StarWind HCA + Mellanox WJH:

Providing Paramount Insights into Hyperconverged Networking Stacks

1. Overview

Businesses today are facing a rapid influx of data and unprecedented demand for faster and better services calls as their customer base and service offerings expand. To remain successful, many companies are building advanced IT infrastructures and placing cost management at the frontline, while shying away from the profit-maximization model. In parallel, continuity and sustainability have become new global business trends, and hyperconverged infrastructure (HCI) has emerged to support them.

HCI combines compute, storage, and network functions in a single turnkey solution, to reduce data center complexity and increase scalability. Its "functions are built in a software layer that covers not only storage and data management but can also run the virtual server instance." Data that flows through a hyperconverged infrastructure combining storage and compute network resources, creates a large amount of data; this is further exacerbated by an increase in mission-critical workloads running on faster and denser HCIs.

The term "telemetry" refers to "automated communications process by which measurements are made and data collected at remote points and are subsequently transmitted to receiving equipment for monitoring." Hyperconverged networking stacks continuously generate telemetry data that "tends to be messy, comes in no particular order, and can be incomplete or erroneous due to transmission problems or [receiver] malfunctions." Since even small HCI deployments may generate petabytes of data daily, legacy network monitoring approaches such as sFlow sampling and SNMP polling, are insufficient for the transportation, processing, and storage of telemetry data.

StartWind HyperConverged Appliance (HCA) simplifies the management of the data and increases the utilization of the entire infrastructure. StarWind HCA employs ProActive Premium support technology, which mitigates downtime and simplifies hardware, storage, and application diagnostics. To provide the best network visibility for minimizing downtime and optimizing management efforts, StarWind has tested the advanced network telemetry technology, "What Just Happened™" (WJH) from Mellanox, on its HCA with ProActive Premium Support. Adding Mellanox's WJH to StarWind HCA enables users to receive network insights in real time. This article reports the findings of how Mellanox WJH resolves the pains of contemporary streaming telemetry collection and management.

2. Mellanox Optimizes Telemetry Data Gathering

2.1. Why Mellanox?

Since 1999, Mellanox Technologies has grown to become the leading developer of end-to-end Ethernet and InfiniBand intelligent interconnect solutions. Offering the lowest latency and highest throughput for servers, storage, and HCIs, Mellanox solutions have been internationally acknowledged with multiple awards, including The Linley Group's Analyst Choice Award for "Best Networking Chip" for ConnectX-5 and the Annual Supercomputing Conference's Award for "Best HPC Interconnect Product or Technology."

Unlike traditional infrastructures that maintain separated compute, storage, and networking, HCI converges all the components in a single infrastructure, including fully virtualized networking functions that are shared for different tasks. However, this makes network management and monitoring all the more challenging. Mellanox's What Just Happened (WJH) goes well beyond conventional streaming telemetry and INT offerings. WJH is "an intelligent network telemetry tool that provides visibility into data plane anomalies" in any IT environment. WJH makes network administrators lives that much easier by allowing them to easily monitor and diagnose the network in her datacenter.

2.2. What Just Happened — Present-Day Hero in HCI Monitoring

Mellanox What Just Happened (WJH) is an advanced streaming telemetry technology that provides network administrators with "real time visibility into the network […] by providing actionable details on abnormal network behavior." Removing the guess work from network troubleshooting, WJH inspects packets across all ports at in-line rate and at multi-terabit speeds—faster than hardware- or software-based solutions. As WJH is native to Open Ethernet, it is integratable into either open source tools or turnkey data centers.

Using the firmware capabilities of Mellanox Spectrum® and Spectrum®-2 switch ASICs, the WJH agent collects rich insights on a variety of areas, including Layer-1 through Layer-4, ACLs and Buffer occupancy. The biggest highlight is being able to configure critical data alarms, bolstering immediate problem-solving. With WJH, there's no data sifting: the switch platform performs traffic inspection, filtering, and issue identification, without any trivial license-per-feature requirements. In short, WJH streams out only relevant data to alarm you about any issues you've pre-configured it to identify.

WJH reduces mean time to innocence (MTTI) and lets you gather particular insights to help improve IT resources utilization and capacity planning. It eliminates the old-fashioned problem of dealing with gargantuan amounts of all "data that is mined from network port counters, statistical packet sampling and hop-by-hop INT." There's no organization or context in such telemetry, which typically results in root-cause identification difficulties, downtime, and monetary costs. Mellanox WJH eliminates all of these headaches. Now let's look under the hood.

3. What Just Happened™ Up-Close

3.1. WJH Offers Details on Abnormal Network Behavior

Conventional telemetry approaches work by gathering data without categorization and flooding it to a unified data center. However, the old-fashioned way of root-causing network issues by analyzing network counters and packet sampling becomes useless in hyperconverged networking stacks. To overcome this challenge, Mellanox WJH provides an event-based telemetry approach where telemetry data can be categorized per user configuration to monitor anomalies in the hyperconverged network. Running on Mellanox Spectrum switches, WJH collects rich, contextual and actionable telemetry data concerning events of interest, and slows rapid problem-solving in cloud applications and business operations.

The collected data is available through a command-line interface. Alternatively, the data can be accessed via a Web UI or Mellanox NEO®. Moreover, it's possible to stream the gathered data out of the switch via gRPC in JSON format using a containerized streaming agent (WJH Telemetry Agent). "Open-source time series DB such as InfluxDB and visualization tools such as Grafana can be used to [neatly] visualize the data." Choosing the style of storing and managing WJH telemetry is flexible.

3.2. Complex Telemetry Solution with Easy Configuration

The main goal of WJH is to provide fast and effective troubleshooting that enables orchestrating smoothly running environments without performance degradation. Traditional telemetry management approaches force users to try and reproduce certain abnormal behaviors to figure out the issue's root cause (apart from frantically analyzing endless packets) — more like "double-trouble-shooting." The Mellanox solution offers a superior alternative that increases data center uptime and crash avoidance, and prevents the loss of millions of USD in annual profits.

**For details on installation and configuration, check the Appendix at the end of the article.*

4. In Sync: Mellanox WJH and StarWind HCA

4.1. StarWind Architecture for Telemetry Gathering

Enterprises and businesses require the critical ability to survey the entire picture of what's going on within their hyperconverged network to avoid any downtime and, respectively, financial losses. With StarWind HCA, customers can enjoy both the convenience of monitoring the state of their HCI from a convenient unified console and data categorization. The integration of WJH into StarWind's HCI monitoring and troubleshooting routine provides engineers with advanced noise filtering and time-saving troubleshooting, which increases work efficiency and precision.

The typical WJH configuration consists of two elements:

- Docker container with an agent that collects data from a designated switch;
- Docker container with a stack of processing and visualizing tools — TIG (Telegraph, InfluxDB, and Grafana) — for data collected from a VM or a server.

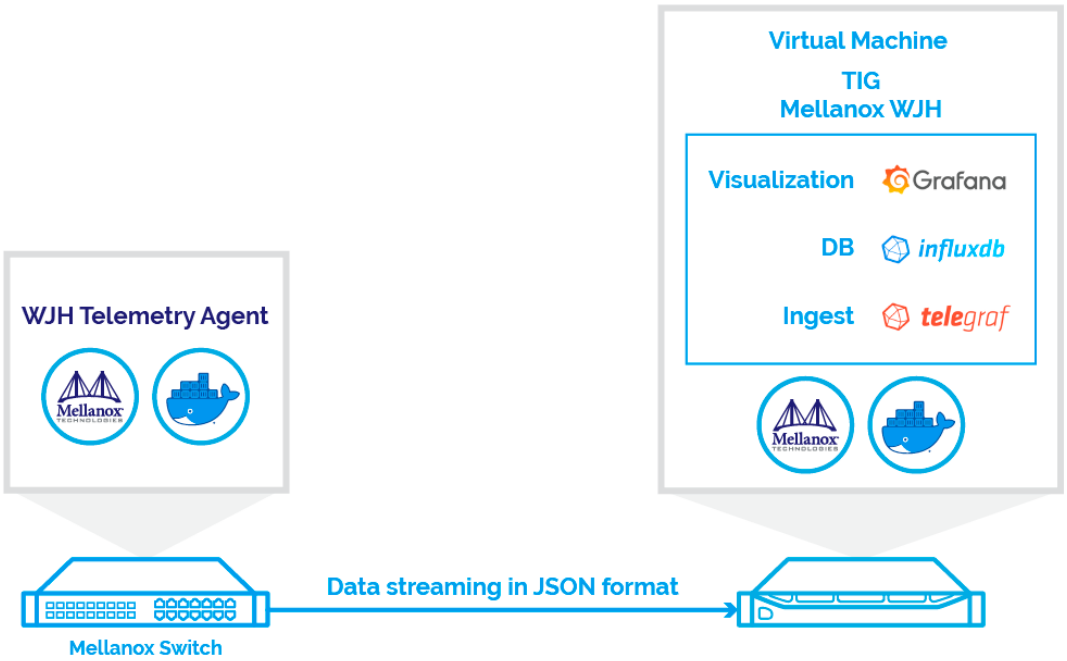


Figure 1: Typical WJH Configuration

4.2. StarWind ProActive Support Way of Using WJH Generated Data

By integrating WJH, StarWind enables engineers to receive alarms from the tool automatically. WJH surveys the hyperconverged network through a lens of dozens of configurable parameters. If it detects any abnormal behavior, WJH immediately generates a detailed description of what's been going on. No other telemetry technology offers such kind of elaboration.

WJH enables the flexible adjustment of triggers, causing different events to erupt during various crashes or malfunctions. That being said, the WJH-enhanced system doesn't simply provide equipment alarms that startle the network administrator, but rather, it provides a comprehensive picture pertinent to the transpired event.

The administrator instantaneously receives complete details of what just happened during a failure. The StarWind HCA technology allows StarWind ProActive Support Engineers to effectively monitor and prevent issues from recurring, but with WJH, the process is now more informed and takes split seconds. There's no need to shuffle through endless logs, monitor the packets, or check the ports anymore — the tool does it all. You just have to adopt an informed decision, and remove the root cause and its aftermaths.

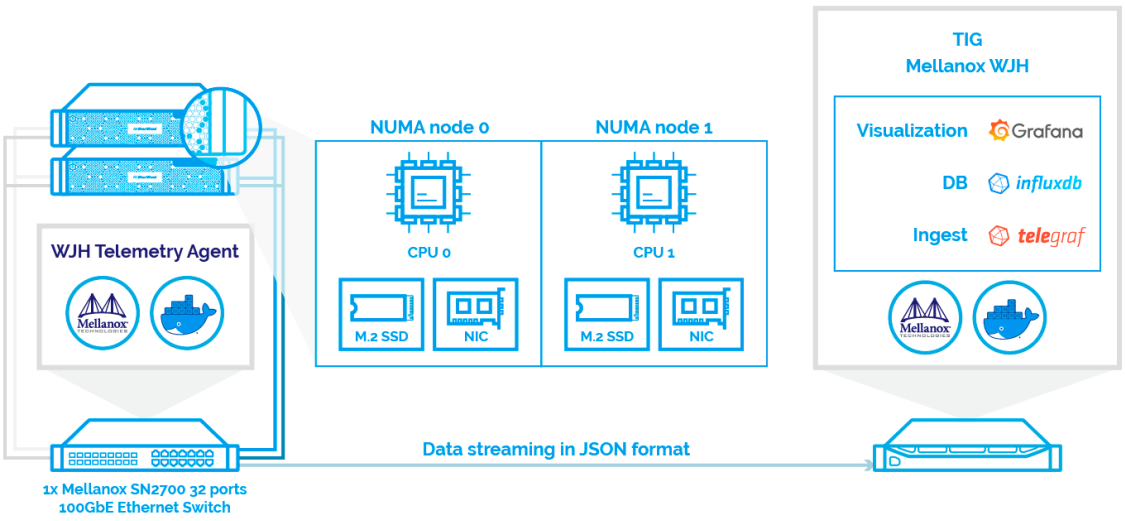
4.3. StarWind RoCE Configuration and Troubleshooting with WJH

To experience the benefits of Mellanox WJH, we used a setup that was built on StarWind HCA coupled with solutions from Intel, Mellanox, and SuperMicro:

- A storage sub-ssytem dependent on StarWind Virtual SAN
- 100GbE host connectivity based on two Mellanox ConnectX-5 NICs on each node
- The traffic was routed by one Mellanox Spectrum-based SN2700 32-Port 100GbE Ethernet Switch

The hardware compilation had the following specifications:

- Platform: Supermicro SuperServer 2029UZ-TR4+
- CPU: 2x Intel® Xeon® Platinum 8268 Processor 2.90 GHz. Intel® Turbo Boost ON, Intel® Hyper-Threading ON
- RAM: 96GB
- Storage: 2x Intel® SSD D3-S4510 Series (240GB, M.2 80mm SATA 6Gb/s, 3D2, TLC)
- Networking: 2x Mellanox ConnectX-5 MCX516A-CCAT 100GbE Dual-Port NIC
- Switch: 1x Mellanox SN2700 32 Spectrum ports 100GbE Ethernet Switch



2-node StarWind HyperConverged Appliance

Platform
Supermicro SuperServer 2029UZ-TR4+

CPU
2x Intel® Xeon® Platinum 8268 Processor 2.90 GHz. Intel® Turbo Boost ON, Intel® Hyper-Threading ON

RAM
96GB

Storage Capacity
2x Intel® SSD D3-S4510 Series (240GB, M.2 80mm SATA 6Gb/s, 3D2, TLC)

Networking
2x Mellanox ConnectX-5 MCX516A-CCAT 100GbE Dual-Port NIC

Switch
1x Mellanox SN2700 32 Spectrum ports 100GbE Ethernet Switch

Figure 2: Benchmark Interconnect/Hardware Schematics

The Mellanox switch has a docker running a WJH container (Telemetry Agent). The Telemetry Agent (https://github.com/Mellanox/wjhgraf) forwards telemetry, based on a preconfigured timeout, to a deployed server that collects and manages the stats with a TIG stack (Telegraph, InfluxDB, and Grafana).

5.0 The TIG WJH Container

The Telegraph, InfluxDb and Grafana (TIG) stack consists of three components and is used to survey and analyze the telemetry data. The below solution is based on the TIG components installed in a single container.

The **TIG** elements include:

Telegraph — a tool that collects and forwards incoming data with a distinctive format, to InfluxDB

InfluxDB — database where telemetry is stored, e.g. discarded WJH packets

Grafana — graphical user interface that visualizes the telemetry collected by InfluxDB

StarWind VSAN deployed on both servers, creating a 2-node HA configuration.

A 200 GB test target was created and synchronized.

1st server 172.27.31.105

2nd server 172.27.31.106

To demonstrate the potential of WJH are two network use cases that may occur within an HCI's network infrastructure.

5.1 Broken Connection

A broken connection may happen due to physical circumstances, hardware failure or human factors. We can simulate the test case by breaking one of the connections between the server and the switch. Switching off the network interface in the server control panel enables us to observe what WJH tells us about the connection or given task. The StarWind 2HA continues to function, demonstrating its fault tolerance in case one of the hosts fail.

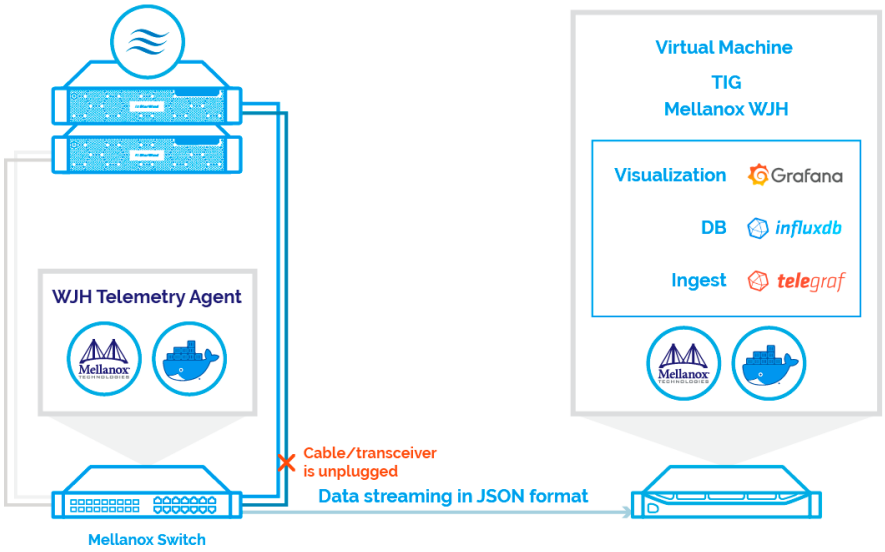


Figure 3: Broken Connection C\ase Schematics

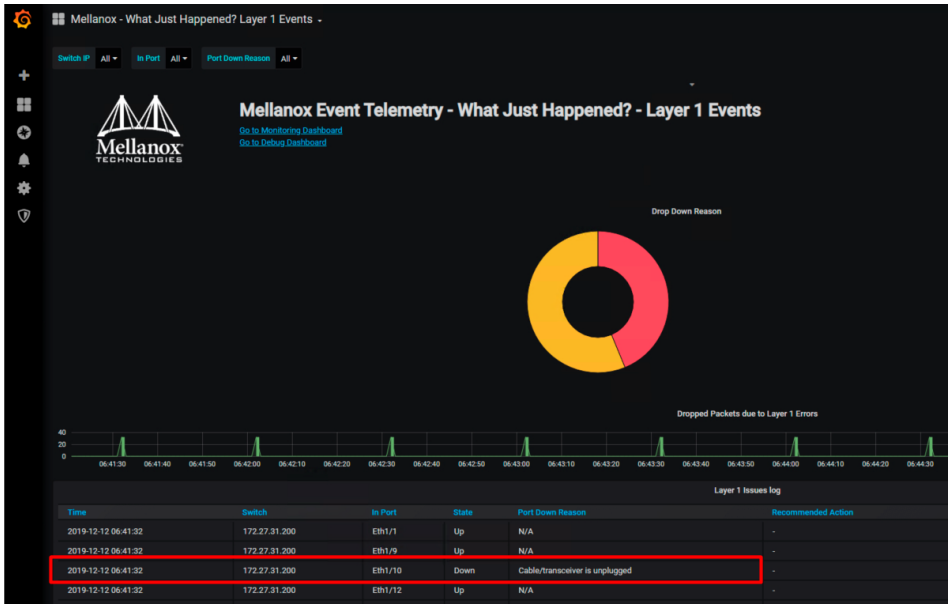


Figure 4: Broken Connection Error Dashboard Notification

5. 2 WJH reacted flawlessly.

The notification about the broken connection appeared immediately on the Dashboard.

1. WJH indicated the port — port 10 in this case
2. The reason — broken connection
3. The context — connection failure between the server and the switch.

WJH provided us with exhaustive information about the current event, allowing us to remove the cause in no time.

5. 3 MTU Mismatch and Consequential Network Packet Drop

To see how WJH reacts to a more complicated problem, we incorrectly configure the packet MTU on the switch so that it operates erroneously within the network. Such cases may commonly occur in mid-size or large hyperconverged networks, where it's plausible to use VLAN-based network topology.

After preliminarily creating two virtual networks (VLAN 10 and VLAN 20), we added the respective server ports to them, assigned MTU 4200 to VLAN 10 and MTU 9216 to VLAN 20, and configured the routing between them. The MTU 9216 packets that were routed to the MTU 4200 VLAN 10 needed to be defragmented for further transmission. This triggered one of the WJH Agent's alarms, which was displayed in Grafana.

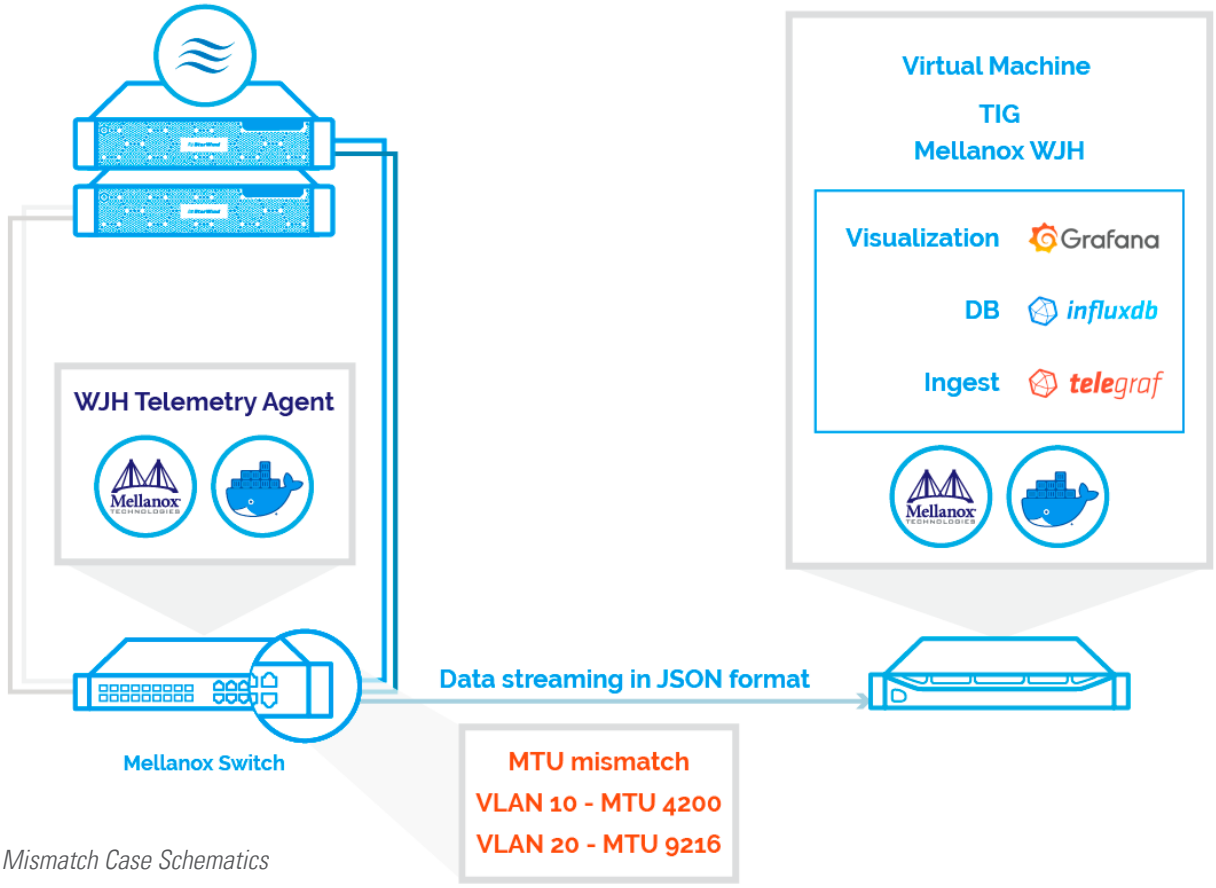


Figure 5: MTU Mismatch Case Schematics

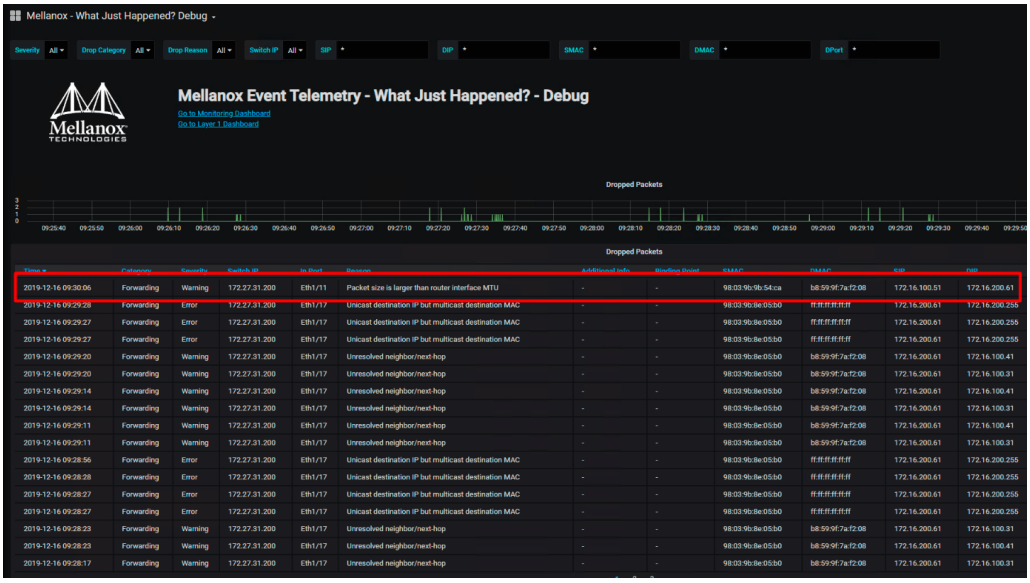


Figure 6: MTU Mismatch Error Dashboard Notification

The WJH system instantaneously sends a notification to the dashboard when a mismatched packet size occurs. Unlike other streaming telemetry tools, WJH uniquely indicates the reason for the drop: the MTU packet size is larger than the interface MTU. In real life this results in connection speed drops or even connectivity errors.

WJH is all about precise and rapid status updates. By looking at the data in the alarm, the engineer understands what's happening to the system exactly. This saves the engineer precious time by not having to find the root cause, which typically, can result from any of a number of interconnected factors. Consequently, as the HCI remains fault-tolerant on the upper level, the engineer can deal with the highlighted issue directly.

An MTU mismatch can be swiftly resolved by reconfiguring the network interface of the switch. It takes just a few minutes to resolve the problem and is seamless to the customer. The reason is that the StarWind 2HA device supports the entire hyperconverged network through its fault tolerance features, while WJH helps the engineer quickly correct the mismatched configurations.

6. Conclusion

In today's world, businesses face an onslaught of data and unprecedented demand for faster and better services calls. As sustainability and continuity their emergence as new global business trends, hyperconverged infrastructure (HCI) frameworks have appeared on the scene to address these issues. HCI combines compute, storage, and network functions in a single turnkey solution. To provide the best network visibility for minimizing downtime and optimizing management efforts, StarWind has tested the advanced network telemetry technology, "What Just Happened™" from Mellanox, on its HyperConverged Appliance (HCA) with ProActive Premium Support. Mellanox WJH resolves the pains of contemporary streaming telemetry collection and management.

About StarWind

StarWind delivers Software-Defined Storage and HyperConverged Infrastructure solutions designed to build high-performing, flexible, and resilient IT infrastructures for SMB and ROBO. Founded in 2009, StarWind has helped to build virtualization infrastructures for over 4,000 paying customers around the world. For more information, visit starwind.com

About Mellanox

Mellanox Technologies is a leading supplier of end-to-end Ethernet interconnect solutions and services for enterprise data centers, Web 2.0, cloud, storage and financial services. More information is available at More information is available at: www.mellanox.com

Appendix. WJH Installation and Configuration

There are three main ways to use WJH:

- Onyx (CLI or Web UI)
- Combined Mellanox Onyx - Mellanox NEO solution
- TIG Stack

Note:

WJH is supported through CLI, WebUI or Mellanox NEO individually but not simultaneously.

The current article outlines installing and working with the TIG stack.

https://docs.mellanox.com/display/TelemetryAgentUMv25/Deployment#Deployment-ISC

https://docs.mellanox.com/display/TelemetryAgentUMv251/WJH+Streaming+and+Integration+with+Telegraf%2C+InfluxDB+and+Grafana+%28TIG%29+Stack

The installation process has three stages:

I. Installing and initiating Docker on the Mellanox switch

II. Deploying the Grafana for WJH Docker

III. Configuring the Telemetry Agent to send data

I. Deploying the Docker Image on Mellanox Onyx-Based Systems

To deploy the Docker image, perform the following:

1. Download the the Telemetry Agent from the Mellanox customer portal at www.mellanox.com/support, and copy it to a remote server.
2. Connect to the Mellanox switch via SSH.
3. Enter the switch CLI mode:

```
switch > enable

switch # configure terminal
```

4. Copy the Docker image from the remote server, for example:

```
switch (config) # image fetch scp://admin:qwerty@10.20.30.100/docker_files/docker_images/

telemetry-agent_<version>.img.gz
```

5. Make sure that the Docker service is running.

```
switch (config) # no docker shutdown
```

6. Load the image, using the docker load <image_name> command:

```
switch (config) # docker load telemetry-agent_<version>.img.gz
```

7. Once the image is copied to the switch, deploy it using the following command:

```
switch (config) # docker start telemtery-agent <version> <container name> now-and-init cpus 0.5
memory 300 privileged network sdk telemetry-agent <version>
```

8. Run the configuration write command:

```
switch (config) # configuration write
```

9. The telemetry agent needs to create trust with the switch in order to get telemetry on LAGs and MLAGs. In order to do this, use the following command:

```
switch (config) # docker exec [docker instance name] " /opt/telemetry/utils/create_trust.sh"
```

10.

a. Copy the key generated and printed on your screen:

```
switch (config) # docker exec neo-agent /opt/telemetry/utils/create_trust.sh
Running exec_name: [/opt/telemetry/utils/create_trust.sh]

Generating public/private rsa key pair.
Crated directory '/root/.ssh'.
Your identification has been saved in /root/.ssh/id rsa.
Your public key has been saved in /root/.ssh/id_ rsa.pub.
The key fingerprint is:
root@switch
The kye's randomart image is:
ssh-rsa Some1Random2Genraced3Key4Wich5Random6Chars7 rooc@swicch
```

b. And run the following command:

```
switch (config) # ssh client user admin authorized-key sshv2 "<paste your key>"
```

11. The Telemetry Agent is waiting for Mellanox SDK installation. Install it, using the following command from the switch prompt:

```
switch (config) # docker
switch (config) # copy-sdk [docker instance name] to /
```

12. Once Mellanox SDK is installed, the Telemetry Agent service should be automatically running on the Docker. In order to verify that the Telemetry Agent is running, do the following:

- Make sure that the Docker has been loaded/started: find your newly created Docker name in the output of the "docker ps" command. If the Docker name exists. Run:

  ```
  switch (config) # docker exec [docker instance name] "/bin/bash"
  ```

- This will bring you into Docker standard Linux prompt. Run:

  ```
  "/etc/init.d/telemetryd status"
  ```

- If service is running, the output should look like the following:

  ```
  #/etc/init.d/telemetryd status Telemetry agent status: Telemetry agent is running
  ```

II. Running the Grafana for WJH Container

Deployment of the Grafana for WJH container should be performed on a Linux host or a VM.

Prerequisites:

4 GB RAM

1 CPU

Docker CE installed: https://docs.docker.com/install/linux/docker-ce/centos/

To run the Grafana for WJH Container:

1. Install Docker CE on CentOS 7.X:

```
yum install -y yum-utils device-mapper-persistent-data lvm2

yum-config-manager --add-repo https://download.docker.com/linux/centos/docker-ce.repo

yum-config-manager --enable docker-ce-edge

yum-config-manager --enable docker-ce-testing

yum makecache fast

yum -y install --setopt=obsoletes=0 docker-ce-17.03.2.ce-1.el7.centos.x86_64 docker-ce-selinux-17.03.2.ce-1.el7.centos.noarch
```

2. Start Docker.

```
service docker start
```

3. Pull the WJHGraf docker image.

- If internet connection is available from the host, and you are pulling the image from the docker hub, run the following command:
  `docker pull mellanox/wjhgraf`
- If you do not have internet access and cannot pull the image from the docker hub, use the following procedure to load the image on your host (assuming you have already downloaded it locally).
  `cp <location>/wjhgraph.img.gz /tmp; docker load -i /tmp/wjhgraph.img.gz`

4. Start your container binding the external ports 3000, 8093.

```
docker run -dit --name wjhgraf --restart unless-stopped -p 3000:3000 -p 8093:8093 mellanox/wjhgraf
```

III. Initial Settings and Configuration

For the Telemetry Agent to start connection attempts to the controller, the controller_ip and controller_port must be changed to the correct provider values and the enable_telemetry parameter must be set to "yes". This is possible to perform using the telemetry configuration script that is located at /opt/telemetry directory on the Docker:

```
docker exec telemetry-agent "/opt/telemetry/utils/run_wjh_session.sh --collector_ip <paste ip
adress> --collector_port 8093 --collector_type influx"
```

The Telemetry Agent will try to establish connection with the controller.

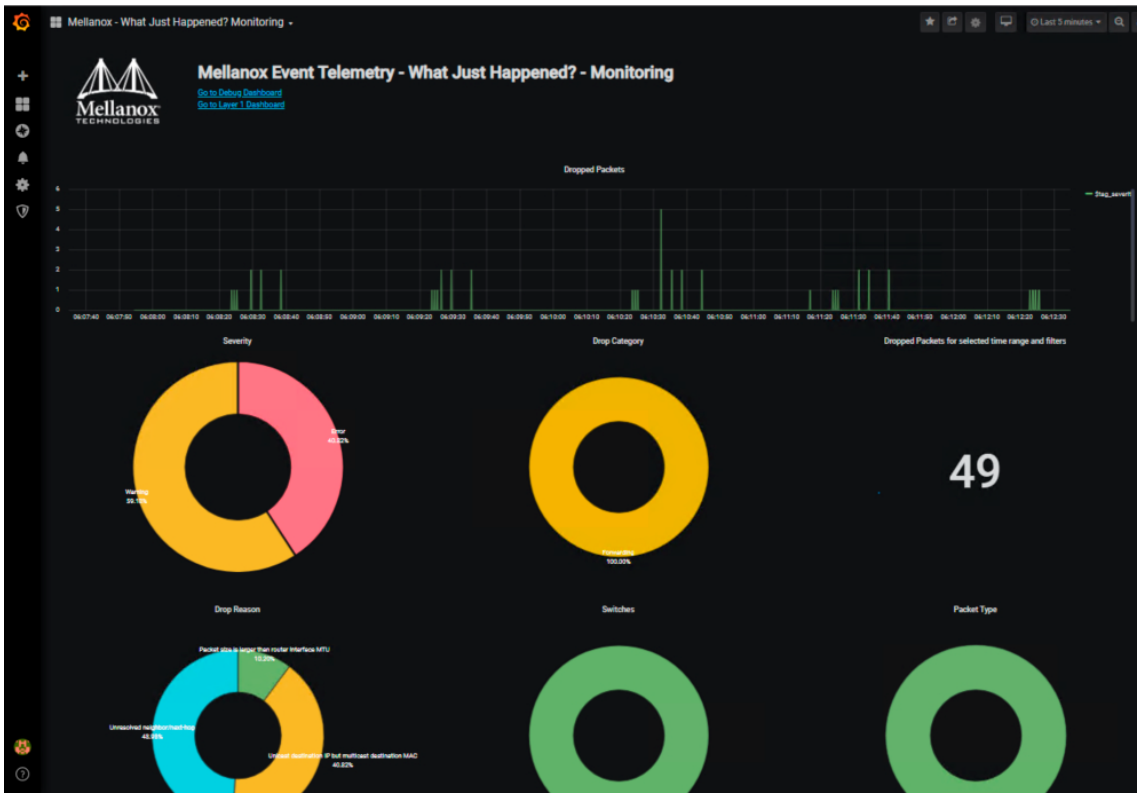


Figure 7: WJH Main Monitoring Page View Using Grafana

350 Oakmead Parkway, Suite 100, Sunnyvale, CA 94085
Tel: 408-970-3400 • Fax: 408-970-3403
www.mellanox.com