Hacking infraestructuras

OpenWebinars

ÍNDICE

BUSCANDO OBJETIVOS: SHODAN

- Shodan es un motor de búsqueda en el que, a diferencia de Google y otros buscadores, no podemos buscar, por ejemplo, una imagen o un texto.
- Este motor de búsqueda está enfocado únicamente a buscar sistemas y servicios conectados a internet.
- Por este motivo, Shodan está clasificado como uno de los motores de búsquedas más peligrosos, por todo el contenido que tiene.
- Podemos hacer todo tipo de búsquedas dentro de su ámbito, incluso podemos utilizar los dorks que incluye este motor de búsquedas, como por ejemplo el dork country:us: que, nos permite buscar por países, en este caso eeuu

- Shodan recoge datos de todos los servicios, incluyendo HTTP (puerto 80,8080), HTTPS (puerto 443, 8443), FTP (21), SSH (22), Telnet (23), SNMP (161) y SIP (5060).

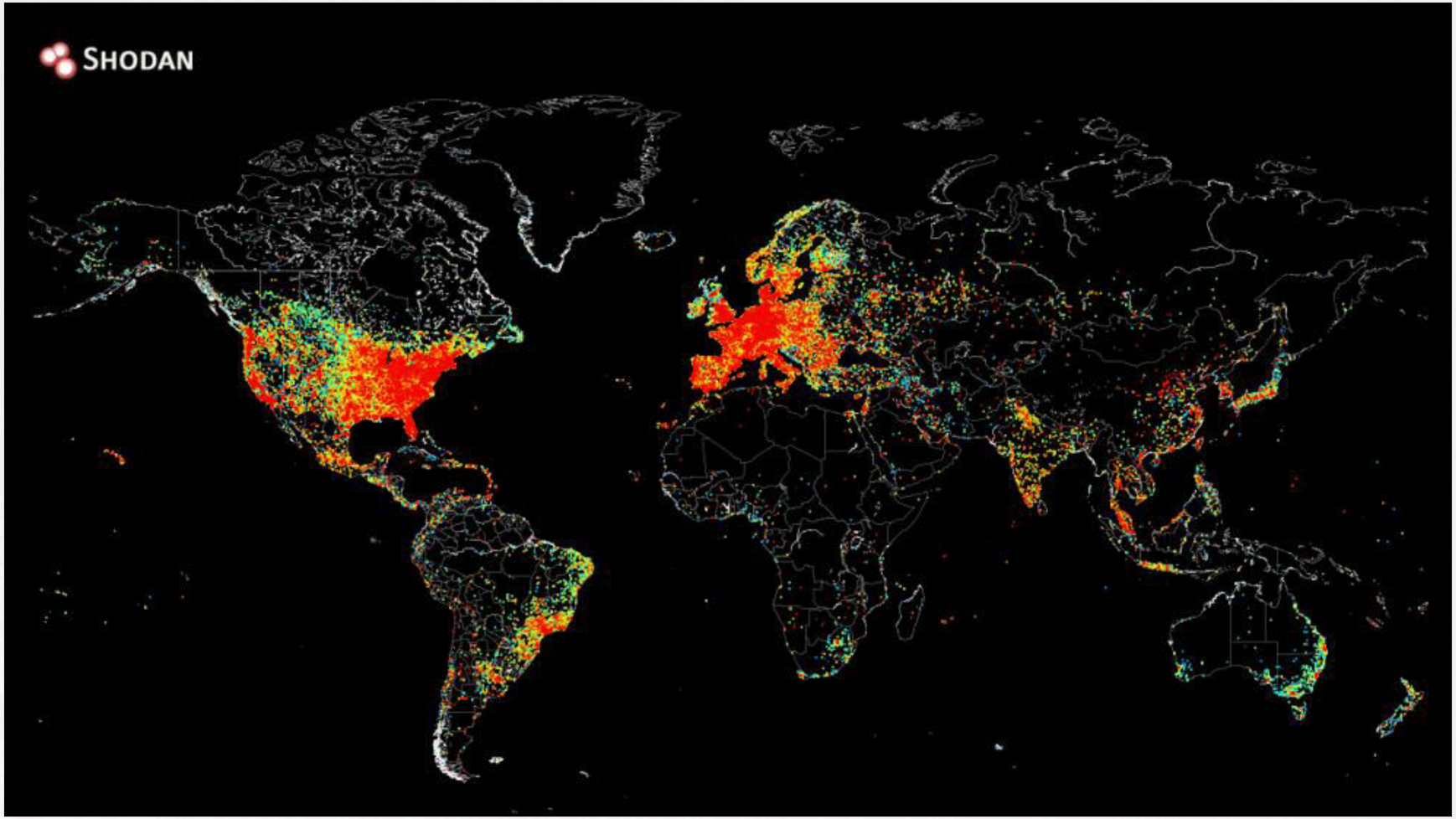
- La parte más peligrosa y negativa de esta detección es que todos estos dispositivos se encuentran conectados a Internet sin que sus dueños sean conscientes de los peligros y riesgos a nivel de seguridad, y por tanto, sin contar con la aplicación de medidas protectoras básicas como el nombre de usuario o una contraseña fuerte y robusta.

BÚSQUEDA DE SHODAN A TRAVÉS DE FILTROS:

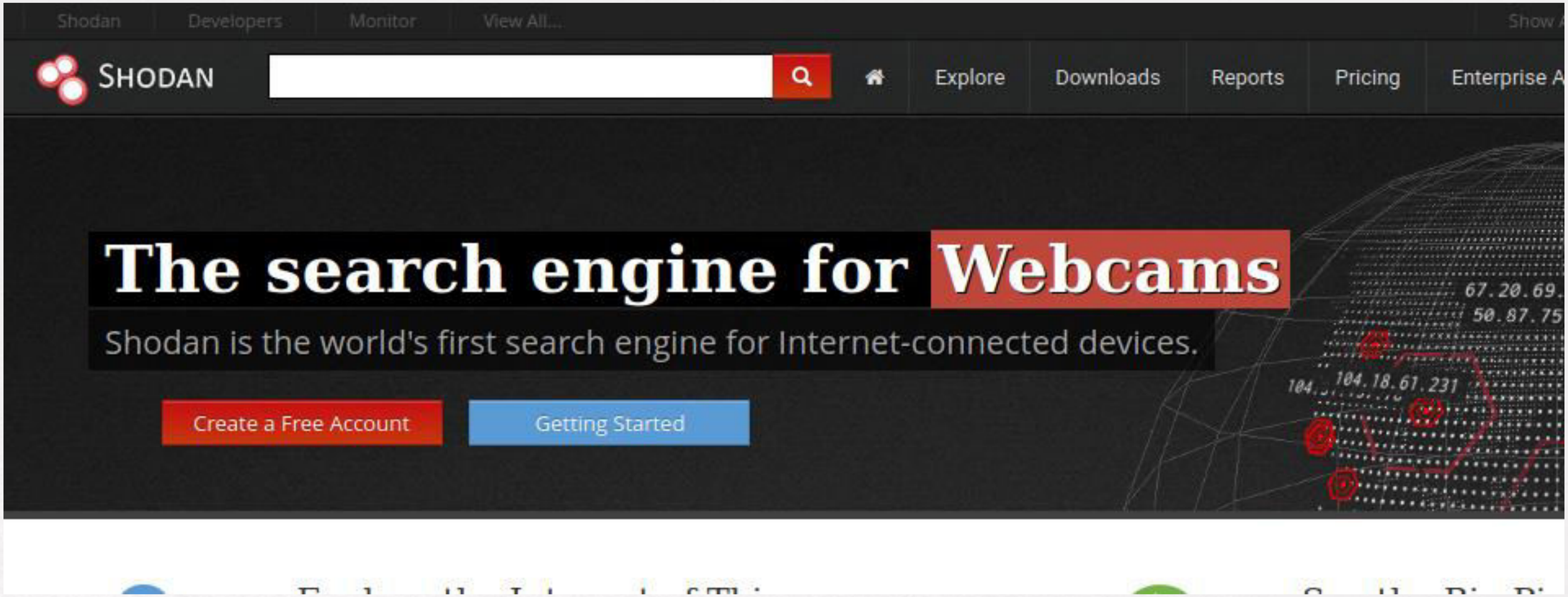
Las cuentas gratuitas en Shodan permiten buscar a través de los siguientes filtros:

- **Country**: Permite encapsular la búsqueda reduciendola a un país específico.
- **City**: Filtro por ciudad.
- **Port**: Permite realizar cada búsqueda dependiendo del puerto abierto o el servicio que se esté ejecutando.
- **Net**: Para buscar una IP específica o rangos de IP.
- **Hostname**: Este filtro sirve para las búsquedas relativas al texto que le indiquemos en la parte del hostname.
- **Os**: Según el sistema operativo.

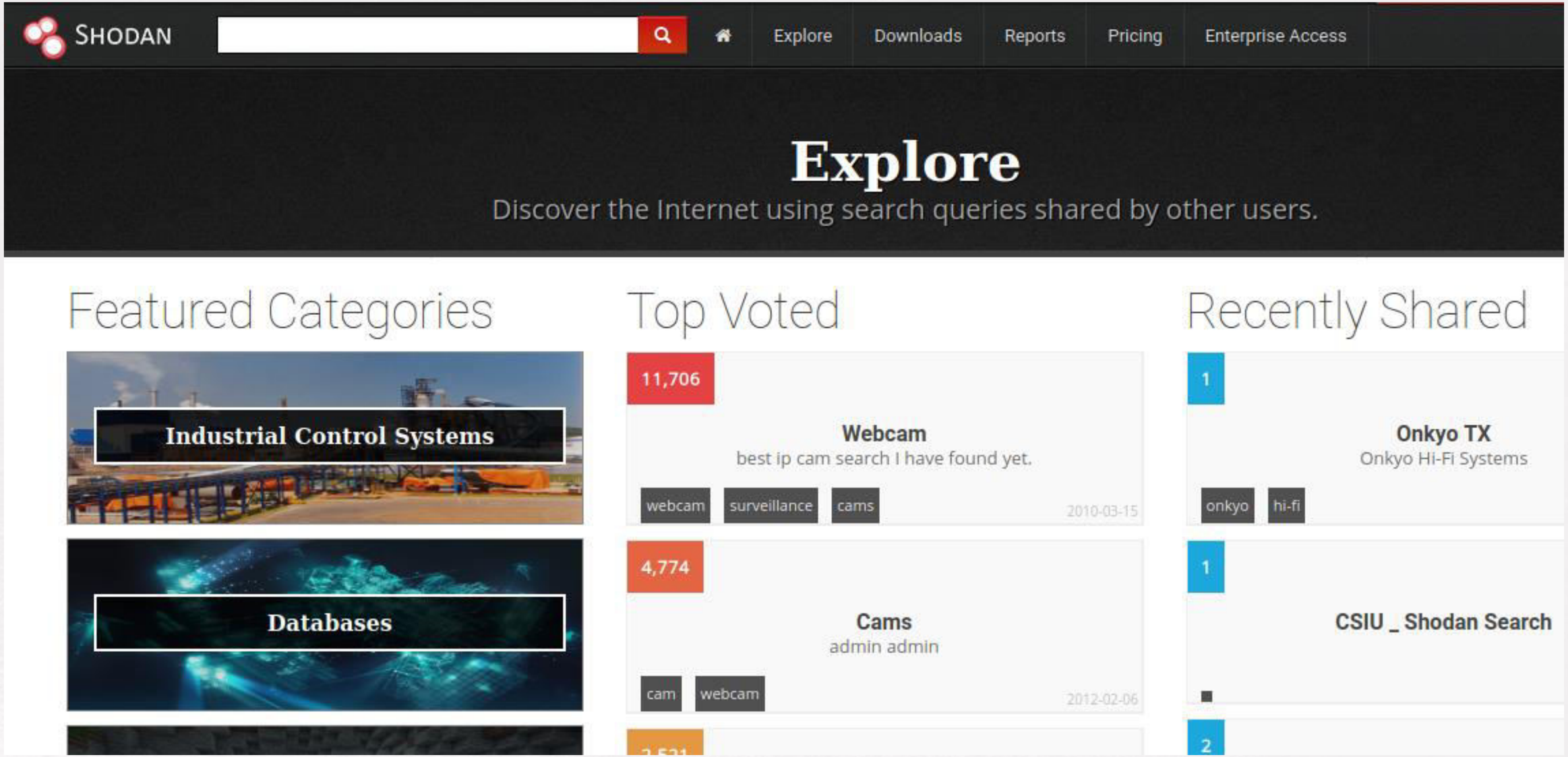
Webcams, neveras inteligentes o lectores de matrículas son algunos de los objetos que se pueden encontrar en la plataforma.



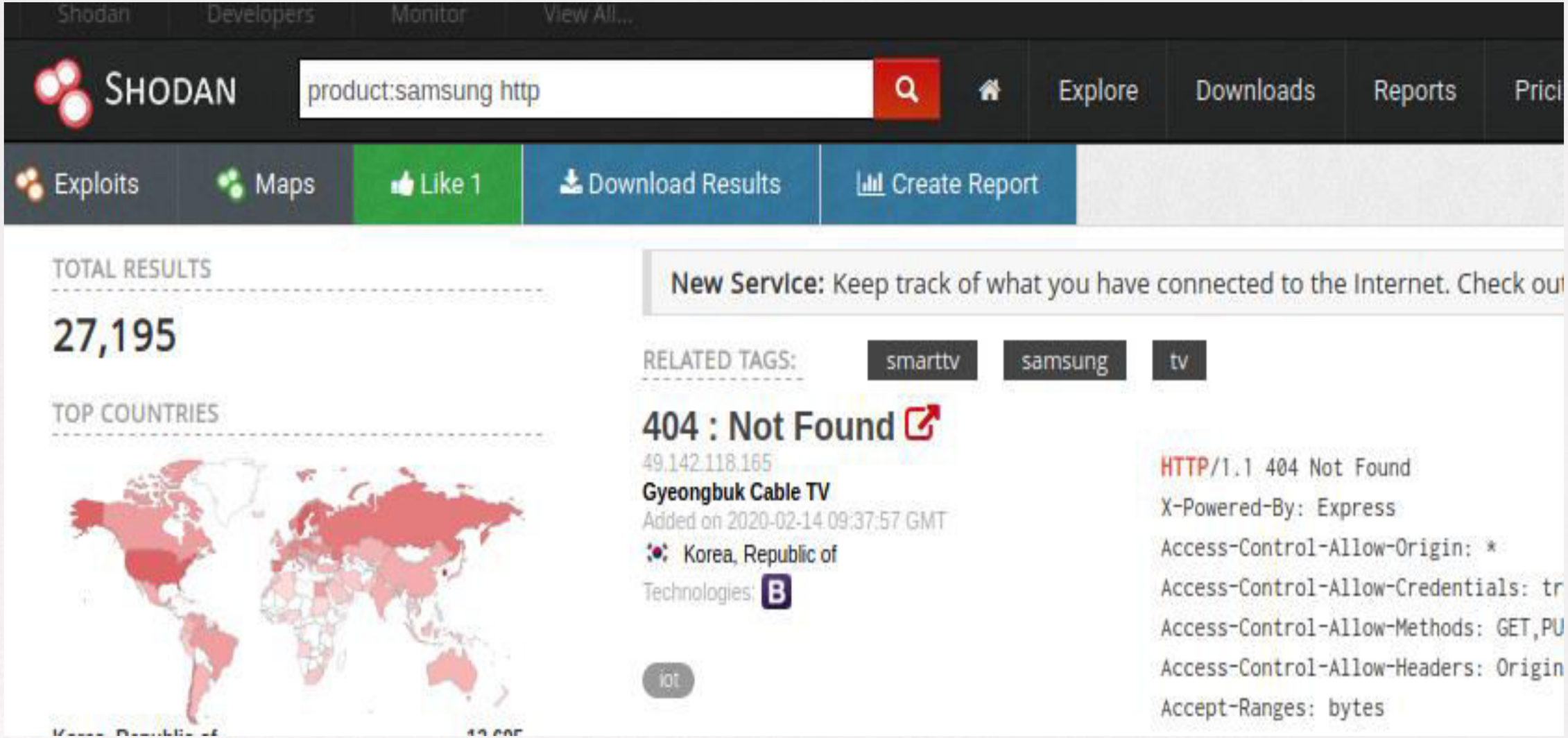
- Iniciando shodan -> shodan.io



- **Búsquedas predeterminadas -> "explore"**

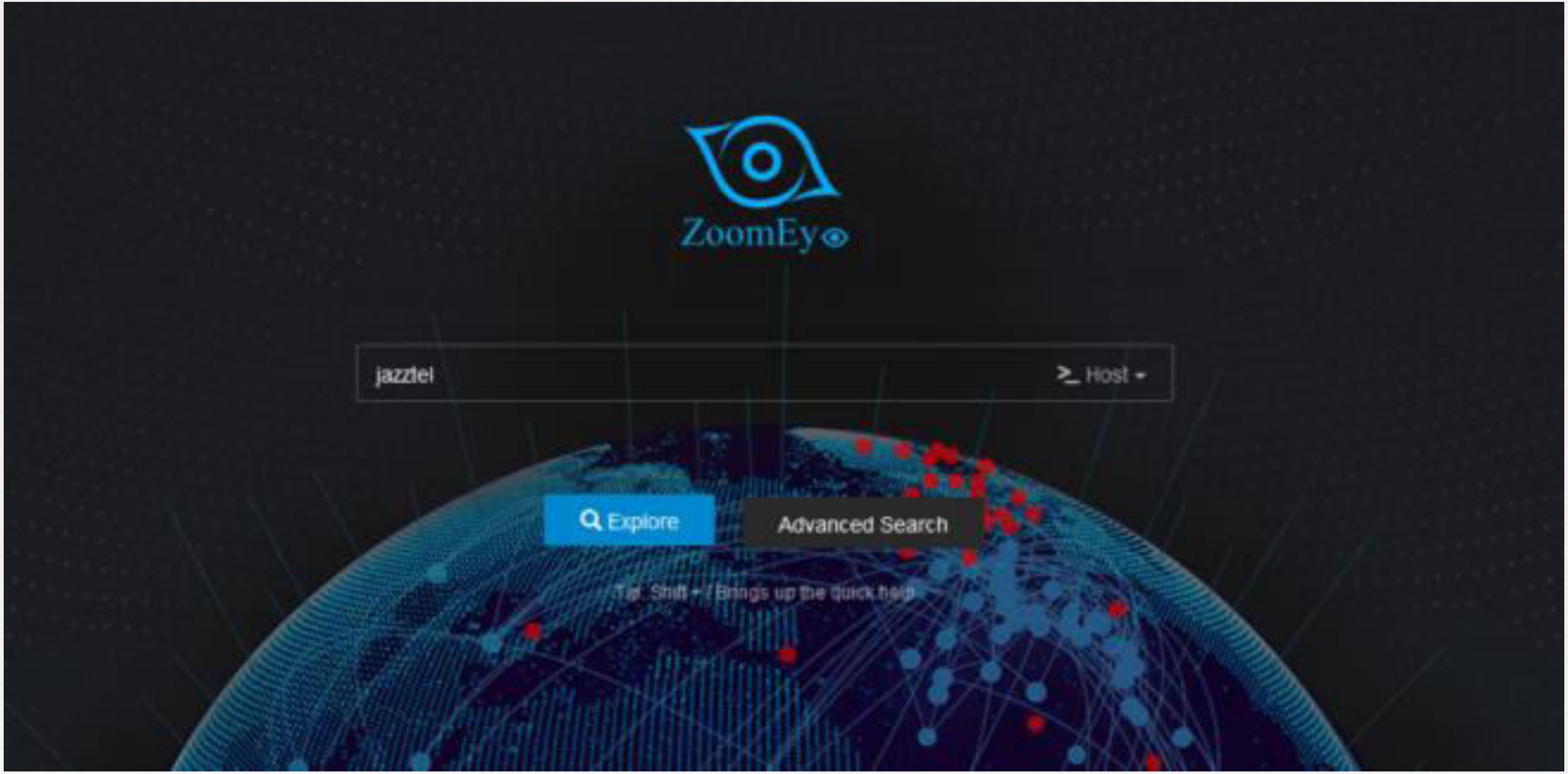


- **Búsqueda tv samsung en shodan.**

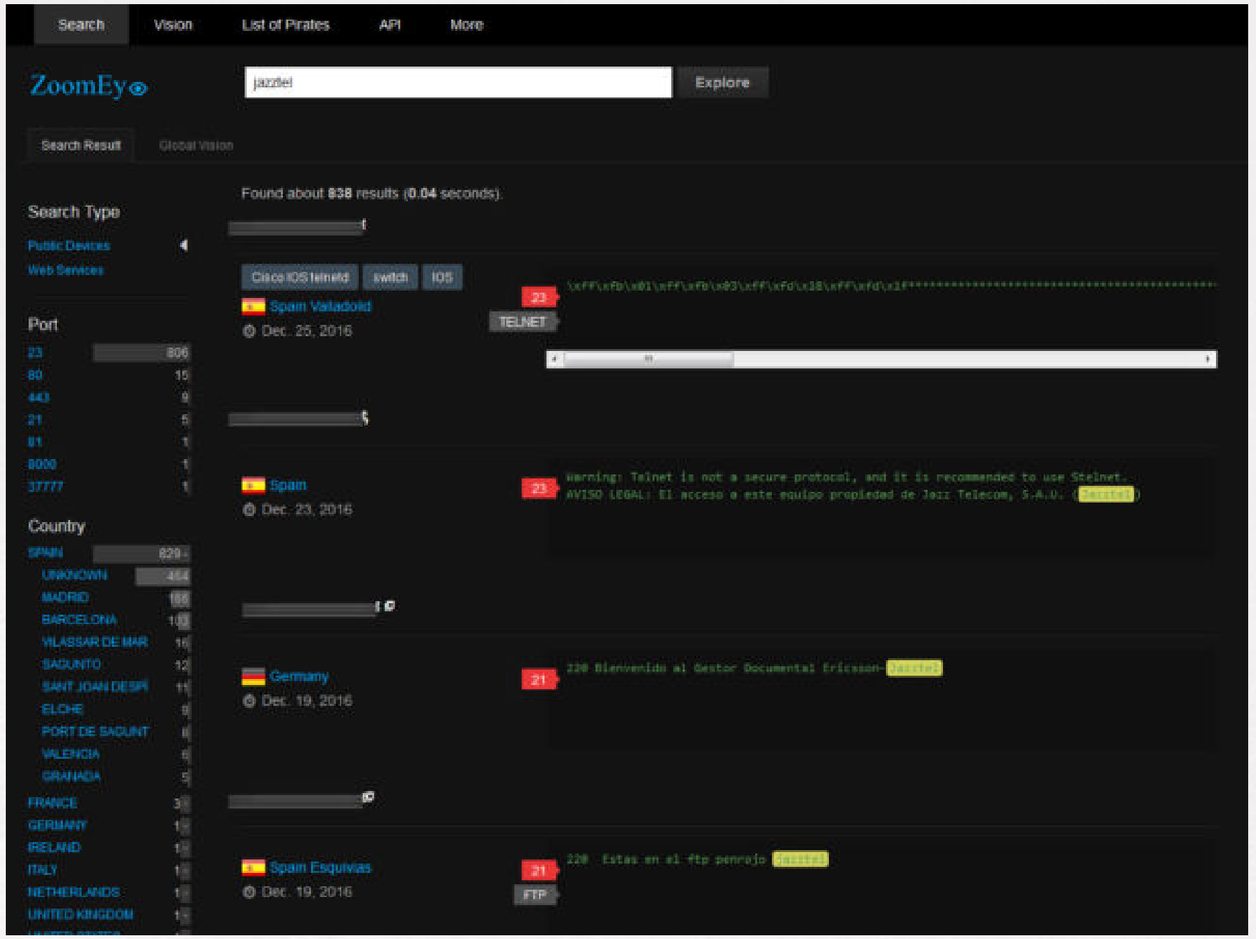


ZOOMEYE, EL OJO QUE TODO LO VE

- Zoomeye es un motor de búsqueda que nos va a permitir encontrar hosts y también webs que cumplan una serie de requisitos.
- Si por ejemplo queremos buscar servicios en Internet con la cadena "Jazztel", Zoomeye es capaz de proporcionarnos un listado de direcciones IP públicas y relacionándolas con el país, donde aparezca dicha cadena.
- También es capaz de buscar las direcciones IP públicas que tengan habitado y expuesto a Internet un determinado servicio, como por ejemplo un servidor web Apache, un servidor FTP visitado e incluso proftpd.
- No solo es capaz de buscar cadenas y servicios, sino también dispositivos loT que están permanentemente conectados.

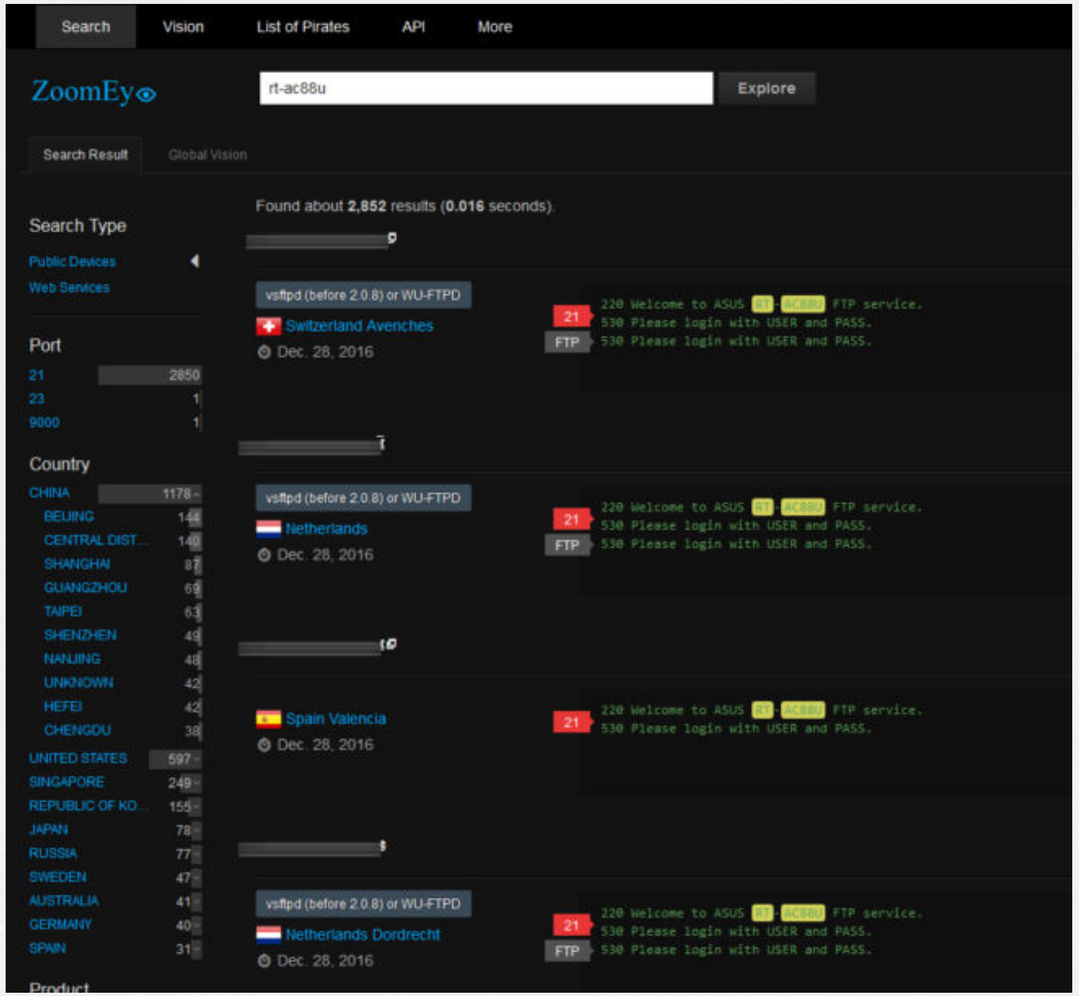

ZoomEye
jazzdel
Host
Explore
Advanced Search
Tip: Shift + / Brings up the quick help

- Si por ejemplo ponemos la cadena "vsftpd"„ nos encontraremos con los servidores FTP . Todos estos servicios son públicos, y Zoomeye nos los muestra, igual que hace Shodan.



- Para los usuarios más avanzados, Zoomeye nos permite realizar una búsqueda avanzada para encontrar posibles objetivos.

- Por ejemplo, podemos definir el sistema operativo por el que queremos filtrar, la ciudad, el país, el tipo de dispositivo, el número de puerto que nos interesa analizar, el tipo de servicio que está proporcionando a Internet e incluso qué palabras clave queremos buscar.



GOOGLE DORKS

- Google dorks son combinaciones de operadores de búsqueda especiales que se utilizan para extraer información valiosa o sensible desde Google. Es un término despectivo ya que *dork* en inglés significa idiota. Además se denomina "googledork" a una persona inepta o tonta según lo revelado por Google.

¿Pero cómo sucede esto?

- Simple, los robots de Google que indexan contenido (es lo que hace que un sitio sea posicionado de acuerdo a las palabras clave que se encuentran) son capaces de interpretar todo tipo de archivos, no sólo páginas Web.

- Por lo tanto, si un idiota deja un archivo con información sensible en un directorio Web que permite ser listado, será accedido e indexado por los robots de Google. La cuestión de fondo, es, ¿por qué alguien querría dejar un archivo con información sensible dentro de un directorio que es accesible públicamente a través del protocolo HTTP?

- La base de datos de Google Hacking está clasificada de acuerdo a diferentes categorías:
  - **Puntos de entrada:** Ejemplo, buscar el backdoor php c99 shell utilizando el dork *filetype:php intext:"!C99Shell"*

- **Archivos que contienen nombres de usuario.**

  Ejemplo, buscar cuentas de usuarios en sistemas , utilizando el dork inurl:"/root/etc/passwd" intext:"home/:"

- **Directorios sensibles.**

  Ejemplo, buscar rutas a directorios home utilizando el dork *intitle:"Apache Status" "Apache Server Status for"*

- **Detección de versión de servidor Web.**

  Ejemplo, buscar servidores Apache utilizando el dork intitle:*"Apache Status" "Apache Server Status for"*

- **Archivos vulnerables.**

  Ejemplo, buscar paneles de control de proveedores de hosting utilizando el dork intitle: *"Control Panel" "Control Panel Login" ArticleLive inurl:admin -demo*

- **Servidores vulnerables.**

  Ejemplo, buscar consultas SQL cacheadas en sitios Wordpress utilizando el dork inurl:/wp-content/w3tc/dbcache/

- **Mensajes de error.**

  Ejemplo, buscar mensajes de error de PHP utilizando el dork inurl:".*php?=.php"intext:"Warning: include" -inurl:.html -site:"php.net" -site:"stackoverflow.com" -inurl:"forums"*

- **Archivos con información interesante.**

  Ejemplo, buscar mapas de sitios utilizando el dork filetype:*xml inurl:sitemap*

- **Archivos que contienen contraseña.**

  Ejemplo, buscar claves probadas SSL utilizando el dork*"BEGIN RSA PROVATE KEY" filtype:key -github*

- **Información sensible de compras online.**

  Ejemplo, buscar versiones vulnerables del software X-Cart utilizando el dork intext: *Powered by X-Cart: shopping cart software" -site:x-cart.com*

- **Datos de redes y vulnerabilidades.**

  Ejemplo, buscar estadísticas de Webalizer utilizando el dork site.*./webalizer intitle: "Usage statistics"*

- **Formularios de acceso (login).**

  Ejemplo, buscar el acceso al panel de administración de sitios joomla! utilizando el dork inurl:*/administrator/index.php?autologin=1*

- **Dispositivos online (impresoras, cámaras,etc.).**

  Ejemplo, buscar el portal de acceso de routers Mikrotik utilizando el dork intitle:*"RouterOs router configuration page"*

- **Reportes de vulnerabilidades.**

  Ejemplo, buscar el plugin Age Verification de Wordpress utilizando el dork inurl: *wp-content/plugins/age-verification/age-verification.php*

  - Site: static.owl.ly/docs/intext:@gmail.com/Password.
  - Filetype: sql intext:wp_users phpmyadmin.
  - Intext: "Dumping data for table "orders".
  - " Index of /wp-content/uploads/backupbuddy_backups" zip.
  - Zixmail inurl:s/login?
  - inurl:/remote/login/intext:"please login" intext:"FortiToken clock drift detected"
  - inurl:WebInterface/login.html
  - inurl:dynamic.php?page=mailbox
  - inurl:/sap/bc/webdynpro/sap/ "sap-system-login-oninputprocessing"
  - intext:"Powered by net2ftp"

ATAQUES MiM

- Un ataque Man in the Middle (en adelante MiM) o ataque de intermediario es el método por el cual un hacker interviene en el tráfico de datos de dos partes vinculadas entre sí en una comunicación haciéndose pasar por cualquiera de ellas, haciéndoles creer que se están comunicando entre ellos cuando en realidad es el intermediario quien recibe la comunicación.

TIPOS DE ATAQUE MAN IN THE MIDDLE

- **Ataques basados en servidores DHCP:**
  - En este ataque, el hacker usa su propio ordenador en una red de área local a modo de servidor DHCP, que en resumidas cuentas sirve para asignar dinámicamente una dirección IP y configuración adicional a cada dispositivo dentro de una red para que puedan comunicarse con otras redes.
  - En cuanto un ordenador establece la conexión con una red de área local, el cliente DHCP reclama datos como la dirección local o la dirección de la puerta de acceso predeterminada, entre otros.

- **ARP caché poisoning:**
    - En este caso nos referimos al protocolo ARP que permite resolver IPs en redes LAN siempre que un ordenador quiera enviar paquetes de datos en una red. Para ello, es imprescindible que conozca el sistema del destinatario. Cuando hace una petición ARP, está enviando al mismo tiempo las direcciones MAC y la IP del ordenador que solicita la información, como la dirección IP del sistema solicitado. Si es correcta toda la petición, la asignación de direcciones MAC a IP locales se guarda en el caché ARP del ordenador solicitante.
    - El objetivo del ataque ARP cache poisoning es dar respuestas falsas en el proceso para lograr que el atacante usar su ordenador como punto de acceso inalámbrico o entrada a Internet. Si es exitoso, el ataque permite leer todos los datos salientes de los ordenadores atacados, aparte de registrarlos o de manipularlos antes de enviarlos al lugar correcto.

- **Ataques basados en servidores DNS:**

    - Este ataque tiene como objetivo manipular las entradas en el caché de un servidor DNS haciendo que den direcciones de destino falsas. Si han tenido éxito, los hackers pueden mandar a los usuarios de Internet a cualquier página web sin que nadie se dé cuenta.
    - El proceso se inicia cuando los datos del sistema de nombres de dominio se distribuyen por diferentes ordenadores de la red. Cuando alguien quiere acceder a una web lo suele hacer usando un nombre de dominio. También necesita una dirección IP determinada por el router que tenga el usuario, para enviar la solicitud. Si hay entradas en el caché, el servidor decidirá la IP con ayuda de otros servidores.

- **Simulación de un punto de acceso inalámbrico:**

  - Centrado en los usuarios de dispositivos móviles, este ataque consiste en recrear un punto de acceso inalámbrico en una red pública, como pueden ser las de una cafetería, un aeropuerto, etc.
  - El atacante prepara su ordenador para que actúe como una vía adicional de acceso a Internet, intentando engañar a los usuarios para que le proporcionen los datos de su sistema antes de que se den cuenta.
  - El peligro real viene si tu dispositivo se configura para comunicarse automáticamente con los puntos de acceso con mayor potencia de señal.

- **Ataque Man in the Browser:**
  - Por último, el ataque Man in the Browser consiste en que el atacante instala malware en el navegador de los usuarios de Internet con la finalidad de interceptar sus datos. La principal causa para verse infectado por este ataque es el hecho de tener ordenadores que no están correctamente actualizados y que, por ello, ofrecen brechas de seguridad muy visibles que dan camino libre para infiltrarse en el sistema.
  - El malware incluye programas en el navegador de un usuario de forma clandestina, registrando todos los datos que intercambia la nueva víctima con las diferentes páginas web que visita. Los hackers obtienen con este método la información que buscaban de forma muy rápida y sin demasiado esfuerzo.

CRACKING WIFI

- En los últimos años, las redes WiFi han recorrido un largo camino y siempre que su red esté usando el cifrado WPA2-AES, una contraseña fuerte, y tiene el WPS deshabilitado será extremadamente difícil de acceder. Hey! vamos a intentarlo.

- Un método para hackear las contraseñas cifradas de Wi-Fi es mediante la fuerza bruta, por lo que cada combinación de caracteres se intenta hasta que llegue a la correcta. Es teóricamente posible, pero esto puede tomar años en la práctica, especialmente si se utiliza una contraseña larga.

PASOS PARA DESCIFRAR CONTRASEÑAS WI-FI CON AIRCRACK-NG

- **Preparando el Adaptador:**

  - Compruebe que Kali puede detectar el adaptador abriendo la terminal y ejecutando el comando: **airmon-ng**
  - Deshabilitar cualquier proceso que pueda interferir con su captura de paquetes, escriba el siguiente comando: **airmon-ng check kill**
  - Ahora poner el adaptador en modo de monitoreo con el comando: **airmon-ng start wlan0**

- Anotar el nombre de la interfaz y ejecutar airodump-ng (nombre de la interfaz) para listar las redes que le rodean, por ejemplo:

//Ejecutar el siguiente comando

//para empezar a escuchar todas las conexiones WiFi disponibles

airodump-ng wlan0mon

- **Encontrando la Red WiFi Objetivo:**

Encuentre la lista, la red que tiene como objetivo descifrar su contraseña. Anote el "BBSID" y "CH" (canal).

A continuación, después de haber visto su objetivo presione las teclas Ctrl+C y ejecute el comando:

airodump-ng -c 6 --bssid 02:08:22:7E:B7:6F --write fichero wlan0mon

- **-C:** Es un número del canal de la red que aparece en la columna CH (En la salida de la pantalla anterior). En mi caso es 6.

- **-bssid:** Es la dirección MAC de la red objetivo. En mi caso es fichero y el BSSID es 02:08:22:7E:B7:6F
- **-write:** Es el archivo de captura en la que se guardarán los paquetes. En mi caso la he nombrado como "fichero".
- **-w:** Es el prefijo del nombre de archivo que contendrá el handshake.
- **wlan0mon:** La interfaz inalámbrica.

- **Lanzando un ataque deauth:**

Ahora abrimos una nueva terminal e iniciamos el ataque deauth para desconectar todos los clientes conectados a la red. Ésto le ayudará en la captura del handshake. Ingrese el comando:

aireplay-ng-0 10 -a 02:08:22:7E:B7:6F -e XXXX wlan0mon

- **-0:** Se utiliza para el ataque deauth.
- **10:** Es el número de paquetes deauth para ser enviados.
- **-a:** Es la dirección MAC de la red WiFi objetivo.
- **-e:** Es el ESSID de la red objetivo, es decir, su nombre.

Después de lanzar el ataque deauth y conseguir el HANDSHAKE WPA, pulse Ctrl+C