UNIVERSIDAD POLITÉCNICA SALESIANA
SEDE CUENCA

CARRERA DE INGENIERÍA DE SISTEMAS

Trabajo de titulación previo
a la obtención del título de
Ingeniera de Sistemas

PROYECTO TÉCNICO CON ENFOQUE SOCIAL:

"PROTOTIPO DE FILTRADO ANTI SPAM A TRAVÉS DE UNA NUBE PRIVADA"

AUTORAS:

DORIS MARIBEL MARCA GUARACA

PAMELA MAYTE VILLARROEL TIGRERO

TUTOR:

DR. PABLO LEONIDAS GALLEGOS SEGOVIA

CUENCA - ECUADOR

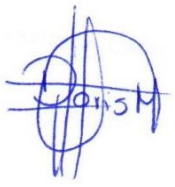
2020

CESIÓN DE DERECHOS DE AUTOR

Nosotras, Doris Maribel Marca Guaraca con documento de identificación N° 0105137574 y Pamela Mayte Villarroel Tigrero con documento de identificación N° 0707075156, manifestamos nuestra voluntad y cedemos a la Universidad Politécnica Salesiana, la titularidad sobre los derechos patrimoniales en virtud de que somos autoras del trabajo de titulación: **"PROTOTIPO DE FILTRADO ANTI SPAM A TRAVÉS DE UNA NUBE PRIVADA"**, mismo que ha sido desarrollado para optar por el título de: *Ingeniera de Sistemas*, en la Universidad Politécnica Salesiana, quedando la Universidad facultada para ejercer plenamente los derechos cedidos anteriormente.

En aplicación a lo determinado en la Ley de Propiedad Intelectual, en nuestra condición de autoras, nos reservamos los derechos morales de la obra antes citada. En concordancia, suscribimos este documento en el momento que hacemos entrega del trabajo final en formato digital a la Biblioteca de la Universidad Politécnica Salesiana.

Cuenca, julio del 2020



Doris Maribel Marca Guaraca
C.I. 0105137574

Pamela Mayte Villarroel Tigrero
C.I. 0707075156

CERTIFICACIÓN

Yo, declaro que bajo mi tutoría fue desarrollado el trabajo de titulación: **"PROTOTIPO DE FILTRADO ANTI SPAM A TRAVÉS DE UNA NUBE PRIVADA"**, realizado por Doris Maribel Marca Guaraca y Pamela Mayte Villarroel Tigrero, obteniendo el *Proyecto Técnico con enfoque social*, que cumple con todos los requisitos estipulados por la Universidad Politécnica Salesiana.

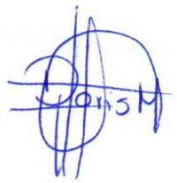
Cuenca, julio del 2020

Dr. Pablo Leonidas Gallegos Segovia
C.I. 0102593589

DECLARATORIA DE RESPONSABILIDAD

Nosotras, Doris Maribel Marca Guaraca con documento de identificación N° 0105137574 y Pamela Mayte Villarroel Tigrero con documento de identificación N° 0707075156, autoras del trabajo de titulación: **"PROTOTIPO DE FILTRADO ANTI SPAM A TRAVÉS DE UNA NUBE PRIVADA"**, certificamos que el total contenido de este *Proyecto Técnico con enfoque social*, es de nuestra exclusiva responsabilidad y autoría.

Cuenca, julio del 2020



Doris Maribel Marca Guaraca
C.I. 0105137574

Pamela Mayte Villarroel Tigrero
C.I. 0707075156

AGRADECIMIENTOS

Quiero expresar mi agradecimiento de manera especial al Dr. Pablo Gallegos por confiar en mí y permitirme realizar mi proyecto de titulación dentro del Grupo de investigación GIHP4C, al ingeniero Byron Carrión por impartir su conocimiento y permitir avanzar con este proyecto de titulación. Quiero expresar mi sincero agradecimiento a una persona muy especial Fabián Salazar quien ha sido un apoyo fundamental dentro de este largo caminar. Finalmente quiero expresar mi más sincero agradecimiento a mis padres y mi hermano quienes han sido un gran apoyo y sobre todo un gran ejemplo para seguir durante este largo camino para lograr este gran objetivo, gracias a sus valores, principios y consejos impartidos han hecho de este sueño una realidad.

Doris Maribel Marca Guaraca.

Agradezco primeramente a Dios por guiarme en este camino, al Dr. Pablo Gallegos por brindarme esta oportunidad de realizar este trabajo de titulación en conjunto con el Grupo de investigación GIHP4C y al ingeniero Byron Carrión por compartir sus conocimientos y su buena predisposición al brindarnos pautas para el desarrollo de este proyecto de titulación. El más profundo agradecimiento a mis padres Luisa y Manuel por ser mi motor y mi motivación para seguir adelante en este camino, por darme su apoyo incondicional y sus grandes consejos que me han sabido guiar en este camino. Finalmente quiero agradecer a una persona muy especial e importarme en mi vida Jonnathan Andrade por acompañarme y ser un pilar fundamente para el logro de este trabajo de titulación.

Pamela Mayte Villarroel Tigrero.

DEDICATORIAS

A mis padres
Por todo el esfuerzo que han realizado para que alcance este gran objetivo, hoy cosechan frutos de todo el esfuerzo realizado para formarme como persona y como profesional, hoy con la frente en alto me permito decirles que me siento orgullosa de María Mercedes Guaraca Guaraca y Ángel Rodrigo Marca Plasencia ya que gracias a ustedes hoy concluyo esta meta, gracias infinitas de corazón.
A mi hermano y amigo
Por permitirme creer en mi persona, por guiarme en cada paso de este duro camino por darme ánimos y consejos para llegar a lograr esta meta tan añorada hoy me permito decirles a Christian Rafael Marca Guaraca y a Fabián Salazar que me siento orgullosa de las personas que son gracias infinitas por brindarme su apoyo incondicional.

Doris Maribel Marca Guaraca.

A mis padres
por toda la confianza que han tenido siempre en mí, por su apoyo incondicional y su gran amor que han permito que logre este sueño. Sin duda alguna nada de esto fuera posible sin ustedes Luisa Aurora Tigrero León y Manuel Hernan Villarroel Valverde, con mucho amor para ustedes.
A mi hermano Ronald Villarroel y mis sobrinos Darwin, Santiago, mi cuñada Carolina por su cariño incondicional.
A mi mejor amigo y amor Jonnathan Andrade por tu infinita ayuda, por creer en mí, por ser mi luz en todo este camino.
A mis mejores amigos Cinthia e Italo por su gran apoyo y buenos consejos.
A ti mi amado Zeus.

Pamela Mayte Villarroel Tigrero.

ÍNDICE GENERAL

Tabla de contenido

ÍNDICE DE FIGURAS

ÍNDICE DE TABLAS

GLOSARIO DE TÉRMINOS

Spam: Correo no deseado o correo basura.

Ham: Correo electrónico que no es spam.

Spammers: Personas o robots que generan spam.

Phishing: Es un método de suplantación de identidad, son utilizados por delincuentes cibernéticos para estafar y obtener información.

Phishers: Se filtran como personas o empresas de confianza que engañan para obtener información.

Virtualización: comprende la abstracción de los recursos de una máquina física con el objetivo de simular varias máquinas "lógicas" o virtuales. Por supuesto este concepto se puede extender no sólo a máquinas físicas, y es aplicado en multitud de ámbitos tecnológicos: almacenamiento, networking, aplicaciones, etc [1].

Machine Learning: es parte de la inteligencia artificial la misma que permite tener un aprendizaje automatizado o aprendizaje de máquinas que es capaz de identificar diferentes patrones.

Algoritmo Bayesiano: Permite calcular eventos explícitos por cada hipótesis, también permite construir modelos comportamientos sumamente buenos ya que tiene una gran simplicidad.

IaaS: Infraestructura como servicio.

SNMP: Protocolo simple de administración de red.

Virus Informático: es como un virus de gripe, está diseñado para propagarse de un host a otro y tiene la habilidad de replicarse, estos no pueden reproducirse ni propagarse sin programar, por ejemplo, un archivo o un documento [2].

Cloud computing: Computación en la nube consiste en prestar servicios a través de internet [3].

Front-end: Es el motor de administración que ejecuta los servicios de OpenNebula.

XML-RPC: RCP es un protocolo de llamada a procedimiento remoto que usa XML para codificar los datos y HTTP como protocolo de transmisión de mensajes

SSH: intérprete de órdenes Seguro que permite la administración remota a los usuarios para que puedan controlar y modificar sus servidores, estando conectados a través de la red.

LDAP: Protocolo ligero de acceso a directorios.

NIST: Instituto Nacional de Estándares y Tecnología.

Servlets: Son módulos de java que sirven para extender las capacidades de los servidores web.

Jetty: Es un servidor HTTP cien por ciento basado en Java y un contenedor de Servlets escrito en Java [4].

MTA: Agente de Transferencia de Correos [5].

SSL: Capa de sockets seguros.

TLS: Seguridad de la capa de transporte [6].

HTTPS: Protocolo seguro de transferencia de hipertexto

Amavis: Filtro de contenido de código abierto para el correo electrónico.

Clamav: Es un software antivirus tipo opensource.

MIME: Multipurpose Internet Mail Extensions o extensiones multipropósito de correo de internet [7]

Lucene: Es una API de código abierto para recuperación de información.

Falsos positivos: Es un error que existe al definir un mensaje válido como spam.

ERP: Enterprise Resource Planning o Sistema de planificación de recursos empresariales.

RESUMEN

En la actualidad el spam, es más conocido como correos basura, estos se han convertido en un problema recurrente en la administración de los servidores de correo empresarial, generando continuas solicitudes de atención por parte los usuarios puesto que llenan sus bandejas de entrada y pérdidas de tiempo, esfuerzo y producción, así mismo las demandas de procesamiento del servidor, el ancho de banda y reducción del rendimiento del servicio. Mas aun cuando, estos correos basura son fuente de ataques de phising, troyanos, malware o de ransomware que están diseñados para el robo información de los usuarios o ataques a infraestructuras críticas.

Una de las estrategias usadas por las empresas es optar por aumentar los recursos de sus servidores con el fin de brindar un buen servicio a sus usuarios, lo que representa que los costos de la infraestructura sean elevados es por ello que no es una buena alternativa para brindar un mejor servicio a sus usuarios, puesto que en algún momento podrían a llegar a colapsar las colas del servidor.

Nuestro proyecto plantea una alternativa para evitar la recepción de correos masivos, y mitigar este inconveniente, dentro de este proyecto se realizó un prototipo de filtrado anti-spam usando la nube privada sobre la plataforma OpenNebula la que servirá como infraestructura en la cual se va a desarrollar nuestra propuesta. El diseño está basado en una nube privada en la que se genera una instancia en la misma que esta el servidor de correo electrónico (Zimbra) y en donde se encuentra configurado el filtro anti-spam, este sistema está basado en un algoritmo bayesiano el mismo que permite filtrar los correos spam y así evitar el ingreso a las bandejas de entrada de los usuarios. Esto se logrará con la ayuda de un script bash el mismo que realiza la función de procesar la cola de los correos antes de que lleguen a la bandeja de entrada, de tal forma que será un intermediario entre el servidor de correo y el filtro anti-spam que está desarrollado en Python con técnicas de machine learnig.

El filtro bayesiano está diseñado en Python con librerías de Machine Learnig:

- Sklearn es una librería abierta que tiene como objetivo brindar soluciones simples y eficientes permitiendo el aprendizaje automático del diccionario construido con archivos de tipo spam (correos basura) y ham (correos no spam), haciendo que sea accesible y reutilizable en diferentes contextos.
- Ntlk es otra librería que tiene la finalidad de la clasificación de texto, tokenización (divide cadenas de texto), derivación, análisis y razonamiento semántico, facilitando la descarga de diferentes corpus (textos que contiene diferentes caracteres, signos,

palabras, enlaces) la cual nos ayuda a tener un conjunto de texto relativamente grande el mismo que será utilizado como una lista de cadenas de texto (tokens) , permitiendo clasificar caracteres en blanco, saltos de línea, signos, etc; para de esa manera poder determinar los archivos de tipo spam y ham.

- La librería NaiveBayesClassifier nos ayuda obtener la tasa de error de acuerdo con el número del diccionario previamente creado.

Finalmente, el filtro bayesiano trabaja de la siguiente manera:

Analiza todos los datos entrenados del diccionario para así poder realizar la clasificación de los mensajes, con el fin de comprobar si son de tipo spam o ham. Para establecer la precisión del filtrado bayesiano realizamos diferentes pruebas enviando simultáneamente mensajes spam a la bandeja de entrada de los diferentes usuarios.

ABSTRACT

Nowadays, spam is better known as malicious bulk emails, these have become a recurrent problem in administration of business email servers, generating continuous requests for attention from users since they fill their inboxes and waste of time, effort and production, as well as the demands of server processing, bandwidth and reduced service performance. Even, these massive emails are the source of phishing attacks, trojans, malware or ransomware that are designed to steal user information or attack critical infrastructure. One of the strategies used by companies is to choose to increase the resources of their servers in order to provide a good service to their users, which means that infrastructure costs are high, which is why it is not a good alternative for provide a better service to its users, since at some point the queues of the server could collapse.

Our project proposes an alternative to avoid receiving mass emails, and mitigate this problem, a prototype of anti-spam filtering was carried out using the private cloud on the OpenNebula platform, which will serve as the infrastructure in which our proposal will be developed. The design is based on a private cloud on which an instance is generated in which an email server will be deployed (Zimbra) and in which the software for the anti-spam filter will be created, this system is based on an algorithm bayesian the same that allows you to filter spam emails and thus prevent entry to users inboxes. This will be achieved with the help of a bash script which performs the following functions, processing the queue of the emails before they reach the inbox, in such a way that it will be an intermediary between the mail server and the bayesian filter.

The bayesian filter is designed in python with machine learning libraries:

- Sklearn is an open library that aims to provide simple and efficient solutions allowing automatic learning of the dictionary built with files of type spam (junk emails) and ham (non-spam emails), making it accessible and reusable in different contexts.
- Ntlk is another library that has the purpose of text classification, tokenization (divides text strings), derivation, analysis and semantic reasoning, facilitating the download of different corpus (texts that contain different characters, signs, words, links) the which helps us to have a relatively large set of text which will be used as a list of text strings (tokens), allowing classifying blank characters, line breaks, signs, etc; in order to determine spam and ham files.
- The NaiveBayesClassifier library helps us obtain the error rate according to the number of the dictionary previously created.

Finally, the bayesian filter works as follows:

It analyzes all the trained data in the dictionary in order to classify the messages, in order to check if they are spam or ham. To establish the precision of Bayesian filtering, we carry out different tests simultaneously sending spam messages to the inbox of the different users.

INTRODUCCIÓN

El correo electrónico es una de las herramientas más importantes dentro de las empresas ya que permite la comunicación crítica entre esta y sus clientes, socios, proveedores, organismos de control, sin embargo, esta se encuentra expuesto en la internet y es muy fácil conseguir la cuenta de correo empresarial usando algunos programas de búsqueda o desde el navegador. Al ser visibles los servidores de correo despiertan la atención de personas o boots que se encargan de generar grandes cantidades de correo, cuya finalidad es la de vender diversidad de productos, estafas, ataques coordinados que comprometen las infraestructuras empresariales, bases de datos, o la reputación de los servicios de correo y web. Los correos masivos maliciosos generan congestión en las colas del servidor de correo, ocasionando un bajo rendimiento, siendo el mayor problema que uno de estos correos se filtre en la organización y puede obtener información confidencial de los usuarios o transacciones financieras que afecten la operación de la empresa, el peor escenario es el ransomware y rapto de las bases de datos.

Según los datos estadísticos de Kaspersky [8]Ecuador en el año 2019 representa un 15.64% de los países atacados por phishing que es más conocido como las estafas en línea. Debido a que "Los correos electrónicos son una de las formas de ingeniería social que utilizan los hackers para propagar virus informáticos" [9]. Por la falta de experiencia o descuido el usuario da acceso a los correos spam estos pueden dirigirlos a registrarse en una página web o descargar un archivo malicioso que lo haría víctima de ataques y esto podría provocar la pérdida de los documentos, e información crítica como: cuentas de bancos tarjetas de crédito, fotografías, etc. o al encriptamiento de los archivos del computador y que el atacante realice extorción por esta información. Como consecuencia, es necesario buscar mecanismos de protección para evitar o mitigar el correo malicioso. Nosotros realizamos un prototipo ante esta problemática aplicando un sistema en la que va a filtrar los correos en una instancia situada en la nube la misma que tendrá como misión clasificar las colas de correo de entrada en los que se procese y clasifique el correo y posteriormente sea enviados hasta los servidores destino de tal manera que solo lleguen los correos seguros.

JUSTIFICACION

Este proyecto busca desarrollar un prototipo de filtrado anti-spam de correo electrónico como servicio, con el fin brindar una posible alternativa al marco general que hace uso del servicio de correo, esto permitirá a los usuarios trabajar en un servicio más seguro ya que el servidor de correo electrónico es uno de los servicios que se ve afectado constantemente a diversos ataques los mismos que hacen que la infraestructura reduzca su rendimiento o aún peor que los usuarios pierdan información valiosa.

Los servidores de correo electrónico en su gran mayoría están alojados en la nube por lo que el prototipo de filtrado anti-spam está alojado en la plataforma de OpenNebula ya que es un software de código abierto el mismo que permite gestionar y organizar diferentes herramientas para así poder crear una nube privada o hibrida, es altamente escalable ya que no tiene una limitante de máquinas virtuales, se puede monitorizar los diferentes hosts, así como también las máquinas virtuales, al implementar esta alternativa se mitigara el ingreso de correos masivos a las bandejas de entrada, esto se hace posible con el desarrollo de un algoritmo bayesiano para el filtrado antispam de correos puesto que es uno de los más eficientes hoy en día ya que utiliza el teorema de bayes para comprobar la probabilidad de que un correo sea spam o no.

OBJETIVOS

OBJETIVO GENERAL

Desarrollar un prototipo de servicio de filtrado de correos antispam en la nube, el cual permitirá el paso de mensajes seguros para los usuarios finales.

OBJETIVOS ESPECIFICOS

- Elaborar estado del arte acerca de las metodologías que se utilizara para llevar a cabo el desarrollo del servicio de filtro antispam de correo electrónico.
- Establecer los requerimientos para el filtrado de correos spam, configuración del servidor de correo que permitan el filtrado antispam de correos en la plataforma OpenNebula.
- Implementar un prototipo de filtrado antispam de correo electrónico mediante metodologías de machine learnig para la plataforma OpenNebula.
- Realizar un protocolo de pruebas de contexto para establecer métricas de efectividad de la solución.

CAPÍTULO 1: ESTADO DEL ARTE Y FUNDAMENTACIÓN TEÓRICA

En esta sección se realizará la fundamentación teórica que se ha realizado para fundamentar el desarrollo de este proyecto.

1. Cloud Computing

Entes te capitulo, se explica que es Cloud Computing, su arquitectura, características y plataformas de código abierto.

¿Qué es Cloud Computing?

Cloud Computing o computación en la nube es una de las grandes evoluciones que se dan en el campo de la tecnología y la informática, la cual nos permite tener acceso desde cualquier lugar a la información que tenemos alojada en la misma, los recursos se ajustan de acuerdo a las necesidades en si son bajo pedido, ya cuando se necesite mayores recursos se los solicita y cuando ya no se los requiera utilizar se puede solicitar de igual manera los que sean convenientes de acuerdo a las necesidades de los consumidores, por eso la escalabilidad y la elasticidad es una de sus propiedades ya que permite el crecimiento sin tener inconveniente alguno con la infraestructura y se puede tener un acceso a un conjunto de recursos que se pueden configurar los cuales son: las redes, almacenamiento, servidores, aplicaciones y servicios con la ayuda de proveedor del servicio [10].

Cloud computing tiene tres modelos de servicio, así y modelos de implementación de los cuales se conocerán a continuación.

Según el NIST el modelo de computación de la nube destaca las cinco características esenciales de la misma proporcionando una línea base para discutir lo que es la computación en nube y un medio para comparar los servicios en la nube y sus estrategias de implementación [11].

Características.

El sistema de Cloud Computing compensa algunas características de interés las mismas que prometen futuras aplicaciones, así como servicios para tecnologías de la información. Según el NIST el modelo de computación de la nube destaca las cinco características

esenciales las cuales son: [11]

Autoservicio bajo demanda.

Un consumidor puede solicitar recursos de computación, tales como tiempo de servidor y almacenamiento en red, a medida que lo necesite, sin requerir interacción humana con el proveedor del servicio [11].

Dentro de los recursos que el consumidor requiera se tiene el tiempo del CUP, almacenamiento acceso a la red, aplicaciones web entre otros [11].

Amplio acceso a la Red.

Las capacidades están disponibles en la red y se acceden a través de mecanismos estándares que promueven el uso heterogéneo de plataformas de cliente ligeras o pesadas (por ejemplo, teléfonos móviles, tabletas, PDAs, computadoras portátiles y estaciones de trabajo) [11].

Agrupamiento de recursos.

Los recursos de computación del proveedor están agrupados (pooling) para servir a múltiples consumidores utilizando un modelo multi distribuido (multitenant), con diferentes recursos físicos y virtuales asignados y reasignados dinámicamente de acuerdo a la demanda del consumidor [11].

Rápida elasticidad.

Las funcionalidades se pueden proporcionar de manera rápida y elástica y, en algunos casos, automáticamente. Sus características de aprovisionamiento dan la sensación y pueden adquirirse en cualquier cantidad o momento [11].

Servicio medido.

Los sistemas en nube controlan y optimizan automáticamente el uso de recursos, potenciando la capacidad de medición en un nivel de abstracción apropiado al tipo de servicio (almacenamiento, procesamiento, ancho de banda y cuentas activas de usuario). El uso de recursos puede ser monitorizado, controlado e informado, proporcionando transparencia para el proveedor y para el consumidor [11].

Arquitectura de la Computación en la Nube.

La arquitectura de la computación en la nube está basada en hardware la misma que engloba la plataforma y aplicaciones que traen consigo los siguientes tipos.

Tipos de Servicios

Para comprender el funcionamiento del cloud computing es fundamental conocer los tres servicios en la nube con las necesidades, las cuales se pueden clasificar en: servicios de software e infraestructura o servicios de hardware [12].

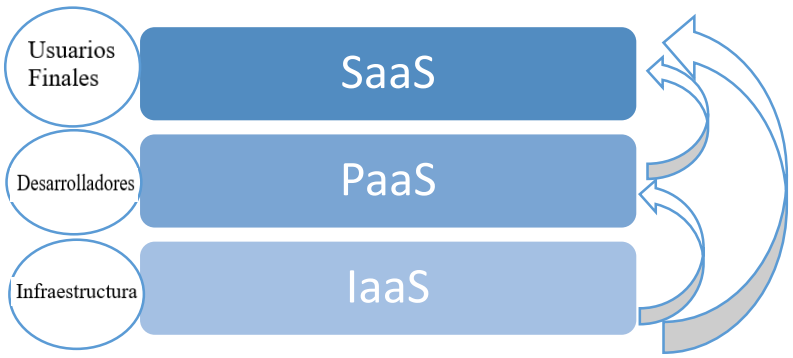


Figura 1 Representación de los servicios en la nube.
Fuente [13].

Software as a Service (SaaS)

Es un modelo de distribución de software que proporciona a los clientes el acceso a éste a través de la red (generalmente Internet). De esta forma, ellos no tienen que preocuparse de la configuración, implementación o mantenimiento de las aplicaciones, ya que todas estas labores se vuelven responsabilidad del proveedor. Las aplicaciones distribuidas a través de un modelo de Software como Servicio pueden llegar a cualquier empresa sin importar su tamaño o ubicación geográfica [14].

Como es orientado a servicios se lo utiliza para fines a largo plazo y los usuarios tienen acceso a estos servicios bajo pago o de forma gratuita dependiendo de la plataforma [13].

Platform as a Service (PaaS)

Es un entorno de desarrollo e implementación completo en la nube, con recursos que permiten entregar todo, desde aplicaciones sencillas basadas en la nube hasta aplicaciones empresariales sofisticadas habilitadas para la nube. Usted le compra los recursos que necesita a un proveedor de servicios en la nube, a los que accede a través de una conexión segura a Internet, pero solo paga por el uso que hace de ellos [15].

El usuario carece de control sobre la infraestructura de almacenamiento o redes [16]. PaaS mantiene la escalabilidad automática, en función de lo que exija la situación. Ejemplo de Paas es Google App Engine, donde los desarrolladores pueden crear sus aplicaciones en Java o Python [16].

Infrastructure as a Service (IaaS)

Brinda la infraestructura como servicio, es decir que el consumidor, va a tener acceso al almacenamiento, sistema operativo, red, entre otras características con el fin de que sus clientes puedan empezar un nuevo proyecto con la infraestructura necesaria, además los proveedores permiten tener escalabilidad y elasticidad ya que los consumidores pueden solicitar los recursos como le sean necesarios. Lo que se debe tener en cuenta es que el usuario no puede administrar o controlar la infraestructura de la nube subyacente [17].

De acuerdo con los diferentes tipos de recursos disponibles se puede separar en tres subcategorías:

Computing as a service (CaaS): Permite a los clientes potencia máxima para los servidores virtuales en la nube o las instancias de las máquinas virtuales, para que bajo una interfaz puedan iniciar, detener, destruir y reiniciar cada de unas las instancias de las máquinas virtuales. También permite la autogestión con la ayuda de interfaces orientadas a la autogestión para obtener un escalado automático y una administración que este automatizada [13].

Storage as a service: Este es un servicio que permite el almacenamiento por medio de pago por los Gb que se desean almacenar en línea [13].

Database as a service (DaaS): Es un servicio secundario de este modelo que permite estandarizar los procesos para para controlar, manipular y acceder. Para que los usuarios puedan realizar estas acciones al momento de ingresar a los datos de la base que esta alojada en la nube [13].

A continuación, una tabla que ejemplifica las nubes de acuerdo con el tipo de servicio.

SERVICIO	NOMBRE	DESCRIPCIÓN
IaaS	OpenNebula	Es una solución de código abierto para construir nubes privadas [18].
		Se basa en el proyecto

	OpenStack	Nebula de la NASA y RackSpace. Surge de su necesidad de manejar grandes cantidades de datos [18].
	Eucalyptus	Se inició como un proyecto de investigación en la Universidad de California [18].
	Nimbus	Plataforma enfocada a la comunidad científica [18].
PaaS	OpenShift	Producto desarrollado por Red Hat [18].
	Cloud Foundry	Desarrollado por VMWare bajo licencia Apache [18].
	Hadoop	Producto desarrollado por Apache [18].
SaaS	OpenBravo	ERP destinado a pequeñas y medianas empresas [18].
	Phreebooks	Solución de contabilidad y ERP basada en Web [18].
	Openi	Proporciona a los usuarios visualizaciones de datos OLAP y bases de datos relacionales [18].
	Jaspersoft	Solución de Business Intelligence [18].

Tabla 1 Ejemplos de nubes de acuerdo con el Tipo de Servicio

Fuente: *[18]*.

Modelos de implementación de computación en la nube

Existen cuatro modelos que se pueden implementar en cloud computing los cuales vamos a describir a continuación:

Public Cloud

Se define como los servicios informáticos ofrecidos por proveedores externos a través de Internet pública, que los pone a disposición de cualquier persona que quiera usarlos o comprarlos. Pueden ser gratuitos o vendidos a pedido, lo que permite a los clientes pagar solo por uso por los ciclos de CPU, el almacenamiento o el ancho de banda que consumen [19].

Los proveedores de la nube pública ofrecen servicios y tecnologías de seguridad, como cifrado e identidad y herramientas de administración de acceso [20].

Existen ciertos inconvenientes en cuanto a su seguridad puesto a que dependerá del

proveedor ya que este debe ser el encargado de utilizar sistemas de seguridad adecuados para que intrusos no puedan acceder a los datos de los usuarios.

Private Clouds

A diferencia de la nube publica las nubes privadas son para uso exclusivo de la empresa u organización las cuales son operadas para el uso interno es decir que un ente externo no podrá tener acceso, esto permite cubrir las necesidades de seguridad [21].

Como se debe implementar la propia infraestructura estas pueden presentar inconvenientes con la elasticidad, ya que su escalabilidad se ve afectada por que expandir una nube privada genera grandes gastos. Pero se debe de tener en cuenta que son mucho más robustas para realizar virtualización e instancias de máquinas virtuales [21].

Hybrid clouds

La nube híbrida es un entorno informático que conecta los servicios de nube privada local de la empresa y la nube pública de terceros en una infraestructura única y flexible para ejecutar las aplicaciones y las cargas de trabajo de la organización [22].

El principio detrás de la nube híbrida es que su combinación de recursos de la nube pública y privada, con un nivel de orquestación entre ellos, le da a la organización la flexibilidad de elegir la nube óptima para cada aplicación o carga de trabajo (y mover las cargas de trabajo libremente entre las dos nubes como las circunstancias cambian) Esto permite que la organización cumpla sus objetivos técnicos y comerciales de manera más efectiva y rentable de lo que podría hacerlo solo con la nube pública o privada [22].

Se pueden realizar las combinaciones como sea conveniente con el fin de reducir los costos.

Community cloud

Este modelo de nube brinda la infraestructura a una comunidad la cual tenga los mismos intereses en la que se apoyan acerca de diversos temas como podría ser la seguridad, como implementar los servicios entre otras inquietudes que surgen, la misma que puede ser administrada por las organizaciones de la comunidad [23].

Este modelo de nube brinda la infraestructura a una comunidad la cual tenga los mismos intereses en la que se apoyan acerca de diversos temas como podría ser la seguridad, como implementar los servicios entre otras inquietudes que surgen, la misma que puede ser administrada por las organizaciones de la comunidad [23].

Plataformas de cloud computing de código abierto.

En este apartado a indica se va a indicar algunas plataformas de cloud computing con el fin de explicar cuál es la más optima, a continuación, las plataformas de código abierto:

OpenStack

Es un software de código abierto el cual se anunció en el año del 2010 y ha tenido un crecimiento rápido por la acogida que le brindan las grandes empresas, es un conjunto de herramientas de software para construir y administrar plataformas de computación en la nube para nubes públicas - privadas y es una colección de proyectos de software de código abierto en el que el especialista en computación de la nube puede usar para configurar y ejecutar su infraestructura de computación y almacenamiento en la nube. Además, proporciona una solución de infraestructura como servicio (IaaS) a través de una variedad de servicios complementarios. Cada servicio ofrece una interfaz de programación de aplicaciones (API) que facilita esta integración [24].

Eucalyptus

Eucalyptus (eucalipto) es una infraestructura (plataforma) open source para la implementación de computación en nube privada en clústers de ordenadores. Eucalyptus es compatible con Amazon Web Services (Amazon EC2 y S3). Está integrado con la distribución GNU/Linux Ubuntu2 9.04 como un útil de “cloud computing”. Eucalyptus puede instalarse fácilmente en la mayoría de las distribuciones GNU/Linux: Debian, CentOS, Red Hat Enterprise Linux (RHEL), SUSE Linux Enterprise Server (SLES), OpenSUSE, Fedora [25].

También puede usar gran variedad de tecnologías de virtualización de hardware incluyendo hipervisores VMware, Xen y KVM para implementar las abstracciones de nube que soporta. Hay 2 ediciones básicas: una propietaria, y otra de código

abierto. Eucalyptus implementa nubes de tipo privado e híbrido, de estilo IaaS (Infrastructure as a Service) [25].

OpenNebula

Es una plataforma de código abierto potente, adaptable e interoperable para la virtualización de centros de datos, así como la gestión de la nube empresarial que proporciona la solución más simple pero rica en características y flexible para la gestión integral de la infraestructura virtualizada en el centro de datos, con el fin de permitir las nubes IaaS en las instalaciones [26].

Además, admite la nube híbrida para combinar la infraestructura local con la infraestructura pública basada en la nube, lo que permite entornos de alojamiento altamente escalables, así como también admite las nubes públicas al proporcionar interfaces en la nube para exponer su funcionalidad para máquinas virtuales (VM), almacenamiento y gestión de redes [26].

De esa manera permitiendo gestionar [27]:

- Redes virtuales
- Máquinas virtuales.
- Clusters.
- Hosts.
- Imágenes.
- Grupos de usuarios

La seguridad también es un elemento importante en OpenNebula, ya que además del sistema de autentificación y los permisos para acceder a las distintas características y herramientas, existe la opción de usar un sistema de control de acceso mediante listas [27].

Arquitectura de OpenNebula.

OpenNebula es una solución simple pero flexible y rica en funciones para construir y administrar nubes empresariales y DC virtualizadas, que combina las tecnologías de virtualización existentes con características avanzadas para múltiples inquilinos,

provisión automática y elasticidad. OpenNebula sigue un enfoque ascendente impulsado por los administradores de sistemas, desarrolladores y necesidades reales de los usuarios [28].

OpenNebula supone que su infraestructura física adopta una arquitectura clásica tipo clúster con un front-end y un conjunto de hosts donde se ejecutarán las máquinas virtuales (VM). Hay al menos una red física que une todos los hosts con el front-end [28].

Esta diseña con tres capas las cuales son: Tools, Core y Drivers son representadas en la siguiente imagen y se explican a detalle a continuación:

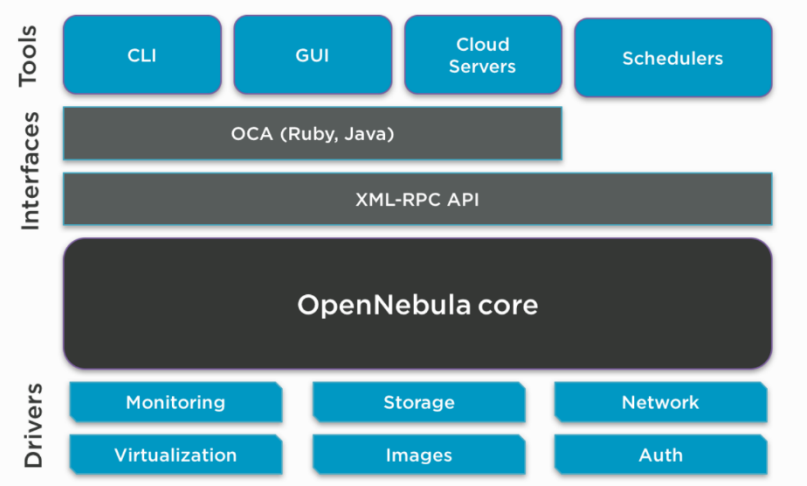


Figura 2 Componentes de OpenNebula

Fuente: [29]

Capa 1 Tools

Esta capa contiene cada uno los módulos que proporcionan las funcionalidades para gestionar la infraestructura, a los administradores y clientes. Ofreciendo las herramientas con la que se accede y controla opennebula [30].

- **Cli (Command Line Interface)**

  Con esta herramienta permite que los usuarios y administradores de OpenNebula gestionen de forma manual la infraestructura virtual ya que acceden por líneas de comandos [30].

  Estos son los comandos que proporciona OpenNebula para interactuar con el sistema según [31]:

oneacct: permite obtener datos contables de OpenNebula.

oneacl: permite gestionar las ACL de OpenNebula.

onecluster: permite gestionar clústeres de OpenNebula.

onedatastore: permite gestionar los almacenes de datos de OpenNebula.

onegroup: permite gestionar grupos OpenNebula.

onehost: permite administrar los hosts de OpenNebula.

oneimage: permite gestionar imágenes de OpenNebula.

Ejemplificación de uno de los comandos, primero se debe ingresar con el usuario oneadmin y ejecutar nuestro comando.

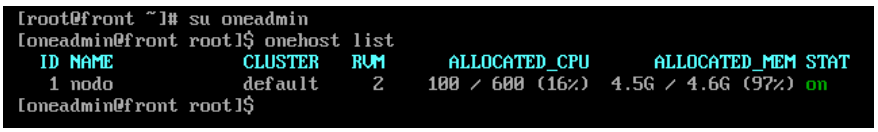


Figura 3 Ejemplificación comando onehost

Fuente:Autor

- **Gui (Graphical User Interface)**

  Permite acceder de forma gráfica a los usuarios, a la administración de OpenNebula, haciendo que sea fácil gestionar cada uno de los elementos [30]. Ofreciendo un entorno informático virtual accesible a través de dos interfaces diferentes nube remota, OCCI y EC2, y a través de dos interfaces web, OpenNebula Sunstone y OpenNebula SelfService. Estos mecanismos acceden a la misma infraestructura, es decir, los recursos creados por cualquiera de los métodos mencionados estarán disponibles instantáneamente en los demás. Por ejemplo, puede crear una VM con la interfaz OCCI , monitorearla con la interfaz EC2 y apagarla usando la interfaz web OpenNebula Sunstone [32].

- **Scheduler**

  Es una entidad independiente en la arquitectura y puede desacoplarse del resto de los componentes la cual emplea la interfaz XML-RPC ya que es un protocolo que permite la llamada a procedimientos en este caso se lo usa para invocar las acciones que se efectuarán en las máquinas virtuales [30]. Ya que este se encarga

de la asignación entre máquinas virtuales pendientes y hosts conocidos. La arquitectura de OpenNebula define este módulo como un proceso separado que puede iniciarse independientemente, sin embargo, se inicia automáticamente cuando se inicia el servicio de OpenNebula.

Como opciones de configuración, podemos encontrar según lo indica el autor [30]:

- Puerto de conexión a oned (por defecto, 2633).

- Tiempo de separación entre dos acciones de planificación (por defecto,

30).

- Número límite de máquinas gestionadas en cada acción de planificación

(por defecto, 300).

- Número máximo de máquinas virtuales con las que comunicamos en

cada acción de planificación (por defecto, 30).

- Número máximo de máquinas virtuales atendidas para un host indicado en cada acción de planificación

Capa 2 Core

Esta capa consta de componentes responsables de manejar las solicitudes de los clientes y controlar los recursos con la ayuda de los siguientes elementos [33]:

- **Request Manager**

  Gestiona las peticiones de los clientes, ofreciendo una interfaz XML-RPC para llamar internamente a un componente [33]. A continuación, los métodos xml-rpc expuestos por OpenNebula. Cada descripción consiste en el nombre del método y valores de entrada y salida.

  Todas las respuestas xml-rpc comparten una estructura común.

Tipo	Tipo de datos	Descripción
OUT	Boolean	Verdadero o falso siempre que sea exitoso o no [34].
OUT	String	Si se produce un error,

		este es el mensaje de error [34].
OUT	Int	Código de error [34].

Tabla 2 Tipos de datos xml-rpc
Fuente: [34]

La salida siempre constará de tres valores. El primero y el tercero son fijos, pero el segundo contendrá el mensaje de error de cadena solo en caso de falla. Si el método es exitoso, el valor devuelto puede ser otro tipo de dato.

Valor	Código	Significado
0x0000	SUCCESS	Respuesta exitosa [34].
0x0100	AUTHENTICATION	El usuario no pudo ser autenticado [34].
0x0200	AUTHORIZATION	El usuario no está autorizado para realizar la acción solicitada [34].
0x0400	NO_EXISTS	El recurso solicitado no existe [34].
0x0800	ACTION	Estado incorrecto para realizar la acción [34].
0x1000	XML_RPC_API	Parámetros incorrectos, por ejemplo, param debe ser -1 o -2, pero se recibió -3 [34].
0x*2*000	INTERNAL	Error interno, por ejemplo, el recurso no se pudo cargar desde la base de datos [34].
0x*4*000	ALLOCATE	El recurso no se puede asignar [34].
0x*8*000	LOCKED	El recurso está bloqueado [34].

Tabla 3 Códigos de error en métodos xml-rpc
Fuente: [34]

- **Virtual Machine Manager**

Se encarga de administrar y monitorizar las máquinas virtuales. Los tres pilares que integran el administrador son la virtualización, la creación de redes y la gestión de imágenes [33].

Estado corto	Estado	Significado
Pend	Pending	De manera predeterminada, una VM comienza en el estado pendiente, esperando que se ejecute un recurso. Permanecerá e n este estado hasta que el planificador decida implementarlo o el usuario lo implemente utilizando el comando onevm deploy [35].
Clon	Cloning	La máquina virtual está esperando que una o más imágenes de disco finalicen la copia inicial en el repositorio (el estado de la imagen aún está en lock) [35].
Prol	Prolog	El sistema está transfiriendo los archivos VM (imágenes de disco y el archivo de recuperación) al host en el que se ejecutará la máquina virtual [35].
Boot	Boot	OpenNebula está esperando que el hipervisor cree la VM [35].
Runn	Running	La VM se está ejecutando (tenga en cuenta que esta etapa incluye las fases internas de arranque y apagado de la máquina virtualizada). En este estado, el controlador de virtualización lo supervisará periódicamente [35].

Migr	Migrate	La VM está migrando de un recurso a otro. Esto puede ser una migración de vida o una migración en frío (la VM se guarda, se apaga o se apaga y los archivos de VM se transfieren al nuevo recurso) [35].

Tabla 4 Estados de las máquinas virtuales en OpenNebula
Fuente: [35]

- **Transfer Manager**

  Presenta un mecanismo general para transferir y clonar imágenes de VM. Es un componente modular que abarca el diseño basado en controladores de OpenNebula, por lo que puede ampliarse e integrarse fácilmente con desarrollos de terceros y prácticamente cualquier arquitectura de almacenamiento en clúster. El nuevo TM le permite reutilizar imágenes de VM ya que puede marcarlas como clonables, y también puede ahorrar espacio ya que OpenNebula ahora crea imágenes de intercambio sobre la marcha. En si es el encargado de gestionar la transferencia de imágenes [33].

- **Virtual Network Manager**

  Es el encargado de gestionar las direcciones IP y MAC para permitirnos crear redes virtuales entre los distintos nodos [33].

  Una red virtual consta de tres partes diferentes [34]:

  - La infraestructura de red física subyacente que lo admitirá, incluido el controlador de red [34].
  - El espacio de direcciones lógicas disponible. Las direcciones asociadas a una red virtual pueden ser IPv4, IPv6, doble pila IPv4-IPv6 o Ethernet [34].
  - Los atributos de configuración del invitado para configurar la red de la máquina virtual, que pueden incluir, por ejemplo, máscaras de red, servidores DNS o puertas de enlace [34].

- **Host Manager**

  Se encarga de monitorizar y administrar los recursos de hardware, posee gran

flexibilidad y nos permite incluir cualquier atributo del host [33].

Los hosts son los servidores administrados por OpenNebula responsables de la ejecución de las máquinas virtuales. Para utilizar estos hosts en OpenNebula, debe registrarlos para que se supervisen y estén disponibles para el planificador. Para crear un host se lo realiza de la siguiente manera [34].

```
oneost create host01 --im kvm --vm kvm
```

Para eliminar un host, al igual que con otros comandos de OpenNebula, puede especificarlo por ID o por nombre. Los siguientes comandos son equivalentes:

```
onehost delete host01
```

También se pueden administrar los hosts usando Sunstone. Seleccionando la pestaña Host y ahí se puede crear, habilitar, deshabilitar, eliminar y ver información sobre sus hosts de una manera fácil.

- **Database**

  Es donde se almacena toda la información generada por OpenNebula. Soporta tanto MySQL como SQLite3, este componente otorga la escalabilidad y fiabilidad que requiere un sistema de gestión de máquinas virtuales [33].

  Esta funcionalidad permite visualizar los logs que están almacenados en la base de datos la que se instala es MySQL para así poder revisar el estado de la nube privada y de cada una de las instancias que están en la misma [33].

  MYSQL

  El software MySQL ofrece un servidor de base de datos SQL (lenguaje de consulta estructurado) muy rápido, multiproceso, multiusuario y robusto. MySQL Server está diseñado para sistemas de producción de carga pesada y de misión crítica, así como para integrarse en software implementado en masa [36].

  Se utiliza la base de datos de MySQL puesto que tiene una versión de código abierto (Gratuita) en la cual se pueden realizar diferentes modificaciones adaptándolo a las necesidades que se tenga. Esta base de datos permite almacenar una gran cantidad de datos puesto que se encuentran organizados por archivos físicos los mismos que permiten optimizar la velocidad, así como

también ofrece flexibilidad al momento de programar.

Características

Según [36] las principales características dentro de MySQL son las siguientes:

Seguridad: cuenta con un sistema de privilegios y contraseñas es flexible, permite la verificación basada en host, cuenta con cifrado durante todo el tráfico [36].

Escalabilidad y limites: soporta gran cantidad de datos que contienen 50 millones de registros, así como también usuarios que usan MySQL Server con 200,000 tablas y alrededor de 5,000,000,000 de filas [36].

El ancho máximo del índice para las InnoDBtablas es de 767 bytes o 3072 bytes [36].

El ancho máximo del índice para las MyISAMtablas es de 1000 bytes [36].

Conectividad: Los clientes pueden conectarse mediante sockets TCP / IP en cualquier plataforma [36].

Localización: el servidor puede proporcionar mensajes de error a los clientes en muchos idiomas, la zona horaria del servidor se puede cambiar dinámicamente, y los clientes individuales pueden especificar su propia zona horaria [36].

Clientes y herramientas: incluye varios programas de clientes y utilidades. Estos incluyen programas de línea de comandos como mysqldump y mysqladmin, como programas gráficos como MySQL Workbench [36].

Capa 3 Drivers

Como su nombre mismo lo dice la capa 3 está formada por Drivers que son los controladores que soportan las diferentes plataformas subyacentes. Estos controladores se ejecutan en procesos separados que se comunican con el módulo Core a través de un protocolo simple de mensajes de texto. Hay controladores para manejar las transferencias de archivos que se implementan mediante protocolos de red como NFS y SSH. Además, hay controladores para administrar máquinas virtuales que dependen de cada hipervisor que se ejecuta en el host [37]. Los cuales

vamos a detallar a continuación [37]:

Monitoreo

Los Controladores de monitoreo (o controladores de IM) se encargan de obtener la información con relación al rendimiento y el estado de las máquinas virtuales que están dentro de los nodos. OpenNebula consulta activamente estos datos o un agente que se ejecuta en los hosts a la interfaz la envía periódicamente, cada uno de los nodos envían la información cada cierto tiempo de los datos y con la ayuda del Front-End que almacena la información en su base de datos para luego procesarlos y mostrarlos en el Dashboard [37].

Virtualización

Esta sección es la encargada de comunicarse con el Hipervisor instalado en los nodos con el fin de manejar las acciones en cada ciclo de vida de una máquina virtual [37].

El hipervisor más utilizado en OpenNebula es KVM, pero también esta Xen, VmWare, en este apartado vamos a tratar acerca de KVM [27].

- **Kvm (Kernel-based Virtual Machine)**

  KVM permite que las máquinas virtuales que utilizan el Kernel de Linux se conviertan en un hipervisor que permita gestionar los recursos de las máquinas virtuales [33].

Almacenamiento

OpenNebula dispone de un repositorio de imágenes de las máquinas virtuales desplegadas o disponibles para ser desplegadas en el futuro. [38]

Este repositorio puede ser implementado de dos formas principalmente [18]:

- **Local:** Las imágenes se almacenan localmente en el controlador de OpenNebula y los hosts que corren los hipervisores. Para desplegar las imágenes se utiliza SSH (SSH Transfer Driver). En esta operación se copia cada imagen bajo demanda desde el almacenamiento local del controlador de OpenNebula al almacenamiento local del host elegido [18].
- **Compartido:** Se utiliza un repositorio común tanto para el controlador de OpenNebula como para los hosts. Se puede utilizar un sistema de ficheros

distribuido para mejorar el rendimiento. La principal ventaja es que no hay transferencia de imágenes por la red, el despliegue es rápido y se pueden migrar máquinas virtuales entre host de manera casi inmediata, facilitando nuevamente el balanceo de carga [18].

Redes

Este controlador brinda un servicio de red el cual permite la conexión, utiliza dos interfaces de red [27].

Red de servicio: Permite acceder a cada uno de los nodos con el fin de obtener la información de estos y también acceder Dashboard [27].

Red de instancia: Permite realizar la conexión con las diferentes instancias que existen, las cuales pueden ser las máquinas, routers virtuales, entre otras [27].

Para que las diferentes instancias se les pueda asignar una ip es necesario crear una red de instancias.

Lo cual se los realiza de la siguiente manera se debe asignar el nombre de la red, bridge y rango de ip.

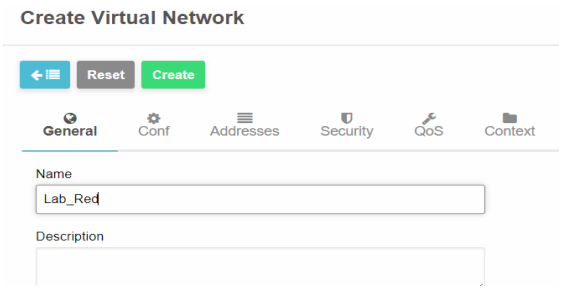


Figura 4 Asignar nombre de la red virtual.
Fuente: Autor

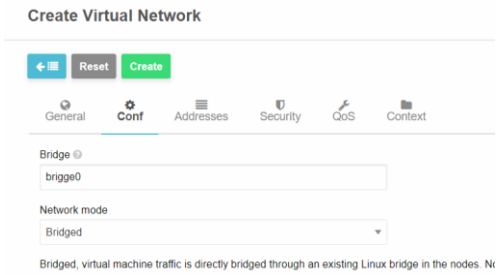


Figura 5 Establecer nombre de bridge en la red virtual.
Fuente: Autor

Autenticación

La autenticación se puede realizar con la ayuda del usuario de Oneadmin y su

correspondiente contraseña, con LDAP o SSH.

Existen cuatro tipos o roles de usuario con sus respectivos privilegios [18]:

• **Administradores:** Pertenecen al grupo oneadmin puede realizar cualquier operación [18].

• **Usuario normal:** Pueden utilizar casi todas las funcionalidades de OpenNebula [18].

• **Usuario public**: Puede realizar operaciones básicas [18].

• **Usuario service:** Se utiliza para interactuar con otros servicios como EC2 o el GUI Sunstone a través de APIs [18].

El acceso a los recursos sigue la política típica de Unix en la que el usuario que crea un recurso es su propietario. Este usuario puede dar permisos a otros usuarios para que utilicen los recursos creados por él [18].

1.1. Correo Electrónico

Es una de las características más utilizadas de Internet, junto con la web. Le permite enviar y recibir mensajes desde y hacia cualquier persona con una dirección de correo electrónico, en cualquier parte del mundo. Usa múltiples protocolos dentro del conjunto TCP / IP. Por ejemplo, SMTP se usa para enviar mensajes, mientras que los protocolos POP o IMAP se usan para recuperar mensajes de un servidor de correo [39].

Es un mensaje que puede contener texto, archivos, imágenes u otros archivos adjuntos enviados a través de una red a un individuo o grupo de individuos especificado [40].

A continuación, se hablará a del siguiente servidor de correo electrónico:

Zimbra

¿Qué es Zimbra?

A Zimbra se le conoce como un servidor de correo el cual ofrece una interfaz web, permite administras todas las actividades que se realizan dentro del servidor, es importante mencionar que este servidor de correo es compatible con otros servidores como por ejemplo Outlook.

Zimbra es la solución de mensajería y colaboración de código abierto líder en el mundo,

en la que confían más de 5000 empresas y clientes del sector público y más de 500 millones de usuarios finales, en más de 130 países [41]. Ofrece dos versiones. Una versión soportada por la comunidad de software abierto (open source) y una versión soportada comercialmente (Zimbra network) que contiene algunas mejoras propietarias [42].

Se deber tener en cuenta que todos tenemos una cuenta de correo electrónico, por eso el informe de The Radicati Group, Inc., 2016-2020, indica que, en 2016, había más de 2600 millones de usuarios de correo electrónico en todo el mundo y para finales de 2020 el número de usuarios de correo electrónico a nivel mundial superará los 3000 millones. Casi la mitad de la población mundial utilizará el correo electrónico a finales de 2020 [41].

Se escogió el servidor de Zimbra dentro de este proyecto ya que es un software libre el mismo que se utiliza como una herramienta de comunicación tanto interna como externa, este servidor trae consigo correo electrónico, calendario, notas, etc, también ofrece alta disponibilidad debido a que compatible con virtualización en infraestructuras como XenServer, KVM y vSphere**,** otra de las razones por la que se escogió este servidor es porque permite realizar copias de seguridad y restauración a nivel de usuario, este servicio ha sido aceptado en el mercado laboral puesto que hoy en día es uno de los lideres a nivel mundial.

Características de Zimbra.

Es importante elegir la plataforma de correo electrónico que se va a utilizar, para ello vamos a explicar algunas características que posee Zimbra esta información se obtiene de la página oficial de Zimbra [41].

Privacidad

Zimbra es uno de los correos en los que se puede confiar ya que este cumple con todos los requisitos de soberanía de datos, por lo cual se brinda completa privacidad al momento de implementar un centro de datos o un entorno de nube en este servicio [41].

Seguridad

Para cumplir con los requisitos de seguridad Zimbra ofrece cifrado de correo electrónico, comunicaciones seguras a través de TLS/SSL, no requiere VPN, compatibilidad con anti-

spam, antivirus. También permite la integración con otras aplicaciones de seguridad externas [41].

Flexibilidad

Permite realizar la implementación de forma local ya sea en el centro de datos, en la nube pública o privada, o a través de un proveedor web que este asociado con Zimbra [41].

Accesibilidad

Zimbra satisface las necesidades de los usuarios ya que este servicio permite acceder a la respectiva cuenta de correo en cualquier lugar y momento, y sobre todo con cualquier dispositivo [41].

Compatibilidad

Zimbra cuenta con un API que permite la integración bidireccional con aplicaciones empresariales como CRM, ERP, también cuenta con un servidor de mensajería el mismo que cuenta con migración de datos. Permite sincronizar la información (buzón de correo, el calendario, los contactos y las tareas) que poseen en otra plataforma la cual puede ser Microsoft Outlook con la ayuda del Conector de Zimbra [41].

Colaboración

Existe nuevas herramientas que se implementaron dentro del servidor como son: Zimbra Chat la misma permite la comunicación entre usuarios, otras de las herramientas es Zimbra Drive, Zimbra Web Client y Zimbra Desktop las cuales ofrecen formas sencillas de almacenar, compartir y colaborar en archivos de todo tipo [41].

Costes operativos reducidos

Permite ahorrar dinero en los costes de licencia y administración [41].

Facilidad administrativa

Con esta opción se puede ahorrar tiempo y dinero con la fácil administración de Zimbra [41].

Código Abierto

Esta es una de las funcionalidades que lo hacen único a Zimbra, ya que las empresas

pueden utilizar Zimbra Open Source Edition, o pueden actualizar a Zimbra Network Edition para obtener funciones como la sincronización móvil [41].

Integración

Permite la integración con otras aplicaciones con la ayuda de Zimlets y nuestras API.

Comunidad

Cuenta con una sólida comunidad: más de 60 000 miembros de código abierto y una red global de más de 1900 socios. Una de las maneras de comunicarse también es mediante los foros que existen [41].

Arquitectura de Zimbra.

Incluye integraciones de código abierto utilizando estándares industriales protocolos El software de terceros ha sido probado y configurado para funcionar con el software Zimbra. [43]

Indican [42] que Zimbra está formado por un conjunto de componentes los cuales permiten brindar una solución completa. Dentro de estos componentes está el núcleo del servidor el mismo que está en Java, se utiliza Jetty que es un servidor http para la comulación con las aplicaciones. Además, el servidor se integra con otros componentes como el MTA, la base de datos y los paquetes de seguridad.

El componente MTA (Mail Transfer Agent) es el agente de transferencia de correos que enruta los mensajes de correo con el servidor de Zimbra. El cual está basado en postfix. Zimbra incorpora varios filtros de seguridad, como antivirus y antispam, entre otros [42].

También soporta por defecto los protocolos principales de cifrado de canal, SSL y TLS. Los filtros en el lado del servidor mediante James/Sieve [42].

A continuación, vamos a detallar cada uno de los componentes de la arquitectura:

Componentes de la arquitectura

Según [44] La utilidad cada uno de los servicios de Zimbra y que función desempeñan en nuestra implantación es la siguiente:

Zimbra Core

Es un servicio que se instala automáticamente en el momento en el que se instala cualquier otro servicio de Zimbra, en si este servicio instala librerías, utilidades, herramientas de monitorización y ficheros básicos de configuración [44].

Zimbra LDAP

La autenticación del usuario se proporciona a través de OpenLDAP software. Cada cuenta en el servidor Zimbra tiene un ID de buzón único que es el principal punto de referencia para identificar la cuenta. El OpenLDAP esquema ha sido personalizado para ZCS. El servidor LDAP de Zimbra debe ser configurado antes que los otros servidores. Puede configurar la replicación LDAP, configurar un servidor LDAP maestro y servidores LDAP de réplica [43].

Zimbra MTA

> Según [44] este es el servicio de MTA (Mail Transfer Agent), es el encargado de recibir el correo mediante el servicio de SMTP mediante el postfix y retransmitirlo al Mailbox correspondiente, esta transmisión se realiza mediante el protocolo de comunicación LMTP (Local Mail Transfer Protocol), el cual es más ligero que el SMTP. Este servicio lleva intrínsecos varios servicios que se pueden habilitar por consola y algunos mediante la interface web. Estos servicios son:
>
> Amavis-New, el cual se encarga de gestionar el AntiVirus y el AntiSpam [44].
>
> Sistema de AntiVirus y AntiSpam [44].
>
> Sistema de POP3, IMAP, IMAPSSL, POP3SSL, etc [44].
>
> Sistema de control del flujo de correo (PolicyD) [44].

Zimbra Store

Según [42] este componente utiliza Jetty como contenedor de servlets el cual permite almacenar el correo electrónico.

Para realizar el anterior procedimiento se lo hace de la siguiente manera cada cuenta es configurada en el servidor, de esa forma la cuenta esta enlazada con el buzón de

correo que contiene todos los mensajes y ficheros adjuntos. El servidor de buzones está formado por [42]:

- El almacén de datos.
- El almacén de mensajes.
- El almacén de índices.
- Las utilidades de conversión de adjuntos a HTML.

Cada uno de los ítems anteriores son importantes ya que el servidor de Zimbra tiene su propio almacén de datos, mensajes y de índices. En el momento de que llega un correo electrónico Zimbra crea un proceso para indexar el mensaje y crea un hilo para la conversión de adjuntos a formato HTML, que a su vez es indexado por otro hilo. En cambio, el almacén de datos es una base de datos MySQL en la cual los identificadores de mensajes son enlazados con las cuentas de usuario. El mismo que relaciona el identificador del buzón con la cuenta de usuario a la que pertenece en el directorio LDAP, finalmente el almacén de mensajes guarda todos los mensajes y sus adjuntos en formato MIME. Los mensajes enviados a múltiples destinatarios dentro del mismo servidor sólo son almacenados una vez. La tecnología necesaria para indexar y buscar la proporciona Lucene. Se mantienen índices sobre cada buzón [42].

Zimbra SNMP y Zimbra Logger

La instalación de estos componentes es opcional, pero a la vez muy importantes ya que Zimbra Logger instala herramientas de agregación de logs, informes, seguimiento de mensajes y esto permite obtener datos muy importantes en el caso de existir fallas o para verificar el estado del servidor y los correos electrónicos. El componente Zimbra SNMP obtiene la información cada cierto tiempo del estado del sistema [42].

1.2.Machine Learning

¿Qué es Machine Learning?

Es una disciplina científica la cual está relacionada con el ámbito de la Inteligencia Artificial, la misma que permite crear sistemas que aprenden automáticamente, en este contexto aprender quiere decir identificar patrones complejos en millones de datos. La

máquina no aprende por sí misma, sino un algoritmo es el que modifica con la constante entrada de datos en la interfaz, y que puede, de ese modo, predecir escenarios futuros o tomar acciones de manera automática según ciertas condiciones. Como estas acciones se realizan de manera autónoma por el sistema, se dice que el aprendizaje es automático, sin intervención humana [45].

¿Cómo funciona Machine Learning?

Machine Learning utiliza algoritmos los mismos que realizan la mayor parte de estas acciones por su cuenta. Los cuales obtienen sus propios cálculos según los datos que se recopilan en el sistema, y cuantos más datos obtienen, mejores y más precisas serán las acciones resultantes. Es decir que las computadoras se programan a sí mismas, hasta cierto punto, usando los algoritmos. Estos funcionan como ingenieros que pueden diseñar nuevas respuestas informáticas, como respuesta a la información que se les suministra a través de su interfaz u otros medios [45].

Tipos de Machine Learning.

Dentro de esta rama podemos mencionar dos tipos de aprendizaje automático los cuales son: Supervisado y No Supervisado, estos tipos poseen adicionalmente se subdividen según el tipo de datos que manejan que pueden ser Continuo y Discreto, a continuación, una tabla que representa los tipos de Machine Learning [46].

	Supervisado	No Supervisado
Continuo	Regresión Lineal	Reducción de dimensión
Discreto	Clasificación	Agrupamiento

Tabla 5 Tipos de Machine Learning
Fuente: [46]

Vamos a explicar cada uno de los tipos de aprendizaje automático.

Supervisado

Este tipo de aprendizaje trabaja con la información que se le brinda de entrenamiento. Una vez que se entrena al sistema proporcionándole cierta cantidad de datos definiéndolos al detalle con etiquetas, buscan generalizar y predecir a partir de la información suministrada [45].

No Supervisado

Realiza el aprendizaje de forma diferente al supervisado ya que estos sistemas tienen como finalidad la comprensión y abstracción de patrones de información de manera directa. Este es un modelo de problema que se conoce como clustering, es un método de entrenamiento más parecido al modo en que los humanos procesan la información [46].

Como se mencionó en la tabla anterior los tipos de Machine Learning se subdividen según el tipo de datos que manejan [46]:

Continuos: Información cuantitativa/numérica.

Discretos: Información cualitativa.

Tendencias en Machine Learning.

Uno de los elementos de esta transformación digital es el aprendizaje automático o Machine Learning es la aplicación de técnicas y algoritmos capaces de aprender a partir de distintas y nuevas fuentes de información, construyendo algoritmos que mejoren de forma autónoma con la experiencia. Esto permite disponer de métodos capaces de detectar automáticamente patrones en los datos, y usarlos para predecir sobre datos futuros en un entorno de incertidumbre [47].

Los componentes principales del Machine Learning se pueden clasificar en cuatro grupos [47]:

- Las fuentes de información, que pueden aportar datos tanto estructurados como no estructurados, y que son la base del resto de componentes [47].
- Las técnicas y algoritmos para el tratamiento de información no estructurada (texto, voz, video, etc.) y para la obtención de patrones a partir de los datos [47].
- La capacidad de autoaprendizaje, que permite que el algoritmo se adapte a los cambios en los datos [47].
- El uso de sistemas y software como vehículo para la visualización de la información y la programación [47].

Por otro lado, algunas técnicas pueden utilizarse para la transformación de información no estructurada (textos, sonidos, imágenes, etc.) en datos que puedan ser analizados y procesados por el sistema. Entre estas técnicas cabe destacar el uso de estadísticos o la

clasificación de palabras en categorías de cara a la comprensión del texto escrito, el uso de redes neuronales para el reconocimiento de voz o de imágenes, la aplicación de cadenas de Markov para la construcción de textos en lenguaje natural, o la aplicación de algoritmos no supervisados de clasificación para la organización de imágenes [47].

Existe un gran número de casos de implementación de técnicas de Machine Learning en distintos sectores. Como ejemplos, destacan aplicaciones en las industrias de la educación (como tutores inteligentes), las finanzas (para el trading automático, roboadvisors, detección del fraude, medición del riesgo o elaboración de modelos prospect con fines comerciales), la salud (el diagnóstico por imagen, la gestión de consultas de tratamiento y sugerencias, recopilación de información médica o la cirugía robótica) [47].

Entre las principales tendencias en la implantación de técnicas de Machine Learning se pueden destacar el uso de fuentes de información que recopilan datos en tiempo real, la mayor importancia que se le da a obtener un poder de predicción elevado en relación con la interpretabilidad de los modelos, la incorporación de la capacidad de que el algoritmo se modifique de forma autónoma en función de los cambios que se van produciendo en la población objetivo, o la inversión en arquitecturas e infraestructuras tecnológicas que garanticen la escalabilidad en la capacidad de almacenamiento de datos y una mayor velocidad de procesamiento, combinadas con soluciones basadas en Cloud [47].

1.3.Spam

Definición

Es spam es un correo electrónico que llega de manera no solicitada a nuestras bandejas, los cuales son enviados automáticamente a un gran número de direcciones al mismo tiempo. Popularmente el spam es conocido como correo basura, el cual suele ser usado con fines de marketing ya que de esa manera crean publicidad de cada uno de sus productos y los hackers pueden hacer uso del spam para enviar malware [48].

Tipos de Spam

Actualmente se vive el ataque de diferentes tipos de spam los cuales vamos a explicar a continuación:

- **Spam en correos electrónicos**

Es más habitual el spam en los servicios de correo electrónico ya que es una herramienta que utilizan la mayoría de las personas ya sea para buscar empleo, para temas personales, para el trabajo y todas esas personas se vuelven vulnerables ante el spam [49].

- **Mensajes con contenidos sexuales o para adultos**

Esto tiene relación con la publicidad erótica o que posea contenido para adultos con el fin de redireccionarlos a los sitios con suscripción por pago, pero actualmente no se da este tipo de spam en abundancia por el acceso libre o gratuito a la paginas con contenido para adultos [49].

- **Mensajes con contenidos sobre salud y medicina**

Este tipo de contenido está protagonizado por productos milagrosos, terapias mágicas o chamanes que superan las posibilidades de los especialistas más reputados en medicina, nutrición, etc. Además, tiene relación con casos de que han recuperado su cabellera con tratamientos o reducción de medidas, pero todo esto lo hacen con montajes y testimonios falsos con el fin de atraer clientes [49].

- **Mensajes sobre contenidos educativos y profesionales**

Esto tipo de contenido afecta cuando surge una emergencia ante el desempleo, los spammers abusan de la inocencia de algunos usuarios ofreciéndoles empleo sin nada a cambio solo para ilusionar a las personas o les ofrecen realizar algún trámite educativo con el mismo de fin de engañar a las personas [49].

Estadísticas de Spam

Según Kaspersky este es el porcentaje de tráfico spam que se generó ah nivel de correos electrónicos en el primer trimestre del 2019.

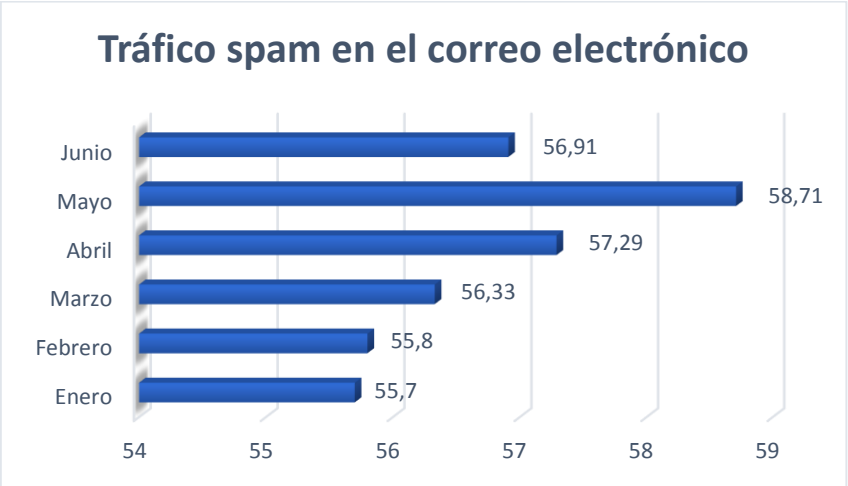


Figura 6 Tráfico spam en el correo electrónico según Kaspersky
Fuente: [50].

Según la figura 6 se puede evidenciar que el mes que tuvo más tráfico por spam en el correo electrónico fue en mayo que es el 58.71%. El porcentaje promedio de spam en el tráfico de correo global fue de 57.64%, un aumento de 1.67 en comparación con el período de informe anterior.

Países fuentes de Spam.

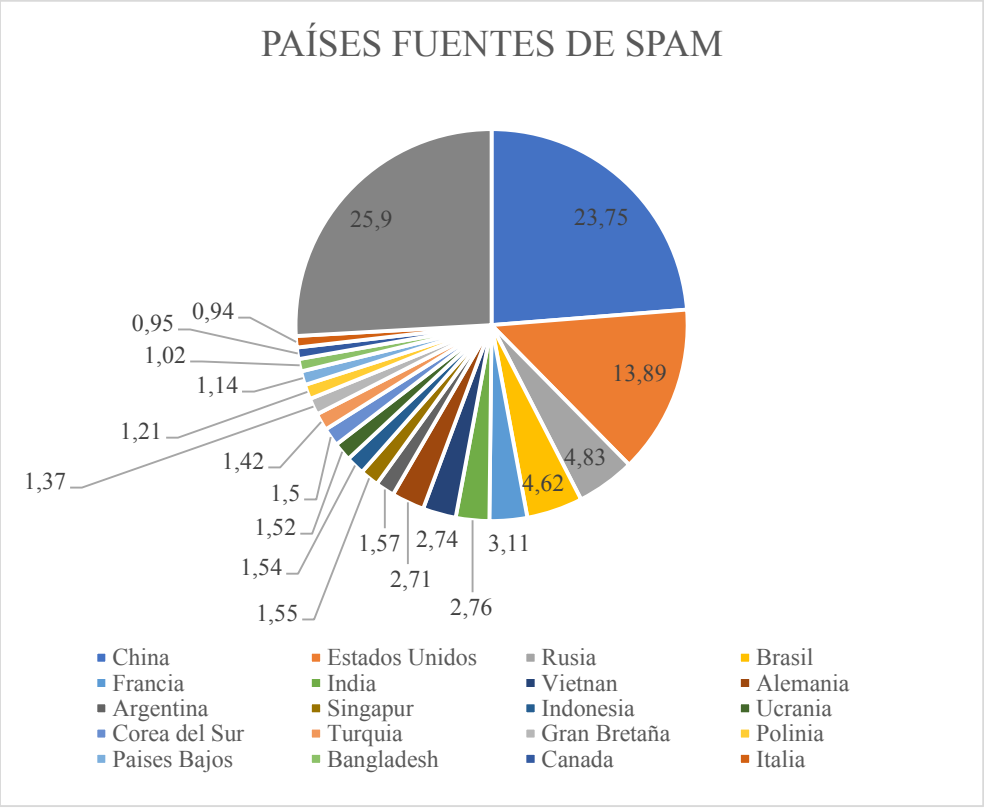


Figura 7 Países fuentes de Spam
Fuente: [50]

Según la figura 7 los principales países que encabezan la lista son los siguientes en

primer lugar fue China con 23.72%, EE. UU. Quedó en segundo lugar con 13.89%, Rusia quedó en tercer lugar con 4.83% y Brasil quedó en cuarto lugar 4.62%.

Países atacados por phishers

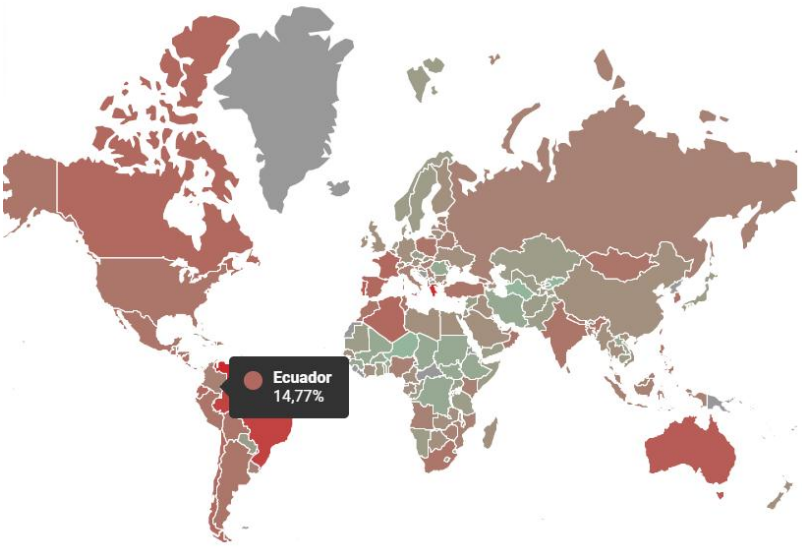


Figura 8 Países atacados por phishers
Fuente: [50].

Según las estadísticas geográficas de ataque que nos brinda Kaspersky podemos visualizar que Ecuador representa un 14.77% de los países que son atacados por phishing, la tabla a continuación indica el top 5 de los países que son afectados.

PAÍS	PORCENTAJE
Grecia	26,20%
Venezuela	25,67%
Brasil	20,86%
Australia	17,73%
Portugal	17,47%

Tabla 6 Top 5 de los países afectados por phishing

Fuente: [50]

1.4. Técnicas Antispam

Existen dos mecanismos antispam que permiten evitar el ingreso de los correos no deseados de nuestras bandejas de entrada en los servicios de correo electrónico los cuales

son:

Filtros estáticos

La efectividad de los filtros estáticos es limitada por que se basan en un conjunto limitado de palabras, frases y reglas estáticas que son sencillas, que permiten analizar el cuerpo del mensaje y la cabecera [51].

Este tipo de filtros trabaja con listas negras, listas blancas, palabras y frases clave en el tema y cuerpo del mensaje los cuales se explican a continuación según el autor [51]:

Listas negras: Permite identificar las direcciones que originan correo basura y bloque los mensajes [51].

Listas blancas: Permite el ingreso de los mensajes de las direcciones que están agregadas en la lista [51].

Análisis de contenido: Analiza el contenido del cuerpo y la cabecera del mensaje con la ayuda de los mecanismos de búsqueda de palabras y de frases clave que posean contenido spam [51].

Pero estas técnicas antispam no son tan efectivas, por eso se proponen otras técnicas como los filtros inteligentes.

Filtros basados en contenidos

Este tipo de filtros utilizan técnicas basadas en contenido, los cuales emplean características extraídas de la cabecera o del cuerpo del mensaje para realizar la clasificación de un correo, para determinar los atributos comunes a los mensajes spam y legítimos. La elección de los términos más representativos de cada mensaje se realiza empleando técnicas de selección de características, estos son los modelos que trabajan de la siguiente manera [52]:

SVM (Máquinas de Vectores de Soporte)

Este tipo de modelo posee una base teórica sólida, que hace uso de la teoría del aprendizaje estadístico. Su funcionamiento se basa en la idea de transformar los datos existentes para encontrar un hiperplano de mayor dimensión donde maximizar la separación existente entre las clases. El mecanismo de aprendizaje está basado en la

idea de minimización de riego estructural (SRM) [52].

Este modelo es apropiado para problemas de categorización de texto y ha sido utilizado con éxito en el ámbito de la detección y filtrado de correos spam. Utilizando SVM no es necesario realizar una selección de términos previa como en otros algoritmos de aprendizaje automático, puesto que su capacidad de aprendizaje no se degrada a medida que se incorporan nuevas características [52].

Boosting de Árboles de Decisión

Este modelo utiliza mecanismos de aprendizaje débiles, los cuales son algoritmos que aprenden con una tasa de error menor que 50%. El funcionamiento de esta técnica se basa en combinar las hipótesis débiles generadas por los mecanismos de aprendizaje débiles, en una única hipótesis de gran precisión. Para la obtención de hipótesis diferentes, este tipo de algoritmos se ejecutan un cierto número de veces sobre distintos subconjuntos de entrenamiento, suele ser utilizado para el filtrado de correo spam [52]**.**

Chung Kwei

Es un modelo que permite la detección de correos spam basado en el análisis de mensajes spam y la identificación automática de patrones, este algoritmo se destaca en su rapidez y capacidad para adquirir conocimiento de forma incremental. El proceso se basa en la ejecución de un algoritmo llamado Teiresias, capaz de encontrar patrones que aparecen dos o más veces en el diccionario de entrenamiento [52].

Filtro Bayesiano

Es el filtro más conocido en el ámbito del filtrado de correos spam para la clasificación de textos mediante modelos de aprendizaje automático. Su base teórica es sustentada en el teorema de Bayes. Los estudios realizados en el campo del filtrado antispam demuestran una gran efectividad porque permite obtener mejores resultados que en relación de los otros métodos de filtrados. Es un sistema basado en la probabilidad que mejora con el tiempo, calcula la probabilidad de que un mensaje sea spam en función de su contenido. A diferencia de los simples filtros basados en palabras, este tipo de filtro aprende del spam entrante y de los correos electrónicos buenos, lo que resulta en un enfoque anti-spam muy robusto, adaptable y eficiente que rara vez arroja falsos positivos [53].

Una vez que el filtro Bayesiano ya está entrenado, puede calcular automáticamente la probabilidad de que cada correo electrónico recibido en función de las palabras que contiene, de esa manera permite detectar si el correo es spam o no es spam. Además, es multilingüe y al ser adaptable, puede utilizarse con cualquier idioma [52].

Teorema de Bayes

El Teorema de Bayes define la realización de cálculos probabilísticos tomando en consideración a aquella información que es empleada para saber cuál es la probabilidad condicional que tiene un suceso. Este concepto de Teorema de Bayes fue desarrollado por el matemático Thomas Bayes. Su intención era determinar la probabilidad de un suceso con respecto a la probabilidad de otro suceso diferente para ello se utiliza la siguiente fórmula matemática [54].

$$P\left[{}^{A_n}/_{B}\right] = \frac{P\left[{}^{B}/_{A_n}\right] * P[A_n]}{\sum P\left[{}^{B}/_{A_i}\right] * P[A_i]}$$

Donde B es el suceso sobre el que tenemos información previa que vendría a ser el diccionario con el tenemos alimentado al filtro y A(n) son los distintos sucesos condicionados que son los correos electrónicos que ingresan. En la parte del numerador tenemos la probabilidad condicionada, y en la parte de abajo la probabilidad total [55].

Cómo un filtro bayesiano examina un mensaje de correo electrónico

El filtro bayesiano analiza las siguientes características en los mensajes de correo electrónico [53]:

- Palabras en el cuerpo del mensaje
- Palabras en el encabezado del mensaje (como el remitente y la ruta del mensaje)
- Otros elementos como el código HTML / CSS (como los colores y otros formatos)
- Palabras y frases

¿Pueden los spammers pasar los filtros bayesianos?

Los spammer solo podrían lograr su objetivo de vulnerar un buen filtro bayesiano, es decir uno que este bien entrenado, si su mensaje spam parece un correo común o que este perfectamente normal, pero a los spammers no les conviene porque su objetivo es que el usuario caiga con (publicidad o que compre algo dando clic en un enlace), esas características no poseen un mensaje bien estructurado la mayoría de las veces. Por ende, resultaría muy complicado para los spammers pasar un filtro bayesiano porque su entrenamiento los hace muy potentes [53].

Capítulo 2: Topología Propuesta

En esta sección se indica la topología propuesta y los requisitos necesarios tanto de hardware y de software.

2. Componentes funcionales.

Servidor DNS

La función del servidor de DNS (Sistema de nombres de dominio) es la traducción de una dirección IP a un nombre, donde cada nombre pertenece a un servicio o sitio el mismo que es el responsable de conservar la información del domino al que pertenece. Es por ello que cada vez que se ingrese el nombre de domino este realizará la petición la misma que permitirá mostrar la información que se encuentra dentro del domino solicitado.

Dentro de este proyecto el servidor DNS cumple una función importante ya que permitirá resolver las Direcciones IP que se utilizan dentro de los diferentes servicios que se están implementado, dando facilidad de acceder a los servicios sin tener confusión alguna al momento de requerir un servicio en específico dentro del navegador.

Redes Virtuales (Vnet)

Las redes virtuales son una combinación de hardware y software cuya interfaz es emular una red física, la misma que permitirá la comunicación entre las diferentes máquinas virtuales, como si estuvieran conectadas dentro de una misma red.

Dentro de este trabajo se ha creado una red virtual para el servicio de correo electrónico cuya función es permitir la comunicación con la plataforma de OpenNebula y con el servidor de correo el cual se encuentra en la red local.

Servidor de correo Zimbra

Este servicio de correo se instalará dentro de OpenNebula, así como también dentro de la red local.

Con la ayuda de este servicio se podrá realizar las pruebas de envío y recepción de mensajes entre diferentes usuarios, los mismos que serán creados dentro de este servicio tanto en el servidor que se encuentra en la nube como en el que se encuentra en la red local.

2.1. Topología

A continuación, la topología propuesta en la cual se implementará el Prototipo de filtrado Anti-Spam usando la nube privada sobre la plataforma de OpenNebula.

Para la implementación del Prototipo de filtrado Anti-Spam en una nube privada, la herramienta que se utilizara como infraestructura es la plataforma de OpenNebula en la cual se instalará el servidor de correo Zimbra y se configurara filtro Anti-Spam el mismo que se ejecuta con la ayuda de un script bash dentro del servidor de correo el mismo script bash se ejecuta el filtrado anti-spam que está en python, permite obtener los correos que son enviados hacia los usuarios, los mismos que serán filtrados para de esa manera comprobar que lleguen correos limpios a las bandejas de entrada de los usuarios.

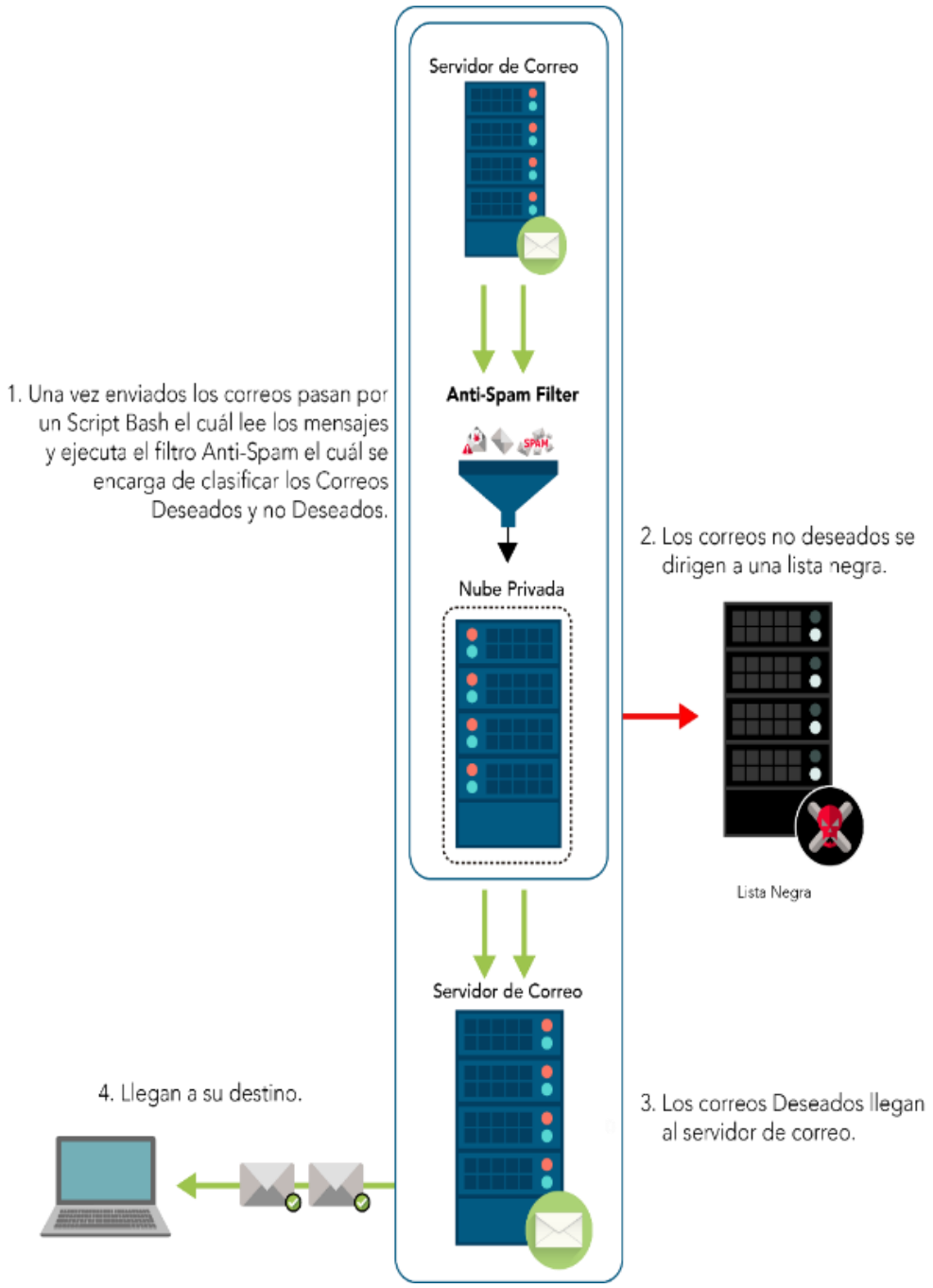


Figura 9 Topología propuesta
Fuente: Autor

A continuación, tenemos la topología de como interactúa la infraestructura física para el análisis de spam, para ello es necesario dos máquinas que funcionan como el Front-End de Opennebula y el nodo, dentro de Opennebula tenemos una instancia (VM) la cual posee el servidor de correo y el filtro anti-spam, en la red local tenemos el servidor DNS que resuelve el nombre de dominio de cada uno de los servicios y un servidor de correos con que el que se realiza las prueba de envió de mensajes ham y spam al servidor de la nube privada, para que realice el análisis, de esa manera determine si el mensaje debe ser enviado a la bandeja de entrada.

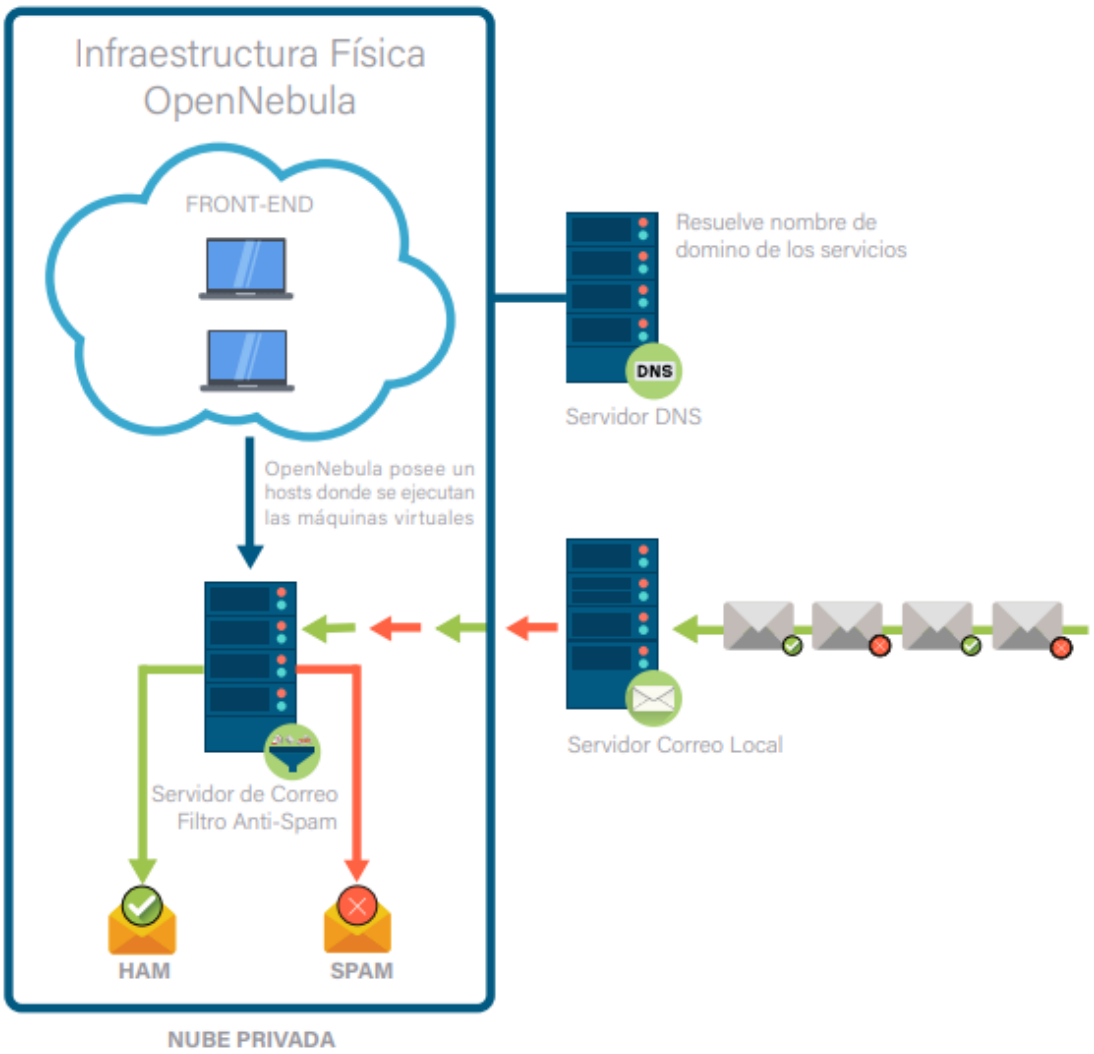


Figura 10 Topología de la infraestructura.
Fuente: Autor

A continuación, se muestra la topología de red en OpenNebula, esta topología se utiliza la red tipo VNet llamada Laboratorios, así como también la máquina virtual en la cual se instaló el servidor de correo, y se implementó el filtro antispam.

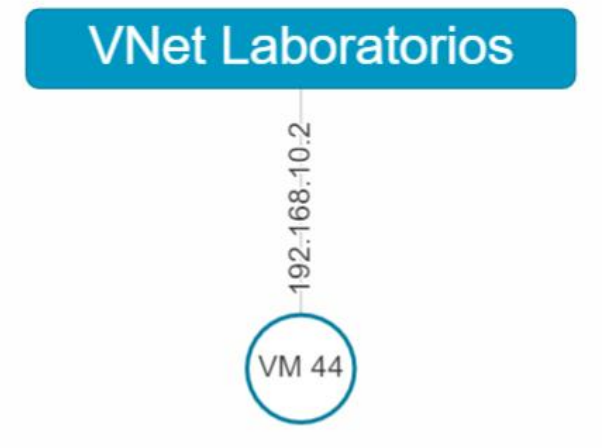


Figura 11 Topología de red de OpenNebula
Fuente: Autor

A continuación, se explica los recursos necesarios en cuanto a la parte del hardware y software que se utiliza en las topologías antes mencionadas.

2.2.Recursos de Hardware y Software utilizados para la instalación de OpenNebula

Para la instalación de OpenNebula se utiliza una PC que posee las siguientes características:

Marca/Modelo	Hypervisor	RAM	Disco Duro	Procesador	Tipo de Procesador
Asus zephyrus gm501gs	VMware Workstation	32	1TB	6 núcleos 12 subprocesos	Intel Core i7 8750H

Tabla 7 Características del equipo físico.
Fuente: Autor

2.3.Recursos de Hardware OpenNebula

Recursos para la instalación de la plataforma de OpenNebula, son dos máquinas virtuales, la una se utiliza para la parte del front-end y la otra se utiliza para el nodo en el cual se va a alojar las diferentes instancias que se creen.

Características del Front:

Sistema Operativo	RAM	Disco Duro	Procesador
CentOS 7 x64	2GB	150GB	1 procesador

Tabla 8 Características físicas la maquina Front
Fuente: Autor

Las características del front-end no son relativamente altas puesto a que el front-end de la plataforma de OpenNebula es la parte principal de administración en la que se ejecutaran los servicios de la plataforma mencionada. La máquina del front-end necesita conectividad de red con todos los hosts, la conectividad que necesita puede ser por red o por enlace directo en este caso la conectividad se realiza por red Vnet.

Características del Nodo:

Sistema Operativo	RAM	Disco Duro	Procesador
CentOS 7 x64	6GB	300GB	6 procesadores

Tabla 9 Características del Nodo
Fuente: Autor

Las características del nodo son más altas que las del front- end puesto que dentro del nodo se alojan los hipervisores que proporcionan los recursos necesarios para ejecutar máquinas virtuales dentro de la plataforma.

Capítulo 3: Marco Metodológico.

En esta sección se describe la metodología que se implementó con el fin de cumplir los objetivos antes mencionados.

3. Metodología.

La metodología seleccionada para el desarrollo de este proyecto es Scrum puesto que es una metodología ágil y flexible la misma permite trabajar de un modo ordenado brindando la habilidad de separar las tareas así como también los procesos que se están realizando, es por ello que con la ayuda de esta metodología se desarrolló el software para filtrado de correos, la configuración de la nube privada y la creación de instancias misma en la que está configurado el servicio de correo y el filtrado, las características del equipo en el que se realizó la implementación de la infraestructura se encuentran en la Tabla 7.

La metodología Scrum consta de cuatro etapas las mismas que permitirán cumplir con los objetivos previamente mencionados, dentro de estas etapas se describirá todo el proceso de desarrollo del proyecto.

Etapa 1: Descripción de librerías de Machine Learning implementadas en el filtro Anti-Spam.

Dentro de esta etapa se hará énfasis a las librerías que se utilizan dentro del filtro bayesiano con el fin de dar a conocer sus funcionalidades de estas.

Para el desarrollo del filtro se determinó utilizar Machine learning con el fin de realizar aprendizaje automático a partir de un conjunto de datos, para lograr el aprendizaje automático es necesario utilizar algunas librerías las mismas que facilitan el desarrollo del filtro bayesiano. A continuación, se mencionarán las librerías utilizadas dentro del filtro, así como también el modo de instalación dentro de Python.

Librerías y sus funcionalidades

1. **Nltk:** es (Natural Language Toolkit) es una biblioteca de Procesamiento de Lenguaje Natural que utiliza el lenguaje de programación Python [56].

   En el filtrado anti-spam es de gran ayuda ya que permite analizar los textos que se están enviando así como también ayuda a dividir oraciones de párrafos, dividir palabras,

reconocer parte del texto de esas palabras, resaltar los temas principales e incluso ayudar a la máquina a comprender de qué se trata el texto [57].

Dentro de la librería nltk está la librería de **NaiveBayesClassifier,** la misma que nos permite obtener la tasa de error del filtro anti-spam, así como tambien permite realizar el cálculo probabilístico que el teorema de bayes requiere.

2. **NaiveBayesClassifier:** El clasificador Naive Bayes es un algoritmo rápido, preciso y confiable. Los clasificadores ingenuos de Bayes tienen alta precisión y velocidad en grandes conjuntos de datos [58].

   Esta librería está relacionada con el teorema de bayes el mismo que se explica en el presente capitulo en la Etapa 2.

3. **Sklearn:** Proporciona una gama de algoritmos de aprendizaje supervisados y no supervisados a través de una interfaz consistente en Python [59].

   El algoritmo utilizado dentro de esta librería es el de aprendizaje supervisado puesto el método bayesiano forma parte de este a tipo de aprendizaje.

   Cuenta con algoritmos de clasificación, regresión, clustering y reducción de dimensionalidad. Además, presenta la compatibilidad con otras librerías de Python como NumPy, SciPy y matplotlib [60].

La instalación de las librerías antes mencionadas se podrá encontrar en el presente capitulo en la Etapa 3.

Etapa 2: Desarrollo del filtro anti-spam y cálculo del teorema de bayes

Dentro de esta etapa se dará a conocer el procedimiento que se llevó a cabo para el desarrollo del filtro anti-spam, también se realizara el cálculo de teorema de bayes de acuerdo con el diccionario de entrenamiento que se construyó con archivos de tipo ham y spam.

2.1. Explicación de los requerimientos para el desarrollo del filtro antispam

Para la propuesta del filtro anti-spam será necesario realizar la selección del lenguaje de programación en el cual se llevara a cabo la programación del mismo, también se deberá establecer la preparación de los datos de texto, creación el diccionario de palabras, extracción de características y por último el entrenamiento del clasificador.

El lenguaje de programación a utilizarse en este proyecto será Python puesto que es de código abierto, fácil de aprender, es multiparadigma y multiplataforma, su código es legible y debido a que es uno de los más utilizados a nivel de Machine Learning según las estadísticas de Satck Overflow.

Python, el lenguaje principal de más rápido crecimiento en la actualidad. Esto significa que, proporcionalmente, más desarrolladores desean continuar trabajando con estos que

con otros idiomas [61].

Python es el lenguaje más buscado por tercer año consecutivo, lo que significa que los desarrolladores que aún no lo usan dicen que quieren aprenderlo [61].

Python lidera en lenguajes de desarrollo de Machine Learning debido a su simplicidad y facilidad de aprendizaje. Python es utilizado por más y más científicos de datos y desarrolladores para la construcción y análisis de modelos [62].

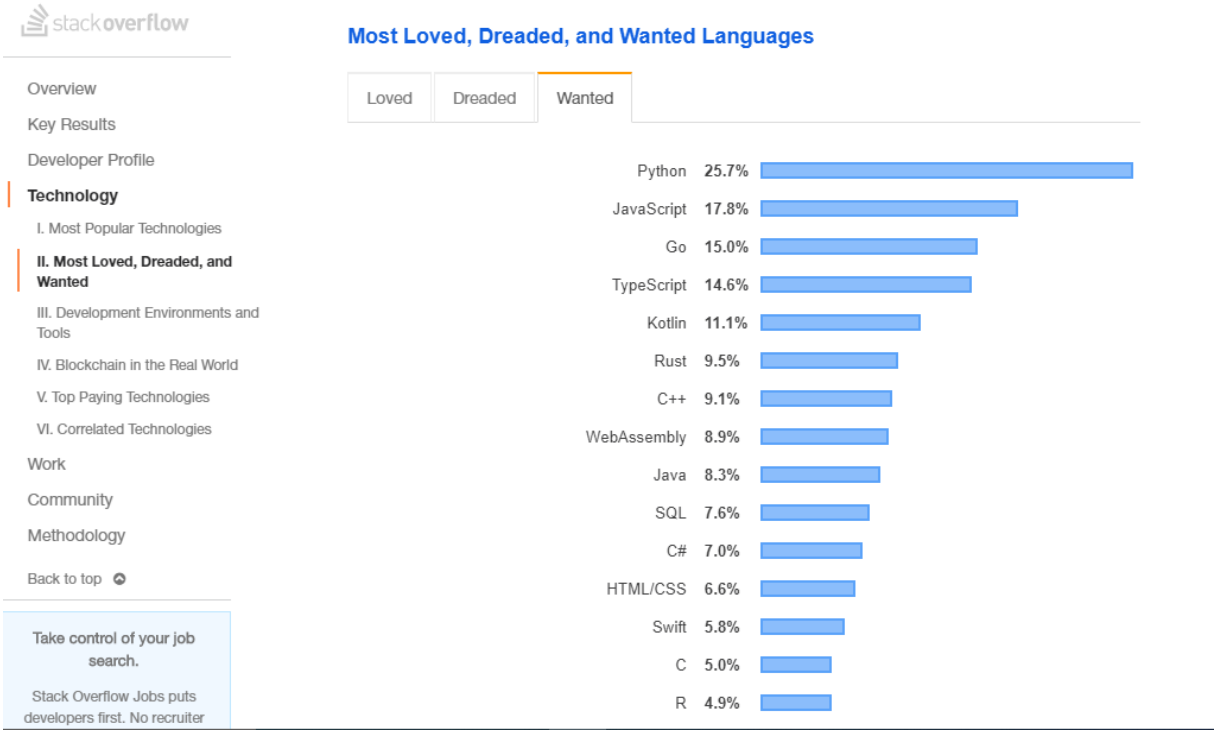


Figura 12 Resultados del lenguaje de programación más utilizado

Fuente: [61]

Los datos que se utilizaran se dividirá en un conjunto de datos de entrenamiento y en un conjunto de datos de prueba, el total del conjunto de datos que se utilizara es 33.968 los mismos que se encuentran divididos imparcialmente entre correos ham y spam. Para poder analizar el texto lo primero que se debe hacer es eliminar de las palabras que no aportan la información que se desea extraer, es decir todo aquello que esté relacionado con signos de puntuación, palabras, dígitos, etc. Luego de haber eliminado las palabras se procede a agrupar aquellas palabras que se pueden analizar como un único elemento. Ejemplo de Agrupar: llover, llueve, llovido, llovía se representará como llover.

Para poder crear el diccionario de palabras se utilizara un conjunto de capacitación de 33.968 de correos los mismo que serán de tipo ham y spam dicho diccionario se puede mejorar con más correos, este diccionario tiene algunas entradas las mismas que son las palabras más frecuentes en los correos.

Para realizar la estación de características el diccionario debe estar completamente listo

de esa manera de podrá extraer el vector de palabras de acuerdo con el tamaño del directorio seleccionado es decir por cada vector de conteo de palabras se obtendrá el tamaño del diccionario en el archivo de entrenamiento.

Por último, tenemos el entrenamiento de los clasificadores, el mismo que usa un modelo matemático el cual permite aprender un límite de decisión en cuanto al espacio de las características para ello se utilizara la librería sklearn la misma que permite entrenar los clasificadores, otra de las librerías utilizadas en el entrenamiento de los clasificadores es Naive Bayes ya que este permite la clasificación de documentos, así como también supone la independencia entre las características, para poder verificar el rendimiento del modelo en el conjunto de pruebas se deberá extraer el vector del conteo de palabras por cada correo del conjunto de pruebas y de esa manera se determinará si es ham o spam.

2.2.Explicación del desarrollo del filtro anti-spam

La propuesta para el desarrollo del filtro anti-spam consta de un algoritmo bayesiano el mismo que está centrado en el teorema de bayes y machine learning como ya se menciona anteriormente, para ello se hace uso de librerías ampliamente utilizadas en el ámbito de machine learning están son nltk, sklearn. NLTK es una librería que se usa para el análisis de texto, tokenización, derivación, facilitando la descarga de diferentes corpus (textos que contiene diferentes caracteres, signos, palabras, enlaces).

Sklearn se encarga de brindar soluciones simples y eficientes permitiendo el aprendizaje automático del diccionario construido con archivos de tipo spam (correos basura) y ham (correos no spam), haciendo que sea accesible y reutilizable en diferentes contextos.

El teorema de bayes se encarga de manejar un conjunto probabilidades para de esa manera poder calcular dicho conjunto obteniendo un resultado completamente corregido al que se presenta inicialmente, para este caso se hace uso de la ecuación que se muestra a continuación.

$$P(A_i \mid B) = \frac{P(B \mid A_i)\, P(A_i)}{P(B)}$$

Según [63] La ecuación del teorema de bayes se basa de un conjunto de probabilidades llamadas “a prioro” o también conocidas como sin corregir, calcula un conjunto de probabilidades “a posteori” o corregidas.

$P(A_i)$ son las probabilidades a priori

$P(B \mid A_i)$ es la probabilidad de B en la hipotesis A_i

$P\ (A_i \mid B)$ son las probabilidades a posteori.

2.3.Calculo con Teorema de Bayes

El diccionario previamente utilizado es un conjunto de entrenamiento y a su vez un conjunto de prueba el mismo que se encuentra dividido en un 60% de spam y un 40 % de ham con un total de archivos de 13.588 los mismo que se encuentran en 6 diferentes directorios para calcular la probabilidad total mediante el teorema antes mencionado.

Ham	**40%**
Spam	60%
Archivos	13.588
Directorios	6

Tabla 10 Porcentaje de spam, ham, número de archivos y directorios
Fuente: Autor

$$\Pr(A) = \frac{40}{100} \qquad \Pr(B) = \frac{60}{100}$$

$$\Pr(A) = 0{,}4 \qquad \Pr(B) = 0{,}6$$

$$\Pr(A|B)\frac{13.588}{6} = 2264.666$$

$$P = \frac{p1}{p1 + q1}$$

Calculo Probabilidad Spam

$$P(B) = \frac{1.358{,}4}{1.358{,}4 + 905{,}6} \qquad P(B) = 0{,}59$$

Calculo Probabilidad Ham

$$P(A) = \frac{905{,}6}{1.358{,}4 + 905{,}6} \qquad P(A) = 0{,}400$$

Al utilizar el teorema de bayes se tiene algunas ventajas puesto que la probabilidad de que el mensaje sea spam o no se la puede realizar a partir de un conjunto de archivos que utilizan como datos de entrenamiento, así como también a partir de un conjunto de palabas tomadas de cualquier mensaje, las estadísticas se obtienen a partir del conjunto incluido dentro del filtro desarrollado, a su vez permitirá clasificar constantemente el mensaje como spam o ham.

Etapa 3: Instalación y configuración de servidores, plataformas y librerías.

3.1.Instalación de la librería Nltk en Ubutnu16.04 python3.6

Para hacer uso de la librería Nltk en Python es necesario instalar para ellos se utiliza los siguientes comandos:

Los resultados de la instalación se encuentran en el Anexo1

Dentro del terminal de Linux se deberá ingresar:

pip install ntlk

Cuando haya terminado la instalación es necesario ingresar al entorno de Python para así poder importar la librería para utilizamos

python3.6

import nltk

Al momento que finalice la importación de la librería se procederá a descargar los complementos de esta para ello ingresamos

ntlk.download();

3.2.Instalación de la Liberia sklearn

Al igual que la librería anterior esta se instalará en el entorno Python3.6 en la plataforma de Ubuntu 16.04, para hacer uso de esta librería es necesario instalar con el siguiente comando:

pip install scikit-learn

El resultado de la instalación se encuentra en el Anexo 2

3.3.Instalación de Nube privada.

El manual completo de la instalación y configuración de la plataforma OpenNebula se encuentra en el Anexo 3.

Las configuraciones que se realizaron en la nube privada son:

- Configuración del Front-End y Nodo.
- Importación de imagen ISO de cada uno de los sistemas operativos.
- Creación de redes virtuales VNET.
- Creación de Templates.
- Ejecución de las instancias.

3.4.Instalación servidor de correo en Nube privada.

Los requisitos básicos para la instalación del servidor de correo:

- Revisar que resuelva el registro MX en la maquina a instalarse.
- Configuración de hosts.
- Descargar el servicio de Zimbra en la versión del sistema operativo de la maquina a instalarse.

- Ejecutar la instalación.
- Configuración del registro MX en la instalación de Zimbra.
- Configuración de contraseña del usuario admin.
- Levantar servicios de Zimbra.

3.5.Configuración de filtro antispam en servidor correo.

Se debe realizar las siguientes configuraciones en el servidor de correos para que pueda utilizar el filtro Anti-Spam que está en la nube privada de esa manera evitar el ingreso de correos spam en la infraestructura.

Primero debemos añadir la ruta donde está el script bash que nos permite ejecutar nuestro filtro Anti-Spam.

Filter: Este usuario maneja todo el contenido de correo.

Ruta: Se debe crear un directorio que sea accesible solo para el usuario "filter".

Pipe: Se debe configurar zimbra para que permita entregar los correos con el agente pipe(permite procesar las solicitudes del gestor de colas de zimbra para entregar mensajes a comandos externos) [64].

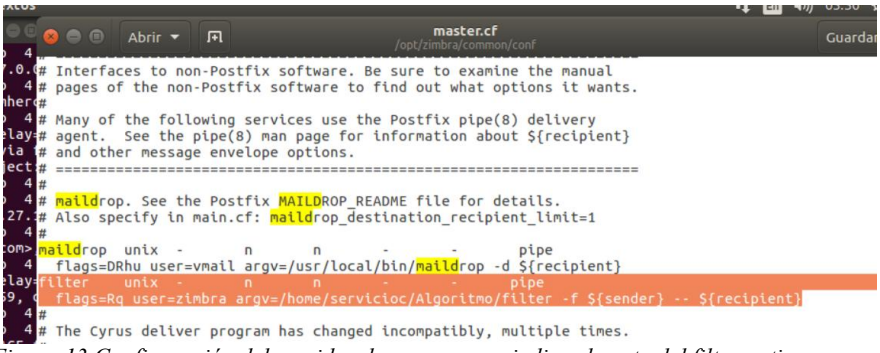


Figura 13 Configuración del servidor de correo para indicar la ruta del filtro anti-spam
Fuente: Autor

Además, se debe realizar las modificaciones en el puerto 10025 para habilitar el filtrado. Lo que hace filter:dummy hace que Zimbra agregue un registro de solicitud del script bash a cada mensaje de correo entrante, con el contenido "filter: dummy". Este registro anula el enrutamiento normal del correo y hace que se envíe script bash para que se ejecute el filtrado bayesiano [65].

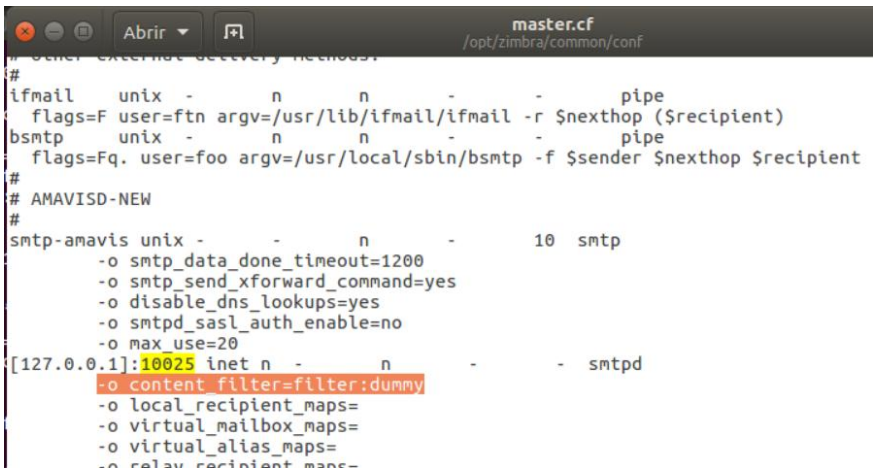


Figura 14 Habilitar filtrado de correo en servidor de correos
Fuente: Autor

Etapa 4: Pruebas del servidor de correo

En esta etapa se realizarán las pruebas de estimación con el fin de demostrar el funcionamiento y la efectividad del filtro Anti-Spam en la nube privada.

4.1.Infraestructura del escenario para pruebas servidor correo.

La infraestructura que se va a utilizar para realizar las pruebas del servidor de correo se compone de la siguiente manera como se muestra en la figura 7:

- Nube privada la cual es OpenNebula
- Máquina virtual con servidor correo y configurado filtro Anti-Spam en la nube privada
- Máquina virtual local con servidor de correo

Detalles de recurso de hardware de cada uno de los elementos que conforma la infraestructura.

Servicio	Sistema Operativo	RAM	Procesadores
Servidor Correo zimbra nube privada	Ubuntu 16.04	4	1
Servidor Correo zimbra local	Ubuntu 16.04 server	4	3

Tabla 11 Especificaciones de hardware a nivel de servidor de correo en la nube y local
Fuente: Autor

4.2.Entrenamiento del diccionario.

Antes de realizar cualquier prueba debemos cerciorarnos de que este realizado el respectivo entrenamiento del diccionario del filtro anti-spam para que de esa manera

pueda realizar el respectivo análisis del contenido de cada uno de los mensajes.

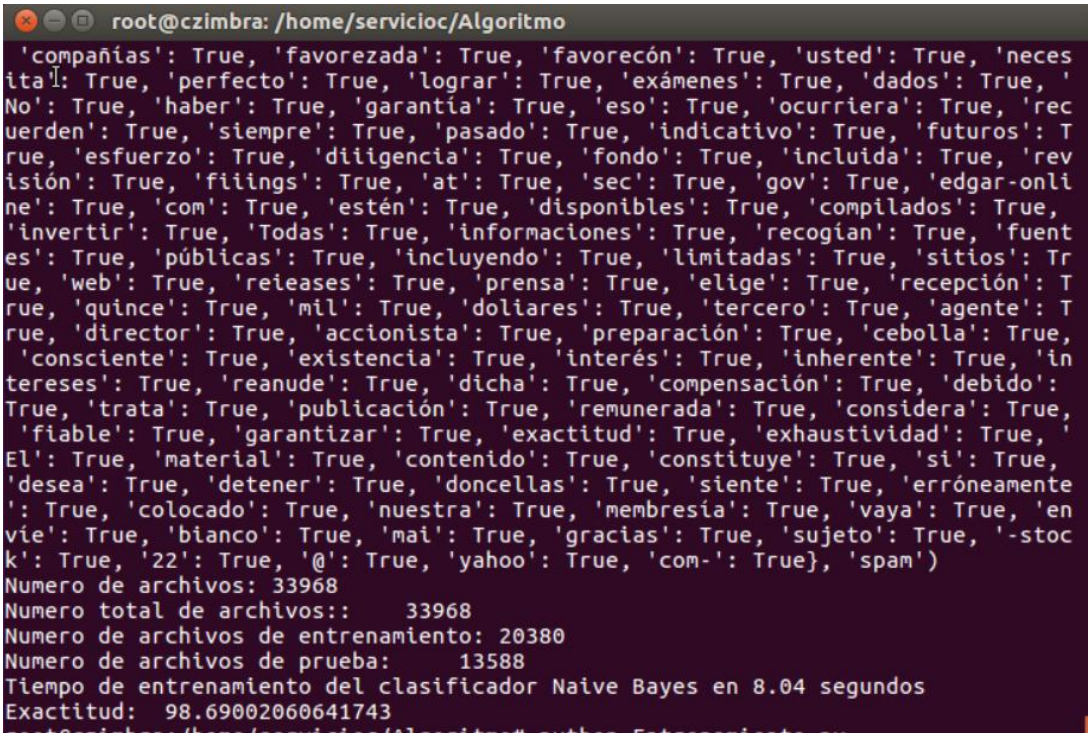


Figura 15 Entrenamiento del diccionario para el filtro anti-spam
Fuente: Autor.

4.3.Pruebas servidor de correo Zimbra.

Para realizar las pruebas con el servidor de correo, se va a utilizar la interfaz web que ofrece Zimbra para los clientes, la cual permite acceder desde cualquier navegador ingresando con la ip o el dominio, una vez que carga la página se debe ingresar con el usuario y contraseña de la cuenta, de esa manera se realizara él envió y la recepción de cada uno de los correos.

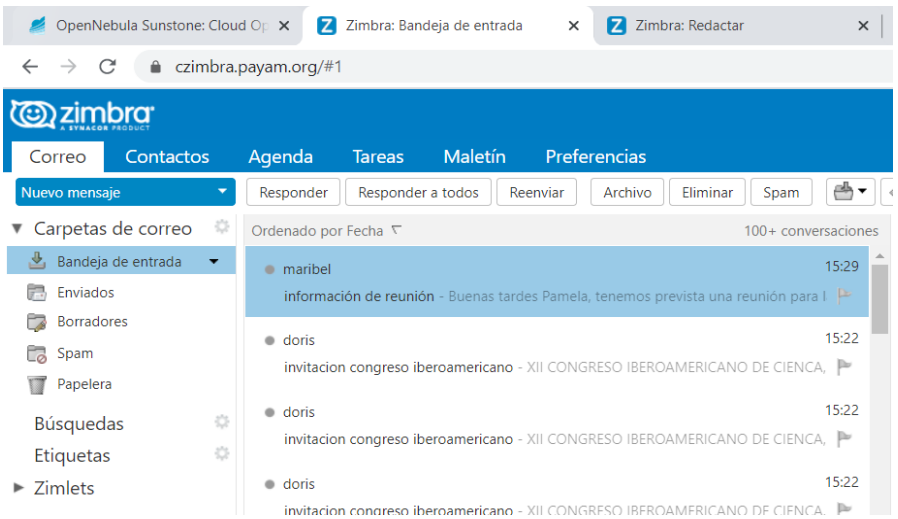


Figura 16 Interfaz gráfica de los clientes
Fuente: Autor.

4.4.Prueba envíos de mensajes.

Para realizar la prueba de envió de mensajes primero debemos ingresar a cada uno de los clientes de nuestros servidores de correo vamos a ingresar al cliente de nuestro servidor de la nube.

Clientes servidor de correo en la nube.

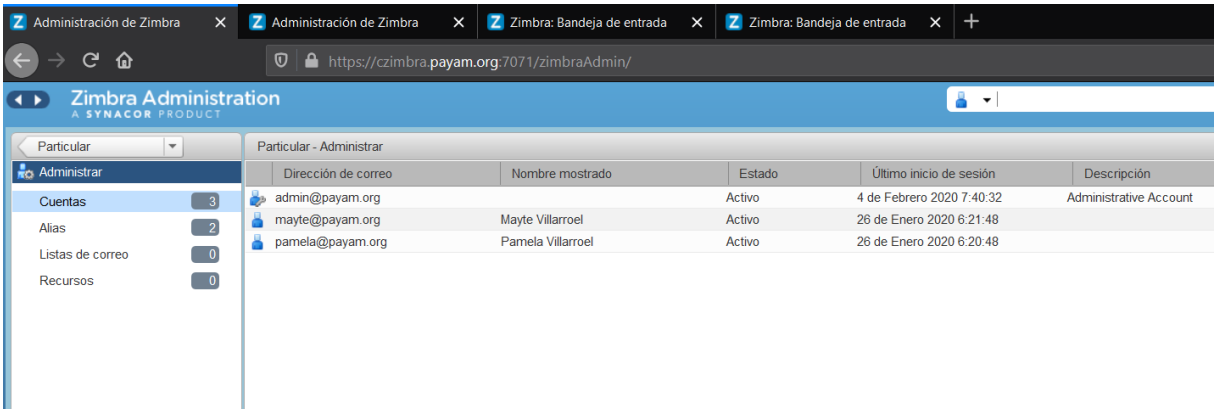


Figura 17 Clientes servidor de correo en la nube
Fuente: Autor

Clientes servidor de correo local.

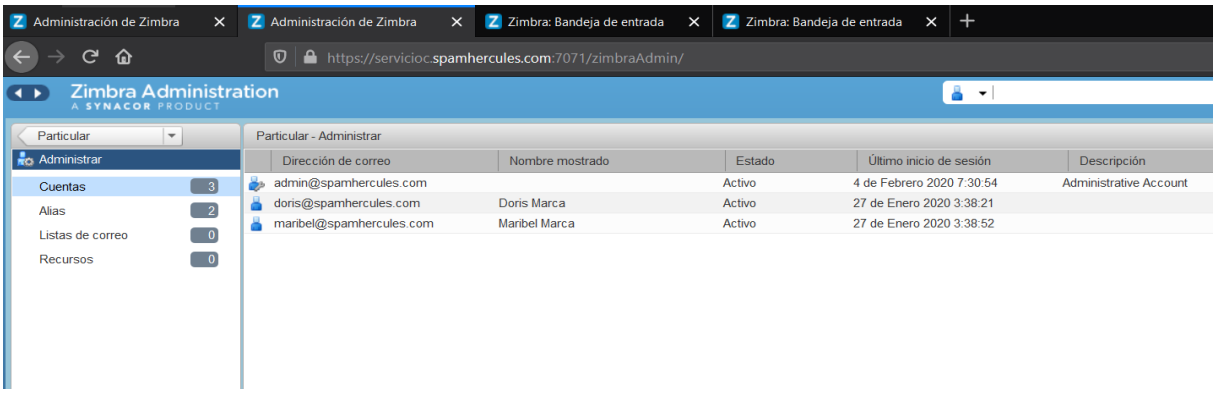


Figura 18 Clientes servidor de correos local
Fuente: Autor.

Ingreso en cliente del servidor de local y envío de mensaje.

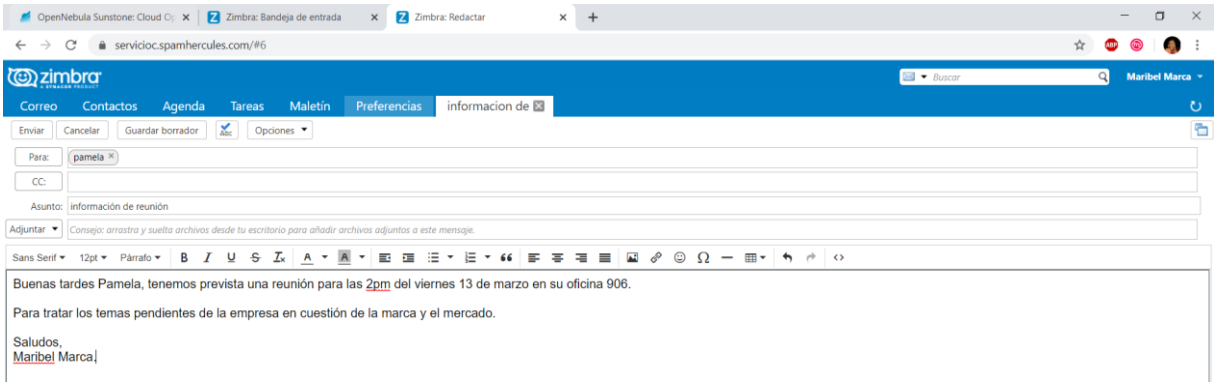


Figura 19 Envió de mensajes servidor de correo local
Fuente: Autor

En los logs del servidor de correo en la nube privada podemos visualizar que se activa el proceso de la cola y le llega el mensaje por su contenido.

```
Mar  9 15:29:52 czimbra postfix/qmgr[4961]: ED5381068D5: from=<maribel@spamhercu
les.com>, size=3357, nrcpt=1 (queue active)
Mar  9 15:29:53 czimbra postfix/lmtp[21844]: ED5381068D5: to=<pamela@payam.org>,
 relay=czimbra.payam.org[192.168.10.230]:7025, delay=0.16, delays=0.02/0.01/0.05
/0.07, dsn=2.1.5, status=sent (250 2.1.5 Delivery OK)
Mar  9 15:29:53 czimbra postfix/qmgr[4961]: ED5381068D5: removed
```

Figura 20 Revisión de logs de mensaje ham
Fuente: Autor

Gráfica generada con Python que representa el número mensajes enviados y el porcentaje de efectividad de un mensaje ham.

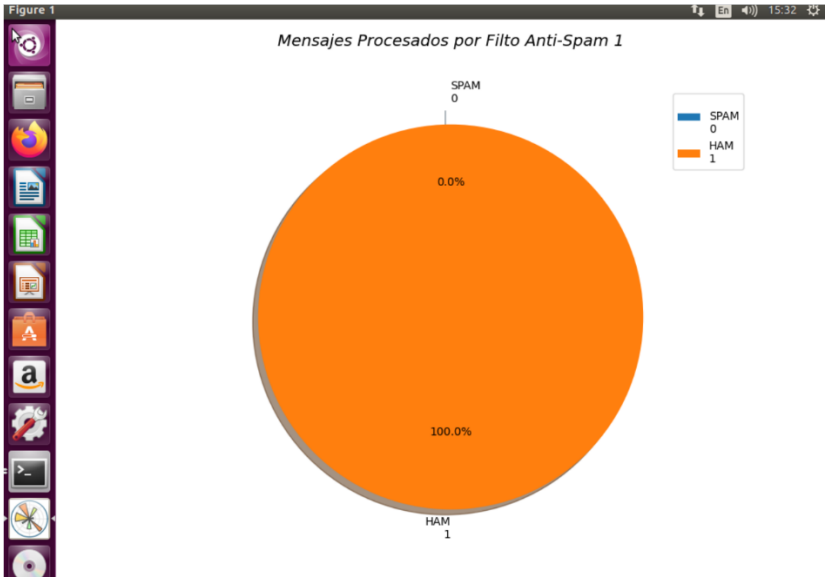


Figura 21Gráfica de porcentaje de efectividad con un mensaje ham
Fuente: Autor.

Prueba de envío de mensaje con contenido spam desde el servidor de correo local al servidor de correo que está en la nube privada.

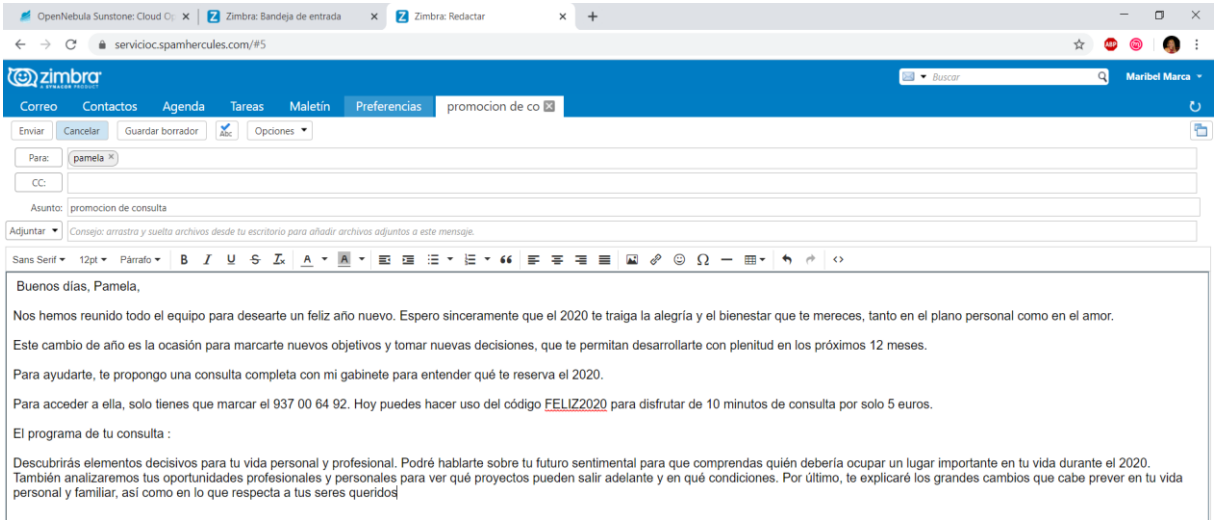


Figura 22 Prueba de envío de mensaje con contenido spam
Fuente: Autor

Revisión de logs del servidor de correo que está en la nube privada como se puede visualizar empieza a realizar análisis por ende manda el mensaje a la cola y una vez que se realiza la respectiva verificación por el filtrado bayesiano y no llega a la bandeja de entrada del usuario.

```
Mar  9 15:24:27 czimbra postfix/qmgr[4961]: 87718106847: from=<maribel@spamhercu
les.com>, size=5389, nrcpt=1 (queue active)
Mar  9 15:24:27 czimbra postfix/amavisd/smtpd[20110]: disconnect from localhost[
127.0.0.1] ehlo=2 starttls=1 mail=1 rcpt=1 data=1 quit=1 commands=7
Mar  9 15:24:27 czimbra postfix/qmgr[4961]: 583751068D1: removed
Mar  9 15:24:31 czimbra postfix/pipe[20111]: 87718106847: to=<pamela@payam.org>,
 relay=filter, delay=4.4, delays=0.01/0/0/4.4, dsn=2.0.0, status=sent (delivered
 via filter service (ingreso 1 ingreso 2 ingreso 3 ingreso 4.2 ingreso 6 ssss co
ntent  102 aux  7174 cont_validar  102 co))
Mar  9 15:24:31 czimbra postfix/qmgr[4961]: 87718106847: removed
```

Figura 23 Revisión de logs de un mensaje spam
Fuente: Autor.

Gráfica generada con Python que representa el número mensajes enviados y el porcentaje de efectividad de un mensaje spam.

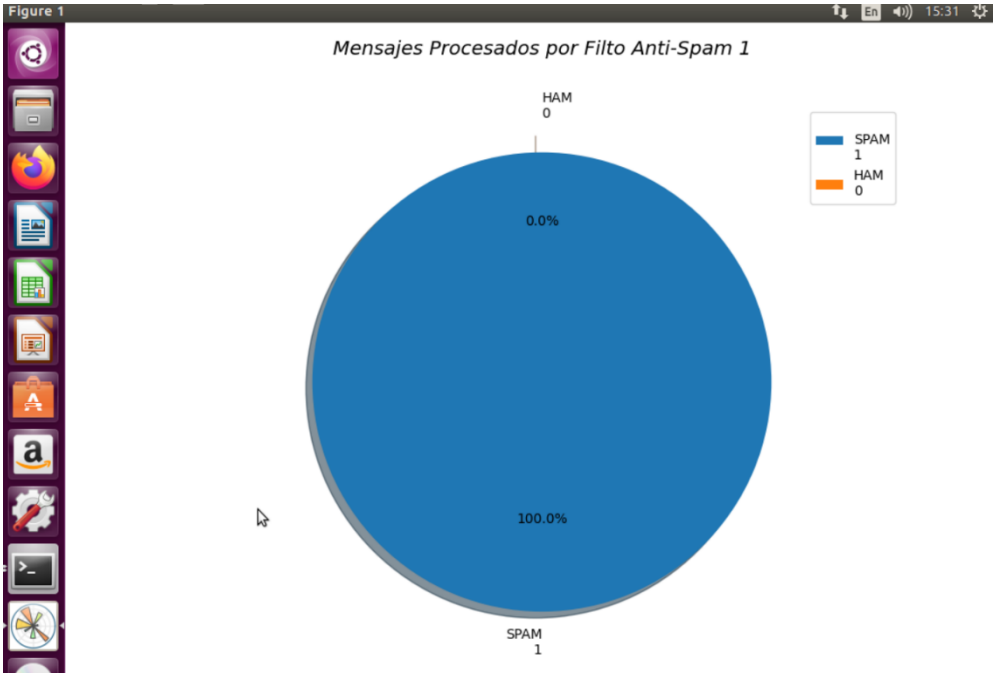


Figura 24 Gráfica de porcentaje de efectividad con un mensaje spam
Fuente: Autor

Capítulo 4: Pruebas con envío de correos masivos y análisis de resultados.

La herramienta multimail permite enviar correos de forma masiva, ingresando la dirección origen y destino, la dirección smtp del servidor, el número de mensajes y la ruta del archivo en el que está el mensaje.

Dentro de las pruebas realizadas, se envió los correos con diferentes cantidades los mismos

que se muestran a continuación

Prueba de envío de 5000 mensajes spam.

- Contenido del archivo que posee el mensaje.

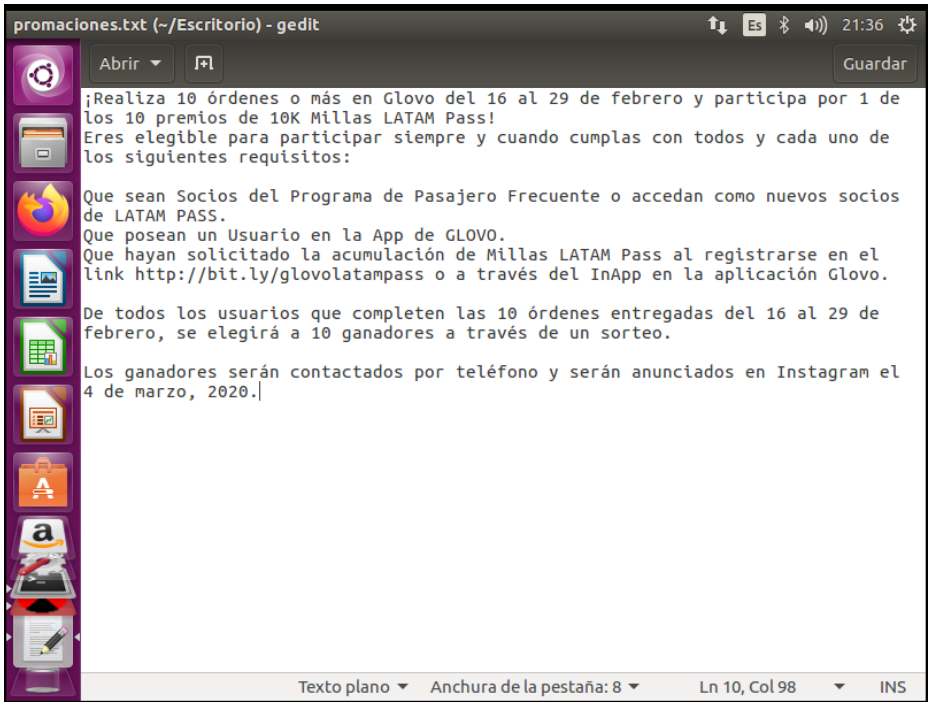


Figura 25 Archivo que posee el contenido del mensaje spam
Fuente: Autor.

- Envió de 5000 mensajes spam con la herramienta MultiMail.

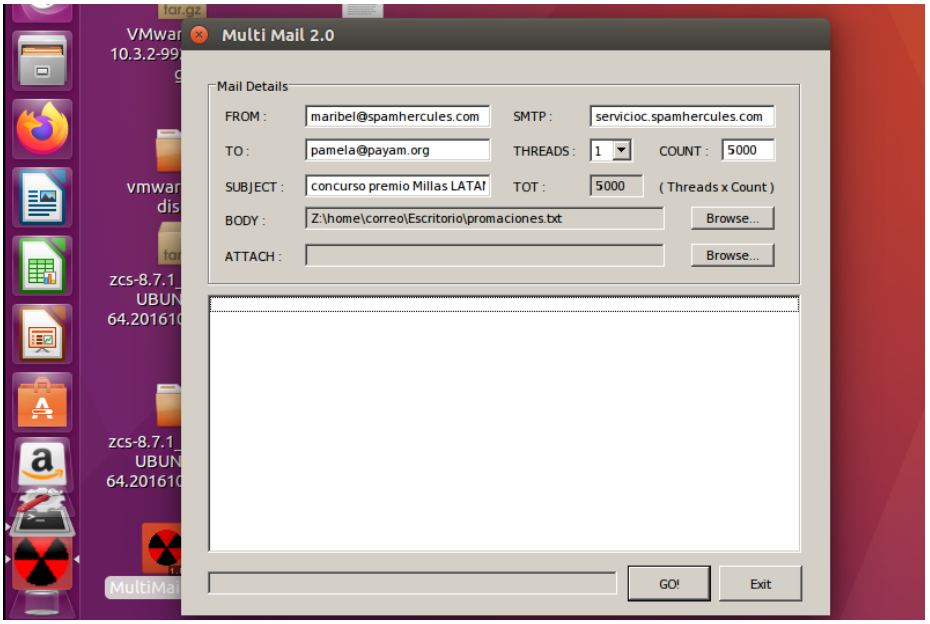


Figura 26 Prueba mensaje masivo con 5000 mensajes spam
Fuente: Autor

- Bandeja de entrada del cliente en la que recibe 5 mensajes.

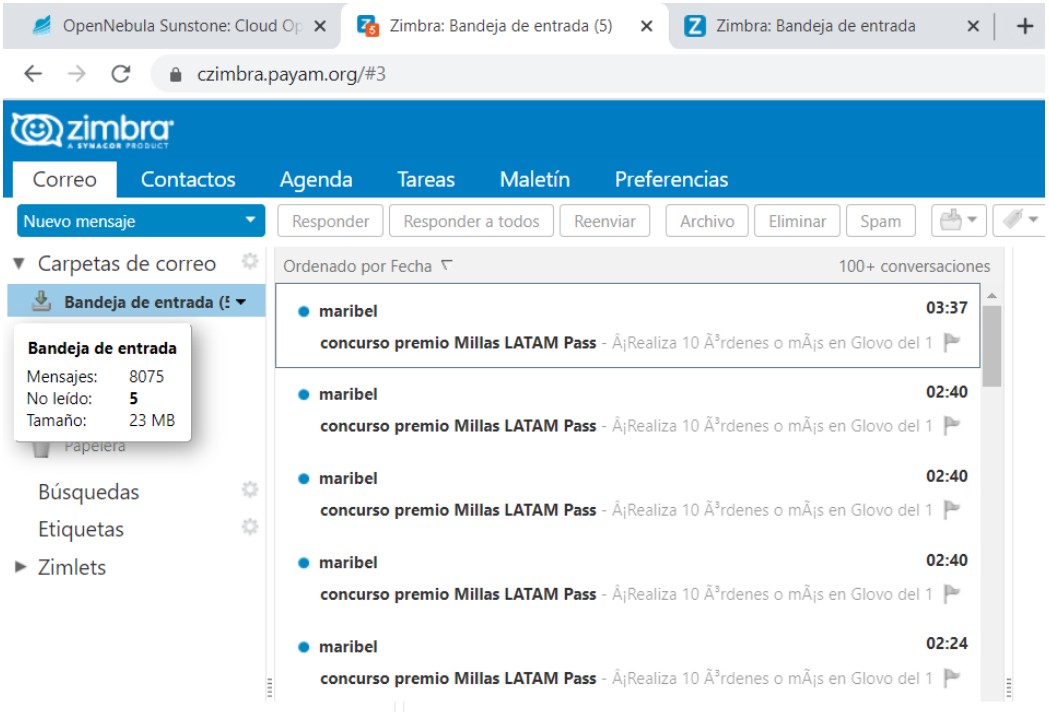


Figura 27 Bandeja de entrada cliente que recibe 5 mensajes spam
Fuente: Autor.

- Gráfica generada con Python que representa el número mensajes enviados y el porcentaje de efectividad de 5000 mensajes spam.

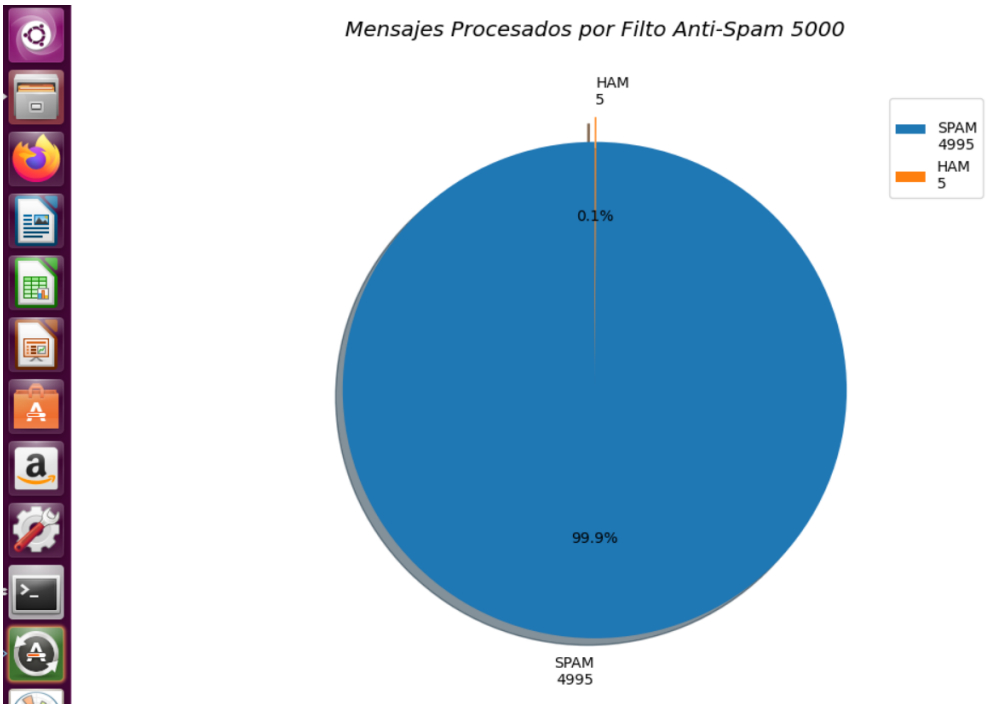


Figura 28 Gráfica de porcentaje de efectividad con él envió de 5000 mensajes spam
Fuente: Autor.

El comportamiento del filtro anti-spam con una prueba de 5000 mensajes spam es positiva ya que se obtiene una efectividad del 99.9%, del envío masivo 5 mensajes fueron

entregados a las bandeja del cliente lo que representa un 0.1% por lo que se estaría cumpliendo con la tasa efectiva mencionada anteriormente.

Prueba de envío de 15000 mensajes spam.

- Contenido del archivo que posee el mensaje.

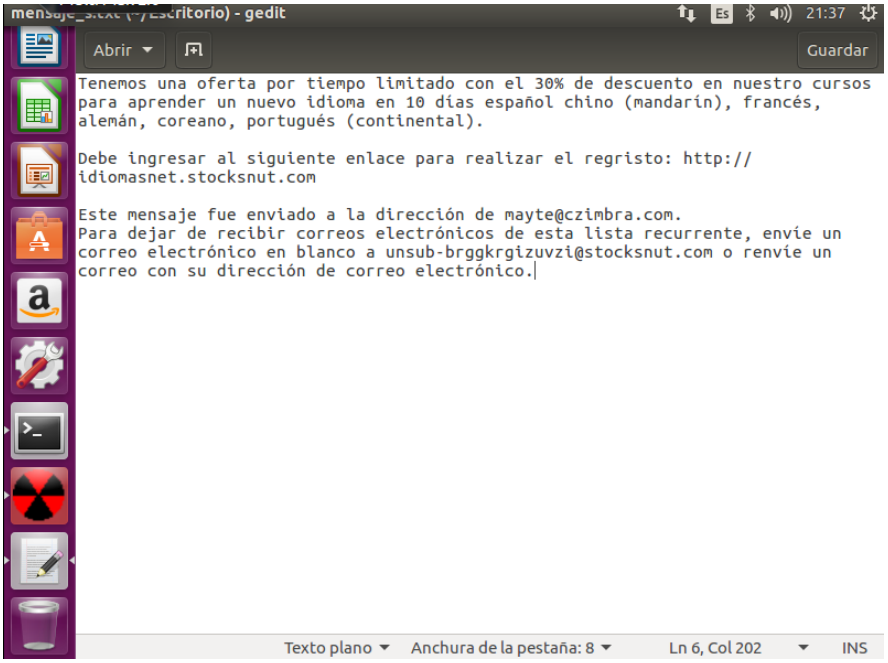


Figura 29 Archivo que posee el contenido del mensaje spam para el envío de 15000 mensajes
Fuente: Autor

- Envió de 15000 mensajes spam con la herramienta MultiMail.

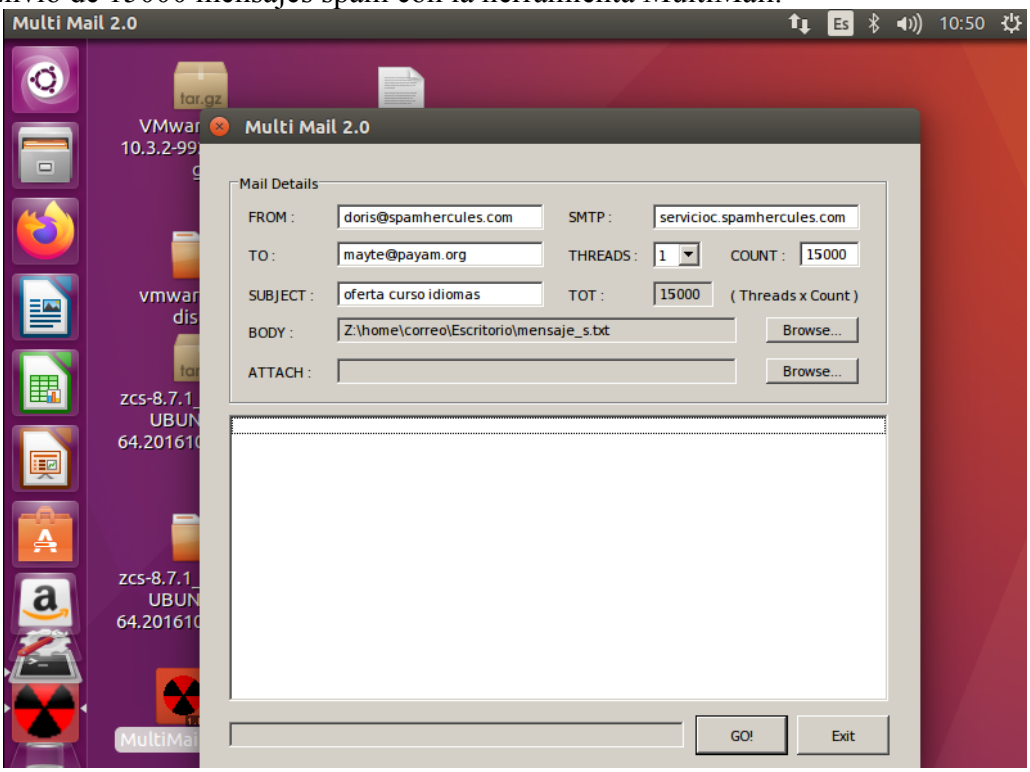


Figura 30 Prueba mensaje masivo con 15000 mensajes spam
Fuente: Autor.

- Bandeja de entrada del cliente en la que recibe 4 mensajes.

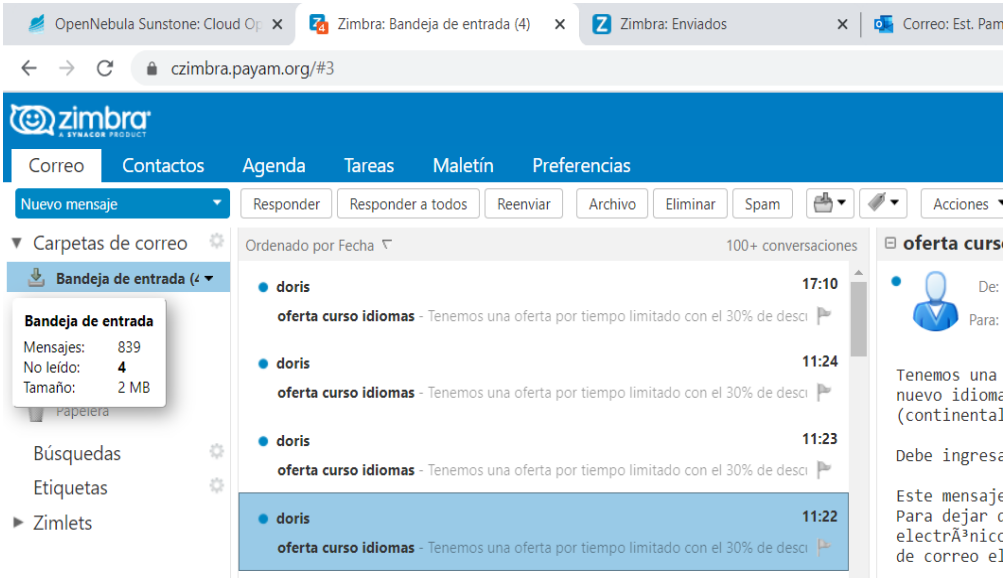


Figura 31 Bandeja de entrada cliente que recibe 4 mensajes spam
Fuente: Autor

- Gráfica generada con Python que representa el número mensajes enviados y el porcentaje de efectividad de 15000 mensajes spam.

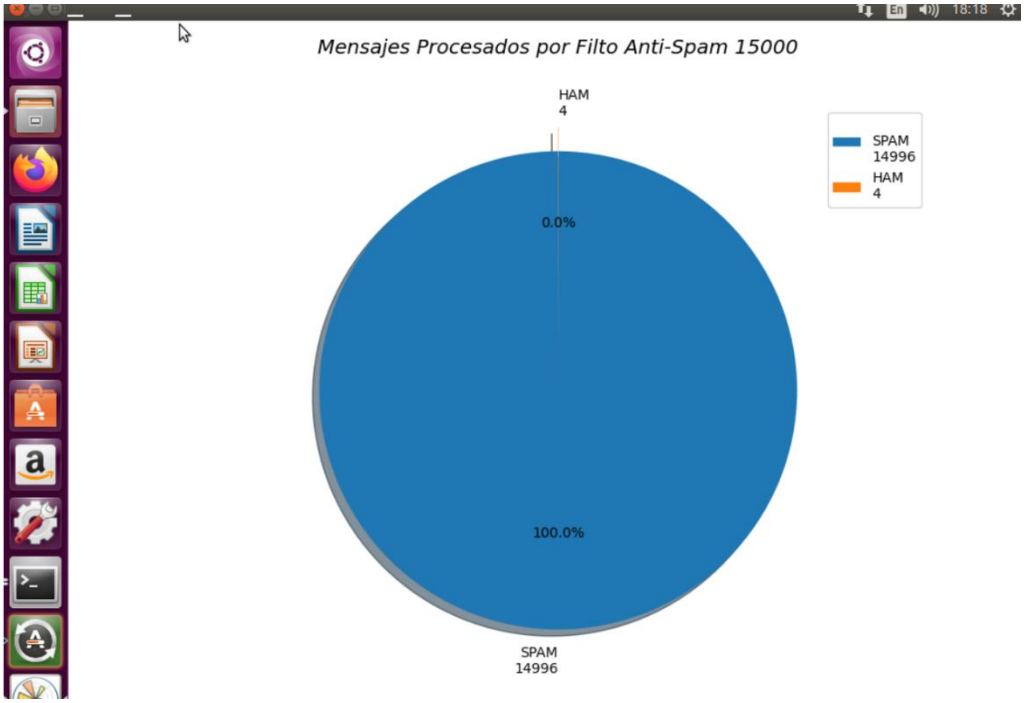


Figura 32 Gráfica de porcentaje de efectividad con él envió de 15000 mensajes spam
Fuente: Autor.

En cambio, cuando se realizó una prueba con 15.000 mensajes se obtuvo el 100% de efectividad a pesar de que 4 mensajes se entreguen en la bandeja del cliente ya que es una cantidad menor en representación a la cantidad de mensajes que se enviaron no generan nada de porcentaje en mensajes ham.

Prueba de envío de 15000 mensajes spam.

- Contenido del archivo que posee el mensaje.

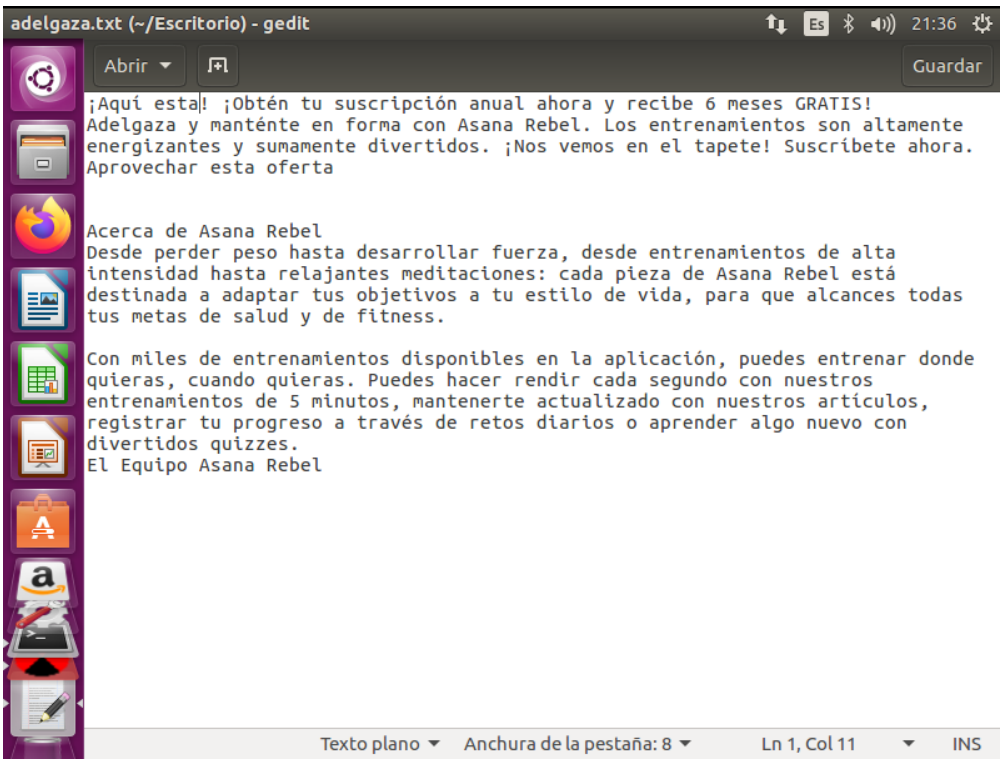


Figura 33 Archivo que posee el contenido del mensaje spam para el envío de 30000 mensajes
Fuente: Autor.

- Envió de 30000 mensajes spam con la herramienta MultiMail.

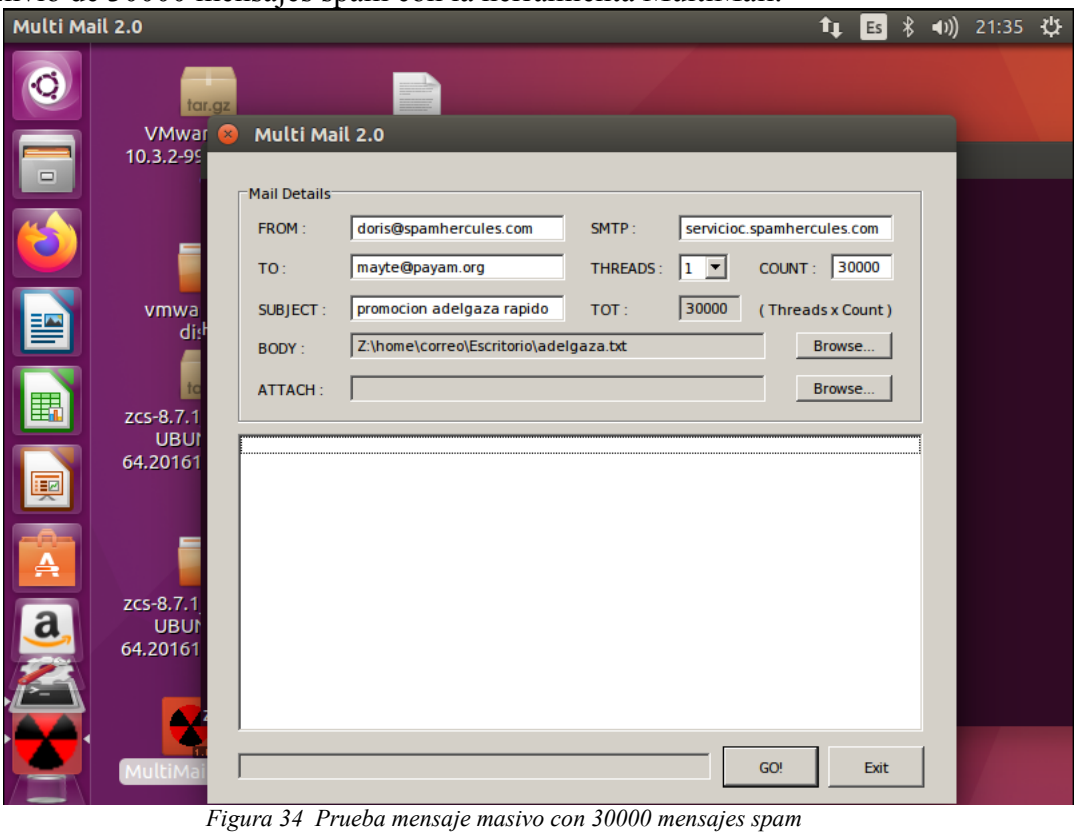


Figura 34 Prueba mensaje masivo con 30000 mensajes spam
Fuente: Autor

- Bandeja de entrada del cliente en la que recibe 11 mensajes.

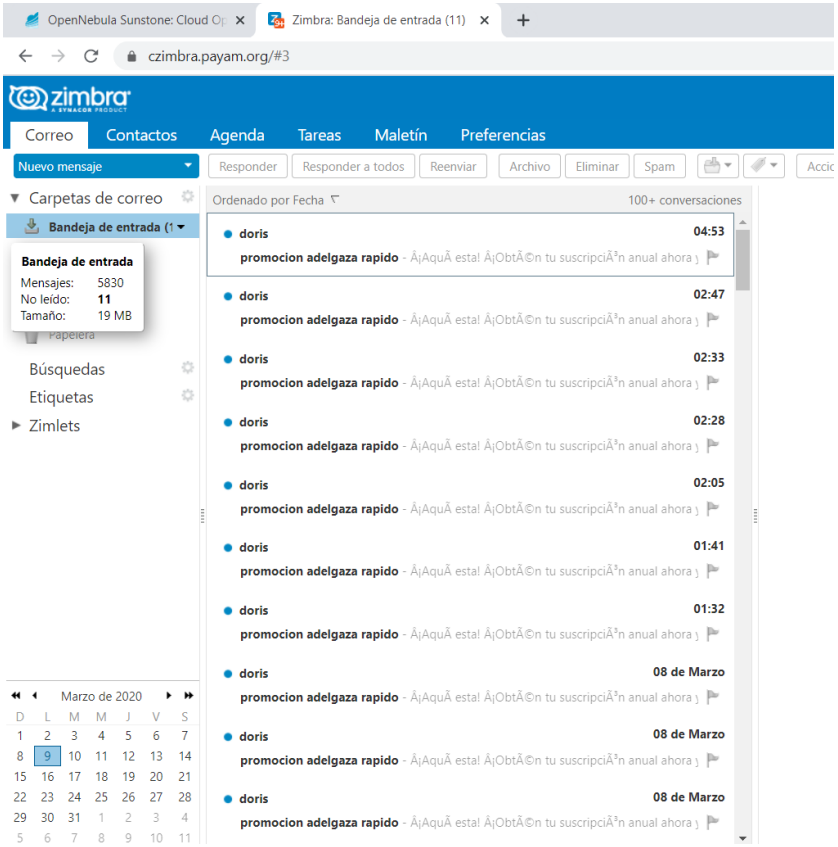


Figura 35 Bandeja de entrada cliente que recibe 11 mensajes spam
Fuente: Autor

- Gráfica generada con Python que representa el número mensajes enviados y el porcentaje de efectividad de 30.000 mensajes spam.

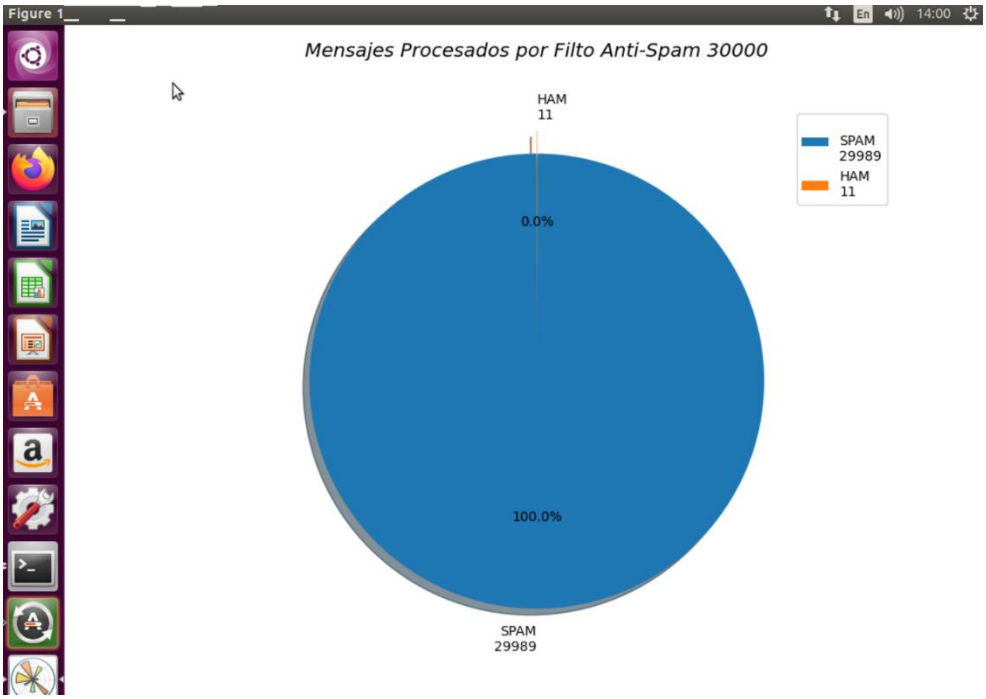


Figura 36 Gráfica de porcentaje de efectividad con él envió de 30000 mensajes spam
Fuente: Autor

Finalmente, al realizar la prueba con 30.000 mensajes spam igualmente se obtuvo una tasa

de efectividad del 100% a pesar de que se entregaron 11 mensajes a la bandeja de entrada del cliente, lo que representa un número considerable de mensajes entregados a la bandeja a pesar del número de correos enviados.

Prueba falsos positivos con 5000 mensajes.

Para este tipo de pruebas no basamos en la facturación electrónica ya que por su contenido en la mayoría de las veces los filtros anti-spam mandan los mensajes a la bandeja de spam.

- Contenido del archivo que posee el mensaje.

Figura 37 Archivo que posee el contenido del mensaje de facturación
Fuente: Autor

- Envió de 5000 mensajes ham con la herramienta MultiMail.

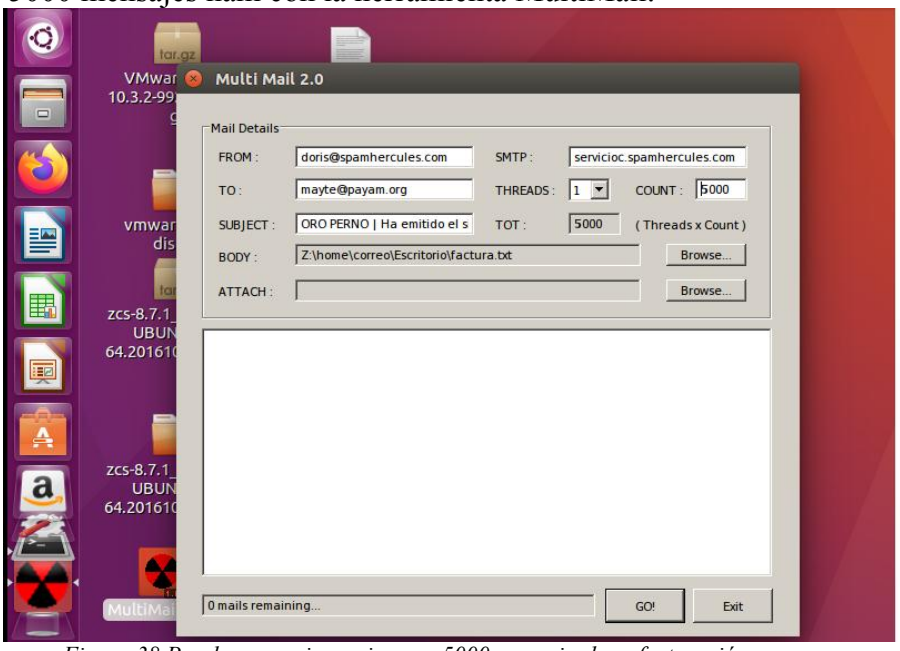


Figura 38 Prueba mensaje masivo con 5000 mensajes ham facturación
Fuente: Autor

- Bandeja de entrada del cliente en la que recibe 64 mensajes.

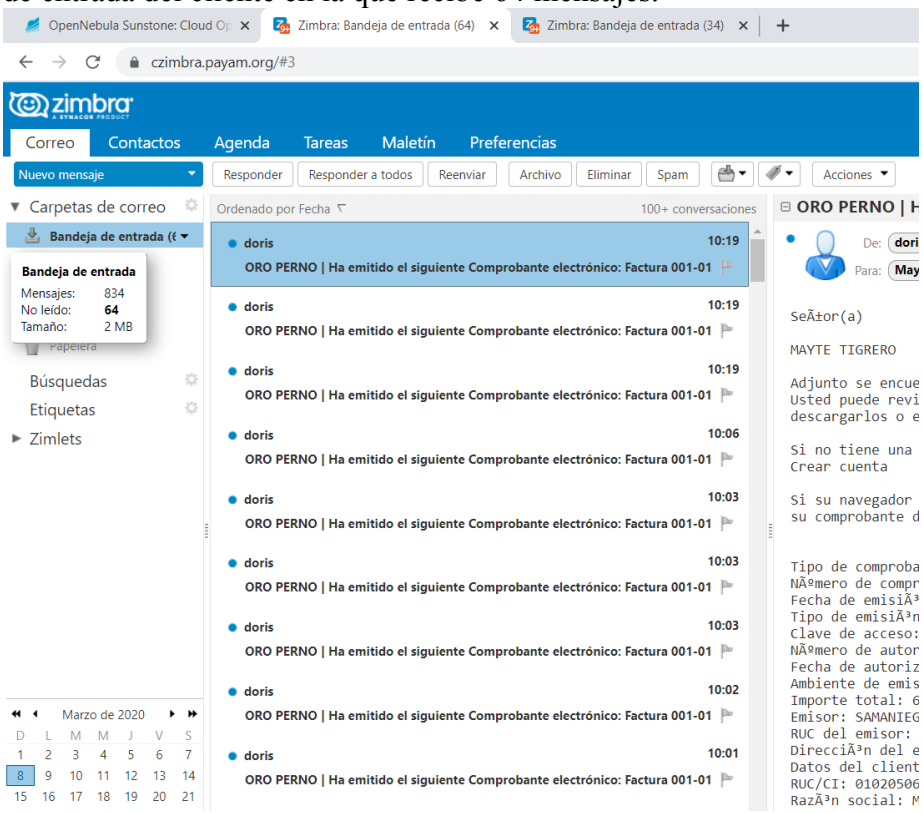


Figura 39 Bandeja de entrada cliente que recibe 64 mensajes ham
Fuente: Autor

- Gráfica generada con Python que representa el número mensajes enviados y el porcentaje de efectividad de 5000 mensajes ham.

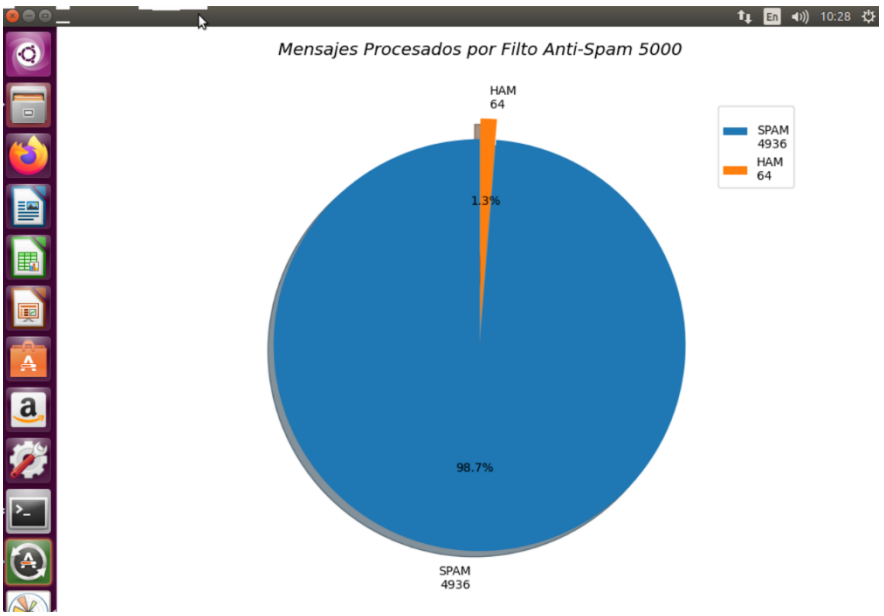


Figura 40 Gráfica de porcentaje de efectividad con él envió de 5000 mensajes ham facturación
Fuente: Autor

Para este tipo de pruebas obtenemos una tasa de error del 98.7% ya que de los 5.000 mensajes de la facturación electrónica solo 64 fueron entregados en la bandeja de entrada y los 4936 son spam según el análisis que realizo el filtro, pero se debe tener en cuenta que

este análisis se los realiza en base a la forma en la que el diccionario está formado, así que ahora vamos la misma prueba pero ahora el filtro esta mejor alimentado con este tipo de contenido y los resultados son diferentes.

- Envió de 5000 mensajes ham con la herramienta MultiMail.

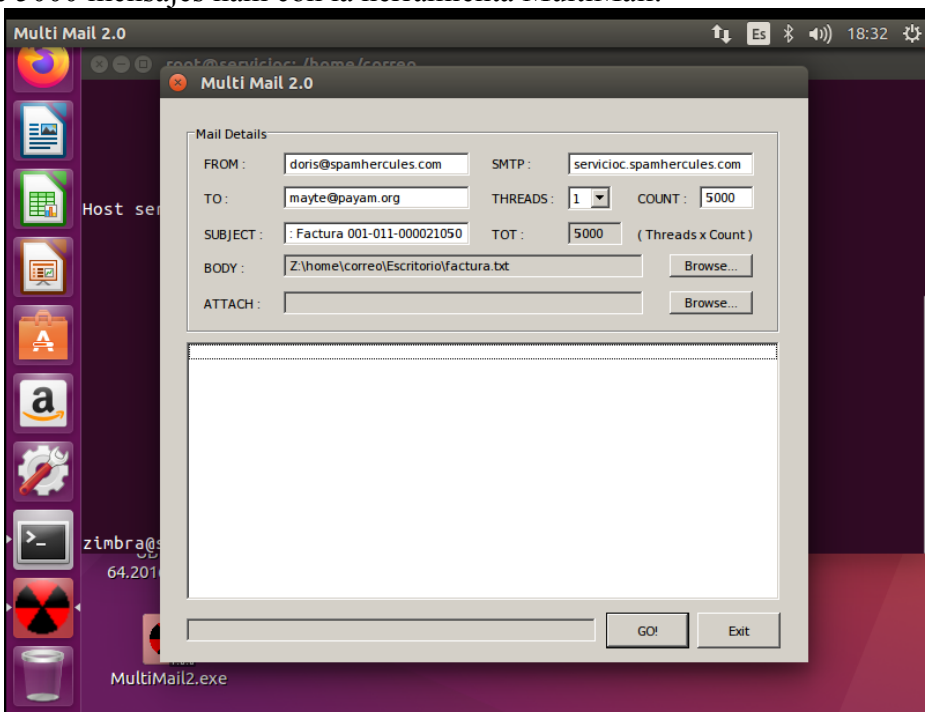


Figura 41 Nueva prueba de envío de mensajes masivo con 5000 mensajes de facturación
Fuente: Autor.

- Bandeja de entrada del cliente en la que recibe 4980 mensajes.

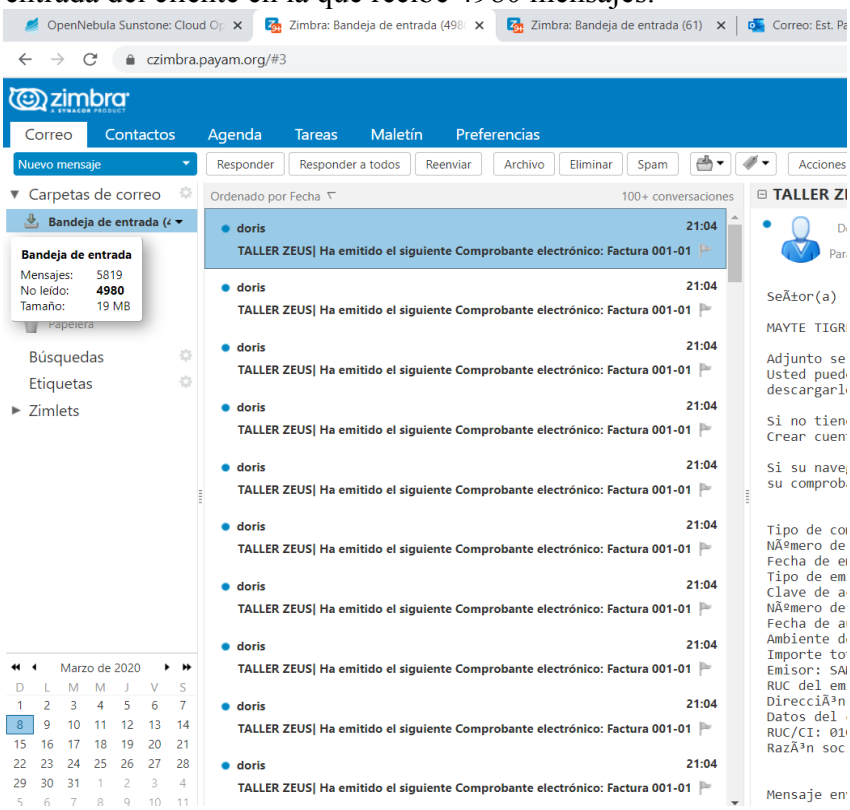


Figura 42 Bandeja de entrada cliente que recibe 4980 mensajes con contenido de facturación
Fuente: Autor.

- Gráfica generada con Python que representa el número mensajes enviados y el porcentaje de efectividad de 5000 mensajes con contenido de facturación.

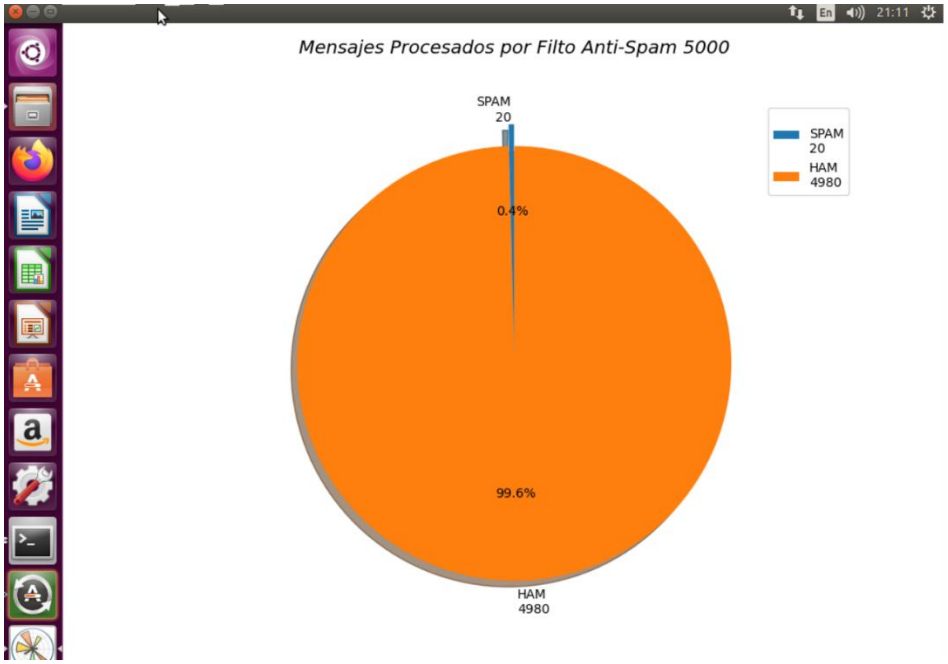


Figura 43 Gráfica de porcentaje de efectividad con él envió de 5000 mensajes con contenido de facturación
Fuente: Autor.

Al realizar nuevamente la prueba de envío masivo con 5.000 mensajes con contenido de facturación se obtiene una tasa positiva del 99.6% de efectividad ya que como se mencionó anteriormente con el correcto entrenamiento del entrenamiento se pueden lograr resultados positivos del filtro anti-spam.

Prueba de envío de 1000 mensajes ham.

- Contenido del archivo que posee el mensaje.

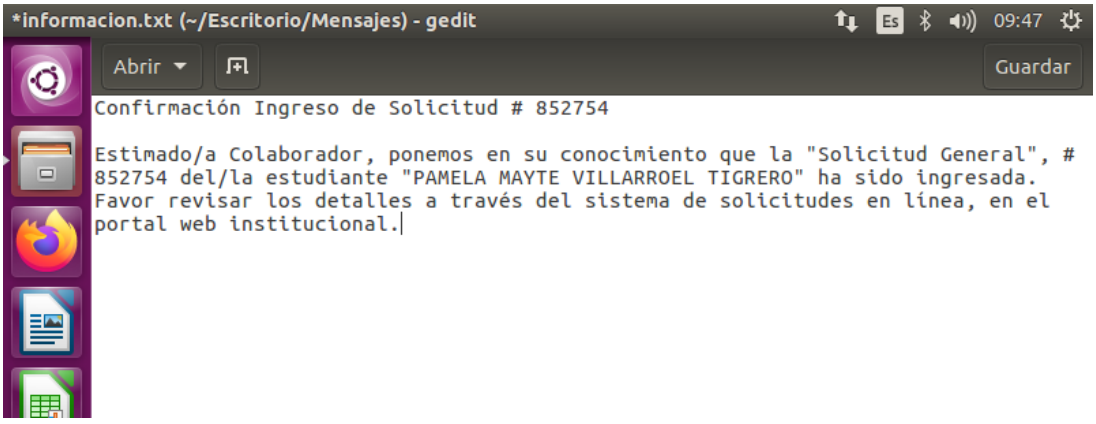


Figura 44 Archivo que posee el contenido del mensaje ham para el envío de 1000 mensajes
Fuente: Autor

- Envió de 1000 mensajes ham con la herramienta MultiMail.

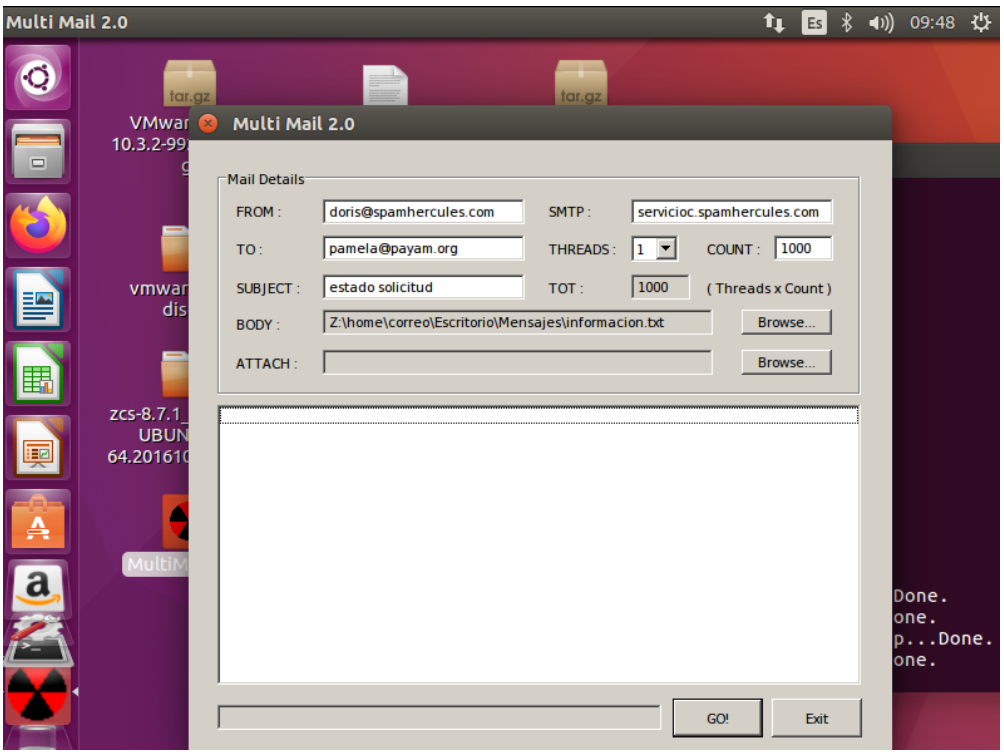


Figura 45 Prueba mensaje masivo con 1000 mensajes ham
Fuente: Autor

- Bandeja de entrada del cliente en la que recibe 999 mensajes.

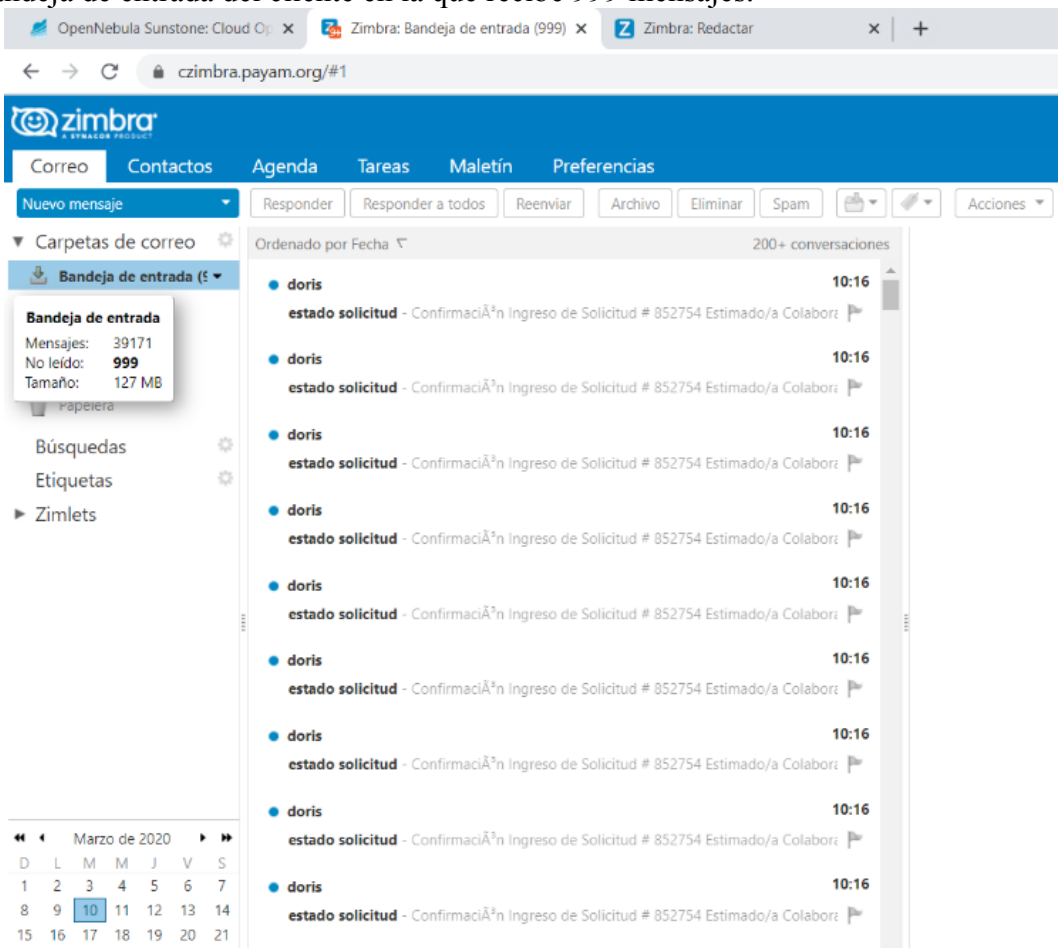


Figura 46 Bandeja de entrada cliente que recibe 999 mensajes ham
Fuente: Autor

- Gráfica generada con Python que representa el número mensajes enviados y el porcentaje de efectividad de 1000 mensajes ham.

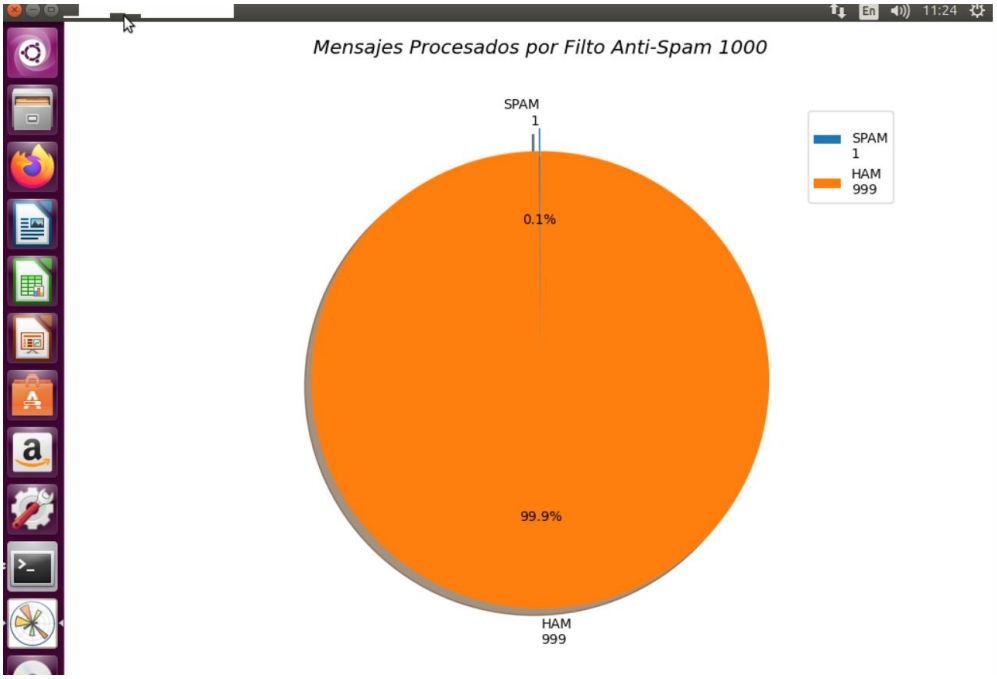


Figura 47 Gráfica de porcentaje de efectividad con él envió de 1000 mensajes ham
Fuente: Autor

Al momento de realizar una pequeña prueba con 1.000 mensajes de tipo ham se obtuvo una efectividad del filtro anti-spam del 99.9% lo que nos da como resultado un valor positivo y un mensaje se perdió en la cola de spam lo que representa el 0.1%.

Prueba de envío de 5000 mensajes ham.

- Contenido del archivo que posee el mensaje.

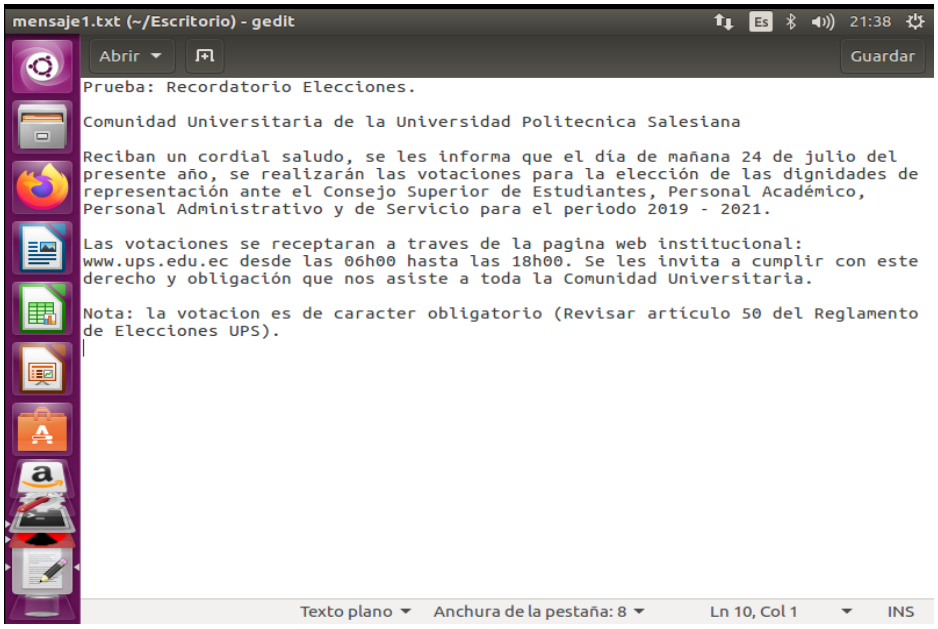


Figura 48 Archivo que posee el contenido del mensaje ham
Fuente: Autor.

- Envió de 5000 mensajes ham con la herramienta MultiMail.

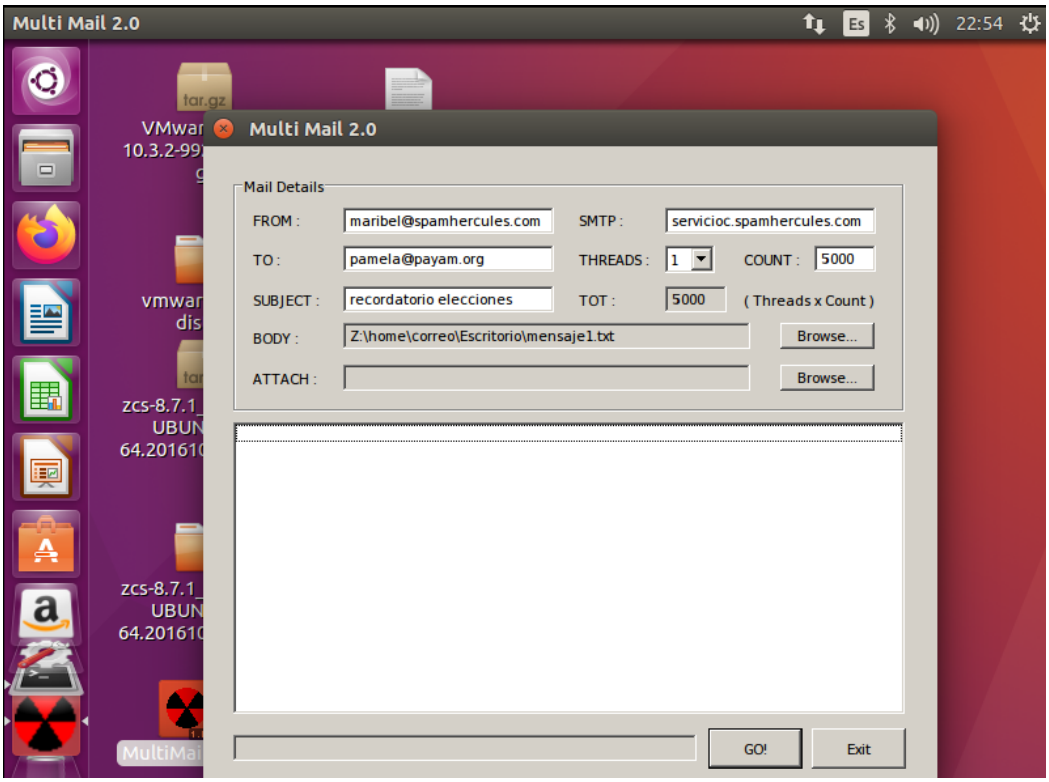


Figura 49 Prueba mensaje masivo con 5000 mensajes ham
Fuente: Autor

- Bandeja de entrada del cliente en la que recibe 4993 mensajes.

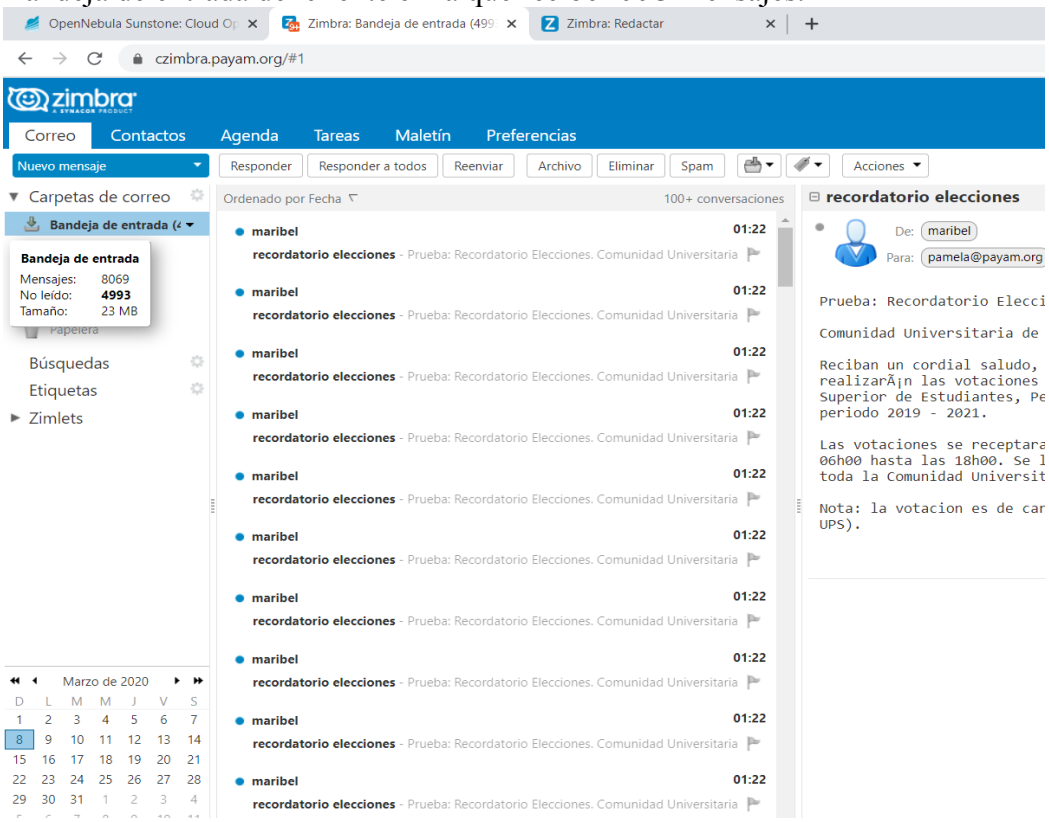


Figura 50 Bandeja de entrada cliente que recibe 4993 mensajes ham
Fuente: Autor.

- Gráfica generada con Python que representa el número mensajes enviados y el porcentaje de efectividad de 5000 mensajes ham.

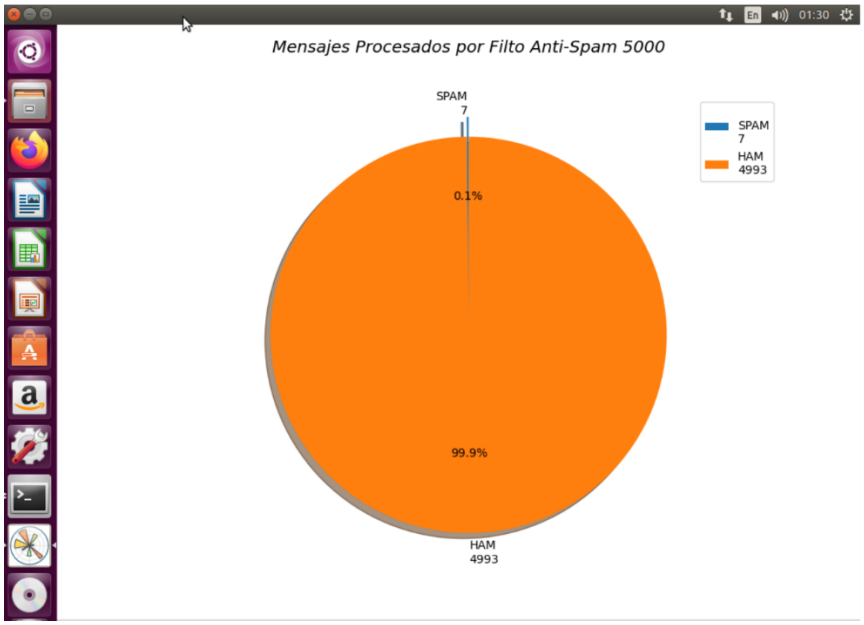


Figura 51 Gráfica de porcentaje de efectividad con él envió de 5000 mensajes ham
Fuente: Autor.

Al momento de realizar el envío de 5000 mensajes ham se obtiene una tasa efectividad del 99.9 % que está dentro de nuestro rango de exactitud que tiene el filtro anti-spam que es del 98.7%, ya que se entregaron a la bandeja 4993 mensajes y 7 mensajes se fueron a spam los cuales no fueron entregados, por lo que se esta prueba seria efectiva según los parámetros establecidos por nuestro filtro.

Prueba de envío de 30000 mensajes ham.

- Contenido del archivo que posee el mensaje.

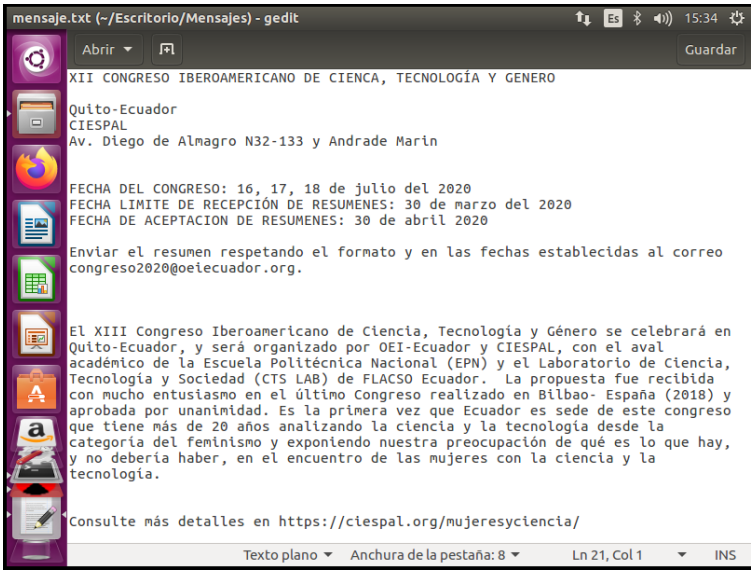


Figura 52 Archivo que posee el contenido del mensaje ham para el envío de 30000 mensajes
Fuente: Autor.

- Envió de 30000 mensajes ham con la herramienta MultiMail

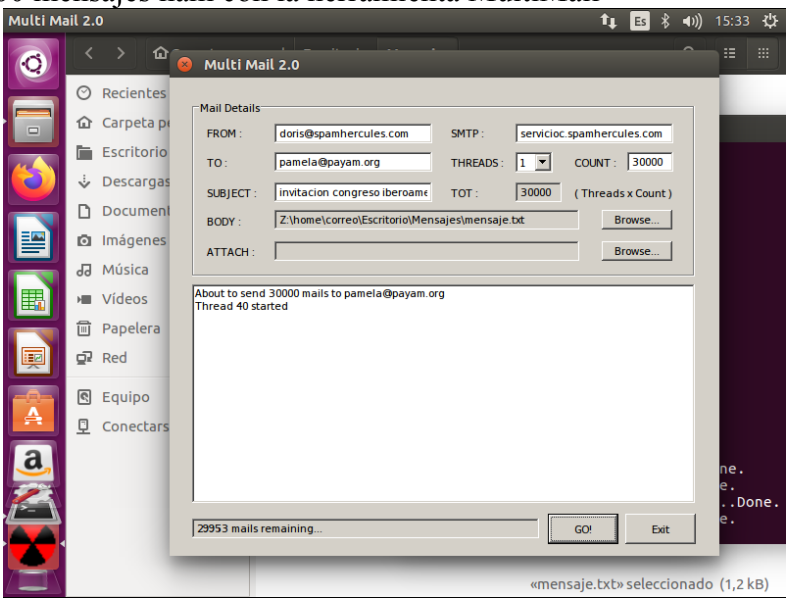


Figura 53 Prueba mensaje masivo con 30000 mensajes ham
Fuente: Autor

- Bandeja de entrada del cliente en la que recibe mensajes.

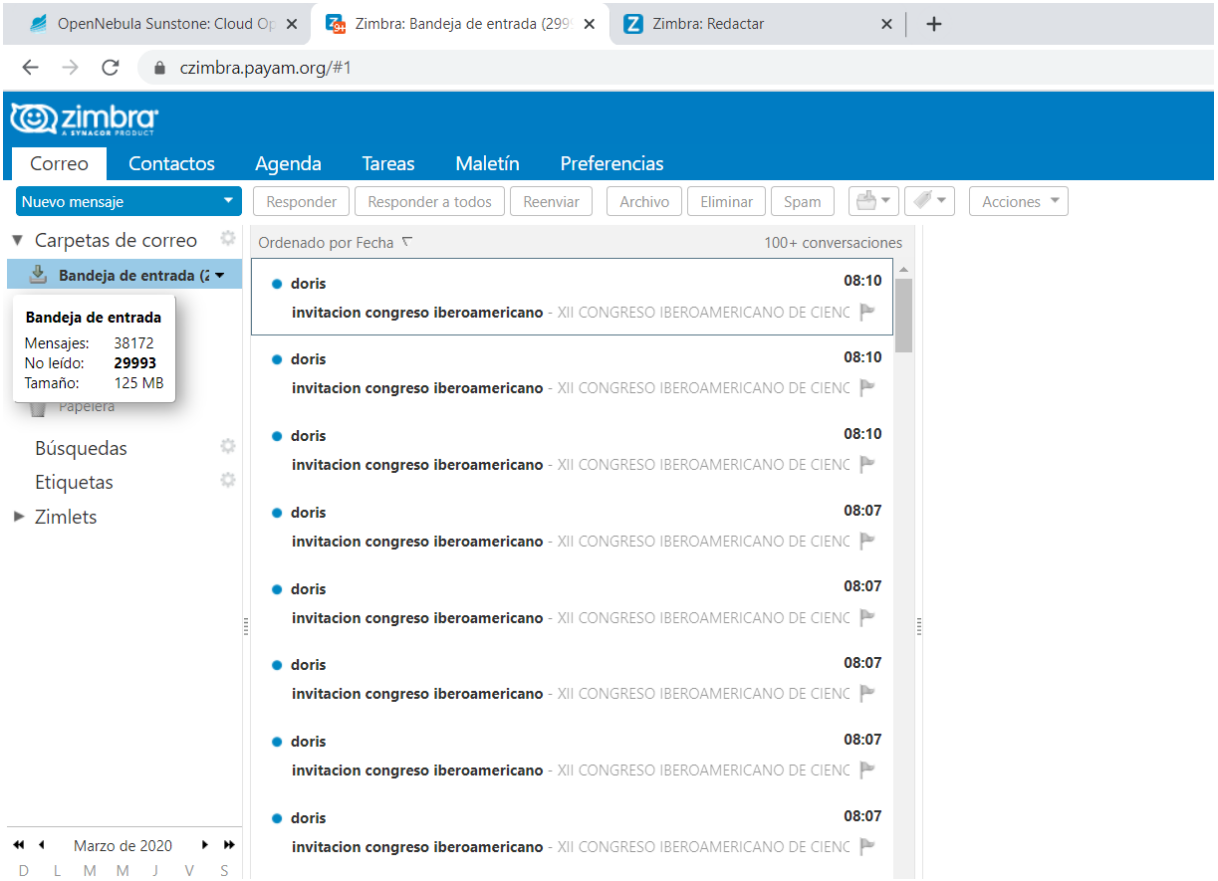


Figura 54 Bandeja de entrada cliente que recibe 29993 mensajes ham
Fuente: Autor

- Gráfica generada con Python que representa el número mensajes enviados y el porcentaje de efectividad de 30000 mensajes ham.

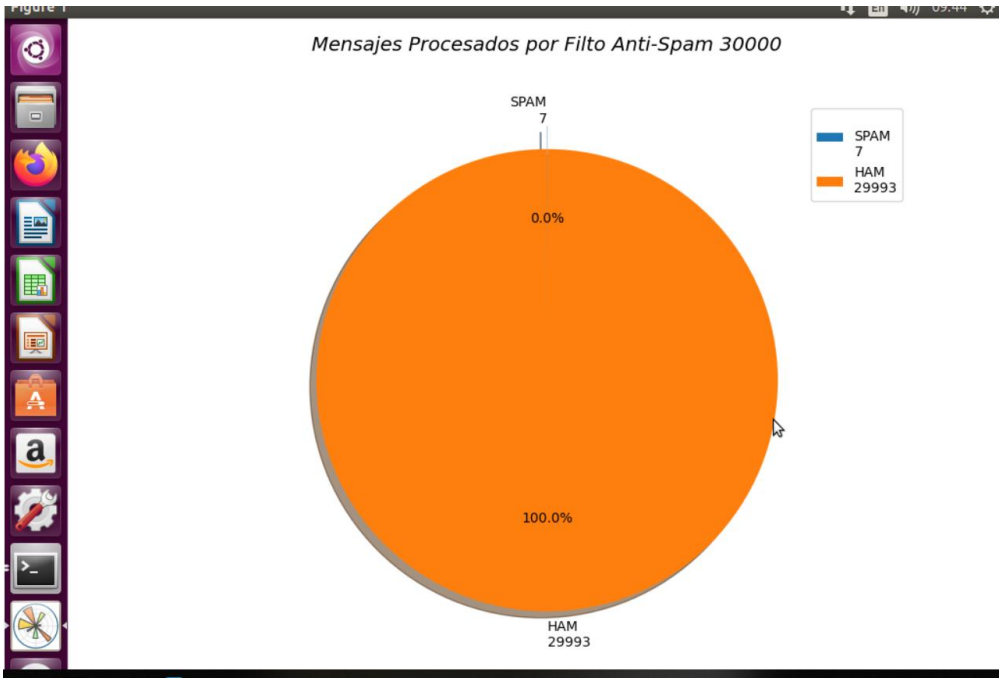


Figura 55 Gráfica de porcentaje de efectividad con él envió de 30000 mensajes ham
Fuente: Autor

Finalmente se realizó una prueba de envío masivo de 30.000 mensajes de tipo ham los cuales se entregaron en la bandeja de entrada 29.993 lo que representa el 100% de efectividad del filtro anti-spam y los 7 mensajes que se fueron a spam representan el 0.0% a pesar de que se enviaron mensajes a spam no representan un valor representativo en el porcentaje por el gran tamaño de envío de mensajes.

Tabla de porcentaje de acuerdo con el número de mensajes SPAM

Como se mostró anteriormente se realizaron diversas pruebas para ver el rendimiento del filtro anti-spam y ver si cumple con la efectividad la cual es del 98.7% ante el envío de correos masivos, se debe de tener en cuenta que la tasa de efectividad puede cambiar sin importar el número de mensajes enviados ya que este valor se lo obtiene en base al contenido del mensaje, por ejemplo con 15.000 mensajes 4 mensajes se enviaron a la bandeja de entrada y con 5.000 los mensajes enviados a la bandeja fueron 5, por lo que el número de mensajes enviados no define la efectividad del filtro si no el contenido que posean esos mensajes, ya que cada una de las pruebas se realizaron con diferentes mensajes.

Número de mensajes	Porcentaje	Mensajes entregados	Número de mensajes spam
5.000	99.9%	5	4.995
15.000	100%	4	14.996
30.000	100%	11	29.989

Tabla 12Tabla de porcentaje de acuerdo con el número de mensajes SPAM
Fuente: Autor

Tabla de porcentaje de acuerdo con el número de mensajes HAM

Al momento de realizar pruebas de mensajes masivos de tipo HAM con un numero de 1.000, 5.000, 30.000 mensajes se puedo determinar que el porcentaje de efectividad va del 99.9% al 100%, esto representa un valor positivo en base a las pruebas realizadas ya que el filtro tiene una efectividad del 98.7%.

Número de mensajes	Porcentaje	Mensajes entregados	Número de mensajes spam
1.000	99.9%	999	1
5.000	99.9%	4.993	7
30.000	100%	29.993	7

Tabla 13 Tabla de porcentaje de acuerdo con el número de mensajes HAM
Fuente: Autor.

Tabla de porcentaje de acuerdo con el número de mensajes enviados con falsos positivos

Al realizar este tipo de pruebas se tomó en cuenta a la facturación electrónica que es uno de los mensajes que en la mayoría de los casos es enviado a spam por algunos de los filtros anti-spam ya que el contenido que posee es algo poco confuso, es por ello que se realiza una primera prueba que dio negativa ante el filtro con un porcentaje de error del 98.7% ya que la mayor cantidad de facturas las envió a spam, pero como se explica anteriormente se debe tener un diccionario robusto para que el entrenamiento sea eficaz, es por ello que se realiza un mejor entrenamiento al diccionario en base a este tipo de contenido, al momento de realizarse la segunda prueba se obtuvo una tasa positiva de efectividad del 99.6% con un numero de mensajes entregados a la bandeja de 4980.

Número de mensajes	Porcentaje	Mensajes entregados	Número de mensajes spam
5.000	98.7%	64	4936
5.000	99.6%	4980	20

Tabla 14 Tabla de porcentaje de acuerdo con el número de mensajes enviados con falsos positivos
Fuente: Autor.

Capítulo 5: Conclusiones y Recomendaciones

En este capítulo se finalizará con las conclusiones y recomendaciones en base a las pruebas que se realizaron en el filtro anti-span de la nube privada.

Conclusiones.

- Para que el filtro anti-spam sea más efectivo lo primero que se debe realizar es el entrenamiento con la ayuda de los archivos que constituye el diccionario, mientras más completo sea el diccionario se va a obtener un mejor porcentaje de efectividad ya que se debe superar el 98.7%.

- En base al entrenamiento realizado del filtro anti-spam este posee una efectividad del 98.7%, pero al momento de realizar las pruebas con el envío masivo de los mensajes ham, spam, falsos positivos se pudo determinar que los resultados están dentro del porcentaje de efectividad ya que los valores obtenidos van del 99.6% al 100%, esto se da debido al tamaño del diccionario de entrenamiento puesto que con este se realiza el aprendizaje automático el mismo que ayuda a clasificar los correos para que llegan a la bandeja de entrada de los usuarios o sean determinados como spam.

- Para realizar las pruebas y obtener resultados del algoritmo se debe utilizar herramientas que validen en un ambiente controlado el envío de correos masivos, por lo que multimail ofrece un entorno de trabajo para generar múltiples peticiones, dando paso a que esta herramienta envié cantidades masivas de correos, para de esa manera poder determinar la eficacia del filtro anti-spam.
- Concluimos que una infraestructura sobre nube es robusta y adecuada para un ambiente de pruebas y producción. La plataforma que hemos utilizado es Opennebula las pruebas se realizaron en la instancia o vm que está en la misma, la cual posee el servidor de correos y el filtro anti-spam, se cumplieron de manera óptima las pruebas sin presentar inconvenientes, pero al momento de tener que procesar una gran cantidad de mensajes es necesario tener un alto número de recursos en el nodo.

Recomendaciones

- La plataforma de Opennebula es mejor instalarla en el sistema operativo Centos 7 debido a que este es compatible con la plataforma, permitiendo acceder a las diferentes librerías y repositorios que se necesita para llevar a cabo dicha instalación con éxito, además, se debe tener cuidado al momento de realizar el paso de las llaves ya que sin eso no se puede tener acceso entre el Font-end y el nodo.
- Al momento de usar Opennebula es recomendable que la máquina que funciona como nodo posea un alto número de recursos para que las instancias que están en el host puedan tener un mejor desenvolvimiento en cada una de las tareas que vayan a ejecutar.
- Antes de realizar la instalar del servidor de correo Zimbra es necesario tener previamente un servidor de correo DNS, luego configurar la sección de hosts para de esa manera no tener inconvenientes en el momento de la instalación, en el momento de la instalación de Zimbra no se requiere instalar el DNS caché puesto que si se instala dicho servicio se tendrá inconvenientes con el servidor de DNS que se va a utilizar.
- Para realizar pruebas de envío masivo de spam con el servidor Zimbra se recomienda utilizar multimail puesto que es compatible con el servidor ya que la herramienta set únicamente funciona con gmail.

Referencias

[1] A. Sánchez , «OpenWebinars,» 29 Enero 2016. [En línea]. Available: https://openwebinars.net/blog/que-es-la-virtualizacion/.

[2] NortonLifeLock, «Software malicioso,» 2019. [En línea]. Available: https://lam.norton.com/internetsecurity-malware-what-is-a-computer-virus.html.

[3] Salesforce, «Cloud Computing - Aplicaciones en un solo tacto,» 2000-2017. [En línea]. Available: https://www.salesforce.com/mx/cloud-computing/.

[4] Wikipedia, «Jetty,» 2020. [En línea]. Available: https://es.wikipedia.org/wiki/Jetty.

[5] Xllauca, «Servidor de correo,» 2015. [En línea]. Available: https://es.slideshare.net/xllauca/servidor-de-correo-55434405.

[6] Ozkgun, «Seguridad de la capa de transporte (TLS),» Academia.edu, 2019.

[7] Ricky, «Multipurpose Internet Mail Extensions,» 20 Marzo 2017. [En línea]. Available: https://www.globalsign.com/es/blog/que-es-smime/.

[8] T. S. T. S. Maria Vergelis, «Spam y phishing en el tercer trimestre de 2019,» 26 Noviembre 2019. [En línea]. Available: https://securelist.lat/spam-report-q3-2019/89777/.

[9] J. Regan, «¿Qué es el phishing? La guía definitiva sobre las estafas y los correos electrónicos de phishing,» 15 febrero 2018. [En línea]. Available: https://www.avg.com/es/signal/what-is-phishing. [Último acceso: 2019 Noviembre 19].

[10] D. G. C. M. C. S. L. G. F. Adriano Vogel, «Private IaaS Clouds: A Comparative Analysis of OpenNebula, CloudStack and OpenStack,» 2016.

[11] C. R. Primorac, «Monografia Adscripción Computación en Nube,» 2014.

[12] Incibe, «Cloud Computing Una guia de aproximación para el empresario,» 2017. [En línea]. Available: https://www.incibe.es/sites/default/files/contenidos/guias/doc/guia-cloud-computing_0.pdf.

[13] D. De, «Mobile Cloud Computing: Architectures, Algorithms and Applications,» Kolkata, Chapman and Hall/CRC, 2016, p. 38.

[14] F. C. Matínez Godínez y B. V. Gutiérrez Galán , «Seguridad Cultura de prevencion para TI,» 2018. [En línea]. Available: https://revista.seguridad.unam.mx/numero-08/computo-en-nube-ventajas-y-desventajas.

[15] Microsoft Azure, «¿Qué es Paas?,» 2020. [En línea]. Available: https://azure.microsoft.com/es-es/overview/what-is-paas/.

[16] A. Benito Carrillo, «Los 3 tipos de servicios que existen dentro del cloud computing en las empresas,» 5 Septiembre 2019. [En línea]. Available: https://www.viafirma.com/blog-xnoccio/es/tipos-servicios-cloud-computing-empresas/.

[17] R. P. Anita Lee-Post, «Cloud Computing: A Comprehensive Introduction,» 2014. [En línea]. Available: https://www.researchgate.net/publication/260038060_Cloud_Computing_A_Comprehensive_Introduction.

[18] F. Magaz, «OpenNebula y Hadoop : Cloud Computing con herramientas Open Source,» Cataluña, 2012.

[19] Microsoft Azure, «¿Qué es una nube pública?,» 2019. [En línea]. Available: https://azure.microsoft.com/en-us/overview/what-is-a-public-cloud/.

[20] M. Rouse, «public cloud,» Julio 2019. [En línea]. Available:

0] https://searchcloudcomputing.techtarget.com/definition/public-cloud.

[2 A. Arias, Computación en la Nube: 2ª Edición, IT Campus Academy, 2015.
1]

[2 IBM Cloud Education, «Nube híbrida,» 16 Octubre 2019. [En línea]. Available:
2] https://www.ibm.com/cloud/learn/hybrid-cloud.

[2 K. Chandrasekaran, Essentials of Cloud Computing, Chapman and Hall/CRC, 2014.
3]

[2 U. R. Pol, «Cloud Computing with Open Source Tool :OpenStack,» *American Journal of*
4] *Engineering Research (AJER),* vol. 3, pp. 233-240, 2014.

[2 . C. D. Cárdenas Clavel, «TIPOS DE ALMACENAMIENTO Y HERRAMIENTAS DE
5] CLOUD COMPUTING,» 23 Octubre 2015. [En línea]. Available:
https://www.academia.edu/22161310/TIPOS_DE_ALMACENAMIENTO_EN_LA_NUBE?auto=download.

[2 L. A. S. K. ,. S. K. J. Rakesh Kumar, «OpenNebula: Open Source IaaS Cloud
6] Computing,» de *National Conference on Computational and Mathematical Sciences (COMPUTATIA-IV)*, 2014.

[2 C. d. S. Caraballo, «Explotación de OpenNebula como plataforma cloud IaaS para la
7] docencia.,» Sevilla, 2015.

[2 Systems, OpenNebula, «OpenNebula Arquitectura de nube abierta,» 2002, 2019. [En
8] línea]. Available:
https://docs.opennebula.org/5.8/deployment/cloud_design/open_cloud_architecture.html.

[2 Opennebula, «Opennebula,» 2002-2019. [En línea]. Available:
9] https://docs.opennebula.org/5.8/deployment/cloud_design/open_cloud_architecture.html#architectural-overview.

[3 P. D. C. D. R. G. Hector Sanjuan Redondo, «Implementacion de un controlador de
0] VirtualBox para OpenNebula,» Madrid, 2011.

[3 docs opennebula, «Command Line Interface,» 2002-2019. [En línea]. Available:
1] https://docs.opennebula.org/5.10/operation/references/cli.html.

[3 archives.OpenNebula.org, «OpenNebula Demo Cloud,» 2014. [En línea]. Available:
2] https://archives.opennebula.org/cloud:cloud.

[3 CESGA, «Instalación y evaluación de OpenNebula,» 2011.
3]

[3 opennebula, «XML-RPC API,» 2002-2019. [En línea]. Available:
4] https://docs.opennebula.org/5.10/integration/system_interfaces/api.html.

[3 docs opennebula, «Managing Virtual Machines Instances,» 2002-2019. [En línea].
5] Available:
https://docs.opennebula.org/5.10/operation/vm_management/vm_instances.html.

[3 MySQL, «Manual de referencia MySQL 8.0,» Oracle Corporation, 2020. [En línea].
6] Available: https://dev.mysql.com/doc/refman/8.0/en/introduction.html.

[3 D. D. N. P. P. E. A. P. G. G. D. S. J. K. B. M. V. S. Thiago Cordeiro, «Open Source Cloud
7] Computing Platforms,» de *Ninth International Conference on Grid and Cloud Computing*, Recife, 2010.

[3 F. M. Villaverde, «OpenNebula y Hadoop: Cloud Computing con herramientas Open
8] Souerce,» 2012.

[3 TechTerms, «Email,» 31 Octubre 2014. [En línea]. Available:
9] https://techterms.com/definition/email.

[4 Computer Hope, «Email,» 1 Diciembre 2019. [En línea]. Available:

0] https://www.computerhope.com/jargon/e/email.htm.

[41] Zimbra Collaboration, «Información general sobre Zimbra,» [En línea]. Available: https://s3.amazonaws.com/files.zimbra.com/public/collateral/Zimbra%20At%20a%20Glance-ES.pdf.

[42] L. T. Y. R. K. R. M. H. C. C. L. A. J. G. Luisa Rave, «INSTALACION Y CONFIGURACION DE ZIMBRA 5-1.0 LISTO,» Antioquia, 2008.

[43] Zimbra Collaboration, «Zimbra Collaboration - Multi-Server Installation Guide,» Febrero 2016. [En línea]. Available: https://s3.amazonaws.com/files.zimbra.com/website/docs/8.7/Zimbra%20Open%20Source%20Edition%20Multi-Server%20Installation%20Guide%208.7.pdf.

[44] O. Mas, «El Blog de Jorge de la Cruz- Zimbra: Arquitectura y Servicios,» 19 Marzo 2014. [En línea]. Available: https://www.jorgedelacruz.es/2014/03/19/zimbra-arquitectura-y-servicios/.

[45] APD, «¿Qué es Machine Learning y cómo funciona?,» 4 Marzo 2019. [En línea]. Available: https://www.apd.es/que-es-machine-learning/.

[46] V. Márquez, «¿Qué es exactamente Machine Learning?,» 29 Octubre 2018. [En línea]. Available: https://medium.com/latinxinai/qu%C3%A9-es-exactamente-machine-learning-77441201a65b.

[47] Management Solutions, «Machine Learning, una pieza clave en la transformación de los modelos de negocio,» 2018. [En línea]. Available: https://www.managementsolutions.com/sites/default/files/publicaciones/esp/machine-learning.pdf.

[48] T. Moes, «¿Qué es spam? El significado y los 5 ejemplos principales,» SoftwareLab , 2019. [En línea]. Available: https://softwarelab.org/es/que-es-spam/.

[49] C. Galvàn, «¿QUÉ ES EL SPAM Y QUÉ TIPOS DE SPAM HAY?,» desafiohosting, 20 Mayo 2019. [En línea]. Available: https://desafiohosting.com/que-es-el-spam-tipos/.

[50] T. S. ,. T. S. Maria Vergelis, «Spam y phishing en el segundo trimestre de 2019,» securelist, 28 Agosto 2019. [En línea]. Available: https://securelist.com/spam-and-phishing-in-q2-2019/92379/.

[51] F. J. A. Lagunes, «Sistema de analisis y filtraje de correo masivo no solicitado SPAM,» Mexico, 2005.

[52] J. M. C. R. Florentino Fdez-Riverola, «Sistemas inteligentes para la detección y filtrado de correo spam: una revisión.,» *Iberoamericana de Inteligencia Artificial.,* nº 23, p. 11, 2007.

[53] A. E. Ortiz, «¿Qué es el filtrado de spam bayesiano?,» 18 Noviembre 2019. [En línea]. Available: https://pcweb.info/que-es-el-filtrado-de-spam-bayesiano/.

[54] T. Gascó, «Definición de Teorema de Bayes,» 3 Enero 2019. [En línea]. Available: https://www.economiasimple.net/glosario/teorema-de-bayes.

[55] J. F. López, «Teorema de Bayes,» 2019. [En línea]. Available: https://economipedia.com/definiciones/teorema-de-bayes.html.

[56] S. Bird, E. Klein y E. Loper, Natural Language Processing with Python, Julie Steele, 2009.

[57] Harrison, «Tokenizar palabras y oraciones con NLTK,» 2015. [En línea]. Available: https://pythonprogramming.net/tokenizing-words-sentences-nltk-tutorial/.

[58] A. Navlani, «Naive Bayes Classification using Scikit-learn,» 4 Dicienbre 2018. [En línea]. Available: https://www.datacamp.com/community/tutorials/naive-bayes-scikit-learn.

[59] J. Brownlee, «Una suave introducción a Scikit-Learn: una biblioteca de Python Machine Learning,» Abril - Agosto 2014-2019. [En línea]. Available: https://machinelearningmastery.com/a-gentle-introduction-to-scikit-learn-a-python-machine-learning-library/.

[60] Univerdidad de Alcalá, «SCIKIT-LEARN, HERRAMIENTA BÁSICA PARA EL DATA SCIENCE EN PYTHON,» 2019. [En línea]. Available: https://www.master-data-scientist.com/scikit-learn-data-science/.

[61] stack overflow, «Developer Suvery Results,» 2019. [En línea]. Available: https://insights.stackoverflow.com/survey/2019#technology.

[62] L. González, «Lenguajes de programación para Machine Learning,» 7 Septiembre 2018. [En línea]. Available: https://ligdigonzalez.com/lenguajes-de-programacion-para-machine-learning/.

[63] G. B. Vega, «Teorema de Bayes,» Calameo, [En línea]. Available: https://es.calameo.com/read/002361664edb38a904ba3.

[64] W. Venema, «pipe - Postfix delivery to external command,» 2019. [En línea]. Available: http://www.postfix.org/pipe.8.html.

[65] postfix, «Postfix After-Queue Content Filter,» 2019. [En línea]. Available: http://www.postfix.org/FILTER_README.html.

5. Anexos

5.1. Anexo 1: Instalación de librería Nltk

A **continuación, se explicará el proceso de instalación de la librería antes mencionada.**

Para hacer uso de la librería nltk es necesario instalarla en el lenguaje de programación Python esto se puede realizar con pip o con conda si se está utilizando anaconda y un entorno virtual, en este caso se utiliza conda como se muestra a continuación:

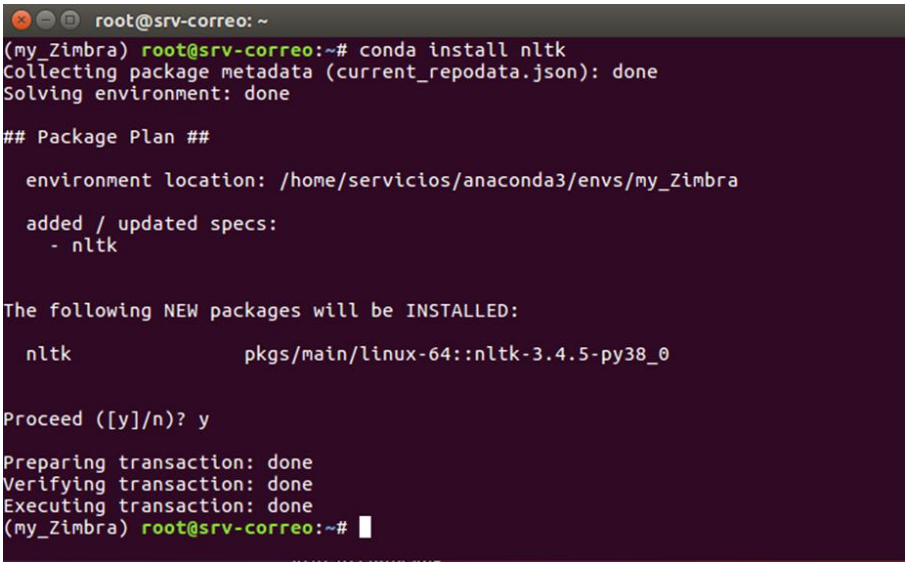


Figura 56:Instalación de la librería nltk

Una vez instalado la librería en Python es necesario ingresar a la consola de Python para poder importar dicha librería, para ello se deberá ingresar los comandos que se muestran en la siguiente figura

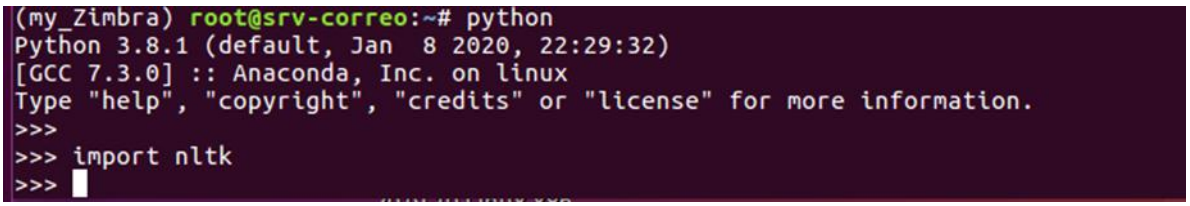


Figura 57: Importación de librería nltk en Python

Cuando haya terminado la importación de la librería es necesario descargar los paquetes de esta para ello se debe ingresar el comando que se muestra en la siguiente figura.

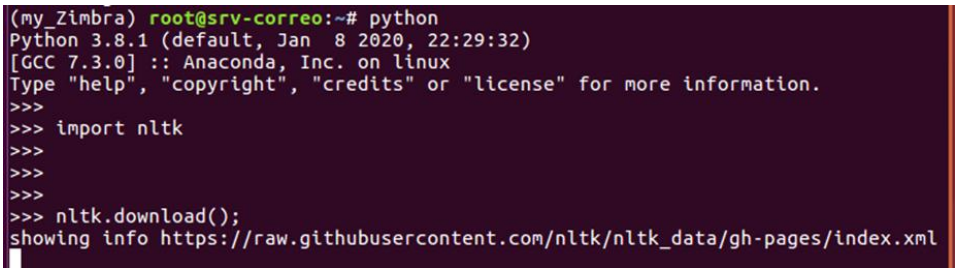


Figura 58: Descarga de paquetes de la librería nltk

Al momento de ingresar el comando que se muestra en la figura 27 aparecerá la siguiente ventana en la cual se deberá presionar el botón de download para que empiece la descarga.

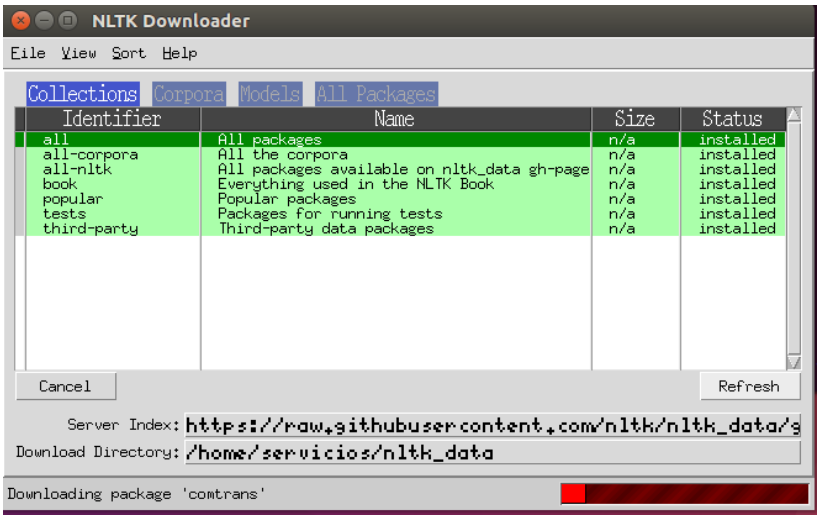


Figura 59: Descarga de paquetes de la librería nltk

5.2. Anexo 2: Instalación de la librería sklearn

Para hacer uso de la librería sklearn es necesario instalar en el lenguaje de programación ptyhon para ello hay que ingresar al terminal e ingresar el comando se muestra a continuación, para que la instalación termine con éxito es necesario **ingresar la letra y para que se esa manera continúe con la instalación.**

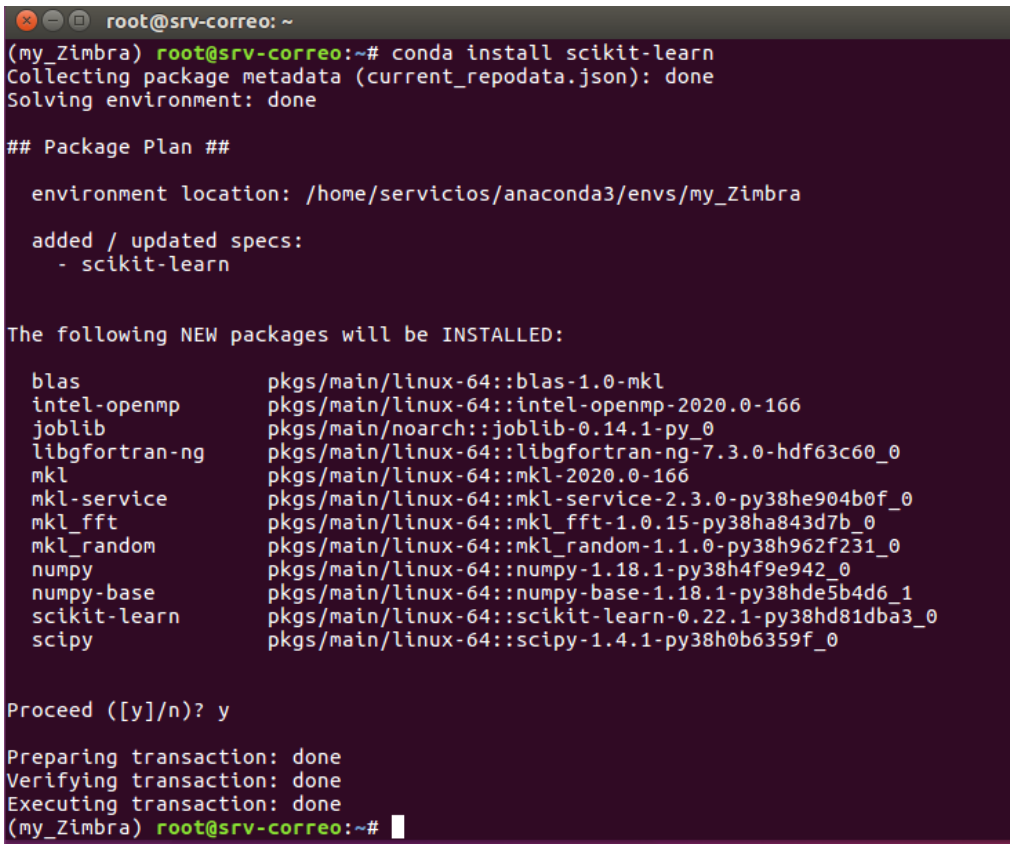


Figura 60: Instalación de la librería sklearn en python

5.3. Anexo 3: Instalación OpenNebula

A continuación, vamos a explicar paso a paso como se realiza la instalación de opennebula.

1. Como primer paso se debe ingresar al archivo /etc/selinux/config y cambiar de enforcing a disable en SELINUX para que quede de esta forma el archivo esto se debe hacer en el Front y en Nodo.

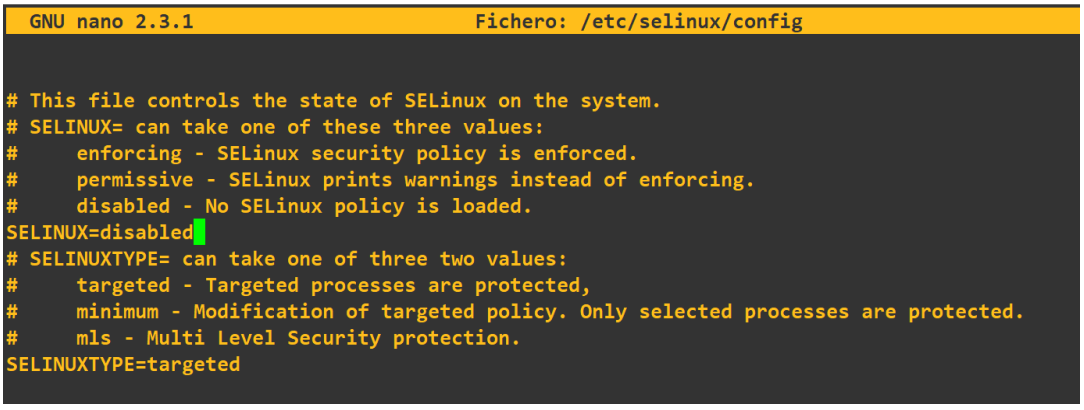


Figura 61 Configuración de archivo config en el front y nodo.

2. Luego se debe realizar un update en cada de una de nuestras maquinas.

Figura 62 Ejecución de update en la máquina de front y nodo.

3. Una vez que estan actualizadas nuestras maquinas se deben ejecutar estas lineas para instalar algunas herramientas y detener el firewall.

```
yum -y install vim net-tools nfs-utils bridge-utils
systemctl stop firewalld
```

4. Ya que se ah desactivado el firewall se debe crear un archivo el cual es el siguiente.

```
touch /etc/yum.repos.d/opennebula.repo
```

5. Luego se debe abrir el archivo que se creo anteriormente **/etc/yum.repos.d/opennebula.repo** se debe escrbir lo siguiente en el mismo.

```
[opennebula]
name=opennebula
baseurl=https://downloads.opennebula.org/repo/5.4/CentOS
   /7/x86_64
enabled=1
gpgkey=https://downloads.opennebula.org/repo/repo.key
gpgcheck=1
#repo_gpgcheck=1
```

Estas lineas permiten descargar opennebula

6. Una vez que esta el archivo se debe ejecutar esta linea, la cual va ah permitir que se descargue opennebula.

```
yum install epel-release
```

Se debe recalcar que los pasos anteriores se los realiza tanto para el Front como para el Nodo.

7. Este paso solo se lo debe realizar en el Front, mandamos a realizar la siguiente intalacion.

```
yum install -y opennebula-server opennebula-sunstone
  opennebula-ruby opennebula-gate opennebula-flow
```

Figura 63 Instalación de requerimientos en el Front

8. A diferencia de este paso que se debe de realizar en la maquina que es nodo.

```
yum install opennnebula-node-kvm
```

Figura 64 Instalación de requerimientos en el nodo.

9. Instalacion de Gemas en el Front.

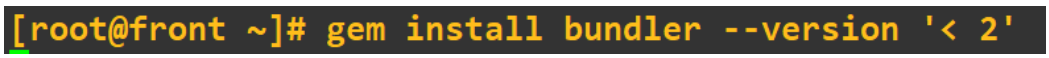


Figura 65 Instalación de bundler en el front

10. Luego verificar que se instalaron las gemas cuando se ejecuta esta linea /usr/share/one/install_gems.

```
Fetching thin 1.7.0
Installing thin 1.7.0 with native extensions
Fetching treetop 1.6.8
Installing treetop 1.6.8
Fetching trollop 2.1.2
Installing trollop 2.1.2
Fetching uuidtools 2.1.5
Installing uuidtools 2.1.5
Fetching zendesk_api 1.13.4
Installing zendesk_api 1.13.4
Bundle complete! 21 Gemfile dependencies, 43 gems now installed.
Use `bundle info [gemname]` to see where a bundled gem is installed.
[root@front ~]#
```

Figura 66 Instalación de Gemas

11. Cambio de contraseña del usuario oneadmin.

```
  GNU nano 2.3.1                          Fichero: /var/lib/one/.one/one_auth

oneadmin:Patito.123
```

Figura 67 Cambio de contraseña

12. Se debe levantar el servicio de opennebula y opennebula-susntone. De hay verificamos con oneuser show la informacion del usuario oneadmin.

```
[root@front ~]# nano /var/lib/one/.one/one_auth
[root@front ~]# systemctl start opennebula
[root@front ~]# systemctl start opennebula-sunstone
[root@front ~]# oneuser show
USER 0 INFORMATION
ID              : 0
NAME            : oneadmin
GROUP           : oneadmin
PASSWORD        : fd232e218fcc3f4f7680be923f104b66e87942bd
AUTH_DRIVER     : core
ENABLED         : Yes

TOKENS

USER TEMPLATE
TOKEN_PASSWORD="78df6fea0ae4e73767801601585fe4d1dcac9643"

RESOURCE USAGE & QUOTAS

[root@front ~]#
```

Figura 68 Verificación de usuario oneadmin

13. Luego se debe crear el bridge en el nodo con la herramienta nmtui.

Figura 69 Herramienta nmtui

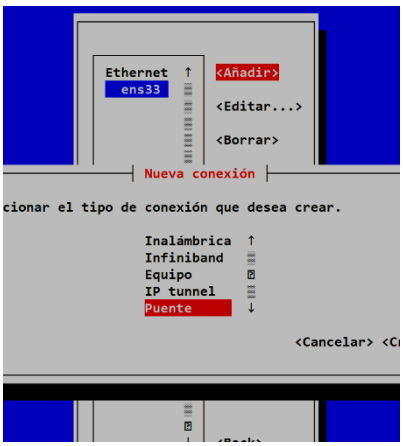


Figura 70 Seleccion de tipo de conexion

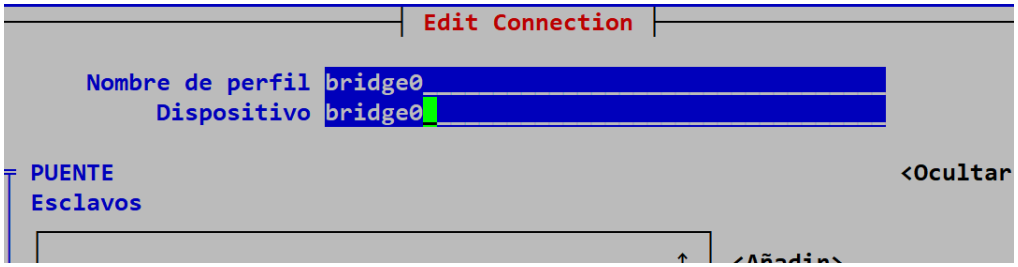


Figura 71 Establecer nombre de conexión

14. Ingresamos en el nodo a la ruta espeficada en la imagen para realizar los cambios en el puente creado.

```
[root@nodo ~]# nano /etc/sysconfig/network-scripts/ifcfg-bridge0
```

Figura 72 Ruta de la conexión creada.

```
STP=yes
BRIDGING_OPTS=priority=32768
TYPE=Bridge
PROXY_METHOD=none
BROWSER_ONLY=no
BOOTPROTO=static
DEFROUTE=yes
IPV4_FAILURE_FATAL=no
IPV6INIT=yes
IPV6_AUTOCONF=yes
IPV6_DEFROUTE=yes
IPV6_FAILURE_FATAL=no
IPV6_ADDR_GEN_MODE=stable-privacy
NAME=bridge0
UUID=2d807397-e992-45d2-9e1c-ca94f47dab4a
DEVICE=bridge0
ONBOOT=yes
IPADDR=192.168.1.199
NETMASK=255.255.255.0
GATEWAY=192.168.1.1
DNS1=192.168.1.254
```

Figura 73 Cambio de direcciones en la conexión.

15. Ahora debemos ingresar en la conexión ens33 para añadir en la misma el puente que se creó.

```
[root@nodo ~]# nano /etc/sysconfig/network-scripts/ifcfg-ens33
```

Figura 74 Dirección de conexión en el nodo.

Figura 75 Cambios realizados en la interfaz ens33

16. Verificamos que nuestro puente este bien configurado

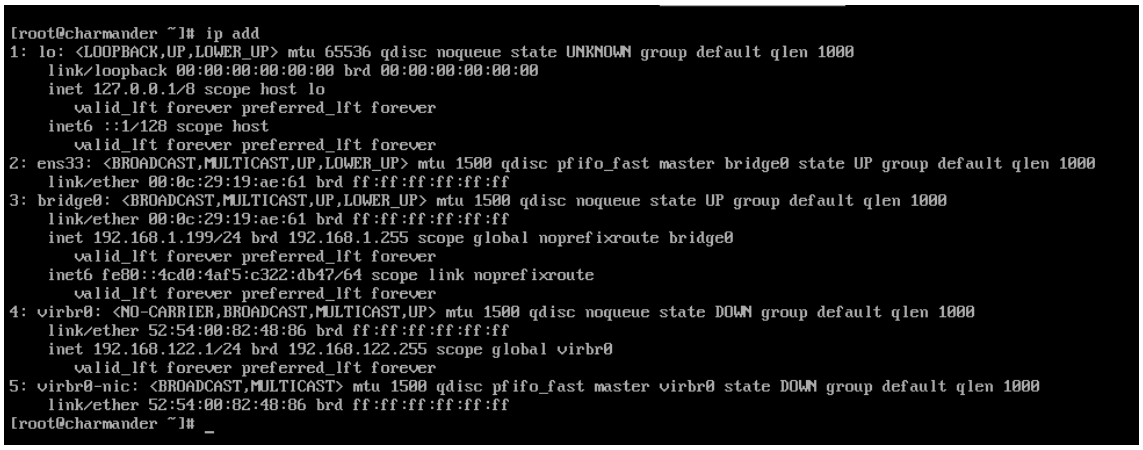


Figura 76 Verificación de conexión con el puente.

17. Una vez que hemos creado el puente y tenemos ping entre el nodo y el front. Debemos asignar una contraseña el usuario Oneadmin.

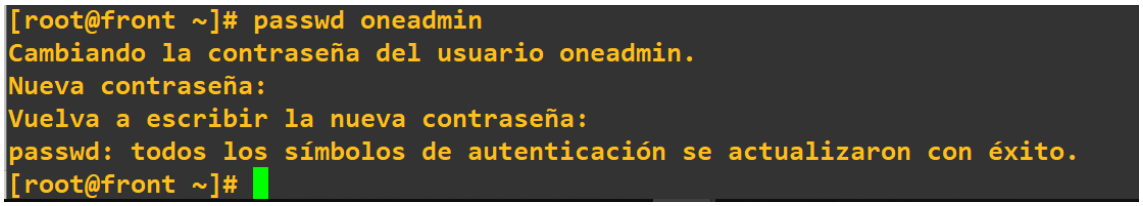


Figura 77 Asignar contraseña en el front al usuario Oneadmin

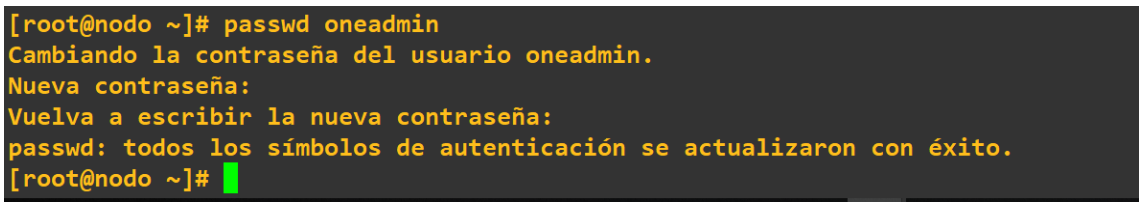


Figura 78 Asignar contraseña en el nodo al usuario Oneadmin

18. En el front primeramente, ingresamos con el usuario oneadmin para fijarnos en esta ruta **cd /var/lib/one/.ssh/** y mostramos sus archivos.

Se debe tener claro que el proceso de eliminar los archivos solo se realiza en el front ya que el nodo no posee esos archivos.

19. Creamos la clave ssh con la ayuda de la herramienta **ssh-keygen** en el nodo y en el front.

```
[oneadmin@front .ssh]$ ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/var/lib/one/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /var/lib/one/.ssh/id_rsa.
Your public key has been saved in /var/lib/one/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:PlnxgtX7x8DcQ+E1Nh2UIbv/05nSe7nJD3KQeo91Id4 oneadmin@front
The key's randomart image is:
+---[RSA 2048]----+
|             ..O*|
|           .  =.*|
|          o .. o |
|         o o =o. |
|        S o =o+o.|
|       . o o.o+oo|
|        + . o.*EO|
|         . . B.X=|
|            . o=O|
+----[SHA256]-----+
[oneadmin@front .ssh]$
```

Figura 79 Generación de clave en el front y nodo.

20. Copiamos la clave pública del Front a el Nodo y de forma visceversa.

```
[oneadmin@front .ssh]$ scp id_rsa.pub oneadmin@nodo:/var/lib/one/.ssh/llaveF
The authenticity of host 'nodo (192.168.1.199)' can't be established.
ECDSA key fingerprint is SHA256:Q3wT43HlnQGVKh73ggd6tNzcZTdboIL2E8J2HxS3b5U.
ECDSA key fingerprint is MD5:1a:89:d8:b2:ac:29:05:46:3a:42:23:8e:07:08:81:09.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'nodo,192.168.1.199' (ECDSA) to the list of known hos
ts.
oneadmin@nodo's password:
Permission denied, please try again.
oneadmin@nodo's password:
id_rsa.pub                                    100%  396   417.4KB/s   00:00
[oneadmin@front .ssh]$
```

Figura 80 Copia de clave del front y nodo

```
[oneadmin@nodo .ssh]$ scp id_rsa.pub oneadmin@front:/var/lib/one/.ssh/llaveN
The authenticity of host 'front (192.168.1.200)' can't be established.
ECDSA key fingerprint is SHA256:Xa7w+UsKqWj1HbUzyg6R6PyYlDkqSLs614lLMkKUGEQ.
ECDSA key fingerprint is MD5:68:15:a9:e9:78:53:79:83:db:b2:e5:fe:c2:a5:f9:b0.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'front,192.168.1.200' (ECDSA) to the list of known ho
sts.
oneadmin@front's password:
id_rsa.pub                                    100%  395   468.4KB/s   00:00
[oneadmin@nodo .ssh]$
```

Figura 81 Copia de clave del nodo al front

21. Las claves públicas del front y del nodo las añadimos en un solo archivo llamado **authorized_keys.** En los dos equipos se realiza esta acción.

```
[oneadmin@front .ssh]$ cat id_rsa.pub >> authorized_keys
[oneadmin@front .ssh]$ cat llaveN >> authorized_keys
```

Figura 82 Añadir claves en el front

```
[oneadmin@nodo .ssh]$ cat id_rsa.pub >> authorized_keys
[oneadmin@nodo .ssh]$ cat llaveF >> authorized_keys
```

Figura 83 Añadir claves en el nodo

22. Damos permisos al archivo que se creó.

```
[oneadmin@front .ssh]$ chmod 755 authorized_keys
```

Figura 84 Permisos al archivo creado en el front y nodo

23. Ahora debemos salir del usuario oneadmin y dejamos sin contraseña al usuario.

```
[oneadmin@front .ssh]$ exit
exit
[root@front ~]# passwd -d oneadmin
Eliminando la contraseña del usuario oneadmin.
passwd: Éxito
[root@front ~]#
```

Figura 85 Quitar contraseña del usuario oneadmin en el front y nodo

24. Una vez que le quitamos la contraseña al usuario oneadmin vamos a realizar algunas modificaciones en el archivo **/etc/ssh/sshd_config** para poder acceder por medio de ssh entre el nodo y el front.

Los cambios que vamos a realizar son en el nodo y en el front:

```
PermitRootLogin yes
PubkeyAuthentication yes
cambiamos ruta
AuthorizedKeysFile /var/lib/one/.ssh/authorized_keys
PasswordAuthentication no--cambiamos de yes a no
```

```
  GNU nano 2.3.1                    Fichero: /etc/ssh/sshd_config


# Logging
#SyslogFacility AUTH
SyslogFacility AUTHPRIV
#LogLevel INFO

# Authentication:

#LoginGraceTime 2m
PermitRootLogin yes
#StrictModes yes
#MaxAuthTries 6
#MaxSessions 10

PubkeyAuthentication yes

# The default is to check both .ssh/authorized_keys and .ssh/authorized_keys2
# but this is overridden so installations will only check .ssh/authorized_keys
AuthorizedKeysFile      /var/lib/one/.ssh/authorized_keys

#AuthorizedPrincipalsFile none

# Don't read the user's ~/.rhosts and ~/.shosts files
#IgnoreRhosts yes

# To disable tunneled clear text passwords, change to no here!
#PasswordAuthentication yes
#PermitEmptyPasswords no
PasswordAuthentication no
```

Figura 86 Configuración del archivo ssh

25. Reiniciamos el servicio de ssh

26. Pruebas de conexión ssh al nodo y al front

```
[root@front ~]# su oneadmin
[oneadmin@front root]$ ssh oneadmin@nodo
Last failed login: Sat Apr 20 19:14:24 -05 2019 from front on ssh:notty
There were 4 failed login attempts since the last successful login.
Last login: Sat Apr 20 18:48:27 2019
[oneadmin@nodo ~]$ exit
logout
Connection to nodo closed.
[oneadmin@front root]$ ssh oneadmin@front
The authenticity of host 'front (192.168.1.200)' can't be established.
ECDSA key fingerprint is SHA256:Xa7w+UsKqWj1HbUzyg6R6PyYlDkqSLs6141LMkKUGEQ.
ECDSA key fingerprint is MD5:68:15:a9:e9:78:53:79:83:db:b2:e5:fe:c2:a5:f9:b0.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'front,192.168.1.200' (ECDSA) to the list of known hosts.
Last login: Sat Apr 20 19:16:56 2019
[oneadmin@front ~]$ ssh oneadmin@nodo
Last login: Sat Apr 20 19:22:01 2019 from nodo
[oneadmin@nodo ~]$ exit
logout
Connection to nodo closed.
```

Figura 87 Pruebas de conexión ssh en el front

```
[root@nodo ~]# su oneadmin
[oneadmin@nodo root]$ ssh oneadmin@nodo
The authenticity of host 'nodo (192.168.1.199)' can't be established.
ECDSA key fingerprint is SHA256:Q3wT43HlnQGVKh73ggd6tNzcZTdboIL2E8J2HxS3b5U.
ECDSA key fingerprint is MD5:1a:89:d8:b2:ac:29:05:46:3a:42:23:8e:07:08:81:09.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'nodo,192.168.1.199' (ECDSA) to the list of known hosts.
Last login: Sat Apr 20 19:21:58 2019
[oneadmin@nodo ~]$ exit
logout
Connection to nodo closed.
[oneadmin@nodo root]$ ssh oneadmin@front
Last login: Sat Apr 20 19:21:52 2019 from front
[oneadmin@front ~]$ exit
logout
Connection to front closed.
[oneadmin@nodo root]$
```

Figura 88 Pruebas de conexión ssh en el nodo

27. Damos permisos a la carpeta a todos lo que tenga en la carpeta **/var/lib/***

```
[root@front ~]# chown -R oneadmin:oneadmin /var/lib/*
```

Figura 89 Permisos en la carpeta /var/lib en el front y nodo

28. Debemos añadir los hosts en el archivo de configuración **/etc/hosts**

```
root@front:~
  GNU nano 2.3.1                          Fichero: /etc/hosts

127.0.0.1   localhost localhost.localdomain localhost4 localhost4.localdomain4
::1         localhost localhost.localdomain localhost6 localhost6.localdomain6

192.168.1.200   front
192.168.1.199   nodo
```

Figura 90Añadir hosts en el front y nodo

29. Como último paso para la configuración de opennebula debemos crear el nodo en el front.

```
[oneadmin@front root]$ onehost create nodo -i kvm -v kvm
```

Figura 91 Creación de nodo en el front

30. Mandamos a listar para ver si se creó el host.

```
[oneadmin@front root]$ onehost list
  ID NAME            CLUSTER   RVM      ALLOCATED CPU      ALLOCATED MEM STAT
   1 nodo            default     0      0 / 400 (0%)     0K / 3.7G (0%) on
```

Figura 92 Lista de hosts

31. Ingresamos al navegador con nuestro dominio y el puerto http://front.spamhercules.org:9869/

Figura 93 Ingresar a opennebula desde nuestro Navegador

LEVANTAMIENTO DE MAQUINAS VIRTUALES EN OPENNEBULA.

32. Primero debemos crear la red que vamos a usar nuestras máquinas para ellos nos debemos dirigir a network y nos vamos a Virtual Network.

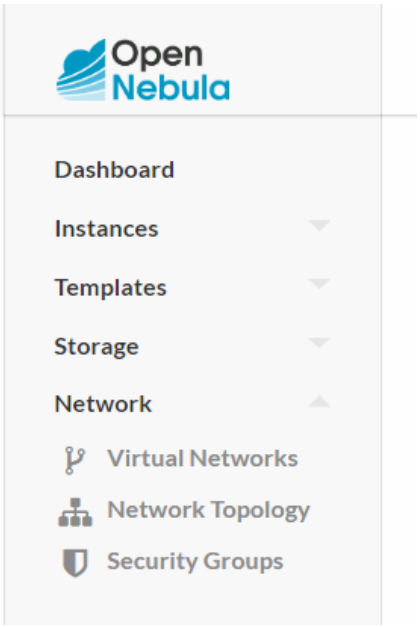


Figura 94 Seleccionar crear red virtual

33. Agregamos un nombre a la red

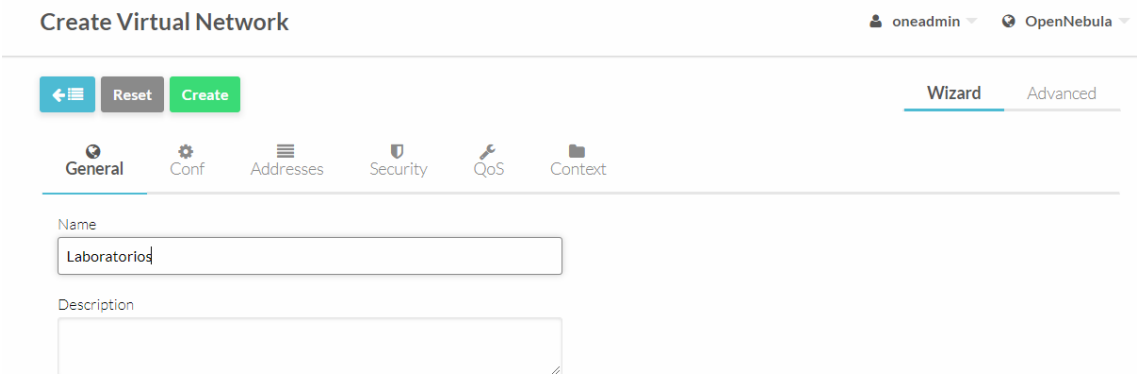


Figura 95 Asignar nombre a la red virtual.

34. Escribimos el nombre de nuestro puente en este caso bridge0

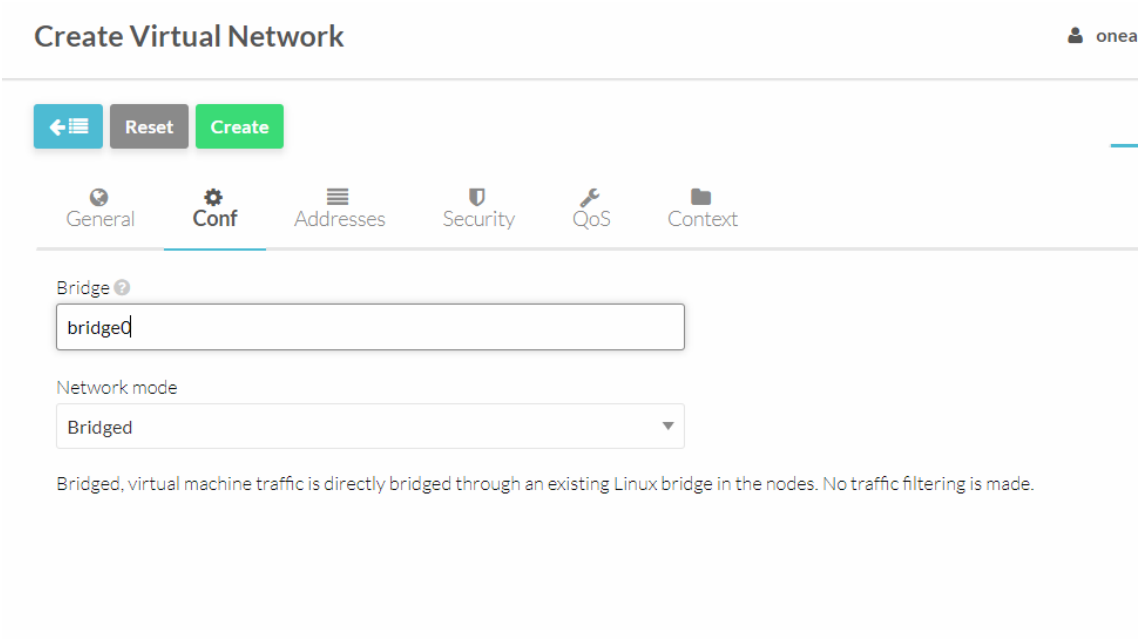


Figura 96 Escribir el nombre del puente

35. Asignamos desde que ip deseamos que se empiece y el rango. Teniendo en cuenta que es no es la ip que va a tener la máquina que va a ser montada.

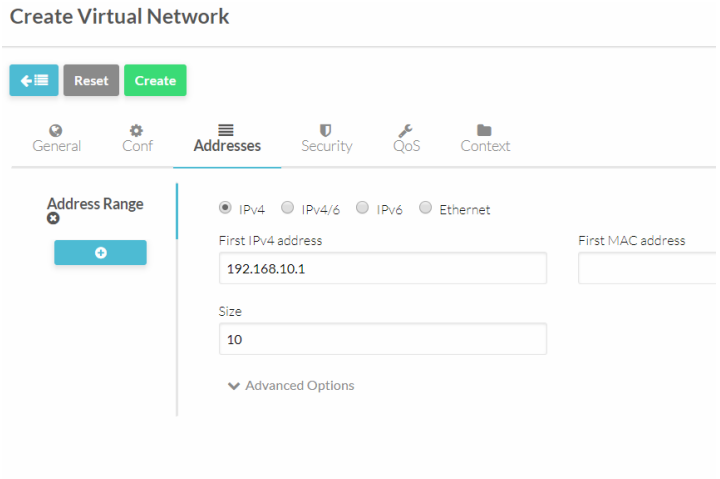


Figura 97 Asignar rango de ip

36. Como último paso le damos crear

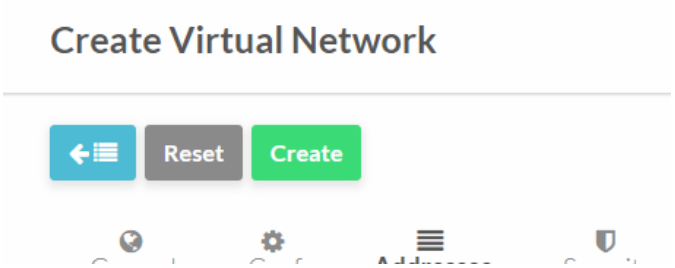


Figura 98Seleccionar dar click en crear

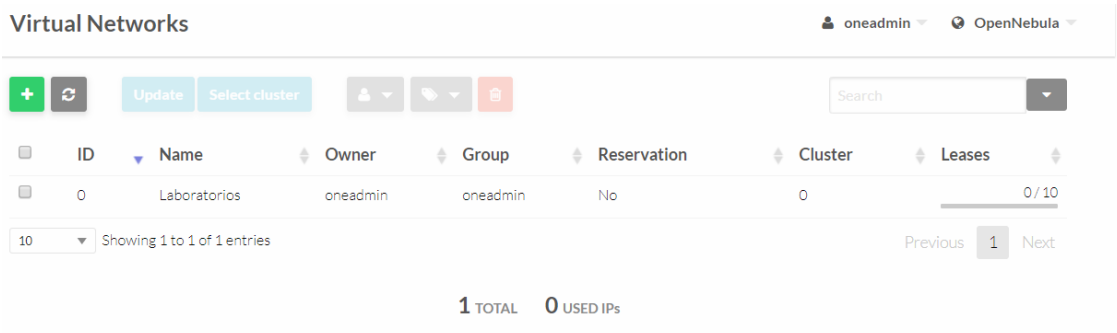


Figura 99 Visualización de red virtual creada.

37. Ahora debemos crear los discos para ellos nos situamos en Storage luego en Images

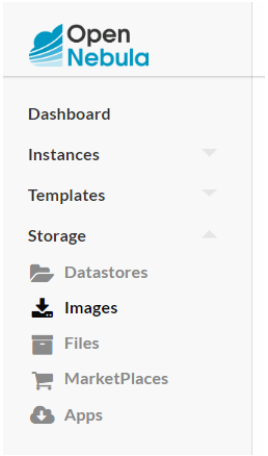


Figura 100 Seleccionar Images

38. Establecemos el nombre de la imagen

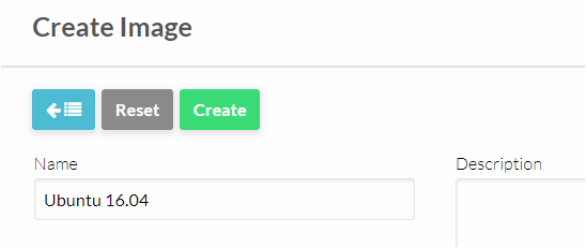


Figura 101 Establecer nombre de la imagen

39. Cambiamos a CD-ROM

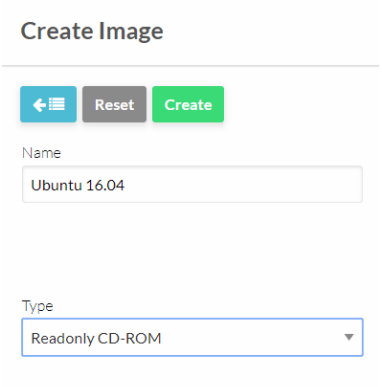


Figura 102 Seleccionamos CD-ROM

40. Seleccionamos upload y subimos la imagen

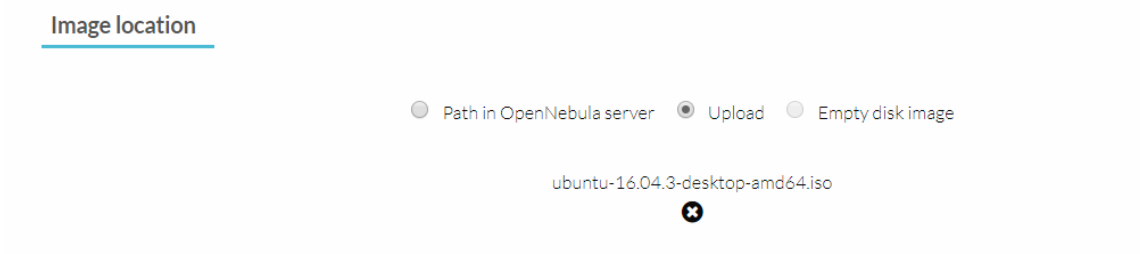


Figura 103 Subimos la imagen

41. Nos vamos a opciones avanzadas y seleccionamos qcow2

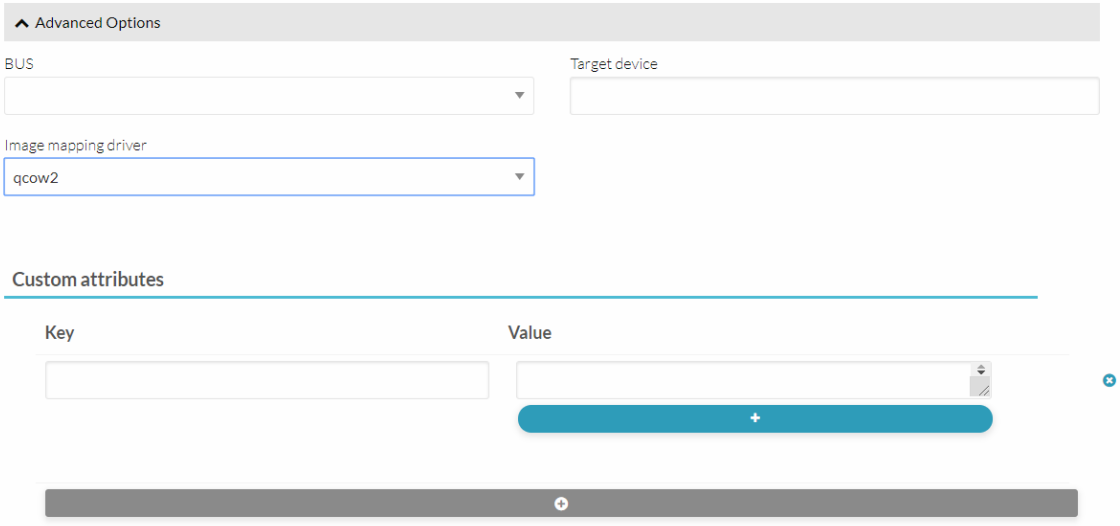


Figura 104 Seleccionamos qcow2

42. De esa manera ya tenemos creado los dos discos

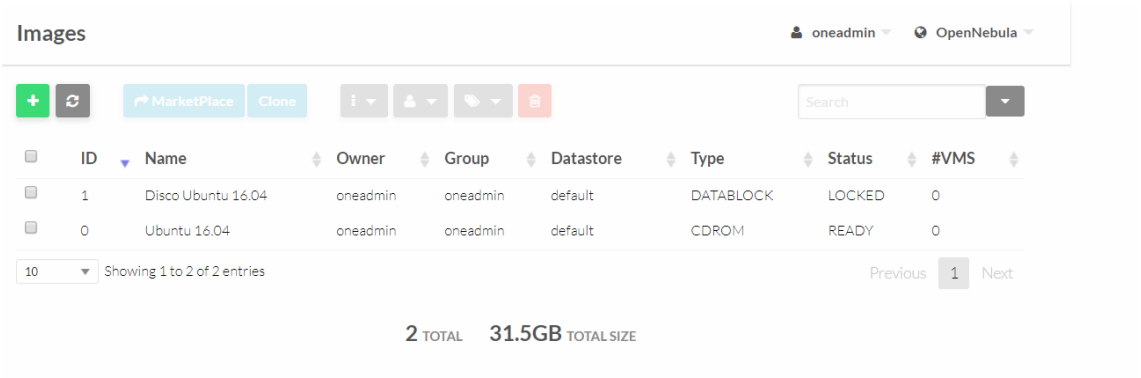


Figura 105 Visualización de Discos creados

43. Debemos crear un VMs en template

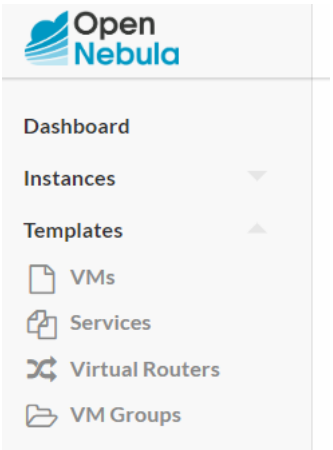


Figura 106 Seleccionar crear VMS

44. Definimos el nombre de la VMs

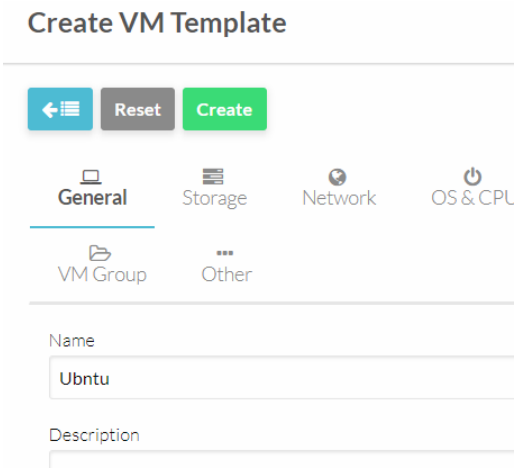


Figura 107 Definir nombre de la VMs

45. Asignamos el espacio y la memoria que va a tener la VMs

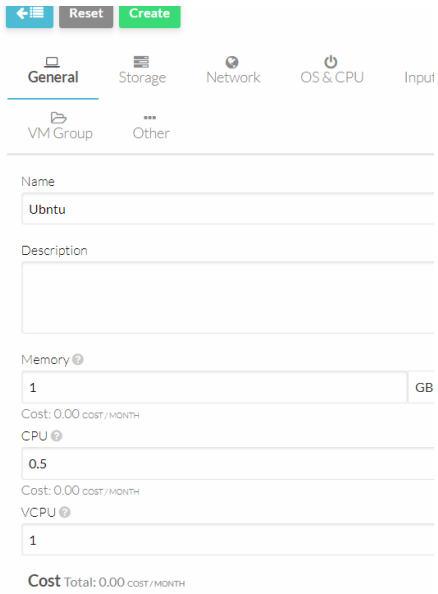


Figura 108 Información de la VMs a crearse

46. Asignamos los dos discos que debemos usar en la VMs

Figura 109 Asignación de Discos a usar

47. Seleccionar la red que vamos a usar

Figura 110 Seleccionar la red de la vm

48. Seleccionamos el disco que contiene la ISO a instalar

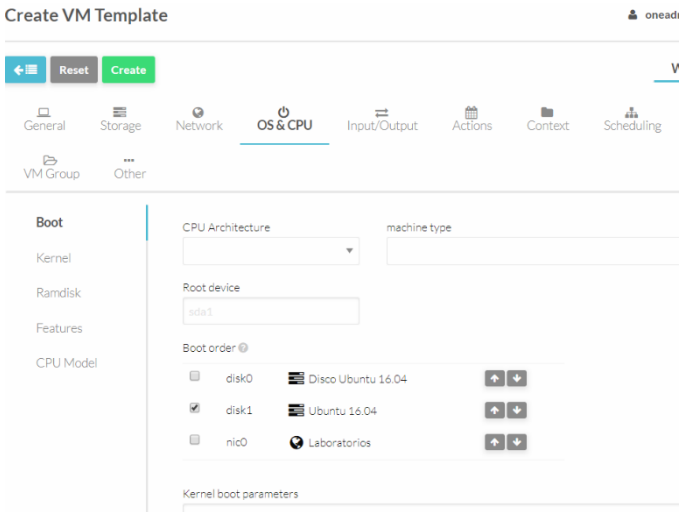


Figura 111 Seleccionar el disco con la ISO

49. Luego damos en crear el VMs.

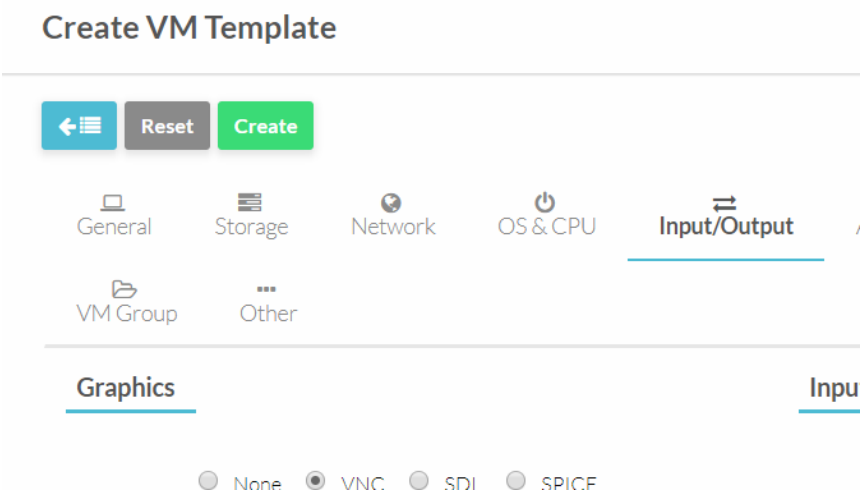


Figura 112 Creación de la VMs.

50. Lista de las instancias.

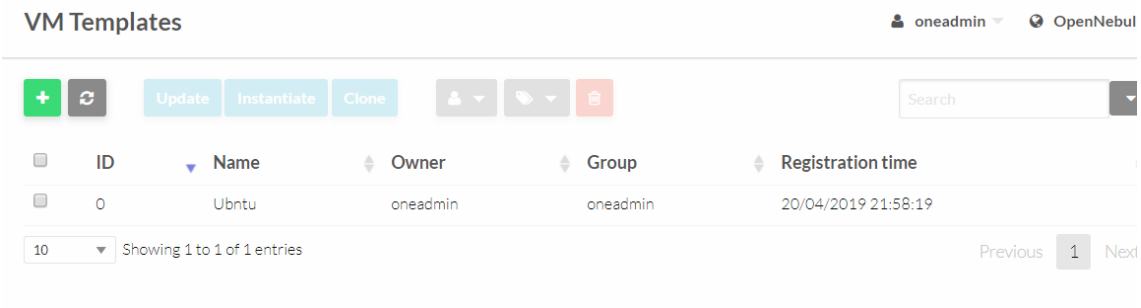


Figura 113 Lista de las instancias creadas.

51. Ingresamos a la VMs y damos click en Instiatate.

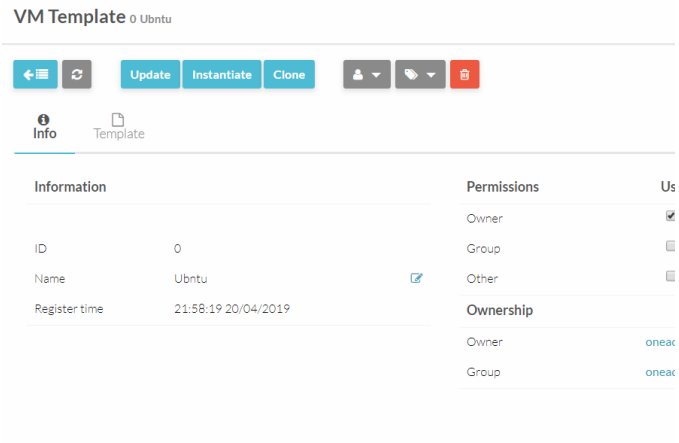


Figura 114 Creación de instantenea.

52. Definimos el nombre de la instancia y volvemos a dar en instantiate estaría creada.

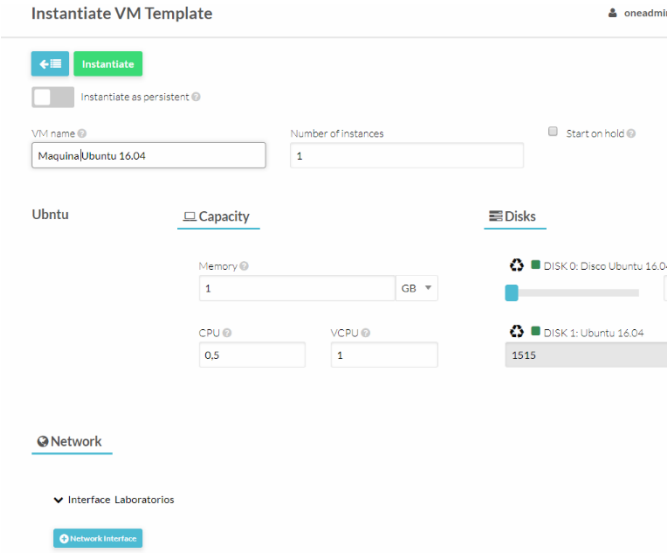


Figura 115 Definir nombre de la instancia

53. Con la ayuda de VNC se ejecutará la instalación Ubuntu 16.04.

54. Máquina virtual instalada.

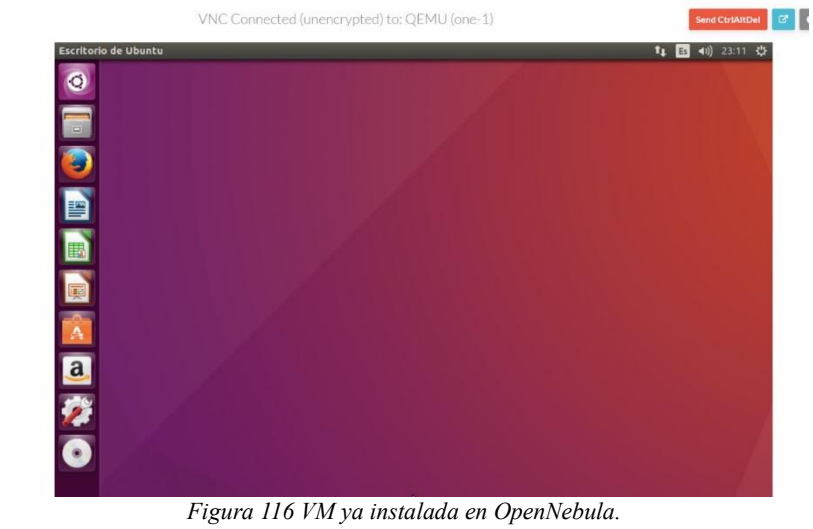


Figura 116 VM ya instalada en OpenNebula.