CSCE 689 2015A
Draft: Final Project Proposal

Robert A. Baykov
Department of Computer Science & Engineering
Texas A&M University

April 7, 2015

Contents

1 Introduction

1.1 Background

The broad goal of the final project, as outlined in prior documentation, is to build on top of existing reverse engineering tools and frameworks in order to contribute to the reversing community in a meaningful and useful way. To this end we are to extend prior work conducted by such tools, as well as the work contributed in previous assignments completed, serving as a course assignment extension and capstone to our work.

1.2 Overview

In this draft we will outline the current state of existing tools and frameworks (such as XED, Boomerang etc.), their classifications and uses. As well as several problems existing with the verification, utilization and integration of existing tools. We then provide our project outline, addressing several problems and describe our end goals in contributing to the reverse engineering community.

2 Problem

2.1 Background

There exist many reverse engineering tools, ranging in application, accuracy, robustness, usefulness and availability. The combination of these tools, coupled with the expertise of the reverse engineer provides a useful framework for the general reversing process.

Often, however, engineers do not have the benefit of intricate knowledge of obscure open source projects, detailed inspection tools and platform specific applications outside of their working environment. Because many such tools, useful in all manner of reverse engineering tasks, do not share a big-picture goal or envision support for programs outside of their narrow scope of application they often lack the ability to combine together in a coherent and useful way in order to provide *interoperability*.

2.2 Approach

To alleviate this problem of *interoperability* we propose a big picture, yet introductory, idea of something which will provide a useful, extensible and real world impact to reverse engineers everywhere: *an integrated reversing environment designed with the explicit purpose of bringing multiple tools together.*

Our initial proposal is to provide an environment composed of multiple plugin-style modules. Each module, represents an existing tool which we wrap to provide input and output in an automatic and configurable manner. In example, our reversing environment (RE) can wrap, parse and process the output of multiple file inspectors in order to provide more comprehensive and accurate binary file information to the engineer. The engineer can then utilize one, or multiple, disassembler (XED) to attempt to dump ASM codes. Additionally the engineer can utilize a plugin which provides the ability to invoke an online tool, such as an online decompiler. All this functionality can be specified

through a configuration file which details which tools are to be invoked, how and where to store their output (or whether a tool's output should serve as the input of another tool) and lastly on which binaries to run the configuration against. Note that we use the plural binaries, such an *RE* will provide a useful and automated way to process multiple files, currently a manual inspection affair.

The technical requirement of achieving such functionality needs an engineering effort in order to wrap the invocation of tools. We already have the capability of invoking online tools through HTTP APIs in many existing programming languages. We have the ability to invoke tools on multiple end hosts, not only a local windows or linux process. Lastly, we have the ability to parse, pipe and statistically process output in meaningful ways. The only requirement is putting the fragmented and disjoint reverse engineering toolspace together in a more consolidate environment.

3 Related Work

Of course, reverse engineering encompasses a bewildering array of tools for us to draw upon. In fact, there exists several attempts to provide an aggregate environment for reversing. The most notable is a linux distribution, Backtrack and later Kali Linux. Additionally repositories such as the Blackarch and Archassault projects provide a rich set of tools to select from.

Different from existing solution which merely provide the tools to invoke, we aim to integrate the invocation and processing of output from existing tools. Additionally, through the use of multiple end hosts we can bring platform specific tools together.

4 Summary

Our project is ambitious, this document serving as a draft for our motivation and envisioned end goal. As mentioned earlier, from a technical perspective, there is little which bars our ability to bring reversing tools together to form such a integrated reversing environment. Such a contribution would provide a meaningful tool for students, researchers and industry engineers.