IP6: Blockchain Transactionmanager

Projektvereinbarung

Faustina Bruno; Jurij Maïkoff

Studiengang:
- iCompetence
- Informatik

Betreuer:
- Markus Knecht
- Daniel Kröni

Auftraggeber:
Fachhochschule Nordwestschweiz
FHNW Campus Brugg-Windisch
Bahnhofstrasse 6
5210 Windisch

2019-10-01

Inhaltsverzeichnis

1 Aufgabenstellung

Blockchains verfügen über verschiedene Mechanismen, um sich gegen Attacken abzusichern. Eine davon ist eine Gebühr auf jeder Transaktion, der sogenannte Gas Price[1]. Dadurch können Denial of Service (DoS)[2] Attacken, bei denen das Netzwerk mit unzähligen Transaktionen geflutet wird, effizient bekämpft werden. Der Angreifer kann die Attacke nicht aufrechterhalten, da ihm die finanziellen Mittel zwangsläufig 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 jedoch 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.

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

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

In einem zweiten Schritt die Umsetzung/Realisierung dieses Netzwerkes.

Um diese Ziele zu erriechen sind folgende Fragestellungen von Bedeutung:

- wie kann die Gebühr für Transaktionen auf null gesetzt und die Sicherheit der Blockchain trotzdem gewährleistet werden.
- Gibt es eine Wallet die unsere gewünschten Anfroderungen unterstützt und nur erweitert werden muss, oder müssen wir selber eine SmartWallet programmieren.
- Wie kann man algorithmisch Attacken vermeiden
  - Anomaly detection
  - Beschränkung von Transaktionen pro Benutzer und Zeitintervall
  - etc

2 Planung

In diesem Kapitel wird beschrieben, wie das Projekt geplant wird. Dazu gehören Meilensteine und die Benennung der wichtigsten Teilaufgaben.

2.1 Meilensteine

In der Tabelle 2.1 sind die Meilensteine für dieses Projekt aufgeführt.

Tabelle 2.1: Meilensteine

Erledigt bis	Meilenstein	Beschreibung
24.09.2019	Kickoff	Besprechung der Rahmenbedingungen
08.10.2019	Testumgebung	Zur Einarbeitung in die Materie, testen und analysieren von Code
15.10.2019	Projektvereinbarung abgeschlossen	Vereinbarung über Rahmenbedingungen, Planung und Ziele des Projekts
22.10.2019	Analyse Phase	Einarbeitung in das Thema Blockchain und Analyse von möglichen Tools
05.11.2019	Gratis Transaktionen in der Blockchain	Eine Blockchain in der jeder gratis Transaktionen ausführen kann
05.11.2019	Wallets analysieren	Einarbeitung in das Thema und Analyse von möglichen Tools
19.11.2019	Erster Berichts Entwurf für Feedback	Eine frühe Version des Berichtes für die Betreuer, damit sich die Studierenden nach den Bedürfnissen der Betreuer richten können
28.11.2019	Zwischenpräsentation	Präsentation des aktuellen Standes für Experte und Betreuer

Erledigt bis	Meilenstein	Beschreibung
10.12.2019	Analyse Smart Contracts	Einarbeitung in das Thema und Analyse von möglichen Tools
14.01.2020	Erweiterung der Wallet	Gewählte Wallet für unsere Bedürfnisse erweitern
11.02.2020	Steuerung für gratis Transaktionen über Gruppen	Einschränkung von gratis Transaktionen auf Gruppen
18.02.2020	zweite Berichts Version	Bericht geht nochmals an die Betreuer für ein finales Feedback vor der Einreichung der Thesis
25.02.2020	Analyse von Algorithmen für Gruppenverwaltung	Analyse von Algorithmen um schadhaftes Verhalten in der Blockchain zu identifizieren / unterbinden
03.03.2020	Implementierung Algorithmen in Smart Contracts	Gewählter Algorithmus mit einem Smart Contract implementieren
19.03.2020	Testen und Überarbeiten von Blockchain	Testen und analysieren der implementieren Lösung, allfällige Korrekturen vornehmen
20.03.2020	Abgabe Bachelorthesis	Übergabe von Thesis an Betreuer
13. - 24.04.2020	Verteidigung	Verteidigung der Thesis vor Betreuer und Experten

In der Grafik 2.1 wird die Tabelle 2.1 dargestellt. Für Januar 2020 ist bewusst nur ein Meilenstein definiert, da dort die Modulschlussprüfungen geschrieben werden müssen.

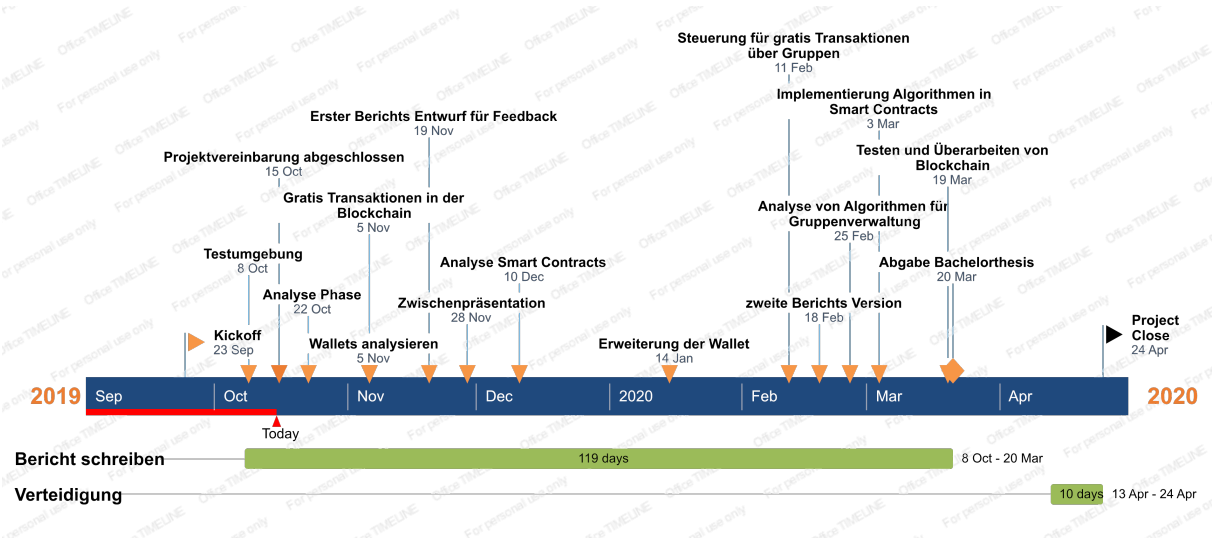

Abbildung 2.1: Zeitstrahl

2.2 Bericht

Wir haben uns dazu entschlossen, den Bericht während des Projekts zu schreiben und nicht nach der praktischen Arbeit. Dadurch können Erkennktnisse und Entscheidungen zeitnahe dokumentiert werden. Dieser Task ist in der Grafik 2.1 ersichtlich.

3 Risiken

In der Tabelle 3.1 sind die wichtigsten Risken aufgelistet. In der Spalte Auftreten wird die geschätzte Wahrscheinlichkeit eines Eintreffens des Risikos beschrieben. Die Spalte Auswirkung beschreibt die Schwere beim Eintreffen des Risikos. Bei beiden Spalten ist der Wert 1 das Minimum und der Wert 3 das Maximum. Der Wert in der Spalte Kategorie wird aus der Multiplikation von Auftreten und Auswirkung gebildet. Ein Risiko kann also von 1 bis 9 gewertet werden. Je höher die Kategorie, umso gefährlicher ist ein Risiko.

Tabelle 3.1: Risiken

Risiko	Auftreten	Auswirkung	Kategorie	Gegenmassnahme
Teammitglied bricht Projekt ab	1	3	3	Gute Kommunikation unter den Teammitgliedern. Protokollieren, wer was erledigt hat. Planung in Zusammenarbeit mit den Betreuern überarbeiten
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
Datenverlust	1	3	3	Daten werden niemals nur auf einem Medium gespeichert. Versionierung mit GitHub
Themen zu Komplex	2	2	4	Intensivierung der Analysephasen. Gegenseitige Unterstützung der Teammitglieder. Informieren der Betreuer und eventuelle Anpassung der Planung

4 Entwicklungsumgebung

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

4.1 Blockchain

Es wird eine Test-Blockchain aufgesetzt. Diese wird benötigt, um geschriebenen Code zu testen und analysieren.

Als Blockchain wird Ethereum[3] verwendet. In den nachfolgenden Absätzen werden mögliche Tools besprochen, die für den Aufbau von einer Testumgebung genutzt werden können.

4.1.1 Client

In der Arbeit wird evaluiert ob Geth[4] als Client den Ansprüchen genügt oder ob ein anderer Client (z.B. Parity, Aleth, Ethereum J, etc.) zum Einsatz kommt.

4.1.1.1 Trufflesuite

Trufflesuite[5] wird verwendet, um eine simulierte Blockchain aufzusetzen. Diese kann für die Einarbeitung in die Materie genutzt werden.

4.2 Wallet

Es werden verschiedene Wallets auf ihre Funktionalität untersucht. Zu den möglichen Wallets gehören z.B.:

- uPort[6]
- Metamask[7]
- Atomic Wallet [8]
- Exodus[9]

Es wird davon ausgegangen, dass keine Wallet alle Bedürfnisse abdecken kann, daher wird die gewählte Wallet im Zuge dieses Projekts erweitert. Für Ethereum existiert ein offizieller Service um eine eigene Wallet zu ertellen: MyEtherWallet[10]

4.3 Smart Contracts

Smart Contracts werden benötigt, um zu bestimmen, wer auf einer Blockchain gratis Transaktionen ausführen kann. Sobald eigene Smart Contracts entwickelt werden, kann die Testumgebung genutzt werden, um diese zu testen.

4.3.1 Programmiersprache

Für die Entwicklung von Smart Contracts werden folgende zwei Sprachen evaluiert:

- Solidity[11]
- Vyper[12]

5 Quellenverzeichnis

[1] M. Inc., „What is Gas | MyEtherWallet Knowledge Base“, 2018. [Online]. Verfügbar unter: https://kb.myetherwallet.com/en/transactions/what-is-gas/.

[2] „Denial-of-service attack - Wikipedia“, 2019. [Online]. Verfügbar unter: https://en.wikipedia.org/wiki/Denial-of-service_attack.

[3] Ethereum, „Home | Ethereum“, 2019. [Online]. Verfügbar unter: https://www.ethereum.org/.

[4] go-ehereum, „Go Ethereum“, 2019. [Online]. Verfügbar unter: https://geth.ethereum.org/.

[5] T. B. G. 2019, „Sweet Tools for Smart Contracts“, 2019. [Online]. Verfügbar unter: https://www.trufflesuite.com/.

[6] uPort, „uPort“, 2019. [Online]. Verfügbar unter: https://www.uport.me/.

[7] MetaMask, „MetaMask“, 2019. [Online]. Verfügbar unter: https://metamask.io/.

[8] A. Wallet, „Atomic Cryptocurrency Wallet“, 2019. [Online]. Verfügbar unter: https://atomicwallet.io/.

[9] E. M. Inc., „Crypte Wallet - Send, Receive & Exchange Cryptocurrency | Exodus“, 2019. [Online]. Verfügbar unter: https://www.exodus.io.

[10] MyEtherWallet, „MyEtherWallet | MEW“, 2019. [Online]. Verfügbar unter: https://www.myetherwallet.com/.

[11] Solidity, „Solidity - Solidity 0.5.11 documentation“, 2019. [Online]. Verfügbar unter: https://solidity.readthedocs.io/en/v0.5.11/.

[12] „Vyper–Vyper documentation“, 2019. [Online]. Verfügbar unter: https://vyper.readthedocs.io/en/v0.1.0-beta.13/#.