

UNIVERSIDAD POLITÉCNICA DE PUEBLA
Ingeniería en Informática



**Proyecto de Estancia Práctica en
Desarrollador en Sistemas de Software y
Administrador de Redes**

“Monitoreo de servidores con Icinga”

Área temática del CONACYT: VII
Ingenierías y tecnologías

Presenta:

Rubén Ely Trujillo Pérez

Asesor técnico

Ing. Rigoberto Martínez López

Asesor académico

M.C Rebeca Rodríguez Huesca

Resumen

La organización Consultoría Contracorriente S.A. de C.V. presenta un riesgo permanente de ataques y sobre carga de usuarios en sus portales, cuenta con un sistema de monitoreo Nagios el cual tarda en notificar a los encargados del área de algún problema, lo cual genera que el tiempo de respuesta sea muy prolongado.

El proyecto tiene la finalidad de implementar un sistema de monitoreo Icinga2, el cual mejora el tiempo de respuesta a las fallas que se presenten ya se generan scripts específicos para cada servidor e Icinga se encarga de revisar constantemente cada uno de ellos. En el momento que Icinga detecta una falla, con los scripts generados la resuelve y notifica a la persona encargada de la falla y si fue resulta.

También se hace la implementación de una interfaz web llamada icignaweb2, la cual permite al área encarga de los servidores tener una interpretación gráfica de los servidores y ver cuáles tienen o presentaron fallas a lo largo de un determinado tiempo, poder acceder desde cualquier parte y no tener que entrar al servidor del monitor para ver los errores que se han registrado.

Toda la implementación del monitor se hizo en un servidor con CentOS y los servidores a monitorear con CentOS y Ubuntu.

Índice

1. Introducción.....	4
1.1 Descripción del problema o necesidad	4
1.2 Justificación	4
1.3 Objetivo General y Específicos	5
2. Metodología y herramientas.....	6
2.1 Proceso.....	6
2.2 Conceptos básicos.....	6
2.3 Herramientas y especificaciones técnicas.....	8
2.4 Servicios ejecutados dentro de los servidores remotos	10
3. Resultados	12
3.1 Instalación de Icinga	12
3.2 Configuración de la base de datos	14
3.3 Instalación de la interfaz web Icinga2	15
3.4 Agregando los servidores al monitor.....	20
4. Conclusiones y recomendaciones.....	22
5. Referencias bibliográficas	23

1. Introducción

En este primer capítulo se hará una descripción de la problemática y la solución, así como también el objetivo general y los objetivos específicos para llevar a cabo este proyecto.

1.1 Descripción del problema o necesidad

Actualmente la organización Consultoría Contracorriente S.A. de C.V. presenta un riesgo permanente de ataques y sobrecarga de usuarios en sus portales, los portales cuentan con un sistema de detección de errores Nagios que tarda en notificar al programador en turno que los mismos no están disponibles. El mantener los portales no disponibles por un periodo de tiempo muy prolongado conlleva a la pérdida de ingresos económicos por parte de las cuentas comerciales que tienen contrato con la organización.

1.2 Justificación

En el mundo actual, donde la tecnología es una herramienta muy útil para la vida cotidiana. La mayoría de todas las actividades que realizamos día con día de alguna u otra manera está ligada con los servicios en internet. Estas aplicaciones funcionan gracias a equipos especiales que ejecutan procesos de acuerdo al tipo de aplicación que alojan llamados servidores.

Sí un servidor llegase a fallar además de dejar de proveer sus servicios desencadena una serie de problemas mucho mayores si más servidores dependen de éste. A tal grado de dejar sin funcionar a sitios, aplicaciones o sistemas completos. Es por ellos que un monitoreo es de vital importancia en empresas que sus proyectos dependes de estos equipos.

Dados los requerimientos de solución del medio de comunicación por su naturaleza son muy específicos, se propone la implementación de un monitor Icinga que ayude a corregir de manera inmediata los ataques y/o sobrecargas en el servidor HTTP y/o en la base de datos del portal.

Con esto se quiere reducir el tiempo de respuesta a la atención de los problemas básicos que se presenten, ya que tanto los contenidos y las estrategias de monetización que tiene la organización no pueden permanecer inactivas en ningún momento.

1.3 Objetivo General y Específicos

Objetivo general:

Implantar un monitoreo avanzado para evitar el tiempo caído de servidores de producción.

La intervención que se llevará a cabo constará de los siguientes objetivos específicos:

1. Instalar Icinga en el servidor para concentrar el monitoreo en el mismo.
2. Crear monitoreos personalizados con scripts en bash.
3. Configuración de controladores definidos para la resolución proactiva de problemas.
4. Creación de scripts complejos para la monitorización de los servidores
5. Mostrando los resultados en Icinga2.

2. Metodología y herramientas

En este segundo capítulo se presentará una aproximación al proceso metodológico y con los conceptos básicos que se han aplicado a lo largo de este proyecto.

2.1 Proceso

Es importante destacar que con los servidores no hay una metodología que seguir para trabajar con ellos, a continuación, se mencionaran los pasos que se siguieron para montar un monitor de Icinga.

2.1.1 Instalación de Icinga

Para la instalación de Icinga se ha utilizado una distribución CentOS 6.10 ya que es la distribución con la que cuenta el servidor previamente utilizado para la instalación del monitor a sustituir llamado Nagios.

2.1.2 Configuración de la base de datos de Icinga

Durante la instalación de Icinga en nuestro servidor será necesario configurar la base de datos sobre la que trabajará nuestro monitor.

Usaremos MySQL que es una base de datos de código abierto, en la cual solo tendremos que definir una nueva base de datos, un usuario con contraseña e Icinga se encargara de hacer la importación de todas las tablas que usara para su buen funcionamiento.

2.1.3 Instalación de la interfaz web Icinga2

Icinga no cuenta con una interfaz web al momento de la instalación, por ello se le instalara la interfaz web Icinga2 para tener un mejor control sobre el monitoreo de los diversos servidores.

2.1.4 Agregando los servidores al monitor

Para poder agregar un nuevo servidor o host a nuestro monitor tenemos que entrar por línea de comandos a nuestro servidor donde se encuentra instalado Icinga, tendremos que modificar el archivo llamado "hosts.cfg" que es donde se encuentra la dirección IP del servidor que agregaremos a nuestro monitor.

2.2 Conceptos básicos

Dentro de las etapas de seguimiento se llevó a cabo la instalación de las herramientas a ocuparse durante el proyecto con base en lo objetivos anteriormente planteados.

2.2.1 Internet

El Internet (o, también, la Internet) es un conjunto de descentralizado de redes de comunicación interconectadas que utilizan la familia de protocolos TCP/IP, lo cual garantiza que las redes físicas heterogéneas que la componen funcionen como una red lógica única de alcance mundial. Sus orígenes se remontan a 1969, cuando se estableció la primera conexión de computadoras, conocida como Arpanet, entre tres universidades en California (Estados Unidos) [1].

2.2.2 Arquitectura lógica de redes

La topología de una red es el arreglo físico o lógico en el cual los dispositivos o nodos de una red se interconectan entre si sobre un medio de comunicación [2].

2.2.3 Cliente servidor

El modelo cliente servidor generalmente adopta la forma de un mensaje de solicitud del cliente hacia el servidor pidiendo que se efectúe alguna tarea específica. De esta forma el servidor recibe la petición, realiza la tarea y devuelve el resultado de ésta. Regularmente, muchos clientes utilizan un número pequeño de servidores, aunque esto depende del tipo de servicios que requiera el cliente para las tareas que realiza.

Un servidor es un nodo que forma parte de una red, provee servicios a otros nodos denominados clientes, estos clientes dependen de manera directa de los servicios y datos que proveen los servidores.

En este modelo, un servidor atiende peticiones de los clientes como se muestra en la figura 1, y además el servidor mismo puede ser cliente de algún otro servicio a la vez. El cliente envía en la petición los datos que requiere que sean procesados o analizados, el servidor los recibe y efectúa los procedimientos que estén definidos en él, los cuales dependen naturalmente de la naturaleza del servicio que presta en la red. Posterior a esto, regresa el resultado del procesamiento, y el cliente los recibe e interpreta a través de algún software como navegadores web y así termina satisfactoriamente el modelo [3].

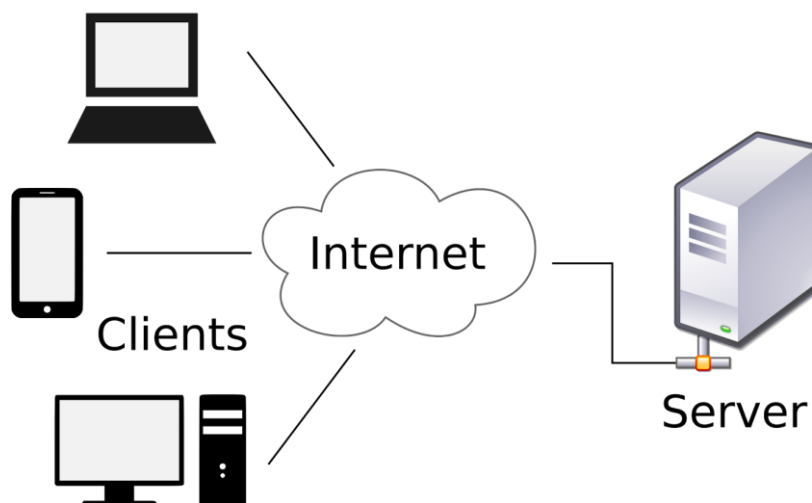


Figura 1 Arquitectura cliente-servidor.

2.2.4 Servicios en la nube

Los servicios de cómputo en la nube proporcionan tecnología de la información (TI) como un servicio a través de Internet o una red dedicada, con entrega según demanda y pago según el uso. Los servicios de cómputo en la nube abarcan desde plataformas de desarrollo y aplicaciones completas hasta servidores, almacenamiento y equipos de escritorio virtuales [4].



Figura 2 Servicios de la nube.

2.2.5 Monitoreo

Teniendo en cuenta los modelos bajo los cuales pueden trabajar los dispositivos en internet, es de gran importancia tomar en cuenta las características de hardware en los dispositivos ya que de ello depende el rendimiento de los mismos; rendimiento que se traduce en respuestas a peticiones, procesamiento de datos, soporte a múltiples conexiones y datos a almacenar. Otro aspecto a destacar es la seguridad de los dispositivos ya que nunca se ha de descartar los ataques malintencionados para así poder extraer información o simplemente causar algún daño en el rendimiento de los dispositivos.

El monitoreo de los recursos y de la seguridad de los dispositivos ayuda al personal de TI a mitigar problemas de causa natural o problemas causados con intención por personas ajenas o propias de la empresa.

2.3 Herramientas y especificaciones técnicas

Es importante destacar las características de los equipos donde residen las herramientas, debido a que estas se alojan junto con los proyectos y el monitoreo

no debe intervenir con procesos de producción. Entonces por ende el hardware y configuración de los servers debe ser la óptima posible.

2.3.1 Icinga

Es un sistema de monitoreo de software libre, utiliza la licencia GNU GPL, se trata de un fork de Nagios creado en 2009 por un grupo de desarrolladores de la comunidad. Es un sistema de monitorización que que vigila las redes y cualquier recurso de red concebible, notifica a usuario los errores y recuperaciones y genera datos de rendimiento para los informes y según los desarrolladores del proyecto añade más funcionalidades a Nagios, entre las que se encuentra una moderna interfaz web, más adaptada la Web 2.0 También incorpora conectores adicionales para bases de datos (MySQL/MariaDB, Oracle y PostgreSQL) Otra importante mejora una REST API para que los desarrolladores puedan crear nuevas extensiones [5].

Características principales:

- Monitorización de servicios de red (SMTP, POP3, HTTP, NTP, ping, etcétera)
- Monitorización de componentes de red (switches, routers, etcétera)
- Notificación a usuarios por correo electrónico
- Nivel de alertas
- Dos interfaces web opciones (Icinga Clasic UI e Icinga Web)
- Gráficos de rendimiento (PNP4Nagios, NagiosGrapher, etcétera)
- Herramientas de configuración e interfaz (Nconf, NagiosSQL)
- Monitorización de procesos de negocio (NagVIS, Nagmap)
- Monitorización de host Windows (NSClient++, Cygwin)

2.3.3 Servidor dedicado Icinga

Se optó por rentar un servidor dedicado exclusivamente para el alojamiento de la herramienta Icinga debido a la cantidad de monitoreos a configurar, y por la carga que suponen estos procesos no sería posible alojar Icinga en un servidor de producción.

Las especificaciones de hardware del servidor dedicado son:

- Procesador Intel Xeon de 11 nucleos a 2.4 GHz.
- 1 GB en memoria RAM.
- 200 GB en disco duro.
- Sistema operativo CentOS en su versión 6.10.
- Sin memoria Swap.

2.3.4 Servidores remotos o clientes a monitorear

Los servidores a monitorear varían en cuanto a características debido a que los proyectos y servicios que alojan dependen de los fines de los mismos.

- Procesadores Intel Xeon de 2 a 6 núcleos.
- De 1 a 4 GB's en memoria RAM.
- Discos duros de capacidades de entre 50 a 500 GB's dependiendo el tipo de disco duro, sólidos o mecánicos.
- Memorias Swap de entre 1 a 10 GB's.

2.4 Servicios ejecutados dentro de los servidores remotos

MYSQL

MySQL es un sistema de gestión de bases de datos relacional, multihilo y multiusuario con más de seis millones de instalaciones.¹ MySQL AB —desde enero de 2008 una subsidiaria de Sun Microsystems y ésta a su vez de Oracle Corporation desde abril de 2009— desarrolla MySQL como software libre en un esquema de licenciamiento dual [6].

Ventajas

1. MySQL es de uso libre y gratuito.
2. Software con Licencia GPL.
3. Bajo costo en requerimientos para la elaboración y ejecución del programa.
4. No se necesita disponer de Hardware o Software de alto rendimiento para la ejecución del programa.
5. Velocidad al realizar las operaciones y buen rendimiento.
6. Facilidad de instalación y configuración.
7. Soporte en casi el 100% de los sistemas operativos actuales.

Desventajas

1. Al ser de Software Libre, muchas de las soluciones para las deficiencias del software no están documentados ni presentan documentación oficial.
2. Muchas de sus utilidades tampoco presentan documentación.
3. Se debe controlar/monitorizar el rendimiento de las aplicaciones en busca de fallos.
4. No es el más intuitivo de los programas que existen actualmente para todos los tipos de desarrollos.
5. No es tan eficaz en aplicaciones que requieran de una constante modificación de escritura en BD.

MongoDB

MongoDB (de la palabra en inglés “humongous” que significa enorme) es un sistema de base de datos NoSQL orientado a documentos, desarrollado bajo el concepto de código abierto. MongoDB forma parte de la nueva familia de sistemas de base de datos NoSQL [7].

En vez de guardar los datos en tablas como se hace en las bases de datos relacionales, MongoDB guarda estructuras de datos en documentos tipo JSON con un esquema dinámico (MongoDB llama ese formato BSON), haciendo que la integración de los datos en ciertas aplicaciones sea más fácil y rápida.

Ventajas

1. Validación de documentos
2. Motores de almacenamiento integrado
3. Menor tiempo de recuperación ante fallas

Desventajas

1. No es una solución adecuada para aplicaciones con transacciones complejas
2. No tiene un reemplazo para las soluciones de herencia
3. Aún es una tecnología joven

Apache

El servidor HTTP Apache es un servidor web HTTP de código abierto, para plataformas Unix (BSD, GNU/Linux, etc.), Microsoft Windows, Macintosh y otras, que implementa el protocolo HTTP/1.1 y la noción de sitio virtual [8].

Ventajas

1. El código fuente de Apache está disponible de forma gratuita para cualquier persona y no se requiere licencia.
2. Se puede modificar para ajustar el código y también para corregir errores.
3. La capacidad de agregar más funciones y módulos lo convierte en uno de los favoritos entre los técnicos.
4. Es altamente confiable y se desempeña mejor.
5. Se puede instalar fácilmente.

Desventajas

1. Una de las características destacadas de Apache es su capacidad para modificar su configuración. Esto, sin embargo, puede causar una seria amenaza a la seguridad, si no se trata adecuadamente
2. Al crear un protocolo personalizado, también se crearán nuevos errores. Así surge la necesidad de depuradores.
3. Requiere una política de actualización estricta que debe realizarse regularmente sin falta.

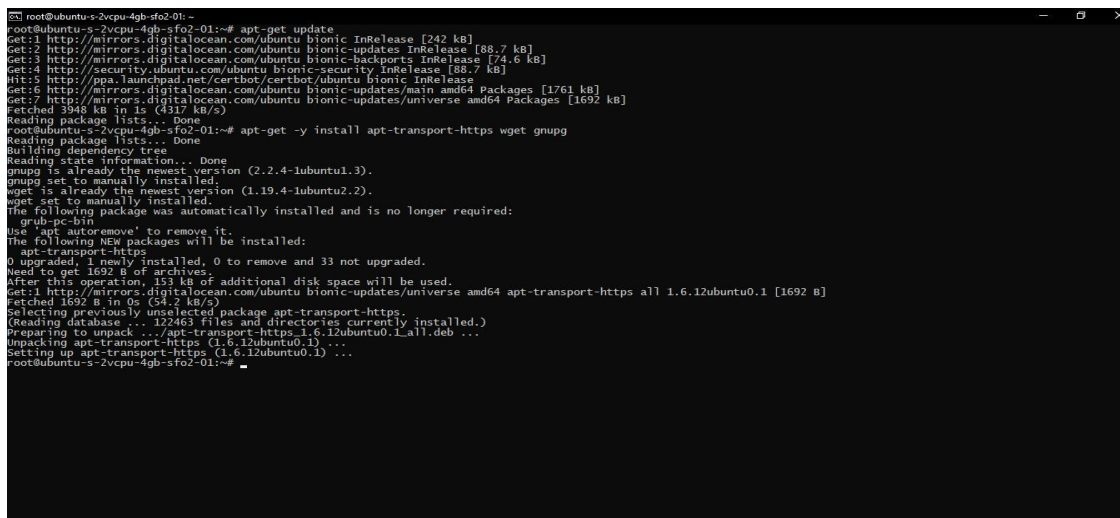
3. Resultados

En el siguiente capítulo se abordará los resultados que se obtuvieron con la instalación de Icinga2 (Por motivos de seguridad se ocultaron las direcciones IP de los servidores, contraseñas de los mismos).

3.1 Instalación de Icinga

Para iniciar la instalación de Icinga2 se debe actualizar el repositorio de nuestro sistema operativo utilizando el comando como se muestra en la Figura 3:

`apt-get update`

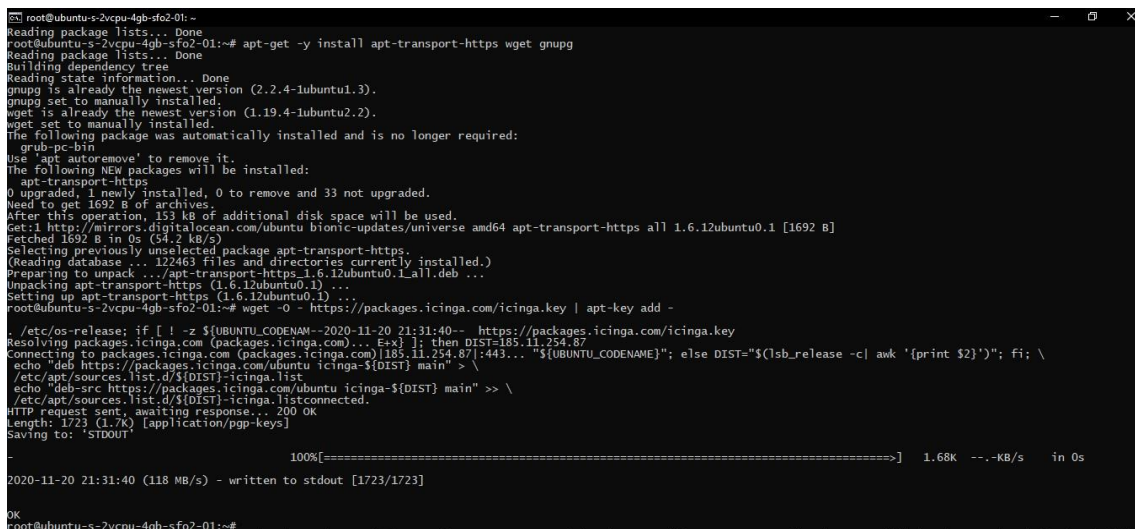


```
root@ubuntu-s-2vcpu-4gb-sfo2-01:~# apt-get update
Get:1 http://mirrors.digitalocean.com/ubuntu bionic InRelease [242 kB]
Get:2 http://mirrors.digitalocean.com/ubuntu bionic-updates InRelease [88.7 kB]
Get:3 http://mirrors.digitalocean.com/ubuntu bionic-backports InRelease [74.6 kB]
Get:4 http://security.ubuntu.com/ubuntu bionic-security InRelease [88.7 kB]
Hit:5 http://ppa.launchpad.net/certbot/certbot/ubuntu bionic InRelease
Get:6 http://mirrors.digitalocean.com/ubuntu bionic-updates/main amd64 Packages [1761 kB]
Get:7 http://mirrors.digitalocean.com/ubuntu bionic-updates/universe amd64 Packages [1692 kB]
Fetched 3948 kB in 1s (4317 kB/s)
Reading package lists... Done
root@ubuntu-s-2vcpu-4gb-sfo2-01:~# apt-get -y install apt-transport-https wget gnupg
Reading package lists... Done
Building dependency tree
Reading state information... Done
gnupg is already the newest version (2.2.4-1ubuntu1.3).
gnupg set to manually installed.
wget is already the newest version (1.19.4-1ubuntu2.2).
wget set to manually installed.
The following package was automatically installed and is no longer required:
  grub-pc-bin
Use 'apt autoremove' to remove it.
The following NEW packages will be installed:
  apt-transport-https
0 upgraded, 1 newly installed, 0 to remove and 33 not upgraded.
Need to get 1692 B of archives.
After this operation, 153 kB of additional disk space will be used.
Get:1 http://mirrors.digitalocean.com/ubuntu bionic-updates/universe amd64 apt-transport-https all 1.6.12ubuntu0.1 [1692 B]
Fetched 1692 B in 0s (54.2 kB/s)
Selecting previously unselected package apt-transport-https.
(Reading database ... 122463 files and directories currently installed.)
Preparing to unpack .../apt-transport-https_1.6.12ubuntu0.1_all.deb ...
Unpacking apt-transport-https (1.6.12ubuntu0.1) ...
Setting up apt-transport-https (1.6.12ubuntu0.1) ...
root@ubuntu-s-2vcpu-4gb-sfo2-01:~#
```

Figura 3 Actualizando los paquetes de S.O.

Después usar el comando, nos da como resultado lo que se muestra en la Figura 4:

`apt-get -y install apt-transport-https wget gnupg`



```
root@ubuntu-s-2vcpu-4gb-sfo2-01:~# apt-get -y install apt-transport-https wget gnupg
Reading package lists... Done
root@ubuntu-s-2vcpu-4gb-sfo2-01:~# apt-get -y install apt-transport-https wget gnupg
Reading package lists... Done
Building dependency tree
Reading state information... Done
gnupg is already the newest version (2.2.4-1ubuntu1.3).
gnupg set to manually installed.
wget is already the newest version (1.19.4-1ubuntu2.2).
wget set to manually installed.
The following package was automatically installed and is no longer required:
  grub-pc-bin
Use 'apt autoremove' to remove it.
The following NEW packages will be installed:
  apt-transport-https
0 upgraded, 1 newly installed, 0 to remove and 33 not upgraded.
Need to get 1692 B of archives.
After this operation, 153 kB of additional disk space will be used.
Get:1 http://mirrors.digitalocean.com/ubuntu bionic-updates/universe amd64 apt-transport-https all 1.6.12ubuntu0.1 [1692 B]
Fetched 1692 B in 0s (54.2 kB/s)
Selecting previously unselected package apt-transport-https.
(Reading database ... 122463 files and directories currently installed.)
Preparing to unpack .../apt-transport-https_1.6.12ubuntu0.1_all.deb ...
Unpacking apt-transport-https (1.6.12ubuntu0.1) ...
Setting up apt-transport-https (1.6.12ubuntu0.1) ...
root@ubuntu-s-2vcpu-4gb-sfo2-01:~# wget -O - https://packages.icinga.com/icinga.key | apt-key add -
./etc/os-release; if [ ! -z "${UBUNTU_CODENAME}" ]; then DIST=$(cat /etc/os-release | grep ^VERSION_CODENAME= | sed -e 's/\"//g'); else DIST=$(lsb_release -c | awk '{print $2}'); fi; \
echo deb-src https://packages.icinga.com/ubuntu icinga-${DIST} main >> \
/etc/apt/sources.list.d/${DIST}-icinga.list
Connecting to packages.icinga.com (packages.icinga.com) [185.11.254.87]:443... "${UBUNTU_CODENAME}"; else DIST=$(lsb_release -c | awk '{print $2}'); fi; \
echo deb-src https://packages.icinga.com/ubuntu icinga-${DIST} main" >> \
/etc/apt/sources.list.d/${DIST}-icinga.listconnected.
HTTP request sent, awaiting response... 200 OK
length: 1723 (1.7K) [application/pgp-keys]
Saving to: 'STDOUT'

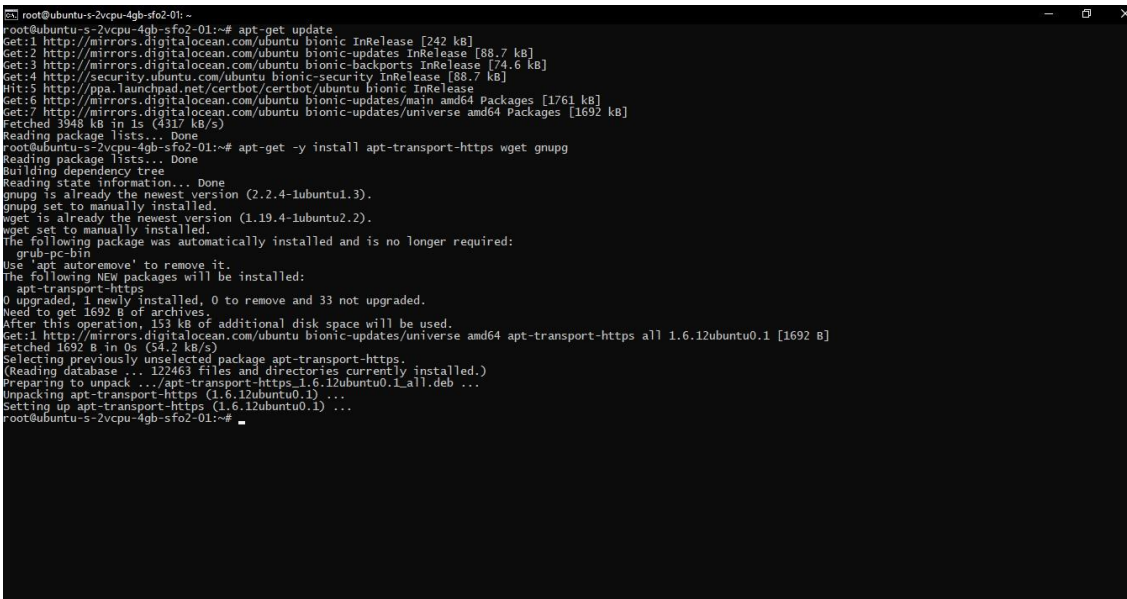
100%[=====] 1.68K --.-KB/s in 0s

2020-11-20 21:31:40 (118 MB/s) - written to stdout [1723/1723]
OK
root@ubuntu-s-2vcpu-4gb-sfo2-01:~#
```

Figura 4 Instalación de paquetes necesarios para Icinga.

Para finalizar la actualización de nuestro repositorio se tiene que volver a ejecutar el comando como se muestra en la Figura 5:

```
apt-get update
```

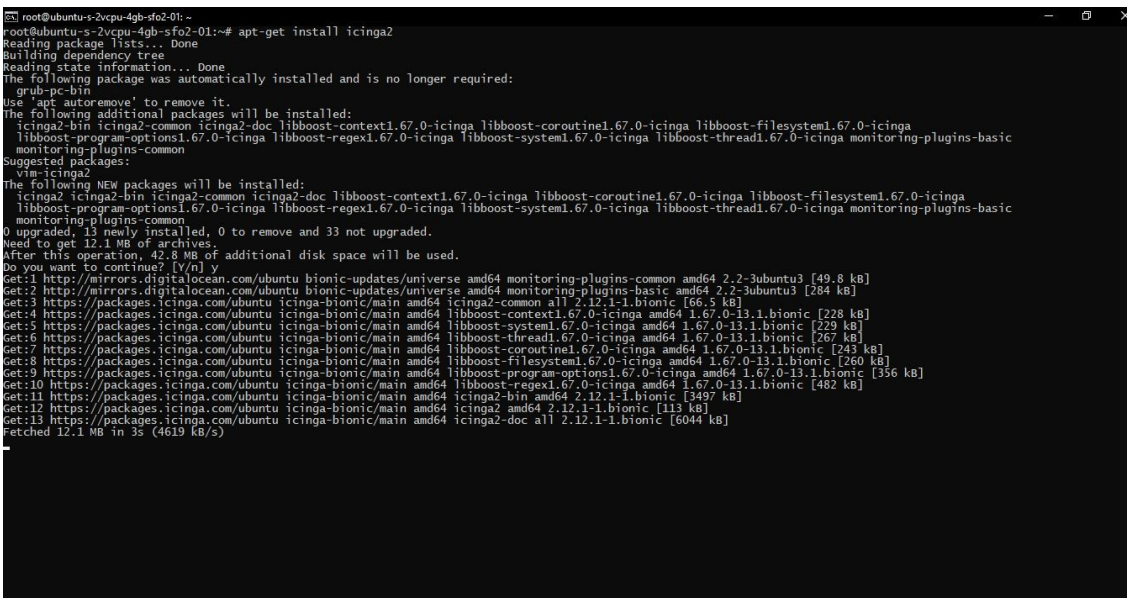


```
root@ubuntu-s-2vcpu-4gb-sfo2-01:~# apt-get update
get:1 http://mirrors.digitalocean.com/ubuntu bionic InRelease [242 kB]
get:2 http://mirrors.digitalocean.com/ubuntu bionic-updates InRelease [88.7 kB]
get:3 http://mirrors.digitalocean.com/ubuntu bionic-backports InRelease [74.6 kB]
get:4 http://security.ubuntu.com/ubuntu bionic-security InRelease [88.7 kB]
Hit:5 http://ppa.launchpad.net/certbot/certbot/ubuntu bionic InRelease
get:6 http://mirrors.digitalocean.com/ubuntu bionic-updates/main amd64 Packages [1761 kB]
get:7 http://mirrors.digitalocean.com/ubuntu bionic-updates/universe amd64 Packages [1692 kB]
Fetched 3948 kB in 1s (4317 kB/s)
Reading package lists... Done
root@ubuntu-s-2vcpu-4gb-sfo2-01:~# apt-get -y install apt-transport-https wget gnupg
Reading package lists... Done
Building dependency tree
Reading state information... Done
gnupg is already the newest version (2.2.4-1ubuntu1.3).
gnupg set to manually installed.
wget is already the newest version (1.19.4-1ubuntu2.2).
wget set to manually installed.
The following package was automatically installed and is no longer required:
  grub-pc-bin
Use 'apt autoremove' to remove it.
The following NEW packages will be installed:
  apt-transport-https
0 upgraded, 1 newly installed, 0 to remove and 33 not upgraded.
Need to get 1692 B of archives.
After this operation, 153 kB of additional disk space will be used.
Get:1 http://mirrors.digitalocean.com/ubuntu bionic-updates/universe amd64 apt-transport-https all 1.6.12ubuntu0.1 [1692 B]
Fetched 1692 B in 0s (54.2 kB/s)
Selecting previously unselected package apt-transport-https.
(Reading database ... 122463 files and directories currently installed.)
Preparing to unpack .../apt-transport-https_1.6.12ubuntu0.1_all.deb ...
Unpacking apt-transport-https (1.6.12ubuntu0.1) ...
Setting up apt-transport-https (1.6.12ubuntu0.1) ...
root@ubuntu-s-2vcpu-4gb-sfo2-01:~#
```

Figura 5 Actualizando nuevamente los paquetes del S.O.

Después de haber instalado y actualizando los paquetes necesarios, procedemos a instalar Icinga2 con el comando (Figura 6):

```
apt-get install icinga2
```



```
root@ubuntu-s-2vcpu-4gb-sfo2-01:~# apt-get install icinga2
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following package was automatically installed and is no longer required:
  grub-pc-bin
Use 'apt autoremove' to remove it.
The following additional packages will be installed:
  icinga2-bin icinga2-common icinga2-doc libboost-context1.67.0-icinga libboost-coroutine1.67.0-icinga libboost-filesystem1.67.0-icinga
  libboost-program-options1.67.0-icinga libboost-regex1.67.0-icinga libboost-system1.67.0-icinga libboost-thread1.67.0-icinga monitoring-plugins-basic
  monitoring-plugins-common
Suggested packages:
  vim-icinga2
The following NEW packages will be installed:
  icinga2 icinga2-bin icinga2-common icinga2-doc libboost-context1.67.0-icinga libboost-coroutine1.67.0-icinga libboost-filesystem1.67.0-icinga
  libboost-program-options1.67.0-icinga libboost-regex1.67.0-icinga libboost-system1.67.0-icinga libboost-thread1.67.0-icinga monitoring-plugins-basic
  monitoring-plugins-common
0 upgraded, 13 newly installed, 0 to remove and 33 not upgraded.
Need to get 12.1 MB of archives.
After this operation, 42.8 MB of additional disk space will be used.
Do you want to continue? [Y/n] y
get:1 http://mirrors.digitalocean.com/ubuntu bionic-updates/universe amd64 monitoring-plugins-common amd64 2.2-3ubuntu3 [49.8 kB]
get:2 http://mirrors.digitalocean.com/ubuntu bionic-updates/universe amd64 monitoring-plugins-basic amd64 2.2-3ubuntu3 [284 kB]
get:3 https://packages.icinga.com/ubuntu icinga-bionic/main amd64 icinga2-common all 2.12.1-1.bionic [66.5 kB]
get:4 https://packages.icinga.com/ubuntu icinga-bionic/main amd64 libboost-context1.67.0-icinga amd64 1.67.0-13.1.bionic [228 kB]
get:5 https://packages.icinga.com/ubuntu icinga-bionic/main amd64 libboost-system1.67.0-icinga amd64 1.67.0-13.1.bionic [229 kB]
get:6 https://packages.icinga.com/ubuntu icinga-bionic/main amd64 libboost-thread1.67.0-icinga amd64 1.67.0-13.1.bionic [267 kB]
get:7 https://packages.icinga.com/ubuntu icinga-bionic/main amd64 libboost-coroutine1.67.0-icinga amd64 1.67.0-13.1.bionic [243 kB]
get:8 https://packages.icinga.com/ubuntu icinga-bionic/main amd64 libboost-filesystem1.67.0-icinga amd64 1.67.0-13.1.bionic [260 kB]
get:9 https://packages.icinga.com/ubuntu icinga-bionic/main amd64 libboost-program-options1.67.0-icinga amd64 1.67.0-13.1.bionic [356 kB]
get:10 https://packages.icinga.com/ubuntu icinga-bionic/main amd64 libboost-regex1.67.0-icinga amd64 1.67.0-13.1.bionic [482 kB]
get:11 https://packages.icinga.com/ubuntu icinga-bionic/main amd64 icinga2-bin amd64 2.12.1-1.bionic [3497 kB]
get:12 https://packages.icinga.com/ubuntu icinga-bionic/main amd64 icinga2 amd64 2.12.1-1.bionic [113 kB]
get:13 https://packages.icinga.com/ubuntu icinga-bionic/main amd64 icinga2-doc all 2.12.1-1.bionic [6044 kB]
Fetched 12.1 MB in 3s (4619 kB/s)
```

Figura 6 Instalando Icinga2.

En la Figura 7 se muestra el proceso de instalación de Icinga y en la Figura 8 se muestra la finalización de ella.

```
root@ubuntu-s-2vcpu-4gb-sfo2-01:~# apt-get install monitoring-plugins
Processing triggers for ureadahead (0.100.0-21) ...
Processing triggers for libc-bin (2.27-3ubuntu2) ...
root@ubuntu-s-2vcpu-4gb-sfo2-01:~# apt-get install monitoring-plugins
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following package was automatically installed and is no longer required:
  grub-pc-bin
Use 'apt autoremove' to remove it.
The following additional packages will be installed:
  libarchive13 libavahi-client3 libavahi-common-data libavahi-common3 libcups2 libdbi1 libdjvips4 libltdb1 libmysqldclient20 libnet-snmp-perl libpopt5 libpython-stdlib
  libpython2.7 libpython2.7-minimal libpython2.7-stdlib libsensors4 libsmbclient libsmbclient-base libsmb30 libtalloc2 libtdb1 libtevent0 libtirpc1 libwbclient0
  monitoring-plugins-standard mysql-common python python-crypto python-ldb python-minimal python-samba python-talloc python-tdb python2.7 python2.7-minimal rpcbind
  samba-common samba-common-bin samba-libs smbclient snmp
Suggested packages:
  lrzps cups-common libcrypt-des-perl libdigest-hmac-perl libio-socket-inet6-perl lm-sensors snmp-mibs-downloader nagios-plugins-contrib fping postfix | sendmail-bin
  | exim4-daemon-heavy | exim4-daemon-light qstat python-doc python-tk python-crypto-doc python-gpgme python2.7-doc binutils binfmt-support heimdal-clients
  nfs-utils
The following NEW packages will be installed:
  libarchive13 libavahi-client3 libavahi-common-data libavahi-common3 libcups2 libdbi1 libdjvips4 libltdb1 libmysqldclient20 libnet-snmp-perl libpopt5 libpython-stdlib
  libpython2.7 libpython2.7-minimal libpython2.7-stdlib libsensors4 libsmbclient libsmbclient-base libsmb30 libtalloc2 libtdb1 libtevent0 libtirpc1 libwbclient0
  monitoring-plugins-standard mysql-common python python-crypto python-ldb python-minimal python-samba python-talloc python-tdb python2.7 python2.7-minimal rpcbind
  samba-common samba-common-bin samba-libs smbclient snmp
0 upgraded, 42 newly installed, 0 to remove and 33 not upgraded.
Need to get 16.9 MB of archives.
After this operation, 77.4 MB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://mirrors.digitalocean.com/ubuntu bionic-updates/main amd64 libwbclient0 amd64 2:4.7.6+dfsg-ubuntu-0ubuntu2.21 [34.4 kB]
Get:2 http://mirrors.digitalocean.com/ubuntu bionic-updates/main amd64 libpython2.7-minimal amd64 2.7.17-1-18.04ubuntu1.2 [335 kB]
Get:3 http://mirrors.digitalocean.com/ubuntu bionic-updates/main amd64 python2.7-minimal amd64 2.7.17-1-18.04ubuntu1.2 [1290 kB]
Get:4 http://mirrors.digitalocean.com/ubuntu bionic/main amd64 python-minimal amd64 2.7.15-rcl-1 [28.1 kB]
Get:5 http://mirrors.digitalocean.com/ubuntu bionic-updates/main amd64 libpython2.7-stdlib amd64 2.7.17-1-18.04ubuntu1.2 [1916 kB]
Get:6 http://mirrors.digitalocean.com/ubuntu bionic-updates/main amd64 python2.7 amd64 2.7.17-1-18.04ubuntu1.2 [248 kB]
Get:7 http://mirrors.digitalocean.com/ubuntu bionic/main amd64 libpython-stdlib amd64 2.7.15-rcl-1 [7620 B]
Get:8 http://mirrors.digitalocean.com/ubuntu bionic/main amd64 python amd64 2.7.15-rcl-1 [140 kB]
Get:9 http://mirrors.digitalocean.com/ubuntu bionic-updates/main amd64 samba-common all 2:4.7.6+dfsg-ubuntu-0ubuntu2.21 [72.6 kB]
Get:10 http://mirrors.digitalocean.com/ubuntu bionic-updates/main amd64 libavahi-common-data amd64 0.7-3.1ubuntu2 [22.1 kB]
Get:11 http://mirrors.digitalocean.com/ubuntu bionic-updates/main amd64 libavahi-common3 amd64 0.7-3.1ubuntu2 [21.6 kB]
Get:12 http://mirrors.digitalocean.com/ubuntu bionic-updates/main amd64 libavahi-client3 amd64 0.7-3.1ubuntu2 [25.2 kB]
Get:13 http://mirrors.digitalocean.com/ubuntu bionic-updates/main amd64 libcups2 amd64 2.2.7-1ubuntu2.8 [211 kB]
Get:14 http://mirrors.digitalocean.com/ubuntu bionic/main amd64 libdjvips4 amd64 2.11-1 [29.3 kB]
Get:15 http://mirrors.digitalocean.com/ubuntu bionic/main amd64 libtalloc2 amd64 2.1.10-2ubuntu1 [36.0 kB]
Get:16 http://mirrors.digitalocean.com/ubuntu bionic/main amd64 libtdb1 amd64 1.3.15-2 [39.7 kB]
Get:17 http://mirrors.digitalocean.com/ubuntu bionic/main amd64 libtevent0 amd64 0.9.34-1 [27.7 kB]
Get:18 http://mirrors.digitalocean.com/ubuntu bionic-updates/main amd64 libdbi1 amd64 2.1.2-3.1ubuntu0.1 [114 kB]
Get:19 http://mirrors.digitalocean.com/ubuntu bionic-updates/main amd64 libpython2.7 amd64 2.7.17-1-18.04ubuntu1.2 [1054 kB]
Get:20 http://mirrors.digitalocean.com/ubuntu bionic/main amd64 python-talloc amd64 2.1.10-2ubuntu1 [9124 B]
Get:21 http://mirrors.digitalocean.com/ubuntu bionic-updates/main amd64 samba-libs amd64 2:4.7.6+dfsg-ubuntu-0ubuntu2.21 [5256 kB]
Get:22 http://mirrors.digitalocean.com/ubuntu bionic-updates/main amd64 libarchive13 amd64 3.4-2-3.1ubuntu0.6 [288 kB]
Get:23 http://mirrors.digitalocean.com/ubuntu bionic-updates/main amd64 libnet-snmp-perl amd64 5.9.2-2-1.0ubuntu0.6 [64.4 kB]
```

Figura 7 Proceso de instalación de Icinga2.

```
root@ubuntu-s-2vcpu-4gb-sfo2-01:~# apt-get install monitoring-plugins
Creating config file /etc/nagios-plugins/config/dhcp.cfg with new version
Creating config file /etc/nagios-plugins/config/disk.cfg with new version
Creating config file /etc/nagios-plugins/config/dummy.cfg with new version
Creating config file /etc/nagios-plugins/config/ftp.cfg with new version
Creating config file /etc/nagios-plugins/config/http.cfg with new version
Creating config file /etc/nagios-plugins/config/load.cfg with new version
Creating config file /etc/nagios-plugins/config/mail.cfg with new version
Creating config file /etc/nagios-plugins/config/news.cfg with new version
Creating config file /etc/nagios-plugins/config/netcfg.cfg with new version
Creating config file /etc/nagios-plugins/config/ping.cfg with new version
Creating config file /etc/nagios-plugins/config/procs.cfg with new version
Creating config file /etc/nagios-plugins/config/real.cfg with new version
Creating config file /etc/nagios-plugins/config/ssh.cfg with new version
Creating config file /etc/nagios-plugins/config/tcp_udp.cfg with new version
Creating config file /etc/nagios-plugins/config/teinet.cfg with new version
Creating config file /etc/nagios-plugins/config/users.cfg with new version
Setcap for check_icmp and check_dhcp worked!
Setting up libboost-program-options1.67.0-icinga:amd64 (1.67.0-13.1.bionic) ...
Setting up libboost-regex1.67.0-icinga:amd64 (1.67.0-13.1.bionic) ...
Setting up libboost-system1.67.0-icinga:amd64 (1.67.0-13.1.bionic) ...
Setting up icinga2-doc (2.12.1-1.bionic) ...
Setting up libboost-filesystem1.67.0-icinga:amd64 (1.67.0-13.1.bionic) ...
Setting up libboost-thread1.67.0-icinga:amd64 (1.67.0-13.1.bionic) ...
Setting up libboost-coroutine1.67.0-icinga:amd64 (1.67.0-13.1.bionic) ...
Setting up icinga2-bin (2.12.1-1.bionic) ...
enabling default icinga2 features
Enabling feature checker. Make sure to restart Icinga 2 for these changes to take effect.
Enabling feature notification. Make sure to restart Icinga 2 for these changes to take effect.
Enabling feature mainlog. Make sure to restart Icinga 2 for these changes to take effect.
Setting up icinga2 (2.12.1-1.bionic) ...
Processing triggers for systemd (237-3ubuntu10.42) ...
Processing triggers for man-db (2.8.3-2ubuntu0.1) ...
Processing triggers for ureadahead (0.100.0-21) ...
```

Figura 8 Fin de la instalación de Icinga2.

3.2 Configuración de la base de datos

Para iniciar con la configuración de la base de datos se tiene que instalar el módulo que se encargará de importar las tablas de la base de datos, para ello tenemos que ejecutar el comando (Figura 9):

```
apt-get install icinga2-ido-mysql
```

```
root@ubuntu-s-2vcpu-4gb-sfo2-01:~# apt-get install icinga2-ido-mysql
```

Figura 9 Instalación de icinga2-ido-mysql.

Después de hacer la instalación de `icinga2-ido-mysql` se hace la creación de las 2 bases de datos que contendrán la información de la interfaz gráfica `icingaweb2` e `Icinga2` (Figura 10).

```
mysql> CREATE DATABASE icingaweb;
Query OK, 1 row affected (0.00 sec)

mysql> GRANT ALL PRIVILEGES ON icingaweb.* TO 'icingauser'@'localhost' IDENTIFIED BY 'VYf<#n3Z3y';
Query OK, 0 rows affected, 1 warning (0.00 sec)

mysql> CREATE DATABASE icinga_ido;
Query OK, 1 row affected (0.00 sec)

mysql> GRANT ALL PRIVILEGES ON icinga_ido.* TO 'icingauser'@'localhost' IDENTIFIED BY 'VYf<#n3Z3y';
Query OK, 0 rows affected, 1 warning (0.00 sec)
```

Figura 10 Creando las bases de datos.

Por último, importamos las tablas a la base de datos con el comando (Figura 11):

```
mysql -u root -p icinga < /usr/share/icinga2-ido-
mysql/schema/mysql.sql
```

```
root@ubuntu-s-2vcpu-4gb-sfo2-01:~# mysql -u root -p icinga < /usr/share/icinga2-ido-mysql/schema/mysql.sql
Enter password:
root@ubuntu-s-2vcpu-4gb-sfo2-01:~# icinga2 feature enable ido-mysql_
```

Figura 11 Importando las tablas a la base de datos desde `icinga2-ido-mysql`.

3.3 Instalación de la interfaz web Icinga2

Para la instalación de nuestra interfaz web se ejecuta el comando (Figura 12, Figura 13):

```
apt-get install icingaweb2 libapache2-mod-php icingacl
```

```
root@ubuntu-s-2vcpu-4gb-sfo2-01:~# apt-get install icingaweb2 libapache2-mod-php icingacl
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following package was automatically installed and is no longer required:
  grub-pc-bin
Use 'apt autoremove' to remove it.
The following additional packages will be installed:
  fonts-droid-fallback fonts-noto-mono ghostscript gsfonts icinga-110n icingaweb2-common icingaweb2-module-doc icingaweb2-module-monitoring imagemagick-6-common
  libapache2-mod-php7.2 libcupsfilters1 libcupsimage2 libfftw3-double3 libgomp1 libgs9 libgs9-common libijs-0.35 libjbig2dec0 liblcms2-2 liblqr-1-0 libltdl7
  libmagickcore-6.q16-3 libmagickwand-6.q16-3 libpaper-utils libpaper1 libsodium23 php php-common php-curl php-icinga php-imagick php-intl php-ldap php-mysql php-xml
  php7.2 php7.2-cli php7.2-common php7.2-curl php7.2-intl php7.2-json php7.2-ldap php7.2-mysql php7.2-openssl php7.2-readline php7.2-xml poppler-data ttf-dejavu-core
Suggested packages:
  fonts-noto-ghostscript-x php-pear libfftw3-bin libfftw3-dev liblcms2-utils libmagickcore-6.q16-3-extra poppler-utils fonts-japanese-mincho | fonts-ipafont-mincho
  fonts-japanese-gothic | fonts-ipafont-gothic fonts-arphic-ukai fonts-arphic-uming fonts-nanum
The following NEW packages will be installed:
  fonts-droid-fallback fonts-noto-mono ghostscript gsfonts icinga-110n icingacl icingaweb2 icingaweb2-common icingaweb2-module-doc icingaweb2-module-monitoring
  imagemagick-6-common libapache2-mod-php libapache2-mod-php7.2 libcupsfilters1 libcupsimage2 libfftw3-double3 libgomp1 libgs9 libgs9-common libijs-0.35 libjbig2dec0
  liblcms2-2 liblqr-1-0 libltdl7 libmagickcore-6.q16-3 libmagickwand-6.q16-3 libpaper-utils libpaper1 libsodium23 php php-common php-curl php-icinga php-imagick
  php-intl php-ldap php-mysql php-xml php7.2 php7.2-cli php7.2-common php7.2-curl php7.2-intl php7.2-json php7.2-ldap php7.2-mysql php7.2-openssl php7.2-readline
  php7.2-xml poppler-data ttf-dejavu-core
0 upgraded, 51 newly installed, 0 to remove and 33 not upgraded.
Need to get 27.2 MB of archives.
After this operation, 112 MB of additional disk space will be used.
Do you want to continue? [Y/n] y
```

Figura 12 Instalando la interfaz web y los complementos para `apache2`.

```
root@ubuntu-s2vcpu-4gb-sfo2-01: ~
Creating config file /etc/php/7.2/mods-available/pdo_mysql.ini with new version
Setting up php7.2-xml (7.2.24-0ubuntu0.18.04.7) ...
Creating config file /etc/php/7.2/mods-available/dom.ini with new version
Creating config file /etc/php/7.2/mods-available/simplexml.ini with new version
Creating config file /etc/php/7.2/mods-available/wddx.ini with new version
Creating config file /etc/php/7.2/mods-available/xml.ini with new version
Creating config file /etc/php/7.2/mods-available/xmlreader.ini with new version
Creating config file /etc/php/7.2/mods-available/xmlwriter.ini with new version
Creating config file /etc/php/7.2/mods-available/xsl.ini with new version
Setting up php-xml (1:7.2+60ubuntu1) ...
Setting up php7.2-cli (7.2.24-0ubuntu0.18.04.7) ...
update-alternatives: using /usr/bin/php7.2 to provide /usr/bin/php (php) in auto mode
update-alternatives: using /usr/bin/phar7.2 to provide /usr/bin/phar (phar) in auto mode
update-alternatives: using /usr/bin/phar.phar7.2 to provide /usr/bin/phar.phar (phar.phar) in auto mode
Creating config file /etc/php/7.2/cli/php.ini with new version
Setting up libapache2-mod-php7.2 (7.2.24-0ubuntu0.18.04.7) ...
Creating config file /etc/php/7.2/apache2/php.ini with new version
Module mpm_event disabled.
Enabling module mpm_prefork.
apache2_switch_mpm Switch to prefork
apache2_invoke: Enable module php7.2
Setting up php-mysql (1:7.2+60ubuntu1) ...
Setting up php-imagick (3.4.3-rc2-2ubuntu4.1) ...
Setting up libapache2-mod-php (1:7.2+60ubuntu1) ...
Setting up php7.2 (7.2.24-0ubuntu0.18.04.7) ...
Setting up php-icinga (2.8.2-1.bionic) ...
Setting up icingacli (2.8.2-1.bionic) ...
Setting up icingaweb2-common (2.8.2-1.bionic) ...
Adding system-group for icingaweb2
Adding user 'www-data' to group 'icingaweb2' ...
Adding user www-data to group icingaweb2
Done.
Setting up icingaweb2 (2.8.2-1.bionic) ...
apache2_invoke: Enable module rewrite
apache2_invoke: Enable configuration icingaweb2
Setting up icingaweb2-module-doc (2.8.2-1.bionic) ...
Setting up icingaweb2-module-monitoring (2.8.2-1.bionic) ...
Setting up php (1:7.2+60ubuntu1) ...
Processing triggers for man-db (2.8.3-2ubuntu0.1) ...
```

Figura 13 Proceso de instalación de la interfaz web y los complementos de apache2.

Después haberlo instalado podremos ir a nuestro navegador (Figura 14) y dirigirnos a la dirección de nuestro host agregando /icingaweb2, ejemplo: soporte.com/icingaweb2.

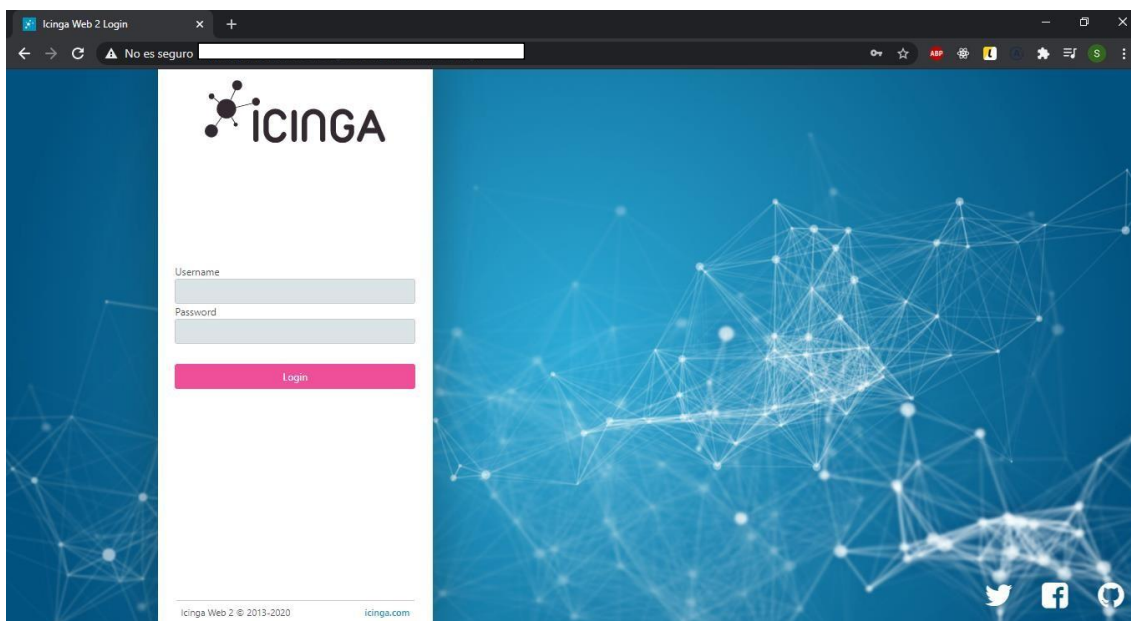


Figura 14 Interfaz web de Icinga.

Para iniciar la configuración de Icinga web tenemos que ir a la dirección /icingaweb2/setup donde podremos empezar a configurar nuestra interfaz con las bases de datos como se muestra en la Figura 15.

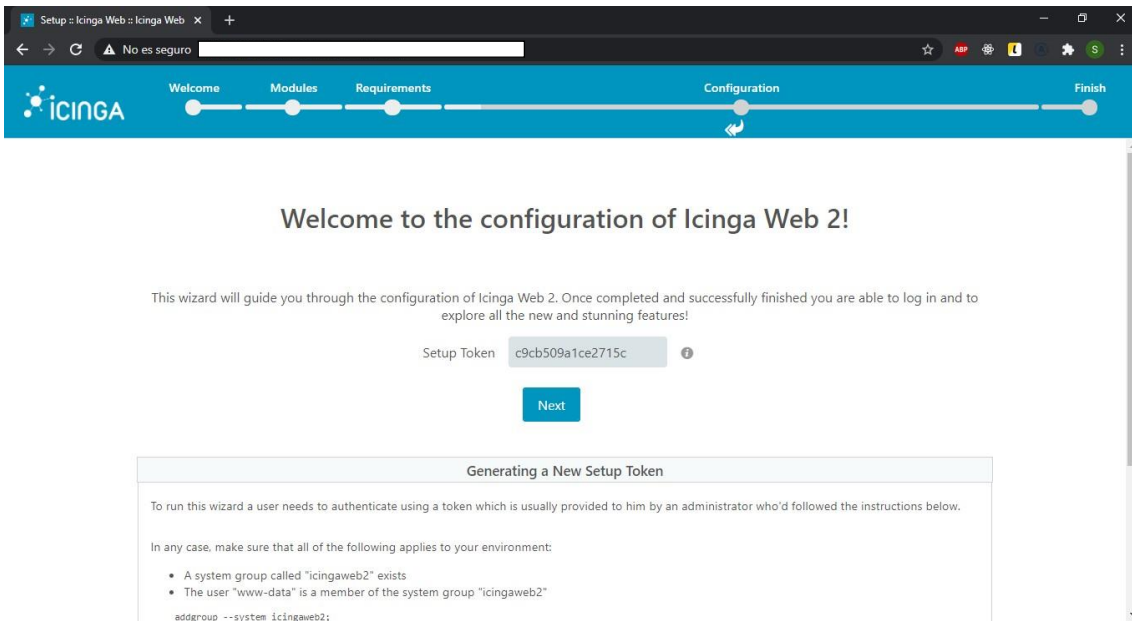


Figura 15 Interfaz de configuración de Icinga web.

Para generar un token para nuestra interfaz tenemos que ir a nuestra terminal y ejecutar el comando:

```
icingacli setup token create;
```

```
root@ubuntu-s-2vcpu-4gb-sfo2-01: /etc/icinga2
root@ubuntu-s-2vcpu-4gb-sfo2-01:/etc/icinga2# icingacli setup token create;
The newly generated setup token is: 46392b6c8334349e
root@ubuntu-s-2vcpu-4gb-sfo2-01:/etc/icinga2#
```

Figura 16 Creando token para la interfaz web.

A continuación, nos muestra los módulos que podemos instalar, “Doc” nos trae toda la documentación de Icinga para configuración, “Monitoring” es el módulo para crear el monitoreo, “Translation” nos permite cambiar el idioma de Icinga (Figura 17).

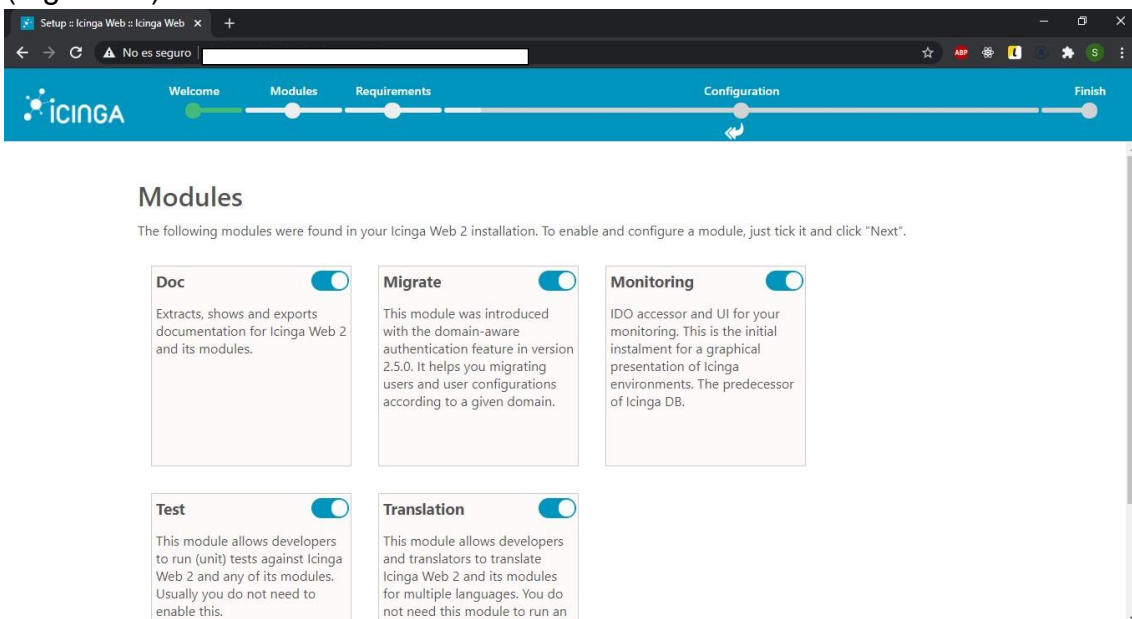


Figura 17 Módulos de Icinga.

Icinga nos mostrará los módulos de PHP que están instalados en el servidor en el que está montado Icinga (Figura 18).

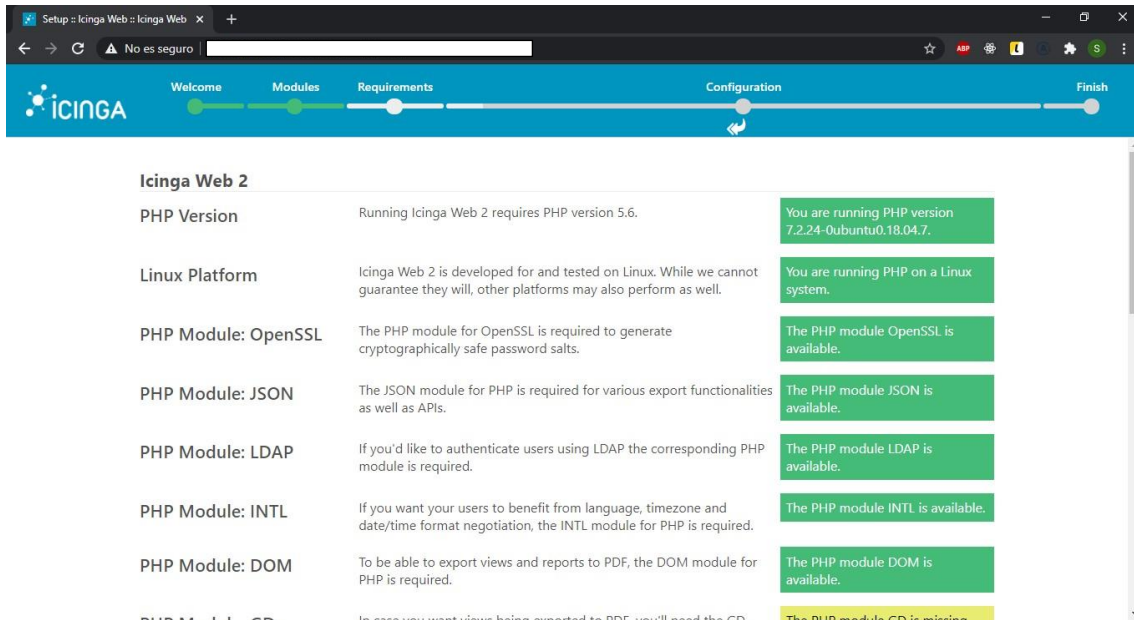


Figura 18 Módulos de PHP en el servidor de Icinga.

Asignaremos el método por el cual se hará la autenticación, se seleccionará por base de datos (Figura 19).

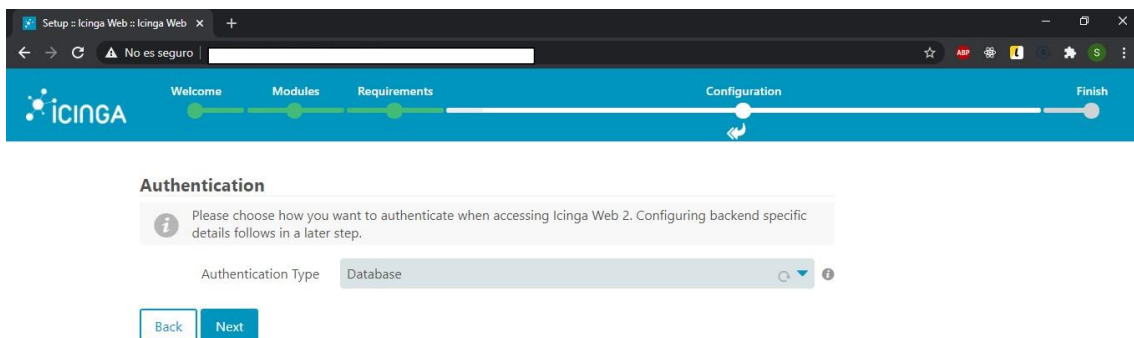


Figura 19 Método de autenticación.

Vinculamos la base de datos con Icinga a través de la interfaz web (Figura 20).

The screenshot shows the Icinga Web 2 setup wizard in the 'Configuration' step. The progress bar at the top indicates the current step. The 'Database Resource' section contains the following information:

- Resource Name: icingaweb_db
- Database Type: MySQL
- Host: localhost
- Port: (empty)
- Database Name: icinga
- Username: root
- Password: (masked with dots)

A message above the form states: 'The configuration has been successfully validated.'

Figura 20 Vinculando la base de datos a Icinga.

Asignación del usuario y la contraseña para el inicio de sesión en Icingaweb2 (Figura 21).

The screenshot shows the Icinga Web 2 setup wizard in the 'Configuration' step. The 'Administration' section contains the following information:

- Username: devecon
- Password: (masked with dots)
- Repeat password: (masked with dots)

Buttons for 'Back' and 'Next' are visible. A note at the bottom states: '* Required field'.

Figura 21 Usuario y contraseña de Icinga.

Para finalizar la configuración de Icingaweb2 se hace la configuración de la protección de las variables (Figura 22).

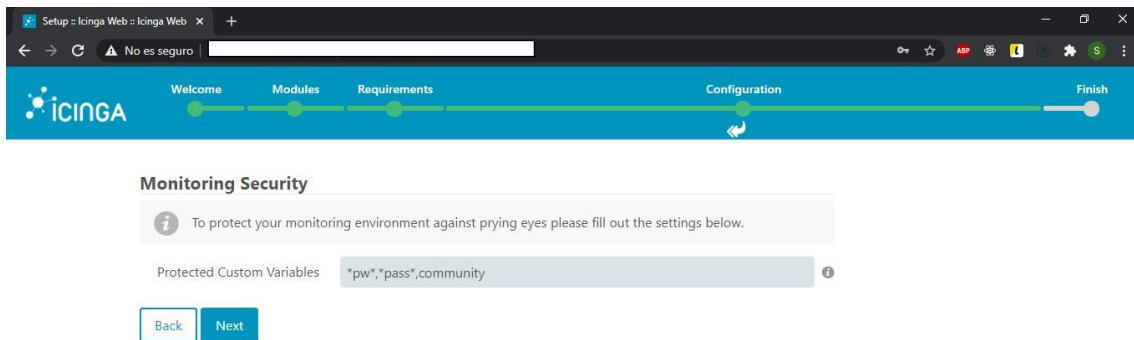


Figura 22 Protección de las variables.

3.4 Agregando los servidores al monitor

Para agregar un host al monitor solo tenemos que editar el archivo "hosts.conf" que se encuentra en "/etc/icinga2/conf.d/" para usamos el comando "nano /etc/icinga2/conf.d/ hosts.conf" (Figura 23).

```
root@ubuntu-s-2vcpu-4gb-sfo2-01: /etc/icinga2
root@ubuntu-s-2vcpu-4gb-sfo2-01:/etc/icinga2# sudo nano /etc/icinga2/conf.d/hosts.conf
```

Figura 23 Comando para la edición del archivo "hosts.conf"

Una vez abierto el archivo agregamos el nombre de nuestro host y la dirección de él (Figura 24).

```
root@ubuntu-s-2vcpu-4gb-sfo2-01: ~
GNU nano 2.9.3 /etc/icinga2/conf.d/hosts.conf

/* No parameters. */
}
vars.disks["disk /"] = {
  disk_partitions = "/"
}

/* Define notification mail attributes for notification apply rules in `notifications.conf`. */
vars.notification["mail"] = {
  /* The UserGroup `icingaadmins` is defined in `users.conf`. */
  groups = [ "icingaadmins" ]
}
}

object Host "Soporte" {
  import "generic-host"
  address = "192.168.1.1"
  address6 = "::1"
  vars.os = "Linux"
  check_command = "hostalive"
}
```

Figura 24 Agregando el host al servidor.

Después de agregar el host se crea un “apply Service” para poder realizar las pruebas necesarias para el host (Figura 25).

```
GNU nano 4.8 hosts.conf
/* Set custom variable os for hostgroup assignment in 'groups.conf'. */
vars.os = "Linux"

/* Define http vhost attributes for service apply rules in 'services.conf'. */
vars.http_vhosts["http"] = {
  http_uri = "/"
}

/* Uncomment if you've successfully installed Icinga Web 2. */
//vars.http_vhosts["Icinga Web 2"] = {
//  http_uri = "/icingaweb2"
//}

/* Define disks and attributes for service apply rules in 'services.conf'. */
vars.disks["disk"] = {
  /* No parameters. */
}
vars.disks["disk /"] = {
  disk_partitions = "/"
}

/* Define notification mail attributes for notification apply rules in 'notifications.conf'. */
vars.notification["mail"] = {
  /* The UserGroup 'icingaadmins' is defined in 'users.conf'. */
  groups = [ "icingaadmins" ]
}

object Host "ServidorSoporte" {
  address = [ "157.245.235.67" ]
  check_command = "hostalive"
  vars.mysql = true
}

object Host "SERVER-01" {
  import "generic-service"
  address = [ "157.245.235.67" ]
  check_command = "hostalive"
}

apply Service "mysql-test" {
  import "generic-service"
  display_name = "MySQL test"
  check_command = "mysql"
  vars.mysql_database = "icingaweb"
  vars.mysql_username = "prueba"
  vars.mysql_password = "shok3r"
  assign where host.vars.mysql == true
}
```

Figura 25 Configurando un servicio

Con ello podemos ver el resultado en nuestra interfaz gráfica (Figura 26).

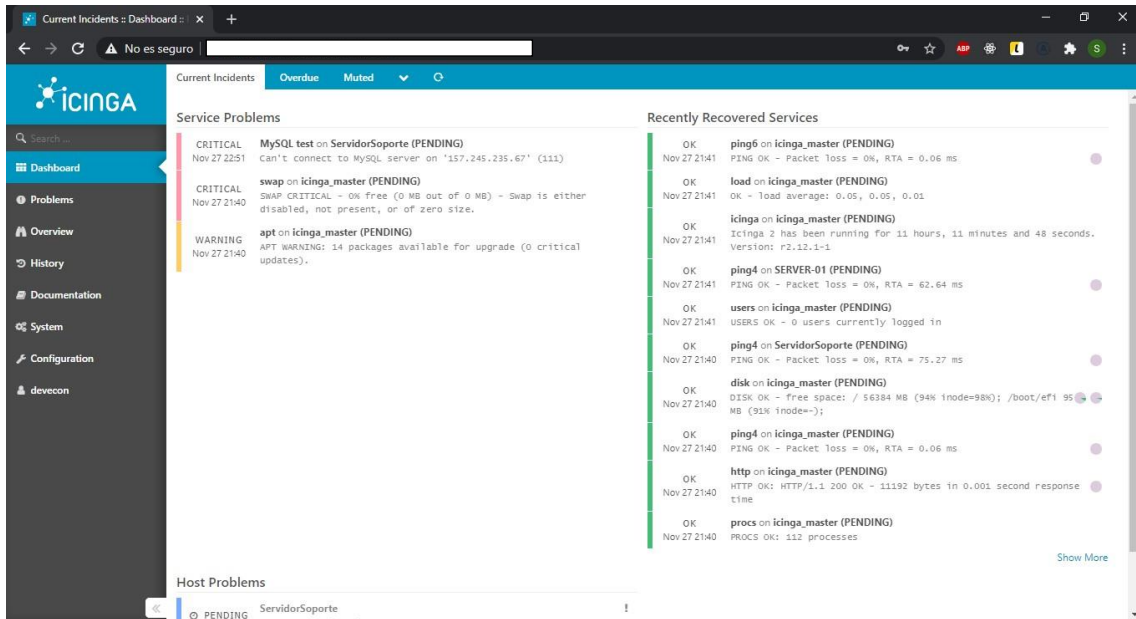


Figura 26 Interfaz gráfica con servidores monitoreados

4. Conclusiones y recomendaciones

A lo largo del desarrollo de este proyecto se presentaron diversas dificultades con la implementación del monitor por parte del complemento de Icinga para la base de datos, ya que cuando se instala “icinga-ido-mysql”, crea su propio usuario y contraseña pero ninguna de las que se ingresaban las aceptaba, después de tratar varias veces se logró, pero cuando se hace la vinculación de Icingaweb2 con la base de datos Icinga nos dice que no se está ejecutando ya que no se logra conectar con el usuario que crea “icinga-ido-mysql”. Por lo tanto, se tuvo que crear un nuevo usuario para poder vincularlo con la base de datos y poder darle los privilegios necesarios al usuario.

Cuando se hizo la creación del servidor maestro y los esclavos, se tuvo que tener mucho cuidado con el token que genera el servidor maestro ya que se manejan muchas contraseñas y usuarios a lo largo de todo el proyecto. Si a la hora de configurar no es el token que generò desde el servidor maestro, la configuración no habrá servido y se tendría que hacer toda la configuración nuevamente.

En general no se tuvo ningún problema grave con la implementación y configuración del monitoreo en Icinga ya que se siguieron los pasos correctamente y se cumplió el objetivo general que se propuso con este proyecto.

Haber realizado un proyecto con servidores se me hizo muy complicado ya que no estoy familiarizado con los sistemas operativos Linux, tuve que empezar desde cero a conocerlos, como navegar en sus carpetas, como crear archivos y como editarlos, pero una vez conociendo los conceptos básicos pues si se logró sacar el proyecto adelante.

Se debe destacar que este fue mi primer proyecto que no se trató de programación y fue difícil, pero una muy buena experiencia ya que es trabajar con los servidores donde se almacenan las páginas web y donde se corren las bases de datos de esas páginas, aspecto fundamental en toda organización para su operación y desarrollo.

5. Referencias bibliográficas

- [1] Porto, J. P. P. (2008). *Definición de internet* — Definición de Internet. <https://definicion.de/internet/>
- [2] Topología y Arquitectura de red - 605 Redes de computadoras. (s. f.). Redes de computadoras. Recuperado 17 de noviembre de 2020, de <https://sites.google.com/site/605bredesdecomputadoras/home/6>
- [3] Porto, J. P. P. (2016). *Definición de cliente servidor* — Definición de cliente servidor. <https://definicion.de/cliente-servidor/>
- [4] Láinez, D. M. D. M. N. (2017, 15 noviembre). ¿Qué son los servicios en la nube? Tipos y ejemplos. Clinic Cloud. <https://clinic-cloud.com/blog/servicios-en-la-nube-tipos-ejemplos/#:%7E:text=Los%20servicios%20en%20la%20nube%20son%20servicios%20que%20se%20utilizan,se%20instalaban%20en%20el%20ordenador.>
- [5] Lemus, I. (2019, 6 julio). Icinga software de herramientas de monitoreo para servidores. Conocimiento Libre. <https://conocimientolibre.mx/icinga/>
- [6] Converse, T., & Park, J. (2009). *La biblia de PHP 6 y MySQL / PHP 6 and MySQL 6 Bible* (Tra ed.). Anaya Multimedia.
- [7] Bellido, S. B. S. (2020). Base de datos NoSQL: MongoDB (N.o 5). http://bibing.us.es/proyectos/abreproy/12037/fichero/PFC_Sergio_Bellido_Sanchez%252FTema5_mongodb.pdf
- [8] Kabir, M. J. (2005). *Servidor apache / Apache Server: 2*. Anaya Multimedia.
- [9] *Installation - Icinga 2*. (s. f.). Icinga 2. Recuperado 28 de octubre de 2020, de <https://icinga.com/docs/icinga2/latest/doc/02-installation/>



Universidad Politécnica de Puebla
Ingeniería en Informática

Rubén Ely Trujillo Pérez
Rigoberto Martínez López
Rebeca Rodríguez Huesca

Este documento se distribuye para los términos de la
Licencia 2.5 Creative Commons (CC-BC-NC-ND 2.5 MX)