Richtlinie zum „Software-Sprint“

SecureDrop – Schutz für Journalisten und Quellen

Schlussbericht

Zuwendungsempfänger:

Eric Hartsuyker

Das diesem Bericht zugrundeliegende Vorhaben wurde mit Mitteln des Bundesministeriums für Bildung und Forschung unter dem Förderkennzeichen 01IS18S32 gefördert. Die Verantwortung für den Inhalt dieser Veröffentlichung liegt beim Autor.

Kurze Darstellung der Aufgabenstellung und Motivation

Dieses Projekt wurde motiviert, SecureDrop für Quellen, Journalisten und Administratoren benutzerfreundlicher zu machen. Die Sicherheit ist kompliziert und oft sind die Unsicherheiten auf einen Benutzerfehler zurückzuführen. Wenn Software einfacher gemacht wird, gibt es weniger Fehler. SecureDrop zielt darauf ab, die Quellen angesichts des von Massenüberwachung ausgehenden Risikos so gut wie möglich zu schützen. Ein neuer Journalisten-Workflow und die Aktualisierung von altem Code, Software-Packages und Betriebssystemen.

Beitrag des Projektes zu den Zielen der Förderinitative „Software-Sprint“

Die Hauptzielgruppe für dieses Projekt sind Whistleblower und die Journalisten, die mit ihnen kommunizieren. Journalisten profitieren von zusätzlichen Quellen, die vertrauliche Informationen senden, und sie fühlen sich durch ihre Anonymität dazu befugt. Quellen profitieren von einem geringeren Risiko der staatlichen Verfolgung, da sie bei der Kommunikation mit Quellen identifiziert werden. Infolgedessen profitiert die gesamte Gesellschaft davon, Machtmissbrauch ausgesetzt zu sein.

Ausführliche Darstellung der Ergebnisse

Wir konnten das Betriebssystem, auf dem SecureDrop ausgeführt wird, von Ubuntu 14.04 auf Ubuntu 16.04 aktualisieren. Wir haben einen Prototyp von einem Qubes basierenden Client mit Qt erstellt. Die Client-Funktionen erforderten Ergänzungen und Änderungen an der API. Der Client verfügt über die meisten wichtigen Funktionen, um von Journalisten verwendet zu werden. Einige der vorgeschlagenen Funktionen konnten nicht implementiert werden, da Daten aus

Benutzerforschungsergebnissen mit unseren ursprünglichen Annahmen in Konflikt standen. Durch die Zusammenarbeit mit einem UX-Forscher konnten wir jedoch erhebliche Änderungen an der UX vornehmen, anstatt die von uns vorgeschlagenen großen Änderungen.

Das Größte, was ich gelernt habe, waren die Grundlagen für die Erstellung eines nativen GUI-Clients mit Qt. Dies wird in Zukunft sehr nützlich sein, um Anwendungen zu erstellen, die benutzerfreundlicher sind als Befehlszeilentools.

Zielgruppe, Nutzen und mögliche Weiterentwicklungen

Quellen profitieren am meisten von Verbesserungen an SecureDrop, die bei Verwendung der Anwendung dem größten Risiko ausgesetzt sind. Ein Teil dieser Sicherheit ist für den Benutzer sichtbar, aber ein Großteil davon befindet sich hinter den Kulissen.

SecureDrop wurde in einer Reihe von Vorträgen und Forschungsarbeiten als hervorragendes Beispiel für sichere Open-Source-Software zitiert, die ein sehr spezifisches Ziel für die Benutzer hat. Andere aus der Community profitieren davon, SecureDrop als Vorlage für die vollständige Bereitstellung einer sicheren Anwendung verwenden zu können. Die Entwicklung von SecureDrop ist für die gesamte Open Source-Community geeignet.

Kurze Darstellung der Arbeiten, die zu keiner Lösung geführt haben

Ein Meilenstein, den wir ursprünglich vorgeschlagen hatten, war die Möglichkeit, einem Journalisten eine Quelle zuzuweisen. Dies war eine Idee, die wir uns ausgedacht und prototypisiert haben. Wir testeten und recherchierten mit Benutzern und stellten fest, dass diese Lösung bei den meisten Arbeitsabläufen nicht funktionierte und nur von einer Minderheit von Journalisten verwendet werden konnte. Dies zeigt, wie wichtig das Testen von Benutzern ist.

Kurze Angabe von Präsentationsmöglichkeiten für mögliche Nutzer

SecureDrop kann auf verschiedene Arten kontaktiert werden:

Website – https://securedrop.org
Chat – https://gitter.im/freedomofpress/securedrop

Kurze Erläuterung zur Einhaltung der Arbeits- und Kostenplanung

Es wurden einige erste Meilensteine festgelegt, die für 6 Monate Arbeit für 1 Person sinnvoll waren. Wir wussten jedoch, dass diese möglicherweise nicht durch zeitliche Einschränkungen oder Planänderungen erreicht werden. SecureDrop hat viele ungelöste Tickets, und wenn mehr Zeit übrig war, wurden andere Tickets zugewiesen.

Kurze Darstellung von etwaigen Ergebnissen bei anderen Stellen

SecureDrop existiert als Teil eines großen Ökosystems verwandter Technologien zum Schutz der Privatsphäre. Ein Beispiel, bei dem ein anderes Projekt unser Projekt durchschnitten hat, war ein Fehler in Firefox. Dieser Fehler verursachte Probleme mit dem Verhalten des Addons NoScript. Der

Tor Browser basiert auf Firefox und wird mit NoScript geliefert. Wir mussten mit den Upstream-Projekten zusammenarbeiten, um dieses Problem zu identifizieren und zu umgehen.