Universidad de Buenos Aires

Facultades de Ciencias Económicas,
Cs. Exactas y Naturales e Ingeniería

Carrera de Especialización en Seguridad Informática

Trabajo Final

Métodos de Detección Automática de Fraudes Informáticos por Suplantación de Identidad

Autor: Lic. Francisco Benjamin Wesner

Tutor: Dr. Pedro Hecht

Año de Presentación: 2020

Cohorte del Cursante: 2018

Declaración Jurada de origen de los contenidos

“Por medio de la presente, el autor manifiesta conocer y aceptar el Reglamento de Trabajos Finales vigente y se hace responsable que la totalidad de los contenidos del presente documento son originales y de su creación exclusiva, o bien pertenecen a terceros u otras fuentes, que han sido adecuadamente referenciados y cuya inclusión no infringe la legislación Nacional e Internacional de Propiedad Intelectual”

Francisco Benjamín Wesner

Métodos de Detección Automática de Fraudes Informáticos por Suplantación de Identidad

El presente documento consiste en una investigación sobre distintos métodos de detección, tanto manual como automática, de correos electrónicos y sitios web fraudulentos que utilizan técnicas de ingeniería social. El trabajo se enfoca principalmente sobre detección de correos y sitios web de suplantación de identidad (phishing) utilizando técnicas de machine learning.

Palabras claves: Ingeniería Social, Phishing, Machine Learning, Automatización.

Índice

1. Nómina de abreviaturas

ML: Machine Learning. Aprendizaje automático.

SVN: Support Vector Machines. Un tipo de algoritmo de aprendizaje automático.

API: Application Programming Interface. Interfaz de programación de aplicaciones.

HTML: HyperText Markup Language. Lenguaje de marcado que se utiliza en el desarrollo de páginas web.

URL: Uniform Rsource Locator. Es la dirección que se asigna a un recurso en una red.

IP: protocolo de internet. También se usa para referirse a la dirección IP, número que identifica a una interfaz de red en el protocolo del mismo nombre.

RFC: Request for Comments. Publicaciones del Internet Engineering Task Force, que describen distintos aspectos y convenciones del funcionamiento de internet y sus protocolos.

DKIM: Domain Keys Identified Mail. Mecanismo de autenticación de correos electrónicos.

SPF: Sender Policy Framework. Identifica los orígenes aprobados de un dominio de correo.

DMARC: Domain Based Message Authentication, Reporting and Conformance. Es un protocolo de autenticación y conformidad de correos.

DNS: Domain Name System. Sistema de nomenclatura para dispositivos en una red.

2. Introducción

2.1. Contexto

La ingeniería social consiste en manipular a una persona a través de técnicas psicológicas y habilidades sociales para cumplir con un objetivo específico, como puede ser la obtención de información confidencial, que permita al atacante acceder a un equipo o sistema, o instalar un programa malicioso. Cuando el atacante realiza ingeniera social para hacerse pasar por una empresa, organización o persona, para que la víctima revele datos confidenciales o realice alguna otra acción, se lo denomina phishing.

El término “phishing” fue utilizado por primera vez por un grupo de hackers en 1996, que robaron información de cuentas de “America Online (AOL)” engañando a sus usuarios. Se definió el phishing, o suplantación de identidad, como un tipo de fraude en el cual se intenta obtener datos altamente confidenciales, como información de inicio de sesión de usuarios o detalles de tarjetas de crédito, haciéndose pasar por una entidad legitima. [1]

Generalmente un ataque de phishing se ejecuta a través de correos electrónicos que llevan a los consumidores a responder el mensaje con información confidencial, ingresar datos personales en un sitio web falso cuya apariencia es similar al sitio original, o bajar un adjunto con código malicioso. Dentro de la información confidencial que se puede filtrar en un ataque de phishing se encuentran, por ejemplo, datos como nombres de usuarios de un banco, contraseñas, números de contacto, direcciones, detalles de tarjetas bancarias, entre otros.

Existen también ataques de phishing que no utilizan el correo electrónico o un sitio web como herramienta para cumplir con su objetivo. Por ejemplo, existe el phishing por mensaje de texto o llamada telefónica. En esta investigación haremos foco en los dos primeros tipos, aquellos que ingresan por correo o llevan al usuario a ingresar en un sitio web.

Los ataques de ingeniería social, particularmente los de tipo phishing, son, al día de hoy, una de las amenazas más comunes en lo que se refiere a seguridad de la información. Existen numerosos ejemplos de ataques exitosos que iniciaron, por ejemplo, con un correo de phishing, y sin embargo, no existen muchas herramientas a la hora de defendernos ante este tipo de amenazas.

2.2. Objetivo y alcance

El objetivo del trabajo propuesto consiste en exponer los mecanismos existentes de detección de correos electrónicos y sitios web de phishing, enfocándose principalmente a los métodos automáticos y en los algoritmos de machine learning que puedan ser utilizados de forma eficiente en la resolución de este problema.

El alcance del trabajo estará dado por la bibliografía descrita en la sección de referencias. Se presentarán las conclusiones obtenidas de este proceso, enfocándose en la comparación de los distintos métodos analizados y una predicción personal sobre el desarrollo de estas técnicas en los próximos años.

En primer lugar, se realizará una revisión del estado actual del arte en lo que respecta a clasificación y detección de ataques de phishing, ya sea correos y sitios web. Se pretende también crear un sentido de alerta entre los lectores para prevenir este tipo de ataques. Posteriormente se evaluarán métodos automáticos que utilicen técnicas de aprendizaje automático y procesamiento de lenguaje natural. Se pretende comparar esas técnicas en la detección este tipo de ataques, analizando la efectividad de estos modelos predictivos.

2.3. Componentes de un ataque de phishing

2.3.1. Mecanismo

Un correo electrónico de suplantación de identidad (o phishing) es un correo electrónico de apariencia legítima que está diseñado para engañar al destinatario y hacerle creer que se trata de un correo genuino. En general el objetivo del correo de phishing consiste en que el destinatario revele información confidencial o descargue un software malicioso, ya sea a través de un archivo adjunto o haciendo clic en los enlaces maliciosos que se encuentran en el cuerpo del correo electrónico.

Los correos electrónicos de tipo phishing normalmente contienen elementos gráficos, de texto o de diseño que engañan al usuario para que piense que proviene de una fuente confiable. En general, hay muy pocos elementos de un correo electrónico en los que se pueda confiar, especialmente sin examinar

la información del encabezado del correo, ya que el remitente del mismo se puede cambiar fácilmente, y los mensajes de correo electrónico se pueden simular con los mismos elementos de diseño que la organización falsificada. El uso de correos electrónicos seguros con firmas digitales al día de hoy no ha despegado realmente, y por lo tanto hay muy pocas señales para identificar de forma inequívoca a un correo electrónico de phishing [2].

Es muy común que durante un ataque de phishing por correo electrónico, el atacante dirija a la víctima a un sitio web fraudulento, que simula en apariencia a un sitio legítimo de una organización, empresa, servicio o persona conocido por la víctima. El comercio electrónico, los servicios de pago en línea y las redes sociales son los más afectados por este ataque. Su efectividad radica en que logra aprovechar las similitudes gráficas entre el sitio falso desarrollado por el atacante y el sitio original.

El mecanismo de un ataque de suplantación de identidad se muestra en la figura 1. En este ejemplo se genera un sitio web falso que es el clon de un sitio web genuino objetivo y contiene algunos campos de entrada con información confidencial. Cuando el usuario ingresa sus datos personales, la información se transfiere al atacante.

El primer paso del ataque consiste en la construcción del sitio de phishing. El atacante identifica a una organización conocida y recopila la información detallada sobre esa organización visitando su sitio web. Luego usa esta información para construir el sitio web falso. Incluso puede copiar el código html y las imágenes del sitio original y armar un sitio visualmente idéntico.

El siguiente paso consiste en el envío de la URL del sitio falso a la víctima. En este paso, el atacante redacta un correo electrónico falso y lo envía a miles de usuarios. La URL del sitio web falso se incluye en el cuerpo del mensaje. En el caso del ataque de phishing dirigido, el atacante envía el correo electrónico a una lista de usuarios seleccionados. También es posible difundir el enlace del sitio web de phishing con la ayuda de blogs, foros, etc.

Por último, se produce el robo de las credenciales. Cuando el usuario hace clic en la URL adjunta en el correo, se abre el sitio falso en el navegador web. Este sitio contiene un formulario de inicio de sesión que se utiliza para tomar las credenciales que ingresa la víctima. El atacante luego puede usar estas credenciales para acceder al sitio real de la organización y robar la información confidencial del usuario. [3]

Figura 1: Ataque de phishing (suplantación de sitio web)

2.3.2. Componentes de un correo

A continuación, se exponen la sintaxis para mensajes de correo electrónico tal cual se describen en el RFC822 [4], que será de importancia en el análisis de todo tipo de correos electrónico, incluidos los correos de phishing. Un correo electrónico está formado por dos partes: el sobre y el contenido. El sobre contiene cualquier información necesaria para lograr la transmisión y entrega del correo. El contenido se compone de dos partes: el encabezado y el cuerpo, y contiene la información que se entregará al destinatario. Notar que tanto el sobre como el encabezado contienen información sobre quién es el remitente del correo, pero la misma puede ser diferente. Esta característica puede ser aprovechada en un ataque de suplantación de identidad. La especificación en el RFC882 aplica solo al formato y semántica de parte de la información en el encabezado del correo electrónico. No contiene ninguna especificación de la información en el sobre.

A continuación, una lista de los campos del header que establece el RFC y que serán de importancia para analizar un correo de suplantación de identidad.

From: especifica el autor del mensaje (persona o sistema responsable de escribir el mensaje). Debe ser una dirección válida de correo.

Sender: contiene información del agente (persona o sistema) que envía el

mensaje. Está pensado para usarse cuando el autor es distinto al agente que envía el mensaje. No es obligatorio si ya se encuentra el campo From.

Date: Fecha de envío del mensaje.

Reply-To: es configurado por el emisor del mensaje y sirve como la dirección para la respuesta del mensaje.

Return-path: es agregada por el sistema que entrega el mensaje al destinatario y contiene la dirección del originador del mensaje.

Received: cada servicio de transporte que redirige el mensaje agrega una nueva linea con este campo. Esta información puede resultar muy útil para realizar un seguimiento del correo.

Message-ID: Identificador único del mensaje.

Otros campos importantes de autenticación son descriptos en el RFC7208 [5] y el RFC7001 [6]. Los que consideramos más relevantes son los campos Authentication-Results y Received-SPF que serán comentados en el siguiente capítulo.

2.3.3. Autenticación de correos

Para evitar que un atacante se haga pasar por un dominio legítimo utilizando el mismo nombre de dominio, existen protocolos de autenticación para correos electrónicos tales como SPF, DKIM y DMARC.

El Sender Policy Framework, o SPF, se basa en el DNS del nombre del dominio y puede certificar si la dirección IP de origen del mensaje tiene el derecho de enviar un correo con ese dominio. Este protocolo se utiliza para evitar el uso fraudulento de un nombre de dominio y evitar ataques de suplantación de identidad. Los detalles de la validación del SPF se incluyen en el campo Received-SPF del encabezado del correo. [7]

El DomainKeys Identified Mail, o DKIM, es un protocolo de autenticación criptográfico basado en el uso de claves públicas que se publican en el DNS. Este protocolo permite firmar los correos de forma tal que el destinatario pueda asegurarse que proviene de un origen legítimo. El objetivo del protocolo DKIM no es sólo evitar la suplantación de identidad, sino también demostrar que el mensaje no ha sido alterado durante la transmisión. [7]

El Domain-based Message Authentication, Reporting and Conformance, o DMARC, es un estándar de autenticación que complementa a SPF y DKIM para combatir más eficazmente el phishing. Ambos protocolos DKIM y SPF son complementarios, sin embargo no dan instrucciones de acción en caso de un ataque. El protocolo DMARC supera esta deficiencia y proporciona indicaciones sobre qué hacer cuando un mensaje no pasa las pruebas de los protocolos de SPF o DKIM. Por ejemplo, la política de “none” sugiere no tomar ninguna acción, y se utiliza solo para monitoreo, la política “quarantine” sugiere mover los mensajes que fallan las pruebas de SPF y DKIM a la cuarentena, y la política de “reject” directamente rechazar todos los mensajes que fallen las pruebas. Muchos proveedores de servicio de correo también suelen enviar informes a los propietarios de los dominios que tienen configurado DMARC con los correos que recibieron y fallaron las validaciones. [7]

Las validaciones de SPF, DKIM y DMARC se detallan en un campo del encabezado del correo con nombre Authentication-Results, y se pueden utilizar para identificar un correo de phishing.

2.3.4. Componentes de un sitio web

Un componente clave en un ataque de phishing es el sitio falso que genera el atacante. Este sitio es una copia de una página legitima que se utiliza para engañar a la víctima para que ingrese la información confidencial. Se pueden identificar cuatro características que pueden usarse para identificar un ataque de phishing a través de un sitio web (ver figura 2). [8]

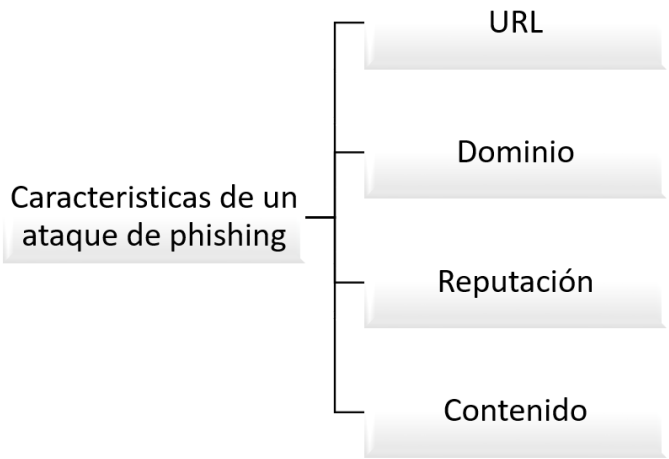

Figura 2: Características generales de un sitio web de phishing

El primer componente es la URL de la página fraudulenta. Esta URL puede ser reconocida analizando su longitud, la correcta escritura de la misma, y si se encuentra relacionada a una marca existente o no. La segunda característica es el nombre de dominio del sitio. En este caso se puede analizar el estado del dominio, si se encuentra en alguna lista negra de dominios maliciosos, la fecha de creación y el propietario del dominio. El tercer punto que se puede analizar es la reputación del sitio. Esta reputación es generalmente determinada por un ranking global de sitios, que obtienen información como actividad de los usuarios, número de visitantes por día, semana o mes, categoría del dominio, entre otros. Por último, se puede identificar el contenido del sitio, cómo por ejemplo, el título, las etiquetas de los meta-datos, texto escondido en el código html, texto en el cuerpo de la página, y las imágenes en el sitio, y los formularios de login.

2.4. Técnicas de aprendizaje automático

El aprendizaje automático o machine learning es una técnica científica en la que las computadoras aprenden a resolver un problema, sin programarlas explícitamente. En otras palabras, se encarga de representar la estructura y generalizar comportamientos en los datos dados. Se basan en crear un modelo a partir de la información suministrada para poder generar conclusiones ante casos nuevos.

En los problemas de clasificación como del que trata esta investigación es muy utilizado el aprendizaje automático supervisado. En este tipo de machine learning, los algoritmos trabajan con datos “etiquetados”, con el objetivo de encontrar una función que, dadas las variables de entrada, encuentre la etiqueta que le corresponde. El algoritmo se entrena con un conjunto de datos de entrenamiento o de prueba que ya se encuentran clasificados. El mismo aprende de estos datos los patrones para poder clasificar casos nuevos con la etiqueta que les corresponda.

Se puede explicar también de la siguiente forma. Supongamos que disponemos de n ejemplos observados en el mundo real, $\{e_1, \ldots, e_n\}$, que vienen definidos a partir de un conjunto de atributos o propiedades (también llamados predictores), $e_i = (p_{i_1}, \ldots, p_{i_m})$, y para cada uno de ellos tenemos una clasificación observada, c_i. El objetivo del algoritmo de aprendizaje automático consiste en recibir esa información de entrada y generar un modelo de tal

forma que ante un nuevo caso en el mundo real, $e_{n+1} = (p_{n+1_1}, ..., p_{x+1_m})$, el modelo deberá encontrar o predecir la clasificación que le corresponde observando sus propiedades.

Existen diversos algoritmos de aprendizaje automático que son muy usados y conocidos por su gran precisión y rendimiento. A continuación, veremos algunos que son muy usados en problemas de clasificación y que serán analizados para el problema de detección de sitios y correos de phishing a lo largo de este trabajo. [8]

2.4.1. Árboles de decisión

Un árbol de decisión está formado por un conjunto de nodos de decisión (interiores) y de nodos-respuesta (hojas). Un nodo de decisión está asociado a uno de los atributos y tiene 2 o más ramas que salen de él, cada una de ellas representando los posibles valores que puede tomar el atributo asociado. Los nodos-respuesta está asociado a la clasificación que se quiere proporcionar, y devuelve la decisión del árbol con respecto al caso de entrada.

De esta forma se puede entender a un árbol de decisión como una encuesta en donde cada nodo interior corresponde a una pregunta, cada respuesta es una rama, y las hojas son la respuesta final del algoritmo.

Evidentemente, la construcción del árbol de decisión no es única, y si aplicamos una estrategia u otra a la hora de decidir en qué orden se hacen las preguntas sobre los atributos podemos encontrar árboles muy dispares. De entre todos los posibles árboles, estamos interesados en encontrar aquellos que cumplan las mejores características como máquinas de predicción.

El ID3 es un algoritmo utilizado para construir un árbol de decisión basándose únicamente en los casos iniciales proporcionados por el conjunto inicial de datos de prueba. Para ello, usa el concepto de Ganancia de Información para seleccionar el atributo más útil en cada paso. De esta forma en cada paso del algoritmo decide la pregunta que permite separar mejor los casos respecto a la clasificación final. El resultado final del algoritmo es un árbol que sirve de modelo para predecir la clasificación de futuros casos. [9]

2.4.2. Algoritmo de bosques aleatorios

Random Forest o bosques aleatorios es una técnica de aprendizaje automático supervisada basada en árboles de decisión. Su principal ventaja es que obtiene un mejor rendimiento de generalización para un rendimiento durante un entrenamiento similar. Esta mejora en la generalización la consigue compensando los errores de las predicciones de los distintos árboles de decisión. [11]

De forma más precisa, un Random Forest es un conjunto de árboles de decisión combinados para crear un único resultado. Ningún árbol ve todos los datos de entrenamiento, sino que ven distintas porciones de los mismos. Estos subconjuntos se forman eligiendo muestras aleatorias (con repetición) del conjunto de entrenamiento. De esta forma cada árbol se entrene con distintas muestras de datos para un mismo problema. Al combinar sus resultados, unos errores se compensan con otros y tenemos una predicción que generaliza mejor.

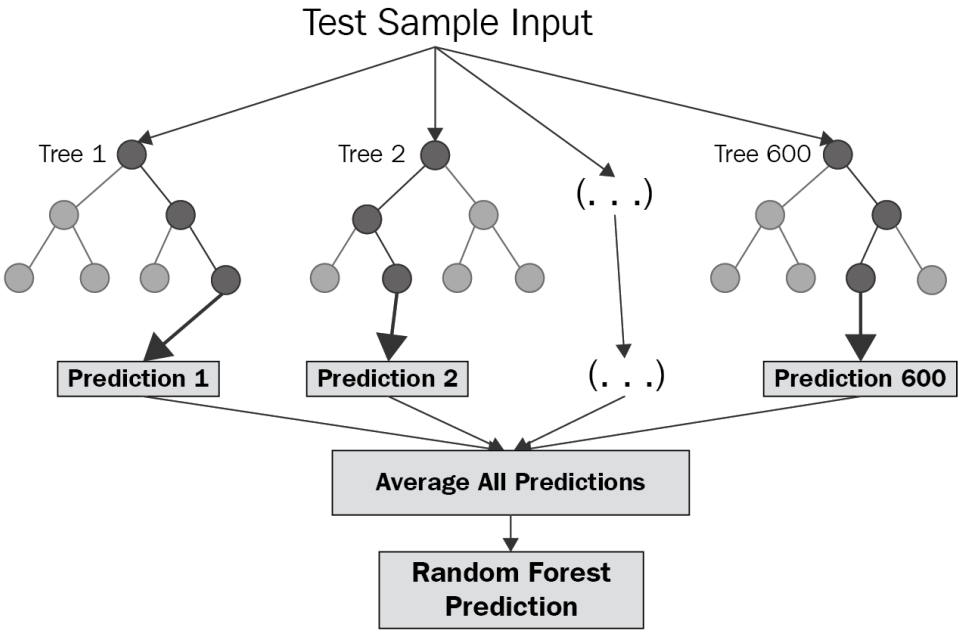

Figura 3: Esquema de un bosque aleatorio de 600 árboles [12]

2.4.3. Clasificadores Bayesianos

El algoritmo Naive Bayes, también conocido como clasificador Bayesiano, es un grupo de algoritmos de clasificación basados en el teorema de Bayes.

Estos algoritmos están basados en una clasificación probabilista que asume una fuerte independencia entre las distintas variables del modelo. Es decir, comparten el principio de que el valor de cada variable o característica del modelo es independiente del valor de las demás, de tal forma que la presencia de una característica particular no afecta a la otra. La ecuación general del teorema de Bayes es la siguiente

$$P(x|Y) = \frac{P(Y|x)P(x)}{P(Y)} \tag{1}$$

Donde $P(x)$ es la probabilidad independiente de x, $P(Y)$ es la probabilidad independiente de y, $P(Y|x)$ es la probabilidad condicional de Y dado x, y de igual manera, $P(x|Y)$ es la probabilidad condicional de x dado y. [8]

En general la variable Y corresponde a un vector de las n características que se utilizan en el modelo, es decir $Y = (y_1, y_2, ..., y_n)$. Por la tanto la fórmula puede reescribirse como:

$$P(x|y_1, y_2, ..., y_n) = \frac{P(y_1|x)P(y_2|x)...P(y_n|x)P(x)}{P(y_1)P(y_2)...P(y_n)} \tag{2}$$

En términos más generales, usando el teorema de Bayes, podemos encontrar la probabilidad de que ocurra A, dado que B ya ha ocurrido. De esta forma, B es la evidencia y A es la hipótesis.

Utilizando por ejemplo el problema de determinar si un sitio es legítimo o malicioso, la variable x será si el sitio es malicioso o no, y el vector Y serán todas las características del sitio que se utilizarán para determinar si es un sitio legítimo o no. Hay que tener en cuenta que esas características deben ser independientes. Luego las probabilidades para completar la formula se pueden obtener del conjunto de datos de prueba. Mientras mayor sea el conjunto de pruebas, mayor sera la precisión de la probabilidad calculada.

Existen distintos tipos de clasificadores Bayesianos, de acuerdo a la naturaleza del vector de variables. Si las variables toman valores binarios el clasificador es de tipo Bernoulli Naïve Bayes. En cambio, si pueden tomar valores enteros, se lo denomina un clasificador Multinomial Naive Bayes, y por último si las variables pueden tomar valores continuos (es decir no discretos) y se asume que corresponde a una distribución Gaussiana, el clasificador es de tipo Gaussian Naive Bayes.

Los clasificadores Bayesianos en general son rápidos y fáciles de implementar, pero su mayor desventaja es que las variables deben ser independientes. En la mayoría de los casos de la vida real, los predictores son dependientes, lo que dificulta el rendimiento del clasificador.

2.4.4. Máquinas de vectores de soporte

Las máquinas de soporte vectorial, máquinas de vectores de soporte o máquinas de vector soporte (Support Vector Machines, SVMs) son un conjunto de algoritmos de aprendizaje supervisado. Estos algoritmos representan a cada muestra del conjunto de entrenamiento como vectores en un espacio de tantas dimensiones como propiedades de las muestras. La idea principal detrás de SVM es encontrar un plano óptimo que separe las muestras en dos clases maximizando el margen. Un margen es la distancia entre los puntos de datos más cercanos de dos clases diferentes. Cada dato se representa como un vector y los puntos de datos más cercanos de dos clases diferentes se denominan vectores de soporte (ver figura 4).

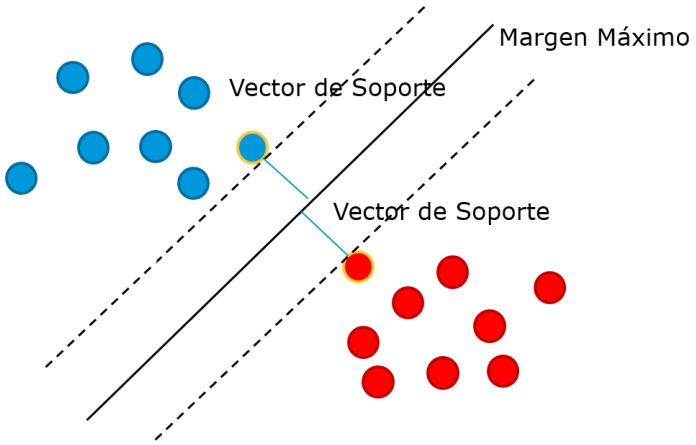

Figura 4: Ejemplo de los elementos de un SVM en dos dimensiones [13]

Hay veces en las que no hay forma de encontrar un hiperplano que permita separar dos clases. En estos casos decimos que las clases no son linealmente separables. Para resolver este problema podemos usar el truco de Kernel.

El truco de Kernel consiste en inventar una dimensión nueva en la que podamos encontrar un hiperplano para separar las clases. En la figura 5 vemos cómo al añadir una dimensión nueva, podemos separar fácilmente las dos clases con una superficie de decisión. Para crear la nueva dimensión se usa una función llamada función de Kernel. Entre las funciones de Kernel utilizadas

más frecuentemente se encuentran las polinomiales y las de base radial.

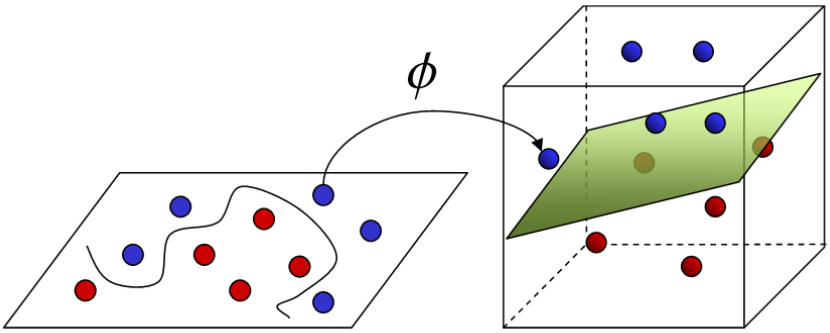

Figura 5: Truco de Kernel[14]

SVM ofrece diferentes beneficios frente a otros algoritmos de aprendizaje automático. Una de ellas es que proporciona un claro margen de separación y funciona realmente bien tanto para datos linealmente separables como inseparables. Además, es bastante eficiente cuando tenemos datos con una gran cantidad de características. Es especialmente efectivo para los problemas de clasificación en los que se tienen pocos datos, pero la cantidad de características o variables es muy alta. Una vez entrenado el modelo, para clasificar un nuevo caso en muy eficiente. Sin embargo, SVM no funciona bien cuando el conjunto de datos de entrenamiento es relativamente grande ya que el tiempo de entrenamiento es mayor. Del mismo modo, su rendimiento se degrada cuando el conjunto de datos de prueba tiene mucho ruido o muchos casos mal categorizados.

3. Desarrollo

3.1. Técnicas de detección

Identificar ataques de phishing es una tarea desafiante debido a que estos ataques intentan explotar vulnerabilidades humanas, y no errores de un sistema. Existen diversos indicios que pueden llevar a un usuario a identificar un correo o sitio de phishing fácilmente, pero requiere capacitar y concretizar al usuario, el cual no siempre estará del todo atento a estos indicadores. [8]

Los primeros métodos de detección automática de correos o sitios de phishing se basaban en listas blancas o negras. Una lista negra es una lista de dominios que son considerados maliciosos y por lo tanto deberían ser bloqueados. Por otro lado, todos los dominios que no se encuentren en la lista negra serán permitidos. Una lista blanca es una lista de dominios que son considerados como legítimos y por lo tanto se permitirá su acceso. En este caso, los dominios que no se encuentren en la lista blanca serán bloqueados.

Los métodos de listas negras y blancas presentan diversos problemas que veremos más adelante. Por esa razón se presenta la necesidad de investigar nuevas formas de detección. Detectar un ataque de phishing puede ser considerado como un problema de clasificación, en donde el sitio o correo necesita ser clasificado como legítimo o no. De esta forma, parece que usar métodos de aprendizaje automático son apropiados para aplicar ante un posible caso de ataque de phishing, ya que estos métodos son capaces de resolver de forma efectiva problemas de clasificación.

Para detectar un ataque de phishing, los algoritmos de machine learning entrenará un modelo de clasificación con algunos atributos o reglas para determinar si el sitio será clasificado como phishing o no. Por lo general, el algoritmo funciona mediante la extracción de las características de una URL o el contenido de una página web y forma un modelo de predicción basado en las características como las que se discutieron en la introducción antes de decidir si la página web es legítima o falsa.

3.2. Análisis manual de correos y sitios de phishing

Los correos y mensajes de phishing son creados de tal forma que tengan el mismo aspecto que los mensajes enviados por una compañía conocida, como puede ser un banco, una compañía de tarjetas de créditos, un sitio de redes sociales, o una tienda en línea. Sin embargo, y a pesar de que los estafadores suelen actualizar constantemente sus tácticas, existen ciertos indicios que se repiten comúnmente en los mensajes de phishing que pueden ayudar a reconocer este tipo de ataque.

A menudo, los correos o mensajes de phishing utilizan amenazas, advertencias, o regalos para engañar al receptor del mensaje para que descargue un archivo adjunto o haga clic en algún enlace. Es importante sospechar de todos los mensajes que indiquen que se ha detectado alguna actividad sospechosa en su cuenta y solicite cambiar las credenciales desde un enlace en el correo. Nunca se deben incluir datos personales, contraseñas o información confidencial en la respuesta a este tipo de correos, ni hacer clic en enlaces o bajar archivos adjuntos. Por ejemplo, si recibe un correo de su banco indicando que existe un problema con su cuenta, nunca se debe hacer clic en el enlace del correo, sino ingresar al sitio del banco desde un enlace guardado y confiable.

Existen otros indicios para reconocer un correo de phishing. Una táctica de phishing muy usada entre los cibercriminales es falsificar el nombre para mostrar de un correo electrónico. Siempre se debe comprobar la dirección de correo electrónico de origen y chequear que su dominio se encuentre bien escrito. En muchos casos, al ser un dominio falso, las comprobaciones de DMARC realizadas para ese dominio no son efectivas, ya que no aplican al dominio legítimo.

También es importante chequear cada enlace del correo y verificar que el sitio web al que redirige sea el correspondiente al sitio legítimo de la empresa u organización.

Los delincuentes saben lo que hacen y el hecho de que un correo electrónico tenga logotipos de marca, imágenes y o dominios de correo electrónico similares o iguales a los de la compañía legitima, no significa que el correo sea legítimo. [15]

Otras formas más avanzadas de identificar a los correos de phishing con-

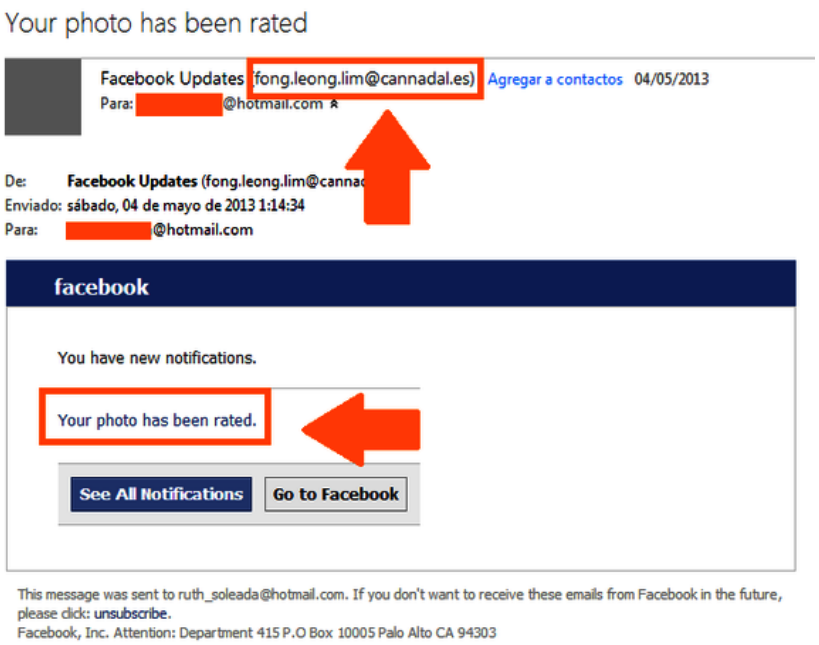

Figura 6: Ejemplo de phishing desde una dirección de correo que no corresponde a la empresa [32]

Figura 7: Ejemplo de phishing en donde se redirige a un sitio fraudulento [33]

sisten en analizar distintos campos del encabezado del correo. Es importante notar que campos como From y Sender no son confiables ya que pueden ser

modificados muy fácilmente.

Las líneas del encabezado que comienzan con Received proporcionan un seguimiento del correo electrónico desde su origen a su servidor de correo. De esta forma puede resultar muy útil para identificar la legitimidad de la fuente del correo.

El campo Return-Path también resulta útil en este tipo de análisis ya que, por lo general, en los correos electrónicos masivos, este campo es diferente al del registro From. De forma similar, es importante revisar la dirección de correo en el campo Reply-To, ya que cuando se presiona responder a un correo electrónico, esa dirección se usa para completar el correo electrónico de los destinatarios, por lo que podría estar respondiendo el mensaje al correo del atacante. [16]

Los campos de validación del SPF, DKIM y DMARC pueden ser extremadamente útiles en los casos en los que el atacante copie exactamente el mismo dominio legítimo para enviar los correos. Sin embargo, esto no siempre pasa, por lo que no es posible confiar en estas comprobaciones sin estar seguro que el dominio de origen sea legítimo. Esta es una de las razones por las que se necesitan otros métodos de detecciones de correos de suplantación de identidad, como los que se verán en los próximos capítulos.

Pero no solo los correos electrónicos maliciosos se utilizan para engañar a las personas para que hagan clic en los enlaces o divulguen información confidencial. Otra táctica común utilizada por los delincuentes implica la creación de sitios web falsos para engañar a las víctimas para que ingresen información confidencial. Si bien muchos sitios parecerán casi indistinguibles de los reales, existen una serie de signos sutiles a tener en cuenta que pueden indicar un sitio web de phishing.

Uno de los puntos más importante consiste en verificar la validez de la dirección web. No solo se debe comprobar que el dominio del sitio sea legítimo y se encuentre correctamente escrito, sino también que el sitio utilice un protocolo seguro, como HTTPS, lo cual indica que la dirección web ha sido encriptada y protegida con un certificado SSL. Sin HTTPS, cualquier dato transmitido en el sitio es inseguro y podría ser interceptado por terceros criminales. Sin embargo, este sistema no es totalmente infalible, y en los últimos años, ha habido un aumento notable en el número de sitios de phishing que utilizan certificados SSL. [17]

Existen otros indicios como los errores de ortografía, los errores gramaticales o las imágenes de baja resolución, que pueden servir como una señal de alerta de que el sitio puede ser fraudulento.

Figura 8: Ejemplo de sitio de phishing con una url falsa [32]

3.3. Sistemas de detección automáticos basados en listas blancas

Los sistemas basados en listas blancas utilizan una lista de dominios que son considerados como legítimos y por lo tanto se permitirá su acceso. Todos aquellos dominios que no se encuentren en la lista serán bloqueados.

En el 2008, Cao, Han y Le presentaron un sistema que creaba una lista blanca de IP registradas para cada sitio con un formulario de autenticación que visitaba el usuario [18]. Cuando el usuario visitaba un sitio, si el mismo no se encontraba en la lista blanca, el sistema le mostraba una advertencia indicando que el sitio era desconocido. Este método presenta dos inconvenientes, el primero es que considera sospechosos todos los sitios la primera vez que es accedido por el usuario. El segundo problema surge cuando un sitio utiliza un rango de direcciones IPs o cambia arbitrariamente su IP asociada por determinada razón, como, por ejemplo, para balancear la carga del mismo.

Jain y Gupta presentaron un método de detección de sitios de phishing en

2016 que utilizaba una lista blanca de sitios legítimos que era actualizada de forma automática [19]. El método consistía de dos módulos. Uno se encargaba del emparejamiento de la IP con el sitio web, y el otro de la extracción de información de los enlaces en el código fuente del mismo.

3.4. Sistemas de detección basados en listas negras

Las listas negras se crean mediante registros de URL o IPs que se conocen como sitios web de phishing. Estas entradas de la lista se derivan de varias fuentes, como sistemas de detección de spam, notificaciones de usuarios, organizaciones de terceros, etc. El uso de listas negras hace que sea imposible que los atacantes ataquen nuevamente a través de la misma URL o dirección IP, que se usaron anteriormente para el ataque.

Un mecanismo actualiza las listas negras detectando direcciones URL o IP maliciosas o los usuarios pueden descargar estas listas instantáneamente desde un servidor y proteger sus sistemas contra los ataques enumerados en esta lista.

Sin embargo, los sistemas basados en la lista negra no tienen la capacidad de detectar un ataque real o un ataque por primera vez (ataque de día cero). Estos mecanismos de detección de ataques tienen una tasa de falsos positivos más baja que los sistemas basados en aprendizaje automático.

El éxito del sistema de detección de ataques de phishing basado en la lista negra es de aproximadamente el 20 % [20]. Por lo tanto, parece que los sistemas basados en listas negras no son eficientes como un mecanismo confiable de detección de ataques. Algunas empresas prestan servicios a sistemas de detección de ataques de phishing basados en listas negras, como la API de navegación segura de Google o el sitio PhishNet. Estos sistemas utilizan un algoritmo de coincidencia aproximado para verificar si la URL sospechosa existe en la lista negra o no. Los enfoques basados en la lista negra requieren actualizaciones frecuentes. Además, el rápido crecimiento de la lista negra requiere recursos excesivos del sistema. [21]

3.5. Sistemas de detección basados en aprendizaje automático

Las técnicas de inteligencia artificial y aprendizaje automático pueden proveer una forma eficiente de detección de correos o sitios web de phishing. Estás técnicas mejoran notablemente el porcentaje de detección para los ataques de día cero de phishing, comparadas a las técnicas tradicionales más manuales como las listas blancas y negras.

En general, para crear un modelo, los diversos algoritmos de aprendizaje automático se entrenan mediante el uso de un conjunto de datos de entrenamiento. Una vez completada esa etapa, se prueba el modelo construido utilizando un conjunto nuevo de datos de entrada, creando predicciones sobre cada uno de los datos de entrada nuevos. Luego la predicción generada se evalúa para verificar su precisión.

Si la precisión estimada se encuentra dentro del rango tolerable, entonces se utiliza el modelo construido para el algoritmo. De lo contrario, se utilizan distintos parámetros de entrada o bien un conjunto mejorado de datos de entrenamiento, y el algoritmo de aprendizaje automático se entrena otra vez desde el primer paso.

De esta forma se construye un modelo de machine learning con una precisión aceptable que sirva para predecir futuros casos que generalmente no se encuentran en la muestra utilizada para construirlo. [22]

A continuación, se exponen diferentes sistemas de detección tanto de correos como de sitios de phishing. En general un sistema de diferencia de otro básicamente por el conjunto de características o componentes del sitio o correo que alimenta el modelo, y el algoritmo de aprendizaje automático elegido. [20]

Para evaluar la efectividad de estos algoritmos se utilizan los siguientes índices:

$$\text{Índice TP} = \frac{\text{núm. de sitios de phishing detectados}}{\text{núm. total de sitios de phishing}} \tag{3}$$

$$\text{Índice FN} = \frac{\text{núm. de sitios de phishing no detectados}}{\text{núm. total de sitios de phishing}} \tag{4}$$

$$\text{Índice TN} = \frac{\text{núm. de sitios legítimos reconocidos}}{\text{núm. total de sitios legítimos}} \tag{5}$$

$$\text{Índice FP} = \frac{\text{núm. de sitios legítimos detectados como phishing}}{\text{núm. total de sitios legítimos}} \tag{6}$$

El índice TP, también se puede denominar precisión, es el número de sitios o correos de phishing detectados como tal, sobre el total de los sitios o correos de phishing. Mientras más alto sea este valor, mejor será el algoritmo o modelo construido.

Los índices FP y FN son los índices de falsos-positivos y falsos-negativos obtenidos respectivamente. El objetivo es mantener estos valores lo más bajo posible.

3.5.1. Sistemas basados en Random Forest

Amani Alswailem, Bashayr Alabdullah, Norah Alrumayh y Aram Alsedrani [23] presentaron en 2019 un sistema basado en un algoritmo de Random Forest para detectar sitios de phishing. El sistema actuaba como un complemento de un navegador de Internet que automáticamente notificaba al usuario cuando detectaba un sitio de phishing. Estudiaron variaciones del algoritmo utilizando un total de 36 propiedades de la url, contenido y ranking del sitio (ver figura 9), con el propósito de encontrar la combinación más poderosa que provea un buen rendimiento en términos de tiempo y poder de computo. El conjunto de datos de entrenamiento y prueba incluía 12000 sitios de phishings recolectados del sitio PhishTank y 4000 sitios legítimos que fueron recolectados de la navegación diaria de 10 usuarios. El conjunto final luego de remover información perdida o duplicada, consistía de 6116 urls.

Al finalizar el estudio, la mejor precisión con el menor número de propiedades las obtuvieron utilizando 26 de las 36 propiedades (ver figura 10) logrando una precisión del 98.8 %.

En 2014, Akinyelu1 y Adewum [30] lograron una precisión de 99,7 % al detectar correos electrónicos de Phishing. En este trabajo se utilizando 15 características de los correos como entrada para construir el modelo.

Una de las caracterizaras que incorpora el modelo es si existen enlaces

Features Based on		
URL	Length of URL	Length of hostname of URL
	Length of the path of URL	Number of dot (.) in the path
	Number of dot (.) in hostname	Number of slashes (/) in URL
	Number of hyphen (-) in hostname	Number of special characters (: ; % & ? +)
	Number of at (@) in the URL	Number of digit in hostname
	Number of underscore (_) in hostname	Number of underscore (_) in path
	Number of certain keyword in URL	Number of hexadecimal with %
	Transport layer security	IP address
	Presence of www	Port redirect
	Unicode in URL	Hexadecimal characters
Page content	Number of forms	Number of forms with action 'GET'
	Number of forms with action 'POST'	Number of script
	Number of outer src script	Number of $< Iframe >$
	Number of $< Applet >$	Number of $< Embed >$
	Number of $< Frame >$	Number of link
	Number of non-link	Number of submit
	Number of input email	Number of input password
	Number of button	
Rank	Alexa rank	Age of domain

Figura 9: Propiedades de un sitio [23]

en el correo que contienen una dirección IP. La URL de muchos sitios web legítimos generalmente contiene el nombre del sitio web. la presencia de las URL basadas en IP en un correo electrónico es una indicación de que el correo electrónico es un posible correo electrónico de phishing.

También si existen disparidades entre el sitio que redirige un enlace y el texto del enlace. El texto del enlace podría ser un texto plano, una URL, una imagen, o cualquier otro elemento. Si el texto del enlace es una URL, debería coincidir con la ubicación del sitio web a la que redirige el enlace.

La presencia de determinadas palabras también puede indicar la presencia de un correo de phishing. El texto de los enlaces presentes en la mayoría de los casos de phishing contienen palabras como “Hacer clic”, “Aquí”, “Iniciar sesión” y “Actualizar”.

La cantidad de puntos en el nombre de un dominio también puede utilizarse

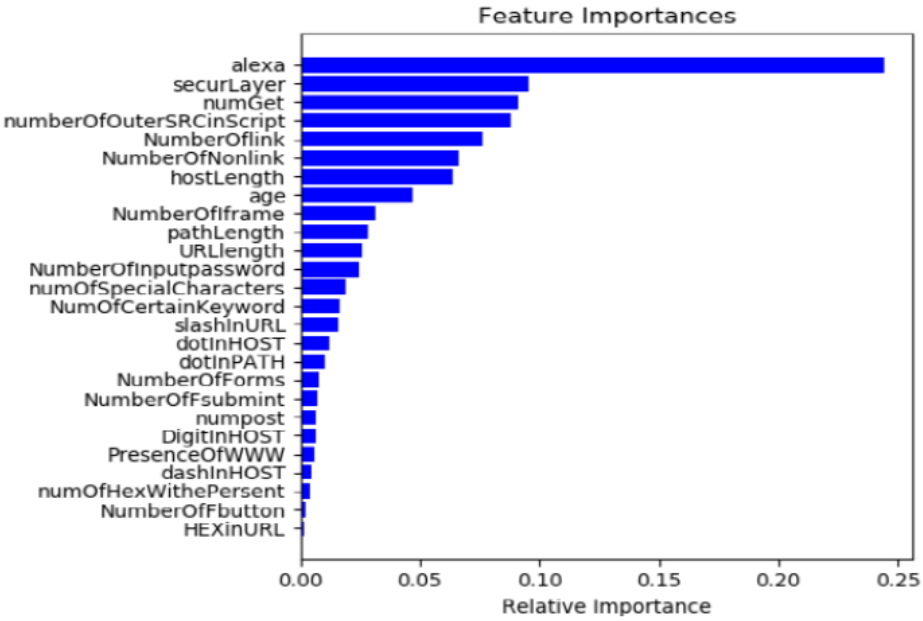

Figura 10: Propiedades elegidas [23]

como un indicador de que puede no ser legítimo. Otras características como la presencia de código javascript, si el correo está en texto plano o html, el número de enlaces, y dominios distintos en los enlaces de los correos, también se utilizaron en el modelo para clasificar los correos.

Los resultados de este algoritmo fueron una precisión del of 99.7 %, una tasa FN de 2.50 % y una tasa de FP del 0.06 %.

3.5.2. Sistemas basados en Support Vector Machine

Che-Yu Wu, Cheng-Chung Kuo, Chu-Sing Yang [24] propusieron un método de detección basado en un algoritmo SVM, debido a su precisión en muestras pequeñas y en problemas de decisión binaria. El sistema se basa en dos características que comparten un gran número de sitios de phishing. La primera es que los sitios de phishing en general utilizan URLs similares a los sitios legítimos, pero variando algunos caracteres. La segunda es que una vez el usuario ingresa su información confidencial, para prevenir que la víctima sea consciente del ataque, el sitio fraudulento transmite la información proporcionada al sitio legítimo y redirige a la víctima a ese sitio, para que el usuario pueda navegar normalmente, y el robo permanece sin detectar. En otros casos los sitios de phishing también pueden usar apis, enlaces o imágenes que corresponden al sitio legítimo, para asemejarse lo más posible al sitio.

Basándose en esas premisas, el algoritmo busca URL similares a la del sitio de phishing que puedan corresponder a la URL del sitio legítimo, implementándose con una arquitectura como se ve en la figura 11. con tres componentes principales: el web crawler, el parser y la SVM.

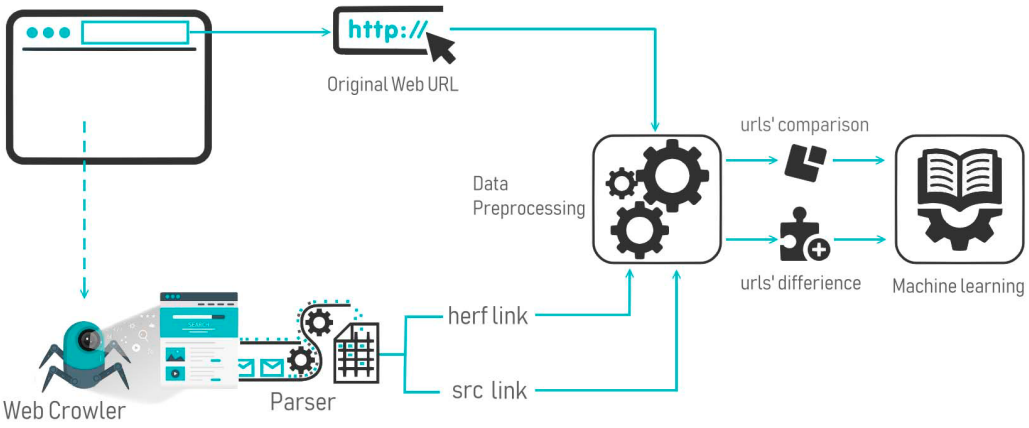

Figura 11: Arquitectura del sistema [24]

El crawler es el componente encargado de encontrar contenidas en el sitio web. En primer lugar, rastrea si el link redirige a otra página web y toma esa dirección de destino como la primera URL. Luego utiliza una herramienta llamada Seleniuym diseñada para automatizar la navegación a través de sitios web, de esta forma se comporta como un navegador real, accediendo al contenido en tiempo real y encontrando las URL a las que redirige el sitio. También recolecta todas las URL que se encuentran en las etiquetas de html como href y src.

El parser es el componente que procesa las URL encontradas por el crawler en diferentes componentes, extrayendo su dominio, subdominio, y path. Luego compara la diferencia entre la URL principal y cada una de las URLs encontradas. Para determinar si dos URL son parecidas, utiliza la distancia de edición. Esta distancia de similitud entre dos cadenas de texto consiste en calcular la mínima cantidad de operaciones necesarias para pasar de una cadena a la otra. Dentro de las variaciones de esta técnica, el estudio utiliza las operaciones determinadas por la distancia Levenshtein, que trabaja con inserciones, borrados o substituciones de caracteres para pasar de una palabra a la otra. Entonces calcula la similitud de acuerdo a la distancia Levenshtein entre el dominio principal y el dominio de todas las URL encontradas en el sitio web. Luego divide a las URL en tres categorías: completamente igual, alta similitud y baja similitud. Por último, calcula la proporción de cada categoría con respecto a todas las URL.

El algoritmo de aprendizaje automático utiliza como entrada las proporciones de cada categoría para los dominios, subdominios y path de las URL encontradas y las carga en un modelo de SVM. La investigación también utilizó algoritmos de árboles de decisión y regresión logística, pero encontró que el modelo de SVM funcionaba mejor en muestras pequeñas como las del estudio.

Finalmente, el experimento se realizó sobre un conjunto de entrenamiento con un total de 15000 sitios web, de los cuales 5000 corresponden a sitios identificados como phishing por Phishtank y los 10000 restantes son sitios recolectados a través de DMOZ, un directorio de contenido abierto de sitios web. El resultado obtenido fue una precisión del 89,3 % al detectar los sitios de phishing con una tasa de falsos positivos del 6,2 %. Al analizar este fenómeno, encontraron que algunas páginas de phishing utilizan una foto del sitio imitado, para evitar ser detectado por herramientas de análisis de texto. Debido a que estos sitios deberían ser detectados por herramientas de análisis de imágenes, fueron excluidos de la muestra, dando una precisión de 92,6 %.

En el artículo “URL Phishing Data Analysis and Detecting Phishing Attacks using Machine Learning in NLP” [25] también utilizan un SVM, pero en este caso los componentes usados para construir el modelo se basan en la longitud de la URL del sitio, su dirección IP, sub-dominios, si utiliza un protocolo seguro o no (http o https), símbolos sospechosos usados dentro de la URL como puede ser el @, y la cantidad de tráfico hacia el sitio (obtenido del ranking de la base de datos de Alexa). De esta forma encontraron un algoritmo muy eficiente en términos de tiempo en comparación con otros mencionados en el artículo, sin embargo, no se aclara la precisión del mismo.

3.5.3. Un clasificador de correos basado en clasificadores bayesianos

En un artículo publicado por la Universidad de Firat [26], se implementa un sistema que analiza el texto de un correo electrónico para determinar si contiene elementos de phishing, utilizando un modelo de red Bayesiana. El modelo utiliza un conjunto de palabras que son comúnmente usadas en correos de spam o de phishing. A cada palabra se le asigna un peso, dependiendo si puede causar emociones como enojo, miedo o ansiedad en la víctima. El sistema también chequea si el correo tiene enlaces a sitios no seguros (es decir que utilizan el protocolo HTTP) o que ya fueron clasificados como maliciosos. Si el valor de riego final asignado al correo de acuerdo a todos los cálculos

realizados es superior a un umbral determinado, el correo es clasificado como malicioso. Caso contrario es devuelto a la casilla del usuario.

La aplicación desarrollada fue llamada “Anti Phishing Simulator” identifica correos de phishing y spam, permitiendo al usuario tanto re-clasificar un correo, como también la utilización de un módulo de control de URLs, de forma que puede examinar los enlaces en el correo sin poner en riesgo al usuario.

En 2018, Gyan Kamal y Monotosh Manna [31] presentaron un algoritmo basado en clasificadores bayesianos para identificar sitios de phishing utilizando solamente características de la URL, logrando una precisión del 97,08 %. Para optimizar la eficiencia del algoritmo utilizaron técnicas como Bagging, Boosting y Stacking.

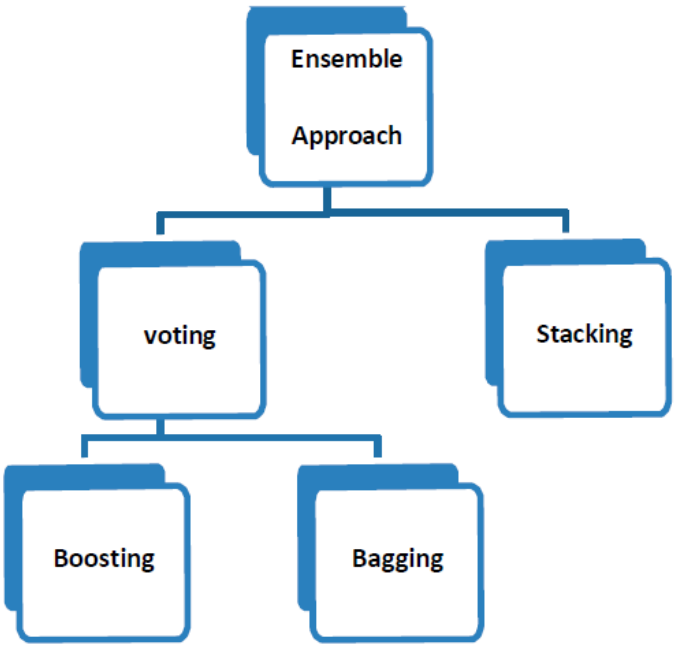

Figura 12: Enfoque combinado para optimizar el algoritmo [31]

Cuando usan Bagging, combinan varios modelos para lograr un único resultado. Cada modelo se entrena con subconjuntos del conjunto de entrenamiento. Estos subconjuntos se forman eligiendo muestras aleatoriamente (con repetición) del conjunto de entrenamiento. Para una instancia desconocida, se registran las predicciones de cada modelo y se asigna la clasificación que tiene el voto máximo entre las predicciones de los modelos.

En el Boosting, cada modelo intenta arreglar los errores de los modelos anteriores. El primer modelo tratará de aprender la relación entre los atributos

de entrada y el resultado. Seguramente cometerá algunos errores. Así que el segundo modelo intentará reducir estos errores. Esto se consigue dándole más peso a las muestras mal clasificadas y menos peso a las muestras bien clasificadas.

En el Stacking, las predicciones de cada modelo diferente se proporcionan como entrada para un clasificador de mayor nivel cuya salida es la clase final. La idea del Stacking es aprender de varios modelos diferentes y combinarlos entrenando un metamodelo para generar predicciones basadas en las predicciones múltiples devueltas por estos modelos más débiles. [34]

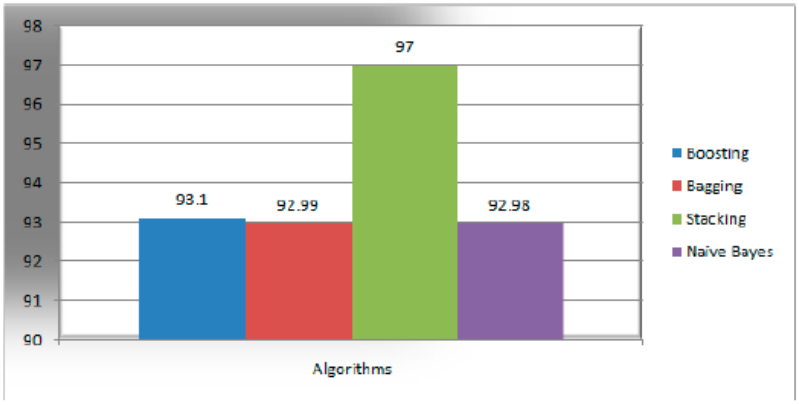

Figura 13: Precisión del algoritmo al combinar las distintas técnicas [31]

3.5.4. Sistemas basados en Procesamiento de Lenguaje Natural

A continuación veremos un enfoque un poco diferente a los anteriores ya que no utiliza un algoritmo de machine learning clásico para problemas de clasificación. Tianrui Peng, Ian G. Harris, y Yuki Sawa [28] proponen un modelo basado en el análisis semántico del texto transmitido por el atacante en un correo. Una oración es considerada maliciosa si solicitan información sensible u ordena la ejecución de una acción que podría exponer información personal. Técnicas de procesamiento de lenguaje natural son aplicadas a cada oración para identificar los roles semánticos de las palabras contenidas en la misma. Basadas en el rol que cumple cada palabra, el sistema determina si la oración es una orden o una pregunta. Luego la aplicación extrae el verbo y el objeto directo de todas las oraciones clasificadas de esta forma y cada par se compara contra una base de pares considerados maliciosos. Para generar esta base de pares verbo-objeto directo maliciosos, se utiliza un algoritmo de aprendizaje supervisado sobre un conjunto de prueba de correos legítimos y

de phishing.

El algoritmo implementado evalúa para cada oración del correo si se corresponde con alguno de los siguientes tipos: pregunta u orden considerada maliciosa, tono urgente, o saludo genérico. Finalmente, el correo es considerado como phishing si contiene enlaces identificados como maliciosos o si se encuentran al menos dos de los tres tipos de oración mencionados. Para determinar si una URL es maliciosa o no, se utiliza una herramienta proporcionada por terceros.

La identificación de preguntas y comandos maliciosos depende de la existencia de una lista negra de temas que es una lista de pares (objeto directo-verbo) cuya presencia en una pregunta o comando sugiere intención maliciosa. Para generar la lista negra usan un clasificador Bayesiano que se usa comúnmente para la clasificación de texto. Este algoritmo genera una etiqueta de predicción para cada par (objeto directo-verbo), y genera un puntaje de confianza para la predicción que va de 0 a 1, siendo el 1 el valor de mayor certeza.

La solución propuesta logra alcanzar una precisión del 95 %.

4. Conclusiones

En los últimos años la utilización de correos electrónicos y sitios web de phishing se ha vuelto un gran problema para la seguridad de la información. La gran cantidad de ataques de este tipo se debe principalmente a su efectividad, es muy difícil, aunque no imposible, lograr mitigar las vulnerabilidades "humanas" que intentan explotar los ataques de ingeniería social. Si bien es posible lograrlo a través de la concientización de la población en términos de seguridad de la información, y los riesgos existentes al hacer clic en un enlace o bajar un archivo enviado por un desconocido, cada vez es más evidente la necesidad de contar con métodos de detección automáticos con buena precisión y tasas de falsos positivos bajas.

Durante el desarrollo de este trabajo de investigación, se analizaron distintos métodos, tanto automáticos como manuales, para la detección de sitios web y correos electrónicos de phishing. En primer lugar, se presentó la situación actual como un problema de clasificación, como se realiza el análisis manual y los distintos algoritmos de machine learning que se podrían aplicar para clasificar los sitios y correos en fraudulentos o no. En el estudio y análisis de las diferentes técnicas se observó que los algoritmos de clasificación de machine learning presentan resultados con una alta tasa de precisión, en algunos casos superando el 98 % y bajas tasas de falsos positivos. Estos valores se obtienen eligiendo de forma efectiva el conjunto de características que se utilizarán en el modelo. De esta forma, la elección de las características del sitio o correo que se utilizarán no es trivial, y es clave para lograr una detección precisa.

En el cuadro 1 se observa un resumen de los algoritmos analizados. Algunos de las investigaciones incluidas en esta investigación no incluyen el grado de precisión obtenido, ya sea porque no era el objetivo del articulo o porque se omitió por alguna otra razón. Sin embargo se incluyeron el la comparación debido a que se consideró que sus ideas eran interesantes para la resolución de este problema.

A través de la comparación de los distintos estudios analizados y citados en las referencias, se puede concluir que los algoritmos que mejor resuelven este problema consisten en los métodos de bosques aleatorios. Siendo estos aquellos que mejor precisión obtuvieron. Un punto muy importante a tener en cuenta, es que la mayor eficiencia se puede obtener al combinar distintos

Algoritmo	Objetivo	Autores	Año	Precisión
Random Forest	Sitios	[23]	2019	98,8 %
Random Forest	Correos	[30]	2014	99,7 %
SVM	Sitios	[24]	2019	92,6 %
SVM	Sitios	[25]	2018	Desconocido
Bayes	Correo	[26]	2018	Desconocido
Bayes	Sitios	[31]	2019	97,08 %
Lenguaje natural	Correo	[28]	2018	95 %

Cuadro 1: Comparación resumida de los diferentes algoritmos

criterios y algoritmos, y lograr una respuesta en su conjunto. También se puede notar que muchos de los trabajos relevados se centran únicamente en detectar phishings en sitios web o en correos electrónicos, es decir, en uno o el otro. Sin embargo, muchos de los enlaces a sitios de phishing son enviados por correo electrónico, así como muchos de los correos de phishing contienen enlaces a sitios fraudulentos. Un enfoque combinado que analice ambos componentes del ataque podría otorgar grandes ventajas a la hora de detectar este tipo de fraude.

Por último, quisiera remarcar que el phishing se encuentra en constante evolución. Los atacantes cada vez cuentan con una mayor cantidad de recursos para llevar adelante los fraudes, y un mayor entendimiento de las medidas de detección para poder sortearlas. Por esta razón, es muy importante que las medidas de detección se encuentren también en constante evolución, adaptándose y, si es posible, anticipándose a las medidas implementadas por los atacantes.

5. Referencias

[1] S. Tayyab and A. Masood, “A review: Phishing detection using urls and hyperlinks information by machine learning approach,” *International Journal of Computer Science and Mobile Computing*, vol. 8, pp. 345–351, March 2019.

[2] N. Moradpoor, B. Clavie, and B. Buchanan, “Employing machine learning techniques for detection and classification of phishing emails,” *2017 Computing Conference*, pp. 149–156, July 2017.

[3] A. K. Jain and B. B. Gupta, “Phishing detection: Analysis of visual similarity based approaches,” *Security and Communication Networks*, January 2017.

[4] “RFC 822 - Standard for ARPA Internet Text Messages.”
2020-04-26 [Online], `https://tools.ietf.org/html/rfc822`.

[5] “RFC 7208 - Sender Policy Framework (SPF).”
2020-04-26 [Online], `https://tools.ietf.org/html/rfc7208`.

[6] “RFC 7001 - Message Header Field for Indicating Message Authentication Status.”
2020-04-26 [Online], `https://tools.ietf.org/html/rfc7001`.

[7] “¿Para que sirven los protocolos SPF, DKIM y DMARC?.”
2020-04-26 [Online], `https://help.sendinblue.com/hc/es/articles/209577385--Para-que-sirven-los-protocolos-SPF-DKIM-y-DMARC-`.

[8] J. Jupin, T. Sutikno, M. A. Ismail, M. Mohamad, S. Kasim, D. Stiawan, K. Bharu, and M. Kelantan, “Review of the machine learning methods in the classification of phishing attack,” *Bulletin of Electrical Engineering and Informatics*, vol. 8, pp. 1545–1555, December 2019.

[9] “Aprendizaje Inductivo: Árboles de Decisión.”
2020-04-26 [Online], `http://www.cs.us.es/~fsancho/?e=104`.

[10] “Introduction to K-means Clustering.”
2020-04-26 [Online], `https://blogs.oracle.com/datascience/introduction-to-k-means-clustering`.

[11] “Random Forest (Bosque Aleatorio): combinando árboles.”
2020-04-26 [Online], `https://iartificial.net/random-forest-bosque-aleatorio`.

[12] “Random Forest Regression.”
2020-04-26 [Online], https://towardsdatascience.com/random-forest-and-its-implementation-71824ced454f.

[13] “Máquinas de Vectores de Soporte (SVM).”
2020-04-26 [Online], https://iartificial.net/maquinas-de-vectores-de-soporte-svm.

[14] “The Kernel Trick in Support Vector Classification.”
2020-04-26 [Online], https://towardsdatascience.com/the-kernel-trick-c98cdbcaeb3f.

[15] “Cómo reconocer y evitar las estafas de phishing.”
2020-04-26 [Online], https://www.consumidor.ftc.gov/articulos/como-reconocer-y-evitar-las-estafas-de-phishing.

[16] “Phishing - Email Header Analysis.”
2020-04-26 [Online], https://mlhale.github.io/nebraska-gencyber-modules/phishing/email-headeranalysis/.

[17] “5 Ways to Identify a Phishing Website.”
2020-04-26 [Online], https://www.metacompliance.com/blog/5-ways-to-identify-a-phishing-website//.

[18] Y. Cao, W. Han, and Y. Le, “Anti-phishing based on automated individual white-list,” *DIM ’08: Proceedings of the 4th ACM workshop on Digital identity management*, pp. 51–60, October 2018.

[19] A. K. Jain and B. B. Gupta, “A novel approach to protect against phishing attacks at client side using autoupdated white-list,” *EURASIP Journal on Information Security*, December 2016.

[20] M. Khonji, Y. Iraqi, and A. Jones, “Phishing detection: A literature survey,” *IEEE Communications Surveys Tutorials*, vol. 15, no. 4, pp. 2091–2121, 2013.

[21] O. K. Sahingoz, E. Buber, O. Demir, and B. Diri, “Machine learning based phishing detection from urls,” *Expert Syst. Appl.*, vol. 117, pp. 345–357, 2019.

[22] A. R. Lekshmi and T. Seena, “Detecting malicious urls using machine learning techniques: A comparative literature review,” *International Research Journal of Engineering and Technology*, vol. 6, June 2019.

[23] A. Alswailem, B. Alabdullah, N. Alrumayh, and A. Alsedrani, “Detecting phishing websites using machine learning,” *2nd International Conference on Computer Applications and Information Security*, pp. 1–6, 2019.

[24] C. Wu, C. Kuo, and C. Yang, “A phishing detection system based on machine learning,” *International conference on Intelligent Computing and its Emerging Applications*, pp. 28–32, 2019.

[25] R. Kumar, S. Gunasekaran, R. Nivetha, K. SangeethaPrabha, G. Shanthini, and A. S. Vignesh, “Url phishing data analysis and detecting phishing attacks using machine learning in nlp,” *International Journal of Engineering Applied Sciences and Technology*, vol. 3, pp. 70–75, December 2018.

[26] M. Baykara and Z. Z. Gürel, “Detection of phishing attacks,” *6th International Symposium on Digital Forensic and Security*, March 2018.

[27] I. Tyagi, J. Shad, S. Sharma, S. Gaur, and G. Kaur, “A novel machine learning approach to detect phishing websites,” *5th International Conference on Signal Processing and Integrated Networks*, February 2018.

[28] T. Peng, I. G. Harris, and Y. Sawa, “Detecting phishing attacks using natural language processing and machine learning,” *12th IEEE International Conference on Semantic Computing*, 2018.

[29] S. Nandhini and V. Vasanthi, “Extraction of features and classification on phishing websites using web mining techniques,” *International Journal of Engineering Development and Research*, vol. 5, 2017.

[30] A. A. Akinyelu and A. O. Adewumi, “Classification of phishing email using random forest machine learning technique,” *Journal of Applied Mathematics*, vol. 2014, 2014.

[31] G. Kamal and M. Manna, “Detection of phishing websites using naïve bayes algorithms,” *International Journal of Recent Research and Review*, vol. 11, 2019.

[32] “Aprendiendo a identificar los 10 phishing más utilizados por ciberdelincuentes.”
2020-05-10 [Online], `https://www.osi.es/es/actualidad/blog/2014/04/11/aprendiendo-identificar-los-10-phishing-mas-utilizados-por-ciberdelincuen`.

[33] “Cibercriminales: ¿Qué es y en qué consiste el Phishing?.”
2020-05-10 [Online], https://www.liderempresarial.com/cibercriminales-que-es-y-en-que-consiste-el-phishing/.

[34] “Ensembles: voting, bagging, boosting, stacking.”
2020-05-10 [Online], https://iartificial.net/ensembles-voting-bagging-boosting-stacking/.