Automatizando la ciberseguridad: Mi herramienta para detectar servidores web de una organización

Introducción

¿Sabías que muchas empresas no conocen todos sus servidores web activos? Es más, ¿sabías que muchas de estas empresas tienen programas de *Bug Bounty*? En mi TFM, exploré cómo los analistas de ciberseguridad, especialmente los *Bug Hunters*, utilizan la enumeración de activos para auditarlos y desarrollé una herramienta que automatiza este proceso con el mínimo esfuerzo por parte del analista, permitiéndole centrarse en auditar los activos en vez de enumerarlos.

Algunos *Bug Hunters* enfrentan su desafío con una metodología de "go deep", es decir, seleccionan un dominio de la empresa (normalmente el principal) y lo analizan a fondo para descubrir fallos. Otros, en cambio, prefieren trabajar con programas que tienen un *scope* más amplio y optan por expandir la superficie de ataque, buscando subdominios o servidores adicionales de la empresa. Esto les da más opciones para encontrar errores al tener una mayor cantidad de activos que examinar. Para ello, muchos analistas recurren a técnicas manuales de enumeración de activos, como el uso de *Shodan*, o a herramientas de automatización como las creadas por *Project Discovery*. Sin embargo, ambos enfoques tienen sus inconvenientes. La enumeración manual consume mucho tiempo, lo que reduce las opciones de encontrar las vulnerabilidades. Por otro lado, las herramientas de *Project Discovery* no ofrecen una visualización de resultados tan clara como *Shodan* ni permiten descubrir activos en redes IPv4 específicas.

Con mi TFM, busqué abordar estas limitaciones, creando una solución que libera a los analistas de seguridad de la tarea de enumerar los activos digitales de una empresa y les permite dedicar su tiempo a auditar esos sistemas de manera más efectiva.

Implementación

Para programar la herramienta, decidí usar Python debido a sus diversas librerías que me facilitaron el trabajo. Además, opté por un enfoque basado en programación orientada a objetos, dividiendo el proyecto en dos enfoques principales, uno para el descubrimiento de activos a partir del de una organización y otro a partir de dominios proporcionados por el analista. En ambos casos, apliqué programación concurrente para ganar velocidad, especialmente en tareas como peticiones web y resolución de DNS.

En cuanto a los servicios externos, me apoyé en servicios como *crt.sh* y *merklemap.com*, además de herramientas como *Smap*, *certgraph* y *Masscan*. También utilicé técnicas pasivas para enumerar posibles *virtual hosts*, optimizando los diccionarios de dominios a probar según la distancia entre dominios y las relaciones encontradas en el historial de certificados.

Como cada analista tiene preferencias distintas sobre los datos que desea auditar, implementé varios modos de ejecución. Esto permite al analista elegir entre un enfoque que prioriza el tiempo de ejecución y la certeza de los datos, o uno que ofrece mayor cantidad de información, incluyendo la enumeración de *virtual hosts*. También implementé un modo medio si no se quiere ser tan extremista en cuanto a volúmen de datos y certeza de estos.

Finalmente, desarrollé un *frontend* con *Flask* para visualizar los datos enumerados, que se almacenan en una base de datos *SQLite*, haciendo más fácil el acceso y análisis de la información.

Resultados

Para probar los modos de ejecución, he ejecutado cada modo de la herramienta diez veces para la empresa Booking, la cual tiene un programa de Bug Bounty. H hecho una media de los resultados ya que estos varían entre ejecuciones debido a los cambios dinámicos en la infraestructura de la organización, estado de la red del analista, comportamiento indeterminista de algunas herramientas…

En cuanto al enfoque basado en Open Scope tenemos los siguientes resultados:

Open Scope	Modo corto	Modo medio	Modo largo
Tiempo de ejecución	48 minutos	122 minutos	161 min
IPs encontradas	37	120	121
Hosts encontrados	50	321	376
Virtual hosts encontrados	2	35	38

Y en cuanto al enfoque basado en Wildcard Domains:

Wildcard Domains (4 dominios)	Modo corto	Modo largo
Tiempo de ejecución	9 minutos	25 minutos
IPs encontradas	66	60
Hosts encontrados	200	219
Virtual hosts encontrados	11	19

Frontend

En cuanto a la visualización de resultados, podemos ver como se ve la interfaz web que he diseñado, el cual dispone de un buscador, aunque se pueden ver todas las respuestas a la vez si así se desea: