11/20/2020

Jonathon Samson-Noble (11000606)

Otago Polytechnic

IN730 Special Topic

Network Automation

**Contents**

[**Introduction** 3](#_Toc56757995)

[**What is Network Automation?** 4](#_Toc56757996)

[**Why use Network Automation?** 4](#_Toc56757997)

[**Network Automation Compliance** 4](#_Toc56757998)

[**Network Automation Advantages** 6](#_Toc56757999)

[**Reduction in operative costs** 6](#_Toc56758000)

[**Consistency** 6](#_Toc56758001)

[**Eliminate manual tasks** 6](#_Toc56758002)

[**Make changes faster** 6](#_Toc56758003)

[**Network Automation Disadvantages** 7](#_Toc56758004)

[**Fear of job loss** 7](#_Toc56758005)

[**Cost** 7](#_Toc56758006)

[**Can break the network** 7](#_Toc56758007)

[**Learning new things** 8](#_Toc56758008)

[**Network Automation Best Practices** 8](#_Toc56758009)

[Take out the links and change them to bullet points i.e. Some of the best practices from Cisco DNA / Ansible are blah 8](#_Toc56758010)

[**Tools for Automation** 9](#_Toc56758011)

[**Ansible** 9](#_Toc56758012)

[**What is Ansible?** 9](#_Toc56758013)

[**Why use Ansible?** 9](#_Toc56758014)

[**How to use Ansible?** 10](#_Toc56758015)

[**Requirements** 10](#_Toc56758016)

[**Inventory** 11](#_Toc56758017)

[**Playbooks** 11](#_Toc56758018)

[**Integrations** 13](#_Toc56758019)

[**Versions** 13](#_Toc56758020)

[**Advantages** 13](#_Toc56758021)

[**Disadvantages** 14](#_Toc56758022)

[**Puppet** 14](#_Toc56758023)

[**What is puppet? expand** 14](#_Toc56758024)

[**Advantages** 14](#_Toc56758025)

[**Disadvantages** 15](#_Toc56758026)

[**Zero Touch Provisioning** 15](#_Toc56758027)

[Cisco DNA 15](#_Toc56758028)

[**Conclusion** 16](#_Toc56758029)

[**References** 17](#_Toc56758030)

# **Introduction**

The network has never been more vulnerable.

With large number of users now working remotely from unsecure home offices and accessing critical systems, applications and other resources in your network the need for automation to secure and expand your network capabilities has never been more obvious.

As modern IT networks are scaling rapidly and require businesses to react swiftly and accordingly to stay relevant this can be achieved through the use of network automation tools such as Ansible, allowing for businesses to deploy updates or new services quickly across their network.

This report will discuss what is network automation, why you would use network automation, compliance with network automation , the advantages and disadvantages of network automation, best practices surrounding network automation and the tools that can be used for automation.

# **What is Network Automation?**

Network automation is the process of automating the configuration, management, testing, deployment, and operations of physical and virtual devices within a network. Everyday network tasks and functions are performed automatically. Using a combination of hardware and software-based solutions, large organizations, service providers, and enterprises can implement network automation to control and manage repetitive processes and improve network service availability.[[1]](https://www.juniper.net/us/en/products-services/what-is/network-automation/)

# **Why use Network Automation?**

As networks grow and become dispersed, hands-on provisioning of each device becomes ever less practical. Automation is the key to efficient setup, avoidance of errors, and economical management of remote network devices.

Without using automation solutions then your IT team will have deal with the repetitive tasks manually, by automating these tasks it allows staff to focus on higher value tasks that can improve productivity.

A big consideration for using automation now is due because covid-19 has led to restrictions in the work environment and is leading to a lot of people to work remotely this has lead to an increase in cyber-attacks, the need for businesses to be able to make large scale changes quickly to their network is becoming more important whether it be for allowing staff to connect to the office network or deploying changes to your network devices.

# **Network Automation Compliance**

With an ever increasing number of cyber attacks occuring everyday and even more with the onset of covid-19, businesses need to make sure that there network is compliant with both government regulations and the businesses security policies, by being able to stay compliant you reduce the risk of disruptions in your network allowing your business to grow.

By using network automation, you can implement solutions and then deploy them across your network automatically instead of having to configure your devices manually this also has the added benefit of reducing human error as having to configure multiple devices repeatedly with similar configuration can lead to a misconfiguration.

Trying to comply with your networks security policies and existing compliance standards can be a challenge for any network and with new data privacy laws coming into effect on December 1st, 2020 *[11]* allowing the ability for compliance notices to be issued, having the ability to be able to react quickly and deploy the changes needed to make sure your network and its devices are in compliance will be key in staying ahead.

After a recent trip to Computers Concept Ltd (CCL) that the Otago Polytechnic Bachelor of Information Technology Project 2 Operations and Security team took, and after speaking with Tim Sewell the Information Security Management Lead at Computer Concepts Ltd he informed us that after the recent DDoS attacks against NZX that now a lot of businesses are now starting to become more concerned about their security this will lead to an increase and a focus on compliance which automation can be used to assist with.

Manually configuring your devices to comply with company policies will eventually lead to human error as configuring the same stuff, these errors could lead to a misconfigured firewall allowing for an attack or a misconfigured router causing an outage in your network.

To ensure that your network automation solutions are staying compliant with your businesses policies and government requirements surrounding privacy and other laws, a solution can be put in place that performs a check against (see Figure 1)

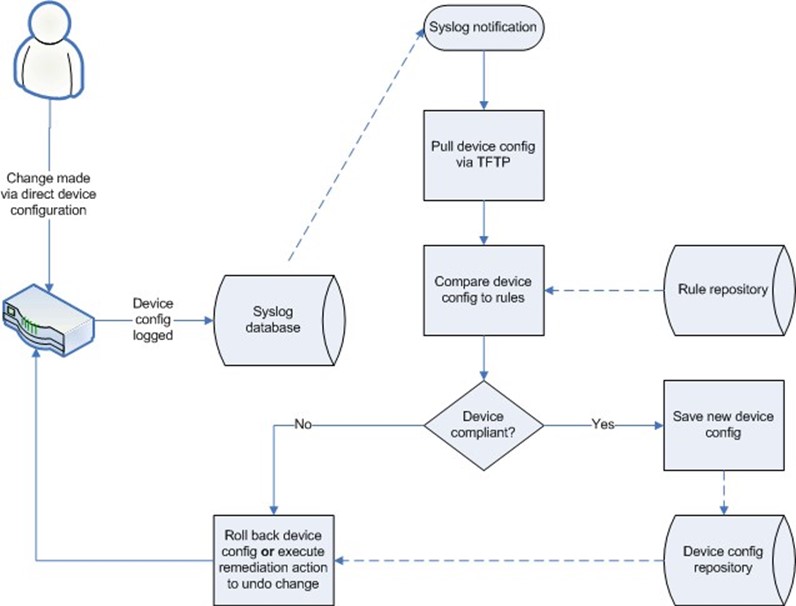


Figure 1Example of an automated remediation[12]

One of the biggest holes in most networks is network devices that the business isn't aware they have. In an environment with large numbers of devices it's easy to overlook one or two that could be hiding somewhere on site or a device that was used for testing purposes but never removed.

Overlooked devices represent the single biggest threat on any network simply because they're ignored, they don't get patched, they don't get the latest configuration changes, and they may not even get audited. They can become compromised without you even being completely aware of where the breach came from. Once compromised, of course, you're no longer compliant or secure.

# **Network Automation Advantages**

## **Reduction in operative costs**

A big benefit of network automation is lower operational costs by reducing tedious manual processes that IT staff have to work on daily

## **Consistency**

Using automation to deploy your configuration will allow for consistency across your devices this will also lead to a reduction in misconfiguration as humans entering the same configuration on multiple devices will eventually lead to a misconfigured command.

This will allow network engineers to work on more important tasks then being tied down by troubleshooting lower-level issues.

## **Eliminate manual tasks**

By automating the repetitive manual tasks that network engineers and system administrators have to deal with they can spend their time working on more complex tasks allowing for greater production and output from the business.

## **Make changes faster**

Using network automation, you are able to make changes quickly to your networks allowing your business to respond rapidly.

# **Network Automation Disadvantages**

## **Fear of job loss**

By automating tasks, you will improve the network more efficiently than a person this will put lower level positions or people who don’t adapt out of a job

## **Cost**

Automation solutions involve considerable initial investment in order to produce but once they are implemented

## **Can break the network**

While automation can help to configure your network and get it working it can also destroy your network, an accidental misconfiguration in your automation solution can lead to disaster if it is pushed out to your network.

But this can be solved by implementing policy and compliance solution to verify the changes (see Figure 2)

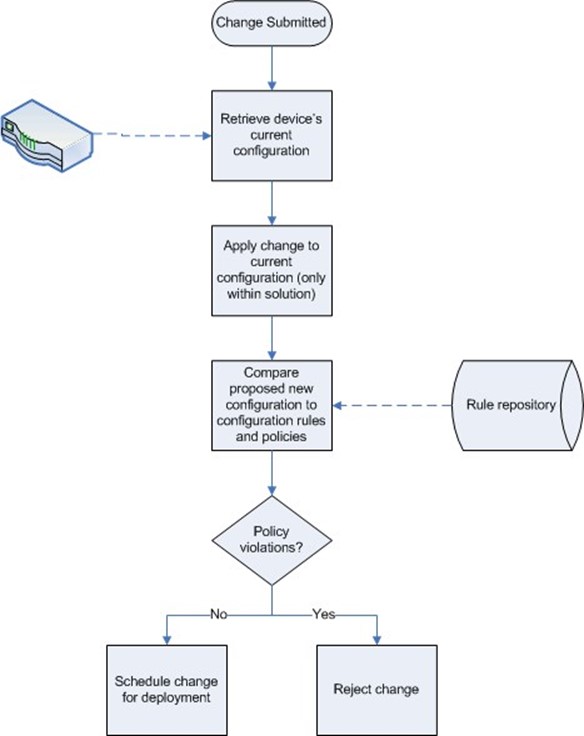


Figure 2 Example of automated change review process[12]

A documented example of this happening to google can be found here <https://status.cloud.google.com/incident/cloud-networking/19009>

Without careful planning, network automation will lead to an automated mess.

## **Learning new things**

Most people don’t like to adapt and change to new things but with the ever-growing presence of IT in the world and the rapidly changing landscape of how networks are accessed and maintained a manual approach will eventually no longer be feasible.

# **Network Automation Best Practices**

A good idea is to apply DevOps best practices to the processes of automation as DevOps best practices are designed to reduce risk and improve efficiency, which works extremely well for network automation overall.

Just like in DevOps certain things should not be automated in networks

* **Sensitive workloads.** Automation is designed to improve upon business operations. When it comes to sensitive workloads, it is more likely to interfere and cause problems rather than resolve them.
* **New and advanced applications.** Newer applications often manage their own automation and load balancing; adding on another layer may actually cause further problems.

While these general practices are good to follow ,most suppliers of network automation tools will have documentation on the best practices when it comes to using there tools and even though the documentation for these are in relation to the tools they provided the concepts can be applied to any form of network automation

### Take out the links and change them to bullet points i.e. Some of the best practices from Cisco DNA / Ansible are blah

Cisco DNA best practices can be found here

<https://www.cisco.com/c/en/us/td/docs/cloud-systems-management/network-automation-and-management/dna-center/hardening_guide/b_dnac_security_best_practices_guide.html>

Ansible best practices can be found here

<https://docs.ansible.com/ansible/latest/user_guide/playbooks_best_practices.html>

# **Tools for Automation**

Add in a bit about tools here before explaining variants

## **Ansible**

### **What is Ansible?**

Ansible is an open source agentless automation tool that can be used for such tasks as configuration management, application deployment and provisioning.

Ansible works by creating an inventory of hosts which it then uses in a playbook to deploy pre-built modules or user created modules against those hosts it achieves this by temporarily connecting via ssh or windows remote management to your targeted hosts, an example of this can be seen below in figure 1.

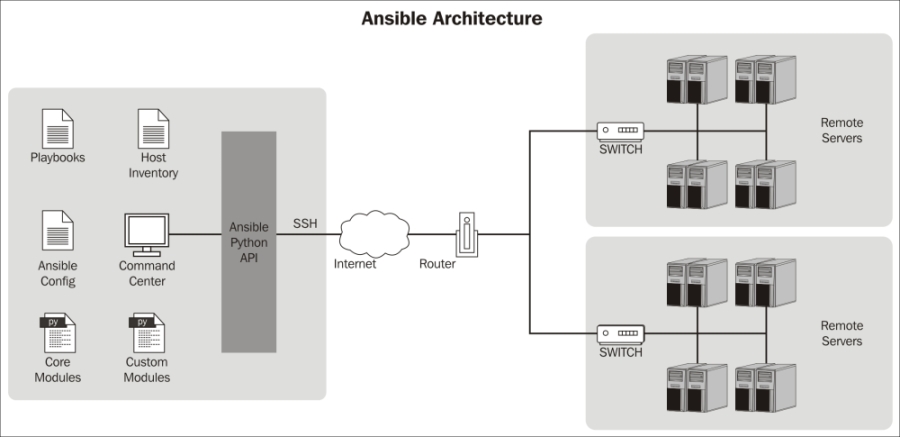


Figure 3 Ansible Diagram [[2]](https://subscription.packtpub.com/book/networking_and_servers/9781783550630/1/ch01lvl1sec09/the-ansible-architecture)

### **Why use Ansible?**

Ansible is a free open-source agentless tool that will naturally reduce the resources consumed and overhead of the network devices used because, Ansible will only need to be installed on the control nodes and no additional software is needed on the managed nodes.

Ansible is simple to learn as the playbooks required to run the configuration tasks are both easy to use and readable, they are based on YAML (YAML Ain’t Markup Language) format, a descriptive language.

As your business grows over time staff will come and go and you may not have the original network engineers and you will have most likely added either new sites or network equipment to your business which will have been configured differently from pre-existing equipment, this can lead to security problems and out of date versions of software or even deprecated software on your devices.

Ansible innately allows for consistent and standardized environments across your network as you can place your nodes into specific groups, for example, you could have a database group and a web group which would all be configured based on your playbook configuration. This in turn will lead to a more agile environment and a reduction in human error because there is less inherent risk due to the administrators of the network having to code less because every line of code is a potential bug.

It also allows less experienced network administrators on your team to run complex playbooks or scripts as they would have already been thoroughly tested beforehand.

By automating everyday network tasks with Ansible this will allow your network engineers more time to focus on more crucial tasks which will turn increase the productivity of your business.

Ansible can be integrated into a wide range of existing vendor technologies that already may be required on your network.

It is also used by such companies as Lockheed Martin[[3]](https://www.ansible.com/network-automation-of-f5-big-ip-devices-with-ansible-tower), Apple[[4]](https://www.ansible.com/blog/enterprise-ansible) and NASA.[[5]](https://www.ansible.com/blog/nasa-automation)

### **How to use Ansible?**

Ansible is very simple to setup as it only requires that the control nodes have the Ansible software installed on it.

### **Requirements**

In order to use Ansible for your network automation solution your control node must have the following requirements met.

* Linux/Unix based system[[10]](https://docs.ansible.com/ansible/latest/installation_guide/intro_installation.html#prerequisites)
* Python 2 (version 2.7) or Python 3 (version 3.5 or higher) installed[[10]](https://docs.ansible.com/ansible/latest/installation_guide/intro_installation.html#prerequisites)

Microsoft windows is currently not supported for the control node.

Documentation for installing Ansible on your given Linux/Unix system can be found at <https://docs.ansible.com/ansible/latest/installation_guide/intro_installation.html>

### **Inventory**

In Ansible the inventory is a description of all the nodes that can be accessed by Ansible, this file can either be in YAML or INI format. The inventory configuration file can have either the hostname or the ip address of each node. Nodes inside the inventory file can also be assigned into a group or multiple groups which are designated by the group name “[*group name*]”

An example of the inventory file which is stored in /etc/ansible/hosts can be seen in figure 2 below.

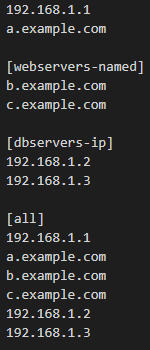
The best idea when creating your inventory file is to have a standardized logical naming convention for your groups this will allow you in the future to add additional devices to that group if they qualify for that group, for example, if you needed to add an extra database server down the line you can add that hostname or ip to your database group in your inventory file.

Figure 2 Example of an inventory file

By assigning the nodes to groups this allows you to utilize Ansible playbooks against a group rather than an individual node this will allow for consistent configuration across all of your machines in a given group and increase productivity as you’re connecting to machines individually to configure them one by one , for example, if you want to make sure all of your nodes in your [dbservers-ip] group were running the same configuration. This also has the added benefit of standardizing your network

### **Playbooks**

Ansible playbooks are one of the core features of Ansible and tell Ansible what to execute. They are like a to-do list for Ansible that contains a list of tasks. Playbooks contain the steps which the user wants to execute on a particular machine.

Ansible playbooks can be used to manage the configurations of and deployments to remote machines. At a more advanced level, they can sequence multi-tier rollouts involving rolling updates, and can delegate actions to other hosts, interacting with monitoring servers and load balancers along the way.[[6]](https://docs.ansible.com/ansible/latest/user_guide/playbooks.html)

Creating playbooks for each individual task rather than having a playbook containing multiple tasks will allow for an improved the readability for users, but if you need to run multiple tasks you can import playbooks into other playbooks allowing you to run multiple tasks.

By creating individual task playbooks it has the added advantage of being able to create a customized playbook rapidly using playbooks you have already created, for example, if you needed to create a new database server that also has some form of web application you can take your database task playbooks and combine them with your web task playbooks importing them into a new custom playbook that you know will work because your reusing previous playbooks have already been tested.

You can also restrict it so that only specified users can run certain playbooks. The way to do this is by assigning roles and permissions, this will help to improve the security of your system as you can limit the more critical and potential system breaking playbooks to authorized users.

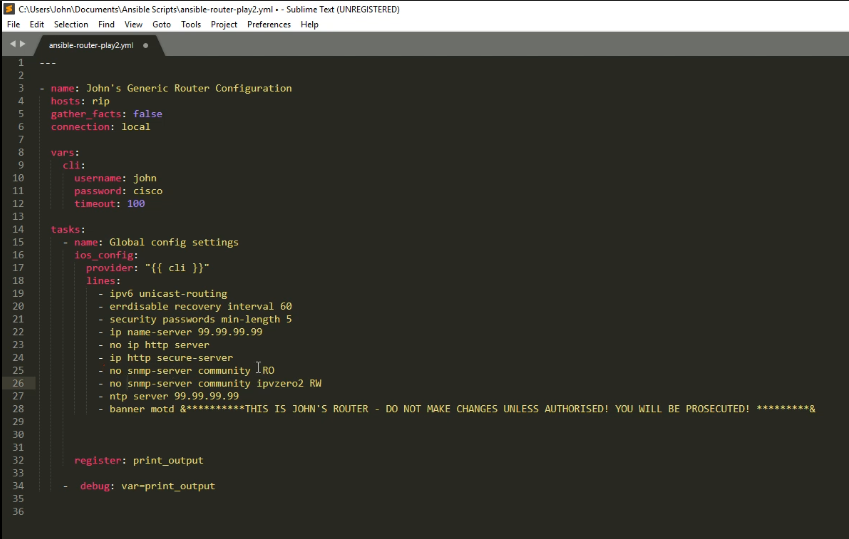


Figure 3 Playbook example of a cisco router configuration [[7]](https://www.youtube.com/watch?v=xw4IinFGVzA)

The Ansible documentation also contains information on best practices when creating your own playbooks which can be found here <https://docs.ansible.com/ansible/latest/user_guide/playbooks_best_practices.html>

### **Integrations**

Ansible can be integrated into existing infrastructure, networks , containers , cloud and DevOps tools for which it includes hundreds of modules to support a wide range of vendors and their technologies such as cisco, AWS, Microsoft Azure, VMware to name a few, a more in depth list can be found here at <https://www.ansible.com/integrations>

### **Versions**

There are a few different versions of the Ansible software ranging from both free and paid versions and CLI to GUI.

**Ansible** – Open-source command line only version of Ansible that is free, it contains all the functionality minus the web-based GUI that Ansible tower offers

**Ansible Tower** – free for up to 10 nodes features web-based GUI but requires that Red Hat Enterprise Linux 7 or 8 be installed on the device running Ansible Tower more information relating to ansible tower can be found here <https://www.ansible.com/products/tower>

**Ansible AWX** – This is the open source version of the Ansible Tower web application which is free, more information relating can be found here <https://github.com/ansible/awx>

### **Advantages**

Some of the main advantages to using Ansible for network automation are the following:

* Ansible is a free open-source automation tool that can be integrated with a wide range of existing vendor tools, they have extensive documentation on modules that can be used with these tools.
* Due to Ansible being an agentless tool this means there is less overhead, and resources being used as only the control nodes require Ansible to be installed.
* Multiple control nodes allow for redundancy so if your main Ansible server was disconnected you would still be able to function on your backup control nodes, by also having multiple control nodes in different parts of your network you could have a script that is executed when you run your playbook on your main device that would contact another Ansible control node and deploy that same playbook to a given segment thus reducing the load on each control node.
* Automation helps to reduce human error by requiring less coding from users.
* Reduction in outages due to Ansible innately creating a standardized network which allows for improved troubleshooting times.
* If you need to update software on your network devices or deploy new services for your clients, this can easily be done by creating new playbooks and deploying it to your network allowing the business to respond rapidly to required changes.
* Scalability, this can easily be achieved, because if your adding more infrastructure to your network you already have the playbooks created from your existing devices and you just need to add the new devices to your inventory file.

### **Disadvantages**

Some of the main disadvantages to network automation would be the following

* The initial cost of learning to use a network automation tool like Ansible as well as the cost of implementing a network automation solution, like any IT solution will require development time, the business will not see the full benefits until deployment.
* Due to the nature of how Ansible works there may be network performance issues when deploying a playbook to your whole network, this can easily be dealt with by having multiple control nodes in different segments of your network thanks to Ansible allowing multiple control nodes in your network.
* By relying on a network automation tool to maintain your network management and deployments among other tasks, you run the risk that if the tool was to go down you may not have solutions in place to maintain your network if some time has passed since its implementation.

## **Puppet**

### **What is puppet? expand**

Puppet is an open source server-client configuration management tool that is written in Ruby.

### **Advantages**

Some of the advantages of puppet are

* Puppet is an open source tool used by many developers and such has a large array of modules that it can use for many devices, it also features comprehensive documentation for these modules an example of one of these cisco\_ios modules can be found here <https://forge.puppet.com/modules/puppetlabs/cisco_ios>
* Puppet has cross-platform support allowing it to operate on Linux distributions, Unix systems and Microsoft Windows
* Puppet is a declarative language allow for the description of system configuration
* Infrastructure is consistent as you execute the same code on every device
* Puppet manifests are readable

### **Disadvantages**

Some of the disadvantages of puppet are

* Puppet works on a server-client method which requires all of your clients (the devices you want to manage) to have puppet installed on them increasing the amount of overhead needed
* Sharp learning curve if Ruby isn’t known

## **Zero Touch Provisioning**

While exploring solutions to deploy windows images from a central server to new devices, I came across Zero Touch Provisioning which allows for new devices to connect to the image server using DHCP and pulling the image though that.

After discovering this I looked into it more and discovered that you can use this same technology in order to configure new network devices remotely, this can be combined with the Ansible technology by first applying a base configuration with SSH access through Zero Touch Provisioning thus allowing Ansible Playbooks to be deployed against those devices

## Cisco DNA

# **Conclusion**

The network has never been more vulnerable.

Covid-19 has rapidly changed the way networks are being accessed now, with large number of users now working remotely from unsecure home offices and accessing critical systems, applications and other resources in your network the need for automation to secure and expand your network capabilities has never been more obvious.

As modern IT environments continue to become more complex by implementing a range of physical and cloud-based technologies and scaling larger in size, network administrators and businesses will need to start preparing to implement network automation solutions that will allow them to maintain a consistent and standardized network environment whilst also having the ability to scale their network.

# **References**

1. What is Network Automation. (n.d.). Retrieved from <https://www.juniper.net/us/en/products-services/what-is/network-automation/>
2. The Ansible architecture (n.d.). Retrieved from <https://subscription.packtpub.com/book/networking_and_servers/9781783550630/1/ch01lvl1sec09/the-ansible-architecture>
3. Ansible, R. H. (n.d.). AnsibleFest Atlanta - Network Automation of F5 BIG -IP Devices with Ansible Tower. Retrieved from <https://www.ansible.com/network-automation-of-f5-big-ip-devices-with-ansible-tower>
4. Ziouani, S. (n.d.). Enterprise companies welcoming Ansible IT Automation. Retrieved from <https://www.ansible.com/blog/enterprise-ansible>
5. London, D. (n.d.). New Case Study: NASA and Ansible Tower. Retrieved from <https://www.ansible.com/blog/nasa-automation>
6. Working With Playbooks. (2020, March 5). Retrieved from <https://docs.ansible.com/ansible/latest/user_guide/playbooks.html>
7. [IPvZero]. (2018, November 29). Ansible Playbooks for Cisco Network Automation! [Video File]. Retrieved from <https://www.youtube.com/watch?v=xw4IinFGVzA>
8. Ansible, Inc. (n.d.). Ansible Project. Retrieved from <https://docs.ansible.com/>
9. Ansible, R. H. (n.d.). Integrations Overview. Retrieved from <https://www.ansible.com/integrations>
10. Installing Ansible. (2020, March 5). Retrieved from https://docs.ansible.com/ansible/latest/installation\_guide/intro\_installation.html#prerequisites
11. <https://privacy.org.nz/privacy-act-2020/campaign?pk_campaign=Privacy%20Act%202020&pk_source=Google%20Ads&pk_medium=SEM&pk_content=Privacy%20Act%202020&pk_cid=11420400459&gclid=EAIaIQobChMIgO2_qYuI7QIVijArCh0eswurEAAYASAAEgKf8_D_BwE>
12. Automating Network Compliance and Security (n.d.). Retrieved from <https://statemigration.com/automating-network-compliance-and-security/>