

# Disable USB on lockscreen

## Name:

George-Cristian Muraru

## Contact Information:

- **Email:** [muraruGeorgec@gmail.com](mailto:muraruGeorgec@gmail.com)
- **Gnome wiki:** George-Cristian Muraru
- **Twitter:** <https://twitter.com/MuraruGeorge>
- **FreeNode IRC Nickname:** georgemuraru
- **Location:** Bucharest, Romania, EET (GMT + 2:00)

## Introduction

More and more people are studying methods on how to block or clean a malicious software. But, today, the threat can even come from something more *physical*.

All the devices pose a risk. Even a *small* USB device can have a *huge* impact on your system. Regarding the intention of a person he may use a modified USB device to “*play*” with your personal computer [0].

You will never have full security for a computer, but there are methods that can enhance it. My “*Introduction to operating systems*” professor have a say: “*You can never have a computer with no vulnerability. From the moment you open your computer and connect it to the internet you may be hacked. Even if you don’t open your PC, your hard disk may still be stolen. Maybe the best solution is a chained computer, thrown in the ocean. Even then, this method will not guarantee 100% security.*”

There must always be an equilibrium between **security** and **usability**. By creating a too secure medium the workflow of the user may be disturbed, thus the user will disable that security method to keep his work pace.

The project **Disable USB on lockscreen** doesn't have this throwback. When the computer is locked, it is clear that the user will not be doing any work on his PC involving USB ports.

Let's assume the following situation:

*You have come with your laptop at a party and in the meantime you had to go to the bathroom.*

*Since you locked your laptop, no one can access it, right? Well...it's **wrong!***

*A person in the room wants to steal all your data and then destroy your laptop with a super-virus (well...programmers always consider the worst case scenarios). He has access to your laptop only through your USB ports, so he will try to connect a device that contains some malware that will triggered automatically by the USB recognition. He connects the device and steals all your work and after destroys your laptop.*

*All of this could have been avoided if the user had a USB locking application.*

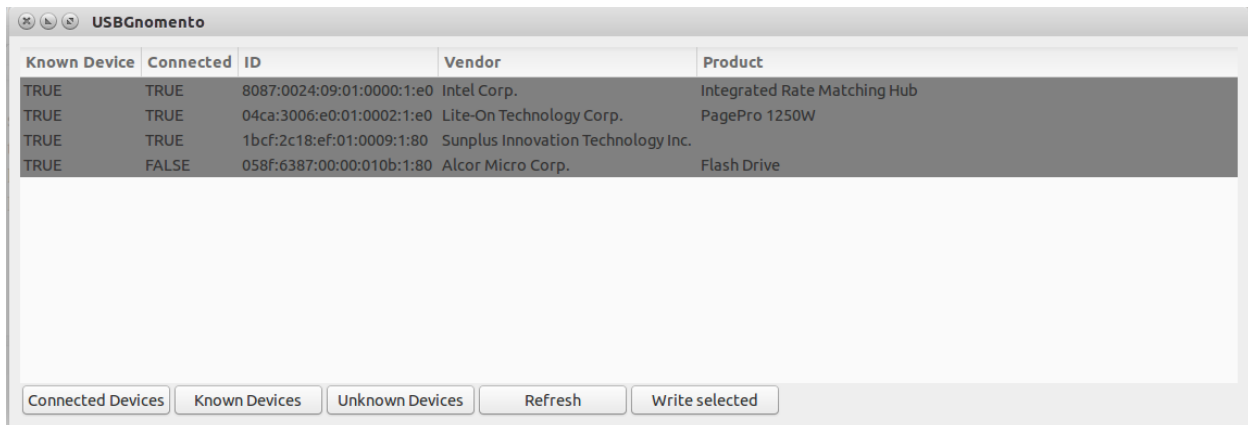
The project aims to protect you from this kind of situations. If someone will try to connect a USB device with unknown descriptors (your PC has not seen them before), then that port will be blocked and the attacker will have no chance to steal/destroy your data through USB ports.

Under the guidance of Tobias Mueller, the final product of the project will bring a more secure environment for users and will put GNOME on a higher position on the security domain.

## USBMemento

Because the projects repository is not yet available, I have come with a “wanna-be application”[1] that can show the status of all known USB devices that were/are connected to the USB ports.

Currently the application can show a list of all the devices (even if they were or are connected):



The screenshot shows a window titled "USBGnomento" with a table of USB devices. The table has five columns: "Known Device", "Connected", "ID", "Vendor", and "Product". There are four rows of data. Below the table are five buttons: "Connected Devices", "Known Devices", "Unknown Devices", "Refresh", and "Write selected".

Known Device	Connected	ID	Vendor	Product
TRUE	TRUE	8087:0024:09:01:0000:1:e0	Intel Corp.	Integrated Rate Matching Hub
TRUE	TRUE	04ca:3006:e0:01:0002:1:e0	Lite-On Technology Corp.	PagePro 1250W
TRUE	TRUE	1bcf:2c18:ef:01:0009:1:80	Sunplus Innovation Technology Inc.	
TRUE	FALSE	058f:6387:00:00:010b:1:80	Alcor Micro Corp.	Flash Drive

The shown columns are:

- *Known Device* - a boolean value to show if the device is on a list of *known devices* (this list is meant to be a collection of trusted USB devices)
- *Connected* - a boolean value to show if the device is currently connected
- *ID* - a string value composed of some of the device descriptors: idVendor, idProduct, bDeviceClass, bDeviceProtocol, bcdDevice, bConfigurationValue, bmAttributes. Here we wanted to create a more unique identifier for each “seen” device (this ID will be used as a key to identify trusted devices and to permit them to connect to the USB)
- *Vendor* - a string value that represent the vendor of the product
- *Product* - a string value that represents the name of the product

The last two columns are got by querying the `/var/lib/usbutils`.

By selecting *Write selected* the program will store the selected USB device information into a file, so that at the next run it will know the device have already been seen.

For the **USB Locking** project we can adjust this feature so that if a device is on the list of *known devices* it can be trusted and is authorized to connect to the port.

Here is a glimpse at the file that contains the *known devices*:

```
"04ca:3006:e0:01:0002:1:e0": {
  "Product": "PagePro 1250W",
  "Vendor": "Lite-On Technology Corp."
},
"8087:0024:09:01:0000:1:e0": {
  "Product": "Integrated Rate Matching Hub",
  "Vendor": "Intel Corp."
},
"058f:6387:00:00:010b:1:80": {
  "Product": "Flash Drive",
  "Vendor": "Alcor Micro Corp."
},
"1bcf:2c18:ef:01:0009:1:80": {
  "Vendor": "Sunplus Innovation Technology Inc."
}
}
~
~
~
~
~
~
"known_devices" [noeol] 17L, 467C                               1,1                               All
```

Each key from the dictionary represents the unique identifier for a USB device and in each entry represents another dictionary that will keep user info about the device (here are only the Product Name and Vendor Name).

The technologies I used to make *USBMemento* are **Python** and **Gtk+**.

Using some of this application features I can implement a **USB Locking** system that forbids some persons (with bad intentions) to use some USB device to get into your computer. The mechanics will work based on the principle that a known device (a trusted one) will be allowed to connect even when the screen is locked (like a USB keyboard or mouse), but an unknown one will not be allowed.

## Tentative Timeline

*Present - May 25*

- continue integrating into the community;
- getting used to Gtk+;
- research about the best approach to block a USB port;
- discussion with the mentor about ideas of implementation;

*Week 1-2: 25 May - 8 June*

- optimization and minor modification for the module that is going to check if a connected USB device has been seen before (trying to improve the code for the USBGnomento);
- finishing the module;

*Week 2 - 4: 8 June - 22 June*

- Designing an efficient blocking method for an unknown USB device; if is not on our *known list* and the screen is locked then it will be blocked;
- Designing an user system warning when he unlocks the screen to see if someone wanted to connect to his PC;

*Week 4 - 6: 22 June - 6 July*

- Trying to make some options for the user like: blocking all the ports despite the fact that the laptop is not locked, to decide to block only some specific ports, to block some devices with specific descriptors;
- Preparing for the **Mid-term evaluation**;
- Modify the existing code based on the reviews;

*Week 6 - 8: 6 July - 20 July*

- The hard step: integrating the application;
- Dealing with eventual bugs that come along the way

*Week 10 - 12:*

- Testing the application with various USB devices;
- Double-checking all user-options (features);
- Writing more documentation;
- Cleaning the code to make it and wrapping everything up;

During 30.05-2015 - 19.06.2015 final exams are held at my university, so I will spend a little less time on the project. However, I will devote an appropriate amount of time so that any request regarding features will be implemented in time.

## **About me**

I am a second year student at the “Politehnica” University of Bucharest, currently pursuing a bachelor’s degree in Computer Science, expecting graduating in 2017.

I first came in contact with Linux a year and a half ago (at a course from my university: “Introduction to operating systems”) and from that moment I left the dark side and “embraced” the terminal with his big advantages (the speed of doing a task, making different kind of scripts and automatising them, faster troubleshooting etc.). After finishing one of my favorite courses, I took an optional one, LPIC organized by ROSEdu[2] (Romanian Open Source Education), where we were prepared for the LPIC1 certification.

After taking the exam[3], the LPIC team proposed to me[4] to become one of the them, to teach students about Linux and to prepare them for the certification.

Last summer I have also participated in some workshops, “*Introduction to Machine Learning*”[5], where we were presented an overview about ML general notions. We learned also about Computer Vision and Pattern Recognition. At this workshop we worked in **Python** where we used the libraries *scikit-learn* and *opencv*.

Another workshop that I have participated in was about “*System Administration (with focus on Linux)*”[6] where I have expanded my knowledge about the duties of a sysadmin and about **shell scripting**.

In June, last year, I also attended “Open Data Summer School”[7], where we worked with **Python, DBpedia, Snorql**. We learned basic stuff about **NLP, Ontologies, Data Cleaning and Normalization**.

I like the idea of *open-source* because it connects people with the same ideas and the same interests. One person can improve the others one work. The whole process can be seen as building a house, brick by brick (“commit with commit”), where the final product will be a majestic beautiful house (the final application).

I am proficient in the following languages: **Python, C** (I was an undergraduate teaching assistant for the course “*Introduction to Computer Programming*”), **Java, shell scripting**. I also have some experience in: **C++, Ruby**.

My areas of interest are the following:

- *Algorithms* (essentials to any programmer);
- *Data structures*;
- Artificial Intelligence;
- Networking
- Security

I want to participate in the Google Summer of Code program, so I can “solve” a real security problem, a problem that a lot of people did not look into it that much and that can cause big amount of damage: malicious hardware devices. I think I am suitable for this project because of my background in Computer Science and because I am an ambitious and dedicated person. Also the project mainly covers one of my favorite topics (**Security**) and by contributing to it I will further develop my knowledge regarding this area.

I am also looking forward dealing with challenging problems that I may come across and from which there are much to be learnt.

- [0] <http://securityaffairs.co/wordpress/28855/hacking/usb-attack-code-released.html>
- [1] <https://github.com/murarugeorgec/USB-checking>
- [2] <http://www.rosedu.org/>
- [3] <https://cs.lpi.org/caf/Xamman/certification> (LPI ID: LPI000315784, Verification Code: 5yltmve39j)
- [4] <http://www.lpic.ro/wiki/info/echipa>
- [5] <http://workshop.rosedu.org/2014/sesiuni/ml>
- [6] <http://workshop.rosedu.org/2014/sesiuni/sysadmin>
- [7] <http://wiki.cs.pub.ro/studenti/summer-schools/2014/open-data>