Twitter & Criptomonedas

// Nueva estrategia de estafa

OCELOT | METABASE Q

metabaseq.com

Twitter y Criptomonedas

// Introducción

Los ataques de ingeniería social son de los más difíciles de combatir, ya que su *modus operandi* no se basa en un algoritmo que pueda bloquearse con una actualización de *software*. Más bien, se aprovechan del desconocimiento y la confianza de las personas usuarias (principalmente las nuevas) para obtener contraseñas o información valiosa sobre sus cuentas. En este *blog*, nos centraremos en los ataques hacia los monederos virtuales de gestión de criptomonedas.

Siendo el Bitcoin uno de sus mayores aceleradores, las criptomonedas se han hecho cada vez más populares, hasta el punto en que algunos países han comenzado a utilizarlas como forma de pago válida en negocios locales. Esta creciente popularidad ha atraído a más y más usuarias y usuarios a participar activamente en el mercado de criptomonedas, invirtiendo tanto grandes como pequeñas cantidades, así como minando las cadenas de código necesarias para la formación de estas monedas. El creciente auge ha llamado la atención de grupos ciberestafadores que, conociendo las grandes cantidades de dinero que manejan estos sistemas, se han visto motivados a buscar una forma fácil y rápida de acceder a las cuentas usuarias que buscan soporte técnico de manera desesperada.

En la era de las redes sociales, Twitter se caracteriza por ser un medio a través del cual las personas expresan ideas, opiniones y quejas de forma resumida. Entre estas quejas, hay comentarios relacionados con los monederos de criptomonedas que son de especial interés para quienes realizan ciberestafas. Las cuentas malintencionadas pueden monitorizar el flujo de *tweets* que contengan palabras clave específicas. Esto lo hacen a través de herramientas de desarrollo proporcionadas por Twitter. Así, buscan *tweets* de personas que necesiten ayuda con sus criptomonedas y responden inmediatamente tratando de engañarlas a través de diferentes técnicas de ingeniería social para extraer la información de sus cuentas.

// Cómo opera la estafa

El funcionamiento es muy sencillo, atacantes realizan una rápida investigación sobre las palabras clave que una persona usuaria novata y desesperada tuitearía cuando necesita soporte técnico, estas palabras y frases podrían ser "necesito ayuda con", "necesito soporte con", "he perdido mis

criptomonedas", "*hackearon* mi", "me robaron mi", además del nombre de la respectiva criptomoneda o servicio de monedero de criptomonedas. Una vez identificadas estas palabras, un *bot* puede monitorizar los *tweets* que contengan esas frases o palabras y responder lo antes posible con un mensaje preestablecido.

Los mensajes maliciosos pueden variar, pero, en esencia, siempre buscan ganarse la confianza de usuarios e instarles a proporcionar información sensible para "resolver" su problema, lo que luego permite a los ciberdelincuentes acceder a sus cuentas y vaciar sus monederos de criptomonedas en cuestión de minutos.

Estos mensajes pueden clasificarse en los siguientes tres siguientes tipos:

1. Suplantación de identidad como soporte técnico

En este caso, atacantes crean una cuenta con un nombre, una foto de perfil y un *banner* idénticos a los de la cuenta oficial de soporte técnico, lo que hace más difícil de identificar si se trata de una cuenta de confianza. La estructura de los mensajes automáticos suele comenzar con una disculpa por las molestias y luego diciendo que, para solucionar el problema, la persona usuaria debe ponerse en contacto directamente para que le den soporte. Si esta etapa inicial tiene éxito y la persona usuaria decide contactar directamente con el supuesto soporte técnico, quien realiza el ataque intentará persuadir a la víctima para que revele la información de su cuenta con el argumento de que es necesario "confirmar su identidad" para recuperar el acceso. Una vez que revela su contraseña o frase de acceso, la billetera queda completamente desprotegida y se puede vaciar libremente su cuenta de forma rápida y sencilla.

Identificar una cuenta maliciosa es relativamente sencillo. Twitter ofrece dos mecanismos principales de defensa para evitar confusiones. El primer mecanismo es que cada persona usuaria debe tener un nombre de usuario único, por lo que, al consultar las cuentas de soporte, la persona debe comprobar siempre si la cuenta coincide con el servicio de soporte oficial que necesita. Como podemos observar en la Figura 1, es fácil identificar que todas esas cuentas son potencialmente maliciosas con solo mirar el nombre de usuario con atención.

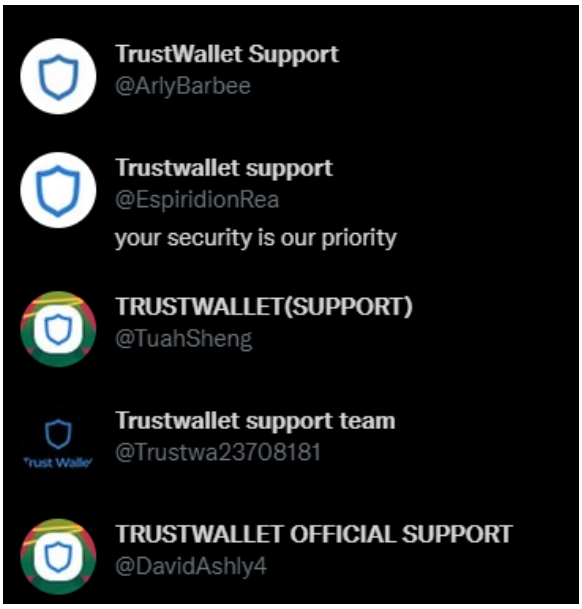

Figura 1: Sustitución de identidad para las cuentas de apoyo de *TrustWallet*

Sin embargo, hay cuentas que intentan ocultar esto utilizando caracteres similares o cambiando la posición de una letra para que una persona usuaria desprevenida pueda ser engañada más fácilmente como podemos ver en la Figura 2 y la Figura 3.

Figura 2: Cuentas con un nombre similar pero eliminando las letras

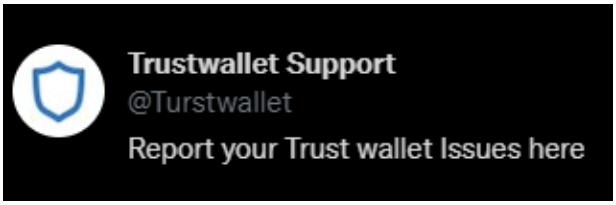

Figura 3: Cuentas con un nombre similar pero cambiando la posición dos letras

Por ello, Twitter cuenta con un sistema de símbolos de verificación (Figura 4) mediante el cual se asegura que la cuenta que tiene este sello distintivo asegura que es la cuenta oficial de la institución a la que dice pertenecer. Algunas cuentas maliciosas son conscientes de ello e intentan simular este sello de verificación con un carácter similar (Figura 5).

Figura 4: Cuenta oficial de *TrustWallet* con símbolo de verificación

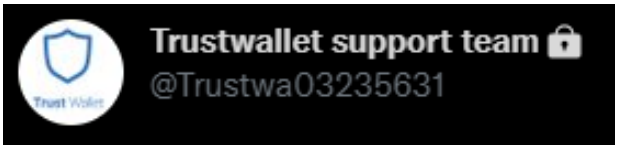

Figura 5: Cuenta fraudulenta con un símbolo de verificación anormal

Hay cientos de cuentas que utilizan este método de estafa para engañar a cuentas despervenidas, por lo que es importante que, si utilizas alguno de estos servicios, sepas identificar correctamente las cuentas oficiales de las redes sociales para evitar estos ataques.

2. Formulario de denuncia falsa

En este caso, atacantes se hacen pasar por una cuenta legítima del mismo monedero de criptomonedas que ha tenido el mismo problema, alegando que hay que rellenar un informe para que el equipo de soporte del monedero lo revise y luego resuelva el problema (Figura 8). Estos *tweets* suelen contener un enlace a un formulario de *Google Docs* (Figura 6) en el que se solicita información como el correo electrónico de la persona usuaria y la clave de recuperación, asegurando que sus datos estarán protegidos y que solo el equipo técnico podrá acceder a ellos (Figura 7).

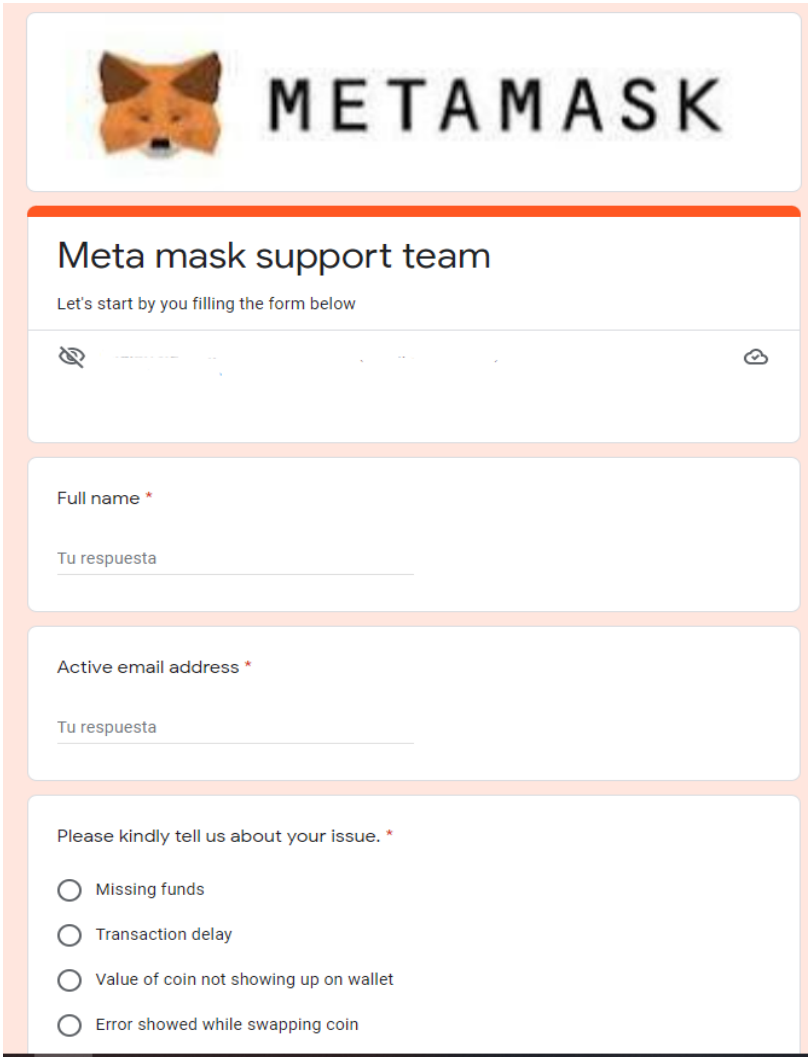

Figura 6: Formulario de informe falso de *Metamask*

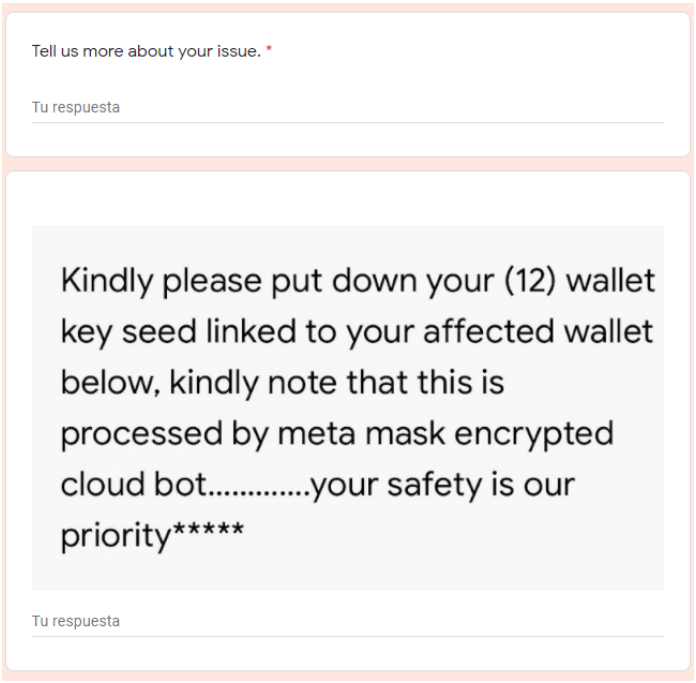

Figura 7: Estrategia para ganar la confianza de usuarios

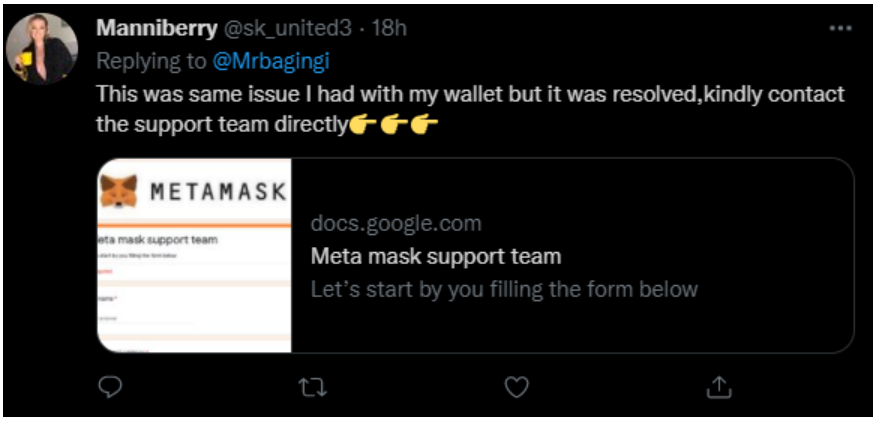

Figura 8: Ejemplo de un *tweet* estafador

Las cuentas oficiales de monederos recuerdan constantemente a sus usuarias y usuarios que sus servicios de asistencia técnica no solicitan contraseñas o claves de seguridad en otras páginas web o aplicaciones que no sean las oficiales.

3. Consejos maliciosos

Esta tercera variante utiliza cuentas (aparentemente) normales. Dejan un comentario automático bajo los *tweets* de las cuentas objetivo, explicando que pasaron por la misma situación, buscando así generar empatía y confianza. Posteriormente, sugieren que se pongan en contacto con otras cuentas de Twitter o Instagram que aseguran que pueden recuperar su cuenta comprometida y resolver sus problemas (Figura 9).

Al establecer contacto con esto tipo ciberdelincuentes, les pedirán la información mínima básica para acceder a las cuentas y, si lo consiguen, la cuenta quedará completamente expuesta y vulnerable. En esencia, este ataque es muy similar al primero, en el que atacantes se hacen pasar por especialistas con ganas de ayudar, extrayendo toda la información posible y utilizándola después a su favor.

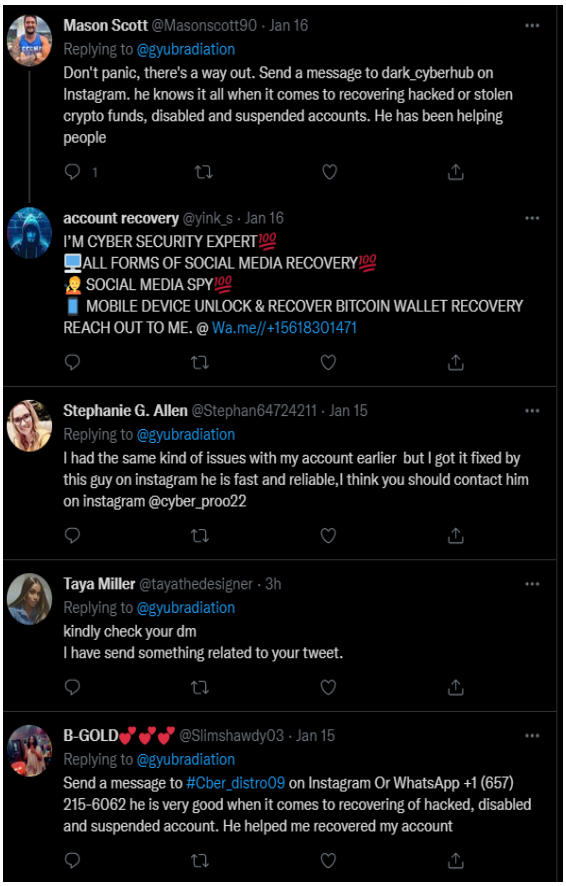

Figura 9: Ejemplos de *tweets* de estafadores

A pesar de que este tipo de ataques es muy sencillo y fácil de identificar es uno de los métodos más utilizados por ciberestafadores, es fundamental no compartir la contraseña personal o las claves de recuperación en canales inseguros.

// Medidas de seguridad contra estos ataques

Una de las mejores herramientas contra los ataques de ingeniería social es reforzar el sentido común en materia de ciberseguridad. Todos los ataques con mensajes automáticos suelen utilizar un lenguaje generalista con frases como "querido usuario" o "lo siento, amigo", por lo que es probable que la mayoría de las cuentas usuarias, tanto las nuevas como las experimentadas, se den cuenta inmediatamente de que algo no está bien en el supuesto consejo que están recibiendo. Sin embargo, cuando se trata de dinero y de la posibilidad de perderlo, solemos dejar que los nervios nos controlen y cometermos errores que pueden acarrear graves consecuencias, por lo que es necesario contar con herramientas que mitiguen la incidencia de los ataques.

Hay que tener en cuenta que el comportamiento de las cuentas fraudulentas está estrictamente prohibido por las normas de uso de Twitter. Por ello, si el algoritmo de Twitter detecta comentarios potencialmente peligrosos, los oculta, y es la persona usuaria quien decide si quiere, o no,ver esos comentarios (Figura 10).

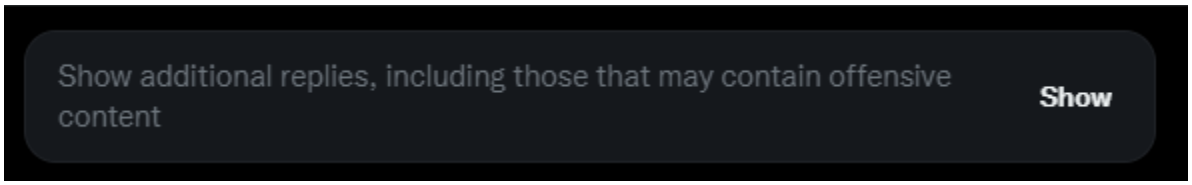

Figura 10: Mecanismo de autodefensa de Twitter

Hay quienes cuentan con más experiencia sobre estas amenazas y crean *tweets* llenos de palabras clave que provocan que estos *bots* comenten (Figura 11). Esto les permite detectar las cuentas fraudulentas y denunciarlas para que sean suspendidas. A pesar de las denuncias y de la constante suspensión de cuentas, este tipo de ataques se sigue produciendo. Aparte de esto, no hay muchas contramedidas para este tipo de ataques.

Figura 11: Ejemplo de una cuenta usuaria que intenta identificar a los *bots* estafadores

// Twitter API

Las aplicaciones que generan respuestas automáticas, también conocidas como *bots*, se generan a través de la Interfaz de Programación de Aplicaciones (API, por sus siglas en inglés) de Twitter, donde se pueden encontrar herramientas que permiten realizar búsquedas o dar respuestas preconfiguradas en cuestión de segundos.

La API de Twitter soporta varios lenguajes de programación y es una de las más permisivas en cuanto a las acciones que se pueden realizar. En este caso, por motivos de investigación, nos hemos centrado en el lenguaje de programación Python, concretamente en utilizar Tweepy[1], una librería que facilita considerablemente el uso de la API de Twitter. Ambas herramientas son gratuitas y fáciles de conseguir.

Twitter es consciente de que una aplicación creada con esta API puede ser problemática, por lo que para conseguir el acceso, es necesario crear una "cuenta de desarrollador", y solicitar el acceso rellenando un formulario. En este proceso, se pregunta qué funciones de Twitter van a utilizar en su aplicación y cuál es la intención principal de ésta. Esta solicitud es revisada manualmente por el equipo de Twitter y, en función de su criterio, se concede o no el permiso. La cuenta de desarrollador más básica está limitada a tener una aplicación a la vez, y un acceso limitado a la cantidad de información (500k *tweets* al mes en el caso más básico).

[1] Tweepy, Joshua Roesslein, 2022

En principio, parece un sistema eficaz para evitar que las aplicaciones sean generadas excesivamente por cualquier cuenta usuaria. Sin embargo, nada garantiza que la persona solicitante diga la verdad.

Una vez que tienen acceso a esta API, se les permite crear aplicaciones que proporcionan claves secretas que dan a la aplicación acceso a las herramientas de Twitter. Cuando se establece la conexión, las funciones de Tweepy permiten obtener la información pública de uno o varias cuentas, sus líneas de tiempo, sus seguidores, sus *tweets* favoritos, etc. Además, puede rastrear ciertas palabras clave en *streaming* y obtener estados en tiempo real. El estado tiene varios atributos de los que se puede obtener información relevante:

- Created_at: Proporcionar un valor de cadena con la hora UTC en que se creó este *tweet*
- Id: Representación del identificador único de este *tweet*
- Text: El texto UTF-8 real de la actualización de estado
- Source: Utilidad usada para publicar el *tweet,* como una cadena con formato HTML
- In_ reply_to_status_id: Si el *tweet* representado es una respuesta, este campo contendrá la representación entera del ID del *tweet* original
- In_reply_to_user_id: Si el *tweet* representado es una respuesta, este campo contendrá la representación entera del ID del autor del *tweet* original. Este no será necesariamente el usuario mencionado directamente en el *tweet*
- Coordinates: Representa la ubicación geográfica de este *tweet* tal y como la ha comunicado la persona usuaria o la aplicación cliente. La matriz de coordenadas interna está formateada como geoJSON (primero la longitud y luego la latitud)
- Place: Cuando está presente, indica que el *tweet* está asociado (pero no necesariamente originado) a un lugar
- Is_quote_status: Indica si se trata de un *tweet* citado
- Quoted_status_id: Este campo solo aparece cuando el *tweet* es un *tweet* citado. Este campo contiene el valor entero del ID del *tweet* citado
- Quoted_status: Este campo sóoo está presente cuando el *tweet* está citado. Este atributo contiene el valor entero *tweet* ID del *tweet* citado

- Retweeted_status: las cuentas pueden amplificar la difusión de los *tweets* escritos por otras cuentas retwitteando. Los *retweets* se distinguen de los *tweets* típicos por la existencia de un atributo retweeted_status. Este atributo contiene una representación del *tweet* original que fue retuiteado
- Retweet_count: Número de veces que este *tweet* ha sido retuiteado
- Favorite_count: Indica aproximadamente cuántas veces ha gustado este *tweet* a usuarios de Twitter
- Possibly_sensitive: Este campo solo aparece cuando un *tweet* contiene un enlace. El significado del campo no pertenece al contenido del *tweet* en sí, sino que es un indicador de que la URL contenida en el *tweet* puede contener contenido o medios identificados como contenido sensible
- Filter_level: Indica el valor máximo del parámetro filter_level que puede utilizarse y seguir transmitiendo este *tweet*. Así, un valor medio se transmitirá en los flujos ninguno, bajo y medio
- Lang: Cuando está presente, indica un identificador de idioma BCP 47 correspondiente al idioma detectado por la máquina del texto del *tweet*, o und si no se ha podido detectar ningún idioma. Ver más documentación aquí.
- User: La cuenta usuaria que publicó este *tweet*

Es posible que algunos de estos atributos no estén habilitados en la cuenta de Twitter y que no proporcionen la información solicitada. Otros están reservados exclusivamente para el uso de una aplicación empresarial. La cuenta objetivo también tiene sus propios atributos de los que se puede obtener aún más información, como:

- Id: La representación entera del identificador único para este usuario. Este número es mayor de 53 bits y algunos lenguajes de programación pueden tener dificultades/defectos silenciosos para interpretarlo
- Name: El nombre del usuario, tal y como lo han definido. No es necesariamente el nombre de una persona. Normalmente, tiene un límite de 50 caracteres, pero está sujeto a cambios
- Screen_name: El nombre de pantalla, el *handle* o el alias con el que se identifica este usuario. Los nombres de pantalla son únicos pero están sujetos a cambios. Suelen contener un máximo de 15 caracteres, pero pueden existir algunas cuentas históricas con nombres más largos

- Location: La ubicación definida por la persona usuaria para el perfil de esta cuenta. No es necesariamente una ubicación, ni es analizable por la máquina. En ocasiones, este campo será interpretado de forma imprecisa por el servicio de búsqueda
- Url: Una URL proporcionada por la persona usuaria en asociación con el perfil
- Description: La cadena UTF-8 definida por la persona que describe su cuenta
- Protected: Cuando este atributo es verdadero, indica que este usuario ha elegido proteger sus *tweets*
- Verified: Cuando este atributo es verdadero, indica que el usuario tiene una cuenta verificada
- Followers_count: El número de seguidores que tiene actualmente esta cuenta. En determinadas condiciones de coacción, este campo indicará temporalmente "0"
- Friends_count: El número de usuarios a los que sigue esta cuenta (también conocido como "seguidos"). En determinadas circunstancias de coacción, este campo indicará temporalmente "0"
- Favourites_count: El número de *tweets* que le han gustado a este usuario en la vida de la cuenta. La ortografía británica se utiliza en el nombre del campo por razones históricas
- Statuses_count: El número de *tweets* (incluyendo *retweets*) emitidos por el usuario
- Created_at: La fecha UTC en que se creó la cuenta de usuario en Twitter
- Withheld_in_countries: Cuando está presente, indica una lista de códigos de países de dos letras en mayúscula de los que se retiene este contenido

La cantidad de información que la API puede obtener con un solo *tweet* es abundante. Una vez identificada la estructura común del *tweet,* el usuario querrá seguir en la corriente. Es fácil dar una respuesta automática con las herramientas que ofrece Tweepy.

// *Modus Operandi*

Empezando por una de las estrategias más básicas, atacantes crean una nueva cuenta con un nombre de perfil y una foto falsos. Como parte de su estrategia para reducir las sospechas, incluso *tweetean* y añaden fotos en el *timeline;* también podrían empezar a seguir a gente, dando la sensación orgánica de que son perfiles auténticos (Figura 12). Una vez que han creado su fachada, atacantes solicitan su

acceso a la API mintiendo en los formularios de Twitter. Este permiso puede tardar un par de días dependiendo del motivo presentado, mientras tanto, el perfil simulado puede seguir construyéndose. Sin embargo, muchos otros simplemente dejan su perfil vacío.

Figura 12: Cuenta de *spam* creada recientemente

Con la API obtenida, la cuenta maliciosa ejecuta la aplicación de respuesta automática, la cual encuentra y responde automáticamente a los comentarios que piden ayuda. Es en este punto donde la estrategia se diversifica más. Algunas variantes pueden utilizar enlaces de falsos documentos de Google, sugerir el envío de un correo electrónico a direcciones falsas, buscar una cuenta externa de un *hacker* "experto y ético" que recupere la cuenta o solucione el problema, pedir el envío de un mensaje directo, etc. Identificar este tipo de mensajes como una amenaza no es complicado, ya que los *tweets* suelen ser muy sospechosos, y basta con entrar en el perfil de Twitter para identificar inmediatamente que es falso.

Sin embargo, existen variaciones que pueden aumentar las posibilidades de éxito de un ataque, por ejemplo, en un primer momento, se podría pensar que todas las cuentas maliciosas son de reciente creación debido a su naturaleza volátil. Pero, al investigar más a fondo, se encontraron cuentas maliciosas con más de diez años de antigüedad (Figura 13).

Figura 13: Cuenta falsa con antigüedad

Una posibilidad es que el atacante tuviera una cuenta antigua y simplemente la renovara para que funcionara como *bot*. Por otro lado, algunos usuarios han informado de que sus cuentas han sido tomadas (Figura 14), siendo utilizadas para perpetrar la estafa desde diferentes ángulos sin necesidad de crear cuentas o crear una fachada convincente (Figura 15).

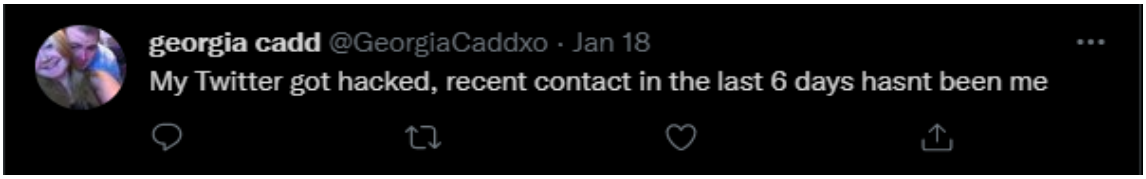

Figura 14: Informe de la cuenta *hackeada*

Figura 15: Cuenta *hackeada* utilizada para realizar *tweets* automáticos de estafa

Este tipo de cuentas son peligrosas porque son difíciles de identificar a primera vista dado que se ven como un perfil común. Si el mensaje es suficientemente convincente, puede eludir a los filtros de Twitter y evitar levantar cualquier tipo de sospecha, por lo que el perfil debería ser revisado cautelosamente para determinar si es peligroso.

Las cuentas que simulan ser sistemas de asistencia son también comunes en este estilo de estafas, robando la identidad de cuentas oficiales con un nombre falso, foto de perfil y fotografía de encabezado. La forma en la que operan es muy simple.

Seleccionan un servicio de monedero de criptomonedas, se dirigen a las redes sociales oficiales y copian cada posible detalle en su perfil falso. Una vez que este paso es completado, el mismo procedimiento señalado anteriormente se realiza, donde respuestas automáticas se publican en *tweets* de cuentas pidiendo ayuda pero enfocándose únicamente en los usuarios de la cuenta seleccionada. El mensaje automático usualmente simula a un integrante del equipo de asistencia y solicita enviar un mensaje directo a la cuenta para recibir instrucciones para resolver el problema (Figura 16).

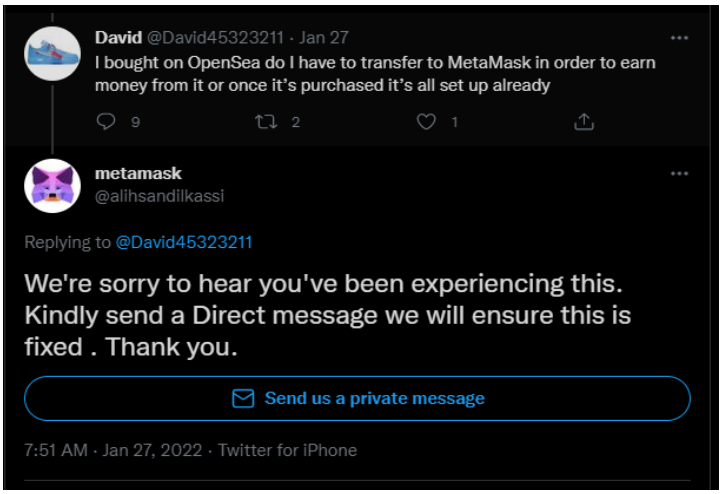

Figura 16: Cuenta fraudulenta de aistencia de Matamask

Si el engaño es exitoso, quien realiza el ataque usará herramientas de ingeniería social para obtener la información necesaria para obtener acceso a la cuenta del monedero de criptomonedas y transferir todo el dinero a sus propias cuentas. El nivel de dificultad en identificar estas cuentas puede variar significativamente dependiendo del nivel de esfuerzo que se ha puesto en duplicar la cuenta.

Otra estrategia utiliza los enlaces de páginas externas (Figura 17) que muestran un nombre como el servicio solicitado. Sin embargo, si se lee cuidadosamente, el dominio real de la página no es fidedigno. Hacer clic en un enlace no confiable es peligroso, ya que no solamente la información de la persona usuaria se ve comprometida, también puede compremeter a toda la computadora con la instalción de código malicioso o *malware*.

Figura 17: *Tweet f*raudulento con enlace falso de página de asistencia

Otras estrategias a las que debemos poner atención son aquellas con *bots* estafadores que diseñan su repuesta con ese conocimiento (Figura 18), para que el mensaje sea más confiable y sea más probable que funcione. Este tipo de mensajes suelen ser muy difíciles de detectar.

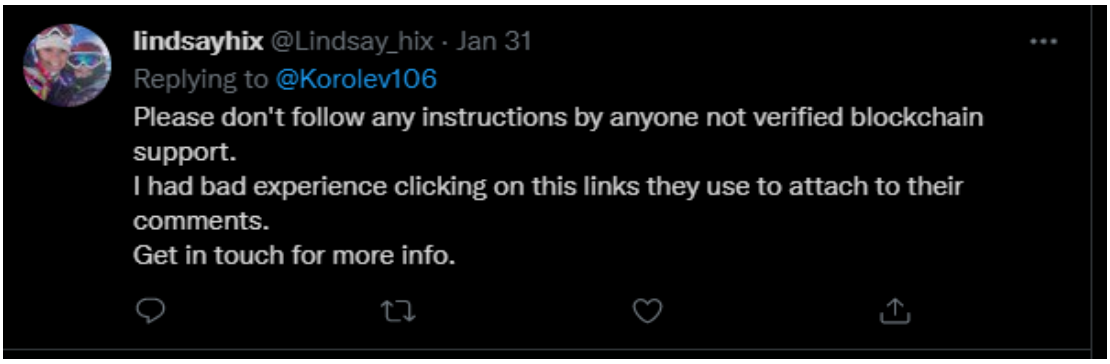

Figura 18: Bot aware scam tweet

Otros grupos o personas atacantes deciden monitorear directamente las respuestas publicadas por cuentas oficiales (Figura 19), buscando *tweets* que soliciten ayuda. Una vez que los *tweets* han sido identificados, responden a los mensajes de la víctima, para hacerlo parecer más creíble y confiable.

Figura 19: Scam response tweet in official account threads

Estas últimas tres estrategias representan las amenazas más peligrosas de todas las variantes que existen, ya que son las que requieren el mayor esfuerzo del lado de las personas atacantes. Además, son más difíciles de detectar dado que su comportamiento puede variar y eludir los filtros de Twitter. De esta forma, nuestra estrategia de defensa se concentrará en ellas.

Es importante resaltar que Twitter bloquea activamente decenas de cuentas de *bots* diariamente, pero para poder hacerlo, las personas usuarias deben reportar manualmente a estar cuentas.

// Buscando a usuarios maliciosos

Twitter API tiene diferentes usos. Actualmente, hay dos versiones, por lo que la cantidad o tipo de información que puede ser accesada depende en el tipo de cuenta disponible y en el método de autenticación usado (que determina la versión de API). En este caso, el equipo de Investigación y Desarrollo de Metabase Q (I&D) tuvo acceso a una cuenta de desarrollador elevada. Con esta cuenta, pudimos realizar hasta 300 solicitudes cada quince minutos; cada solicitud podía obtener un máximo de 100 *tweets*, alcanzando 30,000 *tweets,* y con un límite total de dos millones de *tweets* por mes.

Analizando el *modus operandi* de las estafas de Twitter, sabemos que la fecha de la creación de cuenta no es un parámetro confiable para determinar si la cuenta es real o no, ni podemos confiar en el número de *tweets*, tiempo de respuesta, seguidores, amistades, foto de perfil, ni siquiera el nombre, ya que todos estos factores son variables. Por esta razón, crear una aplicación que identifique todas las cuentas maliciosas en Twitter es complicado. En este caso, decidimos enfocarnos en los *tweets* que utilizan estrategias más agresivas. Esto incrementa la probabilidad que los *tweets* detectados y las cuentas sean realmente maliciosas.

En las secciones previas, mencionamos las propiedades de un *tweet* y un usuario que son accesibles a través del API v1 de Twitter. Si observamos de forma cuidadosa, no hay un parámetro que nos permita obtener las respuestas para los *tweets*, solamente puede mostrarnos si el *tweet* encontrado es una respuesta o no. No obstante, hace apenas un par de años, Twitter sacó una segunda versión del API que contiene más propiedades y nos permite tener acceso a más información. Con API v2, hay una propiedad llamada "campos," que conteine toda la información proporcionada por API v1 más otras nuevas propiedades como la de "ID de conversación".

Con la propiedad del ID de conversación, es posible obtener todos los *tweets* que han sido publicados como respuestas al *tweet* principal, e incluso las respuetas a las respuestas. Una vez que todos los *tweets* han sido obtenidos, pueden filtrarse con parámetros de búsqueda como palabras clave o enlaces, de ser aplicables. A través de los *tweets* filtrados, API proporciona acceso a toda la información pública de la cuenta, como su nombre de pantalla, número de seguidores, línea de tiempo, etc. Posteriormente, toda esta información puede ser almacenada.

La librería Tweepy ha ido actualizando sus funciones para cubrir las funcionalidades ofrecidas por API v2 de Twitter. Hasta este momento, las funciones relacionadas con los *tweets* de transmisión no existen, por lo que cualquier algoritmo construido con esta librería, usando API v2, puede únicamente accesar con el historial de *tweets*.

// Implementación de la búsqueda

Para abordar el problema, el equipo de Metabase Q desarrolló un algortimo (Figura 20), que nombramos "Spotter", capaz de buscar entre los *tweets* más recientes de varias cuentas de monederos de criptomonedas oficiales, filtrando los *tweets* sospechosos que tienen ciertas palabras clave.

Adicionalmente, Spotter integra una herramienta desarrollada por la compañía Banbreach[2], que es un algoritmo que crea nombres de usuario de Twitter similares y revisa si estos existen.

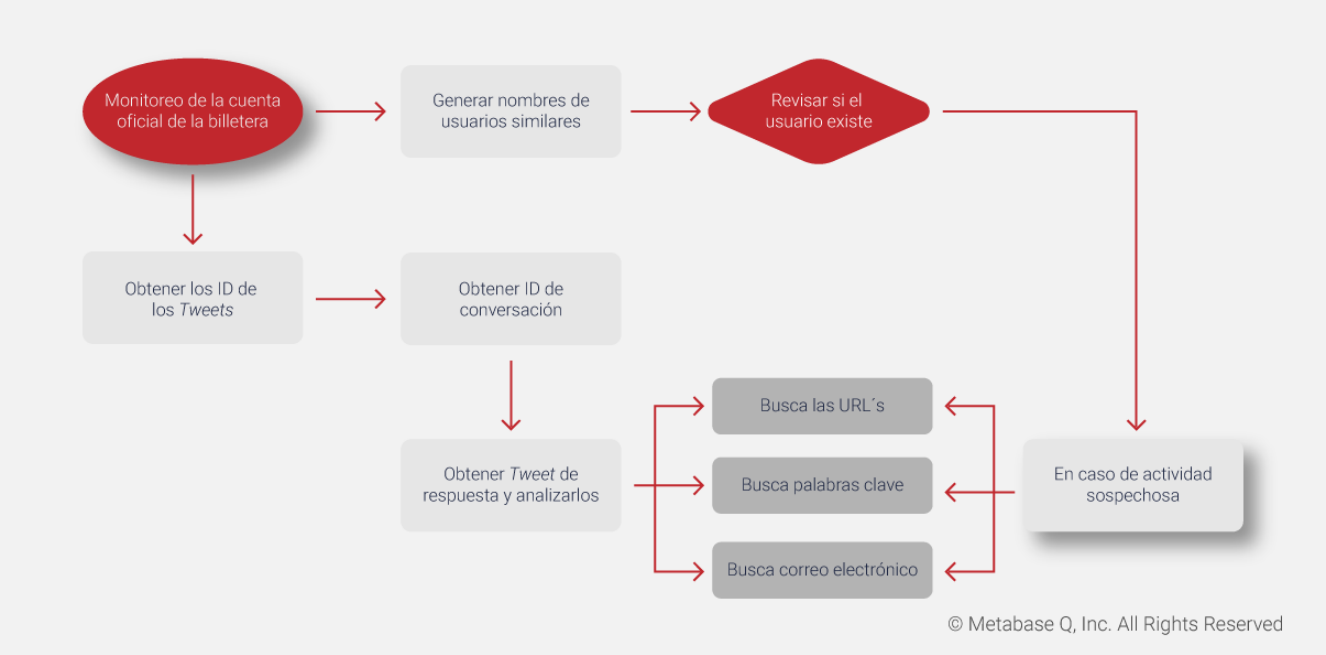

Figura 20: Diagrama de respuesta del escáner de Twitter

Estas dos estrategias nos permiten cubrir un rango amplio de cuentas potencialmente peligrosas usadas por quienes realizan estafas en Twitter. Todos los *tweets* y cuentas sospechosas son almacenadas en una base de datos con toda la información necesaria para determinar si son cuentas maliciosas o "falso-positivas." Los *tweets* que no pasan el filtro también son almacenados para verificar a los "falso-negativos," y de esa manera, mejorar la precisión de los filtros.

Cada quince minutos, Spotter tomará un nombre de la lista (por orden), usando Twitter API v2 y la librería Tweepy, los últimos diez *tweets* de la cuenta oficial son obtenidos, permitiéndonos buscar a todos los *tweets* que han respondido en el hilo, inlcuyendo las respuestas a las respuestas. En la primera búsqueda, todos los *tweets* que tienen este ID de conversación son almacenados en una tabla de una base de datos usando el lenguage de estructura (SQL, por sus siglas en inglés) con la extensión SQLite3 de Python. La data almacenada es:

[2] Tweeptwist, Banbreach, April 28, 2018.

- ID de conversación
- Nombre de usuario
- ID de usuario
- Fecha del *tweet*
- Texto del *tweet*
- ID del *tweet*
- ID del usuario al que se le respondió

En una segunda búsqueda de *tweets* con el mismo ID de conversación, solo los *tweets* con palabras clave son guardadas. Un problema que encontramos es que muchas de las palabras utilizadas por quienes realizan estafas son también usadas por cuentas reales solicitando ayuda, por lo que esto representa un serio problema a resolver con este simple filtro. Como una solución provisional, solamente registramos los *tweets* que no contestan a las cuentas oficiales, solo a aquellas de usuarios pidiendo ayuda. Esta estrategia reduce de forma significante el promedio falso-positivo. Spotter puede funcionar por un periodo largo buscando repetidamente cuentas de la lista, haciendo posible que muchos *tweets* repetidos se obtendrán de nuevo en la búsqueda. Por consiguiente, cada ID de *tweet* obtenido es buscado en la base de datos. Si son detectados en ella, son ignorados y la búsqueda continúa para nuevos *tweets*, evitando así la duplicidad.

La segunda parte de Spotter usa la herramienta de Tweeptwist para buscar nombres de usuario que tengan variaciones mínimas en su estructura. Utiliza Twitter API v1, cuya última versión fue publicada en 2018. Desde entonces, API ha tenido diversos cambios. Por otro lado, el código original fue configurado para operar como un comando y no como una función, por lo que la estructura de Tweeptwist tuvo que ser modificada para operar de forma correcta, para ser incluída en el código principal, y permitir a Spotter manejar la información obtenida.

Con esto fijo y en funcionamiento, Tweeptwist toma el nombre de la cuenta oficial de la lista y a través de diferentes técnicas, genera una lista de posibles nombres falsos que podrían ser utilizados como cuentas en Twitter. Después, utilizando API v1, encuentra si la cuenta existe, proporcionando únicamente usuarios existentes. Además, esta lista también obtiene información como el número de amistades, seguidores o *tweets* que la cuenta tenga.

De la lista de resultados, las cuentas que más interesan son los que están activas, por lo que se seleccionan aqueppas que han publicado *tweets* recientemente. Con el uso de la API v2, Spotter obtiene los últimos *tweets* de cada usuario (en un periodo de 7 días) y, si contienen alguna palabra clave, la cuenta se considera activa.

Todas las cuentas de la lista se guardan en otra tabla de la base de datos, sin importar si están activas o no. Como con los *tweets*, evitamos duplicar información al revisar el ID del usuario en la base de datos.

Finalmente, para prevenir que se excediera el límite de Twitter API, Spotter pausa por quince minutos y reinicia su proceso con el siguiente usuario en la lista. Cuando el último usuario en la lista es alcanzado y aún hay tiempo, la lista se reinicia.

// Prueba experimental

Para la validación de Spotter, se realizó una prueba de dos horas con una lista de once monederos oficiales de criptomonedas:

- Blockchain
- AskBlockchain
- Metamask
- MetaMaskSupport
- TrustWallet
- TrustWalletApp
- Binance
- BinanceHelpDesk
- BinanceX
- Coinbase
- CoinbaseSupport

En este orden específico, buscamos los últimos diez *tweets* y buscamos sus respuestas. Como se mencionó anteriormente, Spotter obtiene cada ID de la conversación y realiza dos búsquedas. La primera obtiene los últimas 100 respuestas de *tweets* (debido a los límites de API) sin filtro, y luego los almacena en una tabla de la base de datos. La segunda búsqueda obtiene las últimas 100 respuestas de *tweets* que contienen las siguientes palabras clave:

- Support
- Join
- Fixed
- Assist
- Reaching
- Helped
- Chat
- Instagram
- Sorry
- DM
- Direct
- Message
- Help
- Reach
- Write
- Contact

Como resultado de nuestra investigación, encontramos que estas palabras son comunmente usadas por personas que realizarán una estafa en sus mensajes. Todos los *tweets* obtenidos están guardados en otra tabla de la misma base de datos.

Por último, Spotter ejecuta Tweeptwist para buscar nombres de usuario similares, y después registra cada usuario existente (que no sea una cuenta oficial) e investiga sus últimos *tweets*. Esta metodología permanece igual, si la cuenta está activa (ha tuitteado en los últimos siete días) y los *tweets* contienen alguna palabra clave del filtro, se cataloga como "activa." Finalmente, todas estas cuentas se registran en otra tabla de la base de datos.

En este punto, Spotter tomaría un retraso de 900 segundos (quince minutos) y esperaría para repetir el mismo proceso con la siguiente cuenta oficial hasta que el tiempo se agote.

// Resultados

La última prueba fue realizada el 28 de febrero y en tan solo dos horas, llegó a alrededor de 10,246 *tweets*. Spotter encontró 6,652 *tweets* marcados como negativos (esto significa que no contenían una palabra clave) de los cuales, se obtuvo un promedio de 1.44% de falso-negativo. Adicionalmente, 3,594 *tweets* fueron marcados como positivos (esto significa que contenían al menos una palabra clave) con un promedio falso-positivo de 0.9% como podemos ver en la Figura 21. Estos porcentajes no consideran *tweets* no deseados. Desafortunadamente, muchos *tweets* no desados también tienen las mismas palabras clave que los *tweets* de estafa.

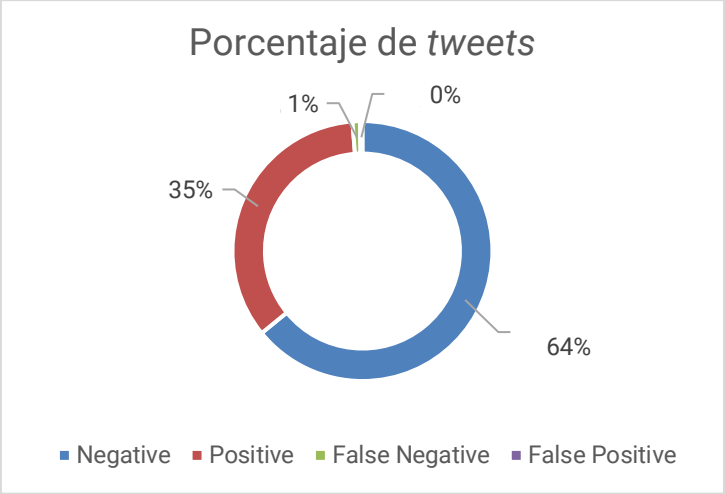

Figura 21: Número y porcentaje engañoso de *tweets* por cuenta durante la prueba

En la Figura 22, podemos observar la cantidad de *tweets* por cuenta registrados en la prueba, con esta información podemos observar qué tan frecuente es este tipo de ataque para algunos monederos de criptomonedas. Además, podemos conlcuir que hay más incidencia en las cuentas principales que en las cuentas de asistencia.

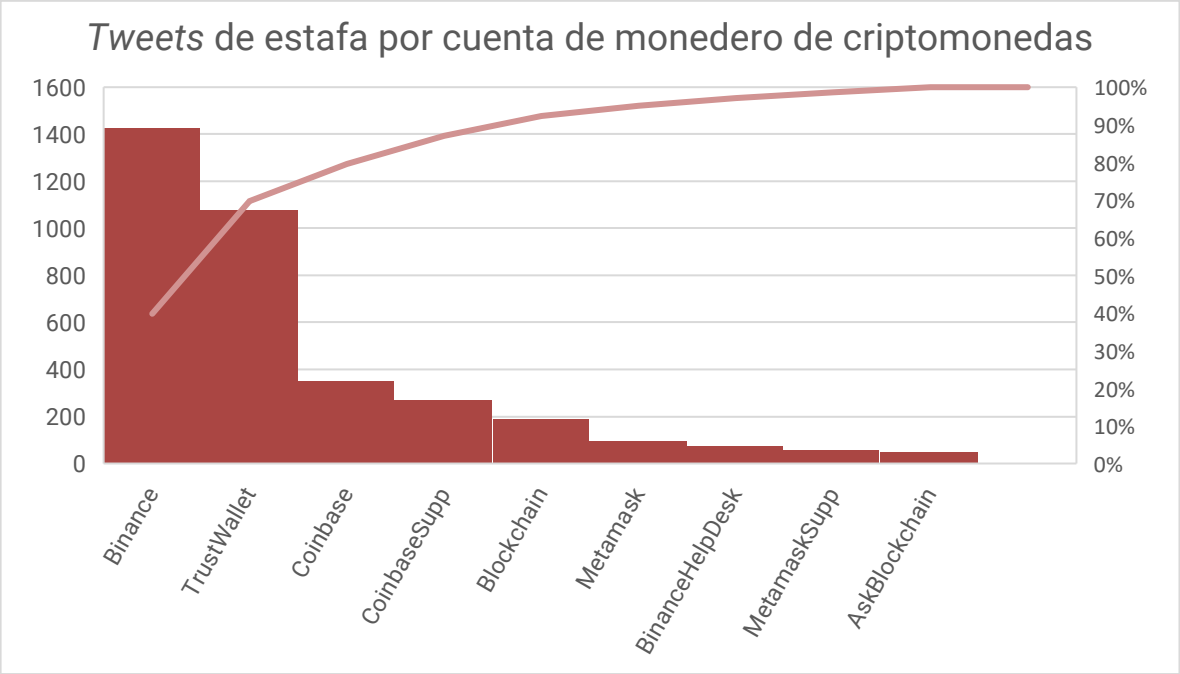

Figura 22: Número y porcentaje de resultados de *tweets* en la prueba de desempeño de Spotter

Para la búsqueda de cuentas de personas usuarias similares con Tweeptwist, de 125 cuentas, solamente cinco estaban activas y una resultó ser una falsa-positiva.

// Discusión

Spotter es confiable para encontrar muchos *tweets* que pretenden robar criptomonedas con un bajo nivel de errores y puede funcionar por largos periodos. Esta herramienta podría usarse para reducir el riesgo de caer en este tipo de estafas.

// Limitantes

Existen múltiples limitaciones en Twitter API que complican la implementación efectiva de un algoritmo de búsqueda. Una de las limitantes es el número de condiciones que pueden usarse para los filtros (máximo 20 condiciones para la cuenta de desarrollador elevada) por lo que es necesario seleccionar de forma cuidadosa las palabras claves (*keywords*) usadas.

Otro problema encontrado es que el filtro registra únicamente las respuestas de *tweets*, por lo que no pudo detectar cada tweet que fuera una estafa en la conversación, y sería retador diferenciar estos *tweets* entre los reales, solicitando ayudar utilizando únicamente los filtros de Twitter API.

Acerca de Metabase Q

Metabase Q lidera la ciberseguridad en América Latina, apoyando a clientes con una seguridad más rápida, más eficiente y escalable para acelerar la innovación digital. Combinada con servicios administrados, investigación y desarrollo, y las mejores tecnologías disponibles, Metabase Q es la fuente confiable para una ciberseguridad integral. La compañía invierte en desarrollar el talento y estándares con el apoyo de negocios globales y líderes gubernamentales para asegurar que América Latina tenga la mejor base de ciberseguridad para un crecimiento digital seguro.

Para conocer más sobre Metabase Q, contáctanos en:

contact@metabaseq.com
+1 (628) 225-1281
+52 55 2211 0920

// Referencias

Twitter bots pose as support staff to steal your cryptocurrency, Lawrence Abrams, Dec. 7, 2021. https://www.bleepingcomputer.com/news/security/twitter-bots-pose-as-support-staff-to-steal-your-cryptocurrency/

Tweeptwist, banbreach, Apr. 28, 2018. https://github.com/banbreach/tweeptwist

Tweepy, Joshua Roesslein. 2022 https://docs.tweepy.org/en/stable/index.html

Twitter API Documentation. Twitter, 2022. https://developer.twitter.com/en/docs