

# 2012 DHS S&T/ASD(R&E) CYBER SECURITY SBIR WORKSHOP



Homeland  
Security  
Science and Technology



## Multi-platform Program Analysis

GammaTech, Inc.

Paul Anderson

paul@grammatech.com

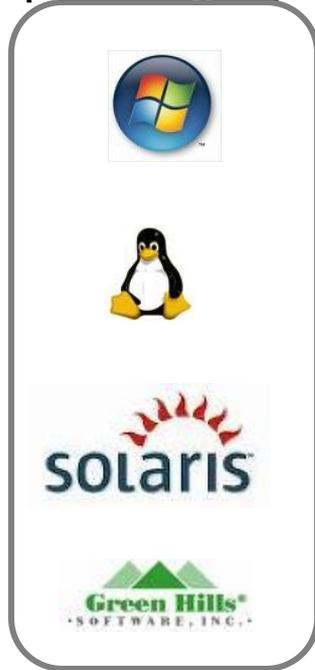
© GammaTech, Inc. 2012

# Key Takeaways

- Platform-specific bugs are especially insidious
- Existing methods for finding such bugs are weak
- Extend existing methods for finding bugs
  - Static analysis for source and binaries
  - Test-case generation
- Use continuous build-and-test technology to harness the cloud
  - Centralize collation and presentation of results
- Leverage existing channels to achieve maximum impact on industry and government

# What is a Platform?

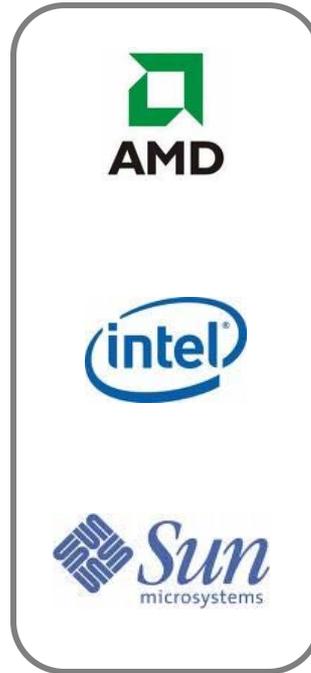
Operating System



X

multiple  
distros

+ CPU



X

32-bit  
64-bit

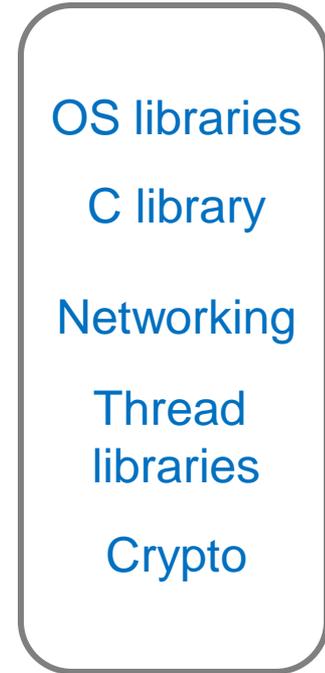
+ Compiler



X

compiler  
flags

+ Libraries



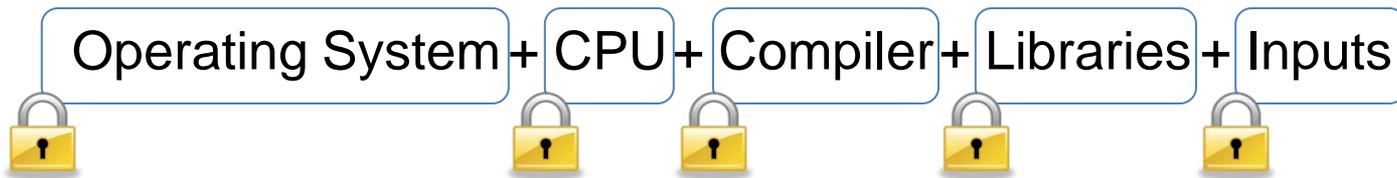
X

multiple  
versions

# Traditional bug-finding techniques

## Testing:

For each test case:

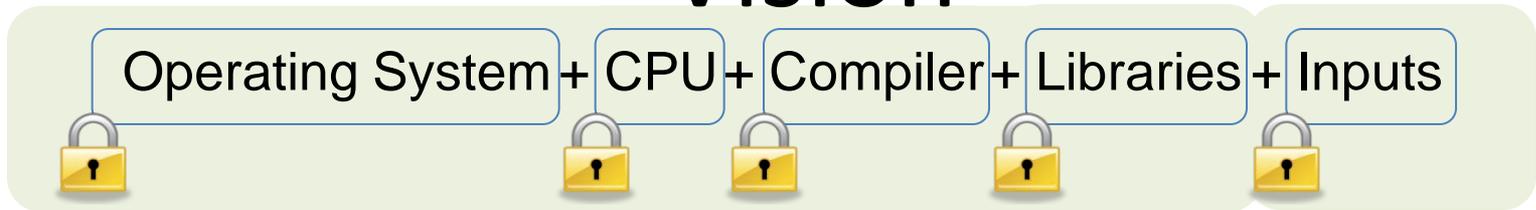


## Static Analysis:

For each scan:



# Vision



- Extend static-analysis technology to account for differences in *Libraries*
- Use binary-analysis technology to address *CPU* and *compiler* differences
- Use Concolic Execution to automatically generate *Inputs*
- Use automated *build-and-test* technology to multiply platform coverage
- Collect and organize results in a central location

# Phase II Approach

- Improve static analysis to refine ability to find platform-specific bugs
- Integrate with other distributed multi-platform build-and-test systems
- Use binary analysis to further refine detection of platform-specific bugs
- Leverage results from other projects on concolic execution late in the project