Why this workshop?

- We’re increasingly dependent on software
  - automation of everyday (business) processes
  - but also PDAs, phones, TV, PVR, cars, …
- Threats from malware and exploits increase
  - products made extensible and adaptable by software
- Need measures to detect and prevent (potential) security issues
- 20 participants (50/50 industry/academia)

http://swerl.tudelft.nl/leon/cobassa2005/
Topics

- Best Practices for Secure Coding
- Pattern Matching Security Properties
- Hardware-based Control Flow Monitoring
- Use Diskdrive CPU for Malware Detection
- Adversarial Software Analysis
  - “Reverse engineering unfriendly code”
- Identified Top 10 of Open Issues
  - from 60 issues collected by participants

http://swerl.tudelft.nl/leon/cobassa2005/
Identified Top 10 of Open Issues

1. Tools that support software security certification (e.g. CC)
2. Mitigate attacks that make use of physical environment
3. Adopt/adapt existing approaches for source code analysis to the binary/assembly level
4. Maintaining performance when applying security measures
5. Transforming legacy code into safe code
6. Severity based ranking detected security issues
7. Metrics for vulnerability assessment
8. Improve C/C++ compilers so they flag or fix unsafe code
9. Take advantage of “reuse” in malware (“fingerprinting”)
10. Detect & prevent invalid memory read and write operations

http://swerl.tudelft.nl/leon/cobassa2005/
CoBaSSA 2005

First International Workshop on Code Based Software Security Assessments

Proceedings, presentations and results available from http://swerl.tudelft.nl/leon/cobassa2005/

Leon Moonen
Spiros Mancoridis