Redes de computadoras

Practicas básicas de hacking ético

El objetivo de esta práctica es dar a conocer los mecanismos involucrados en distintas fases del hacking ético.

OJO! Ninguna de las técnicas a utilizar puede usarse contra sitio web legítimos ya que incurren en delitos, la práctica es con fines didácticos y para aprender a tomar contramedidas de protección de los sistemas que están desarrollando.

Parte 1. Realicemos una sesión de Google hacking

Google hacking es una práctica común para verificar si la víctima posee vulnerabilidades visibles desde la web. Accede al manual en línea de los operadores de búsqueda de google, allí podrás ver para que sirve cada uno y ejemplos que podrías utilizar para las fase de recolección de datos de la víctima.

• site: busca todo lo relacionado al dominio.
• intitle: sitios relacionados al título.
• allintitle: sitios de títulos con todas las palabras definidas.
• inurl: pre sen te en el URL.
• allinurl: todo presente en el URL.
• filetype: tipo de archivo por buscar, extensión.
• allintext: todo presente en el texto por buscar.
• link: quién linkea a de terminado sitio buscado.
• inanchor: busca en el texto utilizado como link.
• daterange: busca entre rangos de fechas.
• cache: busca dentro de los sitios cacheados.
• info: información sobre el sitio web buscado.
• related: busca similares.
• author: autor de mensaje en Google Groups.
• group: busca pertenencia de grupo en Google Groups.
• phone book: busca números de teléfono.
• insubject: busca titulares de mensajes en Google Groups.

• define: busca el significado de determinado vocablo.

El uso del símbolo menos (-) para la exclusión de palabras es muy útil a la hora de buscar entre mucho material; lo mismo pasa con las comillas (“ ”) en caso de buscar una frase textual o el símbolo (+) para relacionar.

Realiza las siguientes búsquedas:

1. Realiza la siguiente búsqueda: filetype:sql "MySQL dump" (pass|password|passwd|pwd)
2. Realiza la siguiente búsqueda: inurl:(service | authors | administrators | users) ext:pwd
3. Realiza una búsqueda que contenga en el título: index.of.config

Realiza las búsquedas acompañadas del operador site:dominio_de_tu_trabajo (opsu.gob.ve, ula.ve, uis.edu.co)

Explica qué significa cada resultado obtenido

Parte 2. Crackea una contraseña utilizado john the ripper

Para esta sección de la práctica trataremos de encontrar password fáciles de romper por medio de la herramienta John de Ripper. Para ello van a realizar los siguientes pasos

1. Instala la herramienta John the ripper en tu linux: apt-get install john

2. Crea un archivo que se llame pass.txt con las siguientes líneas:

usuario1:6UI.2PI9E$K.9irP/QP9BTslIAXHeVL6iMRcCiLAh3r2hfYcZHns9zmGcqa8xt3HL1ImxRfSX3PFNlkQO9TrV87gWh/AuY60:17682:0:99999:7:::
usuario2:6mP.2PI9E$FAPNPl8Lm1pgms./6WffmtlxILFLNLBA62eRc4JbGTPjc6PN5/D.sn/up9WBzbRFMt0VICEOVB8QbhUW8ByX/.:17682:0:99999:7:::
usuario3:6nS.2PI9E$ugTi3mmuMFxKhYn2uaaUgTSyVtt3AwXWYBovNq2EC6/sMHpuFCiIEa6HpNiQR2heC8d07OYEXZT5MpaZXi7tk0:17682:0:99999:7:::

3. Ejecuta John de Ripper y dime cuales claves lograste descifrar y cuales son

Parte 3. Cómo defendernos de Google Hackers (contramedidas)

Los directorios de configuración web proporcionan información que permitirá a un hacker determinar si su sitio es vulnerable. Esta información NO está destinada a ser pública ya que contiene archivos con contraseñas dependiendo del nivel de seguridad o con contenido sobre los distintos puertos y permisos de seguridad.

Estos directorios pueden dar información sobre la configuración de un servidor web.

La razón principal de estas fugas de información es una política de seguridad inadecuada en relación con la información que se publica en Internet. Existen unos pocos métodos con los cuales podemos proteger nuestro servidor web.

Un servidor web de acceso público se utiliza por lo regular para almacenar información a la que se accede públicamente desde internet y si en realidad nos encontramos preocupados por mantener la información de manera privada, entonces la forma más fácil y adecuada es mantenerla lejos de este tipo de servidores. A pesar de que tales archivos o documentos se puedan mantener aislados, es fácil tener acceso a dichas páginas. Todos conocemos los riesgos asociados con el hecho de que se muestren los listados directorios, los cuales pueden permitir a un usuario ver la mayoría de los archivos almacenados en el directorio raíz principal y sus subdirectorios, etc.

Algunas veces, incluso el archivo ***.htaccess*** se muestra en el listado, este archivo es utilizado para proteger los contenidos de contenido del directorio del acceso no autorizado, pero una simple mala configuración puede permitir que este archivo se muestre en la lista y se pueda también conocer su contenido. Esto también es debido a que muchos administradores tienen la costumbre de cargar información importante en sus servidores para permitir el acceso desde cualquier lugar y que luego dichos contenidos son indexados por los rastreadores de los buscadores web.

Una de las reglas simples puede ser que los administradores de los sitios web agreguen un archivo ***robots.txt*** que define lugares específicos del directorio principal, de forma tal que el motor de búsqueda no lo explore y no lo almacene en su caché. Para protegernos de los buscadores, podemos utilizar el archivo *robots.txt* para evitar la indexación de tales documentos o directorios.

Ejemplo: `User-agent: *Disallow: /documentos`

También, para bloquear páginas web específicas o si no queremos que una página en particular sea indexada por algún motor de búsqueda, podemos utilizar algo como el meta tag "`meta name=’spider_name’ content=’NOarchive’`"

Ejemplos de Robots.txt

El siguiente ejemplo permite a todos los robots visitar todos los archivos:

```
User-agent: *
Disallow:
```

Esta entrada mantendrá alejados los robots de todos los directorios:

```
User-agent: *
Disallow: /
```

Podemos especificar directorios particulares que no queremos que sean indexados. El siguiente ejemplo mantendrá alejados los robots del directorio /infosec/ y sus subdirectorios:

```
User-agent: *
Disallow: /infosec/
```

Al no incluir el / final, también podemos evitar que las arañas (web spiders) hagan rastreo de los archivos contenidos en dicho directorio.

El siguiente ejemplo evitará que los robots de Google (googlebots) rastreen cualquier cosa en nuestro sitio, pero permite que otros robots accedan a todo el sitio:

```
User-agent: googlebot
Disallow: /
```

La siguiente meta-etiqueta (meta tag) evitará que todos los robots puedan rastrear cualquier enlace en nuestro sitio:

```
<META NAME="ROBOTS" CONTENT="NOINDEX, NOFOLLOW">
```

También podemos denegar o permitir que ciertas arañas puedan utilizar esta etiqueta:

Ejemplo: `<META NAME="GOOGLEBOT" CONTENT="NOINDEX, NOFOLLOW">`

Para obtener mayor información, podemos visitar: http://www.robotstxt.org/wc/exclusion.html#meta.

El dork de Google para verificar la existencia del archivo .htaccess es *intitle:index of "*.*htaccess*". Esto mostrará los sitios web que tienen el archivo .htaccess en un listado de directorios.

El listado de directorios debería ser deshabilitado a menos que este sea requerido. El listado de directorios también ocurre cuando el archivo principal del sitio web (index.html, index.php, etc.) definido en el servidor web, se encuentra ausente. En servidores web apache, podemos deshabilitar los listados de directorios utilizando un guión o un símbolo menos (-) antes de la palabra **Indexes** en al archivo **httpd.config**.

Realiza 3 recomendaciones de contramedidas en sitios web del dominio ula.ve que no han tomado previsiones de este tipo.