Investigación de software de detección de vulnerabilidades

Brandon Meza

Desde 1990 ha habido un incremento increíble de la tecnología y software creado, así como la cantidad de crímenes de la red como suplantación de identidad, falsificar información, divulgar información ilegal, etc
Todas las vulnerabilidades de un sistema son una amenaza para la información.
Hay muchas organizaciones internacionales dedicadas a investigar vulnerabilidades. La tecnología de análisis de vulnerabilidades necesita tener varias tecnicas de detección para hacer común su uso-

Principio de vulnerabilidad

- **Diseño:** la negligencia puede dejar algunos defectos en el flujo lógico del software, esto hace más fácil a los atacantes crear recursos no autorizados. Esto es una vulnerabilidad en el diseño de software
- **Codificación y testeo:** Ocurre una vulnerabilidad cuando los programadores solo integran código a un proyecto sin entender completamente su funcionamiento
- **Entorno Operacional:** Normalmente el software y hardware son diferentes por varias razones y el testeo antes de sacar un software se pasa por alto en ocasiones. Esto puede vulnerar la seguridad de un sistema
- **Parches del sistema:** después de lanzar el sistema pueden aparecer vulnerabilidades por lo que será necesario eliminarlas a través de parches del sistema

Clasificación de la vulnerabilidad

Buffer overflow

- Cuando el intruso ingresa una string que excede el tamaño especificado en el sistema, el excedente es el código que el intruso quiere ejecutar

Formatting String Vulnerabilities

- Se debe a los malos hábitos del programador. El formateo de strings es útil para el atacante ya que puede inyectar su propio contenido.

Competitive Conditions

- Los sistemas pueden mejorar la eficiencia del CPU y reducir las peticiones de tiempo de operaciones I/O. Esto provoca competición de recursos y se puede causar problemas de sincronización

Random Number Vulnerability

- Un número random en realidad es pseudo-random, cuando el atacante encuentra la forma en que se genera puede haber consecuencias que no son predicibles

Análisis de tecnologías de detección de vulnerabilidades

Detección Éstatica

No requiere de un programa corriendo y puede analizar vulnerabilidades de un programa sin una entrada, puede ser hecha en el proceso de programación

Esta detección es muy eficiente y rápida, además que el testeo de código se cubre muy bien

Existen 3 métodos de esta detección

- **Inferencia de tipo:** asegura que cada operación es apuntada por datos adecuados
- **Análisis de flujo de datos:** se usa para resolver la optimización del compilador, validación del programa, debugeo y testeo-
- **Análisis de restricciones:** se divide el proceso de análsiis de programa en generación de restricciones y resolución de restricciones.

Detección dinámica

- Necesita que el objetivo este compilado en código ejecutable
- El principio es tener un estatus y datos a través de una interfaz de monitoreo sin tener que cambiar el código fuente
- Los métodos más comunes de esta detección son el environmental error injection y data flow andalysis

Detección Hibrida

- En la práctica las detecciones estática y dinámicas son combinadas
- Esta detección se usa para evadir las debilidades de la detección estática y la dinámica.
- Es la tercera técnica de detección de vulnerabilidades de software más usada