Authentication by Typing Patterns Using Dynamic Text

Luke Knepper

Computer Systems Lab 2009-2010

Abstract

This project will analyze and test the accuracy of typing pattern authentication methods when the user is prompted to type dynamically generated text. The program will generate a random (i.e. dynamic) segment of text, measure the user's keystrokes while they type this text, and feed the typing data through a neural network to determine authentication status. Experimentation will be done to determine the most accurate neural network structures and data collection conditions. This process will be an improvement on normal typing pattern authentication techniques, which use static passwords rather than dynamic text.

Background and Introduction

Typing patterns differ by person. People naturally hold down specific keys for specific times and take longer between different keystrokes. These typing characteristics can be, and have been, used for authentication purposes. However, current commercial methods (e.g. Psylock) only measure typing patterns on a single, static passphrase, making them easy to hack using a simple keylogger. In this project, I propose an improvement to the traditional typing authentication methods. Rather than Measuring the typing characteristics while a user types a simple passphrase and/or username, which are both static (i.e. unchanging), I will test the Accuracy of a method which generates dynamic (i.e. random) text which the User is prompted to type. The system then measures the keystroke data while They type this dynamic text. This offers a considerable advantage because A keylogger can no longer record and playback a simple typing sequence to Gain access to the system, and instead will have to mimic the user's typing Exactly. No research has yet been found on the accuracy of using dynamic Blocks of text.

Previous research that has been found on the accuracy of authenticating by Typing patterns while using static phrases mostly concurred on two results: Neural-network algorithms are the optimal approach for this goal, and such Methods are on average 80-90% accurate. These findings suggest that Typing characteristic authentication is not accurate enough to fully replace Passwords and usernames, but can offer a powerful boost to security.

Neural networks are modeled after the human brain. They are composed of Nodes (i.e. neurons) which take inputs from previous nodes, compute a simple function from these Inputs, multiply the result by a given weight and then pass the result on to the next nodes (see Fig. 3 for a simple diagram).

Results and Conclusions

In the first quarter, I completed a proof of concept program to test the theory that Users can be differentiated by their typing patterns. This program prompts two users to type a sentence and records their typing data while they do that. It trains A neural network with their typing data to distinguish the two users. It then prompts the users to choose a "mystery user" from between themselves to type a third sentence (the identity of the user unknown to the program). It then feeds the mystery users' typing data through the neural network and determines Whether the first or second user was the mystery user based on the result from The network. This is only a proof of concept program, using the simplest single-layer network possible. The program has an accuracy of 18 / 20 = 90%, which shows that the concept is accurate enough to be used on a larger scale, but still Leaves much to be desired. (See Fig. 1)

In the second quarter, I completed a data collection program (See Fig. 2) which Will collect mass amounts of data on which to run tests. The program prompts The user with a short amount of text and asks them to type it into the box. While They are typing, it records their keystrokes in the following format: "Key-# / Key-direction / Time-in-millis." This format is flexible and allows for Different types of programs to measure different characteristics from the data. Once the user is done, the program sends the typing data off to the server, Where it is stored in the form of a text file via a PHP script.

I currently have a couple dozen data files and will be getting many more in the Time to come. I am in the process of coding the automated testing system Described in the Discussion section. This system will determine the accuracy of This method when using different network structures and corpus sizes.

Figures

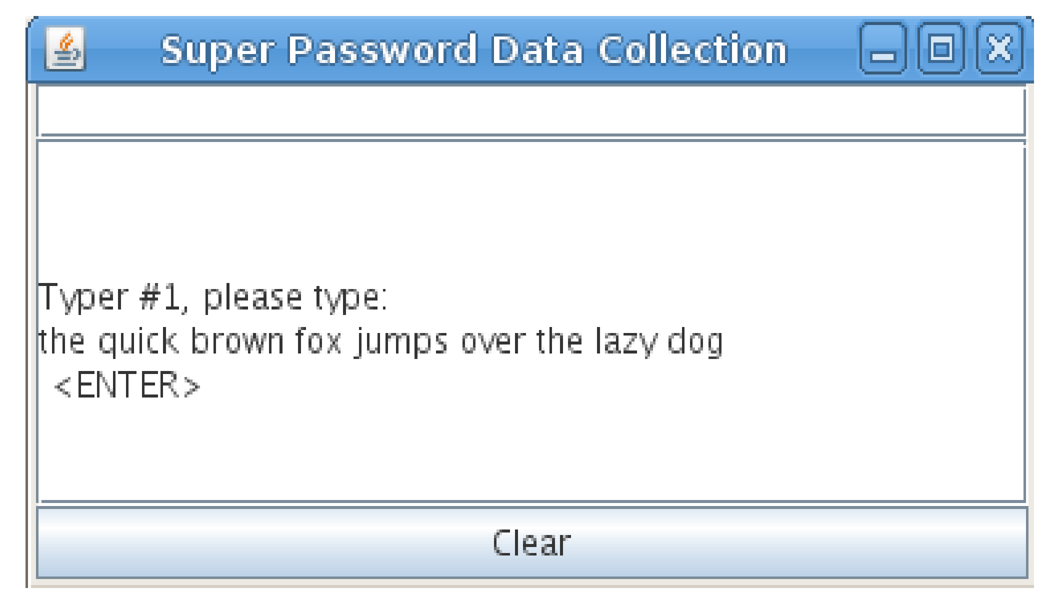

Fig 1: The proof-of-concept program

Fig 2: The data collection program

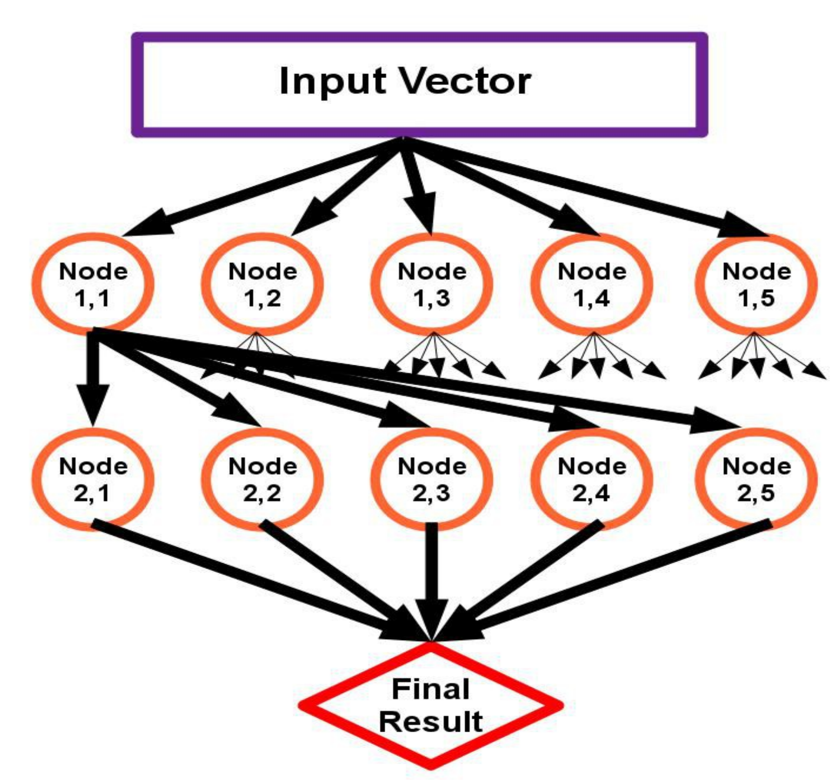

Fig 3: An outline of a simple neural network

Discussion

Three things will be determined from this experimentation:

1. Which network structure is most accurate for this purpose
2. Statistics on the accuracy of the neural network in this purpose
3. The optimal corpus size (i.e. length of text) for this purpose

In order to test these things, I will first need a lot of data to run Through the neural networks. I am collecting this data from my Data collection applet. An automated testing system will then be Developed to create (i.e. train) a neural network of each type for each set of data. Once the networks are created, they will be Tested by running different amounts of typing data from the Collected data through each neural network and recording the Ratio of success and failure (i.e. the accuracy). The ratios for Each network type and corpus size will then be compared to Determine the goals above.

Once this is done, a mock authentication system can be created and more experimentation can be done using this system.