Network Assurance: Your Key to Proactive Operations

Prologue

It’s 4 p.m. on Friday and your cell phone rings. Jim, the director of operations is on the line. “I know you are about to leave for your vacation, but I just got a report that our inventory system crashed. Manufacturing has a big order they need to ship before Monday. Could you please take a quick look?”

This is bad news. You wanted to wait before pushing out the new network policies— but security operations insisted. Last time you got such a call, you and the guys stayed up all night combing every switch configuration. It was a true nightmare. You hesitate: Should you call your wife and tell her that plans have changed? She is going to be furious, and the kids will be so disappointed. You sigh…

What if you could predict the impact of changes and know they won’t break anything? What if you could truly have the confidence your network is operating exactly as you intended it to?

Perhaps it is time to test the new “assurance engine” your team installed last week. Will it save the day as the sales rep promised?

Network Assurance Gap

If you are building or operating data center networks, then you live in exciting, yet challenging, times. Data centers have been going through rapid evolution: growing massively in scale, becoming increasingly sophisticated with cloud adoption, and changing rapidly.

We at Cisco responded to these challenges by offering new intent-based networking platforms. They allow you to manage large-scale infrastructures in highly automated ways. Rather than laboriously configuring network devices using a low-level command-line interface, you can easily program your network at an abstracted level and control it through an intuitive and powerful policy language.

So, life with these modern architectures is easy, giving you extra time at the beach, right? Not quite…

Let’s face it: your operational paradigms are still rudimentary, fundamentally reactive in nature. You grapple with operational issues and outages and respond with war rooms and firefighting using basic telemetry, interface statistics, etc. When you have a breach, you scramble to analyze your policies, trying to find the loophole that was just exploited. With only ad hoc verification, your changes often break the network, making rollbacks more the norm than an exception. You struggle to guarantee compliance with regulations, making audit failures a frequent reality.

You are unable to assure intent in your networks proactively, facing what we call the “network assurance gap.”

If you could, the questions you would rather ask the network are more fundamental.

If you made changes to the network by modifying some high-level policies, how do you know that you haven't introduced some misconfigurations or errors that will bring down the application in a few hours? Or a few weeks? Maybe the security policy you programmed is conflicting with an existing deny policy you don't know about. Or perhaps you have programmed a new subnet that overlaps with an existing one.

Wouldn't it be nice if you had a system that proactively analyzed your entire policy space, every configuration, to eliminate human errors, avoid policy drift, and give you the confidence that your changes are correct and consistent?

Even if you have done all your programming properly, do not forget that your network is a dynamic, distributed system whose state changes over time. For example, the fabric is learning prefixes from the outside world. What if a routing loop was created in the forwarding tables, or it learned a more specific route from the branch so that traffic meant for an internal app gets diverted outside?

These are but few examples of gaps between "business intent" and the actual dynamic state of the network. The gaps are often hard to find, and even harder to reason about. But it is critical to identify them, because they can expose you to potential outages or vulnerabilities that will severely impact your business.

Wouldn't it be nice if you had a system that continuously analyzed your entire network's dynamic state—the forward state, endpoint configurations, etc.—to ensure it is always consistent with your intent?

While using abstract policies to program your network is the way to go, what happens once you realize your network doesn't quite behave as you intended? How do you troubleshoot your network without knowing its bottom-up view? Where are your VLANs, bridge domains, and endpoints sitting? How is connectivity being established between A and B?

Wouldn't it be nice if you had a tool that reconstructed the bottom-up state of the network and correlated it to the policy, enabling you to troubleshoot issues an order of magnitude faster?

You need network assurance: the confidence that your network is doing exactly what you intended it to do, always.

Network assurance encompasses everything you do in data center network operations: having confidence in your changes and configurations, knowing your routing and forwarding state is consistent, ensuring your security policies meet the segmentation goals and compliance requirements, passing audits easily, and so on.

How do others deal with assurance gaps?

You are not the first to experience these "assurance gaps." Other industries have faced similar challenges before you. The process of assuring that a system behaves in accordance with its specification is generally known as verification—an independent procedure used for checking that a system meets requirements and fulfills its intended purpose.

Sure, one can try to manually test a complex system. But that's an extremely time-consuming and error-prone process, and **exhaustive** testing of all possible scenarios is nearly impossible. This is where the well-established research field called **formal verification** comes to the rescue—the use of algorithms that automate the process of exhaustively checking that a system meets its intent. We're not talking about a theoretical application here; automatic verification tools have been put to good use for many years.

Let’s take integrated circuits (ICs, or chips), for example. They pack billions of transistors used for implementing a complex system on a tiny piece of silicon. IC engineers must “get the design right” before starting to manufacture chips in large quantities. Chips are designed and built in huge quantities, while reports on errors are very rare. How is the magic done? A slew of verification tools is used throughout the design process, ensuring the logic, the code, and its physical manifestation are correct and will operate as intended.

A similar approach has been adopted in the software world. Software developers used to write code and then spend a great deal of time manually debugging it. Nowadays, developers employ a suite of verification tools, such as static analysis, dynamic analysis, code coverage, memory testing, etc. Thanks to these verification tools, most problems are caught before the code goes into production.

The networking industry has been late to adopt verification techniques. When we push changes into a network, we simply deploy and monitor. And when problems arise, we troubleshoot. We have been relying for too long on manual efforts by smart network engineers.

With the growing complexity of today’s network, human ingenuity simply isn’t enough. Fortunately, we can apply formal verification techniques to networking. After all, networks are deterministic, and protocols have expected behavior. These facts allow us to mathematically model the network—and automatically verify it.

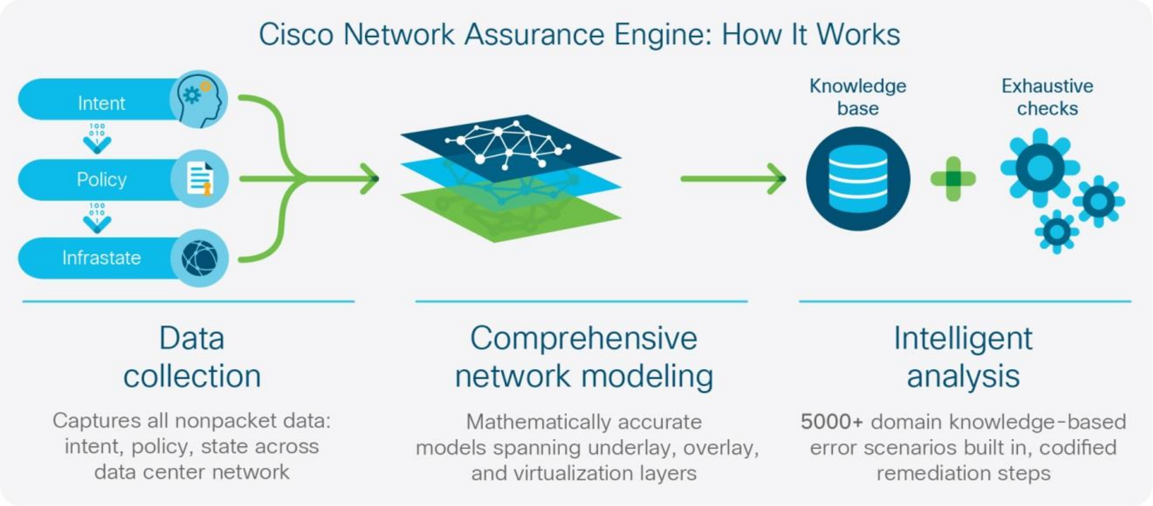
Cisco Network Assurance Engine: How we assure networks

“Let software manage complex software”

Cisco® Network Assurance Engine brings formal verification techniques into networking, helping close the assurance gap. It mathematically verifies and validates the entire data center network for correctness, giving operators the confidence that their network is always operating consistently with their intent, even as it changes dynamically.

Figure 1 shows the building blocks of Cisco Network Assurance Engine.

Figure 1. Cisco Network Assurance Engine



Cisco Network Assurance Engine: How It Works
Intent
Policy
Infrastate
Knowledge base
Exhaustive checks
Data collection
Captures all nonpacket data: intent, policy, state across data center network
Comprehensive network modeling
Mathematically accurate models spanning underlay, overlay, and virtualization layers
Intelligent analysis
5000+ domain knowledge-based error scenarios built in, codified remediation steps

- **Data-collection framework:** The data-collection framework periodically ingests all non-packet data—operator intent and policy and configurations from the controller, software configurations from every switch, and data-plane state from each device—and stores it in a platform-agnostic format.
- **Formal modeling core:** This is the core technology within Cisco Network Assurance Engine that creates the most real-time "formal models" of the network: mathematically accurate representations of the network's actual behavior based on the real-time state and policy. For instance, Network Assurance Engine models all the security contracts, the forwarding state across all the switches, the configurations of all the endpoints across the network, and so on.
- **Continuous analysis engine:** Built on a streaming big-data architecture, the analysis engine runs thousands of failure scenarios continuously against these mathematical models of the network. Over 5000 failure scenarios have been codified in the product, based more than 30 years of Cisco's operational domain knowledge. These scenarios are continuously enhanced using the collective knowledge of common failure patterns learned from thousands of customers.

The analysis runs continuously: Every few minutes the tool polls the entire policy and network state, updates the formal model, and runs the checks against it. When a discrepancy is found, the tool generates a "smart event," which pinpoints deviations from intended behavior and provides expert-level remediation suggestions.

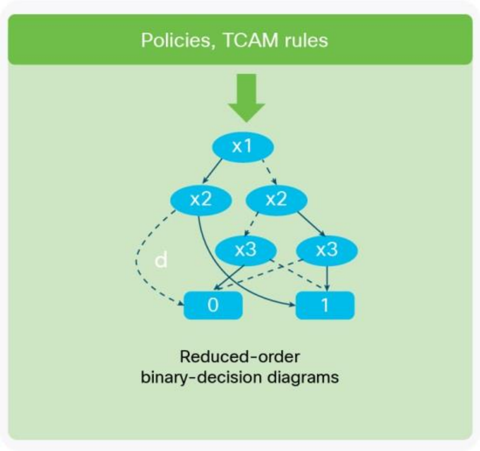
Cisco Network Assurance Engine models multiple behavioral aspects of the network, including:

- Tenant endpoint mobility
- Tenant security
- Tenant policy
- Resources utilization
- Tenant forwarding
- And many more

To better understand how the magic is done, let's look at one example: analyzing tenant security.

Cisco Network Assurance Engine reads all the security contracts and the enforcement policies down in hardware. These policies are translated into a "binary-decision diagram" (Figure 2), a highly optimized mathematical representation of these policies.

Figure 2. Modeling tenant security



Using this model, the analysis engine can answer several questions, such as:

- Who can A talk to?
- Can A talk to B based on policy?
- Do I have isolation between tenants?
- Are any policies conflicting with each other?
- Are some policies aliased?
- Did the upgrade to a new version change my existing security policy enforcement?
- Are the configured policies compliant?
- Which specific policy has been violated?

A similar approach is used for all other aspects of the network: collect the data, build the model, and run an exhaustive set of checks against it. Continuously.

Putting Network Assurance Engine to work

To understand how an assurance tool can dramatically improve your network operations, let's look at a few use cases.

Predict the impact of changes: Applying changes to your network has traditionally been a precarious process. You make your changes and then wait to find out if they introduced any errors into the network—which can sometimes be days later. Once errors occur, you drop everything and scramble to find and fix them. Using Cisco Network Assurance Engine, you can quickly verify whether your changes might result in potential errors. The built-in checks are used to analyze the network model, helping you quickly pinpoint errors and fix them—before they disrupt your network.

Root cause analysis: Detecting a problem in the network is but the first step. You need to fully troubleshoot it and understand its root cause before proper corrective action can be taken. This can be a very time-consuming process. Cisco Network Assurance Engine leverages decades of accumulated networking experience. It applies thousands of checks to the network in real time, and when problems are detected, they trigger smart events, which pinpoint the problem and its possible cause, and offer suggested remediation.

Network security policy and compliance: Achieving regulatory compliance and passing security audits is a very laborious process—one that must be repeated periodically. Using Cisco Network Assurance Engine, you can complete the auditing process with just a few clicks. It stores the full network state, so you can easily scroll back in time and ask, for example, "What was the state of my network 3 weeks ago? Did I have any security issues? Were my policies correctly configured?" Cisco Network Assurance Engine run these checks every few minutes, so you actually have **continuous** compliance checking. There is no more need to scramble every time you have a security audit.

Network resource utilization: One of the challenges network operators face is optimizing the usage of scarce resources. Once such scarce resource is Ternary Content Addressable Memory (TCAM), which is a critical component on many switching platforms. Cisco Network Assurance Engine analyzes how policies are mapped into each TCAM. It provides a detailed multi-dimensional understanding of utilization, identifies policy redundancies, and reports hit counts at a rule-level. This capability allows you to optimize your policies and tighten your security aperture.

The preceding use cases are but few examples of how Cisco Network Assurance Engine can help you **fundamentally transform the operations paradigm from reactive to proactive.**

What makes Cisco Network Assurance Engine Unique?

Cisco Network Assurance Engine combines advanced verification technology with decades of networking experience. We didn't just "borrow" algorithms from other domains, we added the capabilities required to truly close the network assurance gap. The result is a comprehensive intent-assurance suite that brings to networking the advantages of verification-driven, agile, proactive operations.

Some of the main advantages of the Cisco solution are:

- **Codified Cisco domain knowledge:** We've leveraged our deep understanding of network platforms, best design practices from Cisco Services and collective knowledge of common failure patterns learned across thousands of customers through Cisco's technical services. Then we incorporated this knowledge into more than 5000 built-in failure scenarios that are continuously run against the network model and proactively made available to all customers.
- **Deep controller integration:** Integration with the network controller and deep understanding of its policy model enables inclusion of orchestration policies within the mathematical model. We can then actively correlate policies with the dynamic network state, providing the operator with true intent assurance.
- **Comprehensive analysis:** We capture, analyze, and correlate the entire network state, including switch configurations and data-plane state, at the hardware level. The analysis spans multiple aspects of network behavior: controller policy, security contracts, network-wide forwarding state, virtual machine configurations and mobility, and utilization of hardware resources such as TCAM.
- **Unified and indexed network repository:** Cisco Network Assurance Engine includes the industry's most complete repository of network-wide state, configurations, policy, and intent data indexed and stored over time. It is made available for rapid recall, search, and ad hoc analysis.

Cisco Network Assurance Engine is part of Cisco's broad vision for intent-based networking. It seamlessly integrates with the Cisco Application Centric Infrastructure (Cisco ACI™) solution and Cisco Nexus® 9000 Series Switches, making Cisco the only vendor with a complete product suite for intent-based networks. With its open APIs, Cisco Network Assurance Engine easily interfaces with third-party virtual machine managers, and Layer 4–7 ecosystem partners. Furthermore, it can plug right into your alerting frameworks and automation workflows—making it DevOps ready.

Summing it all up

So what does it all add up to? Your data center network is becoming ever more complex, and we are all learning new paradigms to extract full value from these platforms.

Assurance offers a quantum leap in operational maturity for your organization—and the ability to get there much faster.

Cisco Network Assurance Engine can be the major game changer that is needed for your network operations. With continuous verification on your side, it is easier to proactively detect outages, predict change impact, and assure network security policies and compliance. Cisco Network Assurance Engine boosts confidence in your network operation, letting you spend more time on design and optimization instead of reactive troubleshooting. It lends you peace of mind: You can use orchestrators and policies to manage your network, and stop worrying about assurance gaps.

Epilogue

It’s 5 p.m. on Friday. With the help of Cisco Network Assurance Engine, you managed to quickly pinpoint the problem and fix it. Among the dozens of new security policies were two that conflicted, blocking access to the inventory database server. The latest verification report gave the network a clean bill of health, so you get into your car and head home. On the way, your phone rings. It is Jim on the line: “I just heard from the inventory guys—I’m not sure how you pulled off this magic, but you know what? Enjoy your vacation!” You smile and drive home. This was indeed magic, now it’s a whole new way of doing things.

For more information

https://www.cisco.com/go/networkassurance

Americas Headquarters
Cisco Systems, Inc.
San Jose, CA

Asia Pacific Headquarters
Cisco Systems (USA) Pte. Ltd.
Singapore

Europe Headquarters
Cisco Systems International BV Amsterdam,
The Netherlands

Cisco has more than 200 offices worldwide. Addresses, phone numbers, and fax numbers are listed on the Cisco Website at **https://www.cisco.com/go/offices**.

Printed in USA C17-740236-00 02/18