anuta networks

Closed-Loop Automation

A primer with real-world use cases

What is Closed-Loop Automation?

If you want to cook something amazing, you cannot just add all the ingredients in the cooking pan and expect the dish to come out well. You taste, ensure the consistency and ingredients are in perfect quantity and constantly remediate any differences in taste that you may encounter. That's how you cook the perfect dish.

Closed-loop automation (CLA) is a means to create a perfect network, a technique that offers the highest quality for all applications and users, securely. CLA allows you to define and design business, compliance, security, and other policies exclusive to your organization. It continually monitors all devices in the entire network - including routers, switches, load balancers, firewalls, SDN/SD-WAN controllers and other devices, continuously scans for any violation of the defined policies, generates alerts and automatically remediates the known issues. CLA enhances network security, improves network availability, and ensures network consistency by automating workflows and tedious manual tasks.

Why is Closed-Loop Automation required?

A recent survey of network practitioners revealed that 80% of unplanned network outages occur annually due to network configuration changes. Only a mere 3% get rectified before they cause network disruptions. These disruptions also result in a loss of more than $46 million annually. Modern-day networks generate thousands of alerts per device, resulting in a deluge of notifications, making proactive management arduous. Imagine the complexity when this manifests itself in a multi-vendor network. Consequently, there is a need for assurance, automation, along with abstraction and Closed Loop Automation to improve network management and lower the associated OpEx. The result is a network that is more consistent, predictable, and reliable with an increased awareness of network behavior.

How does Closed-Loop Automation work?

Essential components of Closed-Loop Automation

Collection & Monitoring Framework

Closed Loop Automation comprises of multiple layers. One of the primary requirements is a strong collection and monitoring framework. Operational and performance data from multiple data sources such as SNMP, Streaming Telemetry, SNMP traps, syslog, NetFlow, sFlow, and others are extracted to provide deep insights into the network behavior. CLA framework uses these insights to identify violations and pattern mismatches.

An effective collection engine ingests multiple data sets to provide a foundation for a comprehensive monitoring framework. It allows the CLA framework to choose the right data source based on the network requirements such as latency and throughput. A modern stack with a provision to queue messages to meet any number of contingencies helps to maintain a high availability and ensures CLA does not miss vital information.

The presentation of the collected data is equally vital. A unified view of alarms, performance-related data derived from multiple data sources, offers NetOps teams with a single pane of glass to meet all their monitoring requirements. An intuitive and customizable user interface with the dashboards providing insightful data at a region, network, device and interface level, offers NetOps an opportunity for initial triage to in-depth troubleshooting and remediation.

Collection and monitoring are essential to provide an always-on compliance framework. It can provide critical capabilities such as PSIRT to notify vendor vulnerabilities and hardware EOS/EOL, insights on license expiry, and renewal and information on organizational compliance policies.

Alerts, Notifications & Remediation

The collection of operational and performance data alone does not serve the needs of the NetOps teams. The higher the visibility into the network, the faster it helps to ascertain the availability, performance, and user experience to troubleshoot and fix problems as they arise. Alerting & reporting plays an equally important role in NetOps to speed remediation.

Often the same issues are reported frequently. CLA provides a framework to identify and document such issues and define a baseline network behavior. The collection and monitoring framework continuously validates baseline behavior with the current status of the network and scans for deviations. Any violation triggers an action. For known issues, the CLA framework can trigger predefined corrective actions automatically and rectify the problem instantly. A detailed report of all the steps taken is also made available.

For problems where the solution is complicated and must involve human intervention, the framework can alert and notify the stakeholders through email, SMS, Slack, and many other mediums.

DevOps/NetOps

CLA solution could be massive and complicated for which organizations have to leverage open source solutions and may even need to custom build a few elements in-house. They may also need to procure third-party tools and libraries. Integrating all these varied components into a single seamless, scalable solution coupled with periodic maintenance and upgrades, require significant DevOps activities.

CLA use cases change over time, as the products and markets mature. Use cases developed during inception tend to become outdated. New protocols, procedures, and interfaces require regular upkeep and constant code modifications. A more practical solution is to build models and templates that can be enhanced and extended to suit business requirements as and when they change.

To develop a CLA framework specific to an organization's business and network needs requires a strong DevOps culture, which is essential to respond quickly to ever-changing market dynamics.

Integration with ecosystem

A robust CLA solution is not limited to device and service automation. It must also automate the method of procedures (MOPs). MOPs include not only network operations but also business processes such as approval flows, operation sequence, and time of day executions. A CLA framework should also enable administrators and architects to incorporate all these various features and provide an end-to-end business policy.

CLA frameworks must integrate with northbound entities such as ticketing, billing, and ITSM solutions (ServiceNow, BMC Remedy, Jira, and others) and southbound entities such as devices, SDN/SD-WAN controllers and cloud technologies (AWS, GCP). An optimal CLA framework leverages exhaustive open APIs to integrate with any north or southbound elements. The framework should also be bidirectional so that it can be triggered by the operator or via the alerts from the infrastructure.

Multi-vendor coverage

Whether intentional or accidental, the typical network infrastructure is a multi-vendor environment. While some organizations try to standardize on one vendor, they often end up with at least three or four vendors due to business or technical needs. Multi-vendor also avoids vendor lock-in and results in Capex savings; however, they can kill network automation aspirations.

A robust CLA framework should support various attributes such as CLI, NETCONF, API, REST CONF, and YANG models.

Besides efficient provisioning, CLA must be proficient at collecting operational metrics using other formats such as SNMP, SNMP Traps, Syslog, sFlow, NetFlow as well as Streaming Telemetry. Many homegrown automation tools support a subset of vendors and formats, resulting in islands of automation that are difficult to sustain. That's why a robust CLA framework must support legacy vendors as well as new vendors to support agility and improved productivity.

Closed-Loop Automation with Anuta ATOM Platform

Anuta ATOM platform has a feature-rich, closed loop automation framework that can be utilized in all the above use cases. Anuta ATOM platform consists of the following features.

Low-code automation framework coupled with easy and intuitive graphical user interface enables rapid development of even the most complex CLA use cases.

Support for 150+ platforms across 45+ vendors. So, rest assured as most devices in your network would already be supported.

Open APIs enables ATOM platform to easily integrate with all network and business elements such as Ticketing/Billing or ITSM solutions. So, you can integrate ServiceNow or BMC remedy with ATOM CLA platform.

ATOM has a wide collection framework that can collect data from a variety of sources- SNMP, Streaming telemetry, Syslog or SNMP Traps to provide the CLA framework with all the required information

Workflow automation can be easily integrated with closed loop automation to achieve complete end-to-end automation with automatic remediation and instant alerts.

Closed-Loop Automation use cases

Security

Enforce Compliance 24x7

Compliance and security have become a top-of-mind thing for executives and leaders of every organization as it plays a crucial role in defining and consistently enforcing compliance policies.

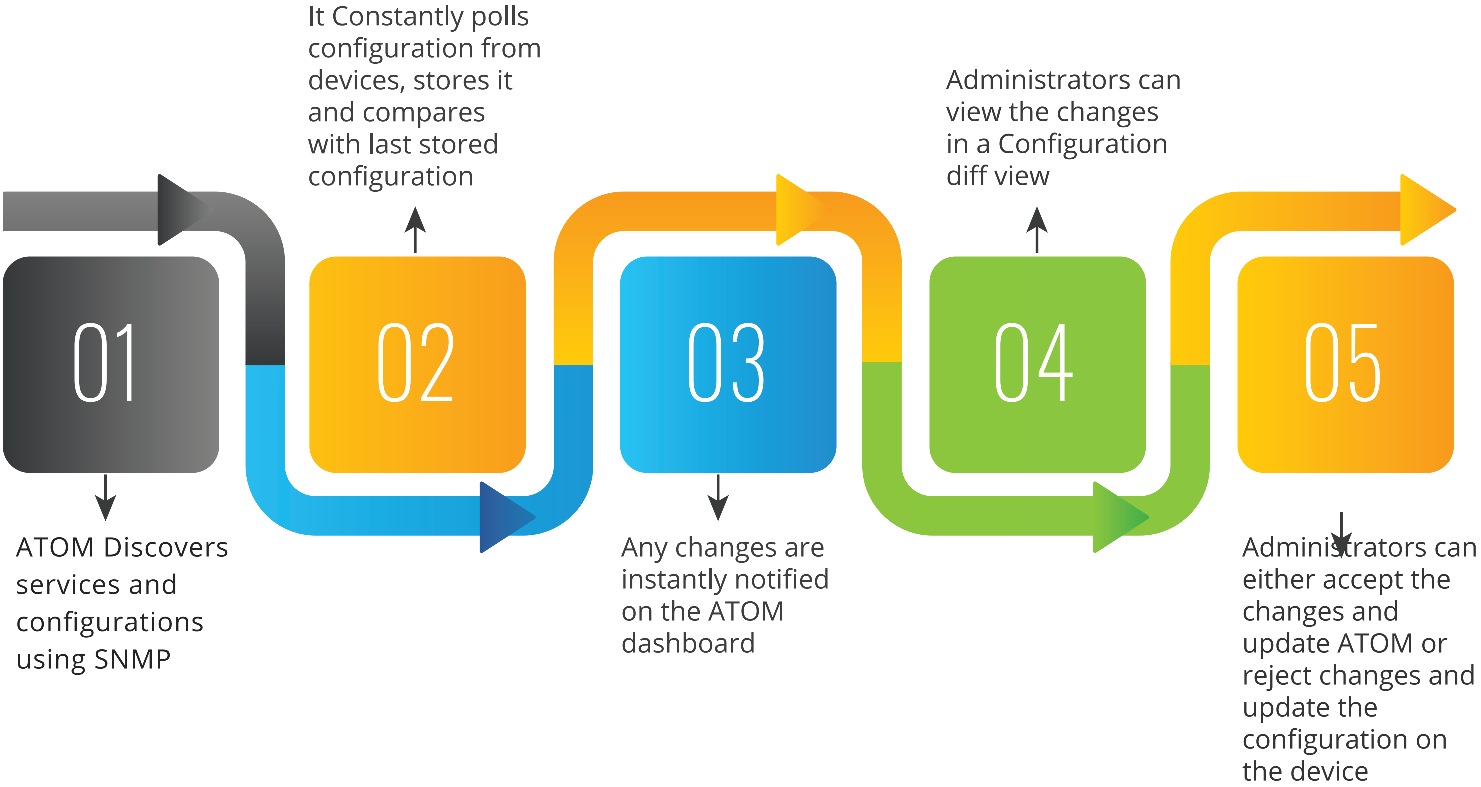
Your organization may have to adhere to many policies such as HIPAA, PCI, SOX, or even internal business policies as business and network security is a significant challenge today. NIST, CIS, and ISO 27001 provide essential security control mechanisms to prevent data breaches, cybercrime, and network frauds.

CLA enables organizations to define these policies and comply with the standards. It allows administrators to define specific formats for mandatory network configurations. It continually monitors all devices to detect any inconsistencies and violations of the defined configurations. If found, CLA can take corrective actions and restore the configurations to the desired format. Hence it is essential in enforcing an always-on compliance environment.

Detect Configuration drift

Preventing unwarranted configuration changes to devices either by a third-party application or manually is critical to ensure security and compliance of the network. CLA framework helps detect unauthorized modifications and reverts to the desired configuration automatically. Constant monitoring for changes prevents fraudulent activities and enhances network security.

Configuration drift with ATOM



ATOM Discovers services and configurations using SNMP

It Constantly polls configuration from devices, stores it and compares with last stored configuration

Any changes are instantly notified on the ATOM dashboard

Administrators can view the changes in a Configuration diff view

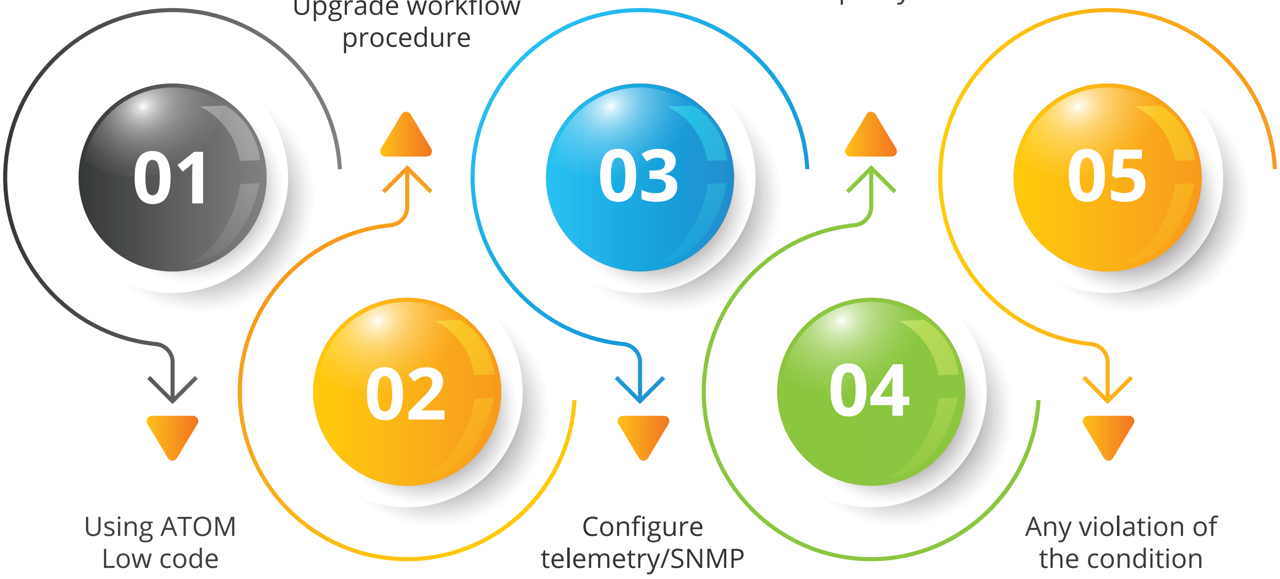
Administrators can either accept the changes and update ATOM or reject changes and update the configuration on the device

Device OS Compliance & Upgrade

Software defects and issues in OS are a significant security threat. For network administrators, it is essential to upgrade the device OS to the recommended versions. However, monitoring all devices in the network is not only highly demanding, but even the upgrade procedure is quite laborious.

To help this, CLA periodically analyzes the OS versions in all devices in the network and notifies and even automatically upgrades violated devices. To prevent downtimes and minimize the impact on the network, CLA has the flexibility to schedule OS upgrades during off-peak hours.

Device OS compliance with ATOM



01 Using ATOM Low code designer, create a complete end to end OS upgrade workflow including pre-checks, post-checks and approval flows

02 Create the following CLA policy

a. Condition: Check for recommended OS version on all devices

b. Action on violation: Run pre-defined OS Upgrade workflow procedure

03 Configure telemetry/SNMP sensors to collect current OS versions from all devices and store in the time-series database

04 CLA engine continually compares the current OS version stored in TSDB with defined OS version in the CLA policy

05 Any violation of the condition triggers OS upgrade workflow automatically

Network Management

Remediate BGP Flap issue with ATOM

Recursive routing failures due to continuous BGP flaps or interface connectivity issues can result in unpleasant customer experiences and subsequent revenue loss. It is quite an arduous task for network administrators to detect such erratically occurring events from network alarms and take remediation actions.

CLA allows threshold-based definitions to detect BGP neighbor state changes across the network, based on SNMP or streaming telemetry data. The relevant remediation actions defined within the CLA framework can be direct action on the network or controlled remediation by triggering a notification towards an ITSM tool, informing the network administrators of the event.

Remediate BGP Flap issue with ATOM

01

Customize out-of-box workflows to shutdown neighbors to suit your network. Out of box "shutdown neighbor," workflow performs 2 actions.

a. If there are more than 5 flaps in 1 hour, then ATOM notifies administrators through ATOM dashboards/ServiceNow/BMC Remedy or any other ITSM solution

b. If there are more than 10 flaps in 1 hour, then ATOM not only notifies administrators but also shuts down the neighbor on approval from the administrator.

02

Define CLA Policy

a. Condition: Using ATOM DSL, define a condition to check if BGP state changes to anything other than established more than 5 times in 1 hour

b. Action on violation: Run "shutdown neighbor" workflow

03

Configure SNMP to receive the BGP peer state. The data is analyzed every 1 minute for a sliding window of 15-minute data to avoid spikes.

04

CLA engine checks every hour for the number of changes to BGP peer state

05

Based on the number of BGP flaps, an appropriate workflow is triggered.

IP SLA Thresholds

Latency and jitter are essential parameters to determine application quality in networks. And IP SLA is one of the ways of determining network performance.

CLA allows threshold definitions to match the expected latency, jitter, RTT, and drops based on ICMP, TCP, UDP, RTP, and HTTP statistics. With continuous polling of network performance data along with scheduled command outputs and comparison at regular intervals in a sequential workflow, CLA can provide granular performance data. This helps CLA to reroute traffic to other links that meet SLA requirements, thus ensuring service continuity.

Monitor Bandwidth Utilization with ATOM

Most devices have an upper limit on the interface bandwidth. High bandwidth causes massive packet drops, high CPU utilization, and most importantly, lowers the customer experience. Managing high interface utilization is essential, and most companies have standard procedures to deal with such a situation. They usually have standard methods of rerouting some of the traffic to reduce interface utilization.

CLA can be used to apply the standard methods automatically. It can continuously monitor for bandwidth thresholds and any violation that applies to predefined procedures to remediate the issue.

Monitor Bandwidth Utilization with ATOM

01

Customize out-of-box workflow to monitor and act upon high bandwidth utilization to suit your network. Out of box “bandwidth utilization” workflow performs the following actions.

a. Condition Info: Notify operations team with severity level info

b. Condition Warn: Notify the Ops team severity level Warn & request for upgrade the link or Change QoS policy

c. Condition Severe: Notify Ops team with severity level Critical & shut down the interface, and request for upgrade the link or Change QoS policy

d. Condition Normal - clear alarm and revert QoS

02

Configure Telemetry or SNMP sensors to receive input data rate, output data rate and bandwidth details of the interface. Utilization computed as (input rate + output rate)/bandwidth

03

Create the following CLA policy

a. **Rule 1:**
   i. Condition: utilization > 70%
   ii. Action: run “bandwidth utilization” workflow with condition info

b. **Rule 2:**
   i. Condition: utilization > 75%
   ii. Action: run “bandwidth utilization” workflow with condition warn

c. **Rule 3:**
   i. Condition: utilization > 80%
   ii. Action: run “bandwidth utilization” workflow with condition severe

d. **Rule 4:**
   i. Condition: utilization < 70%
   ii. Action: run “bandwidth utilization” workflow with condition normal

04

Configure CLA engine to analyze the data every 1 minute with a sliding window of 5 minutes to avoid spikes

05

On any violation, CLA engine calls pre-defined workflow

Device Management

Monitor and Remediate Interface drops

Unpredictable behavior from networks is always challenging to combat. One of the common issues that fall into this category is interface drops. For example, interface drops on one of the primary WAN links during peak hours could impact the performance resulting in revenue loss.

CLA can be tuned to monitor device interface drops at regular intervals to emit an alert or take actions such as moving the traffic to the redundant links, shutting down the interface, applying ACLs, or other operations.

Monitor and remediate interface drops with ATOM

01

Customize out-of-box workflow to monitor and act upon high bandwidth utilization to suit your network. Out of box “bandwidth utilization” workflow performs the following actions.

a. Condition Info: Notify operations team with severity level info

b. Condition Warn: Notify the Ops team severity level Warn

c. Condition Severe: Notify Ops team with severity level Critical & shutdown the interface after approval from the administrator

d. Condition Normal - clear alarm and unshut interface

02

Configure Telemetry or SNMP sensors to receive input drop pkts, output drop pkts, input total pkts and output total pkts details of the interface. Utilization computed as (Input-drops+output-drops) / (Total packets sent + received)*100

03

Create the following CLA policy

a. **Rule 1:**

   i. Condition: utilization > 5%
   ii. Action: run “bandwidth utilization” workflow with condition info

b. **Rule 2:**

   i. Condition: utilization > 10%
   ii. Action: run “bandwidth utilization” workflow with condition warn

c. **Rule 3:**

   i. Condition: utilization > 25%
   ii. Action: run “bandwidth utilization” workflow with condition severe

d. **Rule 4:**

   i. Condition: utilization < 5%
   ii. Action: run “bandwidth utilization” workflow with condition normal

04

Configure CLA engine to analyze the data every 1 minute with a sliding window of 5 minutes to avoid spikes

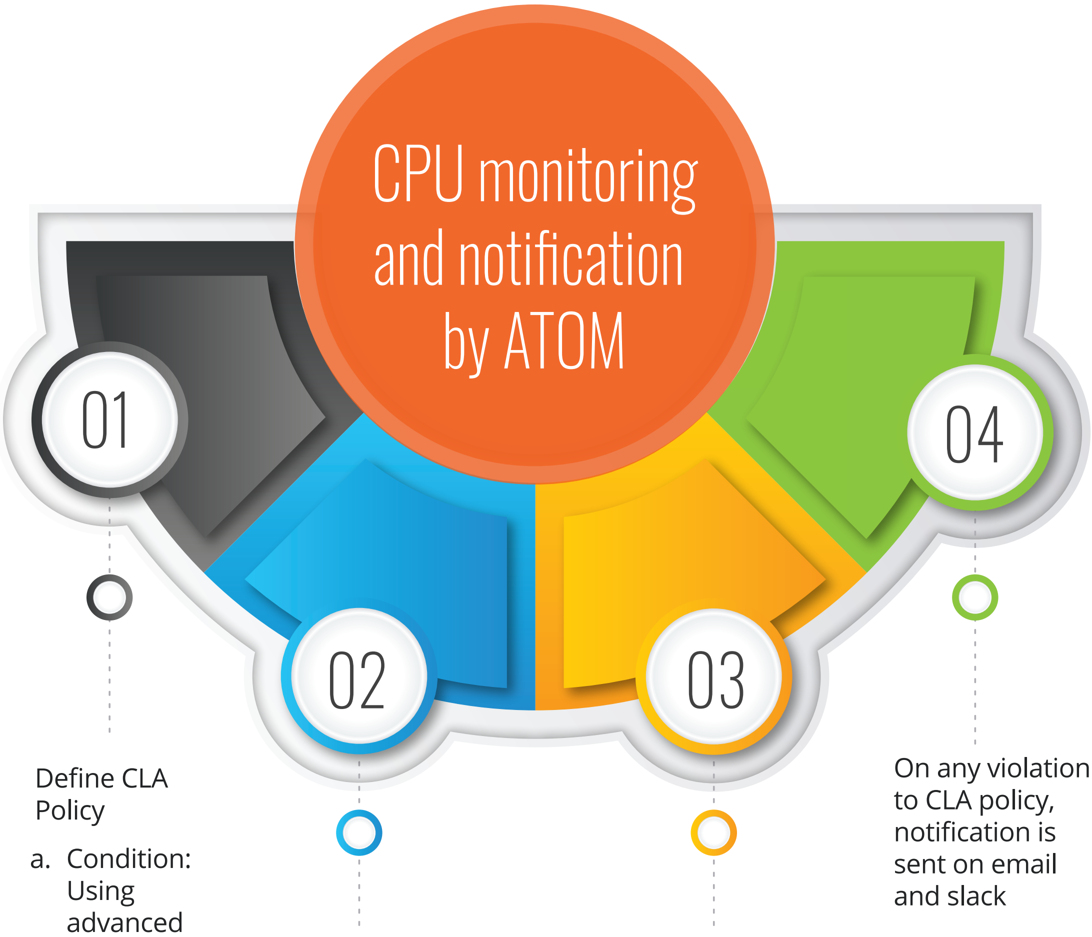
05

On any violation, CLA engine calls pre-defined workflow

CPU monitoring & notification

High CPU usage leads to adverse effects on the device. Higher drops of packets and increased errors on interfaces are a few of the visible impacts. Lowering CPU usage becomes essential to maintain network consistency, but any delay in bringing CPU under control could lead to significant loss of customer experience.

CLA can monitor for high CPU usage by devices and take appropriate remediation steps to reduce the usage.



01 Define CLA Policy

a. Condition: Using advanced CLA, define condition to check if CPU % is > 70%

b. Action on violation: Enter email and slack notification values

02 Configure SNMP/telemetry sensors to receive CPU information.

03 Configure CLA engine to check every 5 minutes the average CPU

04 On any violation to CLA policy, notification is sent on email and slack

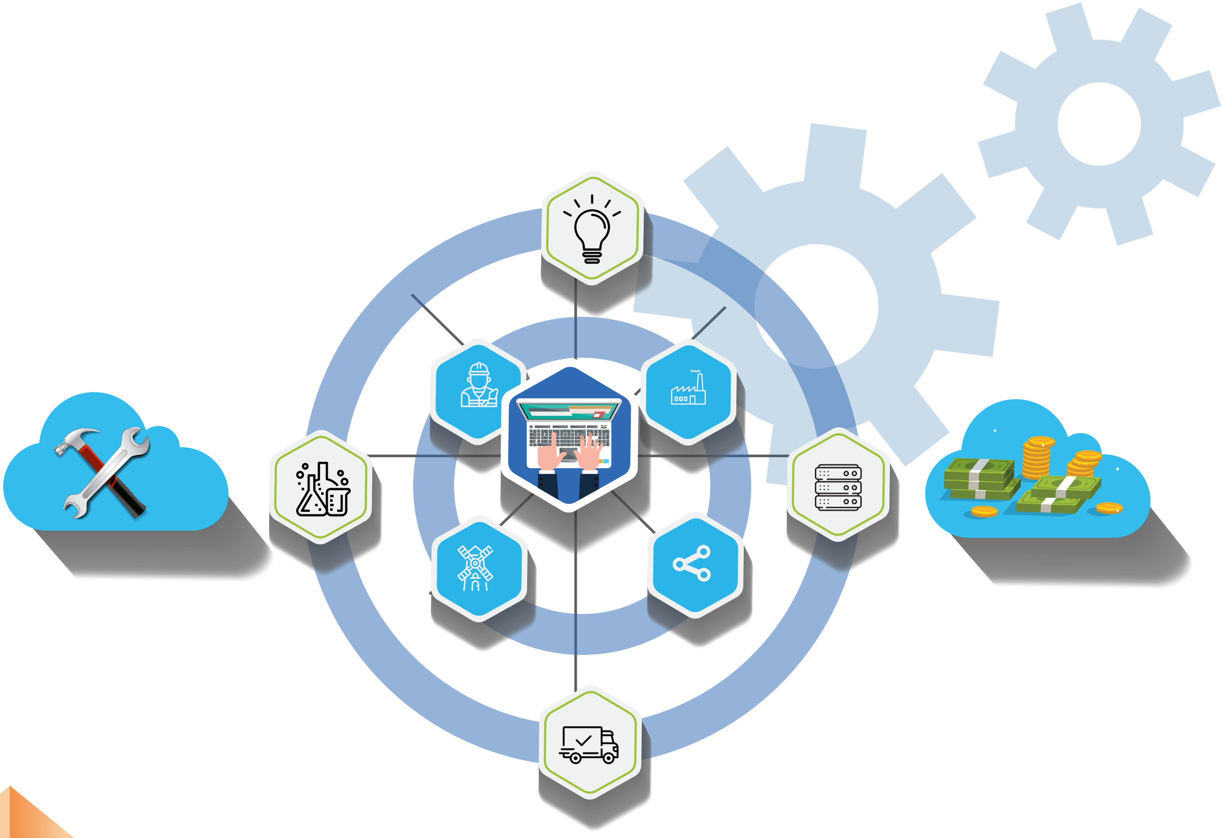
Example Use Case with Anuta ATOM Platform

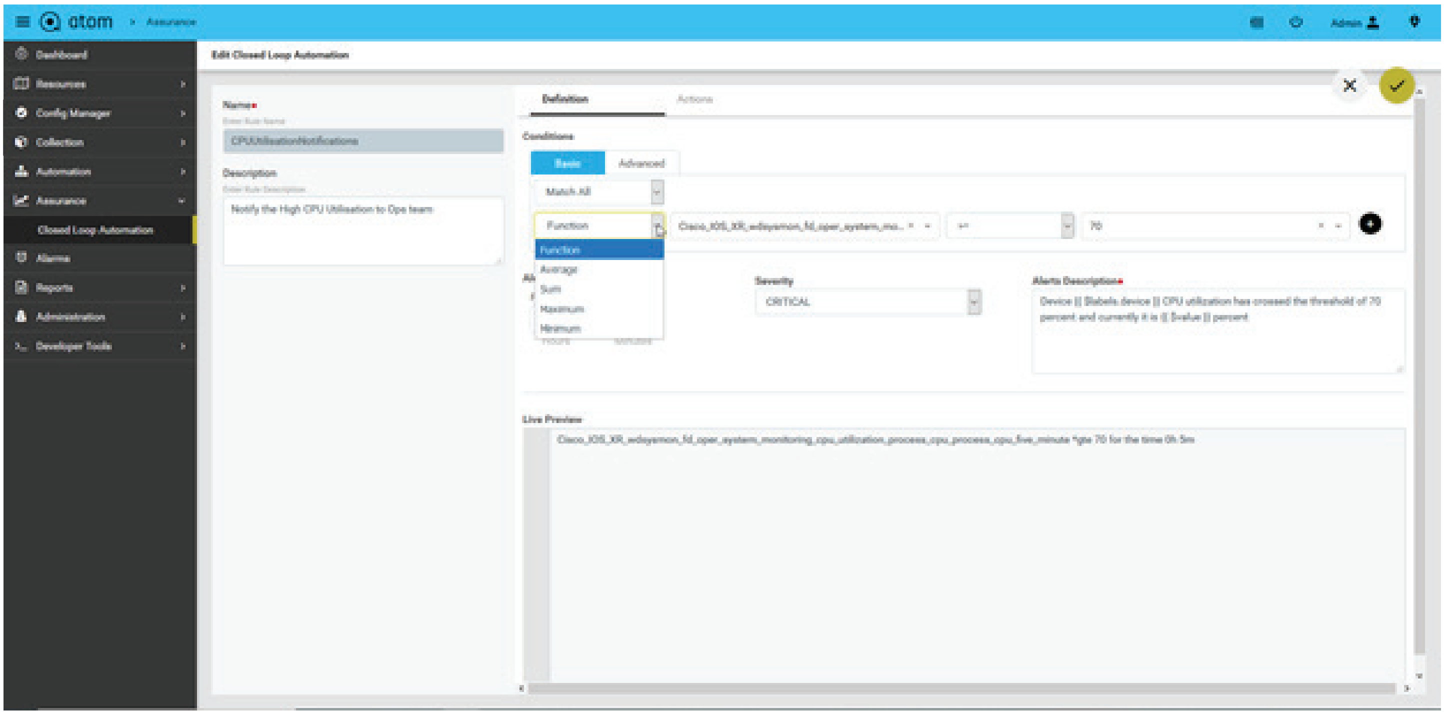
Here is a walk-through of a closed-loop automation use case with Anuta ATOM platform.

Notify High CPU Utilization to Operations Team

The goal was to identify network devices with CPU higher than the threshold limit of 70%. If any device breaches 70% CPU limit, a notification is sent to operations team by email and slack as well as a remediation action is taken automatically. The remediation action in this case is to run a workflow inside ATOM platform. This predefined workflow will take approval from network administrator to shut down one of the interfaces in the router to reduce total packet processing in the device to reduce CPU.

To enable CLA in ATOM platform, one needs to configure the condition to check and the action to trigger on violation of the condition.





Edit Closed Loop Automation

Name: CPUUtilisationNotifications

Description: Notify the High CPU Utilisation to Ops team

Conditions

Match All

Function | Cisco_IOS_XR_wdsysmon_fd_oper_system_mo... | >= | 70

Severity: CRITICAL

Alerts Description: Device {{ $labels.device }} CPU utilization has crossed the threshold of 70 percent and currently it is {{ $value }} percent

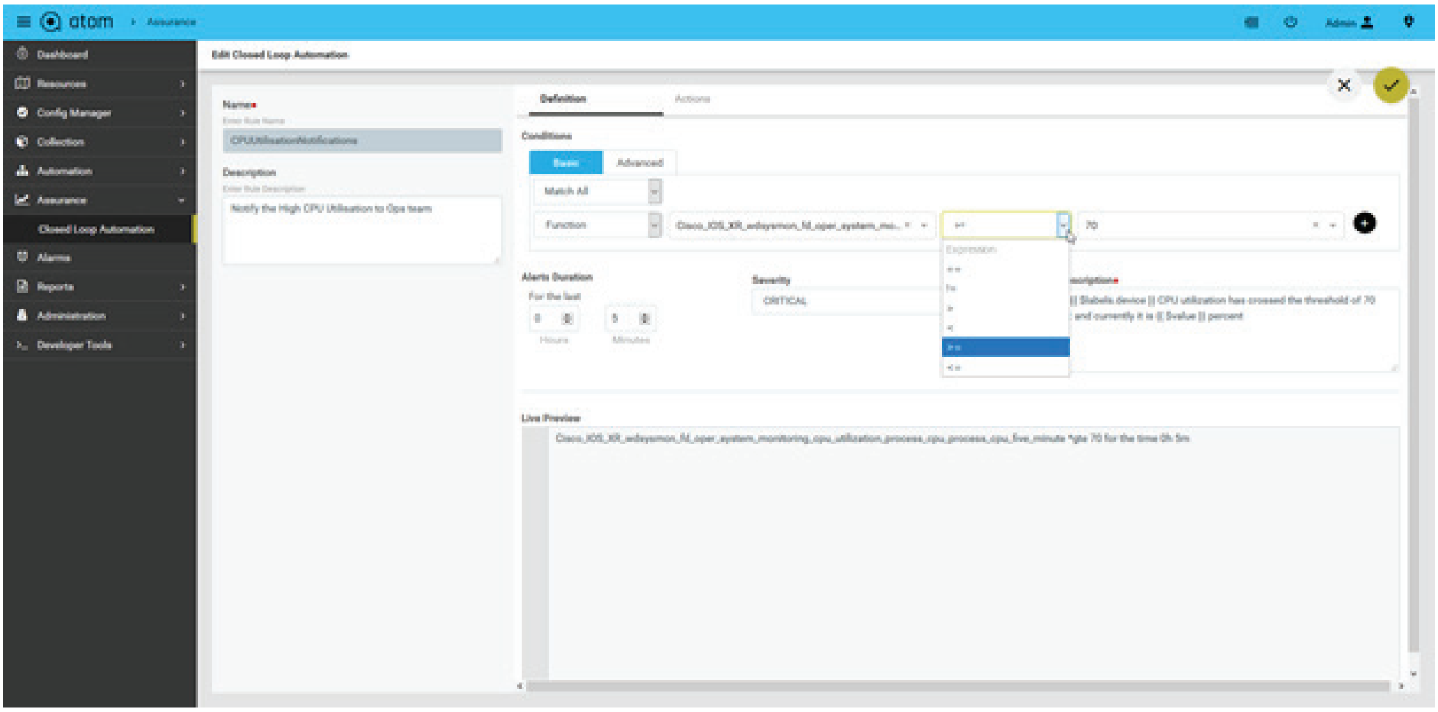
Live Preview

Cisco_IOS_XR_wdsysmon_fd_oper_system_monitoring_cpu_utilization_process_cpu_process_cpu_five_minute *gte 70 for the time 0h 5m

In the ATOM UI above, the condition rules are being defined to monitor in the network. One can define multiple rules, such as Function, Average, Sum, Maximum or Minimum.

For this particular use case, Function is selected as we are interested in the CPU function. Next, select appropriate telemetry profile. The selected telemetry profile collects CPU information from all network devices in the network.

The next step is to define the expression to enforce.



Edit Closed Loop Automation

Name: CPUUtilisationNotifications

Description: Notify the High CPU Utilisation to Ops team

Match All

Function | Cisco_IOS_XR_wdsysmon_fd_oper_system_mo... | >= | 70

Alerts Duration: For the last 0 Hours 5 Minutes

Severity: CRITICAL

Description: {{ $labels.device }} CPU utilization has crossed the threshold of 70 and currently it is {{ $value }} percent

Live Preview

Cisco_IOS_XR_wdsysmon_fd_oper_system_monitoring_cpu_utilization_process_cpu_process_cpu_five_minute *gte 70 for the time 0h 5m

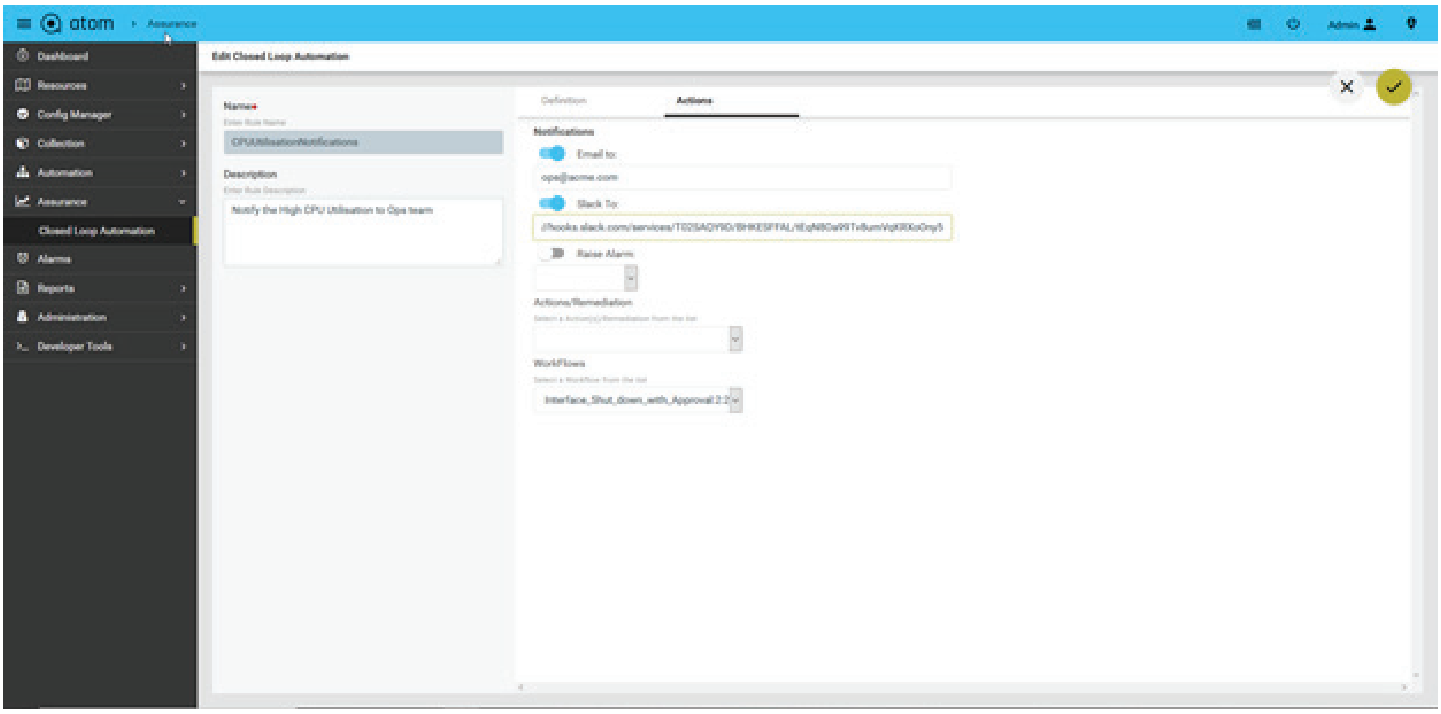
Since we are interested to monitor devices with CPU higher than 70%, we choose “>=” expression.

Finally, enter the CPU threshold value, which is 70% in our case.

One can also customize sampling interval.

The “Alert Duration” represents how frequently ATOM needs to sample CPU values.

The next step is to configure actions to be taken during the breach of threshold.



Edit Closed Loop Automation

Name*
Enter Rule Name
CPUUtilisationNotifications

Description
Enter Rule Description
Notify the High CPU Utilisation to Ops team

Notifications
Email to:
ops@acme.com
Slack To:
//hooks.slack.com/services/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
Raise Alarm

Actions/Remediation
Select a Action(s)/Remediation from the list

WorkFlows
Select a Workflow from the list
Interface_Shut_down_with_Approval:2.2

As can be seen in the screenshot above, a few notify conditions as well as remediation actions are set. In case the CPU goes above 70%, ATOM is configured to send an email to the operations team and also to send a slack notification to the operations slack group. The notification will be sent as soon as threshold is breached. Also, ATOM will initiate a workflow to shut down the interface. This workflow will raise a ticket on an ITSM solution (ServiceNow in this case). It will also look for approval from network administrator. On receiving the approval, ATOM will shut down the interface as defined in the workflow.

The above example depicts how closed-loop automation can be configured in a simple and intuitive manner within the Anuta ATOM platform. It is not limited to conditions and actions defined in the GUI alone. Complicated conditions can be defined using advanced CLA options. ATOM's domain specific language (DSL) can be leveraged to define complex conditions across multiple sensors and devices. DSL can also be used to define additional actions to what is provided out-of-box.

Closed-Loop Automation - a prequel to Intent based networking

Lately, there is a lot of buzz about intent-based networking or IBN. Vendors of all stripes and domains are coming up with their opinion and viewpoints on how they perceive IBN will pan out. It is expected to make monitoring and management of the present complex IT networks simple, and reduce human errors and mean time to repair. It will accelerate the path to digital transformation and enhance the customer experience.

Intent-based networking starts with an Intent. But what is an Intent? Simply put, intent can be defined as the benefit one expects to receive from the network. For instance, payroll app should work without downtime. The Intent-based networking systems will assess and evaluate the intent and, after multiple levels of translations, break it down to appropriate device commands. IBNS obtains constant feedback from the network to ascertain that the intent is never violated.

CLA is an essential component of intent-based networking. In order to constantly enforce the policy defined by the intent, any IBN solution must have a powerful CLA at its core which continuously receives feedback from the network and monitors that status of the intent. Any violation of the intent will trigger automated alerts and remediation. It ensures the intent policy is upheld constantly. Therefore, prior to achieving intent based networking, it is important to understand closed-loop automation perfectly since it is a stepping stone to IBN.

Get a feel of Closed-Loop Automation!
Contact Anuta Networks for a FREE DEMO today!