



The DNS Automation

COOKBOOK

About this eBook

With its easy to use automation platform, BlueCat has made streamlining DNS management as simple as following a recipe. This e-book provides “recipes” for DNS automation using BlueCat infrastructure and the BlueCat Labs repository on GitHub (<https://github.com/bluecatlabs>). We’ve grouped these automation tasks into an initial basic category (which covers standard DNS tasks) and an intermediate category (which covers more in-depth use cases).

Basic DNS automation:

- ▶ Adding host or alias records
- ▶ Selective deployment
- ▶ Bulk imports
- ▶ Reporting

Intermediate DNS automation:

- ▶ Workflow scheduling with Rundeck
- ▶ Connections to shifting third-party domains
- ▶ Failover for BlueCat Address Manager, DNS, and DHCP server (BDDS)
- ▶ IP address allocation and deallocation in the cloud
- ▶ Adding DNS records with ServiceNow

Advanced DNS automation:

- ▶ Automating IP Address Allocation and Deallocation in the Cloud
- ▶ Promoting automation workflows from GitLab to production

Leveraging BlueCat Infrastructure for DNS Automation

By implementing BlueCat solutions, you’ve already laid the foundation for managing your DNS enterprise through automation. Now, it’s time to take advantage of those capabilities to save time, increase efficiency, and implement self-service.

With BlueCat’s core infrastructure solution, you are centrally managing DNS services and automating many of the back-end DNS management functions you used to implement manually. This e-book is a guide to extending those benefits further.

Let’s get started!

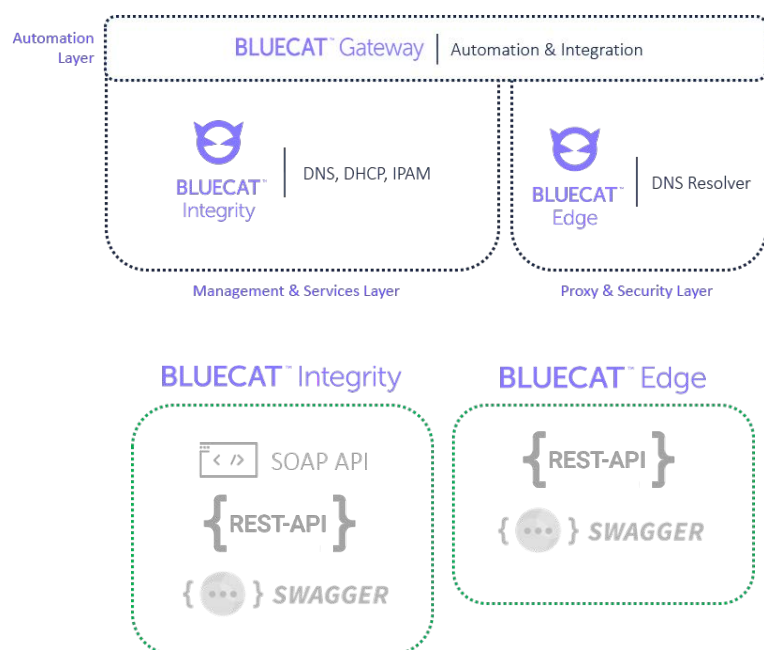


Table of Contents

Appetizers.....	4
Automating Adding Host or Alias Records.....	4
Automating Selective Deployment.....	5
Automating Bulk Imports.....	6
Automation Reporting.....	8
Main Courses.....	10
Automating Workflow Scheduling with Rundeck	10
Automating Failover for BDDS.....	12
Automating Connections to Shifting Third-Party Domains.....	14
Automating IP Address Allocation and Deallocation in the Cloud	16
Automating Adding DNS Records with ServiceNow	18
Managing Options & Roles.....	19
Promoting automation workflows from GitLab to production.....	21

Automating Adding Host or Alias Records

Adding devices to a network is a basic, day-to-day responsibility of any DNS administrator. New computers, IoT devices, servers, and other equipment are constantly changing the network topography. Admins must assign IP spaces to all of them.

While not a difficult task, adding host or alias records is time-consuming. An admin must:

- ▶ Configure the device, assigning permissions and characterizing it in the context of its role on the network;
- ▶ Assign a top-level domain to the device;
- ▶ Add a host or alias record to connect the domain to an IP address;
- ▶ Set an IP address for the host or alias to connect with; and
- ▶ Validate the entire chain, checking for errors and confirming that the device is operating correctly.

When you're adding hundreds of devices to the network every day, this process can be a significant drain on admin resources.

Two certified Gateway workflows on GitHub automate [host record](#) or [alias record](#) management, making adding, updating, or deleting these records much easier.

How it Works

- ▶ Admins set up default configurations for different device types.
- ▶ To set up a host or alias record, users select from one of the pre-built configurations in a dropdown menu.
- ▶ Users can assign a specific IP address, or the IP address can pre-populate based on the chosen configuration.
- ▶ Gateway does all the necessary operational checks automatically on the backend.

The screenshot shows the BlueCat Gateway web interface. At the top, there's a header with the BlueCat logo and 'BLUECAT GATEWAY™'. On the right, it shows 'BAM 10.244.107.88' and 'portalUser'. A sidebar on the left lists 'AVAILABLE ACTIONS' with a dropdown menu open showing 'Add Host Record Example' selected. The main form area is titled 'ADD HOST RECORD EXAMPLE' and contains several fields: 'Configuration' (dropdown menu showing 'config-1'), 'View' (dropdown menu showing 'com'), 'Zone' (dropdown menu showing 'example'), 'IP Address' (text input showing '10.0.0.23' with a green checkmark), and 'Hostname' (text input showing 'printer1'). Below these fields is a 'Deploy Now' checkbox and a blue 'SUBMIT' button. At the bottom, there's a footer with '© 2018 BlueCat Networks, All rights reserved.' and 'v 18.10.2'.

That's it! Because this whole process happens through the Gateway interface, it's also more secure. Admins can offer pre-built configurations to outside users without granting access to core network infrastructure.

Automating Selective Deployment

When admins change anything in their DNS, DHCP, or IP address management (IPAM) architectures it can take time for associated records to filter through the entire network. Usually, admins will schedule a mass change for a low-traffic period and then send all their changes out at once.

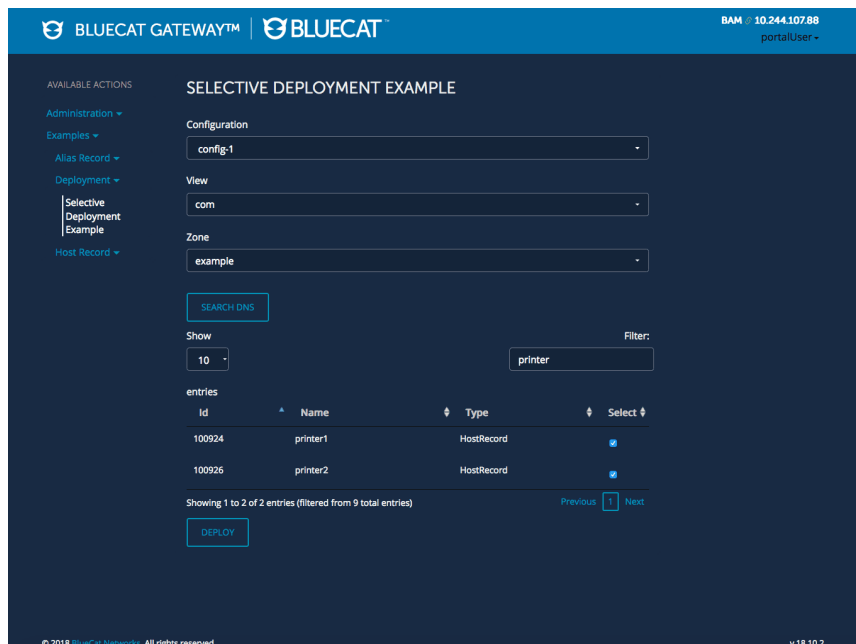
On occasion, admins need to do just one thing. It might be an immediate configuration fix, a single device that needs to be on the network right away, or minor changes on the fly.

This is where selective deployment comes in. Selective deployment also makes stand-up and tear-down of cloud infrastructure much easier by directing specific DNS records to deploy as needed.

How it Works

Using the [deployment certified workflow on GitHub](#), Gateway users can deploy single changes rather than an entire batch. (Note: this workflow is for DNS only.)

Using the selective deployment workflow is easy. Just select the specific change you want to deploy through the Gateway interface, and deploy it. Simple!



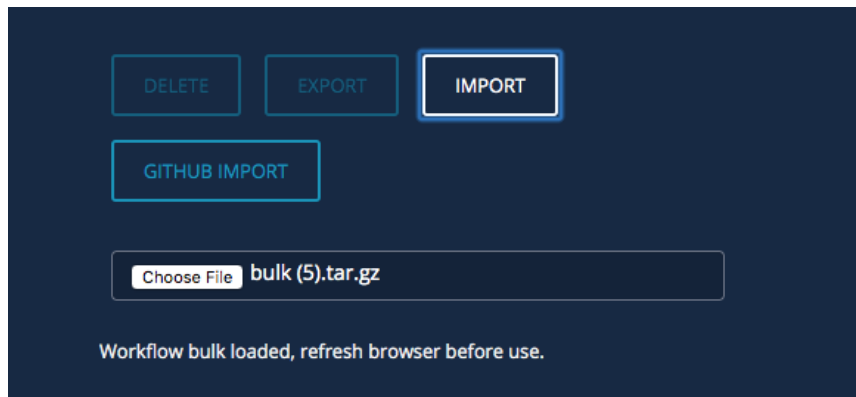
Automating Bulk Imports

Address Manager is our IPAM solution for centrally managing every connected device on your network. While Address Manager has basic import functionality, you're somewhat limited by how many records and the type of data that you can import. Whether it's IP addresses, resource records, or user-defined function values, any object that can be created by BlueCat's API can be bulk imported using Gateway. And while Address Manager is limited to 5,000 records at a go, Gateway's only limit is your computing power. A million records? It can be done.

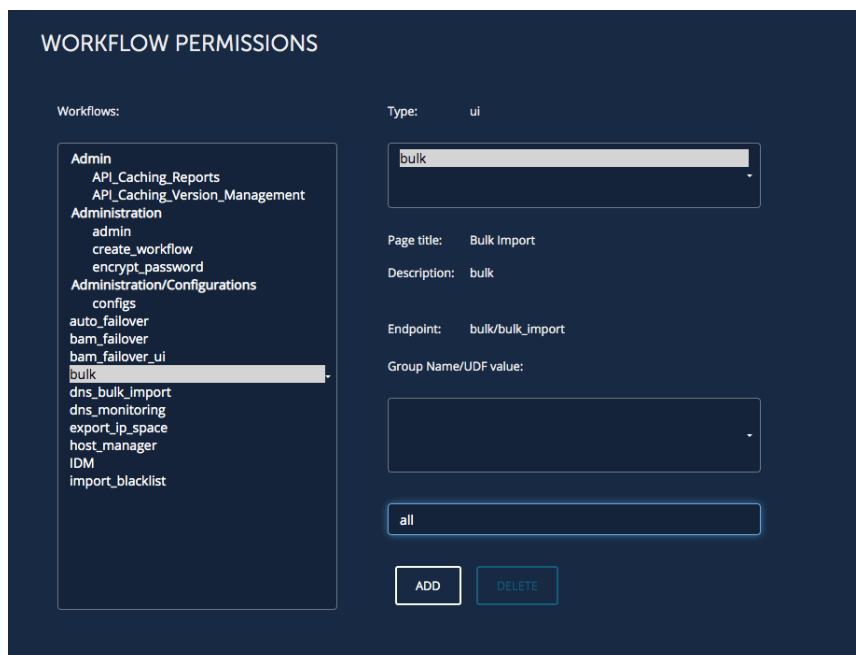
You can also use Gateway to handle more advanced aspects of bulk imports. This includes dependencies and errors, validations and rules, dynamic values and values from multiple sources, automatic and scheduled uploads, tagging, and other advanced metadata.

How it Works

In Gateway, select Workflow Export/Import under the Administration tab. Choose your workflow file to import. The latest community workflows for bulk extraction of users and MAC addresses are available on our BlueCat Labs GitHub repository.



Then, go to Workflow Permissions to set your permissions.



Under Bulk Import, select your configuration and the file to load, such as a spreadsheet of networks. Easy!

BULK IMPORT

Upload your file in CSV format

Configuration

Webinar

Bulk Add File

Choose File example_1.csv

[Template Download Link](#)

PROCESS

While importing host records, Gateway will help identify and handle invalid data from duplicates and typos.

	A	B	C	D	E	F	G	H	I	J	K
1	fqdn	ip	ttr	result							
2	host1.example.com	10.0.0.2	3600	Added host1.example.com							
3	host2.example.com	10.0.0.3	3600	Unable to add host record. BlueCat API exception:Server raised fault: "Invalid destination Zone."							
4	host1.example.com	10.0.0.2	3600	Unable to add host record. BlueCat API exception:Server raised fault: "Duplicate of another item"							
5											
6											
7											
8											

Automating Reporting

Easy, glanceable DNS reports are essential to get information on the health of your network. But manual reporting means logging into Address Manager to navigate your IP space and looking up subnets or IP addresses individually. It's time-consuming. And the information could be outdated by the time you're done.

More and more stakeholders outside your IT enterprise want to use Address Manager for informational purposes about IP spaces or addresses. They may have a need to know, but at the same time, granting broad access to Address Manager for those who aren't DNS admins presents potential security and operational risks. For example, information security personnel investigating an incident may want to see basic information about a suspicious IP address, such as what network it belongs to or where it's located. But they don't really need or want Address Manager access; they have no need to assign addresses or create networks.

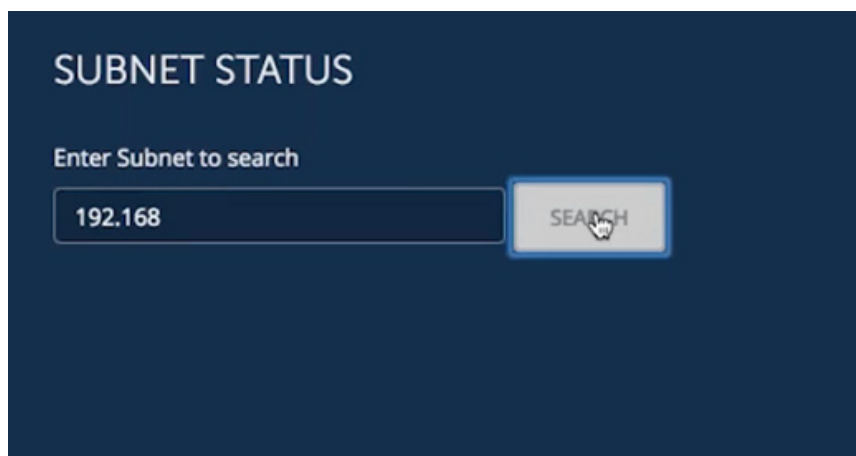
While IP information is exposed in Address Manager, it's usually displayed in the way operators need to see it—on a micro level. Outside stakeholders, on the other hand, have different needs. They usually want to see macro information such as how many IP addresses are actually allocated, how many total are free, if there's an available network, or what's available for the next address.

With Gateway, automated reporting provides a fast and simple way to generate comprehensive IP subnet information while restricting Address Manager access to only those who need it. It's ideal for a quick and thorough health check on your IP addresses.

How it Works

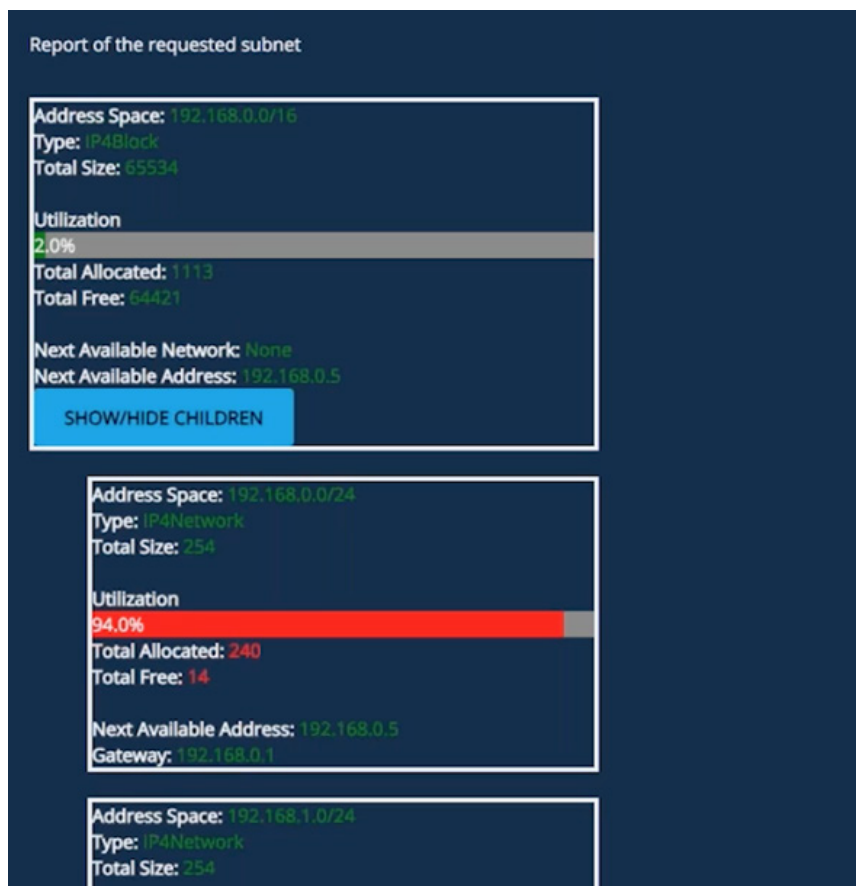
Using the [subnet status community workflow on GitHub](#), you can automate subnet status reporting in a flash.

For at-a-glance capacity management, simply enter in a subnet.



The image shows a dark blue interface titled "SUBNET STATUS". Below the title, there is a label "Enter Subnet to search". Underneath this label is a text input field containing the value "192.168". To the right of the input field is a light blue button with the word "SEARCH" in white capital letters. A mouse cursor is pointing at the "SEARCH" button.

You'll receive an instant read-out on the subnet and its entire hierarchy, children and all. Basic watermarks reported include maximum size, the next available IP address, and if there's space for a new subnet within your block. The report shows you what is actually being utilized: how many addresses are there, what the total free space is, and what the total availability is.



Child networks can be shown or hidden as needed. And reports can be customized to add fields such as location codes or any other user-defined functions on these objects. If there's some descriptive information that admins add into the network, such as location or point of contact, this information can be added and displayed in the report's easy-to-view format.

Reports can easily be emailed in HTML format.

In the next section, we'll discuss how to implement our Rundeck integration for scheduling. This could include scheduling your subnet report to run and be sent to recipients on a regular basis.

SUBNET STATUS

Enter Subnet to search

Found 61 matching subnets

Email

Report of the requested subnet

Main Courses

Automating Workflow Scheduling with Rundeck

Sometimes even automation requires an easy button. Admins might need an hourly pull of resource records or newly added IPv4 networks. Or perhaps devices are being migrated between networks and a scheduled regular clean-up of IPs or records is needed. DNS admins need the ability to schedule workflows that can run in the background without any manual prompting.

Rundeck is a free third-party tool that orchestrates automation workflows. With Rundeck, Gateway users can create on-demand or scheduled jobs, getting background tasks done without needing to manually start the process. Once an admin creates the workflow in Gateway and schedules it in Rundeck, the whole thing happens without any further management.

How it Works

The [community version of Rundeck](#) is available for free on their website. (We recommend installing it as a separate Docker container, although RPM, Debian, or Ubuntu options are also available.)

The [BlueCat Gateway plug-in for Rundeck](#) is available as a community offering on our GitHub repository.

Once in Rundeck, start a new project and click “add a step” to associate the Gateway plug-in with your Rundeck instance.

In the project, navigate to the Gateway integration, which will prompt you for your password. (Rundeck includes key storage for secure password management.) Then select the type of REST call from the dropdown menu and input the relevant Gateway endpoint.

The screenshot shows the 'Workflow' configuration page in Rundeck. At the top, there's a section 'If a step fails:' with two radio buttons: 'Stop at the failed step.' (selected) and 'Run remaining steps before failing.'. Below this is a 'Strategy:' dropdown menu set to 'Node First'. A note states: 'Execute all steps on a node before proceeding to the next node.' There's an 'Explain >' button. Further down is a 'Global Log Filters' section with an '+ add' button. At the bottom, there are buttons for 'Undo', 'Redo', and 'Add a step'.

The screenshot shows the 'gateway-rundeck' configuration form. It has a title 'gateway-rundeck' and a subtitle 'Calls gateway endpoints'. The form contains several fields: 'username' (with value 'portalUser' and hint 'login username'), 'password' (with hint 'login password'), 'URL' (with hint 'url of gateway'), 'REST call' (a dropdown menu set to 'POST' with hint 'the REST call'), 'Gateway Endpoint' (with hint 'the endpoint on the gateway , for REST endpoints please add /api/v1/ to the beginning of the end point'), and 'Additional Data' (a table with one row and one column).

Schedule your Gateway job in Rundeck by running through the prompts, selecting a cadence, and directing a failover option.

Workflow

If a step fails: ☒ Stop at the failed step. ☐ Run remaining steps before failing.

Strategy: **Node First**

Execute all steps on a node before proceeding to the next node.

[Explain](#)

Global Log Filters [+ add](#)

[Undo](#) [Redo](#) [Revert All Changes](#)

1. **gateway-rundeck** Calls gateway endpoints

```
Username: portalUser
password:
URL: https://172.16.22.32
REST call: GET
Gateway Endpoint: api/v1/configurations/
Additional Data: 1 lines
Expected Return: 1 lines
```

[+ Add a step](#)

Nodes ☐ Dispatch to Nodes ☒ Execute locally

Choose whether the Job will run on filtered nodes or only on the local node.

You can test the workflow to make sure that everything works. Then you're good to go!

RUNDECK test

#77
test_demo

Succeeded after 8s at 9:18 pm started at 9:18 pm by you 0.00:07 on 1f0310d-146f-4116-b54f-12f8132d764b 11/11

Complete: 100% 1/1 **0** Failed: **0** Incomplete: **0** Not Started: **0**

[Report](#) [Log Output](#) [Definition](#)

✓ **92bcb023aad** All Steps OK 0:00:03

gateway-rundeck OK 9:18:35 pm 0:00:02

```
[{"id":183893,"name":"unit_test_config_new","properties":{"null":"Configuration"},{"id":186199,"name":"tsig_test_config","properties":{"null":"Configuration"},{"id":186558,"name":"test_rest_api_config","properties":{"null":"Configuration"},{"id":178951,"name":"add_host_record_config","properties":{"null":"Configuration"},{"id":177931,"name":"delete_host_record_config","properties":{"null":"Configuration"},{"id":177184,"name":"update_host_record_config","properties":{"null":"Configuration"},{"id":177269,"name":"generic_ui_test_config","properties":{"null":"Configuration"},{"id":432099,"name":"mail_configuration_config","properties":{"null":"Configuration"},{"id":432122,"name":"stevens_config","properties":{"null":"Configuration"}}
```

Automating Connections to Shifting Third-Party Domains

The domains used by cloud services are rarely static. As the underlying domains for third-party services are updated, load balanced, and taken offline for maintenance, DNS must adjust so that queries resolve and SD-WAN routes correctly. Doing this manually is a significant headache and requires constant real-time vigilance—Office 365 alone uses more than 100 constantly shifting domains.

To avoid outages and maintain direct connections to trusted cloud services, your BlueCat infrastructure can receive automatic updates when the underlying domains are changed.

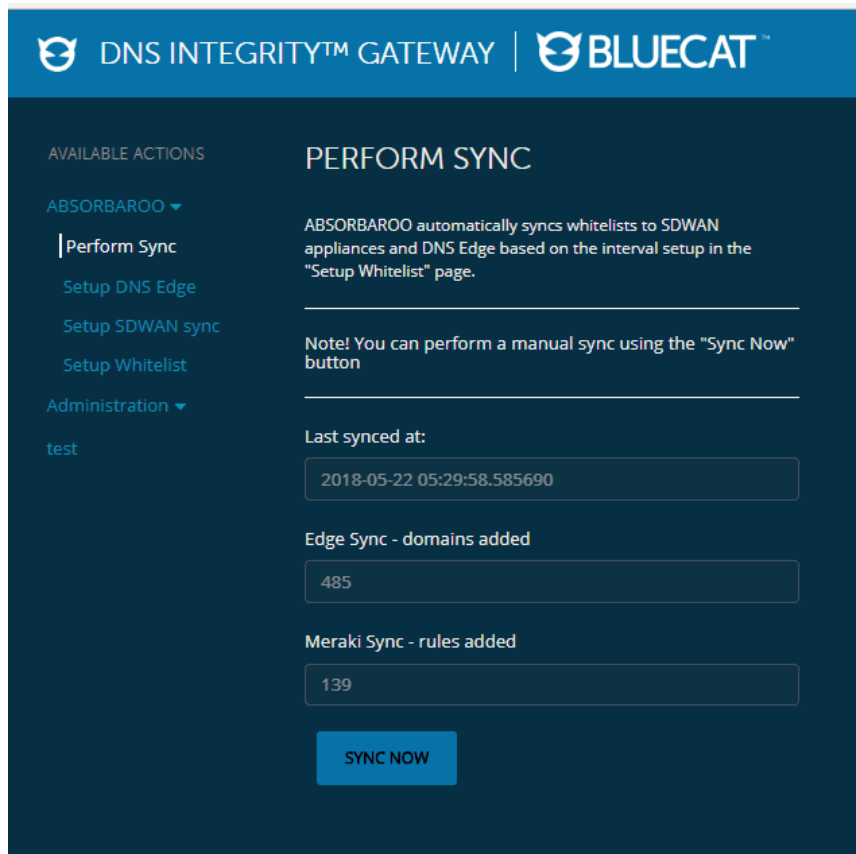
BlueCat offers an automation workflow so networks can maintain direct connections to trusted cloud services even as these underlying domains shift over time.

How it Works

The [certified workflow on GitHub](#) maintains a constant watch on third party sites for domain change notifications, checking at intervals defined by an admin.

When services post a notification, the workflow downloads the new domain list automatically. The domain list is pushed through a whitelist into the SD-WAN system, updating the resolution data for all of the client devices covered by your BlueCat service point.

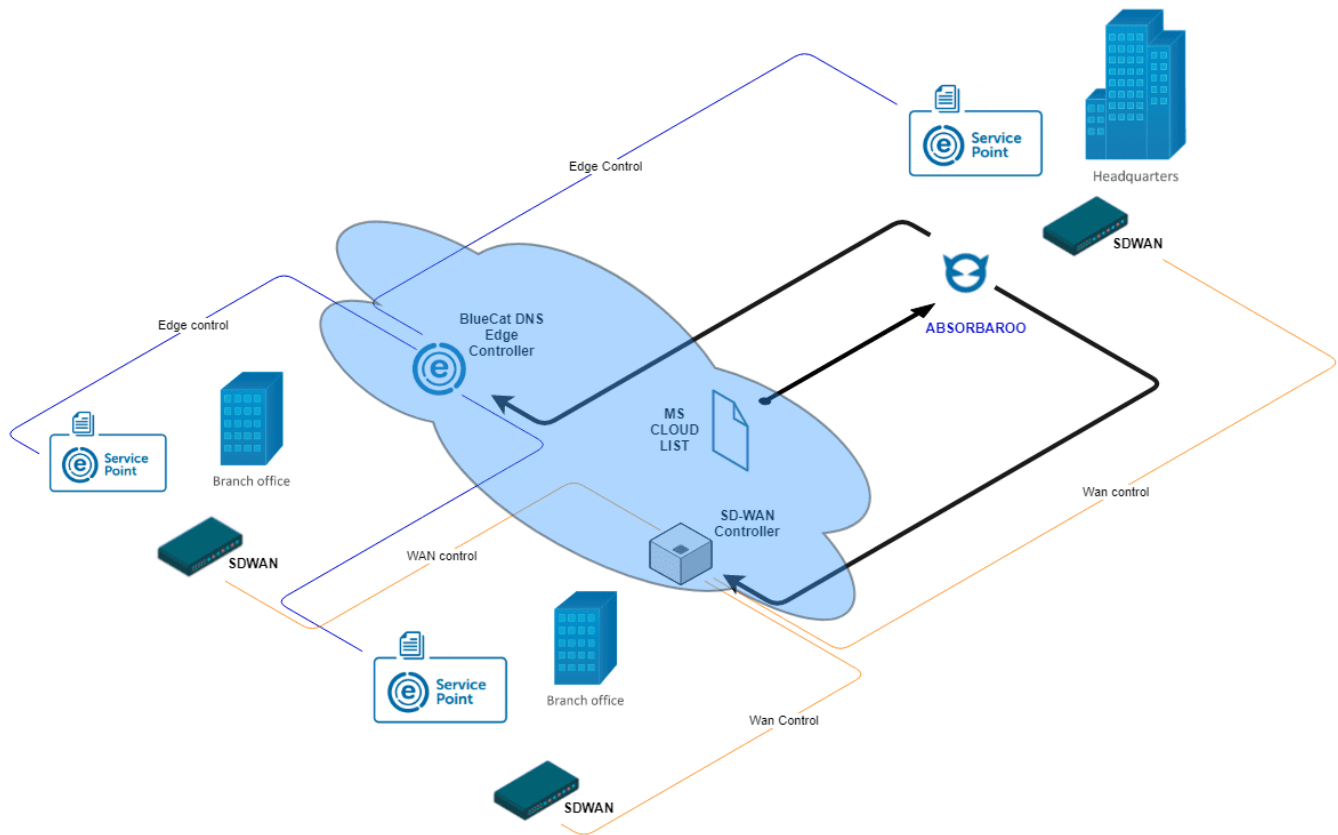
SD-WAN sees the resolution on a mutually held white list and allows the connection to be routed directly out to the trusted service.



The screenshot displays the 'DNS INTEGRITY™ GATEWAY | BLUECAT™' interface. On the left, under 'AVAILABLE ACTIONS', there is a dropdown menu for 'ABSORBAROO' with options: 'Perform Sync' (highlighted), 'Setup DNS Edge', 'Setup SDWAN sync', 'Setup Whitelist', 'Administration', and 'test'. The main panel is titled 'PERFORM SYNC' and contains the following information:

- A description: 'ABSORBAROO automatically syncs whitelists to SDWAN appliances and DNS Edge based on the interval setup in the "Setup Whitelist" page.'
- A note: 'Note! You can perform a manual sync using the "Sync Now" button'
- A section 'Last synced at:' with a timestamp '2018-05-22 05:29:58.585690'.
- A section 'Edge Sync - domains added' with a count of '485'.
- A section 'Meraki Sync - rules added' with a count of '139'.
- A prominent blue button labeled 'SYNC NOW'.

The example workflow posted on our GitHub repository references Office 365 and Cisco Meraki SD-WAN controllers, but the framework can be adjusted for use with any third-party service.



Automating Failover for BDDS

Network failures can happen. To be prepared, a DNS enterprise setup with crossover high availability (xHA) master and standby pairs is a good idea. But failover, even if automatic, can be risky.

Manual BDDS failover requires going into Address Manager and moving all of the roles over one at a time. An admin must bring the old servers down, bring the new ones up, and execute a full deployment.

With an automated BDDS failover, Gateway can monitor your BDDS master server for any performance changes. If it goes down, the system will know to change roles and deploy BDDS to your standby. In a matter of a minute or two, your standby becomes the new master.

A few dynamic updates may take a little longer to propagate, but your net downtime is zero. Your network keeps humming and everything continues as it has been. The consequences of a misfire are virtually nil.

It is important to note that, while this failover may be automated, it not unprompted. Given the gravity of a failover process, some admin intervention is always required to execute it.

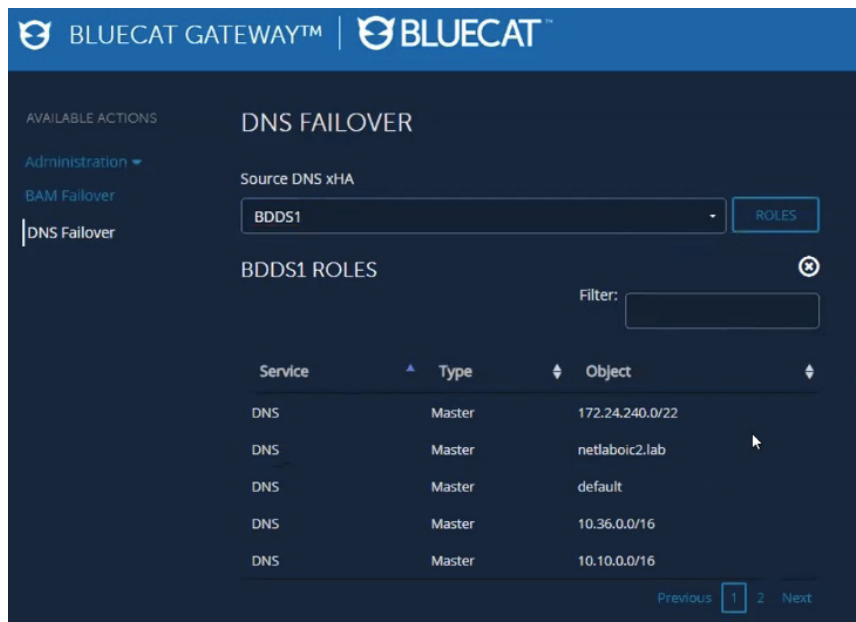
This is the feature you hope you never have to use. But it is your safest and easiest route for fixing things when they have gone horribly awry. It's an easy, repeatable sequence of steps that does all the hard work for you behind the curtain.

How it Works

An automated failover setup for your BDDS servers would include two xHA pairs—four servers total. Two servers are your active masters, humming along. Two are the designated xHA pairs, on standby for failover and some load balancing capability.

Let's say that a catastrophic incident hits your data center, and your master servers go down.

Go into Gateway, which is pulling from Address Manager, and select your source in the xHA pair.



Select the role you want the other server in the pair to perform, and then the destination server.

The screenshot shows a web interface for managing DNS xHA. At the top, there's a dropdown menu set to 'BDDS2' and a 'ROLES' button. Below this is a section titled 'BDDS2 ROLES' with a filter input field. A table lists roles with columns for Service, Type, and Object. The table contains two rows for DHCP services. At the bottom right, there are 'Previous', '1', and 'Next' navigation links.

Service	Type	Object
DHCP	Master	4.0.0.0/8
DHCP	Master	172.24.240.0/22

Click submit, and a large red box will appear asking you to confirm failover.

The screenshot shows a 'Confirm Failover' dialog box with a red header. The message inside says: 'This action will initiate DNS failover. Click 'Submit' to proceed.' There are 'CLOSE' and 'SUBMIT' buttons. In the background, a table of DNS roles is visible.

Service	Type	Object
DNS	Master	default
DNS	Master	10.36.0.0/16
DNS	Master	10.10.0.0/16

That's it. All you have to do is select a source and then a destination, click the button, and then the magic happens. Gateway moves all the roles from one server in the xHA pair to the other, completes all the necessary fact checking, and deploys. Prior to the incident, the destination in the pair doesn't hold any of these roles. Until deployment executes, it doesn't know that it's master over anything.

Automating IP Address Allocation and Deallocation in the Cloud

Syncing IP address management between cloud and on-premises seems like it should be easy, but it often isn't. They are two disparate systems. Neither was designed with seamless integration in mind.

It can be particularly challenging to maintain visibility into IP activity in the cloud. For example, you might be spinning up a virtual machine that is using an internal Microsoft Azure IP address, but that address does not show up in your IP address manager. Whatever tracking you have is probably manual.

By using Gateway for automated allocation and deallocation, you can keep Address Manager as your single source of truth for IP address management, regardless of where your enterprise resides.

Here, automation is a two-part process. First, an event occurs within the cloud itself, such as adding an IP address. Then, it triggers a reaction. Information is sent to Gateway, and the business logic put in place dictates how Address Manager reacts to the event. For example, it could be to ask for more information, to take the information it has been given to create a record, or to deploy certain artifacts. It can be flexible and customizable to your needs.

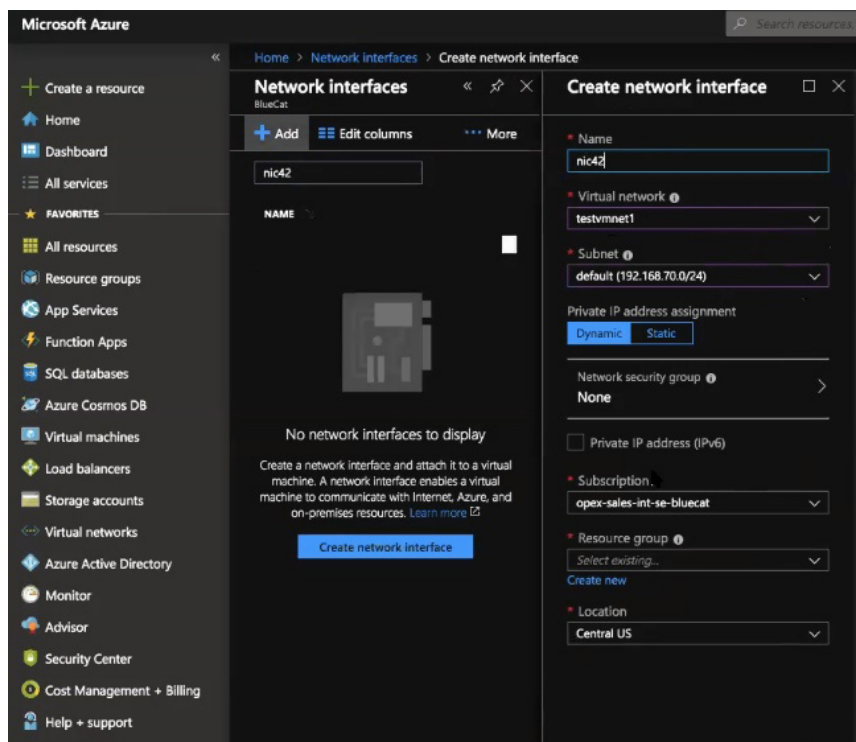
Automation is cloud agnostic as well. While the following example is in Azure, the same idea applies in AWS or Google Cloud, minus some minor tweaks to the implementation.

How it Works

Keeping with the virtual machine example in Azure, we can see automation in action when a new virtual machine adds a network interface to get an IP address. Some front-end work is required to build your alerts and a logic application in Azure as well as to install a Gateway workflow.

You can select the default virtual network and subnet (or change them if needed), and then select your subscription and the resource group in Azure.

Then, you can use your "create network interface" alert in Azure's Monitor feature to check whenever interfaces are being brought up or down. It shows an alert for the new network interface created.



Your work is done. The rest of the process happens in the background without you having to lift a finger. The alert triggers the business logic that communicates with Gateway. In this example, it is to create a record of the IP address in Address Manager.

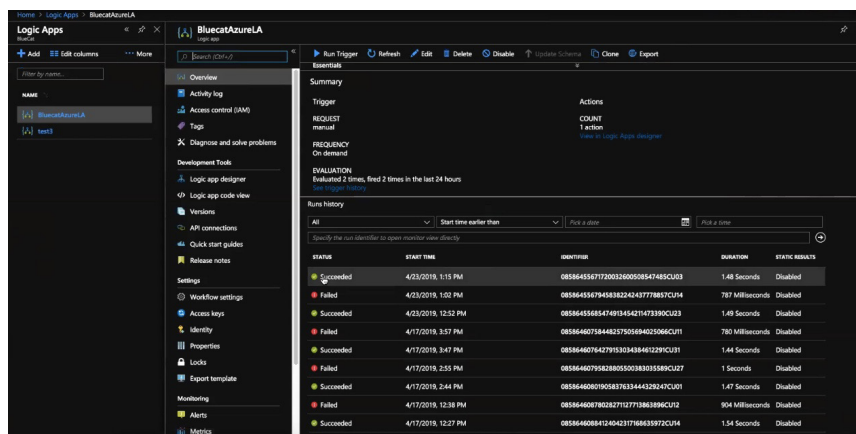
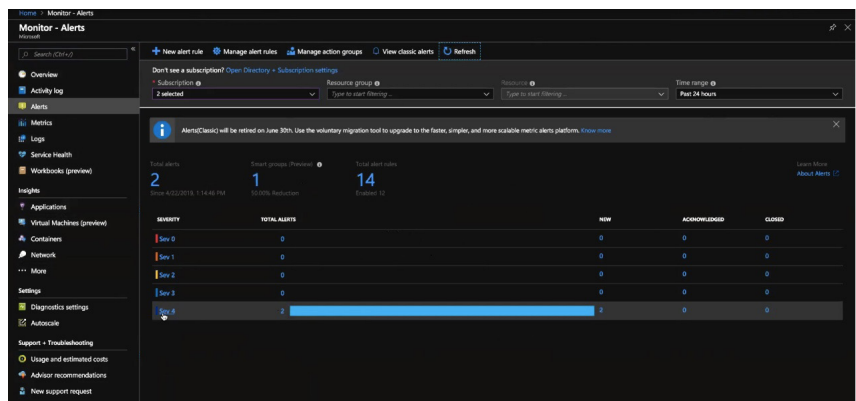
Behind the curtain, the alert calls a logic application in Azure that parses the data. This will call a webhook in Gateway. The Gateway endpoint will accept the alert to create a host a record and assign an IP address in Address Manager.

You can see the “succeeded” message showing that the logic app successfully called the Gateway webhook.

If you go into Address Manager, you should see the record created. It should be assigned the same IP address as it was assigned in Azure and an associated host record created as well.

The same process works for deallocation in the cloud, with a “delete network interface” alert to track whenever network interfaces are deleted in the cloud. It triggers a similar process. The alert will call the logic app, which parses the data, calls the webhook in Gateway, and finds the host record and associated IP address in Address Manager and deletes it.

Either way, no more manually tracking IP addresses assigned in the cloud!



Automating Adding DNS Records with ServiceNow

Handling one of the simplest IP management requests—adding a DNS record—still takes time and effort for an admin to complete. That time and effort can be compounded in a ServiceNow workflow environment, where an IT admin might have to review a work request, validate it, execute the work themselves, and then respond to close it out.

Enter BlueCat's ServiceNow app for adding DNS records, available in the ServiceNow store. It works with Gateway to facilitate an easy connection to Address Manager, automating the process for adding records.

Users build out their forms in ServiceNow, owning the end-user interface and process.

Gateway acts as the middle layer between ServiceNow and BlueCat's API. It knows how to handle the decision trees, errors, and other things that might happen in the process. It can smooth those all out and make it easier for ServiceNow to understand it. As a platform, ServiceNow was not built to process hundreds of lines of code for complex business tasks. Using Gateway as an intermediary shifts all of this hard work to BlueCat.

An important note: This is only the start of our ServiceNow integration package and is not the only way to integrate Gateway with standard DNS ticket requests received through ServiceNow. We will soon expand this offering to handle more complex implementations, such as scheduling Gateway to access unaddressed ServiceNow tickets. Is there a ServiceNow use case you'd like us to address? We're all ears.

How it Works

After installing the app from the ServiceNow store, you'll have to download and permission the itsm_api community workflow from the BlueCat Labs GitHub repository. Then you can access ServiceNow and pull up the form to add a DNS host record.

Search for and select your configuration, your view, and your zone.

That's it!

You may wish to have an approval process attached to this, where an admin puts final eyes on it to ensure everything looks fine after Gateway has already handled the validation. In these cases, an admin can click OK to actually execute the action.

Managing Options & Roles

Harmonizing DNS/DHCP options and server roles across multiple levels of IP address management can be a time-consuming task.

Viewing these settings throughout your database and dealing with conflicts is usually a two step process. First, administrators have to navigate through configurations, blocks, networks, zones, or views to locate individual options. Only then can they edit the value or move its location and level of inheritance. It wasn't possible to automatically find these options and roles, let alone manage them from a single place....until now.

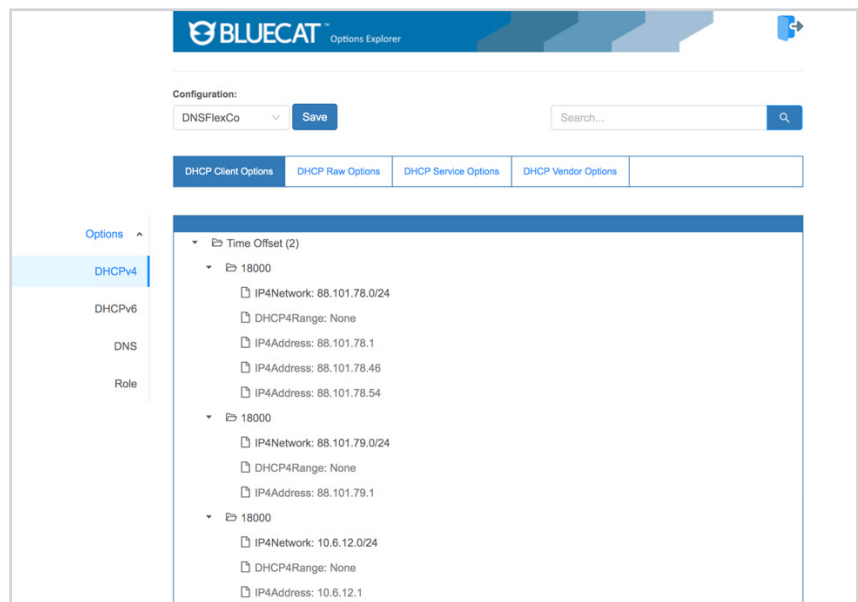
Enter the latest automation workflow from BlueCat. Leveraging the power of BlueCat's automation platform, this new workflow allows users to make bulk changes to DNS/DHCP options and server roles without having to individually navigate through the hierarchy and set them in the BlueCat Address Manager. The workflow automates the search from the options point of view, displaying all of its associated locations and values. This allows users to make bulk changes and consolidate inheritances all in one place.

Administrators save time and effort by managing and modifying numerous options on multiple devices at the same time. The workflow makes it easier to search and find where options are used and see where the inconsistencies exist so that the options can be deployed more efficiently.

The workflow can also help to minimize the number of options assigned by allowing administrators to cut and paste options and their values, thus consolidating inheritance of values at a higher level.

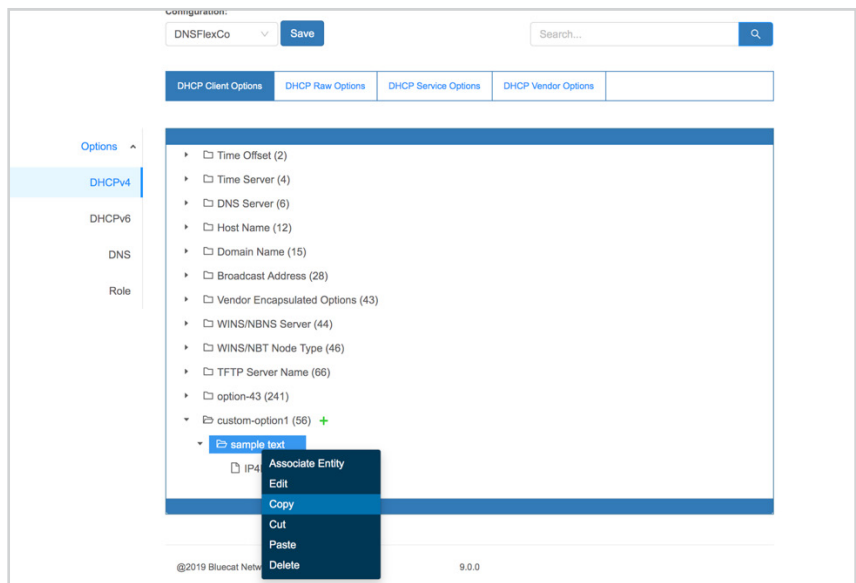
How it Works

The workflow gathers values and locations on options and roles from across a configuration and puts it into a tree structure with the options or server roles at the root. Child to each option is a list of values currently assigned followed by their respective locations.



From this workflow, you can perform the following tasks:

- Search by the option or role name or value, and it will highlight where those names are located in the tree or where those values are assigned.
- See all of the server roles currently assigned within a configuration, add additional roles, delete them, or change their associations within the hierarchy.
- Navigate to a particular option and if it is a custom option, change its name.
- Navigate to a default or custom option and add, modify, delete its value or cut and paste it to a new location.



Simplifying the transition between GitLab and production

The most effective network automation programs are constantly evolving. As business requirements change, as scale enlarges support needs, as new technologies complicate existing procedures, IT managers can't just sit on their laurels. They have to be out in front of organizational change, ensuring that the automated processes they built in the past will stand the test of time.

Labs, repositories, and test environments make this agile development process possible. As the underlying logic and requirements of automation change, these toolkits are what developers and IT managers use to create and adjust what they deploy out in the field.

Smoothing out the production deployment process

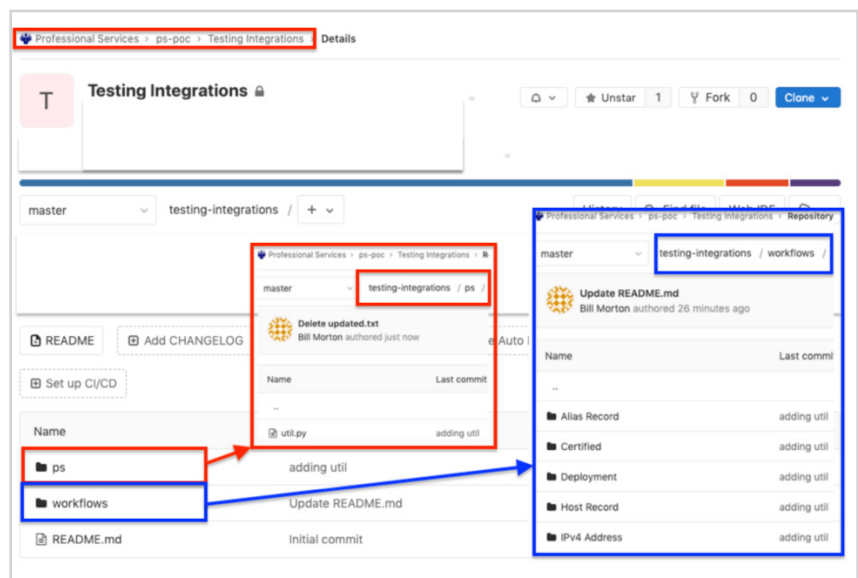
Moving automation workflows from development and testing labs into the production phase usually involves some administrative hassle. You have to set the permissions. You have to reconfigure systems. You have to load everything up. Usually you have to restart the servers. It all takes more time than it should.

There's also a need for some built-in protection against automation workflows which don't perform as designed out in the wild. If you promote a workflow from the lab into a production environment and discover an error, there should be a way to backtrack while preserving the original code in a back-up location.

To promote a more seamless flow between lab and production environments while offering a hedge against errors, BlueCat created a script which connects the GitLab repository used by many internal development teams and BlueCat's own DNS automation platform. The workflow automatically brings resources from GitLab into the BlueCat automation platform without the need to reconfigure and reset a bunch of resources.

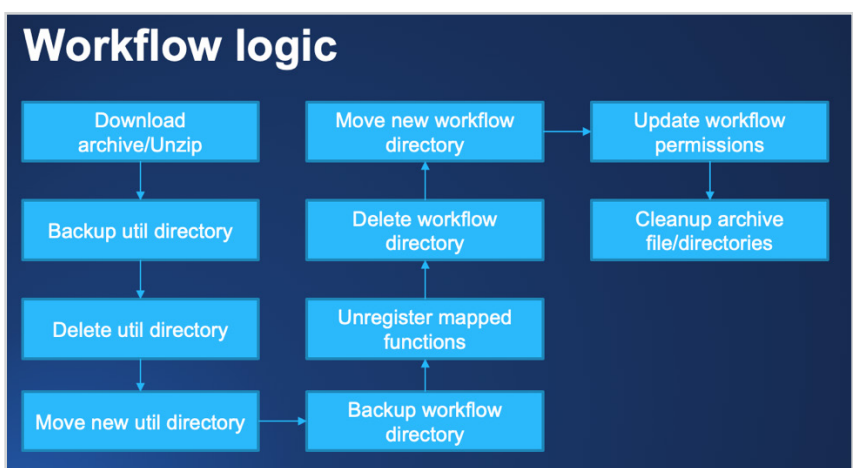
How it Works

When you want to promote a workflow from GitLab to the BlueCat automation platform, you'll be prompted to select the configuration parameters which are already available.



The workflow automatically backs up utility folders, files, and workflows currently deployed on the Gateway server to ensure that nothing is lost in the event of reversion to a previous edition of a workflow.

Download the workflow today on our BlueCat Labs repository.



The word "About" in a bold, sans-serif font, white, set against a yellow rectangular background.

Today's networks are drowning in complexity. If your DNS infrastructure isn't adaptable by design, your network – and your entire business – is fragile. Think outages. Downtime. Data loss.

BlueCat helps your business thrive on complexity with Adaptive DNS. Move beyond static, manually-controlled, off-the-shelf DNS that breaks under network complexity, and move to a DNS that embraces it. Adaptive DNS is powered from the edge of your network, scaling to meet escalating demands by users, applications and services. It's open and automated, giving you the power to enforce business policies, manage security, analyze data, and remediate threats. Thrive on complexity from edge to core with BlueCat Adaptive DNS™. Talk to us.

United States Headquarters

1000 Texan Trail, Suite #105
Grapevine, Texas 76051
+1.817.796.8370
1.833.BLUECAT

Canada Headquarters

4100 Yonge St. 3rd Floor
Toronto, ON, M2P 2B5
+1.416.646.8400
1-866-895-6931

bluecatnetworks.com