Design of a File-Less Deployment of Packer/Loader Systems

Andrew Chapin
George Mason University
achapin2@gmu.edu

Carl Bai
George Mason University
cbai2@gmu.edu

Mitchell Palmer
George Mason University
mpalmer7@gmu.edu

Hunter Rowlette
George Mason University
hrowlet2@gmu.edu

Andre Herrera
George Mason University
aherre12@gmu.edu

Abstract. Our project fulfills a request to produce a software toolkit that allows for remote code execution completely in RAM and file transfer via a service running on a remote host. The goal of our stakeholder, Lockheed Martin Corp. (LM), is for our research to identify a unique way to accomplish this task. The following requirements were provided by LM: (1) the toolkit must be comprised of two separate executables – a "packer" and a "loader"; (2) the "packer" runs locally on Linux, compresses, then encrypts with AES via a user-provided password before sending data to remote hosts; (3) the "loader" runs on a Windows remote host as a service, receives incoming packed data, decrypts/decompresses, and executes any PE files entirely in RAM (i.e. without touching disk). Other loader operating systems were desired. We delivered. A four-part concept of operations was established: (1) a user selects a data block (e.g. executable file) and sends it to the packer where it is packed, (2) the now-packed data is sent over the internet to the remote host, (3) the remote host receives the packed data with the running loader service, (4) the loader decrypts the data block and will either run it in RAM or make it available on the Disk. Specifically, a CLI was built for the packer for user interaction, and a heartbeat process for the loader was designed in order to communicate uptime and availability of remote hosts to the user. Our toolkit, written in C++, implements the desired objectives of our stakeholder**. We utilized Libressl, filesystem, miniz, etc. libraries to accomplish the objectives.** Finally, quality assurance was established through integration and unit tests of the toolkit. The software was then handed over to LM for confirmation and testing in their environment. Alterations were made as requested and the final product was shipped. This paper provides an in-depth analysis of our product and our research into similar products and methodologies to our solution.

Abstract must be between 300 and 500 words and should include concise summary of the project including: Context, Stakeholders, Concepts of Operations, Requirements, Design (including trade-offs of alternate designs), Implementation, Verification, Validation, and Business Plans. Abstracts may include preliminary results. Abstracts must include Title *and* Authors Names*.*