ACTIVIDAD 5 - ANÁLISIS DE VULNERABILIDADES

Presentado Por: Pablo Andrés Briceño

1. Que son los payloads en metasploit

Como ya sabemos un exploit es una vulnerabilidad en el sistema, y los payloads son esa carga que se ejecuta en esa vulnerabilidad [1].

Tenemos dos tipos de conexiones las cuales son **Bind** y **Reverse**, las cuales su principal diferencia es que con la primera nos conectamos a la víctima, teniendo el anonimato, en cambio con la segunda la víctima se conecta a nosotros, con ello no se asegura un anonimato, ya que dentro del payloads estaría nuestra ip [2].

2. A que se denomina fingerprinting y footprinting

Son etapas para ciberseguridad, donde footprinting es la primera etapa la cual consiste en realizar un test de recolección de información. La principal fuente de información es internet, y por último se realiza un filtrado para quedarnos con lo más importante [3].

La etapa fingerprinting, realiza la recolección de información directamente del sistema de una organización. Esta etapa es recomendable realizarla en una auditoria autorizada, para no tener problemas legales siendo el “atacante” [3].

3. Cuales servicios son los más comunes de sufrir vulnerabilidad

Estos son los siguientes:

- Simple Network Management Protocol (SNMP)
- Server Message Block (SMB)
- File Transfer Protocol (FTP)
- Simple Mail Transfer Protocol (SMTP)
- Virtual Private Network (VPN)

4. Que es OWASP Top 10

Es un proyecto abierto de seguridad en Aplicaciones Web, el cual se dedica a permitir que las organizaciones desarrollen, adquieran y mantengan aplicaciones y APIs en las que se puedan confiar [4].

- **Injection**, este se produce cuando se realiza un envío no confiable a un intérprete como parte de un comando o consulta. Con ello se puede engañar al interprete consiguiendo con ello que nos muestre la información que se supone debe estar protegida. Esto puede afectar mucho a la

empresa ya que a través del propio aplicativo web estarían robando información.

- **Broken Authentication**, las funciones que se relacionan con la autenticación y administración de sesiones que en la mayoría son implementadas de forma incorrecta. Esto es muy peligroso, porque se estaría dando acceso a los atacantes de que entren a nuestro sistema sin ninguna sospecha de que ese usuario está comprometido.
- **Sensitive Data Exposure**, muchas de las APIs no protegen los datos sensibles como debería, por ello los atacantes pueden realizar suplantación de identidad, robo o fraudes con la información financiera, etc. Esto no puede pasar en una empresa porque se iría a la quiebra ya que los clientes no se arriesgarían a que algo como eso les suceda.
- **XML External Entities (XXE)**, muchos procesadores XML antiguos o que se encuentren mal configurados evalúan referencias en documentos XML, consiguiendo realizar con ello un escaneo de puertos, ejecución remota de código y ataques de denegación de servicios.
- **Broken Access Control**, las restricciones en varias ocasiones no se aplican adecuadamente, haciendo que el o los atacantes exploten esta falla y consigan acceder a funciones o datos no autorizados, cuentas de otros usuarios, archivos confidenciales, etc. Esto es de vital importancia tenerlo en cuenta porque se debe proteger todo lo relacionado con las cuentas y los accesos que se tengan a nuestra aplicación web, no se debe correr el riesgo de que se difundan archivos confidenciales de nuestra empresa.
- **Security Misconfiguration**, la mala configuración de seguridad es un tema muy recurrente, el cual sucede al dejar la configuración predeterminada las cuales son siempre inseguras, configuraciones incompletas, almacenamiento en la nube abierta, entre otras. Este tema no debe pasar desapercibido porque con ello está en juego la reputación de la empresa, y la generación de confianza para los clientes.
- **Cross-Site Scripting XSS**, estos ocurren cada vez que una aplicación incluye datos no deseados o confiables en una página sin la respectiva validación. XSS permite a los atacantes que ejecuten scripts con los cuales se puede secuestrar sesiones de usuarios, redirigir a otros sitios maliciosos. En este punto sobra recalcar la importancia de la seguridad y la confianza que tiene el cliente al entrar a nuestro aplicativo web, por lo que con solo entrar puede estar infectado sin saberlo.
- **Insecure Deserialization**, esta conduce a menudo a la ejecución remota de código. Donde se puede realizar ataques de injection y ataques de escalada de privilegios.

- **Using Components with Known Vulnerabilities**, las bibliotecas, los marcos u otros modulos de software, los cuales se ejecuten con los mismos privilegios que la aplicación, puede generar una vulnerabilidad la cual se la puede explotar y con dicho ataque se puede generar una gran pérdida de datos o la pérdida del servidor. Esto es algo que la empresa no puede permitirlo porque si el servidor se lo pierde queda comprometida toda información interna y que pueda entrar.
- **Insufficient Logging & Monitoring**, la falta de monitoreo y registro, junto con la falta de respuestas a incidentes, permite que los atacantes puedan acceder más fácil, e incluso permite la permanencia del atacante. Esto en una empresa es imperdonable, es como si estuviera viviendo con el ladrón en la misma habitación y con todas las comodidades, algo que la empresa no puede dejar pasar.

5. A qué hace referencia este ejemplo de explotación:

nc[IP] 25

VRFY root

VRFY aaaaaaa

Hace referencia a una explotación de servicio SMTP donde el comando VRFY permite a un atacante determinar si existe una cuenta en un sistema, proporcionando una asistencia significativa a un ataque de fuerza bruta en las cuentas de los usuarios. VRFY puede proporcionar información adicional sobre los usuarios del sistema, como los nombres completos de los propietarios de las cuentas.

BIBLIOGRAFIA

[1] «Qué es un Payload», *OpenWebinars.net*, oct. 24, 2018. https://openwebinars.net/blog/que-es-payload/ (accedido may 07, 2020).

[2] ~ Cyberh99, «Payload Reverse o Bind ¿Cuál elegir?», *Cyberh99*, abr. 26, 2017. https://cyberh992017.wordpress.com/2017/04/26/payload-reverse-o-bind-cual-elegir/ (accedido may 07, 2020).

[3] «¿Que es footprinting y fingerprinting?» http://www.ticarte.com/contenido/que-es-footprinting-y-fingerprinting (accedido may 07, 2020).

[4] «OWASP Top Ten Web Application Security Risks | OWASP». https://owasp.org/www-project-top-ten/ (accedido may 07, 2020).