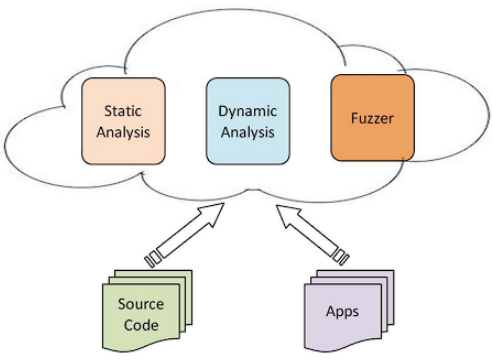
VULVLOUD: SCALABLE AND HYBRID VULNERABILITY DETECTION IN CLOUD COMPUTING

Presentado por: Brandon Meza

Muchos detectores de vulnerabilidades estáticos arrojan muchos falsos positivos, al igual que detectores dinámicos y los casos de prueba son difíciles de crear.
Vulcloud es un hibrido entre lo estatico, dinamico y fuzz testing en la nube



1 Vista general de vulcloud

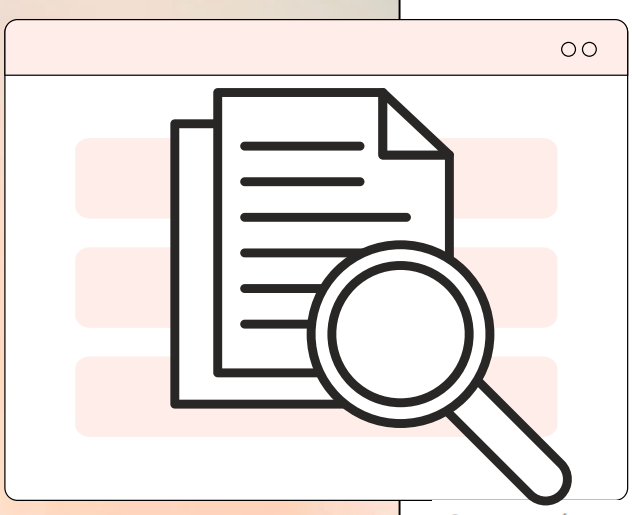
Vulcoud primer analiza estáticamente los objetos y reporta items potencialmente vulnerables, después crea casos fuzzing para los objetos identificados y los testea bajo el monitoreado dinámico en tiempo real. Al último analiza el código fuente de los resultados para verificar si son vulnerabilidades o no

Nodo Estatico 2

El análisis estático es para detectar vulnerabilidades en el software mediante la evaluación del código fuente, se detectan vulnerabilidades como buffer overflow, cross.site scripting y SQL Injection.

$$static(obj) = \Sigma match(obj) \cup \Sigma compile(obj).$$

3 Nodo Fuzzing



En el analsisi estatico se producen muchos falsos positvos donde el fuzz testing ayuda a refinar los resultados, este testeo provee numerosos casos de prueba usando datos invalidos o inesperados para detectar las vulnerabilidades

$$fuzzer(meta) = random(meta) \cup heu(meta) \cup gen(spec),$$

Nodo dinámico 4

Este nodo detecta vulnerabilidades observando el comportamiento de la ejecución de softwares. En Vulcloud, se usa para monitorear en tiempo real la ejecución y cachar las excepciones y crasheos usando diferentes herramientas. El análisis dinámico obtiene muy pocos falsos positivos y los casos fuzzing terminan con esos pocos falsos positivos