netidee

PROJEKTE

open-pdf-sign

Zwischenbericht | Call 16 | Projekt ID 5822

Inhalt

1 Einleitung 3

2 Status der Arbeitspakete 4

2.1 Arbeitspaket 1 - Projektstart 4

2.2 Arbeitspaket 2 - Research 4

2.3 Arbeitspaket 3 - Implementierung 5

2.4 Arbeitspaket 4 - Dokumentation & Marketing 6

2.5 Arbeitspaket 5 - Projektabschluss 7

3 Umsetzung Förderauflagen 7

4 Zusammenfassung Planaktualisierung 7

5 Öffentlichkeitsarbeit/ Vernetzung 8

6 Eigene Projektwebsite 8

1 Einleitung

Unser Projekt „open-pdf-sign“ ist eine Open Source-Bibliothek und Lösung zum sicheren Signieren von PDFs. Als Kommandozeilenapplikation soll open-pdf-sign es Entwicklern ermöglichen, PDF-Signatur in eigene Applikationen und Webanwendungen zu integrieren.

Genauso wie „Let’s encrypt“ die Eintrittsschwelle für TLS-Verschlüsselung gesenkt hat, möchten wir mit open-pdf-sign dasselbe für PDF-Signatur ermöglichen.

Dieser Zwischenbericht gibt eine Übersicht über den aktuellen Stand der Implementierung des Kommandozeilentools, des Installationsassistenten und der zugehörigen Dokumentation.

2 Status der Arbeitspakete

2.1 Arbeitspaket 1 - Projektstart

Inhalt:
**) Vertragsprüfung, unterschreiben und an netidee senden*
**) Projektplanung auf Basis der Excel-Vorlage und an berichte@netidee.at senden*
**) Projektwebsite Inhalt überprüft /erster Blog erstellt; Projektteam ergänzt (optional)*
**) Förderabrufformular für Förderrate 1 an berichte@netidee.at*

Dieses Paket wurde abgeschlossen

2.2 Arbeitspaket 2 - Research

Da es sich bei PDF-Signatur um ein komplexes kryptographisches Verfahren handelt, stand Research zu den Algorithmen und Standards am Beginn des Projekts. Entscheidungen hierzu sind im späteren Projektverlauf nur schwer abzuändern und sind deshalb unter großer Bedacht zu treffen.

Wir haben uns mit den verschiedenen Standards für PDF-Signatur (PAdES-B/T/LT/LTA) und verschiedenen Bibliotheken beschäftigt. Viel Zeit wurde dann dafür verwendet, eine passende Bibliothek im passenden Standard auszuwählen. Gleiches auch bei der Software zur Anbringung der sichtbaren Signatur, die wir schließlich in PDFbox gefunden haben.

Die Probleme dabei haben wir im Blog-Artikel https://www.netidee.at/open-pdf-sign/standards-standards-standards-und-ein-prototyp beschrieben.

Ähnlich komplex wie die Signaturstandards sind auch die Standards für Schlüsselaufbewahrung. Hier haben wir Zeit dafür aufgewendet, zu recherchieren und zu testen, welche Formate in der Praxis verwendet werden (PKCS7/PKCS12/etc)

Im Bereich der Lösung für die automatische Konfiguration von Webservern ist es uns gelungen, einen Weg zu finden, eine solche möglichst einfach umzusetzen. Wir haben uns hier sehr stark an "*certbot*" von LetsEncrypt orientiert und auch bereits mit der Implementierung begonnen.

Besondere Erfolge/ Probleme

Komplexität der verschiedenen Signaturarten und Formate für Keys.

Gab es große Abweichungen zum Plan? Warum?

Der Research hat mehr Zeit in Anspruch genommen als geplant.

2.3 Arbeitspaket 3 - Implementierung

Inhalt:
**) Implementierung der technischen Lösung*
**) NPM Modul*

Erkenntnisse zur Vorgangsweise und Softwarekomponenten

Basierend auf den Erkenntnissen des AP 2 haben wir beschlossen, die Implementierung der sichtbaren Signatur mit Hilfe von PDFbox umzusetzen. Die Signatur und kryptographischen Funktionen haben wir mit der von der EU veröffentlichten Open Source-Bibliothek“DSS” umgesetzt, die uns die Implementierung verschiedener Zertifikate erleichtert. Für die Implementierung der verschiedenen Schlüsselformate verwenden wir die verbreitete Open Source-Bibliothek “bouncy castle”

Kurzbeschreibung der erreichten Ergebnisse

Prinzipiell haben wir beschlossen unsere Lösung in zwei Teile zu splitten. Zum einen haben wir die Kommandozeilenapplikation, die für sich alleine stehen können soll. Diese nimmt als Eingabewert das zu signierende PDF sowie den Schlüssel für die Signatur entgegen und liefert ein signiertes PDF zurückgibt. Dieser Teil ist bereits weit fortgeschritten und kann auch schon experimentell in Produktionsumgebungen eingesetzt werden. Es fehlen aber noch eine bessere Dokumentation und einige Zusatzfunktionen, um die Einsatzfähigkeit vielfältiger gestalten zu können

Der Zweite Teil besteht aus einem über Kommandozeile verfügbarem Installationswerkzeug, das es für Nutzer:innen so einfach wie möglich machen soll, alle PDFs auf einem Webserver zu signieren und eingesetzte Serversoftware automatisch korrekt zu konfigurieren. Hier haben wir ein proof-of-concept erstellt, allerdings ist die Umsetzung in einzelnen Tool noch ausständig, weil wir diese Lösung auf so vielen Plattformen wie möglich laufen lassen wollen und uns hier noch in der Research-Phase befinden, wie wir die Softwarearchitektur idealerweise gestalten können.

Besondere Erfolge/ Probleme

Die Vielzahl an verschiedenen Signaturformaten war nicht nur herausfordernd zu verstehen, sondern stellten auch eine Herausforderung in der Implementierung dar.

Hinsichtlich des Installationswerkzeugs konnten wir Feedback des „certbot“-Team einholen.

Gab es große Abweichungen zum Plan? Warum?

Wir wären gerne schon weiter fortgeschritten mit der Implementierung des Kommandozeilen-Installationswerkzeugs um die automatische PDF-Signatur bei gängigen Webservern zu implementieren. Hier entstanden Verzögerungen aufgrund von beruflichen Tätigkeiten neben netidee und dadurch entstehenden Zeitmangels.

2.4 Arbeitspaket 4 - Dokumentation & Marketing

Inhalt:
**) Dokumentation*
**) Website*
**) Vermarktung*

Erkenntnisse zur Vorgangsweise:

Um diesen Punkt zu starten, muss zunächst ganz klar sein wie unsere Lösungen aussehen. Dies ist nun klar und wir können mit diesem Arbeitspaket beginnen

Kurzbeschreibung der erwarteten Ergebnisse

**) Definition der Lösungen:*

*****) Open-pdf-sign sdk***

A self contained jar that can be run as CLI

Input: pdf + certificates

Output: signed pdf

*****) Open-pdf-sign installer***

Should support server:

Nginx, (Apache), (Wordpress)

Distributions:

Ubuntu latest, (Debian)

Two modes:

On the fly signing, (One time signing and retrieval)

Necessities:

A domain that already has a letsencrypt certificate, Nginx / Apache Server, Ubuntu latest version

Besondere Erfolge/ Probleme

Nein, noch ausständig.

Gab es große Abweichungen zum Plan? Warum?

Derzeit im Zeitplan.

2.5 Arbeitspaket 5 - Projektabschluss

Inhalt:
) Kurzbeschreibung der Haupttätigkeiten
) Erkenntnisse zur Vorgangsweise
) Kurzbeschreibung der erreichten Ergebnisse
) Besondere Erfolge/ Probleme
) Gab es große Abweichungen zum Plan? Warum?

Noch nicht relevant.

3 Umsetzung Förderauflagen

Anm.: Dieses Kapitel ist nur relevant, wenn in der Fördervereinbarung spezielle Förderauflagen festgelegt wurden. In diesem Fall soll in diesem Kapitel dargestellt werden, wie diese berücksichtigt werden.

N/A

4 Zusammenfassung Planaktualisierung

Anm.: Alle Anpassungen des Plan-Excels kurz zusammengefasst

Es wurden einige Stunden von der Implementierung in Marketing verschoben (20)

Es wurde der Netzplan aktualisiert und um 2 Monate verlängert

5 Öffentlichkeitsarbeit/ Vernetzung

Anm.: Beschreibung der bereits erfolgten Öffentlichkeitsarbeit oder Vernetzung, bzw. Beschreibung des Plans künftiger Aktivitäten

Wir waren sehr fleißig beim Blogartikel erstellen. Aktuell sind wir dabei, die Website zu erstellen.

Wir haben auf einigen Treffen und Meetups den anwesenden Leuten von unserer Lösung erzählt und gutes Feedback erhalten.

6 Eigene Projektwebsite

Anm.: Wird zusätzlich zur netidee-Projektwebsite noch eine eigene Website betrieben, so ist hier die Adresse anzugeben.

https://openpdfsign.org/ - derzeit noch in Erstellung, verlinkt auf den bereits veröffentlichten Source Code des Prototypen.