La nueva vulnerabilidad de contrabando de parámetros «ParseThru» afecta a las aplicaciones basadas en Golang

Autor: I. Stepanenko

Fecha: Thursday 11th of August 2022 08:22:59 PM

Investigadores de seguridad cibernética descubrieron una nueva vulnerabilidad llamada ParseThru que afecta a las aplicaciones basadas en Golang y que podría abusarse para obtener acceso no autorizado a las aplicaciones basadas en la nube.

> «La vulnerabilidad recién descubierta permite que un actor de amenazas eluda las validaciones bajo ciertas condiciones, como resultado del uso de metadatos de análisis de URL inseguros integrados en el lenguaje», dijo la compañía israelí Oxeye.

El problema tiene que ver con las inconsistencias derivadas de los cambios introducidos en la lógica de análisis de URL de Golang que se implementa en la biblioteca *«net/url»*.

Aunque las versiones del lenguaje de programación anteriores a la 1.17 trataban los puntos y comas como un delimitador de consulta válido (por ejemplo: example.com?a=1;b=2&c=3), este comportamiento se ha modificado desde entonces para generar un error al encontrar una cadena de consulta que contiene un punto y coma.

«Los paquetes net/url y net/http solían aceptar ';' como separador de configuración en las consultas de URL, además de '&'», según las notas de la versión 1.17 publicadas en agosto pasado.

> «Ahora, las configuraciones con punto y coma sin codificación porcentual se rechazan y los servidores net/http registrarán una advertencia en 'Server.ErrorLog' cuando encuentren uno en una URL de solicitud».

El problema surge cuando una API pública basada en Golang con la versión 1.17 o posterior se comunica con un servicio de back-end que ejecuta una versión anterior, lo que lleva a un escenario en el que el actor malicioso podría pasar de contrabando solicitudes que incorporan parámetros de consulta, que de lo contrario, serían rechazados.

En otras palabras, la idea es enviar solicitudes especialmente diseñadas que contengan un punto y coma en la cadena de consulta, que la API de Golang orientada al usuario ignora, pero procesa el servicio interno.

Esto, a su vez, es posible gracias al hecho de que uno de los métodos responsables de obtener la cadena de consulta analizada descarta silenciosamente el mensaje de error devuelto.

Oxeye dijo que identificó varias instancias de ParseThru en proyectos de código abierto como Harbor, Traefik y Skipper, lo que hizo posible eludir las validaciones implementadas y llevar a cabo acciones no autorizadas. Los problemas se abordaron luego de la divulgación responsable a los respectivos proveedores.

Esta no es la primera vez que el análisis de URL plantea un problema de seguridad. A inicios de enero, Claroty y Snyk revelaron hasta ocho vulnerabilidades en bibliotecas de terceros escritas en lenguaje C, JavaScript, PHP, Python y Ruby, que se originaron como resultado de una confusión en el análisis de URL.