**21 - R710 Proxmox run host as DNS server and Terraform VMs pointing at it**

These notes cover adding a DNS server to the run3 host (from which the VM’s are deployed with Terraform) and adjusting Terraform main.tf to make the VM’s use the new DNS server.

This document builds upon the previous documents.

You will need to have created the VM template with VMID 9000 on both hypervisors.

# Terraform module adjustment:

1. Log into rhys@run3
2. Edit the **modules-prox3/main.tf** file to add this line:

nameserver = "192.168.124.162"

just before the lines:  
  
 # if you want two NICs, just copy this whole network section and duplicate it

network {  
  
and save.  
  
(where 162 is the IP address of ‘run3’ host)

# Install DNS server on ‘run3’ host:

1. Log into: **rhys@run3**
2. Do:  
   **sudo apt install bind9**
3. As root, in **/etc/bind**  
   edit:  
   **nano named.conf.options**
4. Change the contents of this file to be:  
     
   options {

directory "/var/cache/bind";

// If there is a firewall between you and nameservers you want

// to talk to, you may need to fix the firewall to allow multiple

// ports to talk. See http://www.kb.cert.org/vuls/id/800113

// If your ISP provided one or more IP addresses for stable

// nameservers, you probably want to use them as forwarders.

// Uncomment the following block, and insert the addresses replacing

// the all-0's placeholder.

forwarders {

194.168.4.100;

194.168.8.100;

};

//========================================================================

// If BIND logs error messages about the root key being expired,

// you will need to update your keys. See https://www.isc.org/bind-keys

//========================================================================

#dnssec-validation auto;

//dnssec-enable yes;

dnssec-validation yes;

listen-on-v6 { any; };

};  
  
and save.

1. Do:  
   **sudo systemctl restart bind9**
2. Doing:  
   **sudo systemctl status bind9**  
     
   should respond with something like:  
   *● named.service - BIND Domain Name Server*

*Loaded: loaded (/lib/systemd/system/named.service; enabled; vendor preset: enabled)*

*Active: active (running) since Thu 2022-02-17 17:30:22 UTC; 1s ago*

*Docs: man:named(8)*

*Main PID: 4823 (named)*

*Tasks: 26 (limit: 9436)*

*Memory: 50.0M*

*CGroup: /system.slice/named.service*

*└─4823 /usr/sbin/named -f -u bind*

*Feb 17 17:30:22 run3 named[4823]: command channel listening on 127.0.0.1#953*

*Feb 17 17:30:22 run3 named[4823]: configuring command channel from '/etc/bind/rndc.key'*

*Feb 17 17:30:22 run3 named[4823]: command channel listening on ::1#953*

*Feb 17 17:30:22 run3 named[4823]: managed-keys-zone: loaded serial 4*

*Feb 17 17:30:22 run3 named[4823]: zone 0.in-addr.arpa/IN: loaded serial 1*

*Feb 17 17:30:22 run3 named[4823]: zone localhost/IN: loaded serial 2*

*Feb 17 17:30:22 run3 named[4823]: zone 255.in-addr.arpa/IN: loaded serial 1*

*Feb 17 17:30:22 run3 named[4823]: zone 127.in-addr.arpa/IN: loaded serial 1*

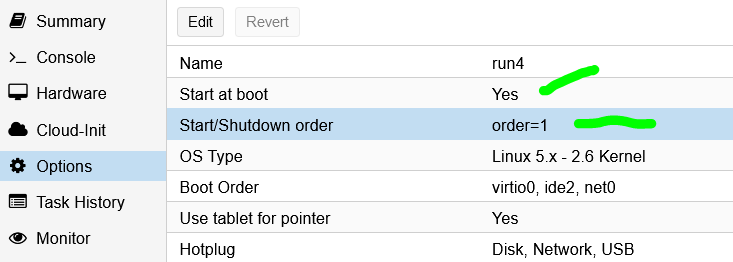
*Feb 17 17:30:22 run3 named[4823]: all zones loaded*

*Feb 17 17:30:22 run3 named[4823]: running*

1. If the above has some error, then run:  
   **sudo named-checkconf**

And fix the error … for example Ubuntu 22.04 with the latest bind9 install as of **5th Feb 2023**, the following needs to be commented out of the **named.conf.options** file:  
 dnssec-enable yes;

Then restart with:  
**systemctl restart bind9**  
and check again with:  
**systemctl status bind9**

1. To test, do the following a few times and the first time its response may be something in the order of 30ms, and the second time near 0ms:  
   **dig www.packtpub.com**
2. For run3, ensure its options for Start/Shutdown order has ‘order=1’ and Start at boot is ‘Yes’, as:  
   

# Build single VM with previously adjusted Terraform file to check its using new DNS server on ‘run3’ host:

1. Log into **rhys@run3**
2. Then in the **terraform** directory, run:

**./clear-sshs**

**terraform init**

**terraform plan  
  
terraform apply -target=module.cw1.proxmox\_vm\_qemu.test\_server**

1. You can then log into the new VM:

**ssh rhys@cw1**

1. Now check the network info for **cw1** with:  
   **cat /etc/netplan/50-cloud-init.yaml**  
     
   which should return something like:  
   *# This file is generated from information provided by the datasource. Changes*

*# to it will not persist across an instance reboot. To disable cloud-init's*

*# network configuration capabilities, write a file*

*# /etc/cloud/cloud.cfg.d/99-disable-network-config.cfg with the following:*

*# network: {config: disabled}*

*network:*

*version: 2*

*ethernets:*

*eth0:*

*addresses:*

*- 192.168.124.224/24*

*gateway4: 192.168.124.1*

*match:*

*macaddress: ee:d4:2a:db:40:ce*

*nameservers:*

*addresses:*

*- 192.168.124.162*

*search:*

*- red.local*

*set-name: eth0*  
  
where 162 is ‘run3’ hosts IP address.

1. You can also check its performance with by doing the following commands a few times:  
   **dig www.packtpub.com**
2. Another confirmation of what DNS server **cw1** is using is to enter command:  
   **systemd-resolve --status | grep "DNS Servers"**  
     
   to get back:  
    *DNS Servers: 192.168.124.162*
3. Finally log out of **cw1** and destroy it with::  
   **terraform destroy -target=module.cw1.proxmox\_vm\_qemu.test\_server**