Software Installation Guide

# Napatech Link-Capture™ Software

## Napatech libpcap Installation

This document explains how to install the Napatech libpcap for Napatech Link-Capture™ Software, and how to configure the SmartNICs when using libpcap.

DN-0428 Rev. 28

Intellectual property rights

This document is the property of Napatech. The information contained herein, in whole or in part, may not be duplicated, used or disclosed outside the recipient for any purpose other than to conduct business and technical evaluation. This restriction does not limit the recipient's right to use information contained in this document if it is obtained from another source without restriction.

Disclaimer

This document is intended for informational purposes only. Any information herein is believed to be reliable. However, Napatech assumes no responsibility for the accuracy of the information. Napatech reserves the right to change the document and the products described without notice. Napatech and the authors disclaim any and all liabilities.

Trademark notice

Napatech is a trademark used under license by Napatech A/S. All other logos, trademarks and service marks are the property of the respective third parties.

Copyright statement

# Modification History

This document has been updated as follows:

| Rev. | Date | Comment |
|---|---|---|
| 1 | 2011-05-30 | Initial version. |
| 2 | 2012-01-31 | Changes have been made in Chapters 2, 3 and 4. A number of minor changes have been made. |
| 3 | 2012-07-10 | NT40E2-4 and NT4E2-4T-BP have been included. Changes have been made in Sections 2.4 and 3.1.1. Changes have been made in Chapter 4. A number of minor changes have been made. |
| 4 | 2013-04-16 | NT40E2-1, NT20E2-PTP and NT4E2-4-PTP have been included. NT20E has been removed. Changes have been made in Section 3.1.1. A number of minor changes have been made. |
| 5 | 2013-05-28 | A number of minor changes have been made. |
| 6 | 2013-12-09 | NT20E has been included. Changes have been made in Section 3.1.3. Section 3.2 has been added. A number of minor changes have been made. |
| 7 | 2014-03-05 | NTPORT4E and NT4E-4T-STD have been included. Changes have been made in Chapter 1. Changes have been made in Sections 2.1 and 2.4. Chapter 4 has been added. A number of minor changes have been made. |
| 8 | 2014-04-10 | NT40E3-4-PTP has been included. Section 2.8 has been added. A number of minor changes have been made. |
| 9 | 2014-05-19 | The document has been reformatted. A few minor changes have been made. |
| 10 | 2014-05-23 | A few minor changes have been made. |
| 11 | 2014-07-21 | NT100E3-1-PTP has been included. The term adapter has been changed to accelerator. A number of minor changes have been made. |
| 12 | 2014-11-17 | A number of minor changes have been made. |
| 13 | 2015-04-28 | NT20E3-2-PTP has been included. A number of minor changes have been made. |
| 14 | 2016-01-22 | The description of the ntpcap.ini file now includes a [Common] section. A list of unsupported libpcap functions has been included. A number of minor changes have been made. |
| 15 | 2016-03-29 | Information about installing libpcap for Napatech Driver Software (2GD) has been included. Installing and Running libpcap Applications has been updated. A number of minor changes have been made. |
| 16 | 2016-07-13 | A few minor changes have been made. |
| 17 | 2016-12-13 | A few minor changes have been made. |

| Rev. | Date | Comment |
|---|---|---|
| 18 | 2017-05-11 | Information about installing libpcap for Napatech Driver Software (2GD) has been removed. Installing and Running libpcap Applications has been updated. A section about installing Bro has been added. A few minor changes have been made. |
| 19 | 2017-10-01 | Changes have been made in Chapter 1. Changes have been made in Sections 3.3.1, 3.4.1 and 3.4.2. A number of minor changes have been made. |
| 20 | 2018-02-21 | Changes have been made in Section 2.1. The term accelerator has been changed to SmartNIC. A number of minor changes have been made. |
| 21 | 2018-09-10 | A number of minor changes have been made. |
| 22 | 2018-10-01 | A number of minor changes have been made. |
| 23 | 2018-10-29 | A few minor changes have been made. |
| 24 | 2020-03-26 | Changes have been made in Section 3.2. A few minor changes have been made. |
| 25 | 2020-12-08 | A number of minor changes have been made. |
| 26 | 2021-04-22 | Changes have been made in Chapter 1. Changes have been made in Sections 3.2 and 5.4. A number of minor changes have been made. |
| 27 | 2021-05-21 | Changes have been made in Section 3.2.1. A number of minor changes have been made. |
| 28 | 2022-06-03 | Changes have been made in Section 2.2. A number of minor changes have been made. |

# Contents

# 1 Introduction

### In this document

This document contains information about installation and configuration of libpcap with Napatech extensions (Napatech libpcap) for Napatech Link-Capture™ Software.

**Note:** Napatech libpcap only applies to Linux operating systems.

**Caution:** If a user-defined installation directory was used for installation of the Napatech Linux driver, this directory must replace `/opt/napatech3/` everywhere throughout this document.

### libpcap versions

Napatech libpcap is based on the libpcap 1.10.0 and 1.10.2 releases.

# 2 Installing the Napatech libpcap

In this chapter

This chapter explains how to install the Napatech libpcap and how to install and run libpcap applications with the Napatech libpcap.

This chapter contains the following sections:

## 2.1 Installing libpcap with Napatech Extensions

Context

**Tip:** You can use the `package_install_3gd-x.y.z.sh` script described in DN-0379 to install Napatech libpcap, instead of using the procedure in this section.

Prerequisite

It is a prerequisite for installing Napatech libpcap that an NT driver package is installed (see DN-0379).

The following are required in order to build and install libpcap:

- GCC version 4.0 or later
- GNU **`make`**
- **`flex`** and **`bison`** or **`lex`** and **`yacc`**. Some versions of **`lex`** are insufficient to compile libpcap; if this is the case, install **`flex`** and **`bison`**.

Steps

To install libpcap with Napatech extensions:

| Step | Action |
|---|---|
| **1** | Download the product package for your operating system from supportportal.napatech.com (the Napatech Support Portal). |
| **2** | Expand the product package archive and enter the `ntanl_package_3gd-x.y.z-linux/pcap` directory:<br>`$ tar xzf ntanl_package_3gd-x.y.z-linux.tar.gz`<br>`$ cd ntanl_package_3gd-x.y.z-linux`<br>`$ cd pcap` |
| **3** | Enter the `nt_libpcap_x.y.z` directory:<br>`$ tar xfz nt_libpcap_x.y.z.tar.gz`<br>`$ cd nt_libpcap_x.y.z` |
| **4** | Determine which version of libpcap with Napatech extensions that you want to install. Refer to the `README` file for more information about available versions.<br>`$ less README`<br>`$ cd libpcap-x.y.z` |
| **5** | Configure for installation in the Napatech driver installation directory:<br>`$ ./configure --prefix=/opt/napatech3` |

| Step | Action |
|---|---|
| 6 | To install the libpcap static and shared library, execute the following command:<br>`$ make shared`<br>`$ sudo make install`<br>`$ sudo make install-shared` |
| 7 | To update ldd cache, execute the following command:<br>`$ sudo ldconfig -f /etc/ld.so.conf` |

## 2.2 Installing and Running libpcap Applications

Context

You can install most libpcap-based application for use with Napatech libpcap interfaces using your distributions package manager.

Prerequisite

If the user group `napatech` exists, you must have root privileges or be a member of the `napatech` group in order to use Napatech libpcap interfaces.

Dynamically linked applications should work the same whether installed before or after the Napatech libpcap.

Applications that are statically linked to libpcap must be recompiled and linked to the Napatech libpcap.

Ensure the use of the Napatech libpcap library

If other libpcap libraries are installed on the target platform, it is uncertain which libpcap library will be used. There are two ways to ensure that the Napatech libpcap library is used:

- By removing all other libpcap libraries from the system
- By forcing the application to use the Napatech libpcap library

To force an application to use the Napatech libpcap library, use one of these methods:

Set **`LD_PRELOAD`** in the environment for the application:

```
$ LD_PRELOAD=/opt/napatech3/lib/libpcap.so <app>
```

Set **`LD_LIBRARY_PATH`** in the environment for the application:

```
$ LD_LIBRARY_PATH=/opt/napatech3/lib/ <app>
```

Common applications are available from your distribution

Most common libpcap-based applications can be installed from your distributions default package repositories, and will run without further configuration. The package name will depend on the distribution.

For example, to install and run Wireshark on RHEL/CentOS:

```
$ sudo yum install wireshark-gnome
$ wireshark
```

On Ubuntu and derivatives of Ubuntu, for example Mint, you must preload the Napatech libpcap shared library like this:

```
$ sudo apt-get install wireshark
$ LD_PRELOAD=/opt/napatech3/lib/libpcap.so wireshark
```

### Extra repositories for popular applications

If a package is not available from your distributions default package repositories, you might be able to add a repository that offer the package. The application homepage will often offer detailed instructions about how to add the repository and install the package.

### Package downloads for popular applications

For other applications you may have to download the package and install locally.

For more information about installing and running supported open-source libraries and applications with Napatech SmartNICs, see DN-1319.

### Application must be recompiled from source

If you need to compile the application yourself from source because no package is available for your distribution, the process will depend on the configure options of the application.

**Note:**

You should preferably link to the Napatech libpcap library to ensure the best match to the version of the Napatech libpcap.

The following examples use the installation directory prefix (`--prefix`) `/opt/napatech3`.

Most applications rely on Autoconf for configuration, and you will often find configure script options that allow you to select Napatech libpcap headers and libraries. You will usually have to specify linker flags.

### libpcap configure script option

For example, for Tcpreplay there is a `--with-libpcap` option that helps the configure script find Napatech libpcap:

```
$ wget https://github.com/appneta/tcpreplay/releases/download/v4.1.0/tcpreplay-4.1.0.tar.gz
$ tar xzf tcpreplay-4.1.0.tar.gz
$ cd tcpreplay-4.1.0
$ ./configure --prefix=/opt/napatech3 LDFLAGS="-L/opt/napatech3/lib -lntapi -lntos"
--with-libpcap=/opt/napatech3 --enable-dynamic-link --enable-force-inject
$ make
$ sudo make install
```

Even if a `--with-libpcap` option is available, you may also need to set compiler flags for the configure script to find Napatech libpcap. This is the case for recent versions of Tcpreplay:

```
$ wget https://github.com/appneta/tcpreplay/releases/download/v4.2.4/tcpreplay-4.2.4.tar.gz
$ cd tcpreplay-4.2.4
$ ./configure --prefix=/opt/napatech3 CFLAGS=-I/opt/napatech3/include
LDFLAGS="-L/opt/napatech3/lib -lntapi -lntos" --with-libpcap=/opt/napatech3
--enable-dynamic-link --enable-force-inject
$ ...
```

### Compiler and linker flags

For Tcpdump, you have to specify the Napatech libpcap using compiler and linker flags:

```
$ wget http://www.tcpdump.org/release/tcpdump-4.9.0.tar.gz
$ tar xzf tcpdump-4.9.0.tar.gz
$ cd tcpdump-4.9.0
$ ./configure --prefix=/opt/napatech3 CFLAGS=-I/opt/napatech3/include
LDFLAGS="-L/opt/napatech3/lib -lntapi -lntos"
$ make
$ sudo make install
```

Some configure scripts does not allow compiler and linker flags as options. In that case, set the flags in the environment:

```
$ export CFLAGS=-I/opt/napatech3/include
$ export LDFLAGS="-L/opt/napatech3/lib -lntapi -lntos"
$ ./configure --prefix=/opt/napatech3 ...
$ ...
```

# 3 Configuration

### In this chapter

This chapter explains how to configure Napatech libpcap.

### Backus-Naur form

In Format of the ntpcap.ini File on page 11 the syntax for the `ntpcap.ini` file is specified using Backus-Naur form (BNF) notation. The BNF semantics used are explained in this table.

| Symbol | Meaning |
|---|---|
| `<...>` | Syntactic category (nonterminal) |
| `'...'` | As is (literal word – terminal) |
| `[...]` | Optional, can appear zero or 1 time |
| `{...}` | Optional, can appear zero or more times |
| `\|` | Or (separates alternatives) |

This chapter contains the following sections:

- Configuring libpcap on page 10
- The ntpcap.ini File on page 11
- Optimization on page 15
- Reconfiguration on page 17
- TX Performance on page 17

## 3.1 Configuring libpcap

### Introduction

The Napatech libpcap library works on devices explicitly created and assigned to a stream in the `ntpcap.ini` configuration file as well as streams created, for instance, by using the **`ntpl`** tool. For each stream a virtual device is created.

### Existing streams

Each stream not created by `ntpcap.ini` appears to libpcap as a device named `nt3g<`*n*`>`, where <*n*> is the stream ID. Examples of names are: `nt3g0` and `nt3g1`. Only streams created by `ntpcap.ini` have TX capabilities in libpcap.

### Streams defined in the ntpcap.ini file

The configuration file `ntpcap.ini` is read whenever a libpcap application initializes its device list.

libpcap devices corresponding to streams defined in the `ntpcap.ini` file have user-defined names. The names are defined in the `ntpcap.ini` file, which must have the format described in Format of the ntpcap.ini File on page 11. Streams defined in the `ntpcap.ini` file can have both RX and TX capabilities in libpcap.

You can create a new `ntpcap.ini` file from scratch in `/opt/napatech3/config/` or copy a default `ntpcap.ini` file to `/opt/napatech3/config/`.

If you have installed libpcap using `package_install_3gd-`*`x.y.z`*`.sh`, a default `ntpcap.ini` file has been copied to `/opt/napatech3/config` (see The Default ntpcap.ini File on page 13).

If you have installed libpcap using the procedure in Installing libpcap with Napatech Extensions on page 7, you can execute these commands to copy a default `ntpcap.ini` file to `/opt/napatech3/config`:

```
$ cd libpcap-x.y.z
$ cp ntpcap.ini /opt/napatech3/config/
```

Considerations on allocation of TX host buffers

When TX capabilities are defined for a libpcap device in the `ntpcap.ini` file, one TX host buffer is used for each application that opens the device. This is in contrast to RX, where all applications share the same host buffers created by NTPL and associated with a stream ID. The maximum number of RX and TX host buffers are specified in the `ntservice.ini` file (see DN-0449).

**Note:** In the default `ntservice.ini` file 4 TX host buffers are allocated, and in the default `ntpcap.ini` file, 4 libpcap devices (**`napa0`**, **`napa1`**, **`napa2`**, and **`napa3`**) with TX capabilities are defined (see The Default ntpcap.ini File on page 13). So when all 4 devices are opened, all 4 available TX host buffers are used, and an attempt from a new application to open one of the 4 devices and thereby use a fifth host buffer will fail. The solution is to allocate more TX host buffers in the `ntservice.ini` file, or to remove TX capabilities in the `ntpcap.ini` file for devices not needing them.

## 3.2 The ntpcap.ini File

In this section

This section explains configuration using the `ntpcap.ini` file.

Main function

The main function of the `ntpcap.ini` file is to be able to tag NTAPI properties, for instance TX port, host buffer allowance, nanosecond resolution and NUMA node, to the libpcap device.

Using NTPL in the ntpcap.ini file

To ease configuration, NTPL commands can be used to set up filters specifically for each libpcap device. This means that it is not necessary to preconfigure filters, for instance, using the **`ntpl`** tool (see DN-0449). They are configured when the libpcap device is activated.

However, NTPL commands configured outside the `ntpcap.ini` file can also be used. The stream ID must then be associated with the libpcap device in the `ntpcap.ini` file.

**Note:** System-wide NTPL commands, for instance **`Setup`** and **`DefineMacro`**, should not be specified in the device sections of the `ntpcap.ini` file. They will be accepted, but will not work as expected. Instead they must be executed separately in the **`[Common]`** section of nrpcap.ini, using the **`ntpl`** tool, NTAPI or a filter configuration file on **`ntservice`** load.

### 3.2.1 Format of the ntpcap.ini File

Description

The configuration file `ntpcap.ini` uses the INI file format.

The configuration file consists of a [Common] section and a section for each libpcap device. For each device section, a number of optional keys specify additional parameters.

**Note:** If a device is configured erroneously, it is not created. However, no error message is shown.

Syntax

The syntax for defining one device is as shown below.

```
['[Common]']
{'Ntpl'<n> '=' '"'<NTPL command>'"'}
'['<device name>']'
{'RX'<n> '=' '"'<NTPL command>'"'}
['StreamId' '=' <ID>]
['HBAllowance' '=' <percentage>]
['TX' '=' <port No.>]
['TXNumaNode' '=' <node No.>]
['TXMinHostBuffer' '=' <minimum buffer size>]
['NTInterface' '=' 'Segment' | 'Packet']
['WithCRC' '=' 'TRUE' | 'FALSE']
```

Semantics

The semantics of `ntpcap.ini` is described below.

| Key | Description |
|---|---|
| **`[Common]`** | A section which may hold NTPL commands for common use, for example NTPL macros and NTPL setup commands. If the **`[Common]`** section is present, then `"Delete=All"` is executed implicitly on entry. libpcap makes sure this is only executed once when multiple devices are opened. After all devices have been closed, opening any device will re-initiate the execution of the NTPL commands in the **`[Common]`** section. |
| **`Ntpl<n>`** | Optional. Must be specified in the **`[Common]`** section.<br><*n*> = 1..256, <*n*> must start at 1 and increment consecutively. The NTPL commands are loaded in this sequence. |
| **`<device name>`** | A user-defined libpcap device name. If the name clashes with an existing device name, this section of `ntpcap.ini` is ignored.<br>**Note:** Streams that are already created on the SmartNIC, for instance by using the **`ntpl`** tool, are automatically available for libpcap. They are named `nt3g`<*x*>, where <*x*> is the stream ID. To avoid naming clashes, do not use these device names in `ntpcap.ini`. |
| **`Rx<n>`** | Optional. NTPL commands for setting up this libpcap device.<br><*n*> = 1..256, <*n*> must start at 1 and increment consecutively. The NTPL commands are loaded in this sequence. A break in the sequence stops loading commands.<br>If at least **`Rx1`**, and optionally more NTPL commands, assigns or refers to the same single stream ID (not a range), the libpcap device is assigned to this stream ID. Make sure to use the same stream ID in these NTPL commands; otherwise this section of `ntpcap.ini` is ignored, and the libpcap device is not created.<br>If a stream with this stream ID has already been created on a SmartNIC, for instance by a previous section in `ntpcap.ini` or by using the **`ntpl`** tool, the NTPL commands are not loaded. Make sure to use a stream ID that is not already in use.<br>If no NTPL command assigns or refers to a single stream ID, use the key **`StreamId`** to assign an existing stream, created for instance in a previous section of `ntpcap.ini` or by using the **`ntpl`** tool. |
| **`StreamId`** | Optional. Assigns the libpcap device to a stream ID. The stream must already have been created.<br>If set, **`Rx<n>`** NTPL commands are not searched for a stream ID.<br>Use the **`StreamId`** key if no NTPL commands assign or refer to a specific stream ID (see the load-balancing example in Examples of ntpcap.ini Files on page 14).<br>Use the **`StreamId`** key to assign the libpcap device to a stream created in a previous section of `ntpcap.ini` or, for instance, by using the **`ntpl`** tool (see the host buffer allowance example in Examples of ntpcap.ini Files on page 14). |
| **`HBAllowance`** | Optional. Specifies the host buffer allowance in percent for this device. Range: 1..100. Default: 100.<br>All applications opening this libpcap device will have the same host buffer allowance value. Create more devices assigned to the same stream ID if different host buffer allowance values are needed.<br>See the host buffer allowance example in Examples of ntpcap.ini Files on page 14 |

| Key | Description |
|---|---|
| **`TX`** | Optional. If specified, determines the port used to transmit on. Default: No TX port is assigned. |
| **`TXNumaNode`** | Optional. Specifies from which NUMA node pool of host buffers the TX buffer is taken. Only relevant if a TX port is specified. Default: 0. |
| **`TXMinHostBuffer`** | Optional. Specifies the minimum size in MB of the TX host buffer needed. |
| **`NTInterface`** | Optional. Selects between segment-based interface and packet-based interface. Default: `Segment`.<br>Segment-based interface is faster and supports the PCAP descriptor (`PacketDescriptor=PCAP` in `ntservice.ini`). However, segment-based interface does not support merging of packets between SmartNICs.<br>Packet-based interface supports merging of packets between SmartNICs, but does not support the PCAP descriptor. |
| **`WithCRC`** | Optional. Default: `False`. If **`PacketDescriptor`** is set to `NT` in the `ntservice.ini` file as shown in the configuration example below, set **`WithCRC`** to `True` to include FCS in a received frame. And set **`WithCRC`** to `False` not to include FCS in a received frame.<br>`PacketDescriptor=NT`<br>`TimestampFormat=NATIVE_UNIX (or TimestampFormat=UNIX_NS)`<br>**Note:** The **`PacketPcapFcsInclude`** parameter in the `ntservice.ini` file does not have any effect if **`PacketDescriptor`** is set to `NT`.<br>**Note:** The **`WithCRC`** value in `ntpcap.ini` file does not have any effect if **`PacketDescriptor`** in the `ntservice.ini` file is set to `PCAP`. See the configuration examples below.<br>The FCS field will be included in a received frame with the following configuration in the `ntservice.ini` file regardless of the **`WithCRC`** value in the `ntpcap.ini` file.<br>`PacketDescriptor=PCAP`<br>`TimestampFormat=PCAP_NS (or TimestampFormat=PCAP)`<br>`PacketPcapFcsInclude=True`<br>The FCS field will not be included in a received frame with the following configuration in the `ntservice.ini` file regardless of the **`WithCRC`** value in the `ntpcap.ini` file. See DN-0449 for further information in detail about parameters in the `ntservice.ini` file.<br>`PacketDescriptor=PCAP`<br>`TimestampFormat=PCAP_NS (or TimestampFormat=PCAP)`<br>`PacketPcapFcsInclude=False` |

### 3.2.2 The Default ntpcap.ini File

Description

The default `ntpcap.ini` file creates 4 devices (if 4 ports are available), named **`napa0`**, **`napa1`**, **`napa2`** and **`napa3`**. Each device has one filter assigned and both RX and TX capabilities.

Contents of ntpcap.ini

This is the contents of the default `ntpcap.ini` file:

```
# Napatech default ntpcap.ini file
# Note that you might have less than 4 ports on your system.
# If this is the case, then napa1 and/or napa2 and/or napa3
# will be silently ignored when using this default config file.

[Common]
Ntpl1 = "Assign[StreamId=252]=Port==0"
Ntpl2 = "Assign[StreamId=253]=Port==1"
Ntpl3 = "Assign[StreamId=254]=Port==2"
Ntpl4 = "Assign[StreamId=255]=Port==3"

[napa0]
StreamId=252
Tx=0

[napa1]
StreamId=253
Tx=1

[napa2]
StreamId=254
Tx=2

[napa3]
StreamId=255
Tx=3
```

### 3.2.3 Examples of ntpcap.ini Files

Load-balancing example

In this load-balancing example, `ntpcap.ini` creates 4 libpcap devices, named **`napa_lb0`**, **`napa_lb1`**, **`napa_lb2`**, and **`napa_lb3`**. All received frames are distributed to streams 20 to 23 using a 2-tuple hash key algorithm. `Hash2Tuple` uses IP address information for specifying the flow. For the device **`napa_lb0`**, the stream ID must specified by a **`StreamId`** key, because the NTPL **`Assign`** command that creates the streams does not specify a single stream ID value.

```
[Common]
Ntpl1="Assign[StreamId=(20..23);TxPort=0;TxIgnore=True]=All"
Ntpl2="HashMode=Hash2Tuple"

[napa_lb0]
StreamId=20
Tx=0

[napa_lb1]
StreamId=21
Tx=1

[napa_lb2]
StreamId=22

[napa_lb3]
StreamId=23
```

Host buffer allowance example

In this host buffer allowance example, `ntpcap.ini` creates 4 libpcap devices, named **`napa_hba`**, **`napa_hba25`**, **`napa_hba50`** and **`napa_hba75`**. They share the same stream, but have different host buffer allowance values. A slow application that uses **`napa_hba25`** will loose packets to give way for a faster application that uses **`napa_hba75`**. For the first libpcap device, the host buffer allowance is 100% (default value).

```
[Common]
Ntpl1="Assign[StreamId=10;TxPort=0;TxIgnore=True]=All"

[napa_hba25]
StreamId=10
HBAllowance=25

[napa_hba50]
StreamId=10
HBAllowance=50

[napa_hba75]
StreamId=10
HBAllowance=75
```

[Common] section example

This example is very similar to the default `ntpcap.ini` file, except that the NTPL commands that assigns ports to streams are collected in the **`[Common]`** section.

```
[Common]
Ntpl1 = "Assign[StreamId=0;TxPort=0;TxIgnore=True]=Port==0"
Ntpl2 = "Assign[streamId=1;TxPort=0;TxIgnore=True]=Port==1"
Ntpl3 = "Assign[streamId=2;TxPort=0;TxIgnore=True]=Port==2"
Ntpl4 = "Assign[streamId=3;TxPort=0;TxIgnore=True]=Port==3"

[napa0]
StreamId = 0
Tx = 0

[napa1]
StreamId = 1
Tx = 1

[napa2]
StreamId = 2
Tx = 2

[napa3]
StreamId = 3
Tx = 3
```

## 3.3 Optimization

In this section

This section gives some guidelines for setting up the `ntpcap.ini` file and the `ntservice.ini` file.

### 3.3.1 Setting Up ntpcap.ini and ntservice.ini for Merging Between SmartNICs

Prerequisite

**Note:** For merging between SmartNICs, time synchronization is required. See DN-0985 and/or DN-0737 for further information regarding time synchronization.

Context

If merging of packets between SmartNICs is needed, the packet-based interface must be used. This means that the PCAP packet descriptor and the PCAP-ns and PCAP-µs time stamp formats cannot be used.

Steps

To set up for merging packets between SmartNICs:

| Step | Action |
|---|---|
| **1** | In the `ntpcap.ini` file, include the following to set up packet-based interface.<br>`NTInterface = Packet` |
| **2** | In the `ntservice.ini` file, include the following to select the standard packet descriptor (default setting).<br>`PacketDescriptor = NT` |
| **3** | In the `ntservice.ini` file, select one of the following to select the native, native UNIX (default setting), native UNIX ns or native NDIS time stamp format, respectively.<br>• `TimestampFormat=NATIVE_UNIX`<br>**Note:** The native UNIX format is based on a 10 ns unit with a 64-bit unit counter. The start time is January 1st 1970.<br>• `TimestampFormat=UNIX_NS`<br>**Note:** The native UNIX ns format is based on a 1 ns unit with a 64-bit unit counter. The start time is January 1st 1970. |

### 3.3.2 Setting Up ntpcap.ini and ntservice.ini for No Merging Between SmartNICs

Prerequisite

Context

If merging of packets between SmartNICs is not needed, it is recommended to use the segment-based interface, as it supports the use of the PCAP packet descriptor with the PCAP-ns or PCAP-µs time stamp format.

If the standard packet descriptor is used, the time stamp conversion must be carried out in libpcap before delivering each packet to the application. Therefore the PCAP packet descriptor is faster than the standard packet descriptor.

Steps

To set up when merging packets between SmartNICs is not needed:

| Step | Action |
|---|---|
| **1** | In the `ntpcap.ini` file, include the following to set up segment-based interface (default setting).<br>`NTInterface=Segment` |

| Step | Action |
|---|---|
| 2 | In the `ntservice.ini` file, include the following to select the PCAP packet descriptor.<br>`PacketDescriptor=PCAP` |
| 3 | In the `ntservice.ini` file, include one of the following to select the PCAP-ns or PCAP-µs time stamp format, respectively.<br>• `TimestampFormat=PCAP_NS`<br>**Note:** The PCAP-ns format is based on a 1 ns unit with a 32-bit unit counter in MSBs of the time stamp descriptor field, and a 32-bit second counter in LSBs of the time stamp descriptor field. The 1 ns unit counter is incremented in quantities of 1 ns. The start time is January 1st 1970.<br>• `TimestampFormat=PCAP`<br>**Note:** The PCAP-µs format is based on a 1000 ns (1 µs) unit with a 32-bit unit counter in MSBs of the time stamp descriptor field, and a 32-bit second counter in LSBs of the time stamp descriptor field. The start time is January 1st 1970. |

## 3.4 Reconfiguration

Allowed NTPL commands

All NTPL commands are possible after the driver is loaded.

**Note:** The NTPL command `Delete=All` will destroy all defined streams and leave the system in a state where no data is shown until a new **`Assign`** command has been issued.

## 3.5 TX Performance

TX performance

Frames are always transmitted as fast as possible and not using a specific IFG (inter-frame gap).

# 4 Statistics

In this chapter

This chapter explains the Napatech implementation of a number of libpcap statistics counters.

This chapter contains the following sections:

- libpcap Statistics Counters on page 18

## 4.1 libpcap Statistics Counters

Napatech's implementation

This table shows Napatech's implementation of the **`pcap_stat`** capture statistics counters.

| Counter | Description |
|---|---|
| **`ps_recv`** | Counts the number of packets received by libpcap before the PCAP filter. |
| **`ps_drop`** | Counts all packets dropped by the NT Driver software. Packets might be dropped if streams are shared and opened with host buffer allowance < 100%. |
| **`ps_ifdrop`** | Counts the number of packets lost due to host buffer overrun for the actual stream. If multiple host buffers are attached to the stream ID used in the actual libpcap device (merging), the individual drop counters are combined into this **`ps_ifdrop`** counter. |

# 5 Troubleshooting and Limitations

In this chapter

This chapter gives some troubleshooting guidelines and describes limitations of NT libpcap.

This chapter contains the following sections:

## 5.1 Performing Troubleshooting

Steps

To perform troubleshooting in case of problems:

| Step | Action |
|---|---|
| 1 | Activate `#define NAPATECH_DEBUG` and `#define NAPATECH_CALL_TRACE` in the `pcap-napatech-intl.h` file. |
| 2 | Reinstall Napatech libpcap. |
| 3 | Look in `syslog` for the debug output. |

## 5.2 Error Codes

Explanation

This table explains two error codes that might occur in special situations:

| Situation | Error Code | Explanation | Solution |
|---|---|---|---|
| After running `./configure` | `>>Syntax error: "fi" unexpected (expecting ")")<<`<br>or similar | Your configure shell script must be recreated. GNU AutoConf 2.59 - 2.62 is supported. | Run **`autoconf`** |
| After running **`make`** or **`gmake`** | `>>gcc: @NAPATECH_LIBS@: No such file or directory<<`<br>or similar | The `make` file you are using has not been fused by configure on your platform. | Run `./configure` |

## 5.3 Missing libpcap Devices

Explanation

If a virtual device does not show up in the libpcap device list, the explanation can be one of these:

- An NTPL **`Assign`** command has specified an invalid/unavailable port. This does not generate an error.
- There are no more host buffers available for RX.
- One or more NTPL commands are erroneous or incompatible.
- No stream ID is defined in an NTPL **`Assign`** command for the virtual device, and the **`StreamId`** parameter is absent.
- Different stream IDs are defined in two RX specifications for the same virtual device, and the **`StreamId`** parameter is absent.

**Note:** To avoid errors in NTPL commands, it is recommended to test an NTPL command using the **`ntpl`** tool, before adding the command to the `ntpcap.ini` file.

**Note:** If the name of a virtual device corresponding to an existing stream clashes with that of a virtual device defined in the `ntpcap.ini` file (see Format of the ntpcap.ini File on page 11), the latter is ignored. To avoid unintended clashes, make sure to use stream IDs in the `ntpcap.ini` file that are not used in other configurations of the installed SmartNICs.

## 5.4 Limitations

Filtering

Filtering using **`pcap_setfilter`** is SW-based and does not utilize the filtering capabilities of the SmartNICs. Filtering using Napatech libpcap with configuration files is HW-based. It is also possible to use the **`ntpl`** tool (see DN-0449) for HW filtering after the driver has been loaded.

libpcap functions

Napatech libpcap supports all libpcap functions with a few limitations:

| libpcap Function | Limitation |
|---|---|
| **`int pcap_setdirection(pcap_t *, pcap_direction_t)`** | Supports only **`PCAP_D_N`**, received packets. |
| **`int pcap_set_promisc(pcap_t *, int)`** | Accepted, but will always run promiscuous mode. |
| **`int pcap_set_timeout(pcap_t *, int)`** | Has no effect. Will always run 100 ms time-out. |
| **`int pcap_set_tstamp_type(pcap_t *, int)`** | Supports only **`PCAP_TSTAMP_ADAPTER_UNSYNCED`** SmartNIC time-stamping. |
| **`int pcap_set_buffer_size(pcap_t *, int)`** | Has no effect. Will always run zero-copy mode. |

# Back Matter

The back matter contains these sections:

# Style Conventions

**Bold typeface** is used for names of, for instance, user interface elements and software components.

*`Italic typeface`* is used for replaceable text.

`Monospaced typeface` is used for code, commands and file names.

# Abbreviations

| Abbreviation | Explanation |
|---|---|
| BNF | Backus-Naur Form |
| BP | ByPass |
| CentOS | Community ENTerprise Operating System |
| CRC | Cyclic Redundancy Check |
| DN | Document Number |
| GCC | GNU Compiler Collection |
| Gen | GENeration |
| GNU | GNU’s Not Unix |
| HB | Host Buffer |
| hba | Host Buffer Allowance |
| http | HyperText Transfer Protocol |
| https | HyperText Transfer Protocol Secure |
| HW | HardWare |
| Id, ID | IDentifier |
| IFG | Inter-Frame Gap |
| libpcap | Packet CAPture LIBrary |
| LSB | Least Significant Bit |
| Min | MINimum |
| MSB | Most Significant Bit |
| NDIS | Network Driver Interface Specification |
| nt, NT | NapaTech |
| NTAPI | NapaTech Application Programming Interface |
| ntpl, Ntpl, NTPL | NapaTech Programming Language |
| Numa, NUMA | Non-Uniform Memory Access |
| pcap, PCAP | Packet CAPture |

| Abbreviation | Explanation |
|---|---|
| PCI | Peripheral Component Interconnect |
| PTP | Precision Time Protocol |
| Rev. | REVision |
| RHEL | Red Hat Enterprise Linux |
| Rx, RX | Receive |
| STD | STanDard |
| SW | SoftWare |
| tx, Tx, TX | Transmission/Transmit |

# References

| Document Number | Explanation |
|---|---|
| DN-0379 | Napatech, Napatech Link Software, Software Installation for Linux, Software Installation Guide |
| DN-0449 | Napatech, Napatech Software Suite, Reference Documentation |
| DN-0737 | Napatech, SmartNICs with Napatech Link-Capture™ Software, Basic Troubleshooting, User Guide |
| DN-0985 | Napatech, Napatech Link-Capture™ Software, Time-Stamping and Time Synchronization, User Guide |
| DN-1319 | Napatech, Running Open-Source Libraries and Applications with Napatech SmartNICs, Application Note |

# Index