



PROYECTO PERSONAL

TEMA: ATTACK SURFACE



ALUMNO: ADRIÁN BENNASAR POLZIN

ASIGNATURA: SEGURETAT EN SISTEMES INFORMÀTICS

PROFESOR: BARTOLOMÉ JAIME SERRA CIFRE

Introducción

El uso del software se ha popularizado a lo largo de las últimas décadas y hoy en día es esencial para ejecutar los procesos complejos del mundo actual.

Esto conlleva un gran problema, ya que el software de hoy en día esta lleno de vulnerabilidades en la seguridad, que invitan al ataque. Los atacantes son especialmente incitados a atacar sistemas de software que contienen datos delicados, que desembocan en las llamadas “data breaches”. Las razones para estas “data breaches” pueden ser financieras, sociales, políticas, ridiculizar a la víctima, etc.

Algunos ejemplos de ataques importantes recientes son [\[1\]](#):

- Vtech Technologies (Datos personales expuestos de 6.4 millones de niños)
- Under Armour (El ataque obtuvo datos personales de 150 millones de usuarios de la aplicación de la empresa)
- Aerolíneas como Air Canada, British Airways, etc (Datos personales robados de 9,8 millones de clientes)

Para proteger los datos sensibles, es esencial considerar la ubicación de estos y si un atacante puede alcanzar o acceder a ellos en su ubicación.

Attack Surface

A partir de lo investigado, creo que una manera de definir el concepto attack surface [\[2\]](#) es el conjunto de maneras para los atacantes de acceder a una arquitectura y generar posibles amenazas para esta. O, dicho de otra manera, consiste en las ubicaciones de datos sensibles que pueden ser alcanzadas por los atacantes.

Entonces, cuanto mayor es la attack surface, mayor es la probabilidad de recibir un ataque.

Los atacantes usan recursos para atacar a la víctima, por esta razón los recursos de la victima forman parte de la attack surface.

Tengo entendido que el escaneo de puertos es el método tradicional para mapear la superficie de ataque de una red. Al enviar paquetes TCP o UDP a direcciones IP y leer las respuestas, los exploradores de puertos como **nmap** y **masscan** [\[5\]](#) devuelven la salida para mostrar qué puertos están abiertos. Un puerto abierto significa que un proceso en el sistema de destino está aceptando conexiones de la red. El uso de datos de escaneo de puertos para realizar pruebas funciona bien para una pequeña cantidad de hosts, pero no escala a las miles o decenas de miles de dispositivos en una red corporativa moderna.

Hay 3 pasos [\[3\]](#) generalizados para tratar con la attack Surface de un sistema/arquitectura:

- Visualizar: Visualizar el sistema, mapeando todos los dispositivos, caminos y redes. Es importante tener en cuenta la gran cantidad de dispositivos que intervienen en este paso^[4]:
 - * Servidores (Web, de aplicaciones, de bases de datos).
 - * Dispositivos finales: ordenadores de escritorio, portátiles, móviles.
 - * Redes, segmentos de red y la nube privada y pública.
 - * Dispositivos de networking, como routers, switches, etc
 - * Dispositivos de seguridad ya sean virtuales o físicos, como por ejemplo los Firewalls.

- Indicadores de exposición: corresponder cada indicador de una vulnerabilidad potencialmente expuesta al mapa visualizado en el primer paso. Algunos tipos de indicadores de exposición son:
 - * Vulnerabilidades software, como debilidades en aplicaciones, navegadores, plug-ins, sistemas operativos, sistemas gestores de BBDD, etc
 - * Malas configuraciones y falta de controles de seguridad en sistemas y software, que permiten a los hackers acceder a datos confidenciales y navegar hasta sistemas con software vulnerable.
 - * Normas demasiado permisivas, como cualquier norma en un firewall que permite pasar cualquier servicio proveniente de cualquier fuente y alcanzar cualquier destino.

- Indicadores de compromiso: indicadores de que un ataque ya ha sido exitoso. Algunos ejemplos son:
 - * Archivos malware detectados en dispositivos finales y servidores, comunicaciones de direcciones IP controladas por hackers o spammers y evidencia de tráfico de red anormal o peticiones de datos no habituales.
 - * Datos recogidos de registros de dispositivos de red y seguridad, herramientas de escaneo de redes, productos antimalware en dispositivos finales.

Reducción de la attack surface

Las estrategias básicas de reducción de la superficie de ataque incluyen las siguientes^[3]:

- **Reducir la cantidad de código en ejecución:** al tener menos código disponible para actores no autorizados, tenderá a haber menos fallas.
- **Reducir los puntos de entrada disponibles para los usuarios que no son de confianza.**
- **Eliminar los servicios solicitados por relativamente pocos usuarios:** al desactivar la funcionalidad innecesaria, hay menos riesgos de seguridad.

Aunque la reducción de la superficie de ataque ayuda a prevenir fallas de seguridad, no mitiga la cantidad de daño que un atacante podría infligir una vez que se encuentre una vulnerabilidad.

Hay herramientas como **Metasploit**^[5] que tienen configuraciones automáticas para enviar exploits a cada puerto abierto, pero los usuarios generalmente tienen la opción de probar todos los ataques posibles contra todos los puertos, lo que puede llevar mucho tiempo, o pueden elegir enviar solo ataques que coincidan con el servicio que se ejecuta en el puerto.

La última opción se basa en una detección de servicios precisa, a veces denominada **banner grabbing**, y muchos servicios no devuelven información de banner válida o útil cuando se consultan. Como resultado, se pueden pasar por alto vulnerabilidades.

Por ejemplo, si consideramos una superficie de ataque con 1000 servidores web, la consola de Metasploit *msfconsole* muestra 327 exploits para servidores HTTP (web) Linux y Windows. Probar cada uno de estos exploits requeriría **327.000 intentos** de exploit separados.

Sin embargo, si hipotéticamente el 75% de estos dispositivos pudieran **agruparse** con precisión en 20 grupos, entonces 245.250 de estas pruebas podrían **reducirse** a 6.550. Esto deja el número total de pruebas requeridas en 88.290, una reducción del 73%.

Una vez que se encontró un exploit exitoso en un sistema en un clúster, se puede verificar rápidamente en el resto. Esto se traduce en pruebas más rápidas con menos impacto en redes y dispositivos.

Sin embargo, he visto que se han empezado a investigar y desarrollar estrategias mucho más detalladas que lo comentado anteriormente. Un ejemplo de estas estrategias o métodos es el presentado en el artículo **Modeling and Reducing the Attack Surface in Software Systems**^[4]:

- 1) Modelar el o los sistemas software, identificando en el sistema software dónde se encuentran los datos delicados, con elementos visuales de componentes y flujos, como se muestra en la **figura 1**.
- 2) Evaluar el modelo para identificar cuales de los lugares donde se encuentran los datos delicados son alcanzables por los potenciales atacantes, lo que resulta en una identificación profunda de lo que compone la attack surface.
- 3) Reducir la attack surface a través de modificar el modelo para:
 - a) Mover los datos delicados.
 - b) Ofuscar los datos delicados.

- c) Denegar el acceso a los datos delicados.
- d) Eliminar la necesidad de seguir almacenando dichos datos delicados.




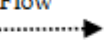







Drawing Elements		Description
		Data flow elements are labeled with numbers; all other elements are labeled with letters.
SD Use Circle		Identifies where SD is used.
SD Data Store		Identifies where SD is stored.
SD Data Flow		Identifies the movement of SD.
Non-SD Data Flow		Identifies the movement of non-SD.
Attack Surface Reduction Elements		
Merged SD Use Circle		Identifies where one or more use circles have been merged, with corresponding deletion of original circles as needed.
Merged SD Data Store		Identifies where one or more SD stores have been merged, with corresponding deletion of original stores as needed.
Obfuscated SD Data Store		Identifies where SD in a data store has been obfuscated (e.g., encrypted, anonymized).
Obfuscated merged SD Data Store		Identifies where one or more SD stores have been merged and obfuscated, with corresponding deletion of original stores as needed.
SD Data Store Put Offline		Identifies where a SD store has been put off line with an appropriate secure updating strategy.
SD Data Store Deleted		Reflected in the model by the removal of the SD data store; no modeling icon needed.
Obfuscated SD Data Flow		Identifies the movement of obfuscated (e.g., encrypted, anonymized) SD from one location to another.
Reduced Accessibility to SD Data Flow		Encloses use circles and data stores that execute on the same computing platform, thus reducing access to traversing data flow.
Description Element		
Legend		Descriptions of above labeled elements.

FIGURA 1
George O. M. Yee
"Modeling and Reducing the Attack Surface in Software Systems"

A continuación, presento un ejemplo del proceso(resumido) aplicado en una empresa de venta online de merchandise:

A. Identificando la attack surface con el modelo

Se crea el modelo base: se dibujan los SD Use Circle y SD Data store juntamente con los caminos de flujo de datos entre ellos. Se etiquetan los SD Use Circle y SD Data store con letras, y el flujo de datos con números. El modelo base para este caso quedaría de la siguiente manera:

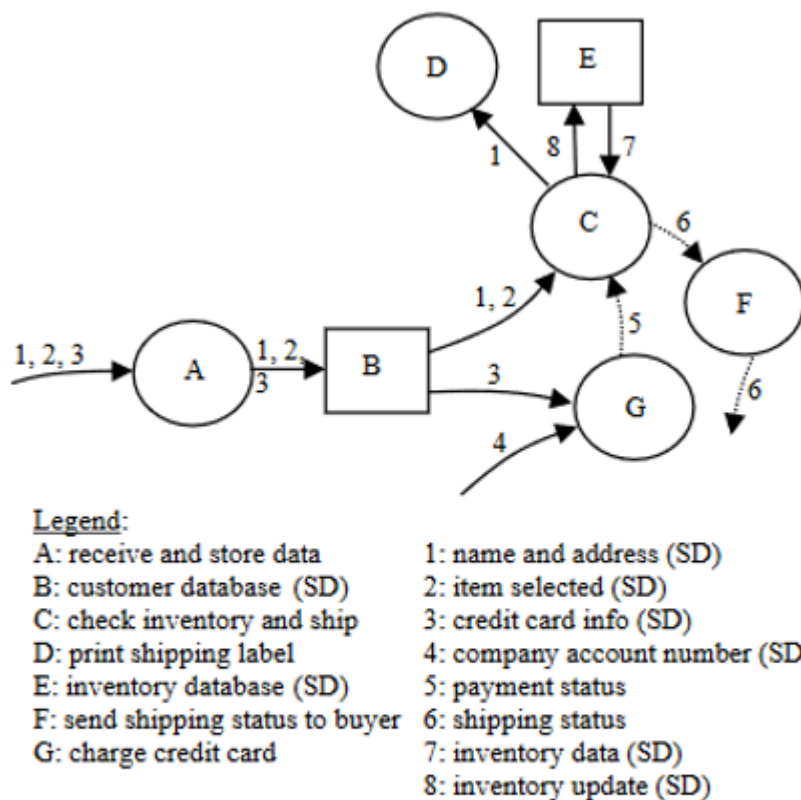


FIGURA 2

George O. M. Yee

"Modeling and Reducing the Attack Surface in Software Systems"

Una vez está montado el modelo base, se inspecciona para identificar la attack surface y se enumeran las posibles localizaciones donde un atacante podría atacar y comprometer los datos sensibles desprotegidos. La efectividad y sentido de estas enumeraciones entiendo que dependerá de la experiencia y conocimiento en seguridad y sistemas. A continuación, se registran estos resultados en una "Attack surface table" formada por 3 columnas:

	Attack Surface	Example Potential Attacks
1	(path into A / 1, 2, 3); (path into B / 1, 2, 3); (path into C / 1, 2); (path into D / 1); (path into G / 3); (path into G / 4); (path into C / 7); (path into E / 8)	Man-in-the-middle attack compromises SD.
2	(A / 1, 2, 3); (C / 1, 2, 7, 8); (D / 1); (G / 3, 4)	Trojan horse or hacker attack on use circle compromises SD.
3	(B / 1, 2, 3), (E / 7, 8)	SQL or insider attack on data store compromises SD.

FIGURA 3
George O. M. Yee
"Modeling and Reducing the Attack Surface in Software Systems"

B. Reduciendo la attack surface

1. **Se empieza aplicando elementos de reducción:** para cada localización alcanzable por un atacante en la "Attack surface table", se intenta eliminar la localización de la attack surface mediante la aplicación de uno o más elementos de reducción del modelo.

Por ejemplo: cambiar funcionalidad de manera que ciertos datos delicados ya no necesiten ser almacenados en el mismo lugar. Se representan estos cambios en el modelo.

2. **A continuación, se transfieren los cambios al diseño real:** se proponen los cambios al equipo de desarrollo de sistemas software. Si hay suerte, el equipo de desarrollo aceptará algunos de los cambios, pero puede que descarten otros debido a factores como el desempeño, restricciones económicas, modularidad, dificultad de implementación, etc.

Aplicando el análisis y los consecuentes cambios sobre el modelo base se obtiene la siguiente versión del modelo. Los cambios son:

- Accesibilidad reducida al elemento SD Data Flow.
- Uso del elemento Obfuscated SD Data Flow (Encriptación sin repudiación para el SD Flow hacia A, por ejemplo, SSL, y encriptación para los SD Flow hacia C y G.)
- Localizaciones con potenciales ataques SQL eliminados de la attack surface mediante el elemento ofuscado SD Data Store(encriptación).
- El Merged SD Use Circle se ha usado para reducir aún más la attack surface mediante la fusión de la funcionalidad D con la funcionalidad C.

Estos cambios se pueden ver de forma gráfica en las siguientes figuras:

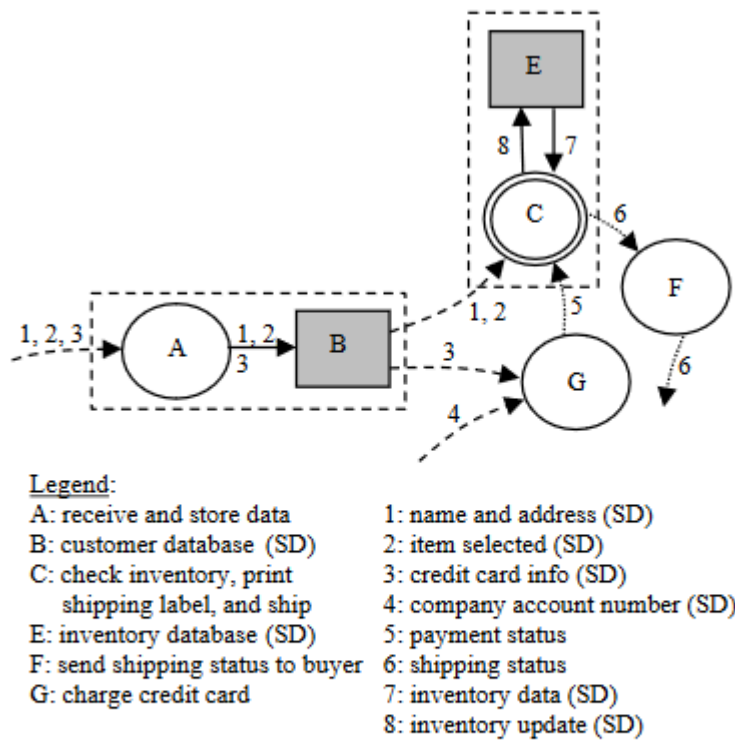


FIGURA 4
George O. M. Yee
"Modeling and Reducing the Attack Surface in Software Systems"

	Attack Surface	Example Potential Attacks
2	(A / 1, 2, 3); (C / 1, 2, 7, 8); (G / 3, 4)	Trojan horse or hacker attack on use circle compromises SD.

FIGURA 5
George O. M. Yee
"Modeling and Reducing the Attack Surface in Software Systems"

En mi opinión, sin ser un experto en seguridad, con todos estos ajustes implementados, ahora disponemos claramente de una estructura **mucho más segura**. Pero aun así, algo que he aprendido en este curso es que **nunca** se alcanza el 100% de seguridad.

DETECCIÓN DE UN PROBLEMA REAL APLICABLE A BALEARES (RELACIONADO CON EL TEMA ASIGNADO)

Un problema real reciente se puede identificar a través de una evaluación del **año 2020** de INTERPOL^[6] del impacto de **COVID-19** en la ciberdelincuencia, que ha mostrado un **cambio significativo** en el objetivo de los individuos y las pequeñas empresas a las grandes corporaciones, gobiernos e infraestructura crítica.

Con organizaciones y empresas que implementan rápidamente redes y **sistemas remotos** para ayudar al personal que trabaja desde casa, los delincuentes también **se están aprovechando** de las mayores vulnerabilidades de seguridad para robar datos, generar ganancias y causar interrupciones.

En un período de cuatro meses (**de enero a abril de 2020**), uno de los socios del sector privado de INTERPOL detectó unos **907.000** mensajes de spam, **737 incidentes** relacionados con **software malicioso** y **48.000 URL maliciosas**, todas **relacionadas con COVID-19**, datos que me resultan **bastante preocupantes**.

La mayor dependencia en línea de personas de todo el mundo también está creando nuevas oportunidades, y muchas empresas e individuos no se aseguran de que sus defensas cibernéticas estén **actualizadas**.

Las conclusiones del informe subrayan una vez más la necesidad de una **cooperación** más estrecha entre el sector público y el privado si se quiere abordar de manera eficaz la amenaza que el COVID-19 también representa para nuestra salud cibernética.

Los hallazgos clave destacados por la evaluación de INTERPOL del panorama de la ciberdelincuencia en relación con la pandemia de COVID-19 incluyen:

- **Estafas en línea y phishing:** los actores de amenazas han revisado sus estafas y esquemas de phishing en línea habituales. Al implementar correos electrónicos de phishing con el **tema COVID-19**, que a menudo se hacen pasar por **autoridades gubernamentales y de salud**, los ciberdelincuentes **atraen** a las víctimas para que proporcionen sus datos personales y descarguen contenido malicioso. 19 temas para el phishing y el fraude en línea desde el brote.
- **Dominios maliciosos, typosquatting relacionado con COVID-19, registro de dominios parecidos a los de las webs de farmacias:** aprovechando la mayor **demand**a de suministros médicos e información sobre COVID-19, ha habido un aumento significativo de ciberdelincuentes que **registran nombres de dominio** que contienen **palabras clave**, como “**coronavirus**” o “**COVID**”. Estos sitios web fraudulentos sustentan una amplia variedad de actividades maliciosas que incluyen servidores C2, implementación de malware y phishing. Desde **Febrero**

hasta **Marzo de 2020**, se detectó y reportó a la INTERPOL por parte de un socio del sector privado, un **incremento del 569%** en registros maliciosos, incluyendo malware y phishing, y un **incremento del 788%** en registros de alto riesgo.

DISEÑO DE UNA SOLUCIÓN AL PROBLEMA

- **Para hacer frente al phishing**, lo primero que recomendaría a las empresas es un **Secure Email Gateway**:

Un **Secure Email Gateway** se utiliza para filtrar emails maliciosos y ponerlos en cuarentena automáticamente aislados de los buzones de los usuarios. Hay varios SEC disponibles hoy en día. Uno de los que recomendaría es [Proofpoint](#), debido a su completitud.

Proofpoint funciona de la siguiente manera:

- **Protección multi-capa de Email**: ofrece un alto nivel de eficacia de seguridad con su solución de seguridad de Email de varias capas. La Email Gateway se basa en una pila de varios niveles de los propios motores de Proofpoint:

- Anti-virus.
- Anti-spam.
- Detección de phishing.

Todos los emails son escaneados en tiempo real antes de ser entregados, captando y bloqueando automáticamente las amenazas. Cabe destacar que los **administradores** pueden ver desgloses detallados del proceso, de manera que pueden determinar de dónde provienen las amenazas de correo y por tanto **profundizar más** en la mejora de la seguridad de la empresa.

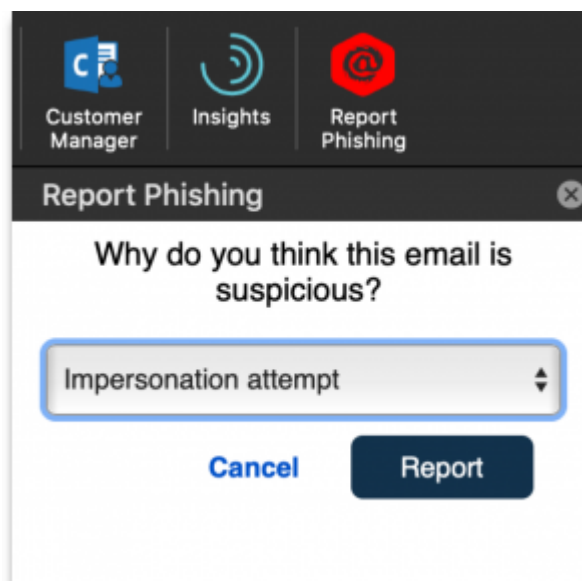
- **Advanced Threat Protection**: la **sandbox de URL y archivos adjuntos** de Proofpoint otorga protección contra ataques de phishing. A través de un **antivirus y sandboxing**, verifica los enlaces y archivos adjuntos en **tiempo real** para evitar que los usuarios abran archivos adjuntos maliciosos o visiten sitios web de phishing.

Sin embargo, uno de los desafíos relacionados con el phishing es que una vez que un Email de phishing está dentro de un buzón, o una cuenta ha sido comprometida y está enviando Emails de phishing internos, puede ser difícil para el administrador acceder al buzón e identificar y eliminar la amenaza. Por esta razón

recomendaria **combinar** lo anterior con una **plataforma de protección Post-Delivery** como [IRONSCALES](#).

Las plataformas de protección Post-Delivery consisten en proteger a los usuarios de las amenazas que hay ya en el buzón. Típicamente utilizan algoritmos potenciados por **machine learning** e **inteligencia artificial**, como los que hemos visto en la **asignatura Inteligencia Artificial** de este curso a los que se entrena con atributos típicos de los Emails de phishing. Una vez hecho esto se aplican a los Emails que los usuarios envían y reciben, para detectar Emails sospechosos.

IRONSCALES también permite a los usuarios reportar emails que creen que son phishing directamente desde su buzón, lo que ayuda tanto a los demás usuarios como a los administradores a identificar más rápidamente y remediar los ataques de phishing:



- **De cara al problema del typosquatting**, una de las posibles soluciones que recomendaria a las empresas es registrar dominios parecidos a los suyos que **redirijan** al usuario al dominio correcto.

Por ejemplo un usuario quiere entrar en la web de una empresa sanitaria llamada:

www.covidFighters2021.com

pero se **equivoca** y escribe:

www.codivFighters2021.com

este segundo dominio **incorrecto** redijirá al usuario al dominio correcto y evitará que el dominio incorrecto sea propiedad de algún **cybersquatter malicioso**.

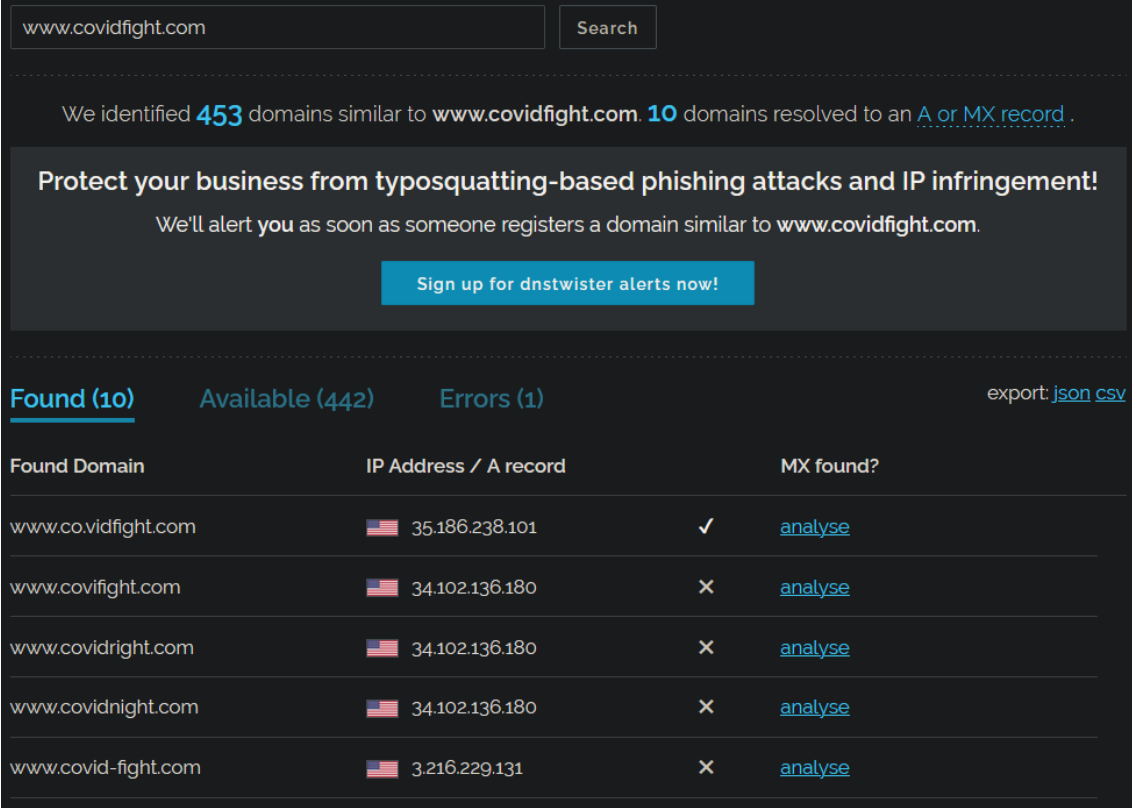
Esta solución parece tener sentido, pero pensar en todos los dominios parecidos al que se quiere proteger puede ser una **tarea tediosa y poco precisa**.

Por esta razón recomendaría el uso de una herramienta diseñada para automatizar este proceso:

DNStwister: <https://dnstwister.report/>

¿Cómo funciona **DNStwister**? Es muy simple, la empresa que quiera proteger un dominio, debe introducirlo en el campo de texto de la herramienta y esta hará un barrido de dominios existentes parecidos al indicado.

Esto permitirá a la empresa identificar rápidamente posibles casos de typosquatting relacionados con su dominio, y podrán proceder a reclamarlo. Para verlo en acción, podemos utilizar el ejemplo del dominio www.covidfight.com:



www.covidfight.com Search






We identified **453** domains similar to **www.covidfight.com**. **10** domains resolved to an [A or MX record](#).

Protect your business from typosquatting-based phishing attacks and IP infringement!

We'll alert you as soon as someone registers a domain similar to **www.covidfight.com**.

[Sign up for dnstwister alerts now!](#)

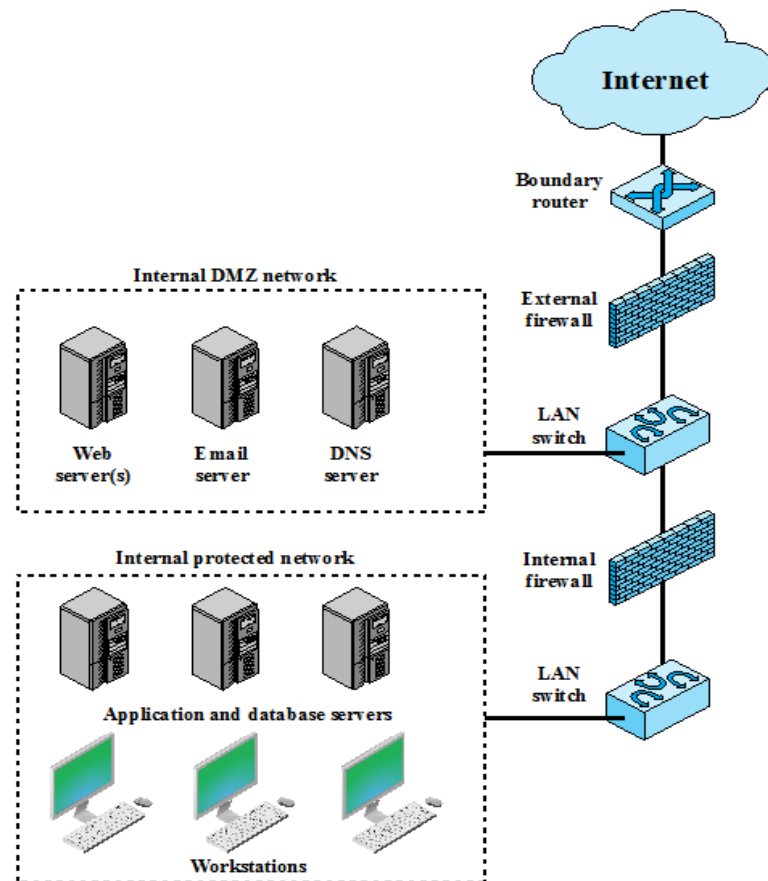
Found (10) **Available (442)** **Errors (1)** export: [json](#) [csv](#)

Found Domain	IP Address / A record	MX found?
www.co.vidfight.com	 35.186.238.101	✓ analyse
www.covifight.com	 34.102.136.180	✗ analyse
www.covidright.com	 34.102.136.180	✗ analyse
www.covidnight.com	 34.102.136.180	✗ analyse
www.covid-fight.com	 3.216.229.131	✗ analyse

Por otra parte, para evitar que los propios usuarios de la empresa sean víctimas del typosquatting y pongan en peligro la seguridad de la red de la empresa, recomendaría a la empresa la posibilidad de utilizar su **propio servidor DNS**, con una lista de dominios a bloquear. La manera de realizar esta configuración depende del servidor DNS en concreto utilizado.

De esta manera, si un usuario quiere acceder a un dominio de uso frecuente por la empresa y lo escribe incorrectamente, no se establecerá conexión con dicho dominio.

La estructura de la empresa debería tener entonces una forma parecida a la siguiente (**estructura estudiada en la parte práctica de esta asignatura**):



Como se puede observar, el servidor DNS utilizado por la empresa, en lugar de encontrarse en la internet, está en una **Demilitarized Zone** dentro de la red de la empresa.

Bibliografía

- [1] George O. M. Yee, “[*Attack Surface Identification and Reduction Model Applied in Scrum*](#)” - **AÑO 2019**.
- [2] Xinlin Liu, Jianhua Huang, Weifeng Luo, Quingming Chen, Peishan Ye, Dingbo Wang, “[*Research on Attack Mechanism Using Attack Surface*](#)” - **AÑO 2020**
- [3] https://en.wikipedia.org/wiki/Attack_surface - **AÑO 2019, 2020**
- [4] George O. M. Yee, “[*Modeling and Reducing the Attack Surface in Software Systems*](#)” - **AÑO 2019**
- [5] Douglas Everson and Long Cheng, “[*Network Attack Surface Simplification for Red and Blue Teams*](#)” - **AÑO 2020**
- [6] <https://www.interpol.int/en/News-and-Events/News/2020/INTERPOL-report-shows-alarming-rate-of-cyberattacks-during-COVID-19> - **INTERPOL – AÑO 2020**