



TEHUTI NETWORKS

PERFORMANCE TUNING GUIDE

Ver 1.4

Trademarks

OptiStrata is a trademark of Tehuti Networks, Ltd.

Windows is a trademark of Microsoft.

Any other trademarks or registered trademarks are the property of their respective owners.

Disclaimer

This document contains information that may be privileged or confidential and is the property of Tehuti Networks Ltd.

This document is only intended for the person to whom it is addressed, therefore if you are not the intended recipient, please note that you are not authorized to read, print, retain, copy disseminate, distribute, or use this document or any part thereof. If you received this document in error, kindly notify the sender immediately and delete all copies of this document.

No license, express or implied to any intellectual property rights, is granted by this document. While every attempt has been made to assure that the information presented in this document is accurate, Tehuti Networks, Ltd assumed no liability whatsoever relating to any possible inaccuracies. Tehuti Networks, Ltd assumes no liability whatsoever relating to fitness for a particular purpose, merchantability, or infringement of any patent, copyright, or other intellectual property right.

Tehuti Networks, Inc. reserves the right to make changes to specifications and product descriptions at any time, without notice.

Table of Contents

1. Overview	3
2. Recommended HW	4
3. Optimized Performance Tuning	5
3.1. Preinstalled Systems.....	5
3.2. OS Optimization.....	6
3.2.1. <i>Windows OS</i>	6
3.2.2. <i>Linux OS</i>	6
4. Benchmark Performance Tuning	7
4.1. OS Optimization.....	7
4.1.1. <i>Windows OS</i>	7
4.1.2. <i>Linux OS</i>	9

Revision History

Version	Date	Owner	Description
1.0	30 May 2013	M.G.	HW minimum requirements and Windows OS optimization for benchmark testing
1.1	30 May 2013	M.G.	Preinstalled systems usage guidelines added
1.2	06 Oct 2013	M.G.	Xeon CPU Machines tuning
1.3	16 June 2015	G.K.	Xeon CPU Machines tuning clarification
1.4	23 March 2016	G.K.	Adding OSx Tuning performance with Server App

1. Overview

Modern systems provide various options for performance optimization by system administrator or home user.

Default system optimization is consistent with general home/small office user needs, but could be insufficient for advanced user or benchmark testing. Specifically, Windows OS is shipped with defaults best suited for 1GbE Networks.

This document intends to provide information for tuning Windows and Linux OS and Driver for best Networking performance.

2. Recommended HW

Tehuti Networks NICs are capable of high performance on any modern PC or server.

Tehuti recommends following Minimum Configuration for optimal results:

- Motherboard supporting:
 - At least one PCIe Gen2 x4(or more) slot
 - Chipset with native support for the installed CPU. For example:
 - H61, H67 or Z68 from B3 stepping or better for Sandy Bridge CPUs
 - B75, H77 or Z77 for Ivy Bridge CPUs
 - Dual Channel memory supported with at least 2 DIMMs (1 for each channel)
- Memory: DDR3 Memory running at 667Mhz (DDR1333), one DIMM per channel populated
- CPU: Sandy Bridge i7 or better. Ivy Bridge i5 or better
- Load BIOS updated and “Optimized Defaults” loaded

3. Optimized Performance Tuning

This part overviews possible Systems and OS configuration options for general user to achieve best performance in common LANs with common traffic.

3.1. Preinstalled Systems

All technical information in following parts is optimized for new installed clean systems. It is approved standard installation of Windows or Linux OS provides full performance.

Some systems historically include specific HW and SW changes which could influence maximum performance of NICs.

Here are a few examples of HW and SW should be checked to increase performance:

- Check if currently installed HW is according to “Recommended HW” list above
 - Latest version of BIOS is recommended for use with “Optimized Defaults” loaded
 - It could be that RAM DIMMs were changed during system live time
 - It could be that same MB supports faster memory
- High CPU usage might degrade traffic
 - Check if specific cores used by NIC for traffic processing also used by other processes
 - Tehuti Networks Driver includes CPU load balancing with optional configurations. System admin could change CPU cores used by the NIC to allow high throughput
- System Admin should check if any Network applications were installed, which could have degradation influence, such as:
 - Wireshark or other network tools
 - Access from remote SW tools
- Check if TCP stack was previously changed from original code or configuration
 - Old DLLs could influence OS behavior
 - Some NICs with full offload support could change it.

- Please advise with those NIC vendors to reverse it to standard module/configuration
-
- Firewall and Antivirus influence
 - Firewalls and Antiviruses check different layers of traffic, therefore it is known it could degraded performance
 - Please advise with your Firewall/Antivirus provider for best throughput configurations

3.2. OS Optimization

3.2.1. Windows OS

Make sure latest updates for Windows are installed.

Microsoft provides Performance Tuning Guidelines according to clients target server usage. Please follow recommendations:

<http://msdn.microsoft.com/en-us/windows/hardware/gg463392.aspx>

3.2.1.1. TCP/IP Tuning

Default TCP/IP optimization for Windows is capable of wire network performance for limited time. Usually TCP connections are limited in time and do not live for more than a few hours. Default Windows TCP/IP stack optimized for long distance media connections.

For long time LAN TCP/IP connection please use Benchmark part of this document for proper OS configuration.

3.2.1.2. RSS

RSS is a mechanism for distributed received traffic across available processors or cores. At least 4 cores must be activated for full bidirectional 10GbE traffic

3.2.2. Linux OS

Default Linux OS optimize IPv4 traffic for best user experience. TCP/IP traffic usually stable for a long period of times and no optimization required. Some distributions could enable different IRQ SMP affinity methods. See [IRQ SMP affinity for Benchmarks](#).

4. Benchmark Performance Tuning

This part describes HW and SW optimization for benchmark testing.

Benchmarks test, contrary to regular traffic, is of a single type at a time and all resources should be applied to accelerate this specific traffic type being tested.

4.1. OS Optimization

4.1.1. Windows OS

For best results, it is recommended that a fresh copy will be installed and all updates applied.

Microsoft provides documentation for high performance testing:

<http://msdn.microsoft.com/en-us/windows/hardware/gg463392.aspx>
(Performance Tuning for Network Workload part)

<http://www.microsoft.com/whdc/system/sysperf/Win7Perf.msp>

(Recommended Test Methodology part)

For stressful benchmark testing administrator might optimize configurations parameters.

All commands should be executed in CMD windows opened with “Run as administrator”.

4.1.1.1. Power Plan

Configuring “High Performance” power plan to be used is best practice for stress test. High performance power plan would verify CPU is working on maximum available frequency during all times.

4.1.1.2. Filtering

It is recommended to disable windows Firewall and Windows Base Filtering Engine (BFE). If firewall is disabled BFE still could have negative influence on CPU performance therefore BFE service recommended to be stopped.

Command prompt command:

```
net stop bfe
```

<https://www.myricom.com/software/myri10ge/392-how-do-i-troubleshoot-slow-myri10ge-or-mx-10g-performance.html#windows>

4.1.1.3. RSS

High stress receive traffic requires parallel processing by number of CPU cores. RSS functionality is responsible for correct affinity of traffic to CPU cores.

Make sure RSS is enabled in Windows OS and in NIC Property pages.

Current status:

```
netsh interface tcp show global
```

Windows OS RSS configuration from Command line:

```
netsh interface tcp set global rss=enabled
```

Driver Advanced Properties Tab:

Use different RSS parameter to define RSS state and cores to be used for RX traffic processing.

http://www.google.co.il/url?sa=t&rct=j&q=mellanox%20performance%20tuning%20guide&source=web&cd=1&cad=rja&sqi=2&ved=0CCkQFjAA&url=http%3A%2F%2Fwww.mellanox.com%2Frelated-docs%2Fprod_software%2FPerformance_Tuning_Guide_for_Mellanox_Network_Adapters.pdf&ei=gG-kUa7rjMmp4gSEi4DICQ&usq=AFQjCNHJ4hTXjyWj17nH0Y6VpAFf2dfNEQ (Ver 1.7 part 4.5.2 - Enable RSS)

4.1.1.4. XEON Machine with more than one CPU

For best performance on Xeon under Windows, we set as default the following values under the driver Advanced property

“Maximum Number of RSS Queues” set to 4

“The maximum processor number of the RSS interface” set to 4

“The maximum number of RSS processors” set to 4

You should not change these defaults on single CPU systems.

4.1.1.5. TCP/IP full offload

Tehuti NIC does not use Windows Chimney interface, therefore it would be best practice to disable it for stressful testing:

Command prompt:

```
netsh interface tcp set global chimney=disabled
```

4.1.1.6. Autotuning

Windows OS start from Vista flavor introduced new TCP management option called autotuning. This configuration, which is enabled by default in modern systems, gives Windows OS a permission to automatically manage TCP connection parameters for better long range performance.

For high stress test administrator should disable it

Command Prompt:

```
netsh interface tcp set global autotuninglevel=disabled
```

<http://www.speedguide.net/articles/windows-7-vista-2008-tweaks-2574>

4.1.1.7. Network Throttling Index

Windows OS implements a restriction mechanism for processing non-media network traffic.

Following registry key controls this index:

```
HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows  
NT\CurrentVersion\Multimedia\SystemProfile  
NetworkThrottlingIndex=ffffff (DWORD)
```

Recommendation is to change default 0x10 to 0xFFFF_FFFF for maximum throughput.

<http://www.speedguide.net/articles/windows-7-vista-2008-tweaks-2574>

4.1.1.8. Benchmark Testing With Iperf

Iperf is free tool for benchmark testing. Though it is well known, Iperf has many capabilities.

Iperf version 1.7.0 is preferred for testing.

For best benchmark throughput result following test steps and parameters should be applied:

- On receive side machine run:

```
iperf -s -l1M -w64K -i5
```
- On Transmit side machine run:

```
iperf -c <receive_side_ip> -t<num_of_sec> -P16 -i5 -l1M -w64K
```

Note: Command should be executed from Command Prompt.

4.1.2. Linux OS

For best results, it is recommended that a fresh copy will be installed and all updates applied.

4.1.2.1. Firewall

For uninterrupted benchmark testing disabling Firewall would be the best practice.

Please follow your Linux flavor documentation to shut down Firewall.

4.1.2.2. SMP IRQ Affinity

Tehuti Networks devices uses Round Robin interrupt allocation method.

Administrator should assure at least 4 cores active to service Tehuti Networks NIC interrupts.

```
# echo <hex_bit_mask> > /proc/irq/40/smp_affinity
```

Note: Administrator should correctly define bit mask. For example: 00000F

Following link provides more information regarding IRQ SMP affinity:

<https://access.redhat.com/site/solutions/15482#4> Determine the correct IRQ SMP affinity mask

4.1.2.3. XEON machines

Servers with XEON servers could require addition tuning. Some tuning option are executed once and are valid until changed. Another are valid until system reboot.

Once time configurations:

1. `service irqbalance stop`
2. `tuned-adm profile throughput-performance`

Configuration options have to be executed after each start

1. `echo f > /proc/irq/<IRQ_Number>/smp_affinity` (the IRQ number may differ)
2. `echo f > /sys/class/net/<ETH-X>/queues/rx-0/rps_cpus` (the eth number may differ)

4.1.2.4. Benchmark Testing With Iperf

Iperf is free tool for benchmark testing. Though it is well known, Iperf has many capabilities.

Iperf version 2.0.5-2 is preferred for testing.

For best benchmark throughput result following test steps and parameters should be applied:

- On receive side machine run:
`iperf -s -l1M -w64K -i5`

- On Transmit side machine run:

```
lperf -c <receive_side_ip> -t<num_of_sec> -P16 -i5 -l1M -w64K
```

Note: Command should be executed from Command Prompt.

4.1.3. OS X

Installation of 'OS X Server' application can cause a performance degradation.

Please follow below instruction to Increase performance

- Comment out ('#') the following lines in /etc/rc.server file
 - ```
if [-f /Library/Server/Firewall/Anchors/combined_anchor.txt]; then
 pfctl -ef /Library/Server/Firewall/Anchors/combined_anchor.txt
fi
```
  - ```
RCServerTunings="/Applications/Server.app/Contents/ServerRoot/usr/libexec/RCServerTunings.sh"  
if [ -x ${RCServerTunings} ];then  
    ${RCServerTunings}  
fi
```