Revista Ciencia Unemi
ISSN: 2528-7737
ciencia_unemi@unemi.edu.ec
Universidad Estatal de Milagro
Ecuador

Benavides-Astudillo, Eduardo; Fuertes-Díaz, Walter; Sánchez-Gordon, Sandra
Un experimento para crear conciencia en las personas acerca de los ataques de Ingeniería Social
Revista Ciencia Unemi, vol. 13, núm. 32, 2020, -, pp. 27-40
Universidad Estatal de Milagro
Ecuador

Revista Ciencia UNEMI
Vol. 13, Nº 32, Enero-Abril 2020, pp. 27 - 40
ISSN 1390-4272 Impreso
ISSN 2528-7737 Electrónico

Un experimento para crear conciencia en las personas acerca de los ataques de Ingeniería Social

Eduardo, Benavides-Astudillo[1]*; Walter, Fuertes-Díaz[2];
Sandra, Sánchez-Gordon[3]

Resumen

La Ingeniería Social es la técnica que permite obtener información confidencial de los usuarios, de manera fraudulenta, con la finalidad de usarla en contra de ellos mismos, o de las organizaciones en las que laboran. Este estudio presenta un experimento enfocado a crear conciencia acerca de las consecuencias de este tipo de ataque, mediante la ejecución de un ataque controlado a personas de confianza. Para lograrlo, se han llevado a cabo un conjunto de engaños y actividades, que los atacantes usan comúnmente para obtener información sensible, incentivando la curiosidad de los contactos de las redes sociales para que visiten un blog personal con información ficticia. A más de esta interacción humana, se ha instalado un complemento oculto y no deseado, para recolectar información del usuario tales como: su dirección IP, país de origen, sistema operativo y tipo de navegador. Con la información recolectada, se realizó un ataque de escaneo a los puertos 80 (Web server) y 22 (SSH Server), para encontrar más información sensible. Posteriormente, se muestran los resultados a las víctimas. Además, luego del ataque se realizó una encuesta a los usuarios acerca de su conocimiento de Phishing y de Ingeniería Social. Los resultados muestran que únicamente el 2% de las personas, sospecharon o preguntaron acerca del verdadero motivo para visitar el Blog. Más aún, demuestra que las personas que visitaron el blog, no tienen conocimiento y conciencia de cómo se puede vulnerar información sensible de una forma relativamente sencilla.

Palabras clave: Ingeniería Social, Phishing, Ciber Ataque.

An Experiment to Create Awareness in People concerning Social Engineering Attacks

Abstract

Social Engineering is the technique of obtaining confidential information from users, in a fraudulent way, with the purpose of using it against themselves, or against the organizations where they work. This study presents an experiment focused on raising awareness about the consequences of this type of attack, by executing a controlled attack on trustworthy people. To accomplish this, we have carried out a set of activities or tricks that attackers use to obtain information, inspiring the curiosity of social network contacts to visit a personal blog with fictitious information. In addition to this human interaction, a hidden plug-in has been installed to collect user information such as his IP address, country, operative system, and browser type. With the information collected, a pentesting attack has been done to ports 80 and 22, in order to collect more information. Finally, the results were shown to the victims. In addition, after the attack, users were surveyed about their knowledge of Phishing or Social Engineering. The results demonstrate that only 2% of people suspected or asked about the real reason to visit the Blog. Furthermore, it reveals that the people, who visited the blog, don not have any knowledge and awareness of how to steal sensitive information in a relatively simple way.

Key word: Social Engineering, Phishing, Cyberattack.

Recibido: 30 de julio de 2019
Aceptado: 15 de noviembre de 2019

[1] Escuela Politécnica Nacional; Universidad de las Fuerzas Armadas-ESPE; Quito – Ecuador; diego.benavides@epn.edu.ec; https://orcid.org/0000-0003-4543-0082

[2] Escuela Politécnica Nacional; Universidad de las Fuerzas Armadas-ESPE; Quito – Ecuador; walter.fuertes@epn.edu.ec; https://orcid.org/0000-0001-9427-5766

[3] Escuela Politécnica Nacional; sandra.sanchez@epn.edu.ec; https://orcid.org/0000-0002-2940-7010

*Autor de correspondencia: diego.benavides@epn.edu.ec

I. INTRODUCCIÓN

La Ingeniería Social es el acto de violar la seguridad de la información de los seres humanos, por medio de todo tipo de engaños, para conseguir que se develе información privada. Su uso se ha incrementado en los últimos años gracias al crecimiento exponencial de los usuarios de las redes sociales, los correos electrónicos y demás formas de comunicación online.

La técnica cibernética de Ingeniería Social más utilizada es conocida como Phishing, por medio de la cual se logra obtener información confidencial de forma fraudulenta. El tipo de información sensible que se trata de obtener, son comúnmente: nombres de usuarios, contraseñas o incluso, información de tarjetas de crédito u otra información financiera de la víctima. El estafador conocido como phisher frecuentemente se hace pasar por una persona o empresa de confianza en una aparente comunicación oficial electrónica. Por lo general, se trata de contactar con sus víctimas, por medio de un correo electrónico, o de algún sistema de mensajería instantánea, redes sociales, SMS/MMS, o incluso utilizando llamadas telefónicas.

Frente a este escenario, este estudio realiza un experimento orientado a causar concienciación en el usuario. Para lograrlo, primero se realiza un ataque de Ingeniería Social controlado, a varios contactos de las redes sociales Facebook y WhatsApp. Para esto, se incentivó que los usuarios de Facebook y WhatsApp accedan a un Blog Personal ficticio con un complemento maligno oculto, para obtener información no consentida de los visitantes. Con esta recolección y análisis de datos, se desarrolló luego un ataque de pentesting de bajo impacto, mediante el cual se escanearon las direcciones IP recolectadas de forma no consentida, y se localizó aquellas que tenían los puertos 80 o 22 abierto, obteniéndose información adicional.

Finalmente, luego de informarles a los usuarios que habían sido víctimas de un ataque, se les realizó una encuesta para analizar el conocimiento de las personas, acerca de Ingeniería Social y Phishing. El objetivo principal de nuestro estudio es el de concientizar a los usuarios acerca de los peligros de Phishing e Ingeniería Social, y, además de demostrarles que mediante herramientas sencillas de red se puede obtener información valiosa de ellos. Otro aporte de este trabajo es el de conocer de primera mano, por medio de una encuesta, el conocimiento de ellos del problema, y su interés en conocer un poco más acerca de Phishing.

La organización del documento es como sigue: En la Sección Trabajos Relacionados, hacemos un completo análisis de la literatura, acerca de los métodos para combatir Phishing. El Materiales y Método, hacemos una descripción de la metodología utilizada para la realización de este experimento, y realizamos el experimento en sí. En resultados y discusión, realizamos un análisis de los resultados encontrados. En la siguiente sección, realizamos una encuesta a las personas que fueron nuestras víctimas y analizamos sus resultados. Finalmente, se muestran las conclusiones de nuestro trabajo.

Trabajos Relacionados

A. Técnicas tradicionales de detección de ataques de Phishing

Frente a este escenario, las opciones propuestas por la comunidad para mitigar los ataques de Phishing se enfocan principalmente a soluciones técnicas de ingeniería, es decir, en su minoría se orientan a educar al usuario acerca de las posibles causas y consecuencias de estos ataques.

De acuerdo a nuestra revisión sistemática de la literatura, las técnicas de mitigación más utilizadas para contrarrestar los ataques de Phishing son:

- Aquellas basadas en Machine Learning (Bahnsen, Bohorquez, Villegas, Vargas, & Gonzalez, 2017; Li, Geng, Yan, Chen, & Lee, 2016; Marchal, Francois, State, & Engel, 2014; Jianyi Zhang, Pan, Wang, & Liu, 2016).
- Aquellas basadas en el análisis de tráfico (Zou Futai, Gang Yuxiang, Pei Bei, Pan Li, & Li Linsen, 2016).
- Las basadas en complementos de navegadores (Marchal et al., 2017).
- Las basadas en arquitectura cognitiva (Williams & Li, 2017).
- Las basadas en la definición de listas negras (Jayshree Hajgude & Ragha, 2012; Hawanna, Kulkarni, & Rane, 2016) y por último
- Las basadas en el análisis léxico de las URL (Bahnsen et al., 2017; Feroz & Mengel, 2015).

Brevemente se describen estas soluciones a continuación:

La solución de (Bahnsen et al., 2017) se centra en el uso de técnicas de aprendizaje automático para la clasificación de sitios de Phishing analizando su URL, para ello, los autores comparan la combinación del análisis léxico con el estadístico, cuya salida actúa como entrada de un clasificador de una red de memoria de corto y de largo plazo Long Short Term Memory (LSTM). PhishStorm (Marchal et al., 2014), es un sistema automatizado de detección de Phishing, que también analiza en tiempo real cualquier URL para identificar si es un posible sitio de Phishing, además, puede interactuar con cualquier servidor de correo electrónico o proxy HTTP, para ello, los autores han definido un nuevo concepto de vinculación llamado intra-URL, y lo evalúan utilizando características extraídas de palabras que componen la URL, basadas en datos de consultas de motores de búsqueda de Google y Yahoo. Estas características se utilizan en la clasificación basada en el aprendizaje de máquina para detectar una URL falsa dentro de un conjunto de datos reales. Otra propuesta dentro del campo de Machine Learning, es la de implementar un sistema inteligente de detección de Phishing (IPD) (Li et al., 2016), de esta manera, primero se generan en forma automática, un conjunto de datos de detección usando los registros globales de nombres de dominio, entonces aplica el algoritmo Naive Bayes, el cual es optimizado por las características basadas en la posición de los caracteres, para lograr la detección con una alta precisión. Finalmente, para encontrar más sitios Web de Phishing, IPD expande el conjunto de datos de detección generando plantillas de URL basándose en los resultados de la detección previa.

Algunos aspectos nuevos de las características comunes que aparecen en las URL de Phishing se observan en (Jianyi Zhang et al., 2016), en que se introduce un clasificador estadístico de aprendizaje de máquina, para detectar los sitios de Phishing que dependen de estas características seleccionadas. Por medio de este análisis, se detecta el sitio Web de destino completamente basado en la propia URL y sin comprobar el contenido Web asociado.

La propuesta de (Zou Futai et al., 2016), se orienta al análisis de tráfico y al comportamiento de los sitios Web de Phishing, analizando los flujos IP directamente en los Proveedores de Servicios de Internet (ISP). Esto conllevó a los autores a proponer un método de detección basado en Graph Mining, utilizando una relación inherente entre la URL y su visitante, pudiendo detectar un posible ataque de Phishing que no pueda detectarse mediante el análisis de las URL.

En (Marchal et al., 2017), se presenta un enfoque moderno para detectar en tiempo real las páginas Web de Phishing a medida que son visitadas por un navegador. Esta propuesta se implementa como un complemento del navegador en el lado del cliente, preservando así la privacidad del usuario. Además, identifica el sitio Web de destino real que una página Web de Phishing intenta imitar y presenta al usuario una advertencia para proceder.

En (Williams & Li, 2017), se informan los resultados de una investigación sobre cómo los usuarios finales se comportan cuando se enfrentan a sitios Web de Phishing y cómo este comportamiento los expone a nuevos ataques. Los autores presentan un modelo de computadora como una prueba de concepto que simula el comportamiento humano con respecto a la detección de sitios Web de Phishing basada en una arquitectura cognitiva.

La propuesta del algoritmo de (Hawanna et al., 2016), hace una verificación en una lista negra de google que se va actualizando constantemente, y utiliza los resultados de este motor de búsqueda, además, muestra un mensaje de alerta de si la URL es una posible página de Phishing, de lo contrario, muestra un mensaje de sitio seguro. Este algoritmo mejora el rendimiento del equipo al analizar URLs de Phishing conocidas o ya analizadas.

En (J Hajgude & Ragha, 2012) los autores usan una técnica en la que se consideran las ventajas de las listas negras, listas blancas y la técnica heurística, para aumentar la precisión y reducir la tasa de falsos positivos. En la técnica heurística, usan el análisis textual y el análisis de URL de los correos electrónicos. Dado que la mayoría de los correos de Phishing tienen contenidos similares, este método aumenta su rendimiento, analizando el contenido textual de los mensajes sospechosos.

Existe una aproximación que clasifica las URL automáticamente, basándose en sus características léxicas (Feroz & Mengel, 2015) y basadas además en el host cliente. En complemento, se usan los servicios de reputación de URL en línea para así clasificarlas, y de acuerdo a esta calificación se las clasifica en categorías, como una fuente suplementaria de información, que permitiría al sistema clasificar las URLs en Phishing o no.

En la Tabla I, se ha clasificado a la literatura encontrada, de acuerdo a los métodos que se usan actualmente para mitigar el problema del Phishing. Además, se han tomado en cuenta solo los trabajos que hagan un análisis de las direcciones URL, sin embargo, estos métodos de detección han tenido sus limitaciones, provocando que los atacantes puedan mejorar y modificar sus estrategias, para así evitar su detección.

Tabla 1. Clasificación de las soluciones

Métodos	Documentos	Análisis de URL	Incentivan al usuario final
Machine Learning	(Bahnsen et al., 2017; Li et al., 2016; Marchal et al., 2014; Jianyi Zhang et al., 2016)	SI	NO
Análisis de trafico	(Zou Futai et al., 2016)	SI	NO
Complemento de Navegador	(Marchal et al., 2017)	SI	NO
Arquitectura Cognitiva	(Williams & Li, 2017)	SI	NO
Listas Negras	(J Hajgude & Ragha, 2012; Hawanna et al., 2016)	SI	NO
Análisis Léxico	(Bahnsen et al., 2017; Feroz & Mengel, 2015)	SI	NO

La técnica más utilizada en la detección de ataques es la basada en Blacklists, sin embargo, esta técnica sucumbe al detectar páginas o correos de phishing, que todavía no se han reportado como malignas. Para enfrentar esta deficiencia, se implementan técnicas de detección temprana con algoritmos de Machine Learning, como las revisadas en esta sección. Las técnicas de Machine Learning tradicionales mencionadas, necesitan a su vez, la supervisión de una persona o algoritmo, que seleccione las características principales de una página de Phishing, para que su exactitud en la detección sea mayor. Para mejorar la exactitud en la detección y prescindir de la selección de las características de una página de Phishing, en los últimos años se han implementado estudios en un área particular de Machine Learning, esta es Deep Learning. En el siguiente apartado, se hace una revisión de estas técnicas.

B. Últimas técnicas de detección de ataques de Phishing

Si bien el paper de (Basnet, Mukkamala, & Sung, 2008) no es relativamente nuevo (2008), este es interesante para entender el enfoque que los autores han tenido para evaluar las técnicas de Machine Learning tradicionales usadas hasta este momento, incluso, la estructura del artículo es completa y fácil de entender. Para este estudio, primero se escogieron 16 características relevantes, luego se recopilaron 4000 URLs entre URLs de Phishing y URLs benignas. De las 4000 URLs, se entrenó con el 50%, y con el otro 50% se realizaron las validaciones, finalmente, se evaluaron los Algoritmos: Support Vector Machines (SVM), Neural Networks, Self Organizing Maps y K-Means. De los métodos estudiados se observa que el más exacto fue SVM con un 97.99% de precisión.

El artículo (Yuan, 2017), utiliza un enfoque mixto, con las ventajas de velocidad de los métodos tradicionales de Machine Learning y la precisión de Deep Learning. Para esto, monitorea constantemente el comportamiento normal o anormal de los programas que se ejecutan en el Sistema Operativo. Si por alguna razón, y basándose en las reglas de Machine Learning, este software resulta sospechoso, entonces pasa a una fase de evaluación usando Deep Learning.

En (Woodbridge, Anderson, Ahuja, & Grant, 2018), se define primeramente un homoglifo como un nombre de suplantación de identidad, basado en la similitud de nombres de dominio reales, con nombres de dominio falsos, aprovechando el intercambio o borrado de caracteres dentro de una cadena legítima, por ejemplo, svchost.exe en lugar de svch0st.exe. Este paper propone una solución por medio del uso de Siamese Convolutional Neural Network (SCNN). Esta solución maneja una técnica utilizada en la renderización de las imágenes., de esta manera, una CNN aprende características que son utilizadas para detectar

similaridades visuales de las cadenas renderizadas. Metodológicamente, primero las cadenas de texto de nombre de dominio se transforman a imágenes y luego estas son pasadas por medio de la SCNN.

En el artículo (Saxe & Berlin, 2017), se trabaja sobre un método como en las cadenas de caracteres de las URLs, el cual se llama ofuscación. Para enfrentar este problema, se propone la utilización de una técnica de Deep Learning denominada eXpose neural network. Con esta técnica, se resalta que los datos ingresados son cadenas cortas de datos crudos, y posteriormente aprende a simultáneamente extraer características y clasificar esos datos, usando incrustaciones a nivel de carácter y usando convolutional neural network.

En (Shima et al., 2018), se usa una técnica llamada Bag of Bytes. Con esta técnica, los datos no entrarían totalmente crudos para ser analizados con los algoritmos de Deep Learning, sino que previamente los investigadores le asignan un valor hexadecimal a cada carácter de la cadena URL, luego, esos valores se emparejan con cada valor a la derecha. Posteriormente estos conjuntos de datos en hexadecimal son pasados por el algoritmo del modelo de red neural para su clasificación. Con este procedimiento, se mejora la exactitud de trabajos previos como eXpose (Saxe & Berlin, 2017).

El documento (Vazhayil, Vinayakumar, & Soman, 2018) hace un análisis comparativo entre los distintos métodos de inteligencia artificial para detectar URLs de Phishing. Primero evalúa los métodos tradicionales de Machine Learning, y luego los compara con los métodos de Deep Learning. En definitiva, se concluye que el método de Deep Learning LSTM es el más exacto, con un 98% de precisión.

En (Epishkina & Zapechnikov, 2016), los investigadores proponen mitigar los ataques pero por medio de la enseñanza de cómo prevenir estos, en el contenido del estudio de los universitarios durante su carrera. Su propuesta es más bien de manera práctica, aplicando el concepto de analíticas de Big Data, para mitigar los ataques de Ingeniería Social.

En el documento (X. Zhang, Zeng, Jin, Yan, & Geng, 2018), se hace uso de dos cosas para evaluar si una página es de Phishing o no: 1) Un análisis sintáctico del texto y 2) Un análisis estadístico del sitio web. Usa la premisa de que un conjunto de caracteres de una página de Phishing es similar a una página legítima. Además de la valoración anterior, se hace una evaluación de las estadísticas de las características de las páginas de Phishing, es decir, si es una página nueva, o por ejemplo si tiene mala reputación, etc. Finalmente, para la ejecución de la propuesta y medición, utiliza los algoritmos AdaBoost, Bagging, Random Forest y Sequential minimal optimization (SMO).

En (Vanhoenshoven, Napoles, Falcon, Vanhoof, & Koppen, 2016), clasificadores como Support Vector Machines (SVM), Random Forest, Naive Bayes, Decision Trees, K-Nearest Neighbors y Multi-Layer Perceptron fueron examinados y comparados con la clasificación binaria de URLs maliciosas. Para evaluar un conjunto de datos públicos de alrededor de 2.4 millones de URLs, este sistema recomienda la utilización de características numéricas para entrenamiento, y para obtener una mejor tasa de precisión en la predicción.

El usar URLs confusas, es decir las URL que se parecen mucho a alguna URL legítima, es una técnica muy común para engañar a los usuarios. Para lograr este tipo de confusión se utilizan varios trucos, los cuales pueden ser detectados mediante la extracción de las características comunes de estas páginas falsas. En este trabajo (Chen, Zhang, & Su, 2018), se recolectan 12 características de las páginas de Phishing, luego, se entrena el algoritmo LSTM Recurrent Neural Network con estas características, lo que finalmente produce un 99.14% de efectividad en la detección de nuevos sitios falsos.

En el paper (Jiahua Zhang & Li, 2017), se propone resolver el problema de datos desbalanceados, por medio de la implementación de un algoritmo Borderline-Smote (Synthetic Minority Oversampling Technique). Para esto, primero son extraídos tres grupos de características de las páginas de Phishing: Características de URL, características de páginas web y características de las imágenes. Con estos datos se entrena el algoritmo y posteriormente se obtiene una efectividad sobre 99% en la detección de nuevos sitios afectados.

Un novedoso y diferente método de detección es el que se hace sobre el flujo real IP (Internet Protocol), desde el proveedor ISP. Así, el paper (Yi et al., 2018), se centra principalmente en la aplicación de un framework para detectar páginas de Phishing. Primeramente, se clasifican dos tipos de características de los sitios falsos, las características originales y las características de interacción. En este trabajo se propone un modelo basado en DBN.

En el paper (Aksu, Turgut, Üstebay, & Aydin, 2019), se hace un estudio comparativo para la detección de sitios de Phishing, entre varias técnicas tradicionales de Machine Learning y la técnica Stacked Autoencoder de Deep Learning. Como resultado, la técnica de DL alcanzó la mejor tasa en la detección con una exactitud del 80%, independientemente de la cantidad de datos ingresados para entrenamiento. Antes de ser analizadas, las URLs son traducidas al código ASCII. Además, hace una determinación del porcentaje de incidencia de cada característica en una página Phishing.

Un enfoque para detectar si una página es de falsa o no, por medio de un mecanismo de Deep Learning no supervisado, es el que se usa en (Zhao, Wang, Ma, & Cheng, 2019). En este trabajo, los investigadores usan gated recurrent neural networks (GRUs), la cual es una variante de RNN. Una ventaja de este trabajo, es que puede además identificar qué tipo de ataque sobre URLs se está tratando de ejecutar. Es decir, en puede clasificar esa URL en Legitima, SQL Injection, XSS Attack, Sensitive File Attack, Directory Traversal, u otro tipo de ataque.

En (Jiang et al., 2018), los autores proponen un sistema de detección de URLs maliciosas, usando un esquema de CNN basado en nivel de carácter. Luego de usar este esquema, se realiza una comparación con otros esquemas, sobre 1000 URLs, y se obtiene que: usando un esquema de selección de características (Feature Selection), se encontraron 282 fallos; usando CNN a nivel de palabras, 158 fallos, mientras que, usando CNN a nivel de caracteres, se detectaron 40 fallos.

En (Spaulding & Mohaisen, 2018), se propone una solución que combate tanto a Phishing como a ataques de DNS tal como DDoS. Así, se propone un sistema llamado D-FENS el cual identifica nombres de dominio maliciosos en tiempo real. Este sistema corre dentro de un servidor DNS. En lugar de identificar características para ser aprovechadas en un sistema de Machine Learning tradicional, se opta por un enfoque de Deep Learning que aprende las características automáticamente de los datos de entrada.

Todos los métodos seleccionados en esta sección, dedicada sobre todo a la solución mediante la aplicación de aprendizaje profundo son importantes, sin embargo, se pueden revisar estos y más métodos especializados en esta área, en el artículo realizado por (Benavides, Fuertes, Sanchez, & Sanchez, 2019).

C. Métodos de detección no tradicionales

Un método poco común, es en el que se aplica la detección directamente sobre el tráfico en un DNS. En (Pereira, Coleman, Yu, DeCock, & Nascimento, 2018), se combate principalmente las direcciones URLs fraudulentas, generadas por medio de Domain Generation Algorithms (DGAs), para esto, se propone la utilización de una herramienta denominada WordGraph, con la que se pueden generar diccionarios similares a los utilizados por los DGAs.

En los métodos propuestos en el paper (Rao & Pais, 2018), no se evalúa ninguno de los métodos de Machine Learning tradicionales, pero si se realiza un estudio comparativo exhaustivo de la exactitud de estos métodos. Sin embargo, incluimos este artículo, porque explica a detalle las diferentes características que pueden ser extraídas de un sitio engañoso.

Por medio de DeepSeq (Sur, 2018), se trata de obtener el DNA o el perfil característicos de las personas que comúnmente son propensos a ser Phishers. Para esto, en base a los logs obtenidos, se compara los datos intrínsecos de las personas (Edad, Sexo, Ocupación, etc), versus los datos de los sitios que se visitan (negocios, arte, social media, etc). Finalmente, luego de realizar un análisis por medio de DNN, se obtiene un perfil tipo DNA.

En el trabajo propuesto en (Vrbančič, Fister, & Podgorelec, 2018), los investigadores proponen el método TDLBA o TDLHBA (Tuning Deep Learning using Bat/Hybrid Bat Algorithm). Este método combina los enfoques de inteligencia de enjambre para la configuración de los parámetros de las redes de Deep Learning. La principal ventaja del método según los autores, es la facilidad del uso con varias topologías Feed Forward Neural Netwoks y diferentes conjuntos de datos.

Un último método poco tradicional estudiado, para detectar ataques de Phishing, es el utilizado en (Rodríguez, Benavides, Torres, Flores, & Fuertes, 2018), en el cual los autores proponen la utilización de un modelo de confianza, a través del robo de la información almacenada en las cookies de los navegadores web. Esta información es recolectada y posteriormente enseñada a los mismos usuarios, para mostrarles lo vulnerables que pueden ser al permitir que se almacene información sensible de ellos en las cookies.

II. MATERIALES Y MÉTODOS

La metodología utilizada en el desarrollo de este estudio fue cuantitativa-descriptiva y por supuesto experimental, pues para obtener los datos, se contabilizaron las acciones que realizarón los usuarios en cada vector de ataque, además, se concluyó con la tabulación de una encuesta realizada a los usuarios acerca de la utilización de este ataque controlado.

Primeramente, se programó un complemento en Kali Linux, por medio del cual se puede capturar información de las personas, el cual se lo colocó en un al Blog Personal. Para incentivar que los usuarios visiten a este blog, se los enganchó indirectamente por Facebook y por WhatsApp. Una vez que el usuario visita el blog, ya es víctima del ataque, porque se obtiene información de su equipo, sin ningún consentimiento. Luego, con la información obtenida hasta ese momento, se realiza una prueba de penetración, obteniéndose de esta manera mayor información delicada. Una vez que el experimento termina, se les informa a los usuarios, que han sido víctimas de un ataque de Phishing. Finalmente, se les pide a los usuarios, que llenen una breve encuesta acerca de Phishing e Ingeniería Social.

Los resultados obtenidos en el experimento y en la encuesta son tabulados y mostrados en una sección posterior.

Ejecución del Experimento

Figura 1. Contador de Visitas al blog

Figura 2. Código QR mostrado en el estado de WhatsApp

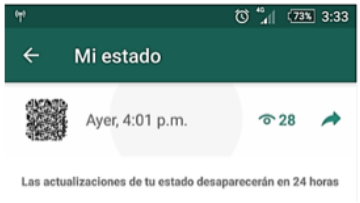

Figura 3. Número de usuarios que vieron el estado de WhatsApp

A. Recolección de datos

En esta primera sección se recolectó información a través de un ataque de Ingeniería Social, para esto, se planteó un escenario real, contando con la confianza de contactos conocidos en redes sociales. Para evitar atentar contra la privacidad de los usuarios, no se ejecutaron todas las fases del ciclo de un ataque de Phishing, únicamente se cumplió con la primera fase, que es la de recolección de información, posteriormente, se le notificó a cada usuario de que habían sido víctimas de un ataque de Phishing

Para cumplir con este primer objetivo se usaron 3 vectores para la recolección de datos:

1. *Implementación de un contador en un Blog Personal*. En la red social Facebook, se publicó un link a un Blog Personal. En este blog, se implementó un contador de visitas (ver Figura 1), sin embargo, el contador de visitas es en realidad un complemento engañoso, que sin que lo detecte el usuario, en realidad recolectó las siguientes características de los visitantes: País de origen, tipo de dispositivo usado, sistema operativo utilizado, tipo de navegador usado, dirección IP de origen, y fecha y hora de la visita al blog.
2. *Implementación de un código QR en un estado de WhatsApp*. Para esto, se configuró un QR Code usando la distribución Kali Linux, y se lo publicó como un estado de WhatsApp, para estimular la curiosidad de los contactos conocidos. Tal como se observa en la Figura 2, este QR Code tiene como leyenda "¿Eres curioso?, Demuéstralo.". Una vez que el usuario escaneaba el código, se lo direccionaba al blog personal, que incluía el contador de visitas "maligno" del paso anterior. En la Figura 3. se aprecia el número de contactos que revisaron el estado publicado.
3. *Elaboración de una encuesta*. Esta encuesta se realizó para analizar cuantas personas

conocen el significado de Ingeniería Social y de Phishing. El link para esta encuesta se lo ubicó dentro del blog personal implementado en el paso 1. Véase Figura 4.

Figura 4. Entrada del blog personal que solicita llenar una encuesta

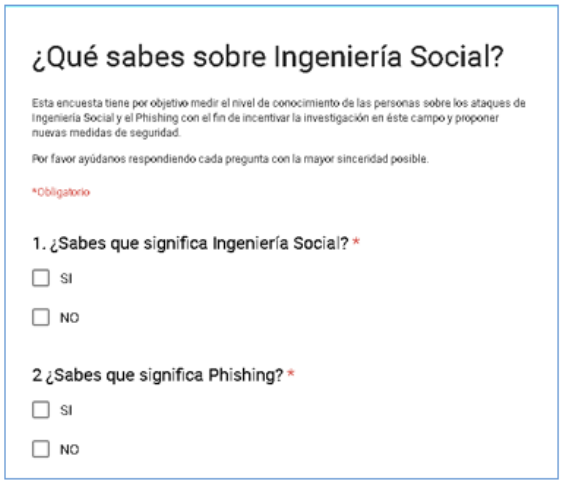

Figura 5. Encuesta elaborada con la herramienta Google Forms

Para realizar esta encuesta se usó la herramienta Google Forms, en la cual se plantearon 5 preguntas, con el objetivo de recolectar información sobre el conocimiento de las personas acerca de los ataques de Ingeniería Social. Véase la Figura 5.

B. Pentesting de las IPs recolectadas

En la segunda parte del estudio se analizaron los datos recolectados fraudulentamente, con el contador de visitas camuflado en el blog personal. Para recolectar información adicional, se utilizaron las direcciones IP capturadas, de los equipos de los usuarios que visitaron el blog.

(1) *Selección de la información del País de Origen y de las direcciones IP públicas*: Mediante el complemento camuflado que indicaba el número de visitantes, se registraron en una base de datos las variables que se muestran en la Figura 6.

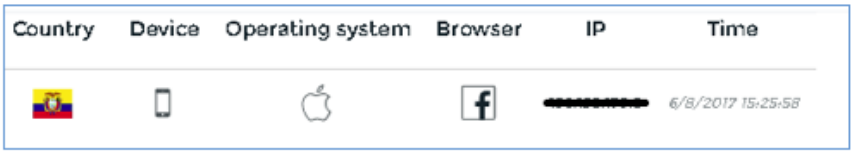

Figura 6. Variables registradas

(2) Filtro por país: Se seleccionaron solo los registros que pertenecen al país Ecuador, con la finalidad de evitar cualquier violación a la privacidad en otros países.

(3) Reconocimiento: Se utilizó software libre de test de penetración para ejecutar un análisis de escaneo de puertos a todas las direcciones IP registradas en la base de datos. Además, solo se seleccionaron los resultados de aquellos registros que tengan los puertos 22 y 80 abierto.

(4) Prueba de acceso: Sobre los equipos cuyos registros se encontraron con los puertos 22 y 80 abiertos, inmediatamente se comprobó mediante cualquier navegador Web (puerto 80), o consola (puerto 22), si se puede visualizar alguna información adicional relacionada a esta IP.

(5) Registro de la información obtenida: Con la información obtenida en

los pasos anteriores, se creó un registro de toda la información obtenida, con la finalidad de mostrar esta información a los usuarios posteriormente.

(6) Publicación de los resultados: Finalmente, se le comunicó a cada uno de los usuarios la información obtenida de ellos, para así concientizarlos de que obtener este tipo de información de ellos, no es un procedimiento complicado y de que en general somos muy vulnerables.

En esta sección se plantea una solución para incentivar a los usuarios a que se interesen en el conocimiento de la Ingeniería Social. A continuación, se realiza una breve descripción:

III. RESULTADOS Y DISCUSIÓN

Como se mencionó anteriormente, la idea de este ejercicio fue hacer que los usuarios ingresen a un Blog Personal, el mismo que estaba "contaminado" con un componente contador de visitas, por medio del cual se pudo obtener información no consentida, como la que se muestra en los gráficos de abajo. Así, 66 personas visitaron el Blog Personal implementado, y además de Ecuador, se registraron visitas de otros países como: Argentina, México, Brasil y Estados Unidos. Véase Figura 7.

El mayor número de dispositivos usados fue de 43 teléfonos celulares, frente a 23 usuarios que accedieron desde una PC. Véase Figura 8. Por lo tanto, el mayor número de usuarios accedió desde un dispositivo móvil con sistema operativo Android. Véase Figura 9. Esto ocurre porque la mayoría de los usuarios accedemos a internet desde nuestros propios dispositivos, en lugar de una computadora personal.

Figura 7. Estadísticas por países visitantes del Blog Personal implementado

Figura 8. Estadísticas por tipo de dispositivo usado

Figura 9. Estadísticas por sistemas operativos de origen

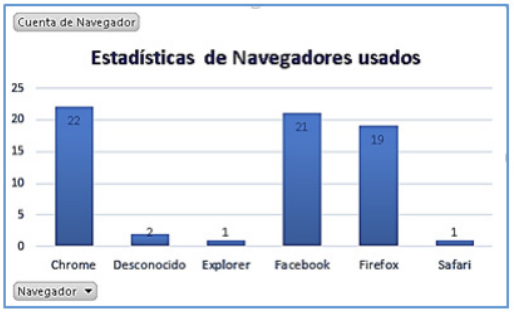

Figura 10. Estadísticas por navegadores usados por los usuarios

También se comprobó que la aplicación más usada fue el navegador Google Chrome, seguido de cerca por la aplicación Facebook y el navegador Firefox. Además, como dato curioso, se observó que también hubo usuarios que usaron navegadores desconocidos, tales como Tor, Proxy o incluso anonimizadores. Véase Figura 10. Adicionalmente, se registraron 28 contactos que observaron el estado de WhatsApp publicado con el código QR. Figura 2. En nuestro medio, la mayoría de teléfonos usan el sistema operativo Android, el cual es además muy amigable con Google Chrome, lo que nos lleva a concluir que por eso se usa en mayor medida Chrome.

A más de los datos obtenidos en este experimento, se realizaron pruebas de pentesting a todos los equipos que se mostraron vulnerables, obteniéndose acceso a los puertos 22 y 80, procedimiento por medio del cual se obtuvo aún más información. Esto demuestra que las personas comunes son muy vulnerables, y pueden fácilmente ser víctimas de un ataque de ingeniería social.

Además de los datos obtenidos de forma no consentida a los usuarios en esta sección, y después de haberles informado a ellos acerca de que habían sido víctimas de un ataque de Ingeniería Social, decidimos hacer una encuesta a los mismos usuarios, para conocer un poco más acerca de sus apreciaciones con respecto a Phishing e Ingeniería Social. Para esto, diseñamos una encuesta con 5 preguntas, las que se discuten en la sección siguiente.

Aplicación y resultados de una encuesta de Ingeniería Social

Para la encuesta, se registraron 57 personas. La confianza generada por la publicación en nuestra red social de Facebook para visitar una página informativa tuvo un buen resultado. Del total de personas que accedieron al enlace solamente una persona preguntó cuál era el motivo por el que se pedía ingresar. Al finalizar la encuesta, se informó a todas las personas que todo fue un ejercicio de Ingeniería Social. Estos resultados arrojaron que el 99% confiaron en nuestro enlace y solo el 1.7% preguntó de qué se trataba este enlace.

Como complemento, cada pregunta de la encuesta revela información importante, como por ejemplo que el 59.6% de personas tiene conocimiento sobre lo que significa Ingeniería Social, sin embargo, solo el 1.7% se cuestionó el por qué o el motivo de esta encuesta. Véase Figura 11.

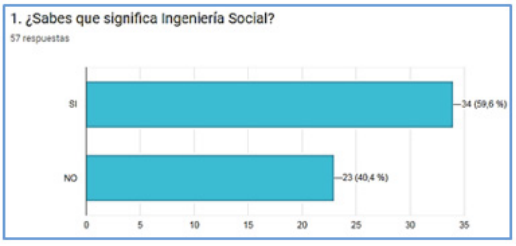

Figura 11. Pregunta 1 - ¿Sabes que significa Ingeniería Social?

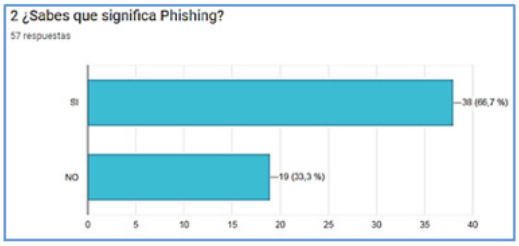

Figura 12. Pregunta 2 - ¿Sabes que significa Phishing?

Además, el 66,7% de personas tiene conocimiento de lo que significa Phishing (Figura 12.), y solo el 24,5% piensa que alguna vez ha recibido un ataque de Ingeniería Social (Figura 13.).

Un dato relevante es que el 77,4% (Vease Figura 14), pensó que sí se podría ejecutar un ataque mediante la publicación de un estado en WhatsApp. En otras palabras, el "enganche" planteado para atraer a los usuarios, tuvo un alto promedio de efectividad.

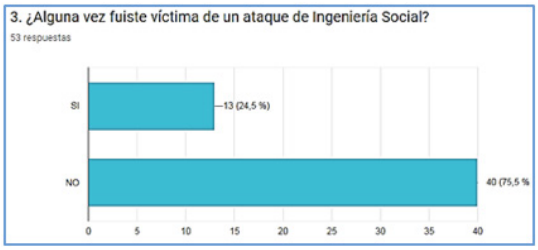

Figura 13. Pregunta 3 - ¿Alguna vez fuiste víctima de un ataque de ingeniería social?

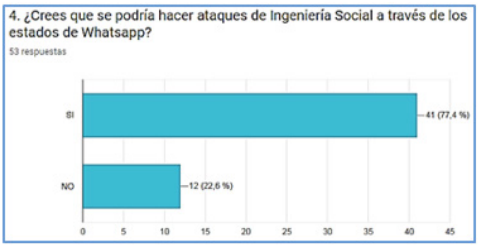

Figura 14. Pregunta 4 - ¿Crees que se podrían hacer ataques de Ingeniería Social a través de los estados de WhatsApp?

Con esta última pregunta (Figura 15.), se comprobó que muy aparte de que los contactos sean de confianza, el 9,1% indicó que se hubiera enojado si el ataque tenia consecuencias, como robar sus credenciales de alguna forma, pero no sería tan grave. Por otro lado, el 7,3% hubiese cortado toda relación de confianza.

Figura 15. Pregunta 5 - ¿Qué harías si un amigo de tu confianza te hace un ataque de ingeniería social y luego te explica y te enseña a defenderte?

Cuando se publicaron estos resultados, el 100% de los participantes se interesó en averiguar más sobre como se pudo obtener información de su navegación. Esto demostró dos cosas: primero, que el incentivo por redes sociales funciona a un buen nivel y segundo, que además genera curiosidad relacionada al aprendizaje de la Ingeniería Social.

IV. CONCLUSIONES

Este estudió presentó una propuesta para incentivar el aprendizaje relacionado a la Ingeniería Social, el cual se basó en un ataque controlado, ejecutado a través de un blog, al que se llegó mediante Facebook y WhatsApp. En este Blog se instaló un complemento oculto, para recolectar información sensible acerca de los visitantes.

El ataque fue un éxito, pues de todas las personas que abrieron el blog se puedo obtener información primaria, la que fue posteriormente utilizada como insumo para obtener más información privilegiada mediante pentesting. Los resultados muestran adicionalmente que la mayoría de víctimas, fueron atacados por medio del uso del teléfono celular.

De la encuesta realizada en nuestro estudio se puede destacar que, al día de hoy, el número de personas que conocen algo sobre Inteligencia Artificial y Phishing, es considerablemente importante, no obstante, se pudo notar que a pesar de saber que existe este tipo de ataque, los usuarios no saben cómo evadirlo. Esto permite inferir que esa es la razón principal por lo que la pregunta en relación a "Si desea aprender de métodos de control de estos ataques", es tan positiva.

Como trabajo futuro se planea utilizar diferentes algoritmos de Machine Learning y seguridad cognitiva para detectar, controlar y mitigar los ataques de Ingeniería Social. También se tiene planificado realizar un algoritmo que utilice Machine Learning para la detección temprana de los ataques de Phishing.

V. REFERENCIAS

Aksu, D., Turgut, Z., Üstebay, S., & Aydin, M. A. (2019). Phishing analysis of websites using classification techniques. In *Lecture Notes in Electrical Engineering* (Vol. 504, pp. 251–258). Springer, Singapore. https://doi.org/10.1007/978-

981-13-0408-8_21

Bahnsen, A. C., Bohorquez, E. C., Villegas, S., Vargas, J., & Gonzalez, F. A. (2017). Classifying phishing URLs using recurrent neural networks. In e*Crime Researchers Summit, eCrime* (pp. 1–8). IEEE. https://doi.org/10.1109/ECRIME.2017.7945048

Basnet, R., Mukkamala, S., & Sung, A. H. (2008). Detection of Phishing Attacks: A Machine Learning Approach. In *Soft Computing Applications in Industry* (pp. 373–383). Berlin, Heidelberg: Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-540-77465-5_19

Benavides, E., Fuertes, W., Sanchez, S., & Sanchez, M. (2019). Classification of Phishing Attack Solutions by Employing Deep Learning Techniques: A Systematic Literature Review. *SISTI,* 51–64. https://doi.org/10.1007/978-981-13-9155-2_5

Chen, W., Zhang, W., & Su, Y. (2018). Phishing detection research based on LSTM recurrent neural network. In *Communications in Computer and Information Science* (Vol. 901, pp. 638–645). Springer, Singapore. https://doi.org/10.1007/978-981-13-2203-7_52

Epishkina, A., & Zapechnikov, S. (2016). A syllabus on data mining and machine learning with applications to cybersecurity. In 2016 *Third International Conference on Digital Information Processing, Data Mining, and Wireless Communications (DIPDMWC)* (pp. 194–199). IEEE. https://doi.org/10.1109/DIPDMWC.2016.7529388

Feroz, M. N., & Mengel, S. (2015). Phishing URL Detection Using URL Ranking. In 2015 *IEEE International Congress on Big Data* (pp. 635–638). IEEE. https://doi.org/10.1109/BigDataCongress.2015.97

Hajgude, J, & Ragha, L. (2012). #x201C;Phish mail guard: Phishing mail detection technique by using textual and URL analysis #x201D; *In 2012 World Congress on Information and Communication Technologies* (pp. 297–302). https://doi.org/10.1109/WICT.2012.6409092

Hajgude, Jayshree, & Ragha, L. (2012). "Phish mail guard: Phishing mail detection technique by using textual and URL analysis." In *2012 World Congress on Information and Communication Technologies* (pp. 297–302). IEEE. https://doi.org/10.1109/WICT.2012.6409092

Hawanna, V. R., Kulkarni, V. Y., & Rane, R. A. (2016). A novel algorithm to detect phishing URLs. In *2016 International Conference on Automatic Control and Dynamic Optimization Techniques (ICACDOT)* (pp. 548–552). IEEE. https://doi.org/10.1109/ICACDOT.2016.7877645

Jiang, J., Chen, J., Choo, K.-K. R., Liu, C., Liu, K., Yu, M., & Wang, Y. (2018). A Deep Learning Based Online Malicious URL and DNS Detection Scheme (pp. 438–448). Springer, Cham. https://doi.org/10.1007/978-3-319-78813-5_22

Li, X., Geng, G., Yan, Z., Chen, Y., & Lee, X. (2016). Phishing detection based on newly registered domains. In *2016 IEEE International Conference on Big Data (Big Data)* (pp. 3685–3692). IEEE. https://doi.org/10.1109/BigData.2016.7841036

Marchal, S., Armano, G., Grondahl, T., Saari, K., Singh, N., & Asokan, N. (2017). Off-the-Hook: An Efficient and Usable Client-Side Phishing Prevention Application. *IEEE Transactions on Computers, 66*(10), 1717–1733. https://doi.org/10.1109/TC.2017.2703808

Marchal, S., Francois, J., State, R., & Engel, T. (2014). PhishStorm: Detecting Phishing With Streaming Analytics. *IEEE Transactions on Network and Service Management, 11*(4), 458–471. https://doi.org/10.1109/TNSM.2014.2377295

Pereira, M., Coleman, S., Yu, B., DeCock, M., & Nascimento, A. (2018). Dictionary Extraction and Detection of Algorithmically Generated Domain Names in Passive DNS Traffic (pp. 295–314).

Springer, Cham. https://doi.org/10.1007/978-3-030-00470-5_14

Rao, R. S., & Pais, A. R. (2018). Detection of phishing websites using an efficient feature-based machine learning framework. *Neural Computing and Applications*, 1–23. https://doi.org/10.1007/s00521-017-3305-0

Rodríguez, G. E., Benavides, D. E., Torres, J., Flores, P., & Fuertes, W. (2018). *Cookie scout: An analytic model for prevention of cross-site scripting (XSS)* using a cookie classifier. Advances in Intelligent Systems and Computing (Vol. 721). https://doi.org/10.1007/978-3-319-73450-7_47

Saxe, J., & Berlin, K. (2017). eXpose: A Character-Level Convolutional Neural Network with Embeddings For Detecting Malicious URLs, File Paths and Registry Keys. Retrieved from http://arxiv.org/abs/1702.08568

Shima, K., Miyamoto, D., Abe, H., Ishihara, T., Okada, K., Sekiya, Y., … Doi, Y. (2018). Classification of URL bitstreams using Bag of Bytes. Retrieved from http://member.wide.ad.jp/~shima/publications/20180219-ni2018-url-clf.pdf

Spaulding, J., & Mohaisen, A. (2018). Defending internet of things against malicious domain names using D-FENS. In *Proceedings - 2018 3rd ACM/IEEE Symposium on Edge Computing, SEC 2018* (pp. 387–392). IEEE. https://doi.org/10.1109/SEC.2018.00051

Sur, C. (2018). DeepSeq: learning browsing log data based personalized security vulnerabilities and counter intelligent measures. *Journal of Ambient Intelligence and Humanized Computing,* 1–30. https://doi.org/10.1007/s12652-018-1084-9

Vanhoenshoven, F., Napoles, G., Falcon, R., Vanhoof, K., & Koppen, M. (2016). Detecting malicious URLs using machine learning techniques. In *2016 IEEE Symposium Series on Computational Intelligence (SSCI)* (pp. 1–8). IEEE. https://doi.org/10.1109/SSCI.2016.7850079

Vazhayil, A., Vinayakumar, R., & Soman, K. (2018). Comparative Study of the Detection of Malicious URLs Using Shallow and Deep Networks. In *2018 9th International Conference on Computing, Communication and Networking Technologies, ICCCNT 2018* (pp. 1–6). IEEE. https://doi.org/10.1109/ICCCNT.2018.8494159

Vrbančič, G., Fister, I., & Podgorelec, V. (2018). Swarm Intelligence Approaches for Parameter Setting of Deep Learning Neural Network. In *Proceedings of the 8th International Conference on Web Intelligence, Mining and Semantics - WIMS '18* (pp. 1–8). New York, New York, USA: ACM Press. https://doi.org/10.1145/3227609.3227655

Williams, N., & Li, S. (2017). Simulating Human Detection of Phishing Websites: An Investigation into the Applicability of the ACT-R Cognitive Behaviour Architecture Model. In *2017 3rd IEEE International Conference on Cybernetics (CYBCONF)* (pp. 1–8). IEEE. https://doi.org/10.1109/CYBConf.2017.7985810

Woodbridge, J., Anderson, H. S., Ahuja, A., & Grant, D. (2018). Detecting homoglyph attacks with a siamese neural network. In *Proceedings - 2018 IEEE Symposium on Security and Privacy Workshops, SPW 2018* (pp. 22–28). https://doi.org/10.1109/SPW.2018.00012

Yi, P., Guan, Y., Zou, F., Yao, Y., Wang, W., & Zhu, T. (2018). Web Phishing Detection Using a Deep Learning Framework. *Wireless Communications and Mobile Computing, 2018,* 1–9. https://doi.org/10.1155/2018/4678746

Yuan, X. (2017). PhD Forum: Deep Learning-Based Real-Time Malware Detection with Multi-Stage Analysis. In *2017 IEEE International Conference on Smart Computing, SMARTCOMP 2017* (pp. 1–2). IEEE. https://doi.org/10.1109/SMARTCOMP.2017.7946997

Zhang, Jiahua, & Li, X. (2017). Phishing Detection Method Based on Borderline-Smote Deep Belief Network (pp. 45–53). Springer, Cham. https://doi.org/10.1007/978-3-319-72395-2_5

Zhang, Jianyi, Pan, Y., Wang, Z., & Liu, B. (2016). URL Based Gateway Side Phishing Detection Method. In *2016 IEEE Trustcom/ BigDataSE/ISPA* (pp. 268–275). IEEE. https://doi.org/10.1109/TrustCom.2016.0073

Zhang, X., Zeng, Y., Jin, X. B., Yan, Z. W., & Geng, G. G. (2018). Boosting the phishing detection performance by semantic analysis. In *Proceedings - 2017 IEEE International Conference on Big Data, Big Data 2017* (Vol. 2018-Janua, pp. 1063–1070). IEEE. https://doi.org/10.1109/BigData.2017.8258030

Zhao, J., Wang, N., Ma, Q., & Cheng, Z. (2019). Classifying Malicious URLs Using Gated Recurrent Neural Networks (pp. 385–394). Springer, Cham. https://doi.org/10.1007/978-3-319-93554-6_36

Zou Futai, Gang Yuxiang, Pei Bei, Pan Li, & Li Linsen. (2016). Web Phishing detection based on graph mining. In *2016 2nd IEEE International Conference on Computer and Communications (ICCC)* (pp. 1061–1066). IEEE. https://doi.org/10.1109/CompComm.2016.7924867