**Keylogger project progress report**

**Report**

Dr. Baris Aksanli Assistant Professor,

Electrical and Computer Engineering Department

San Diego State University

**Subject:** Embedded RTOS Project

Related work:

Through our research we found a number of software projects/frameworks that we can use do develop our keylogger and serve as a reference. We are using a library for X11 called xlib as our primary way of interacting with the keyboard. Xlib gives functions for receiving keyboard scan codes and allows us to convert from scan codes to the actual value of the key that was pressed. It also supports international keyboards such as Norwegian.

We also found a couple of open source projects to use as reference material for how to use the xlib library. Xev is a standard Xorg program which prints out the contents of XEvents. This gave us a reference for how we can parse an XEvent to a usable key value. Xinput serves a similar purpose however it’s -–test-xi2 option gives us a reference for how to get XEvents from the whole system rather than just our program allowing us to log key presses from other programs.

Accomplishments so far:

Using the X11 event system and the reference programs (xev and xinput), we created a keylogger that provides international support. Regardless of what window is focused or if the screen is locked, we get all XEvents that happen on the system from the X11 server. Using a few X11 library function calls we can parse the keyboard XEvents into a description of the key that was pressed. This description is dependent on the keyboard layout, which means if we use a non-US keyboard we can still get the correct key that was pressed.

We get notifications of when the shift key is pressed and released, however when the shift key is modifying the function of a key we do not get the modified key value. This is fine for US keyboard since when the ‘a’ key is pressed along with the shift key we know the result should be ‘A’. But when we are using international keyboards the modified value of the key depends on the layout of the keyboard so we are unable to determine the value of every key press on every keyboard.

We also developed a TCP\_IP server and client using the network sockets built into the Linux kernel so that we could send the recorded keys from the target computer back to our server.

Future deliveries and milestones:

* Develop keyboard interrupt handler (11/13/2019)
* Adapt the interrupt handler into a device driver (11/29/2019)
* Develop benchmarks to compare speed between the custom device driver and our XEvent based keyboard logger (12/06/2019)

We will be skipping milestones 2 and 3 from our original progress report so that we can focus more time on our device driver.

Final Project Demo:

We will demonstrate our keyboard device driver on a Linux computer and show benchmark results of our testing with the device driver and the X11 event system.