Ansible Setup on DebianOS + Device Configuration on CML

# This will be a guide for setting up Ansible on DebianOs. It should work in theory on any UNIX, but here is a [link](https://docs.ansible.com/ansible/latest/getting_started/introduction.html) to the official documentation for your reference. This initial doc will be to confirm reachability to a device via Ansible. Pushing configs through playbooks will come later.

First, we need to get Ansible installed on our device, crazy right? Install it using “*pip install ansible*”. If you’re on a fresh Debian/Linux VM, you might receive this error:



Install pip using the command below. Use the following command to verify installation.

root@DEBIAN-DT:/home/kadeem/debianVM/Ansible# sudo apt-get install python3-pip

root@DEBIAN-DT:/home/kadeem/debianVM/Ansible# python3 -m pip

Install Ansible.

root@DEBIAN-DT:~# apt install ansible

Verify Ansible installation.



Cool, we have our device running Ansible. Now we proceed to the part that we actually care about, using it to automate configuration in our CML lab. This should work on other virtual environments like EVE-NG and GNS3 as long as the hosts are reachable via SSH. Below is the lab I’m going to be working on.

My Ansible control node is hosted on a VM outside of CML.

A diagram of a network

Description automatically generated

Let’s start with “MGMT-SW”. Configure it so that it’s pingable from the Control node. This is what mine looks like:

# Enable SSH:

Now we can proceed to configuring SSH.



This is the error we get when attempting to SSH. From what I’ve read, its because the IOS images on CML run a dated key exchange method that’s been deprecated because of security concerns. No biggie, it’s a lab, so we ignore that.



Go to your control node and navigate to the /~/.ssh/config file. Create one if it’s not already there. Add this.

Mine looks like:



The next test produces this error:



Same thing as before, CML IOS runs a dated host key algo so we have to specify to use that for the host. Under the entry you created before add



End result:



Cool, we have access.



If you noticed in the config I used to enable SSH, I didn’t set a password for the user. I’m going to allow authentication via public keys only, so I’ll go into that next.

# Authentication via public keys:

To allow access via public keys, the server(The switch), has to have the key of the client(DebianHost) that is requesting access. So, let’s generate the key on the client that will be shared. This is lengthy(pause), so I’ll get straight to the point.

Generate key on the client:



Structure it so that it’s readable by the switch:



Configure the switch to only accept public keys for authentication. Add the key to the switch:



Confirm SSH Key fingerprint on the client and switch are identical.

On client:



On server:



PS: Not sure if showing those keys is a security risk but its all good. I don’t have anything, hack me bro.

Now if we test SSH, I should get access automatically without the need for a password.(Lab purposes)



Kidding, go back to the SSH config file at /~/.ssh/config and add this line underneath the host:



Shoutout to David, show him love [here](https://www.youtube.com/watch?v=3lXc7xO8T9k&lc=UgySoQ1nRlqFGypEfAR4AaABAg.A4Mil8wkQW0A4NfnaVKsZ0)

End result:



# Finally, Ansible.(I know, sorry.)

Now we actually do Ansible stuff. First, create your host file. I’m using YAML format since that’s on the CCNP ENCOR and it’s best practice for when you start managing larger environments. The path that Ansible points to by default is /etc/ansible/hosts so let’s create it there.

This is how my inventory file looks.



To explain:

* LabEnvir: is the group where our hosts reside.
* hosts: is pretty obvious, it’s where the hosts are.
  + Within this category, we can specify the names we will use to call the hosts in our Ansible commands. I used MGMT-SW
* ansible\_host: is the IP address assigned to the host

For more info, reference the [documentation](https://docs.ansible.com/ansible/latest/getting_started/get_started_inventory.html).

To confirm Ansible is working, we’ll run an Ad-hoc command and save the configuration changes we made. Result, following.



Playbooks coming next. Thanks for reading!