A PHP Sandbox for the Dissection of Web Malcode and Remote Access Trojans

Peter Wrench
Supervised by Prof Barry Irwin

Overview

- Problem description
- Objectives
- Approach
- Summary
- Questions

Problem Description

- PHP has become the language of choice for developers of Remote Access Trojans (or web shells)
- Proliferation of such shells has become more aggressive in recent years
- Developers disguise their malware by employing extensive code obfuscation techniques
- Sheer volume and clever code obfuscation frustrate efforts to identify and dissect malware

Objectives

- A sandbox-based system capable of safely executing and dissecting potentially malicious PHP code.
- A system for performing normalisation and deobfuscation of input code prior to execution.
- A basic reporting mechanism for feedback on any backdoors or other offensive features detected by the system.

Approach

- Gain an understanding of PHP and the functions commonly used for obfuscation
- Collect web shells for use as inputs to the system
- Find and study related work on semantic code analysis

Approach contd.

- Develop a deobfuscator for PHP code based on research into existing PHP decoders
- Construct the PHP sandbox environment with a focus on code isolation
- Test the final system against existing decoders

Summary

- The goal of the project is to produce a system consisting of three parts:
  - A deobfuscation segment capable of stripping away or overriding unnecessary function calls
  - A sandbox environment capable of detecting malicious features
  - A reporting mechanism that will produce both the deobfuscated code and a list of its major features

Questions