



JPL

Reducing Software Security Risk (RSSR)

David Gilliam, John Powell
California Institute of Technology,
Jet Propulsion Laboratory

Matt Bishop
University of California at Davis

California Institute of Technology, Jet Propulsion Lab



Software Security Checklist (SSC)

■ NOTE:

- This research was carried out at the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration
- The work was sponsored by the NASA Office of Safety and Mission Assurance under the Software Assurance Research Program lead by the NASA Software IV&V Facility
- This activity is managed locally at JPL through the Assurance and Technology Program Office



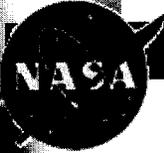
Agenda



- Collaborators
- Goal
- Problem
- Software Security Assessment Instrument (SSAI)
- Model Checking: Flexible Modeling Framework
- Software Security Checklist (SSC)

Current Collaborators

- David Gilliam – Principle Investigator, JPL
- John Powell
- Tom Wolfe
- Matt Bishop – Associate Professor of Computer Science, University of California at Davis
- <http://rssr.jpl.nasa.gov>



Agenda

- Collaborators
- ■ Goal
- Problem
- Software Security Assessment Instrument (SSAI)
- Model Checking: Flexible Modeling Framework
- Software Security Checklist (SSC)



Goal

- Reduce security risk to the computing environment by mitigating vulnerabilities in the software development and maintenance life cycles
- Provide an instrument and tools to help avoid vulnerabilities and exposures in software
- To aid in complying with security requirements and best practices



Agenda

- Collaborators
- Goal
- ■ Problem
- Software Security Assessment Instrument (SSAI)
- Model Checking: Flexible Modeling Framework
- Software Security Checklist (SSC)



Problem

- Lack of Experts: Brooks – “No Silver Bullet” is still valid (IEEE Software Engineering, 1987)
- Poor Security Requirements
- Poor System Engineering
 - Leads to poor design, coding, and testing
- Cycle of Penetrate and Patch
- Piecemeal Approach to Security Assurance



Agenda

- Collaborators
- Goal
- Problem
- ■ Software Security Assessment Instrument (SSAI)
- Model Checking: Flexible Modeling Framework
- Software Security Checklist (SSC)



Software Security Assessment Instrument (SSAI)

- Software Security Checklist (SSC)
 - Software Life Cycle
 - External Release of Software
- Vulnerability Matrix (VMatrix)
 - List and Ranking of Vulnerabilities
 - Vulnerability Properties
 - Classification of Types of Vulnerabilities
 - List Maintained by UC Davis



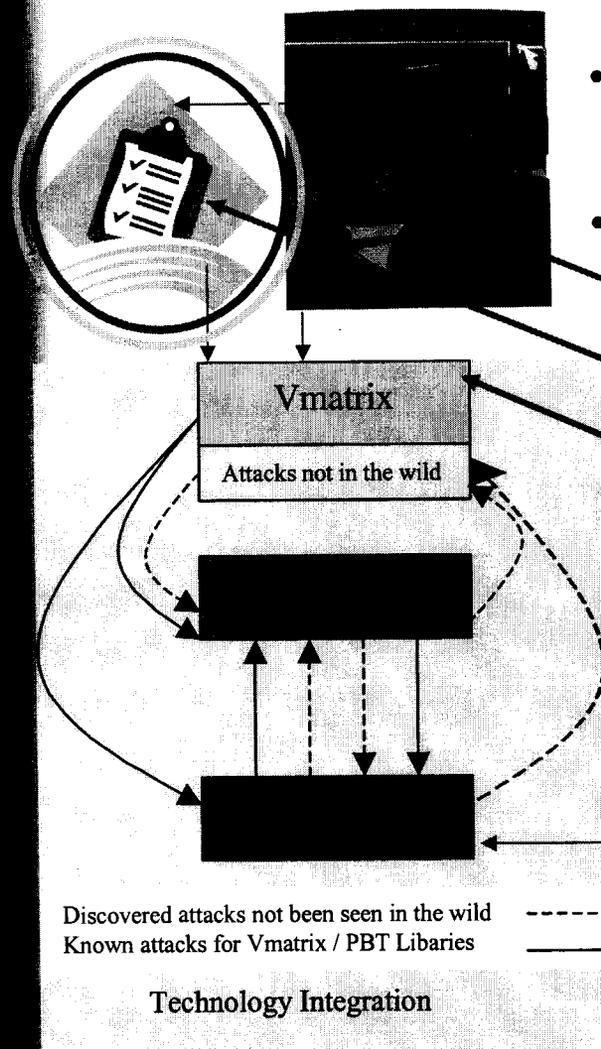
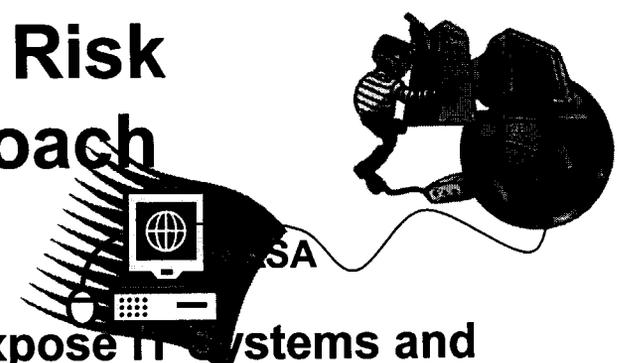
SSAI (Cont.)

- **Model-Based Verification (MBV) and a Flexible Modeling Framework (FMF)**
 - SPIN Model Checker and Promela
 - FMF Developed to Address State Space
- **Property-Based Tester (PBT)**
 - Tests Source Code for JAVA, C, and C++
 - Verifier to ensure security property violations have not been re-introduced in coding

SSAI (Cont.)

- Security Assessment Tool's (SAT's)
 - List of Tools and Purpose of Each
 - Alternate Tools and Sites to Obtain Them

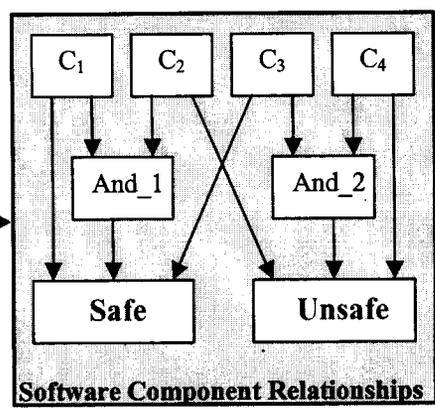
Reducing Software Security Risk Through an Integrated Approach



- Software Vulnerabilities Expose IT Systems and Infrastructure to Security Risks
- Goal: Reduce Security Risk in Software and Protect IT Systems, Data, and Infrastructure
 - Security Training for System Engineers and Developers
 - Software Security Checklist for end-to-end life cycle
 - Software Security Assessment Instrument (SSAI)

• Security Instrument Includes:

- Security Checklist
- Vulnerability Matrix
- Property-Based Testing
- Model-Based Verification
- Collection of security tools



Agenda

- Collaborators
- Goal
- Problem
- Software Security Assessment Instrument (SSAI)
- ■ Model Checking: Flexible Modeling Framework
- Software Security Checklist (SSC)

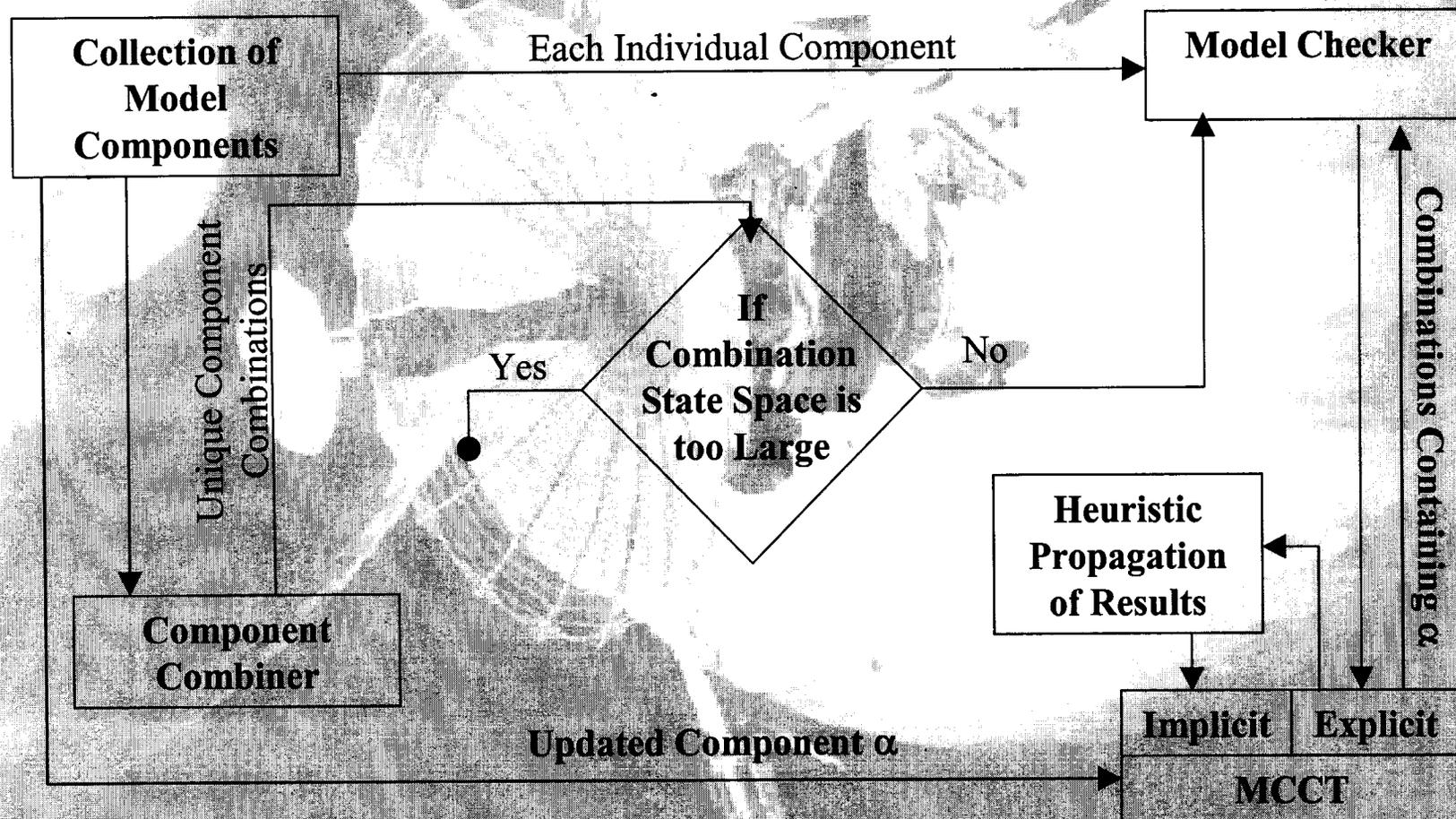


Model Checking: Flexible Modeling Framework (cont.)

- MC with FMF Benefits Software Early in its Lifecycle
 - Earlier Discovery of Software Errors
 - Correction is easier / better / less expensive
- FMF must adapt to early lifecycle events
 - Rapidly changing requirements and designs
 - Varying / Increasing levels of detail defined for different parts of the system



Model Checking: Flexible Modeling Framework



Agenda

- Collaborators
- Goal
- Problem
- Security & the Software Life Cycle
- Software Security Assessment Instrument (SSAI)
- ■ Software Security Checklist (SSC)
- Final Notes



Software Security Checklist (SSC)

■ Two Phases

□ Phase 1:

- Provide instrument to integrate security as a formal approach to the software life cycle
- Requirements Driven

□ Phase 2:

- External Release of Software
- Release Process

SSC (Cont.)

- Phase 1:
 - Pre-Requirements
 - Understand the Problem and Scope
 - Requirements Gathering and Elicitation
 - Be Aware of Applicable Requirements Documents
 - Provide Trace to External Requirements Docs
 - Security Risk Assessment
 - NPG 7120.5B – Project Life Cycle document
 - Potential Integration with DDP Tool
 - V&V Tools Available for Software Life Cycle

SSC (Cont.)

■ Phase 2:

□ Release of Software

■ Areas for Protection:

- Protect People
- Protect ITAR and EAR
- Protect Trade Secrets – Patents
- Protect Organizational Resources

■ Considerations

- Insecure Subsystem Calls
- Embedded IP Addresses or Phone Numbers

■ Delivered to Code R Draft Checklist

SSC (Cont.)

- Project Life Cycle Approach
 - Security Requirements
 - Stakeholders
 - Federal, State, Local Requirements
 - NASA Requirements and Guidelines
 - Design, Development, Test
 - Maintenance and Decommissioning
 - Tools and Instruments
 - Expert Center (IV&V) and People to Assist
 - Training

SSC Tools

- Review Source Code
- Review File Calls
- Review Library Calls
- Check Subroutine Calls in Binaries
 - Provided Perl Scripts
 - System and Programming Tools

Agenda

- Collaborators
- Goal
- Problem
- Security & the Software Life Cycle
- Software Security Assessment Instrument (SSAI)
- Software Security Checklist (SSC)
- ■ Final Notes

Final Notes

- Womb-to-Tomb Process
 - Must Coincide with Organizational Policies and Requirements
 - Notification to Users and Functional Areas when Software or Systems De-Commissioned
 - Regression Test on Decommissioning
 - Re-Verify Security on Decommissioning

Final Notes (Cont.)

- Return on Investment (ROI)
 - Enhanced or Non-Loss of NASA Image
 - Maintenance Costs Decrease

Note on Future Work

- Training Course for SSC and Use of Security Assessment Tools
- Experts and Expert Center Available to Assist with the Instrument and Tools
- Integrate with Deep Space Mission Systems (DSMS)
 - Verifying SSL
 - Potential to Verify Space Link Extension (SLE) Protocol
- Developing an Approach to Project Life Cycle Security Risk Assessment at JPL

FOR MORE INFO...

David Gilliam

JPL

400 Oak Grove Dr., MS 144-210

Pasadena, CA 91109

Phone: (818) 354-0900 FAX: (818) 393-1377

Email: david.p.gilliam@jpl.nasa.gov

John Powell

MS 125-233

Phone: (818) 393-1377

Email: john.d.powell@jpl.nasa.gov

Website: <http://rssr.jpl.nasa.gov/>