Nombre : Adrian Tendero Lara Fecha: 01/04/2019

CASO MOONPIG

Descripción general del caso

<https://blog.trendmicro.com/trendlabs-security-intelligence/high-profile-mobile-apps-at-risk-due-to-three-year-old-vulnerability/>

Según la noticia un investigador en ciberseguridad investigando el código de la API investigada descubrió que se podía acceder a la información personal de los clientes introduciendo un numero de nueve dígitos para emular el ID que se usa en el request header, Esto significa que hay muy gran problema en la confidencialidad de los datos en la aplicación.

Riesgos OWASP involucrados

El principal riesgo OWASP involucrado en este caso es el M4-Insecure Athentification, donde en el backend de la aplicación móvil, no comprobaba correctamente lo tokens de inicio de sesión pudiendo ser evadidos de forma sencilla como introducir un numero de 9 digitos.

Ya que al parecer el backend solo comprobaba el ID del usuario (el numero de 9 digitos) para comprobar si podía acceder a dicha información o no, pudiendo solicitar información sensible realizando la llamada a dicha información aplicando números de ID aleatorios hasta dar con una ID valida.

Según el articulo los números Id fueron creados en orden secuencial… lo que implica que si un atacante obtiene un ID podría obtener los ID la totalidad de los usuarios de la base de datos, con la vulnerabilidad anterior expuesta, esto implicaría poder hacer un listado de todos los clientes de la base de datos con(nombres, Apellidos , Fechas de nacimiento, números de tarjeta de crédito parciales, histoirial de compras y si el servidor lo guardaba historial de localizaciones)

En conclusión esto permitía acceso a la base de datos del backend poniendo en peligro todos lo datos contenidos en ella, incluyendo a otros usuarios de aplicaciones que utilicen dicho backend como son usuarios web ya que aunque el método de acceso es exclusiva de la aplicación, si el backend es compartido pueden acceder a la información sin problemas.

Explotación de la vulnerabilidad

Siguiendo el método explicado en el articulo, utilizando herramientas de debug y análisis de aplicaciones se puede investigar las llamadas a la API del backend, pudiendo identificar las peticiones de acceso a información privada no incluyen ningún tipo de comprobación de credenciales por tanto utilizando un proxy podríamos realizar un conjunto de llamadas aleatorios y si se comprueba que el método funciona el atacante podría automatizar el proceso creando un pequeño programa que realice la petición y guarde los datos confidenciales.

Respuesta de la empresa y consecuencias técnicas y de negocio

Las consecuencias técnicas son claras, que son los datos privados de los usuarios podrían haber sido robados de manera muy sencilla.

Las consecuencias para el negocio son tan graves como el fallo, empezando por una perdida de confianza y credibilidad de la propia empresa (daño de reputación) y las mas que probables multas por no cumplir las leyes de protección de datos vigentes.

La respuesta de la empresa fue el total secretismo de la vulnerabilidad ya que los usuarion no fueron notificados, de hecho se intenta ocultar el parche de seguridad como “mantenimiento rutinario” así como el tiempo en solucionar el fallo de seguridad que el investigador supuso que como muy tarde aparecería a principios del año siguiente (unos 6 meses) y al parecer (según el articulo) tardaron mas de 17 mesas (ya que dice el articulo que no hubo aun respuesta tras ese tiempo).

Medidas de protección específicas:

Implementar comprobaciones de autentificación y autorización para acceder a informacion sensible, realizar acciones de pruebas de la aplicación (dada la facilidad de acceso a la información sensible supongo que no se hizo ninguna) y aplicar algunas medidas de problamacion segura como puede ser limitar el numero de intentos fallido a un inicio de sesión ya que en el investigador dijo *“I hit my test users a few hundred times in quick succession and I was not rate limited”* siendo esto una gran Ayuda para un posible atacante.