Ace Threat Analysis and Modeling

Microsoft ha desarrollado una metodología de análisis y modelado de amenazas, basada en la combinación de ideas propias con las de parte del equipo de @stake y que recientemente ha incorporado a sus filas. Esta metodología ha ido evolucionando y recogiendo ideas de diversos enfoques.

La versión inicial, se basa en el uso de árboles de ataques para luego extrapolar las amenazas y realizar una clasificación y un ranking de estas con el fin de priorizar las actuaciones necesarias para mitigar el riesgo. Mientras que en la segunda versión de esta metodología, han intentado aclarar los conceptos de amenaza, ataque y vulnerabilidad, actualizando su herramienta de modelado y cambiando sustancialmente el punto de vista original.

Desde un punto de vista de un posible atacante se trata de identificar:

- Los puntos de entrada de la aplicación.
- Los activos a proteger.
- Los diferentes niveles de confianza.

Se establece cual es el nivel de seguridad del sistema:

- Mediante casos de uso.
- Conociendo las distintas dependencias.
- Utilizando modelos del sistema.

Se determina cuales son las amenazas:

- Se identifican las amenazas.
- Se analizan y clasifican.
- Se identifican las vulnerabilidades a las que se ve expuesto el sistema.

Pasos del modelado de amenazas según Microsoft

Según la metodología propuesta por Microsoft, los cinco pasos del proceso de modelado de amenazas son:

1. Identificar los objetivos de seguridad: Determinar cuales son los objetivos ayudará a cuantificar el esfuerzo se debe dedicar a los siguientes pasos.

2. Crear una descripción general de la aplicación: Identificar los actores involucrados y las características más importantes de la aplicación facilitará la identificación de las amenazas más importantes.

3. Descomponer la aplicación: Una vez que se conoce la arquitectura, es preciso identificar las funcionalidades y los módulos susceptibles de provocar un mayor impacto en la seguridad.

4. Identificar amenazas: Con la información recopilada, y en función del contexto y el escenario de la aplicación, se procede a la identificación de las amenazas más importantes.

5. Identificar vulnerabilidades: Revisar las diferentes capas de la aplicación para identificar los puntos débiles.