Universidad de Talca
Facultad de Ingeniería
Departamento de Ciencias de la Computación

Seguridad Informática

Tarea 1

Erik Regla
eregla09@alumnos.utalca.cl

27 de abril de 2020

1. Enunciado

Buscar 10 ejemplos de ataques a la Seguridad de la Información, describiéndolos y determinando para cada uno de ellos cual o cuales de las 3 propiedades fundamentales de la seguridad fueron vulneradas (Confidencialidad, Integridad y Disponibilidad).

2. Desarrollo

2.1. CWE-89: Neutralización impropia de elementos especiales usados en un comando [1]

Consiste en la ejecución de comandos arbitrarios sobre una base de datos SQL cuando este no está correctamente manejado en su driver o aplicación. Técnicamente cualquier punto de datos puede ser un vector de inyección, como por ejemplo, las variables de entorno, parametros, servicios internos o externos. Lo que junta a todos estos, es que un ataque puede enviar una carga maliciosa por medio de un intérprete. Ya que en SQL es interpretado, muchos desarrolladores se sienten tentados a escribir las consultas en texto plano, pero también olvidan que es necesario tratar los datos de entrada antes de desplegar o bien escapar caracteres. Entonces, es simple terminar una query del estilo "SELECT FROM something WHERE column='?'", con "'; DROP FROM something;" y borrar todo por ejemplo.

Dado que este ataque puede resultar en pérdida de datos, corrupción o fuga de datos y que eventualmente se puede tomar el control completo del sistema, este tipo de ataques vulnera los tres aspectos de seguridad.

[1] https://cwe.mitre.org/data/definitions/89.html

2.2. CWE-287: Autenticación impropia [2]

Consiste en ganar acceso como un usuario autorizado por medio del uso malicioso de credenciales o bien forjando las mismas. Para este ejemplo, basta con tomar algún sitio que utilize autenticación por medio de cookies y que no contraste su contenido con otros aspectos como geolocalización, IP de origen, etc. Está usualmente presente en sistemas CGI, en los cuales la autenticación por cookies es bastante común. Un ejemplo notable es un ataque realizado a Twitter en 2009 donde un usuario obtuvo acceso de administrador a un servidor dado que este no restringía el numero de intentos fallidos, por lo cual por medio de un ataque de fuerza bruta obtuvo la contraseña. [3]

Dado que dependiendo de la naturaleza del servicio comprometido se puede terminar tomando control total del sistema, este tipo de ataque compromete las tres propiedades fundamentales de seguridad.

2.3. CWE-202: Exposición de información sensitiva por medio de consultas [4]

Es bastante común encontrar aplicaciones en donde información sensible de otros usuarios los cuales no corresponden al autorizado son mostradas por medio de explotación maliciosa o bien porque la misma aplicación no filtra la información que entrega. Esto no está limitado solo a información explícitamente directa de la aplicación ya que en muchos casos es posible inferirla por medio de técnicas estadisticas o bien por medio de accidentes por parte de usuarios o desarrolladores.

Por ejemplo, las versiones iniciales de la aplicación de asistencia desarrollada por la universidad en sus primeras versiones presentaba la fuga de datos de alumnos y profesores al momento de registrar la asistencia.[5]

En este caso solo se ve comprometida la confidencialidad de los datos ya que no es posible ganar acceso ni comprometer la integridad de los mismos. Si bien la información de los alumnos es de dominio público, sus horarios de clases y los cursos que ellos están tomando no lo es.

2.4. Descubrimiento de Buckes en S3 debido a configuración defectuosa [6]

En AWS S3 es necesario configurar la seguridad de los buckets que se van a utilizar para así limitar el acceso a usuarios no autorizados. Sin embargo, ya que las politicas son por defecto presentarlo como público, muchas veces los desarrolladores llevando a producción pruebas conceptuales olvidan este aspecto. En este caso es relativamente fácil debido a la disponibilidad de los datos el realizar ataques de fuerza bruta para encontrar estos buckets y ya sea encontrar su dirección y muchas veces su contenido.

En este caso se ve comprometida la confidencialidad de los datos, sin embargo dependiendo de como haya sido configurado el bucket podría perfectamente afectar la integridad y la disponibilidad

[2] https://cwe.mitre.org/data/definitions/287.html
[3] http://www.wired.com/threatlevel/2009/01/professed-twitt/
[4] https://cwe.mitre.org/data/definitions/202.html
[5] https://github.com/KukyNekoi/crazy-unicorn
[6] https://blog.websecurify.com/2017/10/aws-s3-bucket-discovery.html

de los mismos datos (no del servicio).

2.5. CVE-2019-17026: Confusión de tipos en Mozilla Firefox [7]

En esta vulnerabilidad clasificada como zero-day, un atacante podía inyectar código malicioso para tomar el control del sistema por medio de la confusión de tipos del compilador Just-in-time de IonMonkey. Debido a su categoría de zero-day y parchado recientemente, la información de como este ataque fue realizado no está disponible públicamente aún [8], sin embargo, hay registros de ataques realizados y un tracker del bug en Mozlla.

En este ataque los tres aspectos de seguridad de la información fueron afectados ya que se puede obtener control total del sistema.

2.6. Android/Trojan.Dropper.Agent.UMX [9]

En este caso, la inyección de código malicioso de realizada al momento de la recepción de los equipos, ya que estos eran ofrecidos como telefonos de bajo costo por el gobierno estadounidense por el programa de asistencia para hogares en riesgo social. En este caso, se descubrió que el teléfono en cuestión (Unimax U686CL) venía preinstalado con aplicaciones con sospechas de ser malicioso siendo advertido por distintas fuentes. La prueba de esto fue de que el mecanismo de actualización del celular estaba conectado con una empresa la cual ya estabaes conocida por recopilar datos de usuario por medio de la inyección de backdoors y auto-instaladores.

En este tipo de ataque se vio vulnerada la confidencialidad de los datos pero no su disponibilidad ni su integridad, sin embargo, la integridad del dispositivo si se ve vulnerada debido a este factor externo.

2.7. CWE-223: Omisión deinformación relevante a la seguridad [10]

Otro problema comúnmente encontrado en los sistemas informáticos es la omisión de información que puede ayudar a encontrar problemas de seguridad. Por ejemplo, el almacernar la información de los login, pero no almacenar el número de intentos o de donde estos provienen restringe al desarrollador o mantenedor de la plataforma de deteectar problemas de seguridad relacionados a estos (por ejemplo, un ataque de fuerza bruta) por efectos de parchado como también de monitoreo.

Este tipo de errores por si solos no afectan la plataforma, solo al desarrollador, pero dependiendo de como estos escalen puede terminar comprometiendo todos los aspectos del sistema, ergo, los tres aspectos fundamentales de la seguridad de una plataforma.

[7] https://www.mozilla.org/en-US/security/advisories/mfsa2020-04/

[8] https://es-la.tenable.com/blog/cve-2019-17026-zero-day-vulnerability-in-mozilla-firefox-exploited-in-targeted-attacks

[9] https://www.darkreading.com/threat-intelligence/chinese-malware-found-preinstalled-on-us-government-funded-phones/d/d-id/1336771

[10] http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-1999-1029

2.8. Phishing [11]

El phishing no tiene nada de nuevo, sin embargo en esta noticia se informa de que un nuevo esquema de phishing apunta a el control de la persistencia de datos (a través de la conexión directa de la cuenta a Office365 por medio de una aplicación de terceros) que a robar la contraseña propiamente tal.

El problema principal es que otorgar los permisos de la aplicación entrega el control completo de los archivos y servicios de la parte afectada y dependiendo de las circumstancias, de la organización a la cual pertenezca, por lo que este ataque áfecta la confidencialidad, disponibilidad y la integridad de los datos.

2.9. Exposición de números de teléfonos por parte de Facebook [12]

Este es una instancia del problema general antes mencionado de fuga de datos, pero en este caso, hubo una filtración de identidades, números de teléfono y nombres debido a una base de datos que no estaba completamente asegurada. No está claro si el mecanismo utilizado fue un abuso de la API provista por Facebook o bien si los atacantes tuvieron acceso a la información por otro medio.

Si bien esta información es de caracter público (si el usuario así lo define), la misma aplicación tiene restricciones para poder visibilizarla si el usuario lo define de tal manera. Dado que este ataque consistía en el scrapping masivo de información solo la confidencialidad de información fue afectada para los usuarios que dentro de sus configuraciones de seguridad eligieron no mostrar públicamente esta información.

2.10. Encriptado punto-a-punto inexistente [13]

Conocido por todos y favorito de muchos, es conocido que Zoom no soporta encriptado punto-a-punto de las comunicaciones. Este punto es importante (ademas de todas las otras vulnerabilidades al sistema, intentos de control y extracción de datos por parte de la aplicación, etc), ya que la criptografía punto a púnto permite que una conversación solo sea escuchada por sus interlocutores, volviendola ilegible de la misma aplicación. En simple, zoom no protege la comunicación establecida de ellos mismos, pudiendo observar y extraer información de la misma sin el consentimiento del usuario (si bien está en la licencia, la licencia también habla de el encriptado).

Debido a esto, el punto de confidencialidad de información ha sido violado ya que la disponibilidad de la información o bien su contenido no ha sido afectado.

[11] https://krebsonsecurity.com/2020/01/tricky-phish-angles-for-persistence-not-passwords/
[12] https://www.cnet.com/news/millions-of-facebook-user-phone-numbers-exposed-online-security-researchers-say/
[13] https://theintercept.com/2020/03/31/zoom-meeting-encryption/