Proving Security and Reliability Attributes for Distributed Real-Time Systems using Meta Object Models

John C. Shovic, PhD
A. David McKinnon
David E. Bakken, PhD
SRES Laboratory
Washington State University
What is SRES?

- Secure & Reliable Embedded Systems Laboratory
- Pronounced “Cirrus”
- Focused on Embedded Systems
- Designed to prove attributes about security and reliability
What are we doing?

- Designing a methodology to prove mathematically specified attributes of embedded systems such as communication protocols are **secure** and/or reliable.
How are we doing this?

• Building a system that can be tracked to a mathematical model.

• Building high level Theorems in the system (in software) that can be actually used to build real software.
How does it work?

Object-Oriented Systems and Programming Languages

Denotational Semantics

MOM

Formal Specification

ROC

ROC Mechanization into HOL

Operational Framework

Verification Framework

MOM Thms

ROC Thms

...
A Real Time Process Calculus

• ROC - Robust Object Calculus
  – Describes Processes
  – Can be used to model software and protocols
  – ROC has a notion of sequence
  – Foundation of T-ROC - Real Time Object Calculus
T-ROC

• T-ROC requires a sense of Logical and Absolute Time for Real Time Processes

• T-ROC requires a detailed and a fuzzy sense of time
  – >, <, = Mathematical Equivalence
  – %>, %<, %= Fuzzy Comparisons
• MOM - Meta Object Module
  – Core Distributed Object Model
  – Articulates and captures a variety of objects and object interactions
  – Maps to many programming languages
  – Supports many security methods
  – Fits well with CORBA model
MOM Overview

- Encapsulates methods or objects to protect them from malicious usage attempts
- Each MOM object have the following components (Fig.1)
- Connected using a tree structure
- All communication between the MOM objects are done using messages traversing the tree structure
Communication Path Example in MOM/ROC

• MOM/ROC can be used on software as well as protocols
• CORBA derived protocols can be simulated / analyzed by this method
• TCP is an example
Transmission Control Protocol Overview

- TCP is a standard connection protocol
TCP in MOM

ROOT

OACL

TCP Listener

TCP Method Interface

Method Arbiter

TCP Method Body

CLIENT

TCP Message Handler

Request

TCP Sender

TCP Method Body

TCP Timer

TCP Timer Interrupt Handler

TCP Message Handler

Reply

TCP Listener

OACL

Authorization Check

TCP Message Handler

Request

TCP Method Interface

Method Arbiter

TCP Method Body

SERVER

TCP Method Body

TCP Timer

TCP Timer Interrupt Handler

TCP Message Handler

Return Value

TCP Method Interface

Method Arbiter

TCP in MOM

Computer And Software Engineering

Secure & Reliable Embedded Systems
TCP in MOM

- OACL (Object Access Control List) restricts requests, ack, etc.
- Restrictions can be placed throughout the system
- System can be reduced from MOM to ROC expressions and Theorems proved
ROC Syntax

- $A ::= a \& a$  
  Concurrent Composition
- $n ::= a$  
  Recursion
- $a + a$  
  Non-deterministic Choice
- $a \mid a$  
  Left Preferential Choice
- $x \rightarrow a$  
  Input
- $v \uparrow a$  
  Output
- $a @ v$  
  Application
- $n \backslash a$  
  New Name $n$ in $a$
- $n$  
  Name
- $\text{nil}$  
  Empty Agent
ROC Example
Expressions for TCP

Connection Establishment

• **Client side**
  - Client ::= SYN ^ cSyn_sent
  - cSyn_sent ::= ACK_SYN -> cSyn_recv
  - cSyn_recv ::= ACK ^ cEst

• **Server side**
  - Server ::= Svr_PL (Passive Listener)
  - SvrPL ::= SYN -> SvrPL1 & Server
  - SvrPL1 ::= ACK_SYN ^ sSyn_recv & Server
  - sSyn_recv ::= ACK -> sEst & Server
TCP MOM Message in ROC

- TCPMsg = [source#, destination#, Seq#, Key_List#, M_Body#] # nil
- source ::= [ClientIP#, ClientPort#]
- dest ::= [ServerIP#, ServerPort#]
- Seq# := a 32 bit number
- Key_List ::= [token#, Key_List] | null
- M_Body ::= [Request, Rq_Body#] | [Reply, Rp_Body#] | [Ack, Ack_body#]
Issues using MOM / T-ROC with CORBA

• Tools still being developed
• General Software needs to be built using MOM to keep size of analysis reasonable in theorem prover
• CORBA supports complex protocols that will need to be put into MOM
Conclusion

• T-ROC forms the foundation of a real time theorem proving system
• Actual protocols can be designed in MOM and then ROC extracted and attributes proven
• MOM system can be used to prove attributes about secure and reliable systems for CORBA and other protocols