

# Automated Generation of Attack Signatures and Patches

Tzi-cker Chiueh, Susanta Nanda, Yang Yu

Stony Brook University (<http://www.ecsl.cs.sunysb.edu/dira>)

## Goal

The DIRA project aims to develop a comprehensive set of techniques to generate attack signatures and patches for a wide variety of attacks including control-hijacking attacks such as buffer/integer overflow attacks, web application attacks such as SQL injection and cross-site scripting attacks, and browser attacks through malicious scripts and spyware & adware.

## Importance

The computer security industry's revenue model has evolved from software sale to service subscription, the main value of which is constant update of signatures used in intrusion detection/prevention and anti-virus products. The technologies developed in this project can greatly improve the speed and accuracy of the production of attack signatures and patches, and thus fundamentally change the largely manual process currently used in the industry.

### New Approach

- Speculative dynamic disassembly to analyze/instrument Win32 binaries
- Extensive system call logging on the Windows platform
- Comprehensive data/control flow analysis for program traces and system call logs

### Research Impact

- A binary analysis and instrumentation infrastructure for obfuscated binaries
- A slicing algorithm based on generalized data/control dependency
- A general information flow tracking framework for analyzing and correlating system-wide events

The key thesis of the DIRA project is that it is possible to apply a general execution log analysis framework to derive attack signatures and patches for control hijacking attacks, web application attacks, and web browser attacks. In the case of control hijacking and web application attacks, the log is the program execution trace; in the case of browser attacks, the log is the time-ordered system events recorded at the library call or system call interface.

After an attack is detected, through a data flow analysis of the execution log, DIRA can first deduce the vulnerability being exploited, and then identify the portions of the network packets that are responsible for the detected attack. These bytes form the attack payloads. Through a control-flow analysis, DIRA can derive the *context* under which the detected attack takes place. Combined together, these two techniques could drastically speed up the process of attack signature and patch generation.