

|  |
| --- |
| TFG del Grado en Ingeniería Informática  Simulación de redes con GNS3 |



Presentado por Javier García González

en Universidad de Burgos -- 21 de enero de 2019

Tutores: Alejandro Merino Gómez y Daniel Sarabia Ortiz



D. Alejandro Merino Gómez, profesor del departamento de Ingeniería Informática, área de <...>

Expone:

Que el alumno D. Javier García González, con DNI 71294819V, ha realizado el Trabajo final del GºIng.Informática titulado: Simulación de redes con GNS3.

y que dicho trabajo ha sido realizado por el alumno bajo la dirección del que suscribe, en virtud de lo cual, Se autoriza su presentación y defensa.

En Burgos a de <junio de 2016>

<nombre\_tutor>

**Resumen**

El objetivo de este proyecto es simular una red institucional compleja con el programa GNS3 analizando las opciones que este entorno nos ofrece y viendo la viabilidad para poder representar correctamente todos los elementos necesarios en la red. Se van a representar elementos como servidores, equipos sencillos, conexiones con el exterior, routers y switches, además se va a probar el uso de otros elementos como las máquinas virtuales y toda la configuración entre elementos.

**Descriptores**

GNS3, redes, servidor DHCP, servidor HTTP, docker, máquina virtual, Ubuntu, VirtualPCs, VirtualBox, VMWare, Python.

**Abstract**

*This proyect is about…..*

# **Índice general**

[**Índice general** 6](#_Toc535234095)

[**Índice de figuras** 8](#_Toc535234096)

[**Introducción** 9](#_Toc535234097)

[**Objetivos del proyecto** 10](#_Toc535234098)

[2.1 Lectura, información e instalación de la herramienta. 10](#_Toc535234099)

[2.2 Información sobre complementos a utilizar. 10](#_Toc535234100)

[2.3 Conexión de GNS3 con el exterior. 10](#_Toc535234101)

[2.4 Lanzar GNS3 contra un equipo servidor. 10](#_Toc535234102)

[2.5 Simulación de una red institucional. 10](#_Toc535234103)

[2.6 Configuración remota de la red. 11](#_Toc535234104)

[**Conceptos teóricos** 12](#_Toc535234105)

[3.1 Dockers: 12](#_Toc535234106)

[3.2 QEMU: 12](#_Toc535234107)

[3.3 Máquina Virtual: 12](#_Toc535234108)

[3.4 Servidor Web: 12](#_Toc535234109)

[3.5 Servidor DHCP [1]: 13](#_Toc535234110)

[3.6 Router [2]: 13](#_Toc535234111)

[3.7 Switch [3]: 13](#_Toc535234112)

[3.8 Red LAN: 13](#_Toc535234113)

[3.9 Script: 13](#_Toc535234114)

[3.10 Python: 13](#_Toc535234115)

[**Técnicas y herramientas** 14](#_Toc535234116)

[GNS3 14](#_Toc535234117)

[Máquinas virtuales 14](#_Toc535234118)

[VPCS 14](#_Toc535234119)

[Dockers 14](#_Toc535234120)

[Telnet - Python 14](#_Toc535234121)

[**Aspectos relevantes del desarrollo del proyecto** 15](#_Toc535234122)

[**Trabajos relacionados** 16](#_Toc535234123)

[**Conclusiones y Líneas de trabajo futuras** 17](#_Toc535234124)

[**Bibliografía** 18](#_Toc535234125)

# **Índice de figuras**

# **Introducción**

El objetivo del proyecto consiste en desarrollar y profundizar en las posibilidades de simulación de redes que nos ofrece el programa GNS3. Se trata de simular una red de computadores compleja en la que encontramos elementos como servidores web, servidores DNS, servidores DHCP, conexiones a Internet, filtrado de tramas en switches, routers con acceso externo a la red simulada, una red LAN, etc.

Además, se realiza un estudio teórico y práctico donde se comparan elementos como QEMU y Máquinas virtuales, utilización de Dockers, como conectar la red simulada con el exterior de forma que pueda verse para otros dispositivos como una red real o poder lanzar el programa contra un servidor.

La memoria se divide en varios apartados donde se encuentra una descripción de los objetivos perseguidos en el proyecto, un desglose de conceptos teóricos que veremos más adelante tanto en la propia memoria como en los anexos. Veremos las técnicas y las herramientas empleadas en el proyecto. Se desarrollan los aspectos más relevantes llevados a cabo en el proyecto. Veremos algunos trabajos relacionados con el descrito en este proyecto, así como las conclusiones a las que se han llegado y que posibles líneas de trabajo futuras se podrían llevar a cabo.

En los anexos vamos a encontrar el proceso completo de instalación y configuración del sistema desarrollado y las instrucciones de uso para poder hacer todo tipo de proyectos y comprobaciones en base al trabajo desarrollado.

# **Objetivos del proyecto**

## **2.1 Lectura, información e instalación de la herramienta.**

En este punto se empieza a conocer el programa, cómo funciona, que complementos son necesarios y como se instala. Podemos encontrar esta información en los puntos del trabajo: Conceptos Teóricos y Técnicas - Herramientas.

## **2.2 Información sobre complementos a utilizar.**

Comparativa de uso entre Dockers y Máquinas Virtuales.

Como utilizar Dockers, necesidad de Windows PROFESIONAL, uso de UBUNTU.

## **2.3 Conexión de GNS3 con el exterior.**

Investigar las formas de conectar nuestro proyecto simulado en GNS3 con las redes externas de forma que sea visible.

## **2.4 Lanzar GNS3 contra un equipo servidor.**

Ejecutar GNS3 contra otro equipo (en mi caso portátil contra el equipo de mesa, que hará de servidor).

## **2.5 Simulación de una red institucional.**

Parte principal del proyecto. Se trata de simular una red institucional completa en la cual se van a ir añadiendo ciertos elementos:

1. Servidor WEB
2. Servidor DHCP
3. Router acceso exterior
4. Switches
5. Red LAN
6. VLAN

## **2.6 Configuración remota de la red.**

Programar en lenguaje Python un script que conociendo la topología de una red nos permita configurar los equipos de forma remota una vez estén conectados.

# **Conceptos teóricos**

## **3.1 Dockers:**

También conocidos como contenedores. Se trata de emplear un software determinado que nos permite simular hardware virtual que a grandes rasgos veremos como un equipo totalmente independiente, este equipo podrá tener su propio sistema operativo en el que poder trabajar sin modificar el software anfitrión. La diferencia con las máquinas virtuales más adelante explicadas es que no se carga una imagen completa del sistema operativo, sino que solo se incluyen las partes necesarias.

## **3.2 QEMU:**

Consiste en un emulador de procesadores con capacidad de virtualización en el sistema operativo. En este tipo de sistemas no disponemos de entorno gráfico (se puede incluir con módulos adicionales para Windows). QEMU es capaz de emular un sistema completo capaz de mover varios sistemas operativos y en diferentes plataformas.

## **3.3 Máquina Virtual:**

En los casos de máquinas virtuales, se utiliza un software de virtualización como por ejemplo puede ser VirtualBox o VMWare (las dos utilizadas en este proyecto y gratuitas). En estos casos se instala una imagen de un sistema operativo completo sin dependencia del anfitrión ni con otros sistemas. Tan solo se comparte el hardware real.

## **3.4 Servidor Web:**

Se trata de un programa capaz de ejecutar una aplicación en el lado del servidor, es capaz de realizar conexiones en ambos sentidos o en un solo y también de realizar conexiones tanto en modo síncrono como asíncrono con el lado del cliente. Para procesar esta información el lado cliente necesita una aplicación conocida como navegador Web. Suele utilizarse un protocolo de comunicaciones conocido como HTTP.

## **3.5 Servidor DHCP**

[1] El servicio DHCP (Dynamic Host Configuration Protocol) es el protocolo de configuración dinámica de host, un estándar TCP/IP diseñado para simplificar la administración de la configuración IP de los equipos de nuestra red. El estándar DHCP permite el uso de servidores DHCP para administrar la asignación dinámica a los clientes DHCP de la red, de direcciones IP y de otros detalles de configuración relacionados con el direccionamiento IP, tales como la puerta de enlace o los servidores DNS, por ejemplo, siempre que los clientes estén configurados para utilizar un servidor DHCP, en lugar de estar configurados manualmente con una dirección IP estática.

## **3.6 Router**

[2] Es un dispositivo que proporciona conectividad a nivel de red (o lo que es lo mismo, el nivel tres en el modelo OSI). Su función principal consiste en enviar o encaminar paquetes de datos de una red a otra, es decir, interconectar subredes, entendiendo por subred un conjunto de máquinas IP que se pueden comunicar sin la intervención de un encaminador (mediante [puentes de red](https://es.wikipedia.org/wiki/Puente_de_red) o un switch), y que por tanto tienen prefijos de red distintos.

## **3.7 Switch**

 [3] Su función es interconectar dos o más host de manera similar a los puentes de red, pasando datos de un segmento a otro de acuerdo con la dirección MAC de destino de las tramas en la red y eliminando la conexión una vez finalizada ésta.

## **3.8 Red LAN**

Consiste en varios ordenadores conectados entra sí, normalmente en un espacio pequeño (sala, edificio, etc.) que permite que dichos equipos compartan recursos.

## **3.9 Script:**

Documento con código de programación que incluye instrucciones que permiten realizar las funciones para las cuales está diseñado.

## **3.10 Python:**

Se trata de un lenguaje de programación interpretado, multiplataforma y que dispone de licencia de código abierto.

# **Técnicas y herramientas**

## **GNS3**

La herramienta principal utilizada en el proyecto ha sido el programa GNS3, un simulador gráfico de redes que nos permite el diseño de complejas topologías y poder hacer simulaciones del comportamiento.

Se trata de una herramienta muy potente que nos permite seleccionar muchas opciones de configuración. En nuestro caso en lugar de lanzar la simulación sobre el propio equipo, se va a lanzar sobre una máquina virtual GNS3 sobre VMWare, como podremos contemplar en los anexos, donde veremos el proceso de instalación y configuración.

## **Máquinas virtuales**

Para poder utilizar GNS3 con toda su potencia, debemos utilizar también máquinas virtuales, para ello se hace uso de VMWare y de VirtualBox. Gracias a ello podemos simular equipos (en este proyecto se ha utilizado Ubuntu).

## **VPCS**

Se trata de emuladores de PC que vienen ya configurados en el programa GNS3 con funciones muy básicas para poder probar la red. Gracias a ellos podemos simular equipos que no nos requieran instalaciones ni servicios especiales.

Estos equipos por ejemplo no son útiles para realizar un servidor, pero si son muy útiles para poder simular un equipo terminal sin tener que emplear todos los recursos que nos consume una máquina virtual con un sistema operativo completo.

## **Dockers**

Se trata de virtualización como en los casos anteriores, pero en este tenemos la diferencia de que se configura solo parte del sistema operativo, en cada caso la parte que vayamos a necesitar. De esta forma conseguimos el mismo resultado que con un sistema completo, pero consiguiendo un considerable ahorro de recursos.

El uso de Dockers se ha estudiado apreciando una disminución de los recursos empleados por el equipo anfitrión que ejecuta el proyecto, pero por otra parte dificulta la configuración remota de la red, así como el proceso de exportar el proyecto para un uso posterior, de forma que se ha dejado como un estudio dentro del propio proyecto, pero finalmente se ha descartado su uso y se ha decidido utilizar máquinas virtuales y VPCS.

Es interesante el estudio ya que para proyectos que requieran mucha carga de equipos, o bien no necesitemos funciones como la configuración remota o no queramos exportarlo sí que es más que interesante su utilización.

La disminución de la carga de trabajo que producen es difícilmente medible, ya que depende de muchos factores, como el servicio final que dan o el número de equipos a los que acaban dando servicio. Pero si se puede asegurar que es apreciable dicha disminución.

## **Telnet – Python**

Para la realización de un programa que nos permita la configuración remota de la red se ha escogido Python, que nos permite utilizar Telnet para poder conectar a los dispositivos y realizar configuraciones básicas como son cambios de IP, máscara de red y puerta de enlace.

Se ha escogido Python por encima de otros lenguajes dado que es lenguaje sobre el que más documentación tenemos relativa al uso con redes, además de ser uno de los lenguajes más potentes en cuanto a funciones realizadas.

## **Metodología del proyecto**

## Para el desarrollo del proyecto se ha utilizado la herramienta GitHub, donde se han indicado los pasos seguidos en el proyecto y cada cambio, para ello se ha utilizado la opción de Issues (tareas), de commits (cambios) y se ha utilizado el módulo de proyecto donde se puede ver en cada momento que tareas están hechas, en tratamiento, pendientes, etc.

La metodología ha sido ágil, permitiendo adaptarse al proyecto realizado y cambiando las tareas en función de lo que la herramienta GNS3 nos permitía en cada caso, de esta forma muchos de los objetivos han sido modificados sobre la marcha para adaptarnos.

# **Aspectos relevantes del desarrollo del proyecto**

A lo largo del proyecto se han visto algunos apartados que han llevado mucho tiempo de estudio y se han implementado en poco tiempo, como es el caso de la [4] conexión de la red generada en GNS3 al exterior (Internet), del cual se ha encontrado mucha información y videos de uso, algunos ya descatalogados y otros más actuales. Finalmente se ha escogido el método más actual y se ha dejado la conexión dentro del proyecto.

En cambio, tenemos otro caso como es el del uso de [5] Dockers. La recopilación de información y el estudio de estas herramientas llevó mucho tiempo, su puesta en funcionamiento fue relativamente rápida en comparación con otros elementos, pero se tuvo que descartar su uso por chocar con otros de los objetivos. Esto no se puede considerar algo negativo, ya que se ha demostrado su utilidad, aunque en este proceso no haya sido la mejor opción y se haya descartado su uso. Podemos ver varios ejemplos y el proceso de instalación en los anexos: Apéndice C, apartado C.4, sección uso de Dockers.

Se ha realizado un estudio en el cual se ha comprobado que la cantidad de recursos que el programa GNS3 requiere del equipo anfitrión se puede disminuir si utilizamos un equipo como servidor que lance el programa GNS3 y sobre el cuál se consumen dichos recursos pero trabajamos desde un equipo cliente que es desde el cuál lanzamos el proyecto, este equipo consume ciertos recursos, pero la carga más fuerte se hará sobre el servidor, de esta forma se reparte el consumo de dichos recursos. Podemos ver el ejemplo en los anexos: Apéndice C, apartado C.4, sección Conectar GNS3 contra un servidor remoto.

Por otro lado, tenemos otros componentes cuyo estudio y documentación requiere menos tiempo, debido a que son elementos previamente utilizados a lo largo del grado y de los cuales entendemos su funcionamiento, en cambio la puesta en funcionamiento de estos elementos ha sido más larga y compleja. El servidor Apache fue complejo de implementar ya que las máquinas virtuales utilizadas inicialmente eran las del laboratorio de redes de la politécnica. Estas no nos permitían la instalación de servicios adicionales por temas de permisos, por lo que se descargó e instaló desde cero una máquina virtual con todos los permisos y con la conexión a Internet activa inicialmente, aunque luego para el proyecto se tuviese que volver a dejar sin conexión. Una vez conseguido este paso se realizó la configuración como servidor Apache con todas sus funciones y se estableció una página que nos permite obtener los datos de la red del servidor para poder comprobar su funcionamiento.

En cuanto a las redes virtuales VLAN, no se encontraron grandes problemas, pero al ser un punto con varios aspectos de configuración y necesitar de un Switch que inicialmente daba otros servicios, se decidió aislar la red con su propio switch, de esta forma su funcionamiento es más independiente y cumple mejor su labor.

# **Trabajos relacionados**

Los trabajos relacionados encontrados han sido los siguientes:

Autor: Héctor Delgado Patiño.

Título: "Construcción de una red MPLS y validación de GNS3 para su simulación".

Organización: PFC Universidad de Cartagena.

Repositorio: http://repositorio.upct.es/bitstream/handle/10317/5251/tfg682.pdf?sequence=1

Autor:   
Rubén Isa Hidalgo .

Título:   
"Implementación de un laboratorio virtual para aprendizaje de SDN".

Organización:   
TFG Universidad de Cantabria.

Repositorio: <https://repositorio.unican.es/xmlui/bitstream/handle/10902/9377/387880.pdf>

Trabajos prácticos relacionados: https://github.com/GNS3/gns3-gui/releases

# **Conclusiones y Líneas de trabajo futuras**

……

# **Bibliografía**

[1] <https://2003server.webcindario.com/dhcp/definici.htm>

[2] Router. (2018, 7 de noviembre). *Wikipedia, La enciclopedia libre*. Fecha de consulta: 11:38, enero 14, 2019 desde <https://es.wikipedia.org/w/index.php?title=Router&oldid=111831803>.

[3] Conmutador (dispositivo de red). (2018, 16 de noviembre). *Wikipedia, La enciclopedia libre*. Fecha de consulta: 11:39, enero 14, 2019 desde <https://es.wikipedia.org/w/index.php?title=Conmutador_(dispositivo_de_red)&oldid=112065037>.

[4] <https://docs.gns3.com/1vFs-KENh2uUFfb47Q2oeSersmEK4WahzWX-HrMIMd00/index.html>   
<https://www.youtube.com/watch?v=tgJgUyja0XU>

[5]<https://docs.gns3.com/14EmmKdryY3FiMOQEclSyHQ3MUlycDGE_DwNPC8L4GIc/index.html>

[6]

[7]