MÉTODO NOVEL PARA LA DETECCIÓN DE VULNERABILIDADES DE SOFTWARE BASADO EN UNA TÉCNICA DE FUZZING

BRANDON MEZA

Técnicas tradicionales de detección de vulnerabilidades

Técnica fuzzing

Este método consiste en proveer una larga cantidad de datos semi válidos a un software como entrada, mientras se procesan estos datos el sistema puede comportarse de manera errónea, esto resulta en una exitosa detección de vulnerabilidades

Auditoria de código fuente

Se escaneo el código fuente para detectar vulnerabilidades potenciales en el sistema.

Auditoria de codigo ensamblador basado en IDA

Usando herramientas de desamblado escanean los archivos ejecutables

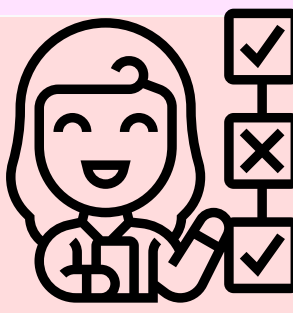
Comparación de parche binario

Se comparan los ejecutables antes y después de una actualización del sistema

Análisis inverso

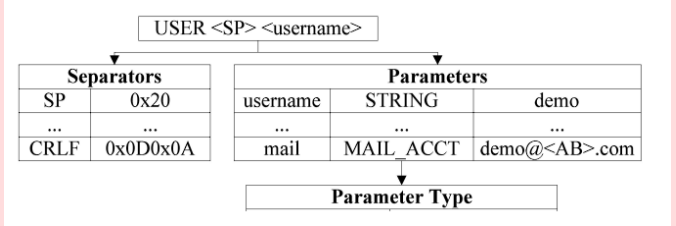
Se encuentran vulnerabilidades por medio del analisis inverso de herramientas de seguridad

Aplicación de fuzzing en el nivel de protocolo



Descripción de protocolos

Todos los nombres de comandos, parámetros y separadores del protocolo FTP son caracteres específicos o strings. Los parámetros de FTP pueden ser separados y cada parámetro del protocolo puede ser descrito por tablas

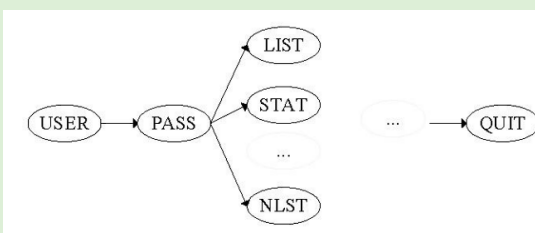


Generación de paquetes mal formados

Estos paquetes malformados son aquellos que cumplen con la gramática del protocolo pero son datos inválidos para el programa objetivo. La aplicación propuesta solo detecta vulnerabilidades de buffer overflow

Control del proceso fuzzing

El orden de los comandos enviados es importante, la detección de vulnerabilidades se activa cuando ciertos paquetes son mandados en orden. En la imagen de la derecha podemos ver este orden, cada secuencia de testeo corresponde a un contexto especifico.



Captura y análisis de las excepciones

El sistema hace uso de una API de debugeo suministrada por MS Winwdos OS para monitorear el proceso del servidor durante el proceso de testeo. Cuando una excepción ocurre el sistema suspende el envió de paquetes de prueba y guarda la información relacionada con la excepción. Posteriormente se analiza de manera automatiza la excepción y da el reporte generado de manera inteligente

Software que el sistema no puede procesar

Algunos sistemas incluyen un sistema anti-debugeo, esto hace que el sistema propuesto no pueda analizar las excepciones generados por estos programas