Revisión Papers 1

Ricardo Conejeros Maldonado

Permission Rationales in the Web Ecosystem- An Exploration of Rationale Text and Design Patterns

¿Cuál es el problema que se está resolviendo?

El articulo aborda la falta de conocimiento sobre como algunos sitios web justifican la solicitud de permisos de acceso a cámara, notificaciones o geolocalización y como estas influyen a la toma de decisiones de los usuarios.

¿Cuál es la nueva idea que están proponiendo?

Se centran en tres aspectos claves de las justificaciones de permisos de los sitios web:

1. Detección y prevalencia: Elaboran un modelo semiautomatizado que evalúa mas 770 mil páginas utilizando web crawling, LLMs y clasificación con BERT

2. Patrones de diseño: Analizan atributos textuales y de interfaz en 3.600 justificaciones y 749 diseños de UI.

3. Impacto en el usuario: Determinan que el 41% de las solicitudes terminan con una aceptación dependiendo de ciertos atributos, esto gracias al uso de la telemetría presente en Chrome.

Puntos positivos

1. La combinación de crawling interactivo, filtrado con LLMs (Mistral-7B) y clasificación con BERT permitieron analizar un gran volumen de datos (6M de textos únicos).

2. La identificación de firmas de código para detectar bibliotecas de terceros como OneSignal, dan profundidad al análisis.

Puntos negativos

1. El crawler solo detecto el 20% de los prompts, el estudio se centró en inglés, dejando fuera patrones culturales/lingüísticos.

2. Los datos obtenidos de Chrome son del año 2022 y podrían no reflejar comportamientos actuales.

Trabajo futuro

1. Integrar agentes basados en LLMs como YURASCANNER para simular flujos de usuarios más realistas.
2. Elaborar un plugin para frameworks como React o Angular que sugiera justificaciones efectivas, junto con métricas de usabilidad.

Calificación 1-5

La calificación otorgada es de 4, al no abordar completamente las limitaciones del crawling y la diversidad lingüística.

Puntos de discusión

La brecha web/móvil, las justificaciones en Android son bloqueantes, mientras en la web son no modales

Pinning Is Futile

¿Cuál es el problema que se está resolviendo?

El documento investiga la efectividad de la estrategia de pinning (fijar versiones especificas en dependencias) en el ecosistema npm para defenderse de los ataques a la cadena de suministros. Los autores demuestran que esto aumenta los costos de mantenimiento y los riesgos de exposición a actualizaciones maliciosas. Además, proponen una estrategia de pinning colectiva que mejoraría la seguridad del ecosistema.

¿Cuál es la nueva idea que están proponiendo?

Los autores demuestran que el pinning local es ineficaz e incluso contraproducente en proyectos con grafos de dependencias complejas mayor a 500 nodos.

1. Pinning coordinado en paquetes críticos (top 100), reduciría el riesgo de propagación de ataques en un 30%.

2. Modificar paquetes npm para que hereden versiones fijas de sus dependencias.

Puntos positivos

1. Robustez en sus conclusiones al utilizar un gran número de simulaciones (10 mil proyectos npm y GitHub).

2. No solo critican el pinning, sino proponen alternativas como el pinning colectivo.

Puntos negativos

Se asume que los mantenedores de paquetes críticos realizaran auditorias perfectas lo cual es difícil de garantizar.

Trabajo futuro

1. Investigar como el pinning transitivo afecta a los pipelines de integración continua.

2. Crear un plugin para npn/VSCode dado un proyecto, pueda identificar si el pinning local es seguro o no dependiendo del tamaño del grafo de dependencias y sugiera paquetes candidatos para pinning colectivo.

Calificación 1-5

La calificación otorgada es de 4.5, Podría explorar más la viabilidad de implementar el pinning transitivo en npm.

Puntos de discusión

¿El pining transitivo es una alternativa factible para adoptar en npm?

¿Estarían los mantenedores npm y la comunidad dispuestos a intentarlo?