Site commands: “curl” & “wget” ←*wget copies index file into the local system

---

Network setup

Kali:

- /etc/network/interfaces
    - auto eth0
    - iface eth0 inet [dhcp/static]
        - address [address]
        - netmask [netmask]
        - gateway [gateway]
- “systemctl restart networking”

CentOS: (uses Vim)

- /etc/sysconfig/network-scripts/ifcfg-[interface type]
    - BOOTPROTO=static ← Change from DHCP/Dynamic
    - ONBOOT=yes
    - IPADDR=[ip address]
    - NETMASK=[subnet]
    - ZONE=[zone] ← If assigning firewall zones
- “systemctl restart network”

Ubuntu:

- /etc/netplan/01-network-manager-all.yaml
    - *(aligned)* ethernets:
        - *(tabbed)* [interface type]: (e.g. “ens18:”)
            - *(tabbed)* addresses:
                - *(tabbed)* - [ip address/subnet]
                  **Do not forget the dash before ip*
            - *(tabbed)* gateway4: [gateway]
              **If adding gateway during this step*
- “netplan apply”

*Make sure to restart a web service if one is up via this connection

---

Apache2

Config files:

- Web doc root: /etc/apache2/sites-available/000-default.conf
- Normally web config in /var/www/

*Make sure to start *and* enable the service

CentOS Firewall setup

firewall-cmd is the essential command for firewall setup

- All following commands follow a *--this-commands=input* syntax
- Commands:
  - --list-all-zones
  - --list-all --zone=[zone]
- You can modify an interface's zone by either going into the interface's config file and adding a ZONE=[zone] or:
  - --change-interface=[interface] --zone=[zone] **--permanent**
- **To forward traffic** (meaning to allow traffic to be sent from the router, which is receiving data, to the receiving local machine on the router's network):
  - --zone=[zone] --add-foward-port=port=[port]:proto=[tcp/udp]:toport=[port]:toaddr[receiving ip] --permanent
- To add a service:
  - --zone=[zone] --add-service=[service] --permanent
- To remove a service:
  - --zone=[zone] --remove-service=[service] --permanent
- Firewall-cmd --reload

*Note: After port-forwarding a connection for an *ssh* service that had previously established a connection with the router for the service, there will be a fingerprint/key mismatch, and therefore the fingerprint in known_hosts will need to be removed and re-established.

SSH Setup

Client-side file for the list of hosts with the known public key:

- /.ssh/known_hosts

Host-side file for the public key that allows a user with the matching private key to authenticate passwordlessly:

- /.ssh/authorized_keys

In order to securely copy a key from a server:

- scp user@targetip:[server/target/file/location] [send/to/local/here]

In order to copy a public ssh key:

- ssh-copy-id -i [ssh_key] user@targetip

SSH

File locations:

- **Server-side key storage location:** /etc/ssh/ssh_host_[key-type]_key[.pub for public]
- **Client-side for established connections:** /home/[user]/.ssh/known_hosts
- **Server-side for allowed key pairs from clients:** /home/[user]/.ssh/authorized_keys

Commands:

- ssh-keygen [**-t [type]** to specify type] [key type] [**-f [location]** to specify location]
- scp [user]@[host-ip]:[key-from-location] [desired-key-to-location]
    - Permissions remote-side must be correct for this to work
    - This is for copying a key from the host and onto the client
- ssh -i [private-key-file-path] [user]@[host-ip]
    - This means to ssh without a password, meaning to use the private key to authenticate instead
- ssh-copy-id [**-i [install]**] [key-file-location] [user@host-ip destination]
    - Take a key generated on the client and install it onto the server

Known_hosts is important for fingerprints and re-establishing a connection that was previously made.

Passwordless authentication is when the client has the private key rather than the public key. It is important that the permissions are correct for this to work properly.

> The permissions for ssh and key files should be 700 and be owned by the user, not root.

DNS

On Ubuntu, this is in a configuration file called /etc/*bind*.