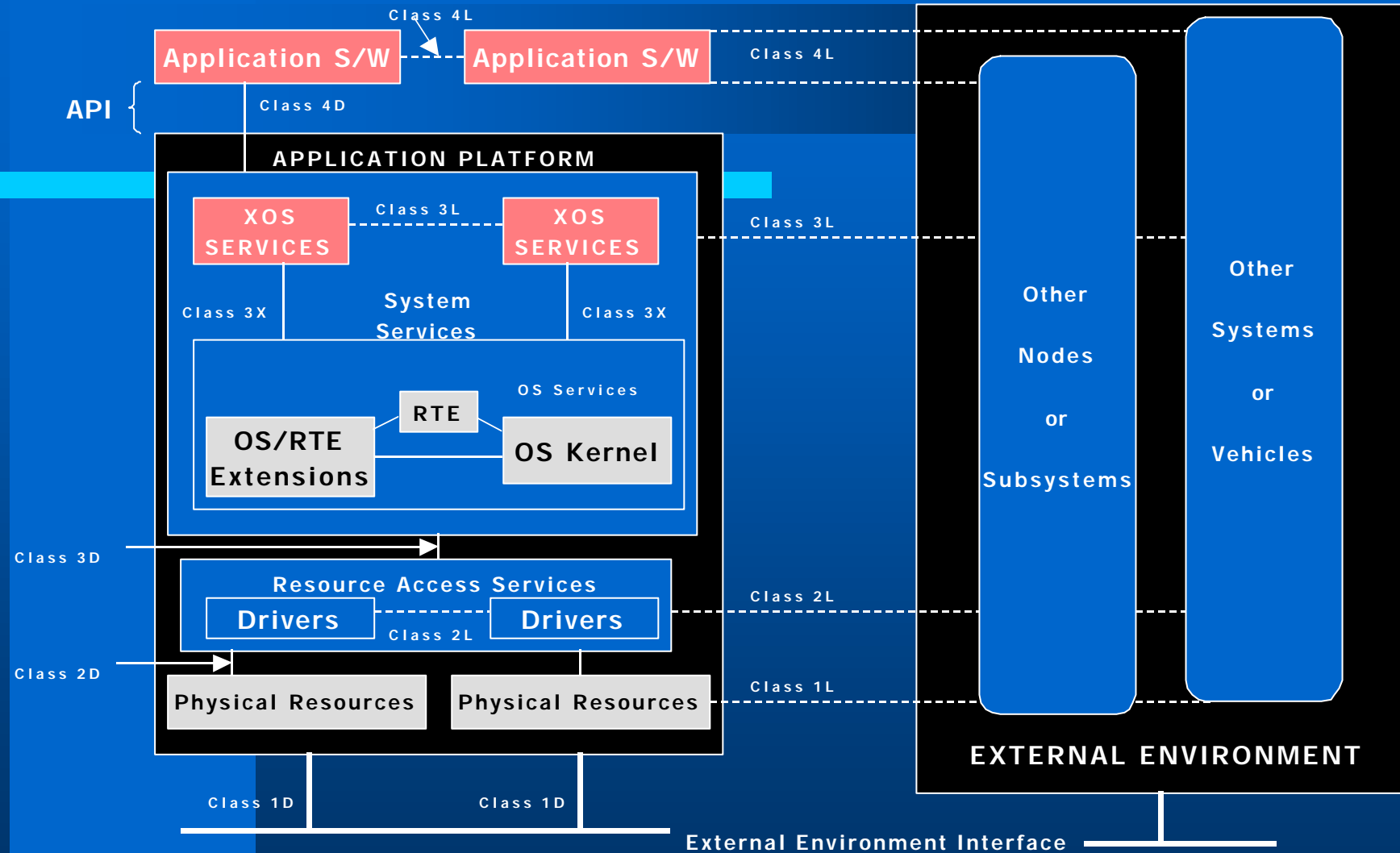
How RTOSes Have Changed (Before ...)



KEY:

API = Applications Platform Interface EEI = External Environment Interface L = Logical D = Direct
 RTOS = Real-Time Operating System

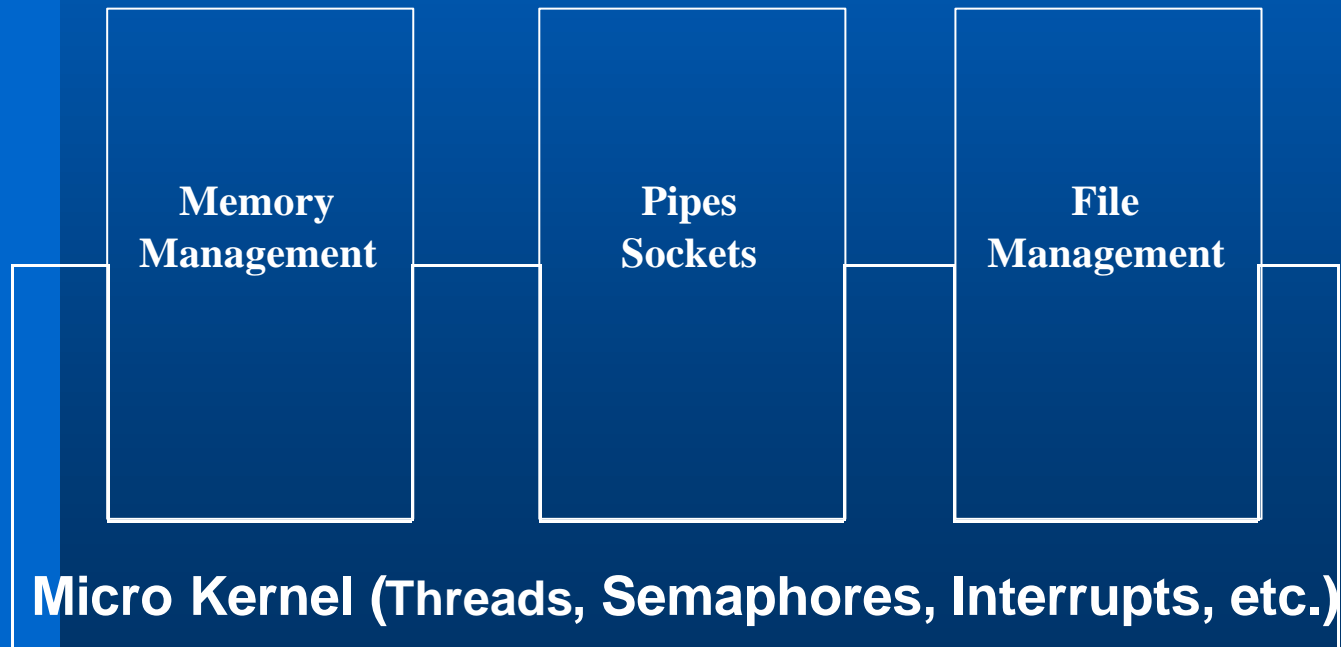
How RTOSes Have Changed (After ...)



KEY:

API = Applications Platform Interface EEI = External Environment Interface L = Logical D = Direct X = eXpress
 OS = Operating System RTE = Run Time Environment

Modular Scalable Architecture



Real-Time Embedded Avionics System Security and COTS Operating Systems

- **What Is Needed:**

- Identification of Features/Properties Needed in the Micro Kernel
- Establish Kernel Capabilities/Mechanisms to Be Made Available to the “User” in a Programming Interface
- Strive for Minimalist Approach Requiring As Little Change As Possible, on the Part of Vendors, to Encourage Participation by the RTOS Community
- Proof-Of-Concept Demonstration
 - Certification Labyrinth for Common Criteria Less Than Clear for Many Potential Users
 - Reference Implementation
 - Adoption of “Standard” Made More Likely Once Approach Has Been Shown Viable