n2disk

User's Guide

n2disk v.2.0.1

February 2013

1. Release History

Release 2.0.1 (January 2013)

Hugepages support. Improved multithread support.

Release 2.0 (October 2012)

First release.

2. Introduction

n2disk is a network traffic recorder application. With n2disk you can capture full-sized network packets at multi-Gigabit rate (above 10 Gigabit/s on adequate hardware) from a live network interface, and write them into files without any packet loss. n2disk has been designed to write files into disks for very long periods, you have to specify a maximum number of distinct file that may be written during the execution, and if n2disk reaches the maximum number of files, it will start recycling the files from the oldest one. This way you can have a complete view of the traffic for a fixed temporal window, knowing in advance the amount of disk space needed.

n2disk uses the industry standard PCAP file format (regular and nanosecond) to dump packets into files so the resulting output can be easily integrated with existing third party or even open/source analysis tools (like Wireshark).

n2disk has been designed and developed mainly because most network security systems rely on capturing all packets (both header and payload), since any packets may have been responsible for the attack or could contain the problems that we are trying to find. Netflow information is more manageable and requires less disk space to be stored, but in some cases, like deep-packet-inspection analysis or controlled traffic regeneration, it is not useful. When we need to collect the entire packet, because we need all the information, n2disk has to be used.

n2disk can be effectively used to perform numerous activities, among these:

- Off-line network packets analysis by feeding a specialized tools like Snort.
- Reconstruct particular communication flows or network activities.
- Reproduce the previous captured traffic to a different network interface.

3. Main Features

Some of the n2disk features include:

- Fully user configurable.
- Use of the standard PCAP file format (regular and nanosecond).
- High-performance packet to disk recording.
- BPF filters supports (using the same format as in the popular tcpdump tool) to filter out the unwanted network packets from the recording process.
- Optimized BPF-like filters support, a faster replacement for BPF filters (a subset of the BPF syntax is supported).
- Multi-core support. n2disk has been designed with multicore architectures in mind. It uses at least 2 threads (one for the packet capture and one for the disk writing) and it is possible to further parallelize packet capture using multiple threads. The communication between threads has been carefully optimized.
- PF_RING acceleration. n2disk exploit the packet capture acceleration offered both by standard PF_RING and PF_RING DNA.
- Direct-IO disk access. n2disk uses the Direct IO access to the disks in order to obtain maximum disk-write throughput.
- Real-Time indexing. n2disk is able to produce an index on-the-fly during packet capture. The index can be queried using a BPF-like syntax to quickly retrieve interesting packets in a specified time interval.

Dump Format

Capture files are saved in the standard PCAP format with timestamps in microsecond resolution. The PCAP format with nanosecond resolution is also an option when using the PF_RING support and network cards with hardware timestamp.

Capture files are stored in sequential order with a per-file limit in duration or size. It is also possible to specify the maximum number of files: when the limit is reached, n2disk recycles the files already written, starting from the oldest one.

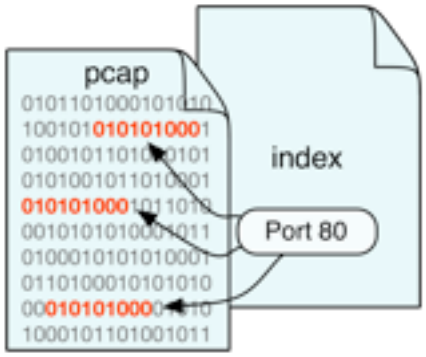
Packet Capture Filters

n2disk uses the same syntax for capture filters as tcpdump and any other program that uses the libpcap library. In fact two type of capture filters are available:

- Standard BPF filters.
- BPF-like filters (a subset of the BPF syntax), a faster replacement for BPF. See Appendix B.

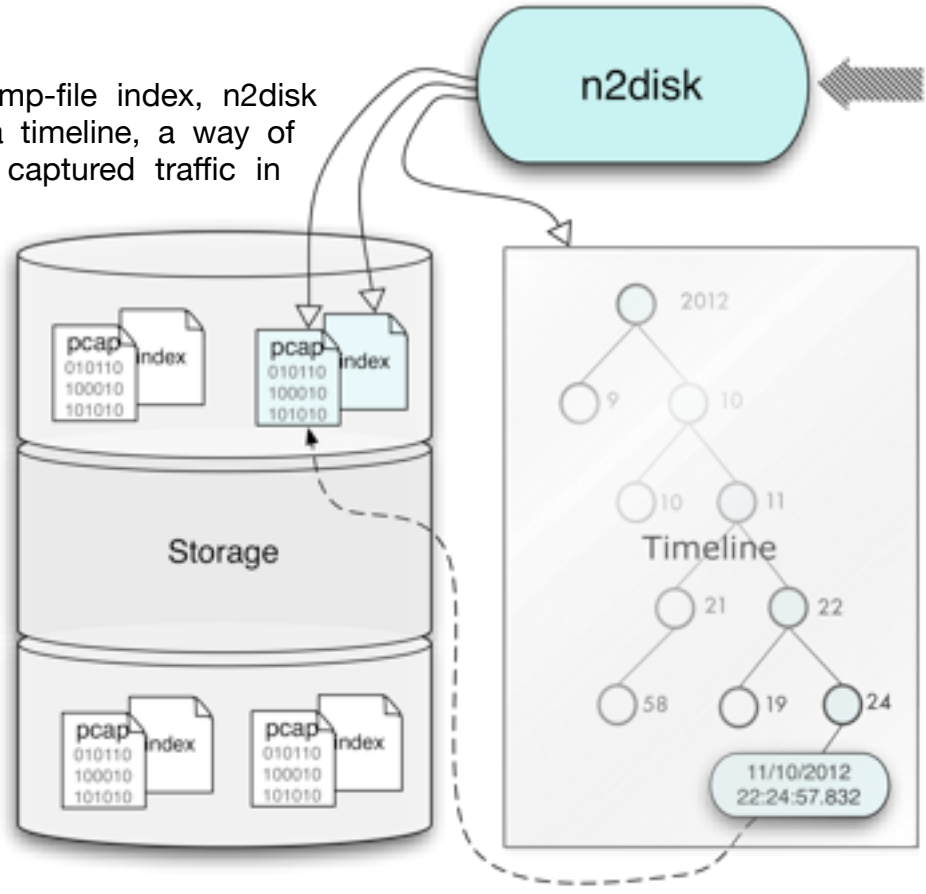
Index and Post-Capture Filters

n2disk produces an index on-the-fly during packet capture. The index is in a separate file associated with the capture file and houses all the flows information and packets offsets.
As soon as the dump file is ready, using the utilities provided with n2disk, it is possible to query the index for specific packets in a time interval using a BPF-like syntax. The packets matching the filter are returned in PCAP format.



Timeline

Besides the per-dump-file index, n2disk can also produce a timeline, a way of keeping the whole captured traffic in chronological order.
Using the utilities provided with n2disk, it is possible to query the timeline for specific packets belonging to the whole dump set in a given time interval.



4. Expected n2disk Performance

The n2disk performance is influenced by two factors:

- Packet capture performance
- Disk write performance

Unless you have both fast capture and write performance, n2disk will not operate properly. In the appendix you can learn how to use DNA drivers to accelerate packet capture performance.

We have implemented in n2disk a testing facility for allowing you to determine if your system has optimal performance. We suggest you to run this performance test prior to use n2disk. n2disk comes with the `-e` command line option

```
[--simulation-mode|-e] <mode> | Simulation mode (debug only)
                              | 1 - Test max dump speed (simulate capture)
                              | 2 - Test max capture speed (simulate storage)
```

The -e option can have two mode:

- 1: packet capture is simulated. Namely n2disk simulates packet capture so that n2disk does not spend time on capturing and thus the capture speed is virtually unlimited. In this mode we can evaluate the disk write performance as packet capture costs virtually zero and it has virtually and infinite speed.
- 2: write to disk is simulated. Namely n2disk does not save packets on disk but it reports that the pcap file has been saved. This way no time is spent writing on disk, and thus we can measure the packet capture performance. Note that in this setup, you must inject traffic on the ingress (-i) n2disk monitoring interface.

In general, disk-write speed should exceed network speed (or packet capture speed) in order to guarantee that all captured packets can be written to disk. Note that disk performance does not depend just on hardware, but also on filesystem type and configuration. Please refer to the appendix for more details on this subject.

5. Versions

The n2disk software comes in three versions: n2disk1g, n2disk, n2disk10g.

n2disk1g

This is the entry-level version, enough for rates up to 1 Gigabit.

n2disk

This is the standard version, with two threads: one for packet capture and one for disk writing. This version can handle rates up to 5 Gigabit.

n2disk10g

This version includes support for multithreaded packet capture, with an internal architecture able to balance the load across up to 32 threads in a zero-copy fashion. This version can handle rates up to 10 Gigabit.

6. Usage

In order to save all the traffic into disks, the n2disk application has to be activated on an interface from which it is possible to capture all the traffic you are interested in. Once activated, n2disk will save the traffic data into the specified directory recycling the files already written, starting from the oldest one, this in case the maximum number of created files is reached.

In the following sections, we discuss all the n2disk 2.x command line options and how to efficiently configure n2disk to capture all the traffic flowing in your network.

Command Line Options

Below the available options and a detailed explanation of each option are listed:

CAPTURE SETTINGS

[--interface|-i] <device>	| Ingress packet device.
[--active-wait|-g]	| Active packet polling.
[--poll-duration|-q] <duration>	| Poll duration (usec). Lower more CPU is used,
	| better response time. Default: 10 usec.
[--snaplen|-s] <len>	| Max packet capture length. Default: 1514.
[--sample-rate|-y] <rate>	| Packet sample rate (e.g. 100 means 1:100 sampling).

FILTERING

[--filter|-f] <filter>	| BPF (tcpdump-like) ingress packet filter.
[--fast-filter|-F] <filter>	| Faster replacement for BPF ingress packet filter.

TIMESTAMPING

[[--time-pulse|-S] [<id>]]	| Enable time pulse thread (optimize packet timestamping)
	| and bind it to the specified core.
[--nanoseconds|-d]	| Dump timestamps in nanosecond format.
[--no-timestamp|-T]	| Do not compute timestamps.
[--simulation-mode|-e] <mode>	| Simulation mode (debug only).
	| 1 - Test max dump speed(simulate capture).
	| 2 - Test max capture speed (simulate storage).

DUMP SETTINGS

[--chunk-len|-C] <len>	Size (KB) of the chunk written to disk (must be multiple of 4096). Default: 64 KB.
[--journaling|-j]	Enable journaling.
[--buffer-len|-b] <len>	Buffer length (MBytes).
[--dump-directory|-o] <dir>	Directory where dump files will be saved.
[--archive-directory|-O] <dir>	Directory where dump files will be archived (slower disks). The -a option is overwritten when using it.
[--max-file-len|-p] <len>	Max pcap file length (MBytes).
[--max-file-duration|-t] <secs>	Max pcap file duration (sec).
[--max-file-packets|-N] <num>	Max packets per pcap file. Default: 0 (no max duration).
[--archive|-a]	Archive pcap file (rename to .old) instead of overwriting if already present on disk.
[--max-num-files|-m] <num>	Max number of files before restarting file name.
[--max-nested-dirs|-n] <num>	Max number of nested dump sub-directories.
[--file-prefix|-x] <prefix>	Dump file prefix.
[--hugepages|-U]	Use hugepages for memory allocation.
[--disable-direct-io|-r]	Disable Direct I/O (experts only).

INDEX SETTINGS

[--index|-I]	Create pcap-index on-the-fly.
[--timeline-dir|-A] <dir>	Time-arranged directory for produced pcaps and indexes.
[--no-index-compression|-D]	Disable index compression.

MULTITHREADING (n2disk10g only)

[--reader-threads|-R] <ids>	Enable multithread support and bind reader threads to the specified core ids (e.g. 0,1,2,3).

CPU AFFINITY

[--reader-cpu-affinity|-c] <id>	Binds the reader thread to the specified core.
[--writer-cpu-affinity|-w] <id>	Binds the writer thread to the specified core. Note: the first available CPU corresponds to 0.

OTHER

[--unprivileged-user|-u] <username>	Use username instead of nobody when dropping privileges.
[--pid|-P] <file>	Set the pid file.
[--version|-V]	Print application version.
[--help|-h]	Print the help.
[--verbose|-v]	Verbose.
[--syslog|-l]	Dump trace messages to syslog.

7. Tuning n2disk performance

In order to achieve a good n2disk setup able to obtain the maximum performance, it is important to take into account the following aspects.

Besides libpcap, n2disk can take advantage from the PF_RING (both standard and DNA) acceleration to capture packets from a live network interface.

In case standard PF_RING is used it is particularly important to reserve enough ring buffer space inside the kernel. Furthermore, in order to reduce the number of clock-cycles needed to capture the packets and cross the network stack, it is possible to turn off the PF_RING transparent mode setting the transparent_mode option to 2.

For further info about PF_RING and DNA please have a look at Appendix A.

Regarding the n2disk start-up parameters particularly important are the following options:

- The buffer length (-b) has to be big enough. 1 GB is sufficient in most cases.
- The write chunk size (-C) has to be greater than or equal to 64 Kbytes.
- The maximum file size (-p) should not be very small. A good value has to be more than 64 Mbytes.
- The core binding for the reader (-c) and writer (-w) thread. It is highly recommended to bind those threads to different core of the same physical CPU (according to the system topology, you should choose the physical CPU where the network card is closer/directly connected). The same applies when using multithreaded packet capture (-R).

Examples

Basic example:

```
./n2disk -i dna0 -o /storage/dna0/ -b 1024 -C 1024 -p 512 -q 1 -
S 0 -c 1 -w 2
```

Multithreaded packet capture (3 threads):

```
./n2disk -i dna0 -o /storage/dna0/ -b 1024 -C 1024 -p 512 -q 1 -
S 0 -c 1 -R 3,4,5 -w 2
```

Packet indexing:

```
./n2disk -i dna0 -o /storage/dna0/ -I -A /index/dna0/ -b 1024 -C
1024 -p 512 -q 1 -S 0 -c 1 -w 2
```

8. Utilities

The following utilities are provided with n2disk.

indexPcap

This utility produces an index file (with the same format of the index produced by n2disk) from a pcap file. Available options are:

[-v]	Verbose
[-h]	Help
[-i] <pcap file>	Pcap file to index
[-o] <index file>	Index file
[-f] <filter>	Index only packets matching the BPF-like filter

Example:

```
./indexPcap -i dummy.pcap -o dummy.index
```

findPacketsWithIndex

This utility, given a pcap file and its index, produces a new pcap file with the packets matching the provided index in BPF-like syntax. Available options are:

[-v]	Verbose
[-h]	Help
[-a] <pcap file>	pcap file from which packets will be extracted
[-b] <index file>	Index corresponding to the pcap file
[-o] <pcap file>	Out pcap file
[-f] <filter>	BPF-like filter

Example:

```
./findPacketsWithIndex -a dummy.pcap -b dummy.index -o
output.pcap -f "host 192.168.1.1 or port 80"
```

findPacketsWithTimeline

This utility, given a "timeline" tree created by n2disk, produces a new Pcap file with the packets matching the provided index in BPF-like syntax. Available options are:

[-v]	Verbose
[-h]	Help
[-t] <timeline dir>	Timeline directory created by n2disk (-A)
[-b] <begin time>	Begin time on which packets will be selected
[-e] <end time>	End time on which packets will be selected
[-o] <pcap file>	Out pcap file
[-f] <filter>	BPF-like filter

Example:

```
./findPacketsWithTimeline -t /tmp/n2disk/timeline -b "2012-10-02
12:00:00" -e "2012-10-03 00:00:00" -o output.pcap -f "host
192.168.1.1 and port 80"
```

Appendix A
PF_RING Support

PF_RING download instructions can be found in http://www.ntop.org/get-started/download/.

PF_RING kernel module installation

Please note that for some Linux distributions an installation package is provided.

In order to compile the PF_RING kernel module from source code you need to have the linux kernel headers installed.

```
cd <PF_RING PATH>/kernel
make
```

The kernel module installation requires root capabilities:

```
cd <PF_RING PATH>/kernel
make install
```

Running n2disk on top of PF_RING

Before using n2disk on top of PF_RING, the pf_ring kernel module should be loaded:

```
insmod <PF_RING PATH>/kernel/pf_ring.ko
```

PF_RING-aware drivers

The PF_RING package constains several aware drivers in "<PF_RING PATH>/drivers/PF_RING_aware/" for different chips of various vendors. With these special drivers it is possible to increase performance changing the PF_RING working mode (via the insmod *transparent_mode* parameter). Available modes are:

- 0 (default): Packets are received via the standard Linux interface. Any driver can use this mode.
- 1: Packets are memcpy'ed to PF_RING and also to the standard Linux path. Both vanilla and PF_RING-aware drivers can use this mode.
- 2: Packets are ONLY memcpy'ed to PF_RING and not to the standard Linux path. Only PF_RING-aware drivers can use this mode.

IMPORTANT: Do NOT use transparent_mode 1 and 2 with vanilla drivers as it will result in no packet capture.

Other available insmod parameters are:

- *min_num_slots*: Min number of ring slots (default: 4096)
- *enable_tx_capture*: 1 to capture outgoing packets, 0 otherwise (default: 1)
- *enable_ip_defrag*: 1 to enable IP defragmentation, 0 otherwise (default 0)
- *quick_mode*: 1 to run at full speed but with up to one socket per interface (default: 0)

Example loading PF_RING and the PF_RING-aware igb driver:

```
cd <PF_RING PATH>/kernel
insmod pf_ring.ko transparent_mode=2 min_num_slot=8192
enable_tx_capture=0 quick_mode=1

cd <PF_RING PATH>/drivers/PF_RING_aware/intel/igb/igb-X.X.X/src
make
insmod igb.ko
```

DNA drivers

If you want to achieve line-rate packet capture even at 10 Gigabit, you should use these drivers. DNA drivers are part of the PF_RING distribution and can be found in "<PF_RING PATH>/drivers/DNA/".

Currently available DNA drivers are:

- e1000
- e1000e
- igb
- ixgbe

Please note that:

- the PF_RING kernel module must be loaded before the DNA driver
- in order to correctly configure the device, it is highly recommended to use the *load_dna_driver.sh* script provided with the drivers (take a look at the script to fine-tune the configuration)

Example loading PF_RING and the ixgbe-DNA driver:

```
cd <PF_RING PATH>/kernel
insmod pf_ring.ko

cd PF_RING/drivers/DNA/ixgbe-X.X.X-DNA/src
make
./load_dna_driver.sh
```

Appendix B
BPF-Like Packet Filtering Expressions

BPF-like filters can be specified using a subset of the BPF syntax.
As the filter expression complexity affects:

- packet capture speed when used for filtering incoming traffic
- index complexity and speed when used for filtering dumped traffic

we will define a set of constraints and allowed expressions.

An expression consists of one or more primitives.
Complex filter expressions are built by using AND, OR and NOT operators.

Allowed qualifiers for primitive expressions:

Protocol: ether, ip, ip6, tcp, udp, sctp
Direction: src, dst, src or dst, src and dst
Type: host, net, port

Additional constraints for packet capture filters:

- it is not possible to use the NOT operator.
- it is possible to use up to two level of nesting with parenthesis
- on the same nesting level, and inside the same parenthesis, it is not possible to mix different operators

Tricks to speed up index filtering:

- Use /32 and /24 IPv4
- Use /128 only IPv6
- Use exact ports (no port range)
- Do not use protocol identifiers

Primitive filter examples:

```
ether host 00:11:22:33:44:55
ether src host 00:11:22:33:44:55

ip host 192.168.0.1
ip dst host 192.168.0.1

ip6 host 2001:0db8:85a3:0042:0000:8a2e:0370:7334
ip6 src host 2001:0db8:85a3:0042:0000:8a2e:0370:7334

ip net 192.168.1.0/24
ip src net 192.168.1

port 80
udp port 9000
tcp src port 80
```

Complex capture filter examples:

```
ip host 192.168.1.1 and 192.168.1.2
ip src 192.168.1.1 and dst 192.168.1.2

ip host 192.168.1.1 and tcp port (80 or 443)
(ip host 192.168.1.1 or 192.168.1.2) and (port 80 or 443)
```

Appendix C
Disk Partitioning and Formatting

n2disk writes files sequentially. In our experience the XFS filesystem is the best option you can select although other filesystems such as EXT4 can also be used. Supposing that you have created the /dev/sda1 disk partition, you can format it as follows:

```
# mkfs.xfs -f -d sunit=128,swidth=1024,agcount=6000 -l size=64m /dev/sda1
```

Once you have formatted the disk you can mount it as follows:

```
# mount -o
noatime,nodiratime,attr2,nobarrier,logbufs=8,logbsize=256k,osyncisdsync /dev/
sda1 /storage/
```

Note that for small partition sizes, you might need to adapt some of the above parameters.

You can test the disk write performance as explained in chapter 4. Please note that depending on the disk types and controller, your write performance can be influenced by the chunk size (-C option) that specifies. the minimal unit of data written to disk. Values in the range between 64 and 512 KBytes should guaranteed adequate performance, although some combination of disks/controllers operate better with 1024 KBytes or more. You can check that with "-e 1".

Appendix D
Installing n2disk

n2disk needs a license in order to operate permanently (i.e. not in demo mode). In order to obtain a license, you need to go to http://shop.ntop.org and order it. Once you have done that, you need to activate your n2disk license as follows:

1. Go to http://packages.ntop.org and select your platform (currently CentOS and Ubuntu), then download the PF_RING and n2disk packages.
2. Install the packages (as root). First you need to install the PF_RING package then the n2disk package. Example:
   a. Ubuntu
      i. dpkg -i pfring.....deb
      ii. dpkg -i n2disk....deb
   b. CentOS
      i. rpm -i pfring.....deb
      ii. rpm -i n2disk....deb
3. Identity your system Id
   # n2disk -V
   n2disk 2.0 [SystemID: **1234567890-760560b5**]
4. Generate your license using the above system identifier and the credentials you have received.
5. Install the license according to your n2disk version
   a. n2disk /etc/n2disk.license
   b. n2disk5g /etc/n2disk1g.license
   c. n2disk10g /etc/n2disk1g.license

You can now start your licensed n2disk copy.

Remember that for good results, you need to use n2disk over DNA. In order to do that please refer to the PF_RING manual (see http://www.ntop.org/products/pf_ring/).

Appendix E
Hugepages Support

Hugepages can be enabled in n2disk using the -U option. This section describes how to enable hugepages into your system, a mandatory step for running n2disk with hugepages.

For the impatient

In order to reserve and make available 2 GBytes (1024 pages of 2 MBytes) of memory for hugepages without any special NUMA node affinity you can use:

```
echo 1024 > /sys/kernel/mm/hugepages/hugepages-2048kB/nr_hugepages
mount -t hugetlbfs nodev /mnt/huge
```

The whole story

Linux typically uses memory pages of 4 KBytes, but provides an explicit interface to allocate pages with bigger size called hugepages. It is up to developers/administrators to decide when they have to be used.

Hugepages advantages:

1. Large amounts of physical memory can be reserved for memory allocation, that otherwise would fail especially when physically contiguous memory is required.
2. Reduced overhead: as the TLB (Translation Lookaside Buffer) contains per-page virtual to physical address mappings, using a large amount of memory with the default page size leads to processing overhead for managing the TLB entries.

The default hugepage size is usually 2 MBytes. The hugepage size can be found in /proc/meminfo:

```
cat /proc/meminfo | grep Hugepagesize
  Hugepagesize: 2048 kB
```

Hugepages can be dinamically reserved with:

```
echo 1024 > /sys/kernel/mm/hugepages/hugepages-2048kB/nr_hugepages
```

The above pages are allocated by the system without node affinity. If you want to force allocation on a specific NUMA node you have to do:

```
echo 1024 > /sys/devices/system/node/node0/hugepages/
hugepages-2048kB/nr_hugepages
echo 1024 > /sys/devices/system/node/node1/hugepages/
hugepages-2048kB/nr_hugepages
```

It is possible to change the default hugepages size and reserve large amounts of memory at boot time using the following kernel options:

```
default_hugepagesz=1G hugepagesz=1G hugepages=4
```

If this commands returns a non-empty string, 2MB pages are supported:

```
cat /proc/cpuinfo | grep pse
```

If this commands returns a non-empty string, 1GB pages are supported:

```
cat /proc/cpuinfo | grep pdpe1gb
```

In order to make the hugepages available for use, Linux provides a RAM-based filesystem called "hugetlbfs" that have to be mount'ed with:

```
mount -t hugetlbfs none /mnt/hugepages
```

With no options the default hugepage size is used. To use a different size it is possible to specify the "pagesize=" option. In order to control the maximum amount of memory bound to a mount point it is possible to specify the "size=" option (size is rounded down to the nearest hugepage size).
Example:

```
mount -t hugetlbfs -o pagesize=1G,size=2G none /mnt/hugepages
```

It is possible to see what pages are currently in use using the following command:

```
cat /sys/devices/system/node/node*/meminfo | grep Huge
Node 0 HugePages_Total:  1024
Node 0 HugePages_Free:   1024
Node 0 HugePages_Surp:      0
```

Appendix F
n2disk License

NTOP END USER LICENSE AGREEMENT

THIS "END USER LICENSE AGREEMENT" ("EULA") IS A BINDING LEGAL AGREEMENT. BY USING THE "SOFTWARES PROVIDED TOGETHER WITH THIS EULA" (THE "SOFTWARE"), OR USING ANY AUTHORIZATION CODE PROVIDED BY THE LICENSOR, YOU ACKNOWLEDGE THAT YOU HAVE READ THIS EULA, THAT YOU UNDERSTAND IT, AND THAT YOU AGREE TO BE BOUND BY ITS TERMS. IF YOU DO NOT AGREE WITH THE TERMS AND CONDITIONS OF THIS EULA, DO NOT MAKE ANY USE OF THE SOFTWARE OR ANY USE OF AUTHORIZATION CODES. YOUR USE OF THE SOFTWARE AND YOUR USE OF ANY AUTHORIZATION CODE IS SUBJECT AT ALL TIMES TO THE TERMS AND CONDITIONS HERE, INCLUDING ALL PROVISIONS REGARDING THE LIMITATIONS OF THE LICENSOR LIABILITY.

1. Grant of License in Favour of Registered Users. "NTOP di Deri Luca" ("Licensor") grants you a non-exclusive license to use the Software (including any updates of the Software that the Licensor may make available to you at his own discretion) and any "documentation files pertaining to the Software" ("Documentation"), only in connection with a single hardware unit with a unique "Media Access Control" ("MAC") address or unique system identifier of any kind. All copyright notices in the Software and Documentation must be retained at all times. The Software and Documentation shall be used only for your own personal, non-commercial use and not for the benefit of any other person or entity. You may also make one copy of the Software in machine-readable form only for back-up purposes, provided you properly reproduce also both any Licensor's copyright notices/credits and any proprietary legends/ disclaimers.

2. IP Ownership. All the intellectual property rights pertaining to Software and Documentation shall be reserved to the Licensor, at all times. Provided the Section 1 above, any other use of the Software by any person or entity of any kind is strictly forbidden and is a violation of this EULA. All the rights not expressly granted to you herein are reserved to the Licensor. You are not allowed to remove any Licensor's copyright notices/credits and any proprietary legends/disclaimers from any copy of Software, Documentation or any component thereof.

3. Restrictions. You are not allowed to or permit/assist any third party to: (a) publish, display, disclose, rent, lease, modify, copy, loan, distribute, or create derivative works based on the Software or any part thereof; (b) reverse engineer, decompile, translate, adapt, or disassemble the Software or any part thereof; (c) attempt to create or otherwise reproduce in any form the source code from the object code of any portion or component of the licensed Software; (d) sublicense the Software or permit the exploitation of the Software by more than a single hardware unit with a unique MAC address or unique system identifier of any kind; (e) attempt to disable or circumvent any technological protection measure of the Software or assist third parties to do so.

4. Confidentiality. You agree to maintain the confidentiality of Software and Documentation: you will not disclose Software or Documentation, or any information or materials related thereto, to any third party without the express written consent of NTOP di Deri Luca. You further agree to take all reasonable measures to limit access to Software and Documentation only to those of your employees who reasonably require such access to perform their employment obligations and who are bound by confidentiality agreements with you (in order to maintain the confidentiality of Software and Documentation).

5. No Warranty. Software is provided "as it is". To the maximum extent permitted by applicable laws, Licensor disclaims warranties of any kind, either explicit or implied, including, without limitation, implied warranties of merchantability and fitness for a particular purpose. Licensor does not warrant that the functions contained in the Software will meet any requirements or needs you may have, or that the Software will operate error free, or in an uninterrupted fashion, or that any defects or errors in the Software will be corrected, or that the Software is compatible with any particular platform. Licensor is not obligated to provide any updates to the Software.

6. Limitation of Liability. To the extent not prohibited by applicable laws, in no event shall the Licensor be liable to you or any third party for any incidental or consequential damages (including - without limitation - indirect, special, punitive or exemplary damages for loss of business, loss of profits, business interruption or loss of business information) arising out of or related to the use of or the inability to use the Software, or for any claim by any other party, even if the Licensor has been advised of the possibility of such damages. In no event shall Licensor's total liability to you for all damages exceed the amount of the license fee paid by you to Licensor for Software and Documentation.

7. Export Restrictions. You may not export Software and Documentation in violation of applicable laws and regulations.

8. High Risk Activities. The licensed Software is not fault-tolerant and is not designed, manufactured or intended for any kind of use with on-line control equipment in hazardous environments requiring fail-safe performance (such as in the operation of nuclear facilities, aircraft navigation or communication systems, air traffic control, direct life support machines or weapon systems in which the failure of the Software could lead directly to death, personal injury or severe physical or environmental damage: all the so called "High Risk Activities"). Accordingly, Licensor specifically disclaims any express or implied warranty of fitness for High Risk Activities.

9. Ethics. The licensor commits itself to use the Software in compliance with all applicable local, national and international laws, rules and regulations, including any laws regarding the transmission of technical data exported from its country of residence. In no case the licensed Software can be used to track, spy, intercept or collect evidence of network communications to be used against individuals or organizations, to prosecute individuals or organizations, or to restrict their freedom.

10. Termination. Licensor may terminate this Agreement at any time if you violate its terms. Upon termination, you must immediately destroy or return to Licensor the Software and Documentation. The provisions of Sections 2 (IP Ownership), 3 (Restrictions), 4 (Confidentiality), 5 (No Warranty), 6 (Limitation of Liability) and the provisions of this Section 9 (Termination) shall in any case survive the termination or expiration hereof.

11. General Provisions, Governing Law, Jurisdiction. This EULA shall be governed by, construed and interpreted in accordance with Italian Law. The licensee agrees that any dispute arising from or connected to this EULA shall be submitted to the exclusive jurisdiction of the Italian Specialized IP Courts. Therefore, the jurisdiction of any other court is expressly excluded. This EULA shall constitute the entire agreement between the parties; any waiver or modification of this EULA shall be effective only if it is in writing and signed by both parties. Should any part of this EULA be found invalid or unenforceable by an Italian Specialized IP Court, the remainder of this EULA shall be interpreted so as to reasonably effect the intention of the parties.

EACH PARTY IS HEREBY CONFIRMING ITS AGREEMENT WITH THE FOREGOING BY SIGNING AND

RETURNING ONE COPY OF THIS EULA TO THE OTHER PARTY.

PURSUANT TO ARTICLES 1341 AND 1342 OF THE ITALIAN CIVIL CODE, THE PARTIES HEREBY ACKNOWLEDGE AND EXPRESSLY APPROVE SECTIONS 1 (Grant of License in Favour of Registered Users), 2 (IP Ownership), 3 (Restrictions), 5 (No Warranty), 6 (Limitation of Liability), 8 (High Risk Activities), 9 (Termination) AND 10 (General Provisions, Governing Law, Jurisdiction).