influxdb™

AN INFLUXDATA TECHNICAL PAPER

Network Monitoring with InfluxDB

OCTOBER 2023 | VERSION 1A

Introduction

Networks play a fundamental role in the adoption and growth of Internet applications. Present in enterprises, homes, factories, and even cities, networks sustain modern society. While assuring responsive and performant networks in today's hybrid, distributed, and containerized application environments occurs behind the scenes in intangible clouds and diagram abstractions, network glitches are more visible and unforgiving than ever when it comes to end users.

Keeping everything running and functional is at the core of IT Operations, but the role of monitoring doesn't stop there. The shift to using more advanced network monitoring analytics platforms, coupled with machine learning and AI frameworks, can address enterprise network problems in real-time and predict network problems before they occur. Network monitoring takes DevOps and IT Operations beyond the cost of doing business, turning it into a business unit that boosts effectiveness, efficiency, and profitability.

Network Monitoring — Pillars for Success and Growth

Developers architect today's applications to work on hybrid, componentized, containerized, and distributed mobile/sensor environments, as do the networks that sustain these applications. Complexity, traffic volume, intolerance to performance degradation, and inefficiency will only increase, demanding real-time and holistic monitoring. Metrics as a Service inside organizations has become a reality, allowing operations to better articulate necessary investments. Facing the new paradigm, DevOps and IT Operations teams must be able to answer both cost-centric and profit-centric questions.

Cost-centric questions	Profit-centric questions
Are network resources available?	Are network resources fully utilized (provisioned vs. use)?
How is the network performing (latency, error rates, bandwidth consumption)?	How are KPIs impacting revenue generation?
Is root-cause identified promptly?	Is the Mean Time to Repair (MTTR) impacting user experience and causing churn?
Who/what is driving consumption?	Are there network misuses or new revenue generation use cases?
Do I have the right tools to acquire actionable insights and prediction models?	Are mission-critical services' KPIs being effectively tracked and tackled?

Network monitoring helps businesses prioritize decisions by helping to answer the questions listed above. There are several sources of instrumentation for network monitoring data, such as endpoint metrics, events, logging, sensor data, traffic flow analysis, packet inspection, and synthetic tests. However, users often collect information in silos, consuming IT resources without providing a complete view of the application environment, all while leaving gaps that can lead to flaws and oversights when assessing network and resource utilization.

The Basics of Network Monitoring

Foundation for network performance monitoring

Accommodating all monitoring requirements for a modern application environment is not a trivial task. Companies need a global and a granular view of the network's impact on services. Additionally, to avoid introducing yet another incomplete solution, a sound approach must account for supporting multi-type data and cross-measurement analytics that can accommodate requirements from the whole organization.

DevOps and IT Operations deal with complex and demanding environments—nevertheless, establishing centralized network performance management can be simple. Three network performance monitoring pillars separate the unmonitored from the monitored network, and by feeding this data into a centralized time series database, you enable holistic network performance monitoring.

1. **Network availability** – This refers to host reachability. If unavailable, there may be something related to the endpoint health or network path (such as a load balancer's session limit or expired SSL) preventing traffic from reaching the host.
2. **Network responsiveness (latency and packet loss)** – Latency refers to the time it takes traffic to cross the network to a target, and packet loss determines the error rate experienced. Latency and error rates above certain thresholds render the network unsuitable for all or some more sensitive applications. For instance, high latency will completely undermine unified communications (voice & video) services.
3. **Network bandwidth consumption (who/what, when, and how much)** – This tracks metrics from the network interface and traffic flow, providing important information about bandwidth load. Using this information, you can set alerts that warn you before the interface becomes saturated, avoiding impact on applications. Correlating the network interface metrics data with flow analysis from network traffic analyzer appliances provides precious insights about sessions and IPs/protocol/port troublemakers that are causing saturation and bandwidth struggles. This information can also be used for identifying resource misuse and potential Denial of Service (DoS) attacks.

Platform approach to network monitoring

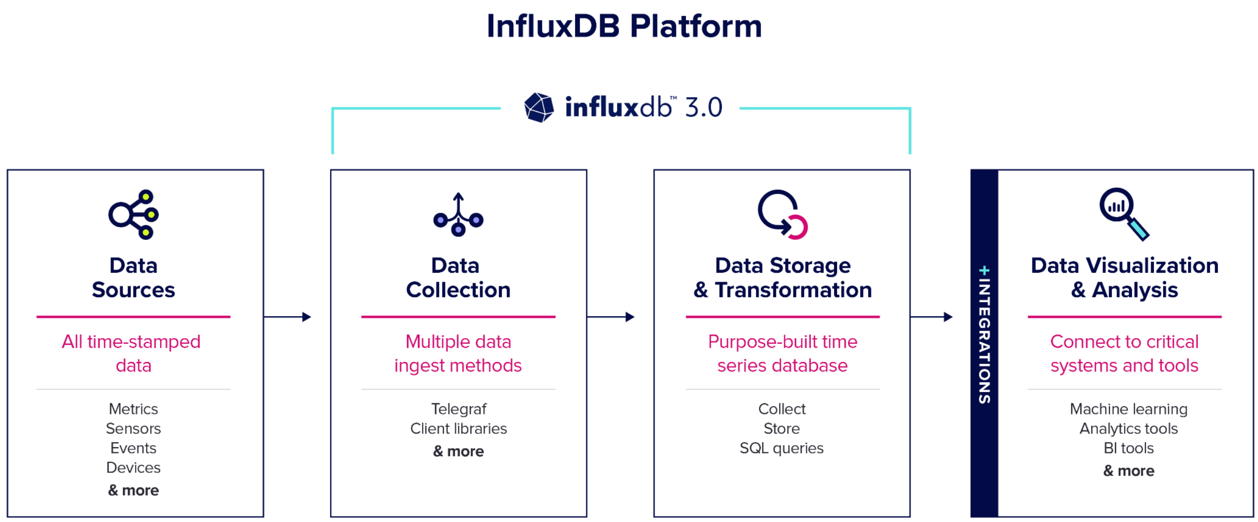
Instrumentation of monitoring data is just the beginning of the journey. Thinking from a strategic standpoint is necessary in order to serve the needs of various constituencies in the organization while avoiding drowning in your own data. Silos are a no-go because they demand dedicated personnel and additional development work to extract the full benefit of data from multiple sources.

A platform to which all monitoring data can converge for storage, alerting, and processing is a strategic approach to monitoring. Corroborating the value of taking a strategic approach, the Open Networking User Group (ONUG) Monitoring and Analytics Working Group approaches network monitoring from this perspective: it should not be restricted to any single data source or type or thought of in isolation.

The functional architecture of the InfluxData monitoring platform consists of two components:

- **Telegraf** – The collection agent with 300+ plugins and a vast client library that can source metrics directly from the system it is running on, pull metrics from third-party APIs, and listen for pushed metrics/events or via streaming consumer services. Telegraf as a monitoring collector supports protocols such as ICMP/Ping, SNMP, NETFlow, sFlow, Syslog, Arista LANZ, and Cisco GNMI.
- **InfluxDB** – The database and storage engine purpose-built to handle time series data. An ideal storage solution for monitoring network data with a powerful query engine that processes multiple data sources, helping avoid a siloed approach.

Below is a graph representation of the functional architecture of the InfluxDB platform:



A Purpose Built Design for Network Monitoring

It is important to choose a monitoring platform built to support efficient network performance monitoring and accommodate the types and volume of data generated in a modern application environment. In contrast, solutions without an architecture that can fully accommodate unified monitoring will always narrow your horizon to fit into their capabilities instead of allowing you to extract full benefits from a monitored network.

Key architectural feature 1: Pull & push collection mechanisms

When implementing a network monitoring framework, it is important to consider both pull and push data-gathering methods. Pull methods are a good choice for collecting performance and usage metrics at regular intervals; however, a push mechanism is necessary to collect actionable events when they happen.

The choice for pull or push also impacts traffic on those networks. If monitoring traffic is not well-planned, network bandwidth and endpoint resources suffer, leading to performance degradation and diminishing benefits from network monitoring. A combination of pull and push monitoring mechanisms is usually necessary. These factors are particularly relevant to IoT and in cases of very chatty monitoring protocols.

Key architectural feature 2: Scalable and durable

With all data converging in one platform, the inability to scale is a show-stopper. Your solution must be able to ingest at high rates (millions of data points per second) and store large amounts of data (hundreds of millions of time series measurements). Your query engine must be performant to support real-time alerting and analytics across measurements and make the data durable for as long as needed. .

Key architectural feature 3: Flexible, multi-type, and multidimensional data model

Flexible data models that allow resource-efficient collection of multidimensional fields and metadata *and* visualization at granular grouping tiers makes it possible to isolate and direct relevant monitored measurements to the right audience.

Support for multi-type data like numeric and non-numeric, accommodates various monitoring use cases from different business units, which may include tracking a string (error codes) or boolean data (true/false status) type.

How to Use InfluxDB for Network Monitoring

Everything starts with collecting the relevant network monitoring data. The following protocols and tools provide a foundation for gathering network monitoring data:

Arista LANZ	BIND 9 Nameserver	Bond
gNMI	Cisco Model-Driven Telemetry	Conntrack
DNS Query	Ethtool	Fail2ban
HAProxy	Icinga	Ipset
JTI OpenConfig Telemetry	Mcrouter	Modbus
Monit	Net	Netstat
Network Response	OpenNTPD	PF
PowerDNS	SNMP	Socket Listener
Ping	Syslog	Netflow, IPFIX, sFlow
x509		

Telemetry can also be streamed from leading network appliances into Telegraf for batching and normalization or even straight into InfluxDB, as is the case for Cisco Pipeline, LibreNMS, Nagios Core, and many other tools. Bringing all this network data together from different protocols, appliances, and methods supports the earlier statement that you can fully understand and manage your solutions only with a unified solution.

Ping for network monitoring

Ping is an IT administration software utility used to test the reachability of a host on IP networks. Ping operates by sending Internet Control Message Protocol (ICMP) echo request packets to the target host and waiting for an ICMP echo reply. It is available as a CLI tool on practically all OS platforms with a Telegraf Ping Plugin. Telegraf Ping Plugin allows you to measure the round-trip for ping commands, response time, and other packet statistics, which helps you understand network availability, latency, and packet data loss.

The plugin can be deployed as a central collector to ping/query all devices per data center, security zone, or subnet.

SNMP for network monitoring

Simple Network Monitoring Protocol (SNMP) is used to collect information about devices (routers, switches, servers, etc) on IP networks. It is widely used in network management and monitoring and exposes the system status and configuration, which can be queried. SNMPv1 is the original version—SNMPv2c and SNMPv3 provide improved performance, flexibility, and security.

SNMP network monitoring is supported with Telegraf Syslog Plugin.

NetFlow, IPFIX & sFlow for network monitoring

NetFlow, IPFIX, and sFlow are standards used for congestion control, troubleshooting, and network management, among other things. From a network performance management perspective, these protocols allow you to collect bandwidth data per traffic flow and per IP/protocol/port.

- Cisco introduced NetFlow and vendors like Juniper and VMWare provide support to collect IP network traffic as it enters or exits an interface.
- Internet Protocol Flow Information Export (IPFIX) is a standard derived from Cisco's proprietary NetFlow v9 used to export traffic information from L2-L7. Its design addresses the shortcomings of NetFlow and is vendor-neutral.
- sFlow (short for "sampled flow") is an industry-standard for packet sampling on Layer 2-7 that provides a means for exporting truncated packets for network monitoring. Many vendors support this standard, including but not limited to Cisco, Arista Networks, Fortinet, Juniper, and IBM.

Although these protocols (except for sFlow) are not currently supported by a Telegraf plugin, an InfluxData partner, ntop, has built a ready-to-use solution to collect network traffic, NetFlow, IPFIX, and sFlow messages stored in InfluxDB.

GNMI

If you are part of a network operations team, the ability to collect data in near real-time is important for network visibility and performance. Many switches, like the Nexus switches, can stream telemetry data using gNMI, and the Cisco gNMI Telemetry Telegraf Plugin allows you to consume this data in InfluxDB. Once in InfluxDB, you can visualize the data in InfluxDB, Grafana, or your custom dashboards. Dashboards of this telemetry data will prove useful in your day-to-day operations, automation, and network planning. In addition, if you have devices using other protocols like SNMP, you can gain a holistic view of all your network devices.

Arista LANZ

Short for Arista Network Visibility Latency Analyzer, LANZ is a tool designed to track interface congestion and queue latency with real-time data collection and reporting—allowing developers to guarantee maximum performance for end users at all times. With the LANZ application layer event export, all the applications you work with can use historical data to predict impending congestion and latency.

The Arista LANZ Consumer Telegraf Plugin is a consumer for use with Arista Networks' Latency Analyzer (LANZ) to stream data via TCP through port 50001 on the switches management IP into InfluxDB. LANZ provides congestion data by continuously monitoring each port's output queue lengths. When the length of an output queue exceeds the upper threshold for that port, LANZ generates an over-threshold event. Collecting these metrics in InfluxDB allows you to gain insights into your networks and enables your applications to react to changes in the network conditions. You can pair this with a number of other Telegraf plugins to get a view into your entire application stack.

JTI OpenConfig Telemetry

Junos Telemetry Interface (JTI) is a push mechanism to collect operational metrics for monitoring the health of a network that has no scaling limitations. Unlike JTI, traditional pull mechanisms like SNMP and the CLI—which require additional processing to periodically poll the network element—directly limit scaling.

The JTI OpenConfig Telemetry Telegraf Plugin reads the Juniper Networks implementation of OpenConfig telemetry data from listed sensors using the Junos Telemetry Interface. This data is helpful when monitoring the performance of any Jupiter device you may have in your environment. Compared to traditional pull mechanisms, the Junos Telemetry data is streamed in real-time to allow network administrators to measure trends in link and node utilization and troubleshoot such issues as network congestion in real-time.

SNMP may still be used in your environment by other networking devices, so using a variety of Telegraf plugins to pull in all of your networking and infrastructure performance data will give you a complete view of your stack.

X.509 certificates

X.509 certificates have an expiration date that can prevent your website or applications from working properly and present users with a warning that the site's security certificate has expired. To avoid this, it is a best practice to check the expiration dates on a regular basis. The X.509 SSL Certificate Monitoring Template, which uses the X.509 Certificate Telegraf Plugin, does just that—monitors SSL certificate expirations.

Other network monitoring data

This is just scratching the surface when it comes to network metrics and events. You may find that there is monitoring data that you and your organization understand and would like to see added to your InfluxDB instance. What's important to note is that as a set of open source projects, you can easily work with the community or contribute directly to InfluxDB or Telegraf. Telegraf's popularity is due to its ease of use and deployment, use case versatility to support a number of datasources, and finally, the ease with which you can contribute to a new Telegraf plugin that supports gathering the data that you need for your monitoring purposes. Writing your own Telegraf plugin is easy due to its well-documented and relatively simple process. With the source code being open and a vast library of plugins already implemented, there are a lot of examples to draw insights and guidance from.

Network monitoring best practices

To get the most out of your network monitoring efforts, consider these best practices:

Regularly update tools

Staying current with software patches is crucial to maintaining the security and functionality of your tools. Regularly updating your software allows you to address any vulnerabilities or bugs, reducing the risk of cyberattacks and optimizing performance.

Set meaningful thresholds

Setting meaningful thresholds helps avoid false alarms and ensures you only receive alerts when necessary. Customizing thresholds based on your specific requirements allows you to filter out irrelevant notifications and focus on critical events.

Set up regular reporting

Performing weekly or monthly checks and generating regular reports provides valuable insights into the health and performance of your systems. Regular reporting lets you track key metrics, identify trends, and make informed decisions to optimize your operations. For example, analyzing weekly reports on website traffic can help identify peak usage periods

.

Segment monitoring

By segmenting your network based on factors such as criticality and sensitivity, you can dedicate more attention and resources to the most important components. For instance, in a large organization, segmenting network monitoring can help prioritize monitoring of sensitive customer data over less critical internal systems.

Backup configurations

Having a recovery plan is essential to minimize downtime and ensure business continuity in case of system failures or data loss. Regularly backing up configurations and important data safeguards against potential disasters and enables swift recovery. Regularly backing up server configurations can significantly reduce recovery time and minimize the impact of hardware failures, network failures, or software corruption.

Monitor essential network devices

By focusing on the key devices in your network, such as routers, switches, and firewalls, you ensure that the backbone of your infrastructure remains robust and secure. In larger networks, prioritizing these essential devices means timely identification of issues, preventing cascading failures that could impact the broader system.

Choose the right polling frequency

Setting the right monitoring interval for each network device is crucial. Polling too frequently can overburden your system while infrequent checks might overlook critical issues. Device availability should be monitored every minute, CPU and memory every 5 minutes, and disk utilization every 15 minutes. Adjust intervals based on each device's significance to ensure system reliability without unnecessary strain.

Select the right network protocol

Selecting the right network protocol is essential after identifying devices and setting monitoring intervals. SNMP is a prevalent choice, supported by many network devices and Linux servers, and bundled with an SNMP agent. While Windows devices typically favor the WMI protocol, it's crucial to adopt a protocol like SNMP, which is secure and minimizes bandwidth consumption, ensuring minimal impact on network performance. Properly configuring SNMP can grant complete control over a device, allowing configuration replacement. A top-tier network monitor enables administrators to set SNMP privileges, ensuring secure and efficient network management.

Conclusion

Choosing a platform that can handle a unified monitoring strategy and accommodate use cases that address the needs of various organizational constituencies is fundamental. So is paying attention to efficiency, which can be an important aspect of IoT support. Monitoring starts with identifying the information that matters. After that, the InfluxDB platform provides the tools to move along the series of steps that make up a comprehensive monitoring strategy: Collect > Store > Process > Visualize & Alert > Automate.

About InfluxData

InfluxData is the creator of InfluxDB, the open source time series database. Our technology is purpose-built to handle the massive volumes of time-stamped data produced by IoT devices, applications, networks, containers, and computers. We are on a mission to help developers and organizations, such as Cisco, IBM, PayPal, and Tesla, store and analyze real-time data, empowering them to build transformative monitoring, analytics, and IoT applications quicker and to scale. InfluxData is headquartered in San Francisco, with a workforce distributed throughout the U.S. and across Europe. Learn more.

InfluxDB documentation, downloads & guides

Download InfluxDB
Documentation
InfluxDB Network Monitoring
Additional resources
Join the InfluxDB community
Get InfluxDB Cloud