Administrator's Manual

USBSecurity version 2.0.5

Control of USB ports is a recurring need when computers are made available to users. Blocking the ports is not feasible when you want to set access permissions to them. There are various locking programs, perhaps also access control programs.

The idea is a system to control access to USB ports, which is manageable, runs from a centralized server and is cross-platform. USBSecurity is a set of programs to manage access control at user, device and computer level. It is cross-platform, works on Linux and Windows and is independent of processor architecture. By default it works in local mode but with a few configurations it can work from a centralized server.

USBSecurity

It is a set of programs to manage access control to USB ports. It consists of the following programs:

- usbsecurity-server
- usbsecurity-monitor
- usbsecurity-gui

usbsecurity-server

This is the web program for managing port permissions. By default it works in local mode at the address 127.0.0.1 behind port 8888. For more information run it from a terminal:

$ usbsecurity-server –h | --help

usbsecurity-monitor

This is the program that monitors the USB ports. By default it works in local mode blocking the ports as initial policy. By default the permissions are managed by usbsecurity-server but it can be configured to use a third party through a REST API.

For more information run it from a terminal:

$ usbsecurity-monitor –h | --help

usbsecurity-gui

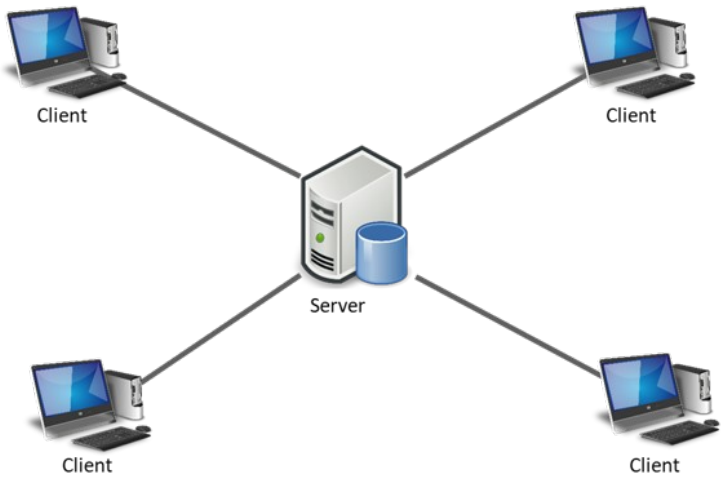
It is the program that offers a graphical interface. By default it works in local mode connected to the address 127.0.0.1 port 8888. For more information run it from a terminal:

$ usbsecurity-gui –h | --help

Centralized server

USBSecurity works as a client-server architecture. In this way each program is an independent process and communicates with each other through the network layer.

Based on this native behavior, it is possible to serve USBSecurity from a centralized server.



Configuration

Assuming that the server runs on the network with the following characteristics:

HOST: 192.168.1.1

PORT: 8888

Server

usbsecurity-server –host 192.168.1.1 –port 8888

Client

usbsecurity-monitor –host 192.168.1.1 –port 8888

usbsecurity-gui –host 192.168.1.1 –port 8888

Third party permits

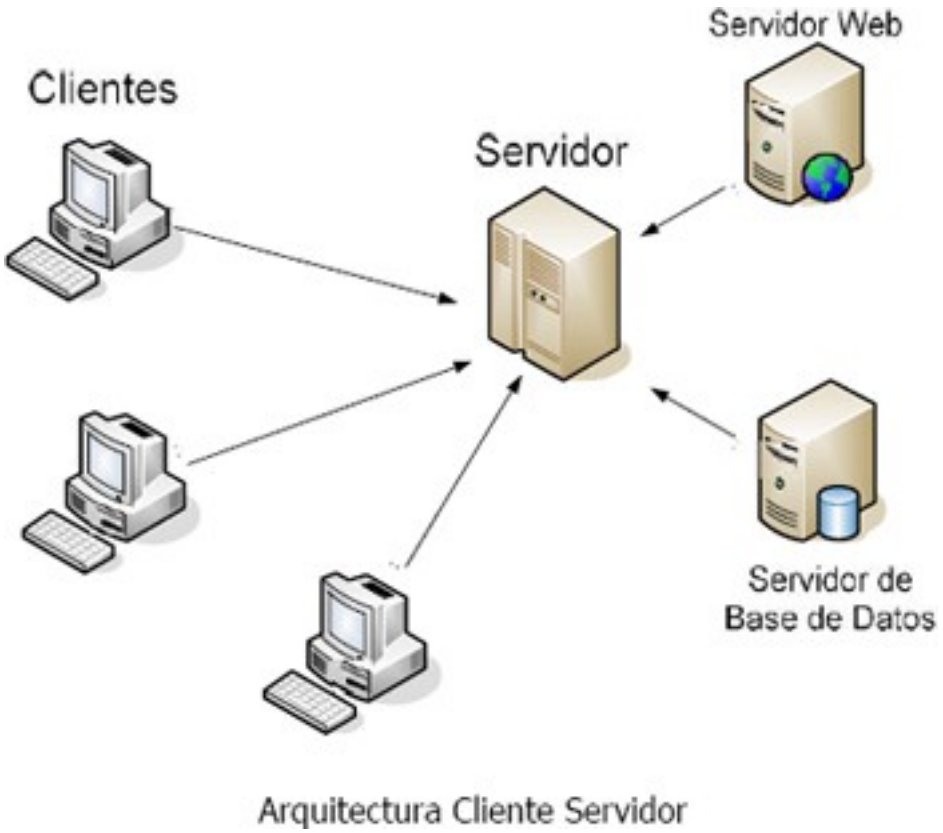
USBSecurity allows to manage permissions from a third party. The way to communicate is through a REST API. The usbsecurity-monitor application has a -url-api parameter for this purpose that should expect two parameters: 'action' and 'id'.

Param ‘action’

This parameter is used to pass the event that occurred on the USB port.

Param ‘id’

In this parameter the id of the device that triggered the action is passed.



Arquitectura Cliente Servidor

Configuration

Assuming that the server runs on the network with the following characteristics:

HOST: 192.168.1.1

PORT: 8888

and the third party server api manages the permissions through the following URL:

URL: http://192.168.1.2:8889/api/action/{action}/{id}

Client

usbsecurity-monitor –host 192.168.1.1 –port 8888 –url-api http://192.168.1.2:8889/api/action/{action}/{id}

* You need the parameters in the URL with the braces: {parameter}.

Administration

By default USBSecurity is administered locally from the browser and the URL http://127.0.0.1:888/admin. With a centralized server you access from http[*s*]://*ip_server*:*port*/admin. From usbsecurity-gui you can access directly.

Default credentials

User: admin

Password: usbsecurity

* Since the manual is public and shows the default credentials, it is strongly **recommended to change the password the first time** you log in.

Management options

The administration interface is intuitive and flexible. You can make the following configurations:

- Create user accounts, collective or personal.
- Create permissions by accounts.
- Crear permisos por dispositivos.
- Create permissions per computer.
- Create permissions by accounts and devices.
- Create permissions per computer and devices.
- View and edit user sessions on computers.