TU WIEN Informatics

Adaptor Signature Based Atomic Swaps Between Bitcoin and a Mimblewimble Based Cryptocurrency

MASTER'S THESIS

submitted in partial fulfillment of the requirements for the degree of

Master of Science

in

Software Engineering Internet Computing

by

Jakob Abfalter, BSc
Registration Number 01126889

to the Faculty of Informatics

at the TU Wien

Advisor: Univ. Prof. Dr. Matteo Maffei
Assistance: Dr. Pedro Moreno Sanchez

Vienna, 6th April, 2020

Jakob Abfalter	Matteo Maffei

Erklärung zur Verfassung der Arbeit

Jakob Abfalter, BSc

Hiermit erkläre ich, dass ich diese Arbeit selbständig verfasst habe, dass ich die verwendeten Quellen und Hilfsmittel vollständig angegeben habe und dass ich die Stellen der Arbeit – einschließlich Tabellen, Karten und Abbildungen –, die anderen Werken oder dem Internet im Wortlaut oder dem Sinn nach entnommen sind, auf jeden Fall unter Angabe der Quelle als Entlehnung kenntlich gemacht habe.

Wien, 6. April 2020

Jakob Abfalter

Acknowledgements

Enter your text here.

Abstract

Enter your text here.

Contents

CHAPTER 1

Introduction

TODO

CHAPTER 2

Motivation & Objectives

TODO

CHAPTER 3

Preliminaries

3.1 Bitcoin

3.1.1 Bitcoin Transaction Protocol

3.1.2 Bitcoin Scaling and Layer Two Solutions

3.2 Privacy-enhancing Cryptocurrencies

3.2.1 Zero Knowledge Proofs

3.2.2 Range Proofs

3.2.3 Mimblewimble (Grin)

3.3 Hash-time-locked Contracts

3.4 Adaptor Signatures

3.4.1 Schnorr Signature Construction

3.4.2 ECDSA Signature Construction

CHAPTER 4

Adaptor Signature Based Atomic Swaps Between Bitcoin and Grin

4.1 General Notation

4.2 Cryptographic Primitives

4.3 Generalized Multiparty Adaptor Signature

We define a Generalized Multiparty Adaptor Signature Scheme from the standard construction of multiparty Schnorr signatures which are defined as follows:

$\underline{GEN()}$

$k \leftarrow\!\$\; \mathbb{Z}_q$

$r \leftarrow\!\$\; \mathbb{Z}_q$

return (k, r)

$\underline{GEN\_PART\_SIG(M, k, r, g^{k'}, g^{r'})}$

$e = h(M || g^k + g^{k'} || g^r + g^{r'})$

$sig\_part = k + e * r$

return $(sig_part, g^k, g^{r'})$

$\underline{VERF\_PART\_SIG(M, k, r, g^{k'}, g^{r'}, sig\_part)}$

$e = h(M || g^k + g^{k'} || g^r + g^{r'})$

return $g^{sig\_part} = g^{k'} + g^{e*r}$

$\underline{FINALIZE\_SIG(sig\_part, sig\_part', g^k, g^{k'}))}$

return $(sig\_part + sig\_part', g^k + g^{k'})$

In order to have adaptable partial signature we add the following procedures

$\underline{ADAPT\_PART\_SIG(sig\_part, x)}$

1 : $sig\_part\_apt = sig_part + x$

2 : **return** (sig_part_apt, g^x)

$\underline{EXT\_WIT(sig\_final, sig\_part, sig\_part\_apt')}$

1 : $sig\_part' = sig\_\mathbf{fi}nal - sig\_part$

2 : $x = sig\_part\_apt' - sig\_part'$

3 : **return** (x)

$\underline{VERF\_APT\_SIG(M, k, r, g^{k'}, g^{r'}, g^x, sig\_part\_apt')}$

1 : $e = h(M||g^k + g^{k'}||g^r + g^{r'})$

2 : **return** $g^{sig\_part\_apt'} = g^{k'} + g^{e*r} + g^x$

4.4 Atomic Swap Construction

4.4.1 Construction Bitcoin side

4.4.2 Construction Grin side

4.4.3 Security Definitions

CHAPTER 5

Implementation

5.1 Implementation Bitcoin side

5.2 Implementation Grin side

5.3 Performance Evaluation

CHAPTER 6

Implementation Security and Privacy Evaluation

6.1 Security Evaluation

6.2 Privacy Evaluation

CHAPTER 7

Related and Future Work

7.1 Payment Channel Networks on Grin

7.2 Payment Channel Networks on Monero

7.3 Atomic Swaps With Related Cryptocurrencies

7.4 Tumbler Based Atomic Swaps

CHAPTER 8

Conclusion

List of Figures

List of Tables

List of Algorithms

Bibliography