Technical Review

Verifying Network Intent with Forward Enterprise

Date: May 2019 **Author:** Alex Arcilla, Validation Analyst

Abstract

This ESG Technical Review documents hands-on validation of Forward Enterprise, a solution developed by Forward Networks to help organizations save time and resources when verifying that their IT networks can deliver application traffic consistently in line with network and security policies. The review examines how Forward Enterprise can reduce network downtime, ensure compliance with policies, and minimize adverse impact of configuration changes on network behavior.

The Challenges

ESG research recently uncovered that 66% of organizations view their IT environments as more or significantly more complex than they were two years ago. The complexity will most likely increase, since 46% of organizations anticipate their network infrastructure spending to exceed that of 2018 as they upgrade and expand their networks.

The percentage of respondents who consider *their IT environment* to be **more or significantly more** complex today than it was two years ago.[1]

The percentage of respondents whose 2019 network infrastructure spending will increase relative to 2018.[2]

Large enterprise and service provider networks consist of multiple device types—routers, switches, firewalls, and load balancers—with proprietary operating systems (OS) and different configuration rules. As organizations support more applications and users, their networks will grow and become more complex, making it more difficult to verify and manage correctly implemented policies across the entire network. Organizations have also begun to integrate public cloud services with their on-premises networks, adding further network complexity to manage end-to-end policies.

With increasing network complexity, organizations cannot easily confirm that their networks are operating as intended when they implement network and security policies. Moreover, when considering a fix to a service-impact issue or a network update, determining how it may impact other applications negatively or introduce service-affecting issues becomes difficult. To assess adherence to policies or the impact of any network change, organizations have typically relied on disparate tools and material—network topology diagrams, device inventories, vendor-dependent management systems, command line (CLI) commands, and utilities such as "ping" and "traceroute." The combination of these tools cannot provide a reliable and holistic assessment of network behavior efficiently.

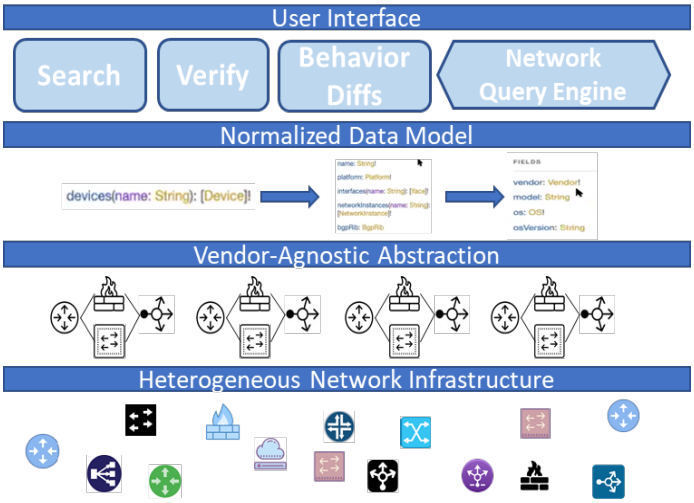
Organizations need a vendor-agnostic solution that enables network operations to automate the verification of network implementations against intended policies and requirements, regardless of the number and types of devices, operating systems, traffic rules, and policies that exist. The solution must represent the topology of the entire network or subsets of devices (e.g., in a region) quickly and efficiently. It should verify network implementations from prior points in time, as well as proposed network changes prior to implementation. Finally, the solution must also enable organizations to quickly detect issues that affect application delivery or violate compliance requirements.

The Solution: Forward Enterprise

Forward Enterprise is a network verification solution that simplifies how an organization confirms that its network implementation aligns with network and security policies as envisioned. Forward Enterprise falls into the category of

[1] Source: ESG Master Survey Results, *2019 IT Spending Intentions Survey*, March 2019.
[2] ibid.

"intent-based networking" solutions. "Intent-based networking" represents the automated analysis of network implementations to verify their alignment with an organization's end-to-end network and security policies. (i.e., the network "behaves as intended").



Forward Enterprise builds a software model of the network infrastructure based on the mathematical and logical analysis of network configurations to determine end-to-end behavior. It first reads the configurations of all existing Layer 2-4 devices (e.g., switches, routers, firewalls, and load balancers), as well as the services and policies enabled on them (e.g., forwarding rules, NAT, ACL, and VLAN). It also collects all the current state information from each device (e.g., routing tables and port status). The solution collects these details according to a predefined vendor-independent data model and stores those details in their underlying database.

Using an agentless architecture, the solution can document a network configuration and the underlying device details non-intrusively, regardless of the number of existing devices, based on periodic read-only access to network devices; it does not rely on live traffic analysis or log files. The architecture enables organizations to take timestamped "snapshots" (a collection of network configurations and state information) of part or all of their network infrastructures in real time. Organizations can use these snapshots to investigate how the network has been configured over time or how behaviors have changed, determine what compliance policies were in place, and compare to the current implementation.

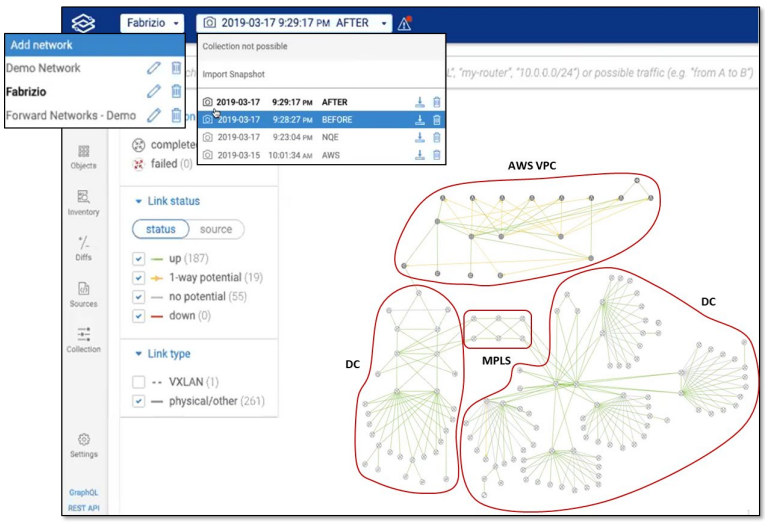
With Forward Enterprise, organizations can detect and isolate issues by searching for and analyzing all possible network paths that conform to a specific policy or intent. When updating a device, organizations can verify whether the change was effective and how it may affect other traffic flows. Organizations can compare snapshots taken over time to track device changes and determine how those changes affected traffic flows between different endpoints.

The solution also facilitates custom network analytics from the collected data model via the Network Query Engine (NQE). In the NQE, the data model is the stored information of network configurations. Device details are normalized across vendors and device types, so queries can be applied against the entire network. The NQE schema is loosely based on OpenConfig,[3] although there are some syntax differences and extensions for additional data stored in the solution. Organizations use the GraphQL query language to build custom searches in NQE. Once the data is retrieved, organizations can write scripts (e.g., Python) to compare, analyze, process, and format the data as needed.

ESG Lab Validated

ESG evaluated Forward Enterprise via a joint testing session hosted at Forward Networks' headquarters in Palo Alto, CA. We validated how Forward Enterprise enables a network administrator to detect devices in a network, verify network configurations, assess how network changes will affect end-to-end behavior and policies, and query the underlying data model to quickly extract and parse data for customer-defined checks.

[3] OpenConfig is an informal working group of network operators developing vendor-neutral data models to promote model-driven network management and operations.

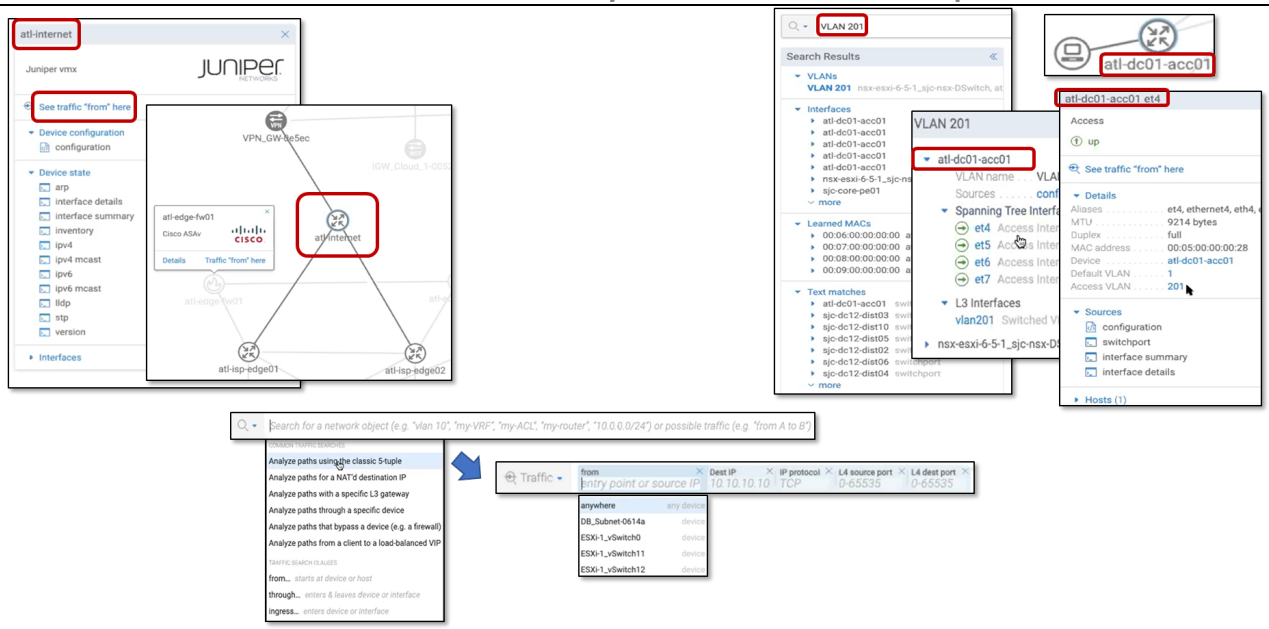


We first navigated to the ***Home*** screen of Forward Enterprise and viewed the graphical topology representation of our test network. The network consisted of two data centers, an MPLS backbone connecting the two data centers, and an AWS Virtual Private Cloud (VPC). The test network had pre-collected four snapshots for analysis—"AFTER," "BEFORE," "NQE," and "AWS"—dated and timestamped for easy reference. ESG immediately noticed how an administrator can use these network snapshots to search for and examine different parts of the network or different policies at different times.

ESG then examined the different ways to search for information within a snapshot. We first accessed the "AFTER" snapshot of the test network. To find out the details of one device, we clicked on the device named "atl-internet" (a Juniper gateway router) to reveal its details, such as the configuration files, the device's interfaces, and forwarding tables (see top left-hand side of Figure 1). The GUI also showed the physical paths directly connected to that device. We then clicked on "See traffic 'from' here" and saw all paths and destinations reachable from this device. We noted that an administrator can immediately assess if the device is isolated or reachable across the entire network.

We then used the ***Search*** functionality to find existing VLANs within the snapshot and typed"VLAN-201" in the search bar (see top right-hand side of Figure 1). Forward Enterprise revealed all elements associated with "VLAN-201;" clicking on any device uncovered its configuration and other details. For example, after clicking on "atl-dcc01-acc01," ESG was able to view its interfaces and MACs. We could also drill down further to reveal the configuration of each interface.

Figure 1. Search for Elements and Conduct Path Analysis Within a Network Snapshot

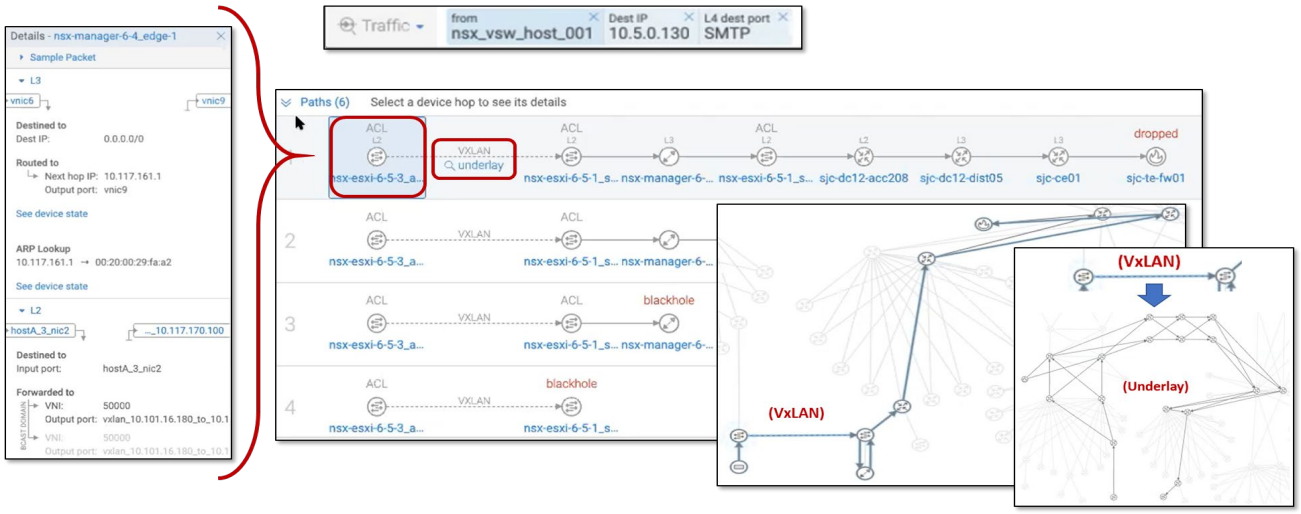


ESG then used the ***Search*** functionality to conduct a path analysis, a list of the available paths that conform with a stated policy or behavior. We proceeded to request all paths from a VM endpoint to a workload in the AWS cloud using SMTP. We navigated back to the search bar and clicked in the field to bring up a list of "Common Traffic Searches" (shown in the bottom of Figure 1). After choosing the first search, "Analyze paths using the classic 5-tuple," a predefined template appeared that only required filling in the given fields, such as source and destination IP. ESG noted how the predefined

searches and templates can significantly reduce the time to construct search queries to reveal hypothetical traffic patterns and related policies, especially as the number of network devices increases.

After inputting our search parameters, Forward Enterprise generated a path analysis (see Figure 2). The GUI presented a map of all possible paths currently available between our endpoints, from the entry point "nsx_vsw_host_001" to the destination IP address "10.5.0.130." The results also listed the devices and hops associated with each path. We could find out a device's configuration and its connection to adjoining devices by clicking on its icon (left hand side of Figure 2).

Figure 2. Results of Path Analysis Between 'nsx_vsw_host_001' and '10.5.0.130'

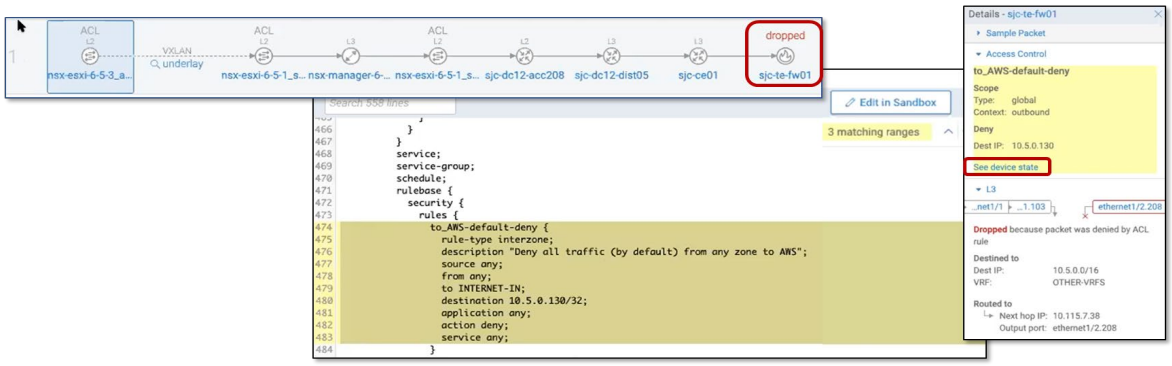


ESG also observed that the path analysis can identify overlay networks and correlate them with their underlay networks. As seen in Figure 2, a virtual extensible LAN (VxLAN) exists in all paths between the first two hops, a pair of VMware virtual switches. The VXLAN tunnel appears as a single virtual hop initially. To reveal the devices and paths in the underlay network, we clicked on "underlay" and revealed a number of possible paths (accounting for redundant paths) that traverse two data centers connected via the MPLS core.

ESG saw how a path analysis can eliminate the need to reference multiple sources when mapping out paths between specified source and destination points or determining the correlation between virtual and physical networks, such as the VxLAN overlay-underlay correlation. Imagine an administrator tracing all possible paths by cross-referencing network topology diagrams with device inventories, then accessing the relevant devices via CLI commands or disparate management systems (e.g., uncover routing rules). This task becomes more difficult as the number of network devices increases. ESG noted how Forward Enterprise can complete such a task quickly via its GUI.

ESG then examined how a path analysis identifies where the network fails to deliver traffic to an intended destination. In one path, our analysis showed that a firewall named "sjc-te-fw01" dropped the traffic (see Figure 3), preventing the traffic from being delivered to the AWS host. We clicked on "sjc-te-fw01," then "See device state" to reveal its configuration file. Forward Enterprise highlighted three parts of the ACL configurations that were likely causing traffic to drop. ESG recognized how an administrator can quickly identify why the firewall is dropping traffic, again decreasing outage time; otherwise, an administrator would use CLI commands and network utilities on multiple devices to find the root cause.

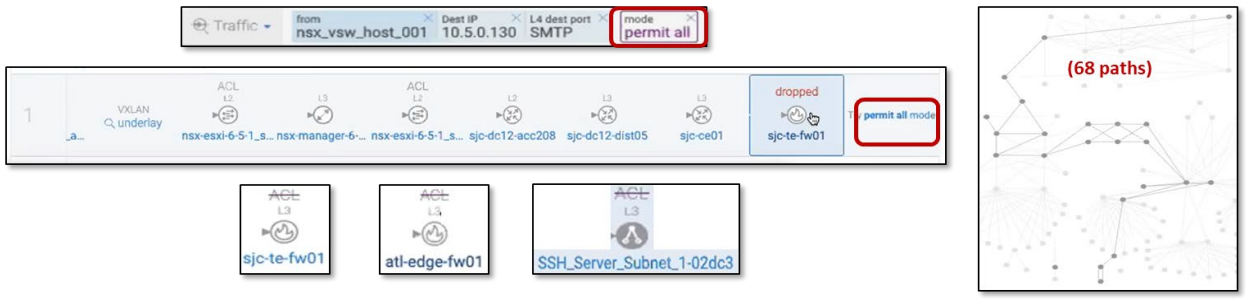
Figure 3. Locating Device Configurations that Result in Dropped Traffic



Even if an administrator changed the firewall configuration based on the previous analysis, that action may have unintended consequences, causing traffic to be dropped in other parts of the network, or potentially opening vulnerabilities that conflict with security policies. To quickly ignore the firewall configuration and allow us to focus solely on network connectivity issues in our search, we clicked on "permit all mode." This generated another path analysis that ignored all security policies existing for all devices between "nsx_vsw_host_001" and "10.5.0.130" (see Figure 4). Forward Enterprise identified several possible paths that now reach the AWS host, ignoring ACLs configured not only on "sjc-te-fw01" but also on "atl-edge_fw01" and "SSH_Server_Subnet_1-02dc3."

ESG noted that the "permit all mode" functionality can help an administrator reduce the time and effort spent on addressing connectivity or accessibility issues, especially in large networks managed by separate network and security teams. Forward Enterprise provides a way to isolate issues as either a networking or security issue immediately, allowing trouble tickets to be assigned accordingly without delay or parallel efforts.

Figure 4. Results of 'Permit All Mode'

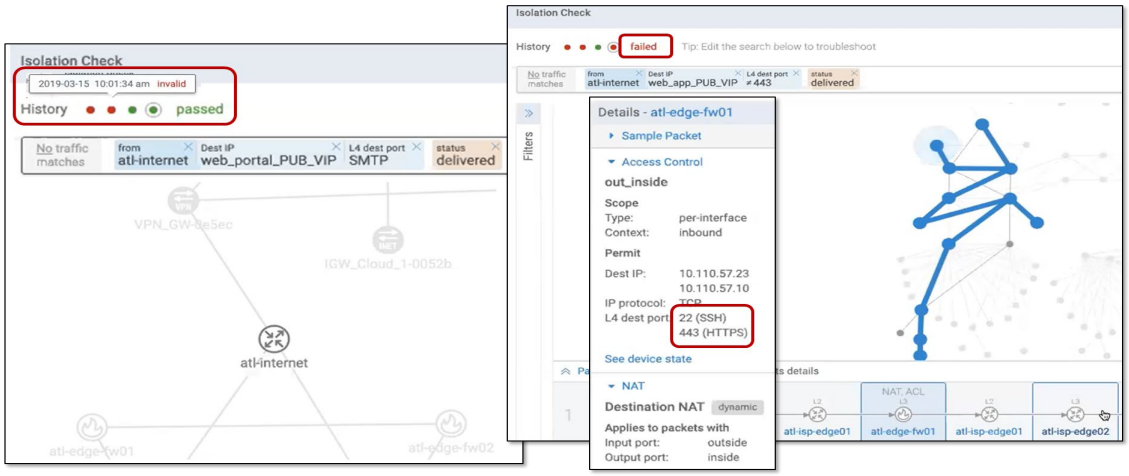


ESG proceeded to explore how an administrator uses the ***Verify*** function to check network compliance against policy requirements, device configuration errors, or custom network checks (such as traffic being sent to its intended destination under certain policies). Using the same snapshot from our previous testing, we clicked on Verify to view two categories of checks, Predefined and Intent. Predefined checks typically confirm that the network and devices have been configured according to best networking practices, such as IP address uniqueness, forwarding loops, VLAN definitions, and MTU mismatches. Intent checks (shown in Figure 5), customized by an organization's network team, focus on checking that device configurations, network topology, and policy requirements specific to the organization align with the desired intent. We observed that each check displayed a "passed" or "failed" status based on the network snapshot examined. We also noticed that we could filter on a check's status (e.g., show only failed checks) to highlight issues to be addressed.

Figure 5. Monitoring Status of Intent Checks

We further examined a passed and failed Intent check by clicking on "passed" and "failed" associated with the circled checks in Figure 5, and the screens shown in Figure 6 appeared. In the "passed" screen, we checked that the network would block SMTP traffic between the endpoints "atl_internet" and "web_portal_PUB_VIP." The check passed since no such paths existed. In the "failed" screen, we checked if non-SMTP traffic was all denied and found at least one viable path existed. We focused on the "atl-edge-fw01" firewall and uncovered that SSH traffic was still allowed.

Figure 6. Comparing Passed and Failed Intent Checks

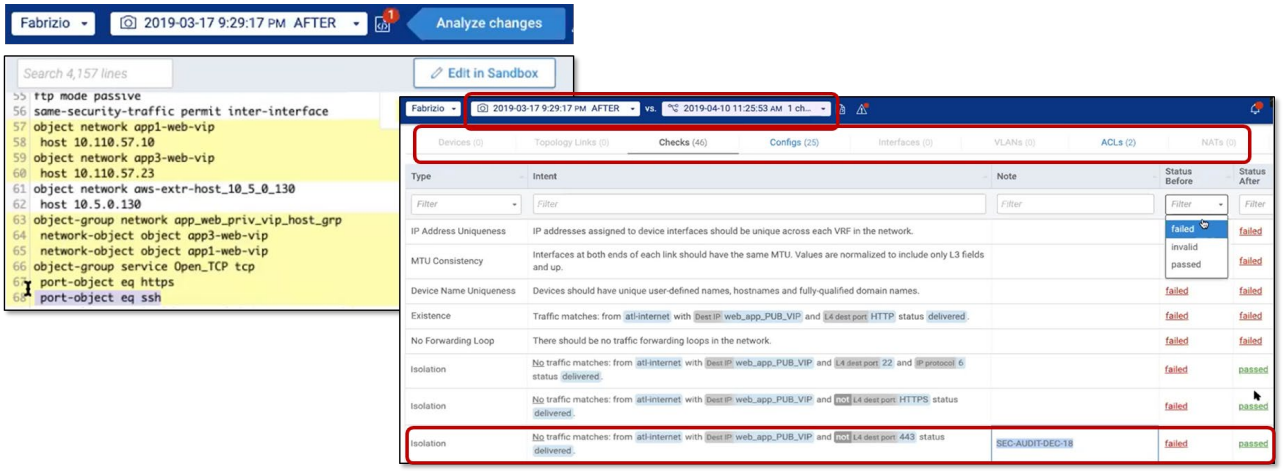


By using ***Verify***, ESG saw how an administrator could continuously monitor a network for potential problems, particularly when conducting network-related health checks and searching for policy violations. When filtering on "failed" checks, an administrator can prioritize what is to be resolved before they become issues in the live network. The administrator can also track the history of changes to isolate when an issue or compliance breach may have been introduced. Over time, using these checks can decrease the time and resources needed to ensure compliance, thus proactively preventing network problems and enabling frequent network changes and updates without adverse risk. Automating these tasks reduces the time spent on these tasks, thus decreasing operational costs.

ESG then observed how ***Behavior Diffs*** can help an administrator to minimize the downtime related to change windows. Behavior Diffs allows the side-by-side comparison of checks between two snapshots, such as before and after a device update. Using the previous example in Figure 6, we compared the status of verification checks before and after testing a change to the firewall that resulted in the "failed" status of the previously examined check. We clicked on "Edit in Sandbox" (see Figure 7), removed the SSH allowed rule, deleted the code "port-object eq ssh," saved the change, and clicked on "Analyze Changes." The solution produced a network snapshot with the proposed change and updated statuses on checks.

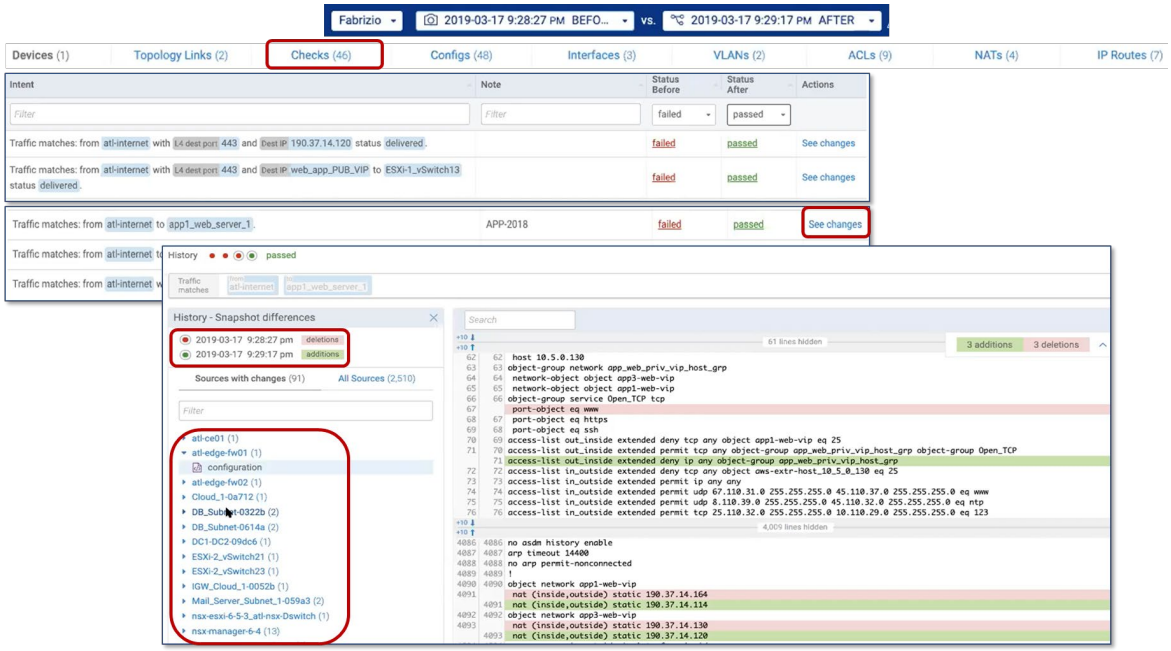
After deleting the SSH port changed, our check's status went from "failed" to "passed," and we viewed whether unintended consequences on other checks occurred (i.e., status changed from "passed" to "failed").

Figure 7. Utilizing Behavior Diffs to Simulate a Configuration Change and Its Effects



ESG then compared two snapshots to view other types of changes, such as adding new devices into the network that result in new connections, routes, and available interfaces. We clicked on ***Diffs*** in the main menu and were prompted to choose two snapshots. We then saw a summary of the number and type of changes made (shown in Figure 8), classified according to Devices, Topology Links, Configs, Interfaces, VLANs, ACLs, NATs, and IP Routes. For example, we could click on "Devices" to view added or removed devices. We clicked on "Checks" then "See Changes" to view a change summary related to an Intent check, including the number of devices that had configuration changes. We could also click on a device name in the list to reveal code changes; deletions are highlighted in pink while additions are highlighted in green.

Figure 8. Using Behavior Diffs to Examine Configuration Changes between Two Snapshots

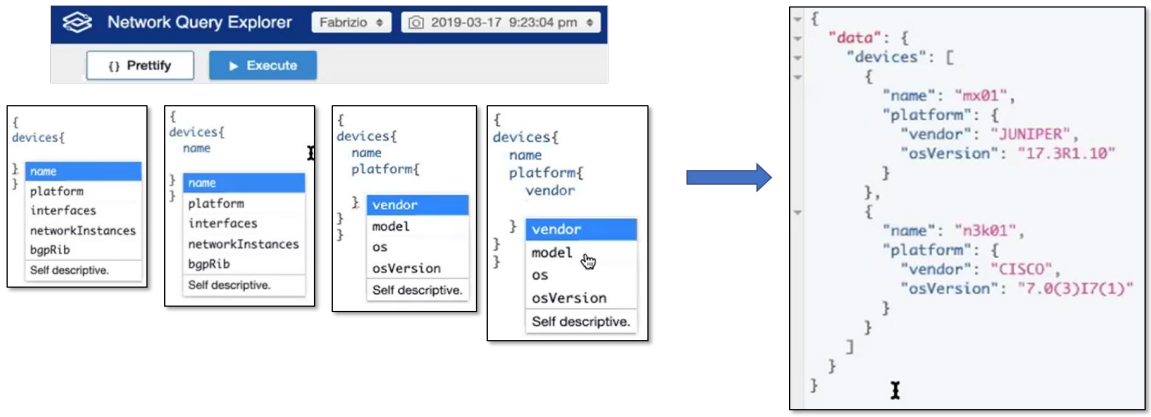


ESG saw how an administrator can leverage Behavior Diffs to analyze complex behavior and compliance issues between any two points in time, including proposed future changes. This would help retroactive compliance tests, which would be almost impossible without a saved working model of the network, as well as provide a quick verification of proposed changes against the current network behavior. This could significantly improve the success rate of change windows and reduce the

possibility of rollbacks. We saw how network operations can accelerate change windows with automated verification of changes, thus decreasing operational costs while reducing risk of adverse impact from changes.

While ESG noted the flexibility of the Search functionality, we also examined how network operations can use NQE to create custom data searches of the underlying model in the Forward platform. We navigated to the NQE screen (see Figure 9) and saw the framework for building custom searches. We noticed how NQE helped with framing a search, such as right clicking at every new line to show the available parameters. While we chose our search attributes, the right side of the screen automatically supplied values, drawn from the chosen snapshot shown at the top of the screen. For this search, we retrieved all devices in the snapshot, showing vendor and OS version.

Figure 9. Using Network Query Engine for Custom Searches



ESG could see how NQE can simplify the process of writing queries against the data model containing relevant configuration details and status of all network devices in a given snapshot. Rather than rely on APIs, an administrator can write a script via NQE to specify exactly the data to extract and process using a few lines of Python code. Network operations can easily create custom checks on network conditions or network policy specific to an organization. NQE also enables the administrator to gather the same data from multiple snapshots quickly to highlight, for example, devices whose IP address or name changed. ESG believes that the NQE eliminates the need to write code for accessing and retrieving device information and parsing configuration files and state tables to extract relevant data fields.

Why This Matters

Large enterprise and service provider networks are becoming more complex as network devices are upgraded and added to serve more users and support more applications. Organizations must manage this complexity and ensure that their networks operate in line with policy requirements and compliance objectives. Verifying network behavior against intent can be a tedious and expensive exercise as network complexity increases.

ESG validated that organizations can use Forward Enterprise to verify that their networks will operate as desired given their current design and stated policies. We found that the solution enables flexible searches to uncover viable paths that conform to desired behavior or highlight where compliance requirements are at risk. Organizations can trace where existing or potential service-impacting issues may lie and isolate potential configuration errors faster than using traditional tools such as network utilities. We also validated that organizations can use Forward Enterprise to test device changes to see if they resolve issues without causing unintended consequences. ESG also confirmed that organizations can track configuration changes over time and learn their impact on network behavior. Finally, ESG validated that Forward Enterprise enables organizations to extract specific data via its NQE in order to perform custom analyses. We saw that Forward Enterprise can help to resolve issues efficiently, minimize configuration errors, and ensure compliance via automated network health checks.

The Bigger Truth

As organizations upgrade and expand their network to accommodate more users and applications, they face the challenge of ensuring that all traffic flows adhere to network and security policies as envisioned. Verifying that network designs and configurations operate according to implemented policies as intended can be tedious, time intensive, and expensive. Organizations have relied on multiple sources of information—network diagrams, device inventories, vendor-specific management systems, CLI commands, and network utilities—to learn a network's configuration, track down service-affecting issues, assess the impact of proposed network changes, and check network behavior against network and security policies. Using these traditional tools can increase the time spent in root cause analysis, trouble ticket resolution, compliance assurance, and network change window validation.

Forward Networks designed Forward Enterprise to obtain a complete view of a network—topology, the underlying devices, and the network rules and policies that govern traffic—without relying on traditional tools and sources of information. Forward Enterprise can update network configuration details in real time so that the organization's knowledge remains current. Forward Enterprise creates this unified view regardless of device type, model, manufacturer, or OS. The solution automates a number of the verification and compliance checks typically associated with daily network management and administration.

ESG validated that Forward Enterprise can help organizations to learn device configurations quickly and align their network implementations with their policy requirements. We observed how organizations can track down and isolate issues easily as Forward Enterprise identifies potential causes and their impact on end-to-end network behavior via path analysis. ESG also validated that organizations can examine the impact of resolving issues and deploying updates as well as assess the extent of potential side effects against their stated intent. Finally, organizations can use NQE to construct custom network data queries easily for custom data extraction and parsing.

Organizations can leverage Forward Enterprise to confirm that the network is "behaving as intended" as defined by the end-to-end network and security policies in place. Should you want a solution that will provide this confirmation in less time and with less resources, thus decreasing operational costs, ESG suggests that you take a closer look at Forward Enterprise.