Review on USB Storage Device Control on Linux

Dhiraj Kumar[1], Yashpal Singh[2]
[1,2]Department of Computer Science &Engineering,
Ganga Institute of Technology and Management,
Kablana, Jhajjar, Haryana, India

Abstract: **The world of communication is moving towards standardization of hardware ports. All kind of communication is now using USB as the port as it is universally recognized hardware medium of communication. It is become flexible and easy to use kind of things with portable USB storage devices to copy data from one system to another system. It is possible to copy data within seconds with the help of portable USB flash memory devices. It has leaded insecurity of data storage on computer system. Various surveys has shown after network copy only USB data copy has made data insecure on computer. It is also the source of malwares in the system. To disable the USB ports is not the solution to this problem because almost all peripheral devices now uses the USB ports for communication. So, we have implemented a system which has complete USB storage enable and disable control for Linux operating system. The administrator will decide the storage devices connected to USB must be enabled or disabled .We experimented the algorithm on Linux kernel version 3.9 onwards on Debian based distributions. We have got 100% success rate of the said system with 0% performance degradation.**

Keywords: linux, debian,usb storage.

I. INTRODUCTION

With the rapid development of information technology, the communication medium has changed a lot. As communication is very important aspect of each and every work, the medium of communication has to be more efficient and more secure. Now-a-days we all had been using the USB as a port for communication for example communication between user and computers, communication between mobile and computers etc. All the peripheral hardware devices are also connected using the USB ports. So that is the reason the USB as a port are more put forward to standardisation. The most important thing that is storing of a data is also done with the USB storage devices. It makes easy to accessible to the host computing device to enable the file transfer between the two. When the USB storage devices are attached to the host computing device it appears as an external drive, to store the data. Like to copy data from one computer to another and from one computer to any storage devices. The demand for these USB storage devices has been tremendously increased. The manufacturers had also increased their production rate of these storage devices with more data storage space. But with all these flexibility, risks has also come into addition. As we are mostly using USB storage devices to transfer data or to keep backup of data, it can lead to the leakage of data. The leakage of data makes the information insecurity. This flexibility of directly accessible of copying any data from the host computing device can make the insecurity of the data. It allows the unauthorised users to access the data and copy the data from your computing device and misuse it in any ways. As now the companies in particular are at more risk when any sensitive data are easily copied with the help of these USB storage devices by the employees and taken out of the office and misuse it or being given to any other companies. This can lead to deal with the worst consequences of losing the information that can include the customer data, business plans, financial information or some confidential documented information about company. The another risk which can lead to information loss is computer virus and other malicious software. As easily we can transfer the files between the USB storage device and the computing device at the same point of easiness the viruses can be transfer from the USB device to your computing devices. These devices has become the primary means of transmission of viruses and malware. Whenever the malware gets onto your storage device it may infect your computing devices as the USB drive is subsequently plugged. These viruses and any malicious software can corrupt your data which leads to data loss. If someone intentionally wants to corrupt all your data it just needs to plugged the storage device which contains the viruses and transfer it to your computing device. Now a days the loss of data is mainly through the computer viruses as told by the most of the surveys. For all these reasons of information insecurity, USB drives are used in a wrong manner. As information is the most valuable asset, it has to be more secure and confidential. To make the data more secure on your computers, one way is that to disable the USB ports so that no USB storage device can be plugged to your computer . But now a day's almost all peripheral devices uses the USB ports for communication.

So this cannot be the option to deal with the information insecurity. So in this paper a new way has been described which can be use to make our data more secure on our systems. This method has been experimented and the success rate is 100% with the 0% performance degradation. This idea has been implemented on the Linux platform which are

Debian based distros and the kernel version 3.9 onwards. The idea is like only to disable the USB mass storage devices by doing some

simple steps. As in Linux, only the root user has all type of authority so it can decide which user should use the USB ports for storage devices or which users should not have the privileges to plugged their mass storage devices and use it. This method can definitely improve the security of the information without losing the data and without corruption of data by any unauthorized users or by some harmful viruses.

II. LITERATURE SURVEY

As the part of academics, we take practical examination of students regularly every semester. While conduction of these examinations, we have observed that some students have tendency to copy the program from usb pen drives or through any other usb storage device. We searched the techniques to find the solution on it. We got several techniques to access and use the usb ports.

These are summarized below.

The general technique to access the usb storage device [4] is with the help of commands. The command of Linux that is, usb-devices gives the listing of everything about the usb such as,

T: Bus=01 Lev=00 Prnt=00 Port=00 Cnt=00 Dev#= 1 Spd=480 MxCh= 4

D: Ver= 2.00 Cls=09(hub) Sub=00 Prot=00 MxPS=64 #Cfgs= 1

P: Vendor=1d6b ProdID=0002 Rev=03.11

S: Manufacturer=Linux 3.11.0-19-generic ehci_hcd

S: Product=EHCI Host Controller

S: SerialNumber=0000:00:1a.7

C: #Ifs= 1 Cfg#= 1 Atr=e0 MxPwr=0mA

I: If#= 0 Alt= 0 #EPs= 1 Cls=09(hub) Sub=00 Prot=00 Driver=hub

Such kind of information is made available using the command but its is difficult and almost not possible to know about usb storage devices connected to the system.

III. ARCHITECTURE

The method of disabling the USB ports for mass storage devices consists of set of Linux kernel activities that has to be included in some of the configuration files and your USB storage devices will not be detected. This method can be done only through the root account or with the help of root permission.

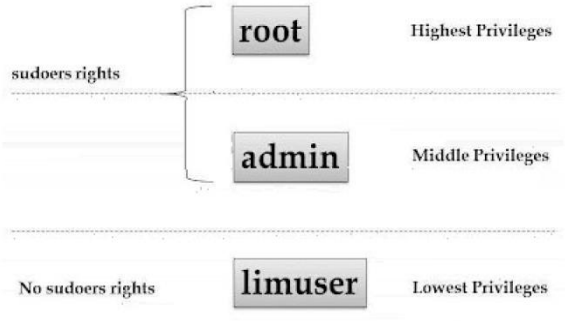

Fig.1 Account types in Linux.

The root in Linux has highest privilege rights. Administrator has middle privileges. It means in order to do any administrative activity it need the password. This is referred as sudoer rights.

Sudo implies- Super User Do. The limited account user 'limuser' is not having rights to do any administrative activity in the system. Step-1:

blacklist usb_storage

in blacklist.conf file. This configuration file reside in the /modprobe.d directory which in turn reside in the /etc directory.

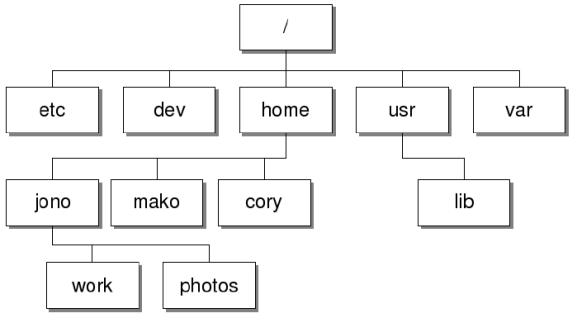

Fig.2 The Linux directory structure.

The /etc means the etc directory resides under root i.e. / represents the root. Almost all the system related configuration files are present in the /etc directory or in its sub-directories. These configuration files controls the operation of program. The modprobe.d indicates that modprobe is a directory. The modprobe is a program which adds or removes more than one module, because one module may have many dependencies, so this program is required for adding or removing multiple modules. This modprobe.d contains many .conf extension files which specifies many options that are required to be used with the modules that are to added or removed from thekernel. The format of the .conf file underneath modprobe.d directory is very simple. It just includes one line command to configure a program and some comments regarding the command which starts from the "# ". The modprobe.d directory include a file, name blacklist.conf file which specifies the modules that has to be ignored from

loading. This file helps to stop performing any operation that you don't want by not loading the modules. In blacklist.conf file, if you want to disable some operation, the keyword blacklist is used. This keyword blacklist means

all of a particular module specified is ignored. In this file if you want to disable the operation, you have to use the blacklist keyword followed by the name of the module. As in this paper we are concentrating in disabling the USB storage devices so the above command i.e. blacklist keyword followed by the module name usb_storage is used. This will prevent the modprobe program for loading the USB storage driver upon demand. The usb_storage is a module related to the USB storage devices. The device drivers is the bridge between the user space applications and the hardware space. Linux kernel constantly scans all your computer buses for any changes and new hardware. Whenever the device is attached, the hardware detection is done by the USB host controller. The device signals the motherboard and the USB chip controller gets the message and says the information to the kernel with the help of an interrupt. The kernel then re-initialise the USB bus and says it to the udev that some new device is attached. The device can be detected by their identity as like all the devices have a vendor name and a model id. Then the kernel uses the modprobe program to load the driver and says the udev that there is a device of so and so vendor and model number and then the udev tries to mount the device. As discussed that the kernel invokes the modprobe program to load the drivers and modules, the modprobe program will search the configuration file whether the driver is listed or not and when it is found that the module is listed as a blacklist then the modprobe fails to load the module and the kernel could not send any information about the device and the udev could not mount the device on your system.

Step-2:

modprobe -r usb_storage

in the rc.local file which is under the /etc directory. The /etc/rc.local file is common to all major distributions. This file is empty on fresh installation and it is reserved for local administrators.

The rc.local is a script file which contains any specific commands for the system that runs whenever the system startup. This file runs at the end of the system boot process, so the commands that we want to run at the time of system startup can be written in this file. The rc denotes the runcom or run command. This file can be helpful to write the commands that you want to execute at every boot time. In this file as the above command modprobe is used which helps to add or remove the modules and the dependent modules also. The kernel also depends on modprobe program to load or unload a module. There are many options that can be used with modprobe command like -a, -i etc, so the option **-r** is used in the above command which is used to remove the modules. This option is used to remove the modules and also try to remove the unused modules which are dependent on it. As whenever the storage devices are attach, the usb_storage module gets loaded which is a module related to mass storage devices and that module is used by those devices. You can see that the usb_storage module has been loaded by using the lsmod command and how many mass storage devices are attached and has used that module. This lsmod command shows the contents of the modules file under the /proc directory and the contents are the loadable kernel modules that are currently loaded on your system. So to remove that usb_storage module the modprobe program is used with -r option. To remove or unload any module you can use the modprobe program with -r option followed by the module name like usb_storage. When the system boots, at the end of all the initialization done, the rc.local file under the /etc directory gets executed and the modprobe command gets call and the usb_storage module gets unloaded. And if we just unload the usb_storage module without including the usb_stoarge module in the blacklist.conf file then the mass storage devices get mount whenever attach because it reloads the module that had been already unloaded. So to make it persistent, the usb_storage module must beinclude in the blacklist.conf file, thus restricting the module to get reload as the usb_storage module is blacklisted. Now if we want to enable the USB mass storage devices, i.e. the storage devices to get mount we have to just remove the commands that we have added in the blacklist.conf file under the /etc/modprobe.d directory and the rc.local file under the /etc file. As the commands are removed and we attached the storage device to the system the usb_storage module get reloaded and the USB storage device get mount and can be used as normally we do use.

CONCLUSION

In this paper, we have proposed a system which will control the USB mass storage devices according to the administrator or root authority. The administrator can decide whether to enable or to disable the USB storage devices of the system. As a result, after disabling the USB storage devices, those devices cannot get detected and the malicious activity or malwares through these storage devices can be fully controlled by the administrator or root authority by preventing the system or confidential information to get leak or corrupt by some unauthorized users.

REFERENCES

[1] Disabling USB storage drives, March 2008, National Security Agency, USA department of defense.
[2] Defense against Malware on Removable Media, National Security Agency, USA department of defense.
[3] USB Debugging and Profiling Techniques Kishon Vijay, Abraham I, and Basak Partha, Texas Instruments, Published on http://elinux.org
[4] Robert Love; "Linux Kernel Development", 3rd Edition.
[5] Manual pages of Linux/Unix security commands..