1 IP-Sniffer

1.1 Introduction

IP-Sniffer is a command line tool that captures the IP traffic for a given network adapter. The tool is available for Windows and Linux systems. By using of protocol-, IP-address- and/or IP-port-filters a very special IP-traffic can be captured. An integrated Interprocess-Communication (IPC) client allows forwarding the filtered IP-traffic to other applications which run on the same system, e.g. the ***decontev*** IP-Proxy tool. (btw. IP-Proxy can be used to then forward this data to remote sites.) Additionally it can be set whether only the payload data or the entire IP-Frame shall be forwarded. Several filters can be defined and activated at the same time and for each of these filters a separate IPC channel can be opened. For easier and fast definition of filters an own configuration tool is provided.

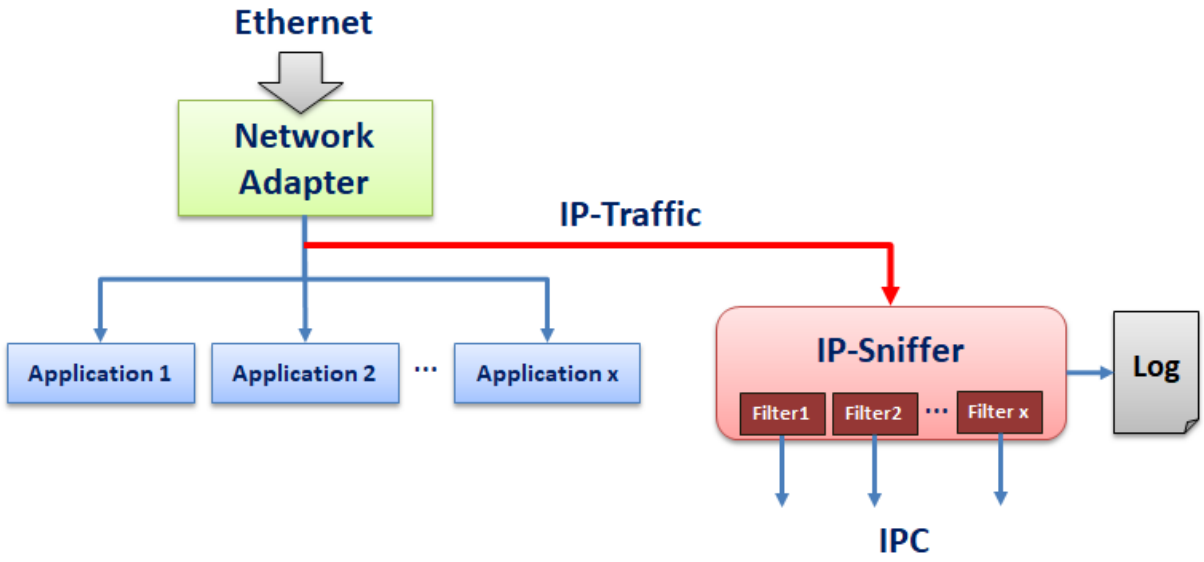

Figure 1: IP-Sniffer Overview

1.2 Usage

!!! Note !!!
IP-Sniffer requires administration privileges on Windows systems as well as root privileges on Linux systems.

1.2.1 Windows or Linux Desktop

IP-Sniffer can be called from a Command-Line Window (Windows) or from a Terminal Window (Linux) as following

```
//WINDOWS
IpSniff [parameters]

//Linux
sudo ipsniff [parameters]
```

mandatory Parameters	
-a <ip-address>	IP-address of the network adapter e.g. –a 192.166.29.43 *Note: this parameter is mandatory only for Windows systems, on Linux systems no network adapter has to be specified because all Ethernet traffic is analyzed and the appropriate IP-traffic is filtered by the tool*

optional Parameters	
-h	get help
-f <config-file>	load a configuration provided by a configuration-file
-p <protocol>	defines a protocol filter [0=all 1=ICMP 2=TCP 3=UDP]
-is <ip-addr>	defines a filter for a source IP address
-id <ip-addr>	defines a filter for a destination IP address
-ps <port>	defines a filter for a source port
-pd <port>	defines a filter for a destination port
-log	activates writing of log-data
-dump	output of data in a well-formed readable format
-ipc	activates sending of data via IPC (interprocess communication)
-fr	the entire IP-frame is sent via IPC, otherwise only payload data are sent

1.2.2 Linux without GUI

On Linux Server-sytems IP-Sniffer can be called as following:

```
sudo ipsniff [parameters] >[log-file] &
```

Parameters	
[parameters]	parameters as described in the chapter above
> [log-file]	the whole output will be redirected into a log-file
&	the application will be started as background process *Note: the application has to be terminated by killing the appropriate job, e.g.* ***kill %1*** *for killing job-number '1'*

1.2.3 Terminate IP-Sniffer

The application can be terminated by pressing the key combination **Ctrl+C**

1.3 Filter Configuration

The filter configuration of IP-Sniffer is done by means of a special configuration file. This file has to be formatted as the well-known Ini-File format, i.e.

```
[Section_1]
parameter_1 = value_1
parameter_2 = value_2
:
parameter_x = value_x

[Section_2]
:

[Section_x]
:
```

In following an example is shown:

```
[Log]
LogPath=C:\Users\ipuser\ipsniff\Log
LogLevel=3
DiskSize=1000
NoOfFiles=500
OldestFile=500
ZipArchive=1

[General]
NoOfFilters=2
AdapterIp=192.168.0.2
DumpFrame=1
Logging=1

[Filter_1]
Protocol=3
DestIp=192.168.0.48
SrcIp=0.0.0.0
DestPort=12345
SrcPort=0
IPC=1
IpFrame=0

[Filter_2]
Protocol=2
DestIp=192.168.0.2
SrcIp=192.168.0.83
DestPort=40001
SrcPort=53201
IPC=0
IpFrame=1

[Filter_3]
:
:
```

All parameters are described in following:

[Log]	
LogPath	path where the log-file shall be written *Note: the name of the log-file will be set automatically <inst>_<yyyy>_<mm>_<dd>.log*
LogLevel	log-level 1=error, 2=warning, 3=info, 4=debug
DiskSize	minimum of free disk space in MB *Note: the logging is stopped automatically if the value falls below this threshold*
NoOfFiles	maximum number of files in the log-directory *Note: the oldest files are deleted automatically if value exceeds this threshold*
OldestFile	oldest log-fil in days in the log-directory *Note: the file are deleted automatically if they exceeds this threshold*
ZipArchive	older log-files are compressed automatically 0=off, 1=on

[General]	
NoOfFilters	number of filter sections
AdapterIp	IP-address of the network adapter *Note: only necessary for Windows*
DumpFrame	0 = dump only hex-data 1 = additionally dump data in well-formed output format
Logging	0 = logging off 1 = logging on

[Filter_X]	
Protocol	protocol filter 0 = all 1 = ICMP 2 = TCP 3 = UDP
DestIp	destination IP-address filter 0.0.0.0 = no filter
SrcIp	source IP-address filter 0.0.0.0 = no filter
DestPort	destination port filter 0 = no filter
SrcPort	source port filter 0 = no filter
IPC	0 = IPC off 1 = IPC on *Note: the corresponding IPC-ID is generated automatically as following:* *decontev_sniffer_ipc_<instance_no>_<filter_no>* *e.g. decontev_sniffer_ipc_1_1*
IpFrame	0 = send only payload data via IPC 1 = send entire IP-frame via IPC

1.4 Setup-Tool

For easier configuration of filters the IpSniff-Setup tool can be used. The tool is installed in the same directory of IP-Sniffer. The setup-tool is also available for Windows and Linux.

Start the tool by calling the appropriate application file *'IpSniff_Setup'* from the installation directory. The use of the tool is largely self-explanatory.

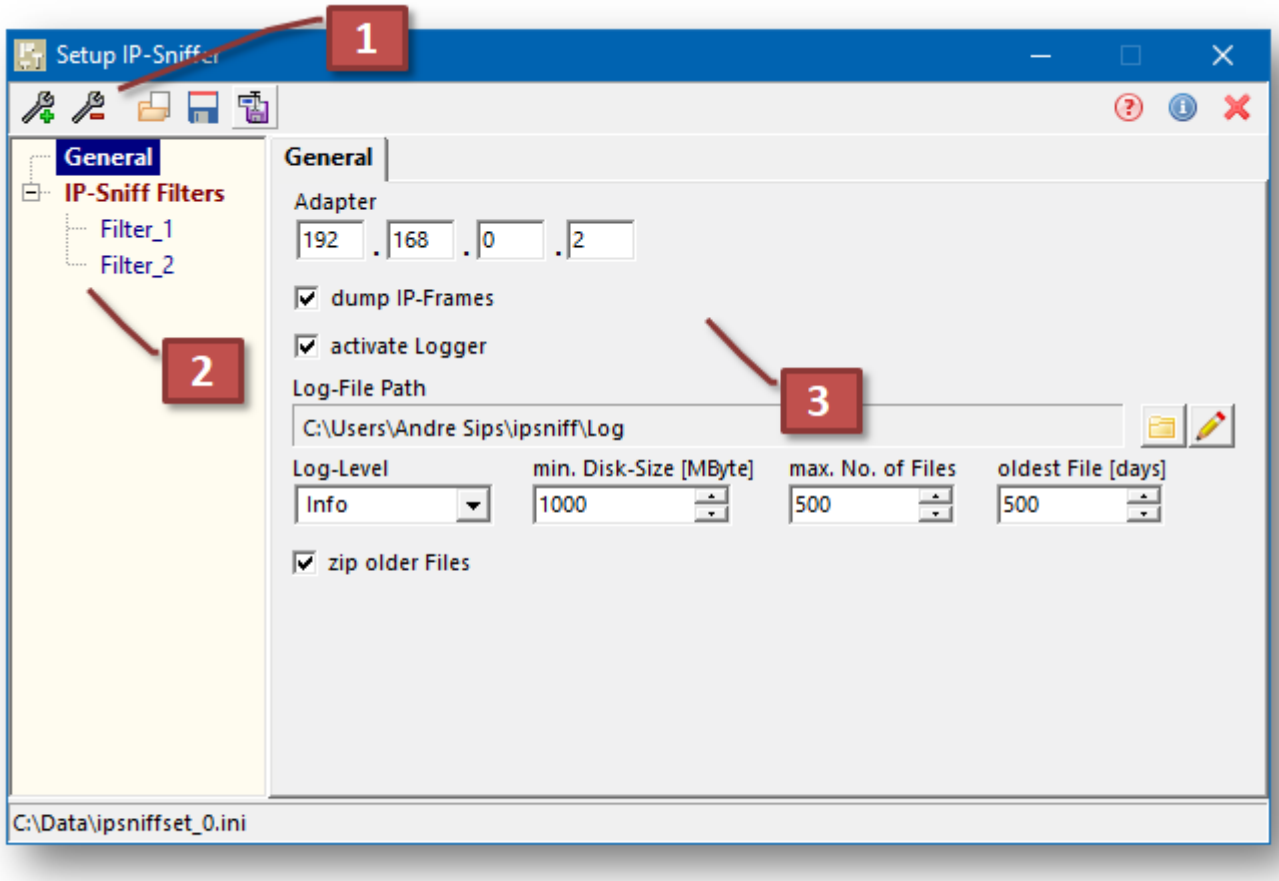

Figure 2: IP-Sniffer Setup-Tool

[1] tool-bar
[2] tree-view of settings
[3] panel for specific settings for the selected item in the tree-view, i.e. general settings or filter settings

Tool-Bar

Figure 3: Tool-Bar

[1] add filter
[2] delete selected filter
[3] load configuration file
[4] save current configuration file
[5] save configuration file under new filename

1.4.1 General Configuration

Select 'General' in the tree-view to setup general configurations.

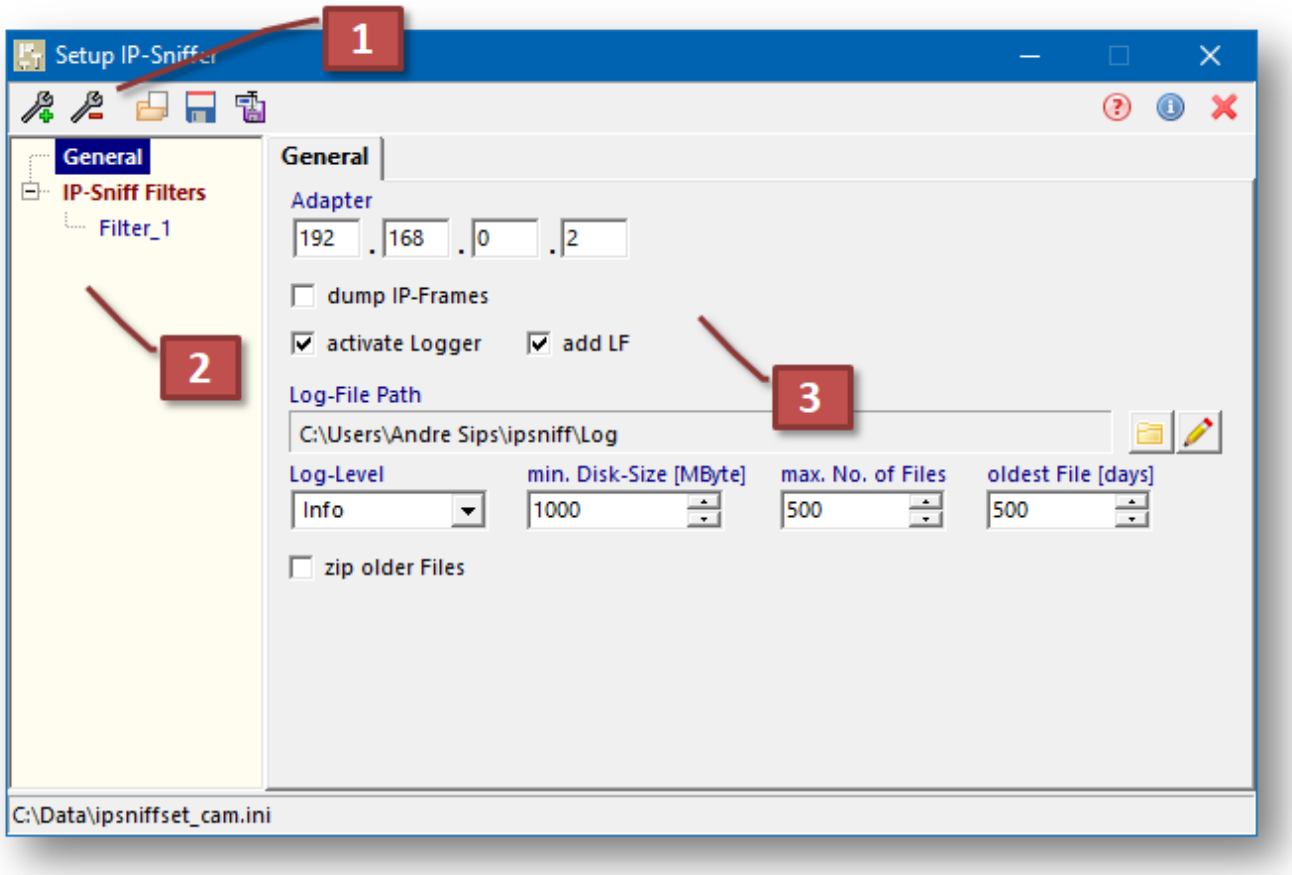

Figure 3: IP-Sniffer Setup-Tool - general configuration

[1] set the IP-address of the adapter which has to be sniffed (only for Windows OS, see the 'note' below)
[2] setup the logging parameters (the parameters are described in chapter 1.3)
[3] click to open a dialog for selecting the log-directory
[4] click to edit the log-directory manually

Note:
A specific adapter IP-Address can only be set for Windows OS.
For Linux systems the whole incoming and outgoing network traffic is captured, i.e. you have to define appropriate 'intelligent' filters to filter the traffic for a specific adapter interface.

dump IP-Frames

if checked then the sniffed IP data are displayed in more comfortable way, as shown following

```
17:18:50,983 [33] : 450000211E4200008011000C0A80002C0A80002D90EB806000DF4BC1234567890
================================================================================
ID : 1          size: 33        type: UDP        TTL: 128
Src:     192.168.0.2:55566          Dest:     192.168.0.2:47110
-------00-01-02-03-04-05-06-07-|-08-09-0A-0B-0C-0D-0E-0F-----0123456789ABCDEF
000000 45 00 00 21 1E 42 00 00 | 80 11 00 00 C0 A8 00 02     E..!B..........
000010 C0 A8 00 02 D9 0E B8 06 | 00 0D F4 BC 12 34 56 78     .............4Vx
000020 90                                                    .

--------------------------------------------------------------------
```

activate logger

if checked, the sniffed IP data are logged in a log-file

add LF

if checked, a line feed is inserted in the log file after each log entry to improve readability

1.4.2 Filter Configuration

Select a filter-node in the tree-view to setup the corresponding filter configurations.

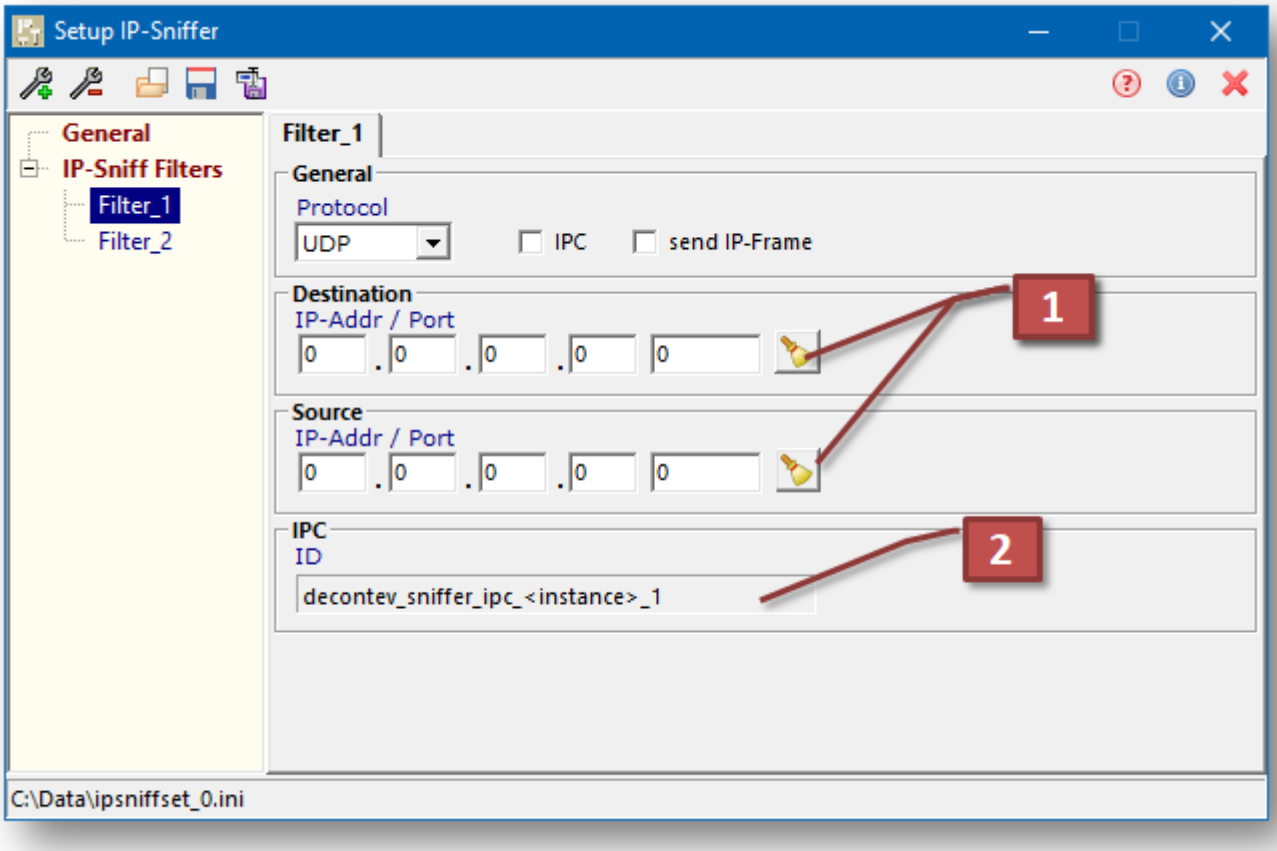

Figure 4: IP-Sniffer Setup-Tool - filter configuration

[1] click to reset the IP address and port
[2] IPC-ID for the selected filter
Note: it cannot be changed

Note:

- **an IP address '0.0.0.0' means there is no address filtering**
- **a port number '0' means there is no port filtering**

IPC
if checked then the data output for this filter are sent out via IPC

send IP-Frame
if checked then the IP frame as shown above is sent out via IPC, otherwise the IP data are sent out as hexadecimal data stream

1.5 Applications