1. Fing CLI vs. FingKit
   - What is the difference between Fing CLI and Fing Kit?
   - What are the Fing Kit requirements?
   - Can I use Fing Kit using the Fing CLI?

2. Requirements
   - What are the system requirements?
   - What are the user requirements?
     - Windows
     - OS X
     - Linux - OpenWRT
     - Docker

3. Installation
   - How can understand the architecture of my system?
   - What is the installation procedure?
     - Windows
     - OSx
     - Linux
     - OpenWrt
     - Docker
   - Which is the proper command to invoke the Fing CLI?
     - Windows
     - OS X - Linux - OpenWRT
     - Docker

4. Uninstallation
   - What are the steps of the uninstallation procedures?
     - Windows
     - Linux
5. Configuration Files

Which is the format of the configuration files?
Where are configuration files stored?

Windows
OS X - Linux - OpenWRT - Docker

Which customization can I do with the configuration files?

6. Tools

How to get the network information on the host?
How to scan a network?
How to discover services?
How to measure network round-trip time?
How to trace the route of packets to destination host?
How to wake device on LAN or WAN?
How to launch the Fing Kit using the Fing CLI?

7. Fing CLI / Kit as a Service

What does it mean running Fing CLI as a Service?
What are the system requirements?
Is it possible to run the kit as a service?
How can I configure my system to run the Fing Kit as a Service?

Windows
OS X
Linux - OpenWRT
Docker

Can I run other Fing CLI command as service?
8. Interactive mode

How can I start fing in interactive mode?

Windows  
OS X - Linux - OpenWRT  
Docker

What tools are available?

9. Other

How to extract data from CSV files?
1. Fing CLI vs. FingKit

What is the difference between Fing CLI and Fing Kit?

The **Fing Command Line Interface (Fing CLI)** is stand-alone executable delivering a suite of network tools for troubleshooting (NIC information, scan, ping, traceroute and so on).

The **Fing Kit for Device Recognition (FingKit)** is a software designed to provide device scanning and device recognition technology. It consists of two components:

1. The local scanner, which scans the network for devices
2. The device recognition API, providing the recognition technology

The FingKit is delivered as a library for mobile (*iOS, Android*), desktop (*Windows, Linux, OSx*) and embedded (*OpenWRT*) platforms.

What are the Fing Kit requirements?

The FingKit system and user requirements coincides with those of the **Fing CLI**.

Moreover, the FingKit requires a license to gain the access to our Device Recognition API. Write to our sales team on sales@fing.com to get pricing information.

Can I use Fing Kit using the Fing CLI?

Yes, the CLI contains also the FingKit functionality as a specific option that generates text outputs that can be integrated easily with other software. See section 6 and 7 for further information.
2. Requirements

Fing CLI is available desktop (Windows, Linux, OSx) and embedded (OpenWRT) platforms.

It also can be installed on a Raspberry PI - mounting a standard Raspbian - and on any Linux system with Docker containerization system.

The installable files may be downloaded here.

What are the system requirements?

Fing runs on the following Operating Systems:

- Windows Vista, Windows 7, Windows 8.x, Windows 10
- OS X Lion (10.7), OS X Mountain Lion (10.8), OS X Maverick (10.9), OS X Yosemite (10.10), OS X El Capitan (10.11), OS X Sierra (10.12), OS X High Sierra (10.13)
- Debian >=5.0, Raspbian, Ubuntu >=8.x, CentOS >=5.0, Fedora >=10, Slackware, Gentoo
- OpenWRT Chaos Calmer / LeDe on standard target/subtarget architectures
- Docker running on Linux Systems

What are the user requirements?

You don’t need to be a Network Administrator nor a Domain Administrator to run Fing.

An administrative user account is required to install and run the software; this is a security restriction enforced on every operating system to make sure the network is appropriately protected.

If you are unsure about the type of user account you are using on a machine, please refer to the list below.

Windows

On Windows, you must be logged as a User having Local Administrator privileges. If you are unsure of your user type, click on::

Start | Settings | Control Panel | User Accounts

---

1 Packages are available for 32 and 64 bit architectures.
2 We need the docker to run with host networking. As per Docker documentation: “The host networking driver only works on Linux hosts, and is not supported on Docker Desktop for Mac, Docker Desktop for Windows, or Docker EE for Windows Server”.
A dialog will pop-up and you shall see the account type (Standard or Administrator).

You shall be listed as 'Administrator'.

To run the software, a Command-line tool cmd.exe shall be executed as a User Administrator.

**OS X**
On OS X, you must be logged as a User having Administrator privileges.

If you are unsure of your user type, click on your User Name in the menu-bar and select "Account Preferences...". A dialog will prompt a list of all users and types (Standard, Admin, System).

**You shall be listed as either 'Admin' or 'System'**.

**Linux - OpenWRT**
You must be logged as the root user. Other equally-powerful accounts and the installation through sudo command are correct, too.

N.B. Find more about privileged access [here](#).

**Docker**
You already have root permission inside the docker. You must have the permission to start docker instances on the host system.
3. Installation

The installation is straightforward on all platforms: download the package and install it. Below some specification on how to install on the different operating system.

How can understand the architecture of my system?

Windows and OSx are single architecture operating system: there is only one installer for them.

For Linux, you can use the following command:

```
uname -m
```

The output should be something like this:

```
x86_64
```

On OpenWRT, you should find all the information on files:

- /etc/openwrt_version
- /etc/openwrt_release

What is the installation procedure?

Windows

Double click on the .exe file to trigger the installer and follow the mandatory steps.

If you do not have WinPcap packet capture library in your system, you will be also prompted to install it. When asked to enable the WinPcap 'NPF' Windows service, you can decide whether:

- to install and start the NPF service, thus allowing fing (and other packet capture tools) to be used by any non-admin user in the system
- not to install the NFP service, so that only admin user accounts will be allowed to use fing

OSx

Download and open the .pkg file, and follow the instructions provided by the installer.
Linux
The software has been packed using Debian (DPKG) and Read Hat (RPM) package manager.

It’s also available as tarball (TGZ) for Linux distros exploiting different package management systems.

Architecture supported:

- \textit{i686} and \textit{amd64} (intel 32/64 bit)
- \textit{arm} and \textit{arm64} (arm soft float 32/64 bit)
- \textit{armhf} (arm hard float 32 bit)
- \textit{mips} and \textit{mips64} (mips big endian 32/64 bit)
- \textit{mipsel} and \textit{mipsel64} (mips little endian 32/64 bit)

Download the package most suitable for your distribution and open a Terminal window.

You may refer to this \textbf{List of Linux Distributions} to find out the package format your platform requires:

- Debian / Ubuntu / ...
  
  \texttt{sudo dpkg -i fing-<version>-<architecture>.deb}

- Fedora / Gentoo / ...
  
  \texttt{sudo rpm -i fing-<version>-<architecture>.rpm}

- Generic
  
  \texttt{sudo tar -zxvf fing-<version>-<architecture>.tar.gz \ --strip-components 1 -C /}

OpenWrt
Given the number of target and sub-targets of OpenWRT operating system. The package can be installed using:

\texttt{sudo opkg install fing-target-subtarget.ipk}

Docker
Uncompress the tarball using the command
Before all, you have to create locally the fing image using the following command:

```
docker load -i fing-<version>-docker_image.tar
```

### Which is the proper command to invoke the Fing CLI?

The command 'fing' without any additional parameter executes a discovery on your Local Area Network with a plain text output sent to console. This is only the simplest of reports Fing may generate.

#### Windows

As soon as installation is complete, you can open a command prompt and launch the software:

```
fing
```

#### OS X - Linux - OpenWRT

After the installation has successfully completed, open a Terminal and type:

```
sudo fing
```

#### Docker

You can run the container with the following command\(^3\):

```
docker run --tty --interactive --network host \n  registry.fing.io/docker/kit:<version> \n  /usr/bin/fing
```

\(^3\) Your users must to belong to docker group (i.e. it needs the rights to run docker command)  
\(^4\) We need the docker to run with host networking to gain permission to scan the user network.
4. Uninstallation

What are the steps of the uninstallation procedures?

Below the procedure for each architecture⁵.

Windows
To uninstall Fing on Windows platform, use the Start menu to locate the Fing folder. An Uninstall menu item is available; selecting it will run the uninstaller application that will remove the program from the system.

Linux
If you have installed fing using dpkg, type:

```bash
sudo dpkg -r fing
```

If you have installed fing using rpm, type:

```bash
sudo rpm -e fing
```

Otherwise, type:

```bash
sudo /usr/local/bin/fing-uninstall.sh
```

OS X
Open a Terminal window and execute:

```bash
sudo /usr/local/bin/fing-uninstall.sh
```

OpenWRT
Open a shell and type:

```bash
sudo opkg remove fing
```

---

⁵ Before you uninstall a kit running as a service, make sure you have shut down any other service that loads the kit at the machine's start-up. See Fing Kit as a Service for further details.
Docker

You have to be sure that there are not running instances of fing image:

```
docker rm -f $(docker ps -a | grep registry.fing.io/docker/kit | awk '{print $1}')
```
	hen type:

```
docker rmi -f registry.fing.io/docker/kit:<version>
```

Please refer to Docker official documentation for any deepening.
5. Configuration Files

Which is the format of the configuration files?
Fing configuration files are all in the same format called “properties”. Further information [here](#).

Where are configuration files stored?

**Windows**
The application data folder is:

```
%APPDATA%\Fing\conf
```

You also have a shortcut to that folder using:

```
Start | Fing | Fing Configuration
```

**OS X - Linux - OpenWRT - Docker**
Configuration files are placed in:

```
/etc/fing
```

Which customization can I do with the configuration files?
There are a lot of possible customizations that may be performed using the Fing properties file.

As an example, to change the separator used in the Comma-Separated-List export, simply open your `fing.properties` configuration file and add/edit the setting below:

```
overlook.fing.formatter.csv.separator = ,
```
6. Tools

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Short Option</th>
<th>Long Option</th>
<th>Mandatory Argument(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Info</td>
<td>Show network information</td>
<td>-i</td>
<td>--info</td>
<td>-</td>
</tr>
<tr>
<td>Discovery</td>
<td>Run a network discovery</td>
<td>-n</td>
<td>--netdiscover</td>
<td>Network/Host List</td>
</tr>
<tr>
<td>Service Scan</td>
<td>Scan services on a host/network</td>
<td>-s</td>
<td>--servicescan</td>
<td>Network/Host List</td>
</tr>
<tr>
<td>Ping</td>
<td>Ping a list of host</td>
<td>-p</td>
<td>--ping</td>
<td>Hosts List</td>
</tr>
<tr>
<td>Traceroute</td>
<td>Trace the route to the target</td>
<td>-T</td>
<td>--traceroute</td>
<td>Hosts List</td>
</tr>
<tr>
<td>Wake On Lan</td>
<td>Wake a local device</td>
<td>-w</td>
<td>--wol</td>
<td>Hosts List</td>
</tr>
</tbody>
</table>

Since now afterwards, we refer as `<cmd>` to the command described in this section.

How to get the network information on the host?

The Fing Kit can provide detailed information about the status of your network card and other useful information on your local network:

- Local Network Interfaces:
  - Type (Ethernet, WiFi, etc.)
  - Hardware Address
  - IPv4
  - IPv6
- Default gateway
- DNS Server

```
<cmd>  --info
```

How to scan a network?

An administrative user account is required to run the software on all platforms. The network discovery provides a complete view of any network: fing engine automatically
detects the underlying network type and uses the best technique to take the picture of the target network.

The best results are achieved on Local Area Networks, both on wired Ethernet connection and wireless Wi-Fi connections, where Fing can make use of the dedicated data-link layer discovery engine which is faster one more accurate. Fing’s engine can detect all the hosts present in the network, even behind a Firewall!

Discoveries performed on non-local networks (or non-ethernet networks) are handled by a network layer discovery engine, which relies on TCP/IP network layer, i.e. ICMP (ping) and TCP queries. When you start a discovery fing tells you the actual engine which is being used; in case of specific needs, it is possible to configure and tune each engine for optimal results, creating dedicated discovery profiles in the related configuration properties file:

```
discovery.properties
```

When you start Fing without arguments, it takes the nearest of your available networks and starts performing a discovery on it, reporting stuff directly on console. But you can perform discovery on any network, by providing your target network to fing in the command line:

```
<cmd> -n 192.168.1.0/24
<cmd> -n www.domotz.com/24
```

If you do not provide any output parameter, fing uses default ones (as specified in its fing.properties configuration file). But for a running discovery you can setup as many output formats as you need, by providing a command line argument like:

```
<cmd> -o setupFormat1 setupFormat2 ... setupFormatN
```

The output format setup syntax is pretty simple: there are 2 main categories of output flows, table and log. The table flow produces a network table dump every time a discovery round is completed, while the log flow logs each network event as soon as it’s detected.

If you want to change the discovery round frequency, edit the related round.interval setting in discovery.properties configuration file. Note that for each profile you declare you must provide both configurations for data-link and network layer discovery classes.
The log flow allows you to log network events in real-time, on the console itself or in a specific file. Currently there are two formats supported for log flows: text and CSV.

E.g. to start fing producing textual log in console and a CSV log in a file:

```
<cmd> -o text,console csv,ling.txt
```

The table flow instead produces a network table view refreshed each time a discovery round finishes. The most popular formats are text and HTML but here it is the complete list: stext (short text for 80-columns console), text (plain text), html, csv, and xml.

E.g. on Windows to make fing report network discovery to an HTML file on your:

```
<cmd> -o table,text table,html,"%USERPROFILE%\Desktop\network.html"
```

The network table report contains details for each host found in the network, and it's refreshed in real-time at each round: IP address, MAC address, hostname and host friendly name. The latter is a friendly name you can associate to the hosts by means of the hosts.properties configuration file, where you are able to define your custom names for hosts (by IP address or MAC address) and for networks.

By default, when you close fing the discovery session is lost, unless you want to save session data into a session file; in this case fing can be closed and restarted when you need, without losing any discovery session data. To exploit discovery session feature you have to simply provide fing the session file name to use.

E.g. on Windows to make fing generate an HTML report and save session data in a folder named report:

```
<cmd> -n 192.168.1.0/24 -o table,text,C:\report\lan.html \ --session C:\report\lansessiondata
```

**How to discover services?**

The service discovery feature, also known as service scan, quickly detects active TCP services on a target host or network. Service discovery also gives its best with ethernet-based networks, where TCP SYN scan technique can be applied to audit active services on any host in a few seconds.

You can scan a local or remote host but also entire networks. You may specify a maximum number of ports for the scan to make it faster:

```
<cmd> -s 192.168.1.1 # (single host)
<cmd> -s 192.168.1.1 -m 1000 # (single host with max ports)
```
By default discovered services are reported on console as a plain text output, but you can choose between different output formats: like text, CSV, XML and HTML. So it is actually possible not only to use it as a command line administrative tool, but also integrated with your 3rd party applications:

```
<cmd> -s www.fing.com       # (domain)
<cmd> -s www.google.com/24  # (entire network)
```

How to measure network round-trip time?

The **ping feature** is capable of performing multiple quick pings to a list of target hosts. You can scan a set of local or remote hosts.

```
<cmd> -s 192.168.1.1 -o html,report.html
<cmd> -s www.fing.com -o xml,scan.xml
```

By default ping summary is reported on console as a plain text output, but you can choose between different output formats: like text, CSV, XML and HTML. So it is actually possible not only to use it as a command line administrative tool, but also integrated with your 3rd party applications.

The output format setup syntax is pretty simple: there are 2 main categories of output flows, table and log. The table flow produces a network table dump every time a discovery round is completed, while the log flow logs each network event as soon as it's detected. The log flow can be removed using the option **--silent**:

```
<cmd> -p 192.168.1.1 www.domotz.com
```

How to trace the route of packets to destination host?

The **traceroute feature** is capable of for displaying the route (path) and measuring transit delays of packets sent to a target host. You can scan a set of local or remote hosts.

```
<cmd> -T 8.8.8.8
```
The summary is reported only on console as a plain text output.

+--------------------------------------------------------------+
|               === Fing 5.4.0 - www.fing.io ===               |
+--------------------------------------------------------------+
| #hop |    host      |   avg    |   loss   |   min   |  max   |
|------+--------------+----------+----------+---------+--------+
|  1   | 192.168.7.1  |  1.831   |    -     |  1.371  |  2.474 |
|  2   | 10.0.1.1     |  2.223   |    -     |  2.007  |  2.514 |
| 11   | 8.8.8.8      |  30.49   |    -     |  30.22  |  30.92 |

How to wake device on LAN or WAN?

Wake-on-LAN (WOL) is an Ethernet computer networking standard that allows a computer to be turned on - or woken up - by a network message.

WOL is implemented at hardware level by the motherboard of a computer (BIOS) and the network interface (firmware). Consequently, it is independent of the Operating System (and NIC drivers) running on the hardware, so it works on Windows, OS X and Linux machines. Some operating systems can control Wake-on-LAN behaviour via hardware drivers. The WOL message must be sent on the local network. To wake up machines from an external network (Wake-on-WAN), WOL Gateways must be configured to receive WOL packets from outside networks, and forward them to the local network.

Fing is able to send WOL magic packets to wake local or remote computers that have the WOL feature enabled. The feature works with devices that are in your connected local network, but you may also send remote WOL signals outside your local network to hosts/routers configured to be WOL gateway services by specifying the host address and port.

The syntax is very simple:

```
<cmd> --wol
```

Each target must have the following syntax:

```
MAC[@network[:port]]
```

or:
Examples:

- On docker, to send WOL to MAC 010203040506 in current LAN:
  
  ```
  docker run --network host registry.fing.io/docker/kit --wol 010203040506
  ```

- On OS X, to send WOL to MAC 010203040506 and 112233445566 in current LAN, silently:
  
  ```
  sudo fing --wol 010203040506 112233445566 -silent
  ```

- On Raspberry Pi, to send WOL to MAC 01:02:03:04:05:06 in network 192.168.0.1/24:
  
  ```
  sudo fing --wol 01:02:03:04:05:06@192.168.0.1/24
  ```

- On Linux, to send a remote WOL for MAC 01:02:03:04:05:06 to the host myremoterouter.com configured on UDP port number 9:
  
  ```
  sudo fing --wol 010203040506@myremoterouter.com:9
  ```

**How to launch the Fing Kit using the Fing CLI?**

The command to launch the **FingKit** using the **FingCLI** is (as administrative user!):

```bash
<cmd> --kit <kit-configuration-file-path>
```

Since now afterwards we refer to the Fing CLI assuming that FingKit is just an option of the tool.
7. Fing CLI / Kit as a Service

What does it mean running Fing CLI as a Service?

A **service** is a computer program that runs as a background process, rather than being under the direct control of an interactive user. Fing CLI can run continuously in background and write the output – results and logs – in the respective folder.

What are the system requirements?

Every Operating System has its own service management framework as part of the core features. You may need to install some (lightweight) tools to run Fing CLI as a service.

Is it possible to run the kit as a service?

Yes, this section has been designed for that use case indeed.

How can I configure my system to run the Fing Kit as a Service?

Fing Kit requires a configuration file at this location:

- **Windows:** `"%APPDATA%\Fing\conf\kit.properties"
- **Linux/Unix/OSx:** `"/etc/fing/kit.properties"

The file is created after the first execution of fing:

```
enrichment.enabled = true
license = 08d26f6b3113ca9733ab7fee378257087d01796656c6889f2e0e40
refresh.interval = 3600000
rounds = 0
output.folder = /var/data/fing/kit
```

After that you can setup your system.

**Windows**

The command-line interface of Fing is a simple but powerful tool to integrate Fing in your daily work. It's easy to perform network discoveries in the background, or generate XML reports at regular intervals. If executed as a Windows Service, Fing automatically starts when you power on your computer, and automatically stops when you shut it down. As all Windows services, Fing Services can be managed from the **Windows Service Control Panel**.

6 You can ask for your own license ID by sending an email to sales@fing.com
You must use absolute paths for every file you provide in the command line (both output files and session file). Fing Service will run as LocalSystem user and will use the Home Folders of the user that registers the service to read and store configuration files, log files and any other accessory file.

To manually install it:

```
fing --installservice FingKit --kit "%APPDATA%\Fing\conf\kit.properties"
```

The corresponding Fing network discovery with device recognition is installed as service and immediately started. Later, if you want to later uninstall the service, just type:

```
fing --uninstallservice FingKit
```

Note that, at the moment, Fing service removal is not supported when completely uninstalling Fing. Hence, if you are in the process of upgrading or removing Fing software from your computer, you should stop and uninstall running Fing services as an initial step.

OS X
Copy or link the launched script from the installation folder to OSx users’ daemon folder:

```
sudo cp /usr/local/lib/fing/launchd/com.fing.fingkit.FingKit.plist /Library/LaunchDaemons
```

To manually load and start the service in background, type:

```
sudo launchctl load /Library/LaunchDaemons/com.fing.fingkit.FingKit.plist
```

To unload:

```
sudo launchctl unload /Library/LaunchDaemons/com.fing.fingkit.FingKit.plist
```

Linux - OpenWRT

The widest, simplest and most stable daemon manager for GNU/Linux operating systems is System V. The new systemd of Red-Hat family and systemctl of Ubuntu/Debian family are the replacement for SysVinit and Upstart. Because of they are compatible with

7 It’s included also the Debian distro for Raspberry PI (Raspbian)
SysVinit, and lots of embedded device runs Operating System (like OpenWRT) that still work with SysVinit, we do provide the instruction for that daemon manager.

Copy or link the init.d script from to System V folder:

```
sudo ln -s /usr/local/lib/fing/init.d/fing-kit /etc/init.d/
```

Then add to your services with:

```
sudo chkconfig --add fing-kit
```

or:

```
sudo update-rc.d fing-kit defaults
```

To manually control the service:

- start:

  ```
sudo service fing-kit start
  ```

- stop:

  ```
sudo service fing-kit stop
  ```

Docker

Before starting the docker, you have to prepare the following directory tree:

```
.
├── data/
│   ├── kit.cfg
└── log/
```

---

8 Default on Red-Hat family. The tool can be installed through the command: `sudo apt-get install chkconfig`

9 Default on Debian family. The tool can be installed through the command: `sudo apt-get install sysv-rc`
where data/ and log/ are empty folder and kit.cfg is the configuration file of kit tool as described here.

From the root directory of the tree, tap:

```
docker run --network host --detach \
    --mount type=bind,source=$(pwd)/kit.cfg,target=/var/fing/kit.cfg \
    --mount type=bind,source=$(pwd)/log,target=/var/log \
    --mount type=bind,source=$(pwd)/data,target=/var/data \
    registry.fing.io/docker/kit:<version>
```

N.B. The mount options is required to have the log and the output folder on your system.

Can I run other Fing CLI command as service?

No
8. Interactive mode

The interactive mode will guide you through Fing tools with a step-by-step procedure. Fing is a command-line tool, and you would need to provide a (possibly long) list of arguments to specify your settings. In the interactive mode, the tool itself will guide through the available features and configurations, from basic operations to complex discoveries, in just a few seconds. The feature configures Fing and starts your discovery and additionally displays the whole command-line arguments, according to your chosen options.

How can I start fing in interactive mode?

Windows
You can start interactive mode from fing shortcut in:

Start menu | Applications

Alternatively, you may execute a Command-Line shell using:

Start menu | Execute

and type:

``cmd``

Then, a Shell window will open, allowing you to enter the command:

``fing --interactive``

OS X - Linux - OpenWRT

You shall open a Terminal window and execute:

``sudo fing --interactive``
Docker

You shall open a shell on the host system and execute:

```bash
docker run -it registry.fing.io/docker/kit:5.4.0 \
/usr/bin/fing --interactive
```

What tools are available?

Once started, the interactive mode will provide the following options:

- (D)iscover
- (S)can
- (P)ing
- (T)raceroute
- display (I)nfos

Each letter will start the corresponding Fing tool, guiding you through a series of question regarding the networks to scan, the output formats, the output destinations and such. At the end of each tool, the interactive mode will print the full set of parameters - should you need to run the same command directly in the future - and will ask you if you wish to execute it now.

```text
=== You have completed the procedure ===
The equivalent command is:
fing -p localhost -o csv,console
```
9. Other

How to extract data from CSV files?

Fing for Console may output the content of the discovery in a classical Comma-Separated Output, which actually uses a semicolon to separate the content.

You may generate the CSV output using the appropriate output format, as in the example below:

```
<cmd> -r1 -n10.255.0.0/16 -o csv,fingscan.csv
```

The file contains the discovery data is in the following order:

1. IP address
2. Custom Name of the Node (if you have assigned custom names in the .properties file)
3. The state (UP/DOWN)
4. The timestamp of last change
5. The host name (if you have enabled DNS)
6. The Hardware address (if you are discovering a local network)
7. The Hardware vendor

You may eliminate one of the column with simple scripts. On Unix and Mac Os X, you may use an "awk oneliners", like:

```
awk -F ";" '{print $1, $2, $3, $5, $6, $7}' yourfile.csv
```

On Windows, you may use the PowerShell, like:

```
Get-Content yourfile.csv | %{ $_.Split(';')[1]; }
```

which extracts the first column. There is also the import-csv cmdlet which provides many features.