**Password Hardening Project Implementation Report**

CS6238

October 19, 2015

Project Team:

Kyle Koza – [kyle.koza@security.gatech.edu](mailto:kyle.koza@security.gatech.edu)

Anant Lummis – [jimmy.lummis@security.gatech.edu](mailto:jimmy.lummis@security.gatech.edu)

# Background and Purpose

For the first project in the Fall 2015 Secure Computer Systems course, we were assigned a project to develop an authentication program which implements the password hardening scheme proposed by Monrose, Reiter, and Wetzel in their paper Password hardening based on keystroke dynamics [1].

The basis of the proposed hardening scheme is leveraging the speed in which a user types various characters contained within their password to calculate a hardened password (Hpwd). This implementation creates a second factor that is still part of the basic password authentication.

The purpose of this paper is to document specific implementation details from our project, as well as to document the challenges we faced and any design decisions made based on those challenges.

# Implementation Details

## Key Classes

There are four main object classes we developed including a history file object, an instruction table object, a user object, and a login object. It’s worth noting at this point that Python was used to develop this project.

The history file object implements the history file, which captures the last five instruction tables for a given user. The history file serves two purposes, first to allow the computation of a mean and standard deviation across previous instruction tables and allows the selection of specific features within the most current instruction table. Second, the history file is encrypted with Hpwd and the successful decryption of the history file represents a successful authentication.

The instruction table object implements the current instruction table that is used to compute the polynomial that ultimately defines Hpwd. In addition the individual fields within the instruction table are encrypted with the users password.

The user object represents a user and captures the users user name and password. A specific user is then associated with an instruction table and a history file.

The login object is the main method within the program is the object which is called to perform the authentication step given a user, their password, and associated features.

## Data Structures

To facilitate the various computations that needed to be performed we selected a queued array as the data type for the History File as well as the Instruction table.

## Encryption Functions

Several encryption functions were used throughout the program. To perform hashing of the user’s password we used SHA256 with a salt. To perform the encryption and decryption of the history file we used Fernet’s asymmetric encryption algorithm using the Hpwd as the key. Finally, to perform encryption and decryption of the Alpha and Beta values within the current Instruction Table we used the KDF HMAC function.

In order to select a large prime number for the various functions used to compute the Hpwd, we used Python crypto library to generate a 160 bit prime number.

## History File Padding

In order to ensure that history file does not leak information about a users password, specifically the length of the password, the history file should be padded to an equal length for all history files across all users.

In order to perform padding of the history file, we chose to pad with random characters collected from /dev/urandom to a standard size of one kilobyte.

Method of verifying instruction table successfully decrypted

# Implementation Challenges

The principle challenged we faced is we were not able to complete the Lagrange interpolation to retrieve the Hpwd from the set of coefficients collected from the instruction table. This means a proper Hpwd is not retrieved and the history file cannot successfully be decrypted to complete the strong authentication process.

# Future work

With some assistance and feedback from reviewing our code, we would like to complete the process of the Legrange interpolation to complete the project.

# Citations

1. Monrose, Fabian, Michael K. Reiter, and Susanne Wetzel. "Password hardening based on keystroke dynamics." International Journal of Information Security 1.2 (2002): 69-83.