INSTITUTO POLITÉCNICO NACIONAL

ESCUELA SUPERIOR DE CÓMPUTO

Aplicaciones para comunicaciones en Red

***Practica DNS***

Servidor DNS

Alumno:

Cortez Enriquez Jovanny Wilver

3CV6

Contenido

[Introducción 3](#_Toc42469737)

[Planteamiento del Problema 4](#_Toc42469738)

[Requisitos 4](#_Toc42469739)

[Desarrollo 5](#_Toc42469740)

[Creación de la arquitectura en GNS3 5](#_Toc42469741)

[Instalación del servidor DNS 6](#_Toc42469742)

[Configuración del servidor DNS primario. 7](#_Toc42469743)

[Configuración del segundo servidor DNS 9](#_Toc42469744)

[Pruebas 11](#_Toc42469745)

[Conclusión 12](#_Toc42469746)

# Introducción

Resolución de nombres

Una de las aplicaciones de los espacios de nombres es el almacenamiento y recuperación de recursos mediante nombres.

En general, dado el nombre de ruta deberá ser posible encontrar el recurso asociado al nombre. Al proceso de búsqueda de un nombre en un espacio de nombres se le llama resolución de nombre.

Supongamos que tenemos un nombre de ruta de la forma:

N: etiqueta1, etiqueta2, etiqueta2, … ,etiquetan

La resolución de nombre inicia en el nodo N, entonces se busca la etiqueta1 en la tabla de directorio del nodo N, si existe, se obtiene el identificador del nodo siguiente. Ahora se busca la etiqueta2 en la tabla de directorio del nodo actual, si existe, se obtiene el identificador del nodo siguiente.

El proceso continúa hasta encontrar el nodo correspondiente a la etiquetan

Un ejemplo de resolución de nombres es el servidor DNS.

DNS (Domain Name System) es un protocolo que cumple la función de resolver nombres de dominio.

Se postulo por primera vez en 1983 en el RFC 881 por Jon Postel y seguido en el RFC 882 y RFC 883 por Paul Mokapetris.

En 1984 el mes de Octubre se emite el RFC 920 definiendo lo que hoy en día ha evolucionado hacia el DNS moderno reemplazando los RFC 882 y 883 por RFC 1034 y 1035.

Utiliza los puertos 53/UDP y 53/TCP.

Un DNS asocia un nombre a una dirección IP para su fácil navegación para el ser humano y viceversa, asocia una IP a un nombre.

# Planteamiento del Problema

Se debe crear dos servidores ubicados en diferentes subredes para su configuración.

* Debe haber un servidor primario encargado de gestionar los DNS
* Debe haber un servidor secundario o esclavo.

# Requisitos

* Programa GNS3
* Arquitectura planteada por el profesor
* Sistema Operativo Linux Ubuntu Server 20
* Paquete BIND9
* VirtualBox

# Desarrollo

## Creación de la arquitectura en GNS3

Se debe crear la siguiente arquitectura en GNS3 para realizar la práctica del servidor DNS

Captura de pantalla de un videojuego

Descripción generada automáticamente

Imagen 1: Arquitectura de red

Se realizan las configuraciones siguientes:

* Asignar IP de manera manual en los VPC’s, PC’s de sistemas operativos y router como esta estipulada en la imagen anterior.
* Configurar la tabla de ruteo de manera dinámica con el protocolo RIP.

## Instalación del servidor DNS

1.-Correr la PC agregada a la arquitectura de nombre UbuntuServerRedes2 en VirtualBox antes de iniciar la configuración para tener los paquetes necesarios

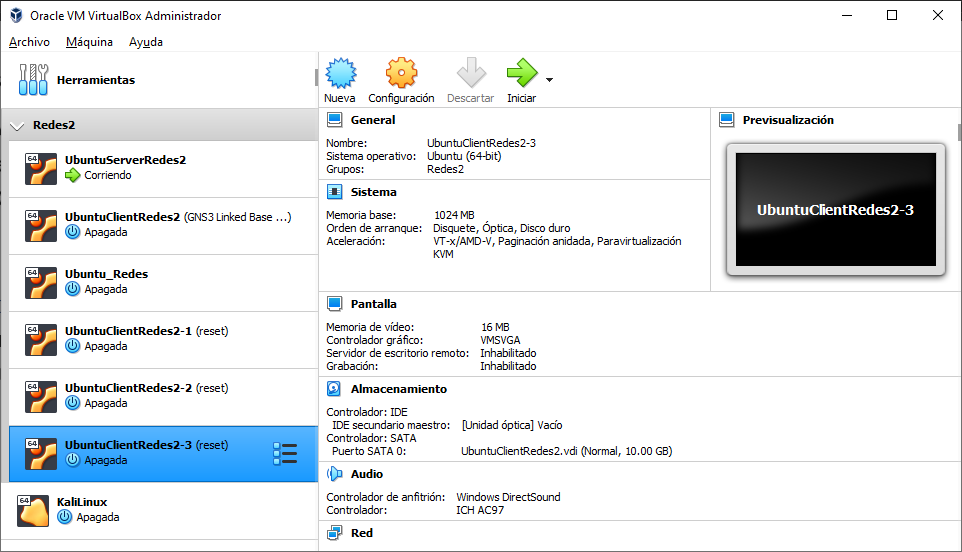


Imagen 2: VirtualBox corriendo la máquina UbuntuServerRedes2

2.- Actualizar los repositorios.

$sudo apt update

3.- Instalar los paquetes necesarios o los sugeridos por el sistema mismo.

$sudo apt upgrade -y

3.- Instalar los paquetes BIND9 con el siguiente comando:

$sudo apt install bind9 bind9utils

4.- Verificar que se haya instalado el paquete tecleando el comando de estado service bind9 status

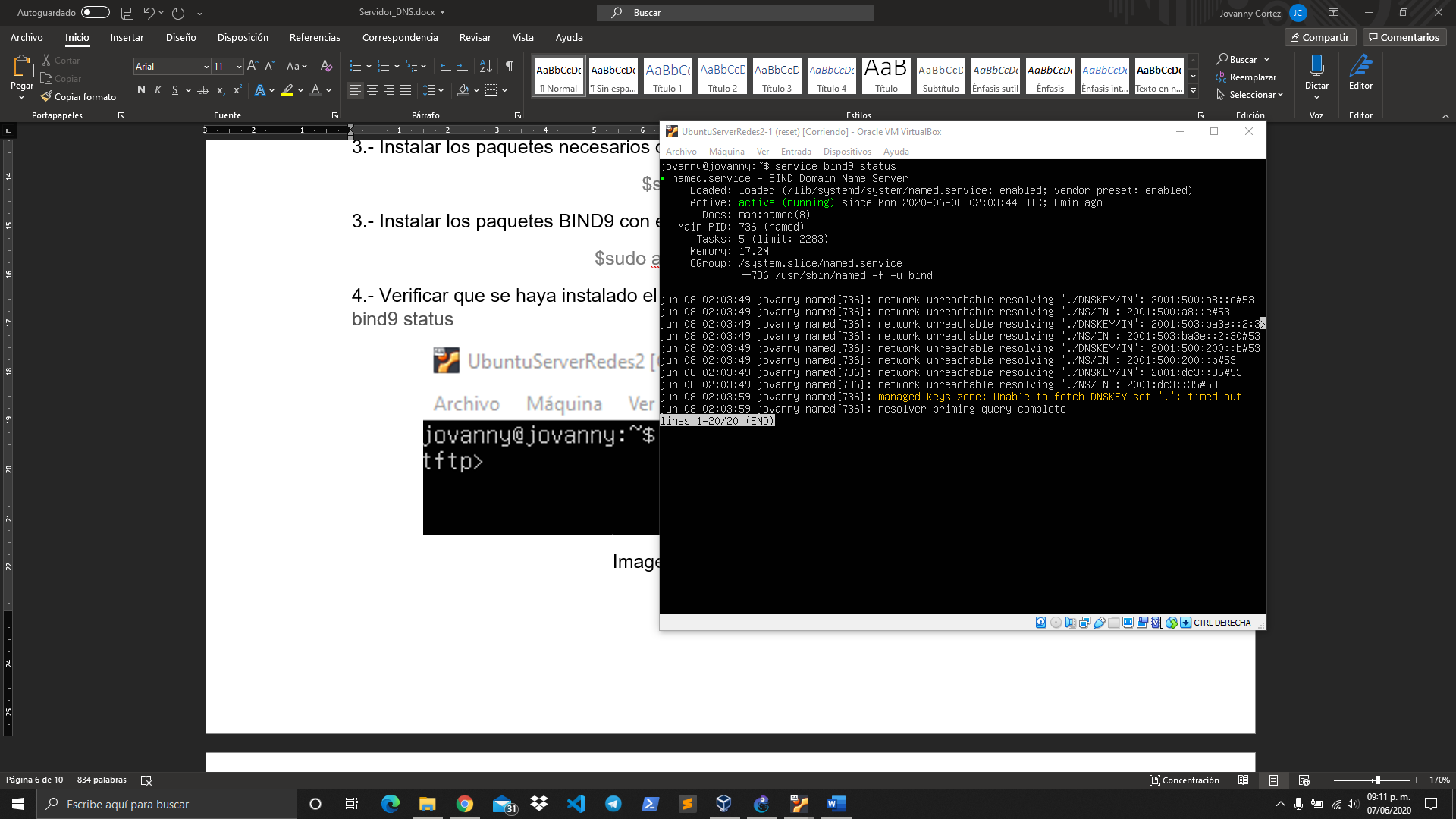


Imagen 3: Estado del servicio de BIND9

## Configuración del servidor DNS primario.

1. Una vez instalado el paquete, se debe correr la máquina dentro de GNS3\*
2. Se realiza la configuración del archivo “bind9” localizado en “/etc/default/bind9” con lo siguiente:

OPTIONS=”-4 -u bind”

1. Configurar las opciones en el archivo named.conf.options ubicado en “/etc/bind/named.conf.options”, se agrega dentro del bloque options las líneas siguientes:

recursion yes; #permite la recursion

allow-recursion { any; }; # cualquiera puede hacer las consultas

listen-on { 192.168.1.4; }; # de manera privada del servidor 1

allow-transfer { none; }; # desactiva la zona de transferencia por defecto.

1. Configurar el archivo local named.conf.local ubicado en “/etc/bind/named.conf.local”.
   1. Zona directa

zone "Jovanny.local" {

type master;

file "/etc/bind/db.jovanny.local"; # zone file path

allow-transfer { 192.168.3.4; }; # ns2 private IP address – secondary

notify yes;

};

* 1. Zona inversa

zone "168.192.in-addr.arpa" {

type master;

file "/etc/bind/db.192.168"; # 192.168.0.0/16 subnet

allow-transfer { 192.168.3.4; }; # ns2 private IP address – secondary

notify yes;

};

1. Creamos el db.jovanny.local haciendo copia del archivo db.local, debe de quedar de la siguiente manera el archivo:

$TTL 604800

@ IN SOA server1.jovanny.local. admin.jovanny.local. (

3 ; Serial

604800 ; Refresh

86400 ;Retry

2419200 ; Expire

604800 ) ; Negative Cache TTL

;

; name servers - NS records

@ IN NS server1.jovanny.local.

@ IN NS server2.jovanny.local.

; name servers - A records

server1.jovanny.local. IN A 192.168.1.4

server2.jovanny.local. IN A 192.168.3.4

; 192.168.0.0/16 - A records

pc1.jovanny.local. IN A 192.168.1.2

pc2.jovanny.local. IN A 192.168.1.3

cliente1.jovanny.local. IN A 192.168.2.4

cliente2.jovanny.local . IN A 192.168.3.5

1. Creamos el archivo db.192.168 haciendo copia del archivo db.127, debe quedar de la siguiente manera el archivo:

$TTL 604800

@ IN SOA server1.jovanny.local. admin.jovanny.local. (

3 ; Serial

604800 ; Refresh

86400 ; Retry

2419200 ; Expire

604800 ) ; Negative Cache TTL

; name servers

@ IN NS server1.jovanny.local.

@ IN NS server2.jovanny.local.

; PTR Records

4.1 IN PTR server1.jovanny.local.

4.3 IN PTR server2.jovanny.local.

2.1 IN PTR pc1.jovanny.local.

3.1 IN PTR pc2.jovanny.local.

4.2 IN PTR cliente1.jovanny.local.

5.3 IN PTR cliente2.jovanny.local.

1. Verificamos la creación de los archivos

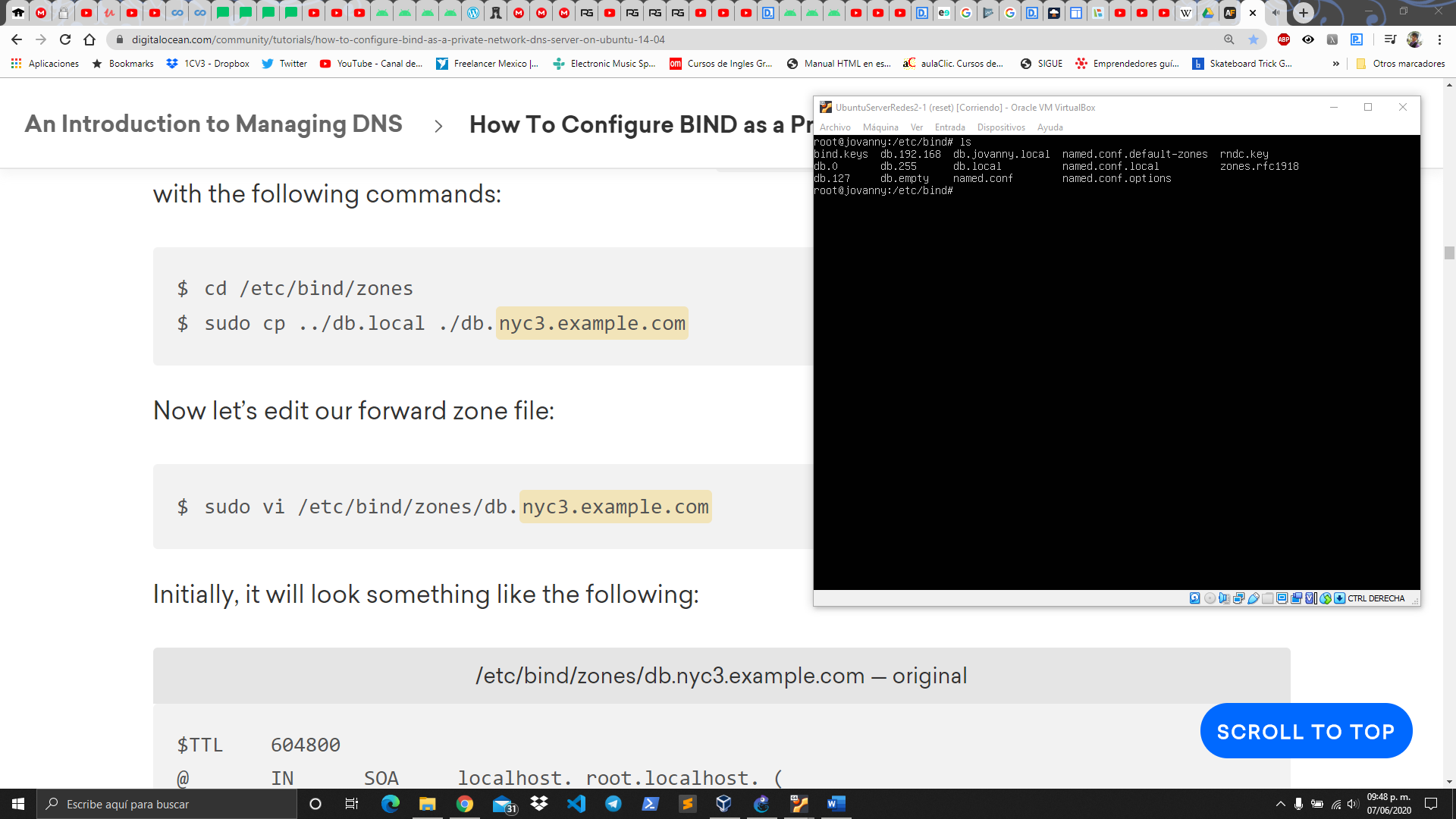


Imagen 4: archivos creados

1. Verificamos la sintaxis del archivo de configuración local

$sudo named-checkconf

1. Verificamos el nombre de la zona directa

$sudo named-checkzone Jovanny.local db.jovanny.local

1. Verificamos el nombre de la zona inversa

$sudo named-checkzone 168.192.in\_addr.arpa db.192.168

1. Reiniciar el servicio de bind9 y verificar el estado corra sin problemas

$sudo service tftpd-hpa restart

$sudo service tftpd-hpa status

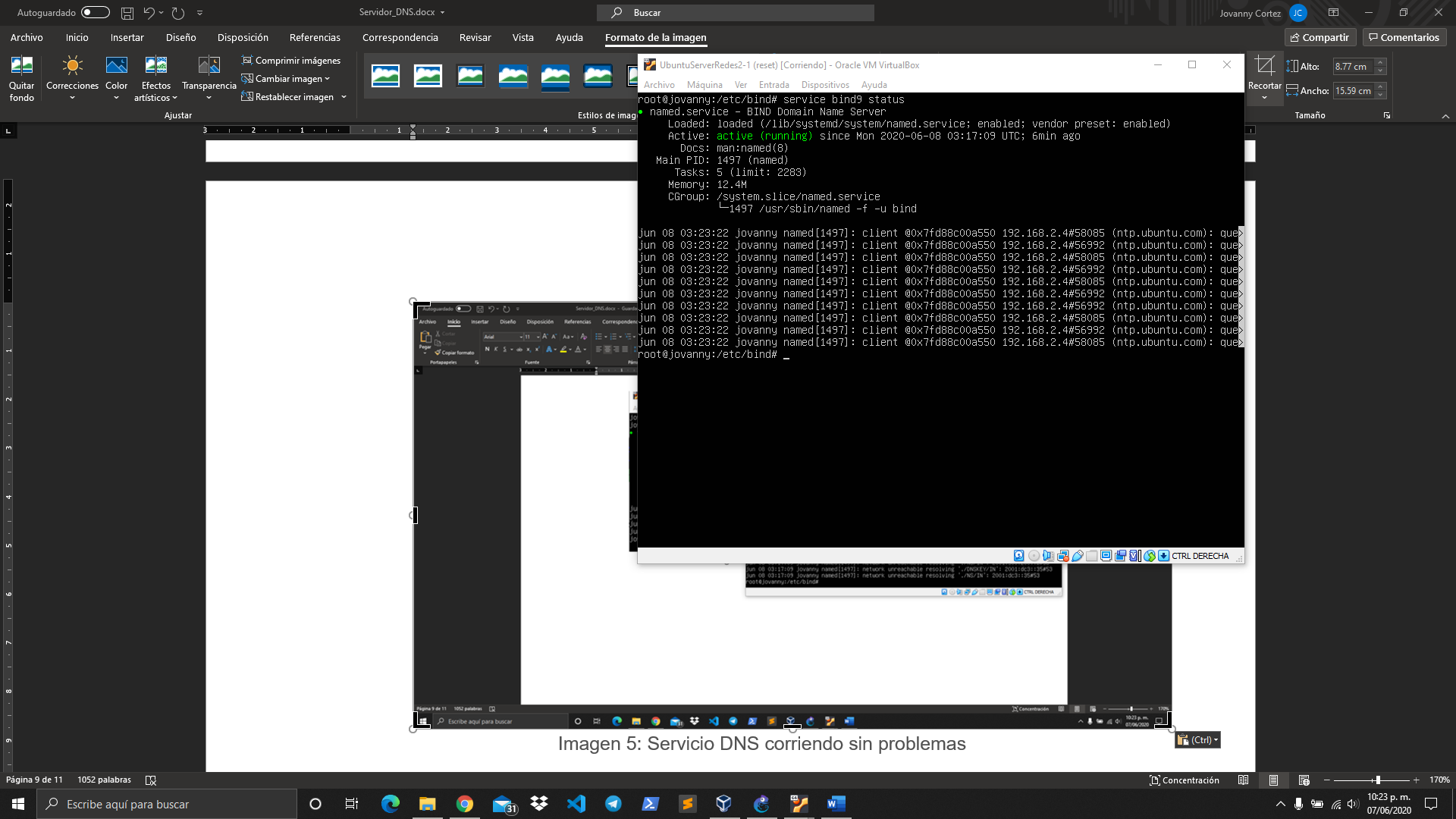


Imagen 5: Servicio bind9 corriendo sin problemas

## Configuración del segundo servidor DNS

Es más fácil configurar el servidor 2

1. Configurar las opciones en el archivo named.conf.options ubicado en “/etc/bind/named.conf.options”, se agrega dentro del bloque options las líneas siguientes:

recursion yes; #permite la recursion

allow-recursion { any; }; # cualquiera puede hacer las consultas

listen-on { 192.168.3.4; }; # de manera privada del servidor 2

allow-transfer { none; }; # desactiva la zona de transferencia por defecto.

1. Configurar el archivo local localizado en “/etc/bind/named.conf.local”
   1. Zona directa

zone "jovanny.local" {

type slave;

file "slaves/db.jovanny.local";

masters { 192.168.1.4; }; # ns1 private IP

};

* 1. Zona inversa

zone "168.192.in-addr.arpa" {

type slave;

file "slaves/db.192.168";

masters { 192.168.1.4; }; # ns1 private IP

};

1. Verificamos la sintaxis del archivo de configuración local

$sudo named-checkconf

1. servicio de bind9 y verificar el estado corra sin problemas

$sudo service bind9 restart

$sudo service bind9 status

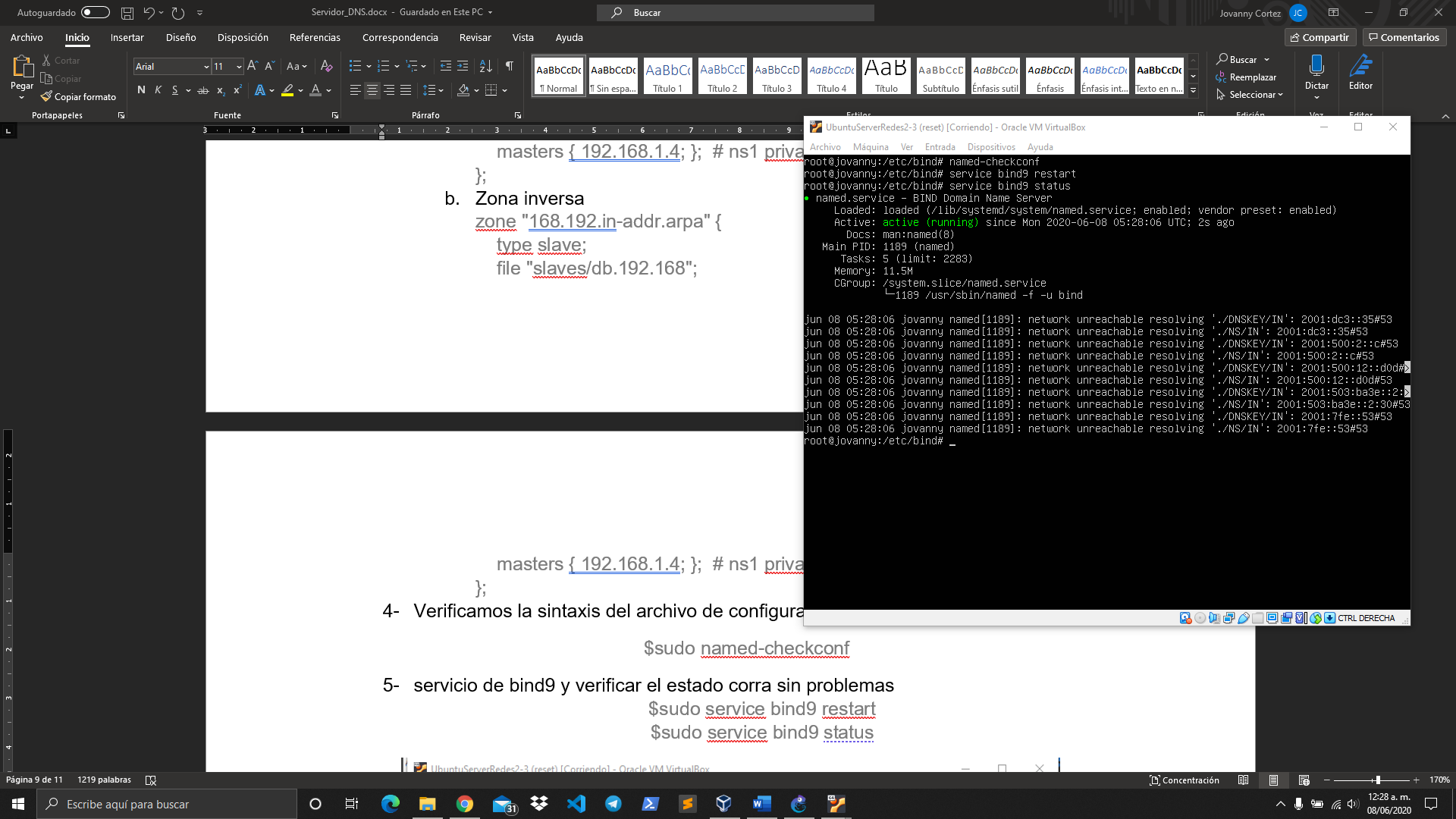


Imagen 6: Servicio corriendo en servidor 2

## Pruebas

1. Verificamos si tenemos instalado resolvconf, de lo contrario se instala con el comando:

$apt install resolvconf

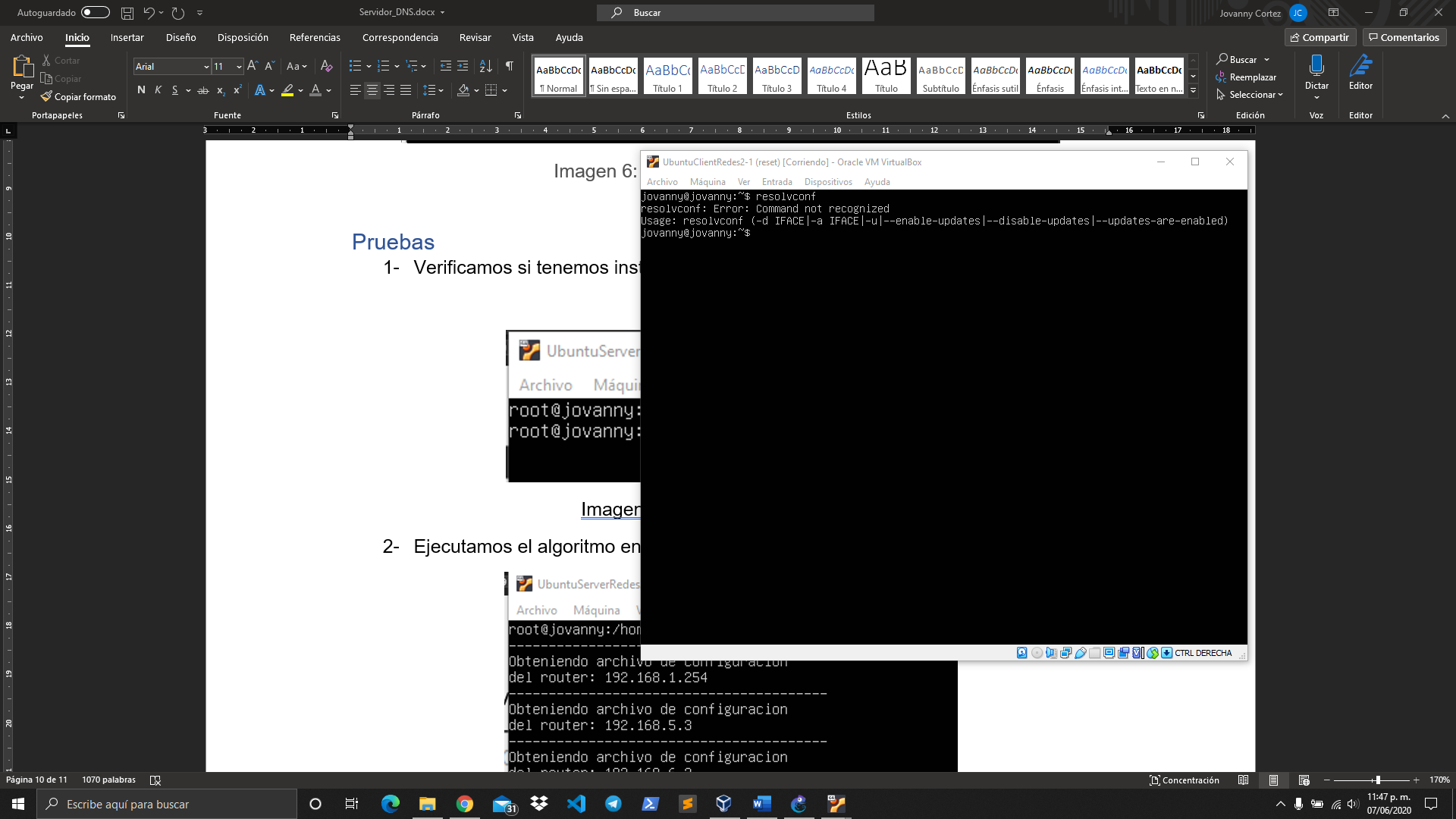


Imagen 7: comando resolvconf instalado.

1. Agregamos las siguientes líneas al archivo head localizado en:

Search jovanny.local # your private domain

nameserver 192.168.1.4 # ns1 private IP address

nameserver 192.168.3.4 # ns2 private IP address

1. Ejecutamos el comando:

$sudo resolvconf -u

1. Hacemos ping a los ip registrados.

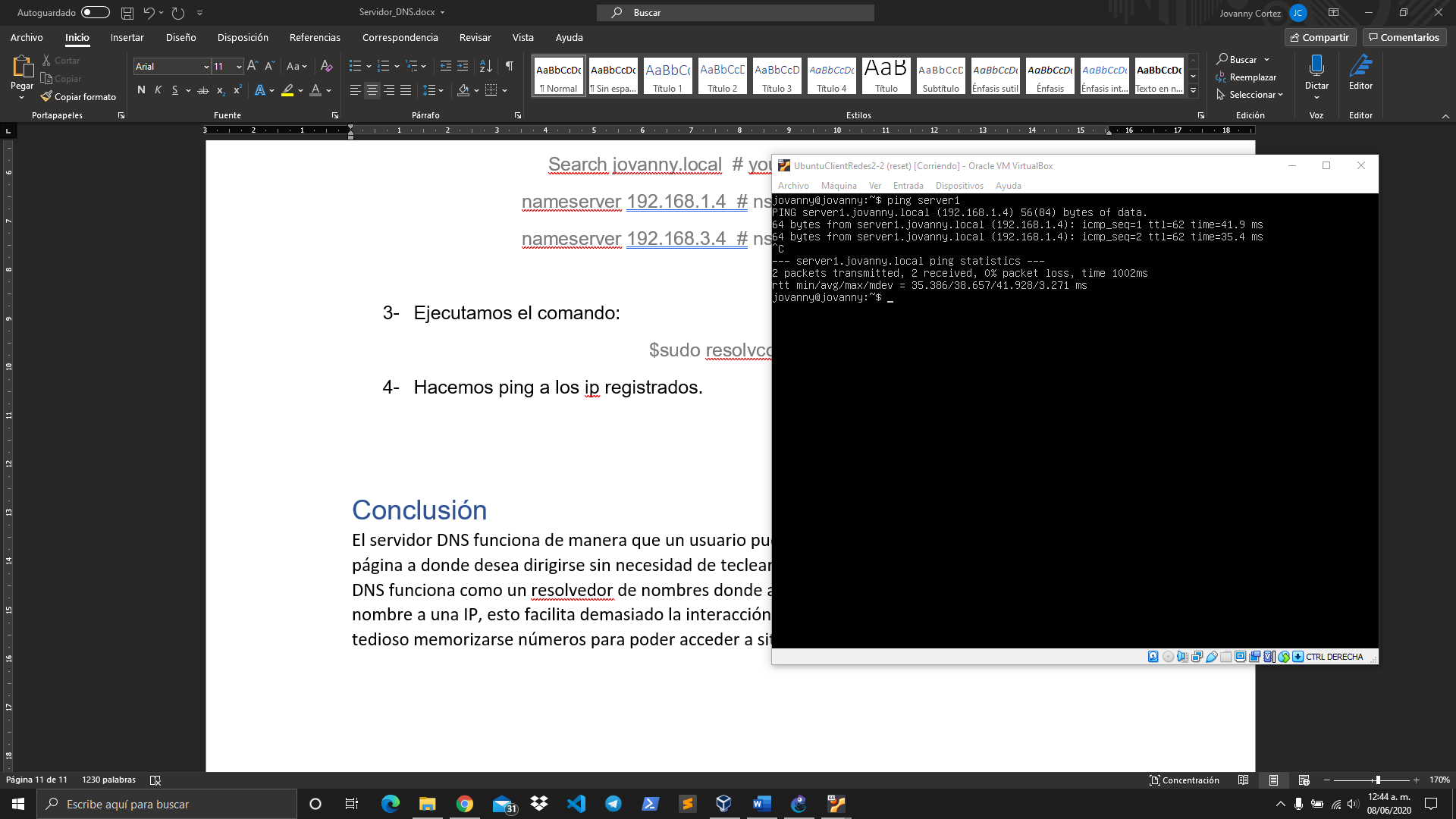


Imagen 8: petición ping resuelto con el nombre desde el cliente 2

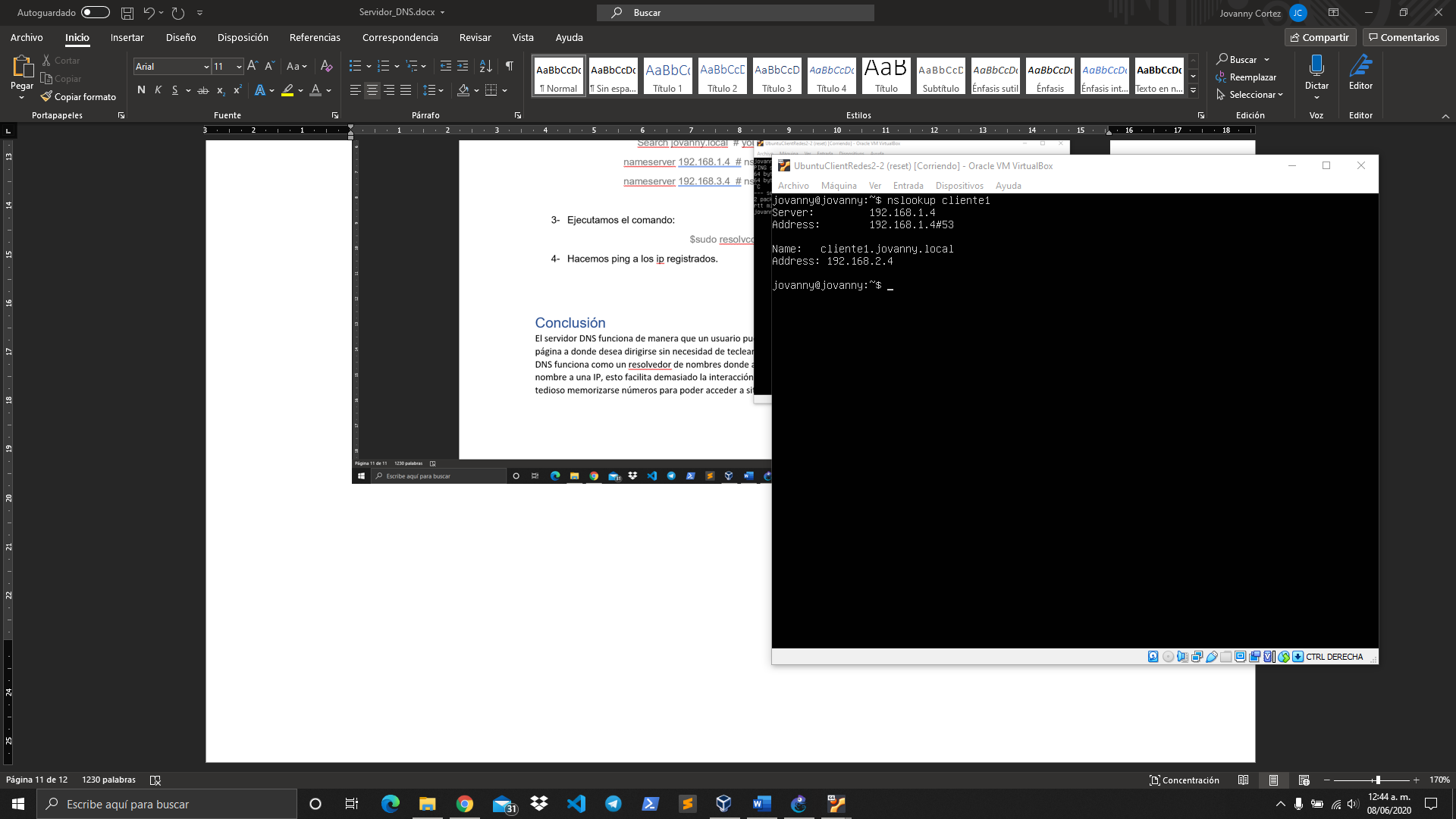


Imagen 9: comando nslookup resuelto por el servidor 1 desde el cliente2

1. Cuando el servidor uno cae, actúa el servidor secundario

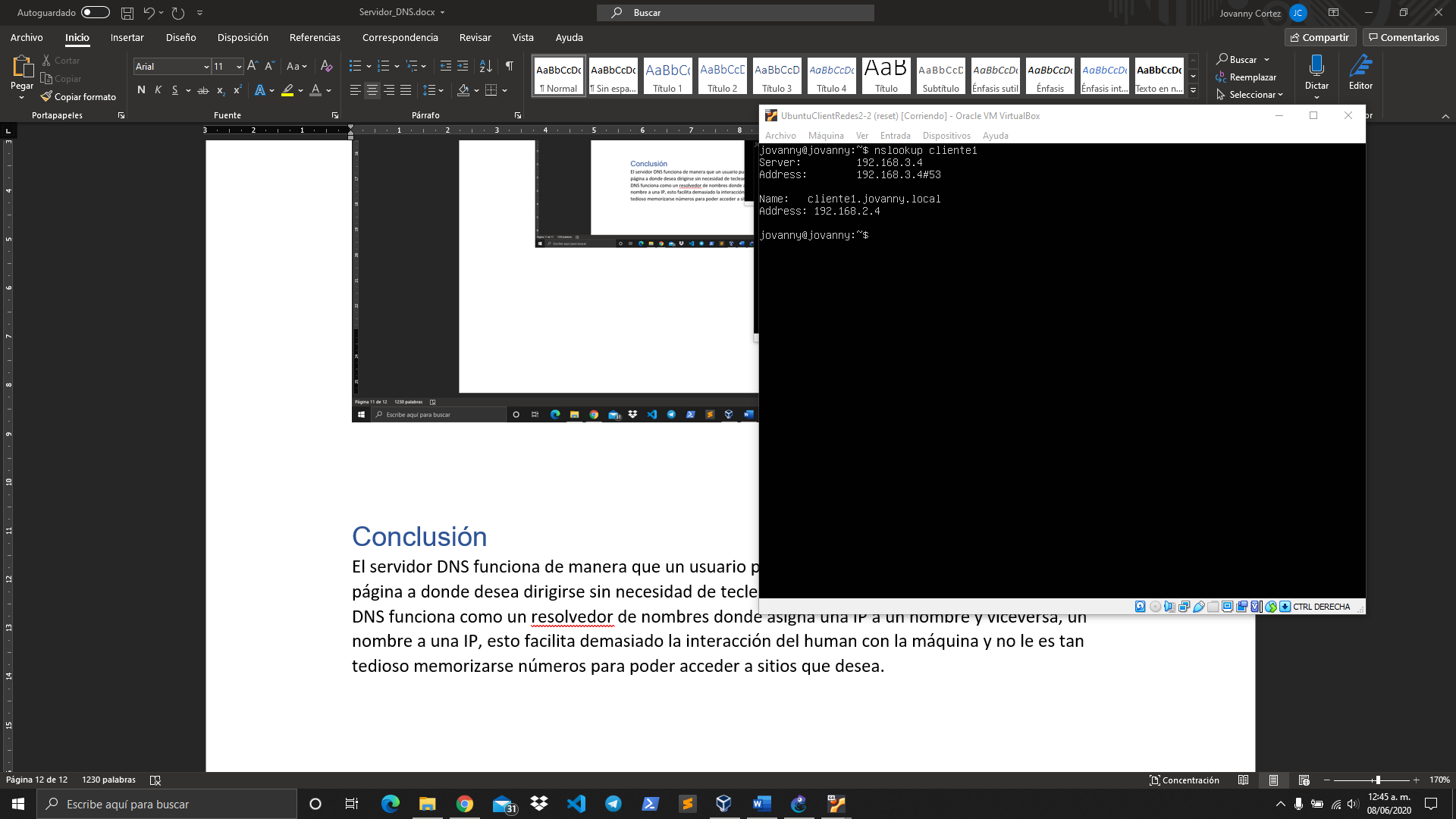


Imagen 10: comando nslookup resuelto por el servidor secundario desde el cliente2

# Conclusión

El servidor DNS funciona de manera que un usuario puede teclear un nombre y este le devuelve la página a donde desea dirigirse sin necesidad de teclear numero o la dirección IP en este caso, el DNS funciona como un resolvedor de nombres donde asigna una IP a un nombre y viceversa, un nombre a una IP, esto facilita demasiado la interacción del human con la máquina y no le es tan tedioso memorizarse números para poder acceder a sitios que desea.