



# **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, 6<sup>th</sup> 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

<b>Abstract</b>	<b>vii</b>
<b>Contents</b>	<b>ix</b>
<b>1 Introduction</b>	<b>1</b>
<b>2 Motivation &amp; Objectives</b>	<b>3</b>
<b>3 Preliminaries</b>	<b>5</b>
3.1 Bitcoin . . . . .	5
3.2 Privacy-enhancing Cryptocurrencies . . . . .	5
3.3 Hash-time-locked Contracts . . . . .	5
3.4 Adaptor Signatures . . . . .	5
<b>4 Adaptor Signature Based Atomic Swaps Between Bitcoin and Grin</b>	<b>7</b>
4.1 General Notation . . . . .	7
4.2 Cryptographic Primitives . . . . .	7
4.3 Generalized Multiparty Adaptor Signature . . . . .	7
4.4 Atomic Swap Construction . . . . .	8
<b>5 Implementation</b>	<b>9</b>
5.1 Implementation Bitcoin side . . . . .	9
5.2 Implementation Grin side . . . . .	9
5.3 Performance Evaluation . . . . .	9
<b>6 Implementation Security and Privacy Evaluation</b>	<b>11</b>
6.1 Security Evaluation . . . . .	11
6.2 Privacy Evaluation . . . . .	11
<b>7 Related and Future Work</b>	<b>13</b>
7.1 Payment Channel Networks on Grin . . . . .	13
7.2 Payment Channel Networks on Monero . . . . .	13
7.3 Atomic Swaps With Related Cryptocurrencies . . . . .	13
7.4 Tumbler Based Atomic Swaps . . . . .	13
	ix

<b>8 Conclusion</b>	<b>15</b>
<b>List of Figures</b>	<b>17</b>
<b>List of Tables</b>	<b>19</b>
<b>List of Algorithms</b>	<b>21</b>
<b>Bibliography</b>	<b>23</b>

# 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





# 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:

<u><math>GEN()</math></u> 1 : $k \leftarrow \mathbb{Z}_q$ 2 : $r \leftarrow \mathbb{Z}_q$ 3 : <b>return</b> $(k, r)$	<u><math>GEN\_PT\_SIG(m, k, r, g^{k'}, g^{r'})</math></u> 1 : $e = h((m    g^k + g^{k'}    g^r + g^{r'}))$ 2 : $sig\_prt = k + e + r$ 3 : <b>return</b> $(sig\_prt, g^k, g^r)$ <u><math>VER\_PT\_SIG(m, k, r, g^{k'}, g^{r'}, sig\_prt)</math></u> 1 : $e = h(((m    g^k + g^{k'}    g^r + g^{r'}))$ 2 : <b>return</b> $g^{sig\_prt} = g^{k'} + g^{e+r}$ <u><math>FIN\_SIG(sig\_prt, sig\_prt', g^k, g^{k'}, g^r, g^{r'})</math></u> 1 : <b>return</b> $(sig\_prt + sig\_prt', g^k + g^{k'}, g^r + g^{r'})$
---	---

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

<u><math>APT\_PT\_SIG(sig\_prt, x)</math></u>	<u><math>EXT\_WIT(sig\_final, sig\_prt, sig\_part\_apt')</math></u>
1: $sig\_part\_apt = sig\_prt + x$	1: $sig\_prt' = sig\_final - sig\_prt$
2: <b>return</b> $(sig\_part\_apt, g^x)$	2: $x = sig\_part\_apt' - sig\_prt'$
	3: <b>return</b> $(x)$
<u><math>APT\_PT\_SIG(sig\_prt, x)</math></u>	
1: $e = h((m    g^k + g^{k'}    g^r + g^{r'}))$	
2: <b>return</b> $g^{sig\_part\_apt'} = g^{k'} + g^{e*r} + g^x$	

## 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



## 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