IP6: Blockchain Transactionmanager

Projektvereinbarung

Faustina Bruno & Jurij Maïkoff

2019-10-01

Inhaltsverzeichnis

Betreuer:	Markus Knecht
	Daniel Kröni
Auftraggeber:	Markus Knecht
	Daniel Kröni
Projektende:	20.03.2020

1 Aufgabenstellung

Blockchains verfügen über verschiedene Mechanismen um sich gegen Attacken abzusichern. Eine davon ist eine Gebühr auf jeder Transaktion, die sogenannte Gas Fee. Dadruch können Denail of Service (DoS) Attacken, bei denen das Netzwerk mit unzähligen Transaktionen geflutet wird, effizient bekämpft werden. Dem Angreifer kann die Attacke nit aufrecht erhalten, da ihm die finanziellen Mittel ausgehen.

Obwohl dieser Schutzmechanismus auf einer öffentlichen Blockchain sehr effizient und elegant ist, eignet er sich nicht für eine Lernumgebung. Hier sollen Anwender die Möglichkeit haben, Transaktionen ohne anfallende Gebühren ausführen zu können. Dadurch wird die Blockchain anfällig für DoS Attacken.

Die Projektaufgabe besteht darin, eine Lösung zu finden, bei der die Sicherheit der Blockchain auch ohne eine Transaktionsgebühr gewährleistet werden kann.

1.1 Ziele

Das Ziel der Arbeit ist es zuerst eine Konzeptionelle Erarbeitung eines Testnetzwerkes welches:

- nicht permanent ist (Reboot möglich)
- kostenlose Transaktionen ermöglicht
- Anonymität gewährleistet
- Sicherheit gewährleistet

und in einem zweiten Schritt die Umsetzung/Realisierung dieses Netzwerkes.

Um diese Ziele zu erriechen sind folgende Fragestellungen von Bedeutung:

- Der Zustand in der Blockchain soll synchornisiert werden können ohne von einem Teilnehmer manipulliert zu werden

- wie kann die Gebühr einer Transaktion auf null gesetzt werden und gewährleistet werden, dass nicht zu viele Transaktionen (zB Attacken) getätigt werden
- Supportet Uport unsere gewünschten Anfroderungen einer SmartWallet oder müssen wir selber eine SmartWallet programmieren
- Wie kann man Attacken vermeiden (zB algorithmisch: nur eine Anzahl Transaktionen pro Monat möglich)

1.2 Risiken

Risiko	Auftreten	Impact	Risk level	Gegenmassnahme
Teammitglied bricht Projekt ab	1	3	3	Gute Kommunikation unter den Teammitgliedern
Unterschätzen des Projektumfanges	2	2	4	Sorgfältige Planung und regelmässig Rücksprache mit den Betreuern
Ausfall von einem Teammitglied (mehr als 2 Wochen)	2	2	4	Sofortiges Informieren von Betreuern. Planung überarbeiten und Ausfall berücksichtigen

2 Entwicklungsumgebung

In diesem Abschnitt wird die geplante Testumgebung und deren Verwendung beschrieben.

2.1 Betriebssystem

Beide Teammitglieder verwenden Windows 10 als Betriebssystem.

2.2 Blockchain

Um unser erworbenes Wissen auch testen zu können, wird eine Test-Blockchain aufgesetzt. Zu Beginn bietet die Testumgebung eine Möglichkeit, das Gelernte anzuwenden und so das Verständnis für das Thema zu vertiefen. Später im Projekt wird die Umgebung benötigt um ausgearbeitete Ansätze zu testen und analysieren.

Als Blockchain wird Ethereum verwendet.

2.3 Wallet

Für die Verwaltung von Identitäten und Transaktionen auf einer Blockchain werden sogenannte Wallets verwendet. Diese Verwaltung ist auch auf einer Lernumgebung nötig, daher muss geprüft werden, ob vorhandene Wallets, wie zum Beispiel uPort, unseren Ansprüchen genügen oder ob diese im Rahmen von diesem Projekt entwickelt werden müssen.

2.4 Smart Contracts

Für die Entwicklung von Smart Contracts wird die Sprache Solidity verwendet. Auch hier wird die Testumgebung genutzt, um Gelerntes anweden zu können. Sobald eigene Smart Contracts entwickelt werden, kann die Testumgebung genutzt werden, um diese zu testen[1].

3 Termine

Datum	Event
24.09.2019	Kickoff
28.11.2019	Zwischenpräsentation
20.03.2019	Abgabe Bachelorthesis
13. - 24.04.2019	Verteidigung

Quellenverzeichnis

[1] M. Lipovača, „Learn You a Haskell for Great Good!“, 2019. [Online]. Verfügbar unter: http://learnyouahaskell.com/.