

## Article

# Design of USB Protection System for Physical Devices

Poornima<sup>1</sup>, Kishan Singh<sup>2</sup>

<sup>1,2</sup>Student, Department of Electronics and Communication Engineering, Guru Nanak Dev Engineering College Bidar, Karnataka.

## INFO

**Corresponding Author:**

Poornima, Department of Electronics and Communication Engineering, Guru Nanak Dev Engineering College Bidar, Karnataka.

**E-mail Id:**

kishanskrish@gmail.com

**How to cite this article:**

Poornima, Singh K. Design of USB Protection System for Physical Devices. *J Adv Res Power Electro Power Sys* 2020; 7(4): 28-31.

Date of Submission: 2020-08-28

Date of Acceptance: 2020-09-02

## ABSTRACT

In this paper we are discussing about USB encryption techniques using three levels of security. Our system builds by using username, password, and biometric data like human face using camera and IP address of the machines where USB devices are connected. When any file transfer from one USB to another USB then it asked for username/ password, it will verify the face of the person who connected the USB device to system and it also verify the IP address of the system where the USB device is connected. When everything is matched then only USB device start working on the system and data exchange is possible. Our system is designed by using ARM 32-bit micro controller which supports different features and algorithms for the development of automotive secure embedded systems.

**Keywords:** Interference Modeling, Physical Layer Security, Stochastic Geometry

## Introduction

In current generation data transfer will play a key role in each and every sector with security. In order to share data one device to other device most of the people are using store devices especially The Universal Serial Bus (USB) is a ubiquitous interface standard being widely used for connecting storage to consumer devices. Because of its convenience and ease of connectivity, the USB port has become an essential component of consumer electronics devices such as flash disks, keyboards, cell phones, chargers, speakers, and printers. However, the USB interface has the following three weaknesses when it is used for consumer storage devices : anyone (e.g., an unauthorized user) could read or steal confidential information easily since the information is stored in plaintext format; and (2) an adversary could intercept or attack the transmitted information since the transmit channel between the device and the computer is not secure. Therefore, despite their practicality, USB Mass Storage Devices (MSDs) have been prohibited in an enormous number of environments.<sup>1</sup>

To tackle these issues, and expand the uses of USB shopper stockpiling gadgets, a confirmation convention can be execute to guarantee secure correspondences between the gadget and the PC. Since the time Lamport proposed the main validation convention, numerous confirmation conventions have been proposed for various applications.

Hwang and Li planned an authentication protocol using a smart card. Though, their protocol could not endure the masquerade attack. To get better security, Ku and Chen planned an enhanced authentication protocol using a smart card. Afterward, Yoon, Ryu found that Ku and Chen's improved validation convention was anyway defenseless against the equal meeting assault, and in this way proposed another confirmation convention utilizing a brilliant card, yet Hsiang and Shih later showed that it was defenseless against three sorts of assaults.<sup>2</sup>

Hsiang and Shih proposed their new verification convention utilizing a keen card; nonetheless, Shim found that Hsiang and Shih's convention was helpless against the disconnected secret word speculating assault. Kim and Hong proposed a

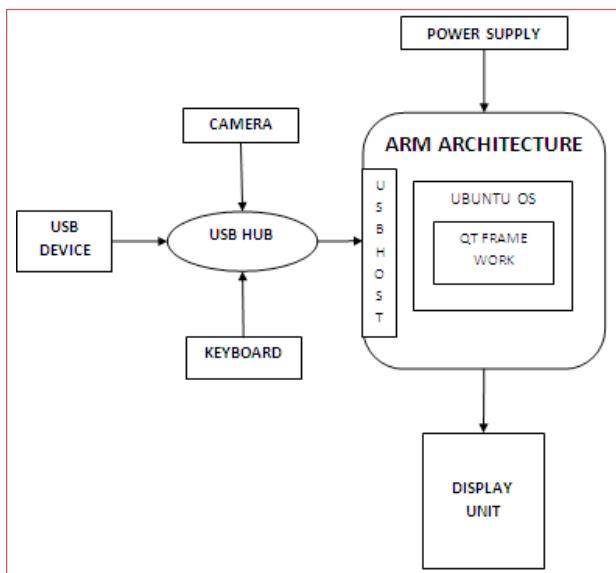


multimodal biometric verification convention that utilized teeth, picture and voice in versatile conditions.<sup>3</sup>

To improve execution, Kim, Chung and Hong, and Lee, Kim and Cho proposed two new conventions that all pre-owned individual explicit verification utilizing individual biometric qualities, for example, face, teeth, and voice. Notwithstanding, every one of these conventions are not preferably appropriate for USBMSDs on the grounds that their put away data can without much of a stretch be perused out or require critical nearby complex calculations. To ensure the protection of a record moved to a capacity gadget, Yang, Wu and Chiu proposed the principal secure control convention utilizing the Schnorr signature conspire.<sup>5</sup>

To solve those problems, our system builds with three-factor authentication will help to transfer data with high security for authentication.<sup>6</sup>

## Block Diagram



### **Figure 1. Block diagram**

## **Hardware Implementation**

## ARM Architecture

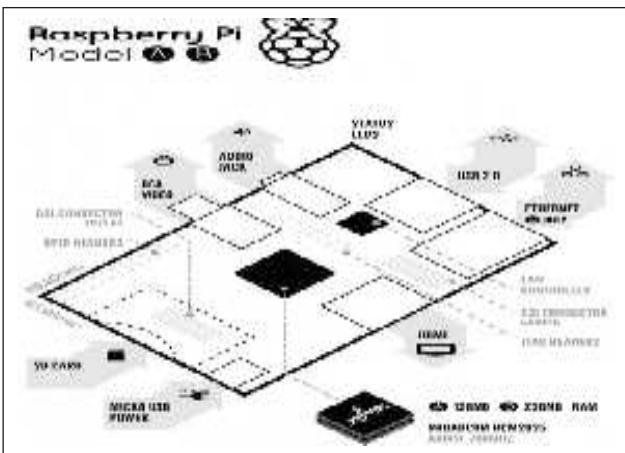
The Raspberry Pi is a credit card sized single-board PC created in the UK by the Raspberry Pi Foundation with the aim of advancing the teaching of essential software engineering in schools. The Raspberry Pi is made in two board arrangements through authorized assembling managers Newark element 14.<sup>7</sup>

These organizations sell the Raspberry Pionline. Egoman produces an adaptation for dispersion exclusively in China and Taiwan, which can be recognized from other Piece by their red shading and absence of FCC/CE stamps. The equipment is the equivalent across all makers. The Raspberry Pi has a BroadcomBCM2835 framework on a chip It does exclude an underlying hard circle or strong

state drive, however utilizes a SD card for booting and diligent storage.



## Figure 2.Arm Architecture



**Figure 3.Raspberry PI model**

The Foundation provides Debian and Arch Linux ARM dispersions for download. Devices are accessible for Python as the principle programming language, with help for BBC BASIC (by means of the RISC OS picture or the Brandy Basic clone for Linux), C, Java and Perl.<sup>8</sup>

## Ethernet

#### Ethernet LAN Features:

- Bus topology, Wired LAN in IEEE 802.3 physical layer standard
  - 10 Mbps, 100 Mbps (Unshielded and Shielded wires) and 4 Gbps (in twisted pair wiring mode)
  - Broadcast medium-Passive, Wired connections based
  - Frame format like the IEEE 802.2
  - SNMP (Simple Network Management Protocol) Open system (therefore allows equipment of different specifications)
  - Each one connected to a common communication channel in the network listens and if the channel is idle then transmits. If not idle, waits and tries again
  - Multi access is like in a Packet switched network

IIVC Driver Camera

A UVC (or Universal Video Class) driver is a USB-category driver. A driver enables a device, such as your webcam,

to communicate with your computer's operating system. And USB (or Universal Serial Bus) is a common type of connection that allows for high-speed data transfer. Most current operating systems support UVC. Although UVC is a relatively new format, it is quickly becoming common.<sup>9</sup>



**Figure 4.UVC Driver Camera**

#### There are Two Kinds of Webcam Drivers

The one included with the installation disc that came with your product. For your webcam to work properly, this driver requires some time to install. It is specifically tuned for your webcam, designed by your webcam manufacturer and optimized for webcam performance.

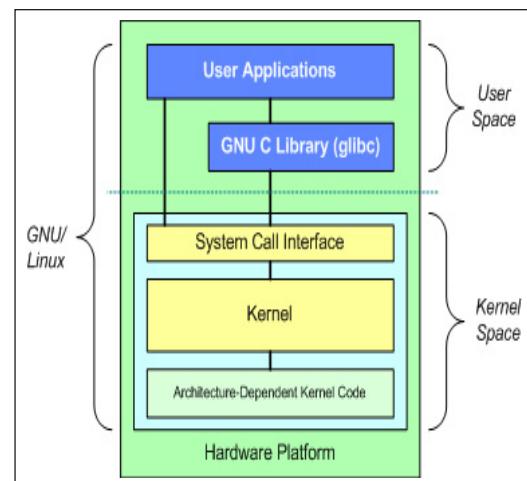
A UVC driver:- You can only use one driver at a time, but either one will allow you to use your webcam with various applications.

#### Software Requirements

##### Linux Operating System

Linux or GNU/Linux is a free and open source software operating system for computers. The operating system is a collection of the basic instructions that tell the electronic parts of the computer what to do and how to work. Free and Open Source Software (FOSS) means that everyone has the freedom to use it, see how it works, and changes it. There is a lot of software for Linux, and since Linux is free software it means that none of the software will put any license restrictions on users. This is one of the reasons why many people like to use Linux.

A Linux-based framework is a particular Unix-like working framework. It determines quite a bit of its essential plan from standards set up in UNIX during the 1970s and 1980s. Such a framework utilizes a solid bit, the Linux bit, which handles measure control, systems administration, and fringe and document framework access. Gadget drivers are either coordinated straightforwardly with the piece or added as modules stacked while the framework is running.<sup>10</sup>



**Figure 5.Architecture of Linux Operating System**

#### Qt for Embedded Linux

Qt is a cross-stage application structure that is generally utilized for creating application programming with a graphical UI (GUI) (in which cases Qt is named a gadget toolbox), and furthermore utilized for creating non-GUI projects, for example, order line instruments and consoles for workers. Qt utilizes standard C++ yet utilizes an extraordinary code generator (called the Meta Object Compiler, or MOC) along with a few macros to improve the language. Qt can likewise be utilized in a few other programming dialects by means of language ties. It runs on the significant work area stages and a portion of the versatile stages. Non-GUI highlights incorporate SQL information base access, XML parsing, string the executives, network uphold, and a bound together cross-stage application programming interface for document dealing with. It has broad internationalization uphold.<sup>11</sup>

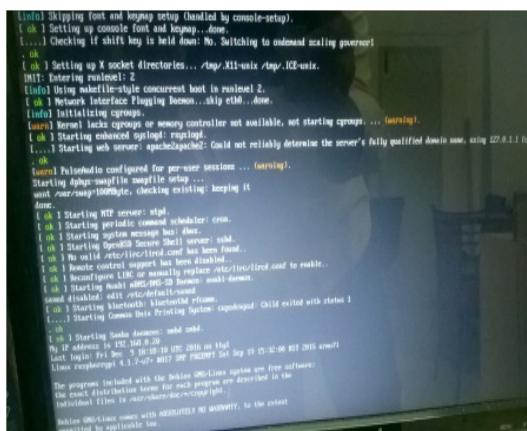
#### Open CV

Open CV (Open Source Computer Vision) is a library of programming capacities for ongoing PC vision. It is created by Willow Garage, which is likewise the association behind the popular Robot Operating System (ROS). Presently you'd state MATLAB additionally can-do Image Processing, at that point why open CV? Expressed underneath are a few contrasts between both. When you experience them, you can choose for yourself. Preferences of OpenCV over MATLAB (Collected from different online journals/discussions):<sup>12</sup>

Speed Matlab is based on Java, and Java is based upon C. Along these lines, when you run a Matlab program, your PC is caught up with attempting to decipher all that Matlab code. At that point it transforms it into Java, and afterward at last executes the code. Open CV then again, is essentially a library of capacities written in C/C++. You are nearer to straight forwardly give machine language code to the PC to get executed. So at last you complete more picture preparing

for your PCs handling cycles, and not more deciphering. Thus, programs written in Open CV run a lot quicker than comparative projects written in Matlab. All in all, end? Open CV is damn quick with regards to speed of execution. For instance, we may compose a little program to recognize individuals' grins in an arrangement of video outlines. In Matlab, we would ordinarily get 3-4 edges investigated every second. In Open CV, we would get in any event 30 edges for each second, bringing about ongoing recognition.

## Result



```

[  ok ] Setting font and keymap (handled by console-setup).
[  ok ] Setting up console font and keymap... done.
[....] Checking if shift key is held down: No. Switching to minimal scaling governor!
[  ok ]
[  ok ] Setting up X socket directories... /tmp/.X11-unix /tmp/.ICE-unix.
[  ok ] Entering runlevel: 2
[  ok ] Network interface Flamingo... skip eth0... done.
[  info] Initializing cpusnoop.
[  warn] Kernel lacks cpusnoop or memory controller not available, not starting cpusnoop... (warning).
[  info] Starting enhanced splashlogd.
[  info] Starting web servers: apache2@pi: Could not reliably determine the server's fully qualified domain name, using 127.0.1.1 for ServerName.
[  info] Pidfile /var/run/apache2.pid configured for per-user instances ... (warning).
Starting dhcpcd: skipping existing; keeping it.
want over/swap/100MByte, checking existing; keeping it.
[  info] Name:
[  ok ] Starting NTP server: stpd.
[  ok ] Starting periodic command scheduler: cron.
[  ok ] Starting system message bus: dbus.
[  ok ] Starting OpenRC init system: openrc.
[  ok ] No such file or directory: /etc/init.d/ircd.conf has been found.
[  ok ] Remote control support has been disabled.
[  ok ] Configure LIRC as remote receiver: /etc/init.d/lircd.conf to enable.
[  ok ] Starting NetworkManager: (wlan0) connection auto-detect.
[  ok ] Starting Bluetooth: bluetoothd: (wlan0) connection auto-detect.
[  ok ] Starting Connman: (wlan0) connection auto-detect.
[  ok ] Starting Samba daemon: (wlan0) connection auto-detect.
My IP address is 192.168.10.10 (UTC 2016-05-25 02:06:05 2015 armv7l)
last login: Tue May 25 02:06:06 BST 2015 pi@raspberrypi:~$
```

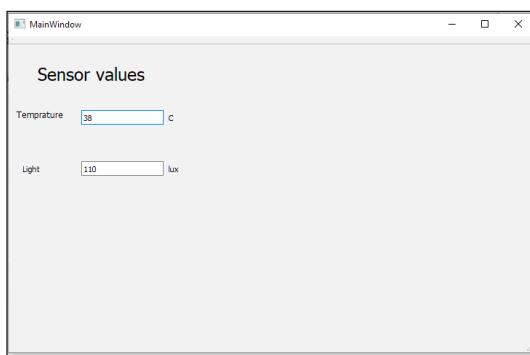
The progress included with the Debian OS/Line option are free software. Direct distribution terms for any software described in the individual files is available at [creativecommons.org/licenses/by-sa/4.0/](http://creativecommons.org/licenses/by-sa/4.0/).

Below OS/Line comes with ABSOLUTELY NO WARRANTY, to the extent permitted by applicable law.

**Figure 6. Software Setup**



**Figure 7. Hardware Setup**



**Figure 8. Hardware Setup**

## Conclusion

The paper "Design of USB protection system for physical devices" has been successfully designed and tested. It has been developed by integrating features of all the

hardware components and software used. Presence of every module has been reasoned out and placed carefully thus contributing to the best working of the unit. Secondly, using highly advanced ARM board and with the help of growing technology the project has been successfully implemented.

## References

1. Xiao J, Zeng FS. Design and implementation of embedded Web server. Computer Science & Education (ICCSE), 2012 7<sup>th</sup> International Conference on, 2012; 479-482.
2. Limpraptono FY, Sudibyo H, Ratna AAP et al. The design of embedded web server for remote laboratories microcontroller system experiment. TENCON 2011, 2011 IEEE Region 10 Conference, 1198, 1202. 2011; 21-24.
3. Soni RS, Asati D. Development of Embedded Web Server Configured on FPGA Using Soft-core Processor and Web Client on PC. *International Journal of Engineering and Advanced Technology (IJEAT)* 2012; 1(5). ISSN: 2249-8958.
4. Ugurel G, Bazlamaççı CF. Context Switching Time and Memory Footprint Comparison of Xilkernel and µC/OS-II on MicroBlaze. ELECO 2011 7<sup>th</sup> International Conference on Electrical and Electronics Engineering, 1-4 December, Bursa, TURKEY
5. Xilinx Company. OS and Libraries Document Collection" UG643. 2012. [internet] Available at <http://www.xilinx.com/support/>
6. lwIP examples using RAW and Socket APIs: <http://savannah.nongnu.org/projects/lwip/>
7. MacMahon S, Zang N, Sarangi A. Light Weight IP (lwIP) Application Examples. XAPP 1026 (V3.1) 2011.
8. <http://developer.yahoo.com/yui/2/>
9. Alzarouni M. The reality of risks from consented use of USB devices. in Proc. 4<sup>th</sup> in Proc. 4<sup>th</sup> Australian Information Security Management Conference, 2006; 312-317.
10. Wu W Lee, Tsaur WJ. A secure authentication scheme with anonymity for wireless communications. IEEE Communications Letters, 2008; 12(10): 722-723.
11. Hwang MS, Li LH. A new remote user authentication scheme using smart cards. *IEEE Trans Consumer Electron* 2000; 46(1): 28-30.
12. Ku WC, Chen SM. Weaknesses and improvements of an efficient password based remote user authentication scheme using smart cards. *IEEE Trans Consumer Electron* 2004; 50(1): 204-207.
13. Ryu YE, Yoo K. Further improvement of an efficient password based remote user authentication scheme using smart cards. *IEEE Trans Consumer Electron* 2004; 50(2): 612-614.