netidee

PROJEKTE

Opaque

Zwischenbericht | Call 17 | Projekt ID 6374

Inhalt

1 Einleitung

Das Projekt läuft trotz seiner technischen Komplexität sehr gut. Bei der Entwicklung der Software Pakete sind wir bereits in einem Stadium, bei dem wir uns auf Optimierungen fokussieren können. Eine erste Version der Dokumentation existiert, aber es fehlen noch Inhalte. Auch hier liegen wir gut in dem Zeitplan.

2 Status der Arbeitspakete

2.1 Arbeitspaket 1 - *Detailplanung und Formales am Projektstart*

Das Projekt hat etwas später gestartet als im Antrag ursprünglich geplant, aber sobald begonnen wurde, konnte die Detailprojektion, Liste der Projektergebnisse, Projektwebseite, erster Blogeintrag und Ansuchen zur ersten Förderrate recht flott umgesetzt werden.

2.2 Arbeitspaket 2 – *Beta (Web Client & Node)*

Zu Beginn wurde das Rust Projekt inkl. Web Assembly Build aufgesetzt und mit der Implementierung des Registration und Login Vorgänge begonnen. Diese Arbeiten erfolgten um einiges schneller als erwartet.

Daraufhin wurde ein starker Fokus auf eine automatisierte Test-Suite gesetzt. Es gibt sowohl Unit als auch end-to-end Tests. Des Weiteren wurden mehrere Beispielprojekte angelegt, um die unterschiedlichen Umgebungen (verschiedene Web-Bundler als auch Node) zu testen. Dies war besonders wichtig, da wir hier auf unerwartete Probleme mit der Einbindung (wegen WebAssembly) des Software-Pakets gestoßen sind. Diese Probleme konnten in AP3 gut gelöst werden.

2.3 Arbeitspaket 3 – *Production ready (Web Client & Node)*

Dieses Arbeitspaket wurde großteils schon umgesetzt und es gibt eine produktionsreif Version des opaque Paketes in der Variation ristretto255 und P256. Wir haben beide angeboten, da beide in der Spezifikation des Protokolls angegeben sind und Kevin Lewi von Meta empfohlen hat auch beide anzubieten.

Wie oben erwähnt, hat der Einsatz von WebAssembly zu verschiedenen Problemen in unterschiedlichen Umgebungen geführt. Wir konnten das Problem lösen, indem wir das Laden von WebAssembly code in das Software Paket eingebettet haben. Hierbei wird der Binary Code als base64 encoded und beim initialisieren des Moduls geladen.

Als Test haben wir das opaque Softwarepaket auch in eine Applikation eingebunden. Dies war wichtig, da uns dadurch ein paar unverständliche API Entscheidungen aufgefallen sind, welche wir mit einem API Redesign ausmerzen konnten.

Des Weiteren haben wir Tests durchgeführt, welcher Rust-WebAssembly Memory-Allocator sinnvoll ist und die Ergebnisse in der Dokumentation dokumentiert.

In den letzten Wochen haben wir auch mit Cure53 und NCC Group für ein Security Audit auch Kontakt aufgenommen. Unser Budget dafür beträgt etwa 3.000-4.000 EUR. Das Angebot von Cure53 mit zirka 21.000 EUR ist leider deutlich darüber. Cure53 hat uns empfohlen mit https://www.opentech.fund/labs/red-team-lab/ Kontakt aufzunehmen, um sie als Sponsor für ein Audit zu gewinnen.
Wir hoffen, von der NCC Group ein günstigeres Angebot zu bekommen. Alternativ würden wir noch ROS kontaktieren.

2.4 Arbeitspaket 4 – *React Native Integration*

Ursprünglich war geplant, dass mit diesem Arbeitspaket erst Mitte Juni begonnen werden sollte. Der Hauptentwickler Stefan Oestreicher hat jedoch nach den frühen Erfolgen in Arbeitspaket 2 vorgeschlagen, dass wir den Start hier deutlich vorziehen, da er mit React Native nicht vertraut war. Dies hat sich als richtige und wichtige Entscheidung herausgestellt, da der Aufwand hier unterschätzt wurde.

Mittlerweile haben wir eine produktionsreife Version des react-native-opaque Paketes in der Variation ristretto255. Die Generierung der Variante P256 ist noch ausständig und wird im nächsten Monat umgesetzt.

Weiters gibt es auch eine Test-Suite die sowohl auf iOS, Android und der react-native-web Version läuft.

2.5 Arbeitspaket 5 – *Basic Documentation*

In diesem Arbeitspaket wurde die grundlegende Dokumentations-Website erstellt. Dies beinhaltet das automatisierte veröffentlichen nach jeder Änderung, die Struktur der Website, die Suche und das Design.

Teil dieses Arbeitspaketes war es auch das Logo und Branding für das Projekt zu designen. Bettina Ecker hat diese Arbeiten bereits abgeschlossen.

Ebenfalls wurden die ersten Inhalte und eine grobe Struktur angelegt.

2.6 Arbeitspaket 6 – *Polished Documentation*

Das Getting Started wurde fertig gestellt. Diverse weitere Guides sind in Arbeit. Besonders wichtig wird die Landing Page. Hier geht es darum das Projekt visuell ansprechend und einfach dem WebsitebesucherInnen nahe zu bringen.

Wir erwägen auch einen kleinen Videokurs zu produzieren. Ob dies sinnvoll ist, werden wir testen und je nachdem dann umsetzen.

2.7 Arbeitspaket 7 – *Dokumentation und Formales am Projektende*

Die erforderlichen Aufzeichnungen wie z.B. Zeitaufzeichnungen werden durchgehend gemacht und alle 2 Monate haben wir einen Blog-Post veröffentlicht.

3 Umsetzung Förderauflagen

Für dieses Projekt wurden keine besonderen Förderauflagen bestehen.

4 Zusammenfassung Planaktualisierung

Der Link zur eigenen Projekt-Dokumentation ist vorläufig unter https://opaque-documentation.netlify.app/ und nicht wie ursprünglich geplant unter https://www.serenity.re/en/opaque zu erreichen.

Die Stundendokumentation wurde eingetragen. Allerdings sind die Juli Stunden nur von Nikolaus Graf und nicht von Stefan Oestreicher und Susanne Kristufek eingetragen. Wir erledigen dies immer mit Abschluss des Monats.

Der Status der Arbeitspakete und die tatsächlichen Stunden wurden eingetragen.

5 Öffentlichkeitsarbeit/ Vernetzung

Vor Beginn des Projektes haben wir mit Kevin Lewi, einer der Autoren der Opaque Protokoll Spezifikation und auch Hauptverantwortlicher für opaque-ke bei Meta, Kontakt aufgenommen. Herr Lewi hat uns während der Entwicklung immer wieder Feedback gegeben und Unklarheiten in der Spezifikation aufgelöst. Er heißt unser Projekt gut, es wird mittlerweile auch schon direkt im Readme von opaque-ke verlinkt (https://github.com/facebook/opaque-ke#resources).

Nächste Woche findet das Security Meetup in Wien statt. Wir werden uns für einen Vortrags-Slot im Oktober oder November bewerben. Ebenso wollen wir Opaque beim Rust, ReactVienna und ViennaJS Meetup vortragen.
Des Weiteren soll ein kurzer Videokurs erstellt werden. Diese kann dann auf diversen Plattformen wie Youtube oder Egghead veröffentlicht werden.
Für die nächste Konferenz-Saison im Frühling werden wir uns bei Konferenzen wie JSNation bewerben.

6 Eigene Projektwebsite

Die aktuelle Adresse der Projektwebsite ist https://opaque-documentation.netlify.app/. Wir erwägen eine eigene Domain zu kaufen.