

Onderzoek naar
Gedistribueerde netwerken en identiteit
binnen Blockchain technologie

Jeffrey van Hoven
1 oktober 2018

Quintor



Voorwoord

Voor u ligt mijn afstudeerverslag; *“Onderzoek naar gedistribueerde netwerken en identiteit binnen Blockchain technologie”* waarin ik schrijf over de uitvoering van het onderzoek dat gedaan is ten behoeve van mijn afstuderen voor de opleiding Informatica aan de Haagse Hogeschool. Het betreft verslaglegging van het proces dat doorlopen is om de uitdagende opdracht zoals voorgesteld door Quintor uit te voeren.

In de opdracht is er na uitvoerig onderzoek tot de conclusie gekomen welke technieken geschikt zijn om toegepast te kunnen worden om de Blockchain onderdelen Distributed Network en Identity Management te realiseren. Gedurende het afstudeertraject kon ik altijd met vragen terecht bij zowel mijn bedrijfsbegeleider, Ben Ooms, als de Blockchain expert, Pim Otte.

Hierbij bedank ik mijn begeleiders, vanuit Quintor en vanuit de opleiding, voor hun begeleiding, inzichten en ondersteuning tijdens het afstudeertraject. Daarnaast wil ik graag mijn mede afstudeerders bedanken voor hun meedenken en inzichten in het vinden van oplossingen. In het bijzonder bedank ik mede afstudeerder Kevin Bos, waarmee de samenwerking gedurende de opdracht aangenaam en productief is geweest.

Als laatste bedank ik mijn familie en vrienden voor hun ondersteuning, indirect of direct, niet alleen tijdens het afstuderen, maar ook tijdens mijn studieloopbaan. Zonder hun ondersteuning zou dit niet mogelijk geweest zijn.

Ik wens u veel leesplezier toe.

Jeffrey van Hoven
Den Haag, 1 juni 2018

Inhoudsopgave

1	Inleiding	1
2	Quintor	2
2.1	Software Factory	2
2.2	Visie	3
2.3	Organisatie	3
2.4	Infrastructuur	4
2.5	Betrokkenen	5
3	Opdracht	6
3.1	Probleemstelling	7
3.2	Doelstelling	7
3.3	Resultaat	8
3.3.1	Producten	8
4	Aanpak	10
4.1	Inrichting	10
4.2	Fasering	10
4.3	Fase 1: Onderzoek	11
4.3.1	Vooronderzoek	11
4.3.2	Hoofdonderzoek	11
4.4	Fase 2: Ontwerpen	11
4.5	Fase 3: Realisatie	12
4.6	Planning	13
5	Oriëntatie	14
5.1	Omwenteling en implicaties	14
6	Vooronderzoek en resultaten	16
6.1	Blockchain	17
6.1.1	Conclusie	18
6.2	Toepassing	19
6.2.1	Politiek	19
6.2.2	Ontwikkelpлатform	20
6.2.3	Conclusie	21
6.3	Architectuur	22
6.4	Gedistribueerd netwerk	23
6.5	Identiteit	26
6.5.1	Autorisatie	27
6.6	Obstakels	29

6.7	Conclusie	29
7	Onderzoek	30
7.1	Opzet	31
7.1.1	Dataverzameling	31
7.1.2	Dataomschrijving	31
7.1.3	Analysemethode	31
7.2	Selectie protocollen	32
7.2.1	Coinmarketcap	32
7.2.2	Attributen	33
7.2.3	Totstandkoming	34
7.2.4	Resultaat	35
7.2.5	Afwijking	35
7.3	Soorten netwerken	36
7.3.1	Aanpak	36
7.3.2	Conclusie	37
7.4	Functionaliteit en gevaren	38
7.4.1	Aanpak	39
7.4.2	Conclusie	40
7.5	Identiteit	42
7.5.1	Aanpak	42
7.5.2	Conclusie	43
7.6	Conclusie	44
7.7	Resultaat	45
8	Advies	46
9	Ontwerpen	47
9.1	Methode	47
9.1.1	Kruchten	47
9.1.2	Rozanski en Woods	48
9.2	Opstellen scenarios	49
9.3	Development View	50
10	Ontwikkelen	52
10.1	Selecteren methoden, technieken en tools	52
10.1.1	Programmeertaal	52
10.1.2	Versiebeheer	53
10.1.3	Deployment	53
10.2	Configureren	54
10.2.1	Maven	54
10.2.2	Docker	54
10.3	Testen	55
10.3.1	Test Driven Development	55
10.3.2	Acceptance Test Driven Development	55

10.3.3	Behaviour Driven Development	56
10.3.4	Keuze	56
10.3.5	Frameworks	56
10.4	Sprint een: Realiseren Peer to Peer netwerk	58
10.4.1	Protocol	58
10.4.2	Ontwerpen	59
10.4.3	Versturen berichten	60
10.4.4	Resultaat	61
11	Evaluatie	62
11.1	Producten	62
11.1.1	Plan van Aanpak	62
11.1.2	Onderzoeksrapport	62
11.1.3	Architectuurdocument	63
11.1.4	Proof of Concept	63
11.2	Aanpak	63
11.2.1	Onderzoek	63
11.3	Beroepstaken	64
11.4	Functioneren binnen Quintor	65
11.5	Leerpunten	66
12	Aanbevelingen	68
12.1	Directed Acyclic Graph	68
12.2	Bitcoin Lightning Network	68
12.3	Ethereum Casper	68
12.4	EOS	69
12.5	Network Address Translators (NAT) Hole Punching	69
	Literatuurlijst	70
	Bijlages	72
I	Opdrachtformulering	73
II	Afstudeerplan	76
III	Plan van Aanpak	81
IV	Onderzoeksrapport	94
V	Architectuurdocument	132
VI	Voortgangsverslag	149
VII	Bezoekverslag	151
VIII	Implementatie selectie	153

Lijst van figuren

2.1	Organogram van Quintor.	4
3.1	Indeling opdracht Blockchain Quintor.	6
6.1	Blockchain architectuur	22
6.2	Distributed Hash Table	24
6.3	Bitcoin Node functionaliteiten	25
6.4	Asymmetrische encryptie	26
6.5	Gebruik van asymmetrische encryptie	26
7.1	Snapshot Coinmarketcap	32
7.2	Opbouw beantwoording "Soorten netwerken"	36
7.3	Opbouw beantwoording "Functionaliteit en gevaren"	39
7.4	Opbouw beantwoording "Identiteit"	42
9.1	4+1 view-model	47
9.2	Voorbeeld use-case: Connectie leggen deelnemer	49
9.3	Use-case diagram	50
9.4	Development View	51
10.1	Werking Docker	53
10.2	Configuratie Maven	54
10.3	Configuratie Docker	54
10.4	Gedetailleerd overzicht Network component	60
10.5	Protobuf	61

Lijst van tabellen

4.1	Globale planning	13
7.1	Attributen opgesteld voor initiële selectie implementaties.	33
10.1	Betrokken architectuur onderdelen implementatie Peer-to-Peer netwerk	58
1	Bekeken implementaties uit de initiële selectie met de onderzochte attributen.	154

Woordenlijst

F

fork Splitsing in het netwerk dat veroorzaakt is door een kleine wijziging in het protocol. 24

full node Node die alle functionaliteit kan uitvoeren die de Blockchain implementatie aanbiedt. 25

H

Hard Fork Een verandering in het Blockchain protocol die een nieuwe regel in het netwerk introduceert, waardoor het protocol geen compatibiliteit heeft met eerder versies. 24

I

IPv6 Zesde versie van het Internet Protocol (IP). 69

M

majority attack Een aanval waarbij meer als 51% van de voting power in handen is van een kwaadwillende deelnemer. 38

mining node Node die als enige taak heeft om het mining proces uit te voeren. vii, 25

N

node Computer dat in verbinding staat met het netwerk van de Blockchain. vii, 25, 37, 38

O

OTAP Best practice voor inrichting software ontwikkelstraat, waarbij er een Ontwikkelomgeving, Testomgeving, Acceptatieomgeving en Productieomgeving gehanteerd wordt. 52, 53

P

packet Een encapsulatie van data dat gebruikt wordt door Transmission Control Protocol (TCP) en User Datagram Protocol (UDP) implementaties.. 58, 59

S

selfish mining Aanval waarbij er door een kwaadwillende mining node blocks achtergehouden worden. 40

Smart Contract Een protocol dat gebruikt wordt om een digitale onderhandeling te faciliteren, verifiëren of forceren van een contract. 19, 21

Soft Fork Een verandering in het Blockchain protocol die terugwaartse compatibiliteit heeft met eerdere versies van het protocol. 24

stake Investing in de Blockchain proportioneel naar het type consensus, meestal gebruikt in PoS implementaties. 37

T

tunnel Communicatiekanaal zoals in gebruik bij Monero. 41

W

wallet (node) Node die een gereduceerde staat van het Blockchain bevat, waarin alleen de transacties opgenomen worden die betrekking hebben op de public- en private key combinatie. 25

Afkortingen

A

ATDD Acceptance Test Driven Development. 55

B

BDD Behaviour-driven development. 55, 56

C

CI Continuous Integration. 56

D

DAG Directed Acyclic Graph. 68

DApp Distributed Applications. 20

DHT Distributed Hash Table. 23

I

I2P The Invisible Internet Project. 41

ICOs Initial Coin Investment. 63

IDE Integrated Development Environment. 52, 54

IP Internet Protocol. vii

J

JDK Java Development Kit. 54

JRE Java Runtime Environment. 54

JVM Java Virtual Machine. 52, 60

N

NAT Network Address Translators. iv, 69

P

P2P Peer-to-Peer. 23, 25, 58, 60, 61, 69

PoS Proof of Stake. vii, 41

T

TCP Transmission Control Protocol. vii, 58, 59

TDD Test-driven Development. 55

U

UDP User Datagram Protocol. vii, 58, 59

1 | Inleiding

Dit verslag is geschreven in het kader van mijn afstudeeropdracht bij Quintor en dient ter beoordeling van de werkzaamheden die uitgevoerd zijn voor de bachelorstudie Informatica aan de Haagse Hogeschool.

Door de snelle groei van het Blockchain domein heeft Quintor in 2017 in samenwerking met DUO/-MinOCW, Groningen Declaration Network, Stichting ePortfolio Support, TNO en Rabobank, het Blockchain Field-lab Education gestart in Groningen. Het Blockchain-lab is opgezet om expertise en kennis uit te wisselen op regionaal, nationaal en internationaal gebied. De oprichting van het Blockchain Field-lab Education heeft er mede voor gezorgd dat Quintor meer kennis wilt opdoen over het Blockchain domein om zo inzicht te krijgen in hoe Blockchain technologie ingezet kan worden binnen vraagstukken vanuit klanten.

In hoofdstuk 2 is de organisatie beschreven waar het afstudeertraject heeft plaatsgevonden. Vervolgens wordt in hoofdstuk 3 de opdracht gepresenteerd. In hoofdstuk 4 wordt de aanpak van de opdracht onderbouwd en in hoofdstuk 5 wordt de orientatie van de opdracht besproken. In hoofdstuk 6 worden de werkzaamheden van het vooronderzoek besproken waarin de basis van Blockchain technologie ter sprake komt. In hoofdstuk 7 wordt het hoofdonderzoek naar de segmenten Identity Management en Distributed Network in bestaande Blockchain implementaties beschreven. Hoofdstuk 8 presenteert het advies dat gegeven wordt naar aanleiding van het gedane onderzoek. Hoofdstuk 9 beschrijft de totstandkoming van het ontwerp voor de architectuur. In hoofdstuk 10 worden de keuzes die gemaakt zijn voor de indeling van het Proof of Concept beschreven en wordt er kort uitgelicht hoe de realisatie van het Peer-to-Peer netwerk is verlopen. In hoofdstuk 11 wordt er geëvalueerd over de producten, de aanpak, de geselecteerde beroepstaken, het functioneren in het bedrijf en de leerpunten die getrokken zijn uit het project. Tot slot worden er in hoofdstuk 11 aanbevelingen gegeven voor vervolgonderzoek.

2 | Quintor

In dit hoofdstuk zal er inzicht gegeven worden over het bedrijf Quintor waar het afstudeerproject heeft plaatsgevonden. Er wordt verteld over de diensten die Quintor levert en wat de doelen zijn van de organisatie. Er zal ook kort toegelicht worden waar de afstudeerder binnen het bedrijf opereert en welke werknemers vanuit Quintor betrokken zijn bij de afstudeeropdracht.

Quintor is een toonaangevend bedrijf op het gebied van Agile software development, Enterprise Java / .NET technologie en mobile development. Het bedrijf is begonnen in 2005 in Groningen en is opgericht door Johan Tillema, de huidige CEO van het bedrijf. Sinds 2005 heeft het bedrijf een gezonde groei doorgemaakt en heeft inmiddels 150 medewerkers, verspreid over vestigingen in Groningen, Amersfoort en Den Haag. Vanuit deze vestigingen ondersteunt het bedrijf klanten bij de uitdagingen die grootschalige Enterprise projecten met zich meebrengen. Het succes van Quintor is te danken aan drie pijlers: techniek en architectuur, een hoogwaardige ontwikkelstraat en het Agile/Scrum proces. Tevens beschikt Quintor over een Software Factory waarin in-house projecten voor klanten worden uitgevoerd.

2.1 Software Factory

In de Software Factory staat alle kennis en expertise die Quintor heeft verzameld over de jaren heen. Het is een hoogwaardig platform waarin de tooling, standaarden en best practices en tevens een complete oplossing is voor het managen en hosten van Scrum projecten. Dit wordt onder andere gebruikt om klanten te helpen bij het professionaliseren en efficiënter inrichten van softwareontwikkeling. Een groot deel van de werkzaamheden die Quintor dan ook uitvoert voor klanten is consultatie bij o.a. het implementeren van Agile/Scrum werk- en denkwijze. Naast Java en .NET development behoort ook mobile development tot de kerncompetenties van Quintor, en is dan ook opgenomen in de Software Factory. Hieronder zijn een aantal onderdelen uitgelicht die voorkomen in de Software Factory.

Enterprise architectuur Vanuit een pragmatische insteek en op basis van jarenlange ervaring helpen de software architecten van Quintor organisaties bij het maken van de juiste keuzes op het gebied van architectuur. Hierbij gaat het om het zowel opstellen als implementeren van een architectuur.

Informatie analyse Het in kaart brengen van informatie door het gebruik van diverse analyse- en ontwerptechnieken zoals UML, user-stories en use-cases in een Agile omgeving.

Java en .NET development Het realiseren van duurzame IT-systemen, in-house of bij klanten, die naadloos aansluiten bij de wensen van de business. Hierbij zijn er een groot aantal van omvangrijke systemen ontwikkelt.

Agile/Scrum Agile/Scrum is een effectieve en flexibele methode die uitgaat van een iteratiefontwikkelp proces. Het trainen van complete projectteams met een op maat gemaakte training, waarbij aansluitend support en coaching gegeven wordt.

Mobile development Mobiele applicaties voor iPhone, iPad en Android. Specifiek hiervoor is het 'Mobile development center' opgezet, waarin er apps ontworpen, ontwikkelt en beheert worden. Dit betreft de realisatie van stand-alone tot volledige geïntegreerde Enterprise apps.

2.2 Visie

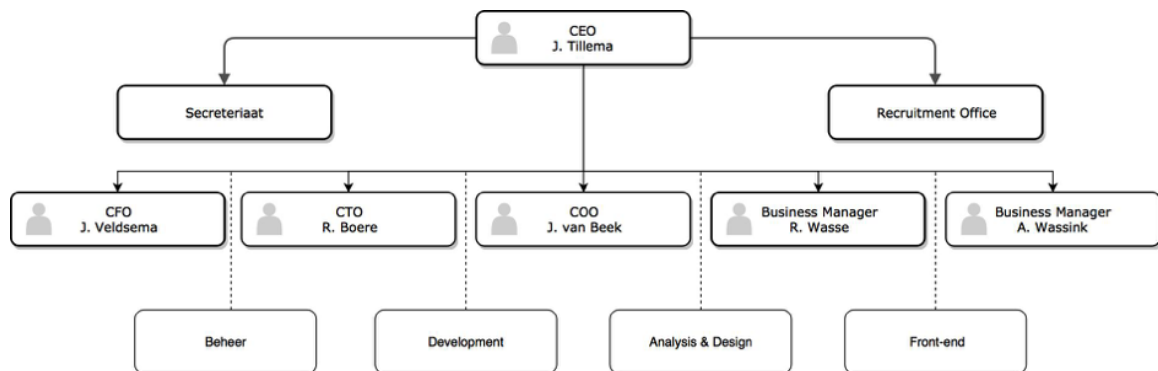
Een van de doelstellingen die Quintor heeft is het professionaliseren van software development. Aansluitend daarop probeert het bedrijf continu voor te lopen op de concurrentie door het opdoen van kennis op het gebied van nieuwe technologieën, waarbij professionalisering van de werkwijze voorop staat.

"Onze ambitie: professionaliseren van software development."
Johan Tillema, Chief Executive Officer

Door het aanbieden van uitdagende afstudeeropdrachten wordt er kennis opgebouwd die benodigd is om nieuwe technologieën, zoals bijvoorbeeld Machine Learning of Blockchain, in te zetten om klanten te adviseren bij de uitdagingen die grootschalige Enterprise projecten met zich meebrengen en zal, middels het volwassen genoeg is, opgenomen worden in de Software Factory.

2.3 Organisatie

In fig. 2.1 wordt de organisatie van Quintor weergegeven. Bovenaan staat Johan Tillema, de oprichter en CEO. Direct eronder staat het Secretariaat en het Recruitment Office. Hierin is te zien dat er vier segmenten zijn waarop er consultatie aangeboden wordt: development, analysis en design en front-end.



Figuur 2.1: Organogram van Quintor.

Zelf val ik onder het development segment, waarbij er aangestuurd wordt door Ben Ooms, (beschreven in 2.5). Er wordt zelfstandig gewerkt aan de opdracht waarbij er een aantal praktijken van Scrum toegepast zijn tijdens het afstudeertraject. Zo is er elke twee weken een zogenaamde demo dag waarbij iedere afstudeerder een demonstratie over waar hij of zij de afgelopen tijd mee bezig is geweest, en of er ergens tegenaan gelopen wordt zodat er samen nagedacht kan worden over mogelijke oplossingen.

2.4 Infrastructuur

Quintor maakt intensief gebruik van het Agile principe en dit is dan ook terug te vinden in de infrastructuur die ingericht is voor de consultants binnen Quintor. Voor het uitvoeren van projecten wordt Atlassian JIRA gebruikt om het Agile proces te ondersteunen. Hierin is het mogelijk om taken te creëren en toe te wijzen aan projecten. Elke taak is dan individueel op te pakken door een team die op een bepaald project gezet is.

Voor het waarborgen van de kwaliteit van de software wordt er gebruik gemaakt van Atlassian BitBucket. BitBucket is een web-based versiebeheer systeem dat het Mercurial of Git revisiesystemen ondersteund. Daarnaast werkt het uitstekend samen met JIRA, waarbij het mogelijk is om naar taken die opgepakt zijn binnen JIRA te refereren binnen BitBucket.

Communicatie binnen de organisatie gaat via het interne mail-systeem die functionaliteiten ondersteund zoals bijvoorbeeld het interne chat-systeem waarbij het mogelijk is om elke medewerker te benaderen en een kalender die het mogelijk maakt om afspraken te koppelen aan medewerkers en locaties.

2.5 Betrokkenen

Binnen Quintor zijn er een aantal medewerkers die nodig zijn om het project tot een geslaagd einde te brengen. Hieronder zijn deze medewerkers kort benoemd en wat hun rol is binnen het afstudeertraject.

Ben Ooms is de teamleider van Quintor Den Haag en is tevens de begeleider tijdens het afstudeertraject. Zijn uitvoerende taken hierbij zijn dan ook onder andere advies geven over de aanpak van de opdracht en waarbij mogelijk de voortgang van de opdracht te waarborgen.

Pim Otte is de Blockchain expert binnen Quintor en heeft veelal ervaring met de toepassing en realisatie van applicaties die gebruik maken van Blockchain technologie. Hij is beschikbaar gedurende de afstudeeropdracht om inzichten en feedback te geven op de uitgevoerde werkzaamheden.

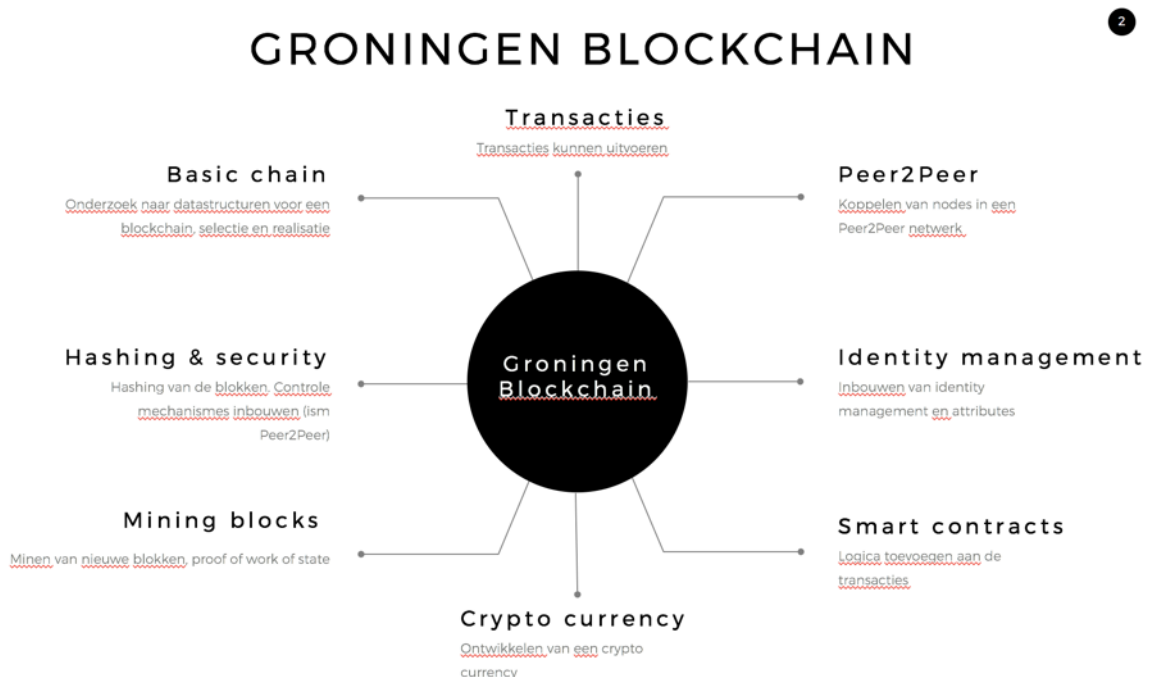
Kevin Bos is afstudeerder afkomstig van Avans Hogeschool. Hij is verantwoordelijk voor het lokale gedeelte van de Blockchain opdracht. Tijdens de afstudeeropdracht is hij een stakeholder van het project en zal er een zekere mate van samenwerking aanwezig zijn.

3 | Opdracht

In dit hoofdstuk wordt de opdracht uitgelegd zoals gegeven door Quintor. Het betreft de aanleiding van de opdracht en het uiteindelijke doel Quintor wilt behalen door het faciliteren van de afstudeeropdracht.

Sinds de opkomst van Bitcoin is de Blockchain technologie, de techniek die het mogelijk maakt om het op een gedecentraliseerde manier te laten werken, steeds populairder geworden. Alhoewel de Blockchain-technologie nog in de kinderschoenen staat, gaan de ontwikkelingen in het domein zeer snel. Zo worden er toepassingen bedacht die niet alleen voor de financiële markten interessant zijn, maar ook voor bijvoorbeeld het digitaliseren van contracten en contractbeheer.

Door de snelle groei van het Blockchain domein heeft Quintor in 2017 in samenwerking met DUO/-MinOCW, Groningen Declaration Network, Stichting ePortfolio Support, TNO en Rabobank, het Blockchain Field-lab Education gestart in Groningen. Het Blockchain-lab is opgezet om expertise en kennis uit te wisselen op regionaal, nationaal en internationaal gebied. De oprichting van het Blockchain Field-lab Education heeft er mede voor gezorgd dat Quintor afstudeeropdrachten aanbiedt voor het Blockchain domein om zo de huidige kennis over het domein uit te breiden en/of te toetsen.



Figuur 3.1: De indeling van de Blockchain opdracht zoals gegeven door Quintor. Door het domein op te delen in segmenten is het mogelijk om elk individueel segment uit te lichten in de vorm van onderzoek, zoals te zien in bijlage I.

Aangezien het Blockchain domein complex en veelomvattend is, is het domein opgedeeld in segmenten. De verschillende segmenten, zoals weergegeven in fig. 3.1, worden aangeboden als individuele afstudeeropdrachten. De focus van deze opdracht zijn de segmenten Peer2Peer en Identity Management, waarbij door middel van gestelde uitgangspunten op het gebied van snelheid, beveiligingsniveau en toepassingsmogelijkheden een Proof of Concept gerealiseerd dient te worden. Alvorens het Proof of Concept gerealiseerd kan worden, worden de alternatieve architecturen op het gebied van Peer2Peer en Identity Management in kaart gebracht. Dit wordt gedaan door het uitvoeren van literatuur onderzoek naar keuzes die gemaakt zijn in huidige Blockchain implementaties.

Daarnaast worden de volgende eisen gesteld aan het Proof of Concept:

1. Er worden geen Blockchain libraries gebruikt.
2. Het moet resistent zijn tegen aanvallen.
3. Het moet gedistribueerd zijn.
4. Er wordt op decentrale wijze consensus bereikt.

3.1 Probleemstelling

Doordat de toepassing en adoptie van Blockchain technologie steeds groter wordt wil Quintor de toepassingsmogelijkheden en technieken onderzoeken om te kijken of het haalbaar is om een Blockchain implementatie te realiseren, waarna er gekeken zal worden of Quintor de technologie kan gebruiken in vraagstukken vanuit klanten.

3.2 Doelstelling

Door het Blockchain domein op te delen in segmenten, te zien in fig. 3.1, is het mogelijk om de segmenten te behandelen in afstudeeropdrachten die Quintor aanbiedt. De focus in deze opdracht ligt op de Blockchain onderdelen Identity Management en Peer2Peer (Distributed Network). Hierdoor is er een globaal doel en een doel die specifiek voor deze opdracht geldt. Het streven van het globale doel is het opdoen van kennis omtrent het realiseren van een Blockchain implementatie door het creëren van individuele segmenten. Het doel van deze specifieke opdracht is middels het opstellen van een Proof of Concept van de Blockchain onderdelen Identity Management en Distributed Network, zonder gebruik te maken van bestaande Blockchain oplossingen, kennis te ontwikkelen voor Quintor op het gebied van Blockchain technologie.

3.3 Resultaat

Indien de opdracht succesvol afgerond is, zijn de segmenten Identity Management en het Distributed Network gerealiseerd in de vorm van een Proof of Concept en voldoen aan de eisen die gesteld zijn in de opdrachtformulering. Dit Proof of Concept zal werken met de segmenten, Basic Chain, Hashing & security en Mining blocks, die gerealiseerd zullen worden door Kevin Bos. Zowel het Proof of Concept als het onderzoek zal voor Quintor inzicht bieden in het Blockchain domein en de ontwikkelingen daarin.

3.3.1 Producten

Als onderdeel van de afstudeeropdracht zullen er verschillende producten worden opgeleverd aan Quintor en aan de Haagse Hogeschool. Deze staan hieronder gespecificeerd.

De op te leveren producten aan Quintor zijn:

- **Adviesrapport**

Presentatie over de resultaten van het onderzoek waarin verschillende technieken geadviseerd worden die toegepast zijn in de realisatie van het Proof of Concept.

- **Sprint demo presentaties**

Elke twee weken zal er een presentatie gegeven worden over de voortgang van het project waarbij het mogelijk is om feedback te krijgen over blokkades of aanpakken.

- **Broncode van het Proof of Concept**

De gehele broncode van de applicatie waarin technieken vanuit het adviesrapport gerealiseerd zijn.

- **Onderzoeksrapport**

De resultaten van het onderzoek dat uitgevoerd is om inzicht te krijgen in de segmenten Distributed Network en Identity Management.

De op te leveren producten aan de Haagse Hogeschool zijn:

- **Afstudeerscriptie**

Beschrijving van het proces tijdens de uitvoering van de afstudeeropdracht ter beoordeling van de bekwaamheid van de student en de geselecteerde beroepstaken.

- **Verslag bedrijfsbezoek**

Verslag van het bedrijfsbezoek dat tijdens het afstudeertraject gedaan wordt.

- **Voortgangsverslag**

Verslag van de voortgang van de afstudeeropdracht.

4 | Aanpak

In dit hoofdstuk wordt de totstandkoming van de aanpak voor de opdracht besproken. Het gaat uit van de beginsituatie zoals beschreven in het afstudeerplan, in te zien in bijlage II. Daarnaast worden de afwijkingen besproken tegenover de originele opdracht, zoals geformuleerd in hoofdstuk 3.

4.1 Inrichting

Aangezien Quintor een groot voorstander is van het Agile werken zien zij ook graag het afstudeertraject in die vorm uitgevoerd worden. Tijdens de studie is er veel ervaring opgedaan met het Scrum framework dat het Agile principe ondersteund waardoor het een logische keuze is om de structuur van het project te bepalen. Omdat een groot gedeelte van het project bestaat uit het individueel uitvoeren van onderzoek zijn niet alle best practices overgenomen.

De rollen binnen het scrum proces (Schwaber & Sutherland, 2011) zijn als volgt gedefinieerd:

- **Scrum Master** - Ben Ooms
- **Product Owner** - Johan Tillema / Ben Ooms
- **Development Team** - Jeffrey van Hoven

Een sprint zal bestaan uit twee weken waarbij aan het eind van de sprint een demo van de huidige status van het project wordt gegeven aan de afstudeerbegeleider vanuit Quintor. Tijdens dit moment is het mogelijk om advies te krijgen over de uitvoering van de werkzaamheden of blokkades waar tegenaan gelopen wordt. Daarnaast wordt er onder de afstudeerders dagelijks een stand-up gehouden waarin zaken zoals de status van het project, welke werkzaamheden gepland staan voor de dag en of er obstakels zijn besproken worden.

4.2 Fasering

Er wordt uitgegaan van drie fases binnen het project: **onderzoek**, **ontwerp** en **realisatie**, waarbij de fases ontwerp en realisatie Agile uitgevoerd worden volgens de Scrum richtlijnen. Deze fases zijn gedefinieerd naar aanleiding van de opdrachtomschrijving, waarin geadviseerd wordt om in deze vorm het project uit te voeren.

4.3 Fase 1: Onderzoek

4.3.1 Vooronderzoek

In het afstudeertraject wordt er met technologieën gewerkt welke onbekend zijn. Er is er dan ook voor gekozen om aan de hand van vooronderzoek kennis op te doen over het Blockchain domein. Er zal eerst onderzocht worden wat een Blockchain is waarna er ingegaan wordt op de toepassing van de techniek. Vervolgens zal er worden gekeken naar de architectuur van de Blockchain en uit welke componenten het bestaat. Uiteindelijk zal er kennis opgedaan worden voor de onderdelen Identity Management en Distributed Network om zo een afbakening te creëren van de onderdelen. Deze kennis zal gebruikt worden, in overleg met Quintor, om de opdracht vorm te geven en inzichten op te doen over de mogelijkheden met de opdracht.

Dataverzameling

Voor het opdoen van voorkennis zullen er gepubliceerde research papers, wiki's en blogs gebruikt worden. Hierna zal er een selectie van Blockchain implementaties gemaakt worden die bestudeerd zullen worden in het onderzoek.

4.3.2 Hoofdonderzoek

Het onderzoek zal bestaan uit een literatuurstudie naar de onderdelen Identity Management en Distributed Network zoals geïmplementeerd in bestaande Blockchain implementaties. Het resultaat van het onderzoek zal een adviesrapport zijn die in overeenstemming met het bedrijf doorslaggevend zal zijn voor de technieken en/of protocollen die gebruikt zullen worden in de te realiseren van Proof of Concept.

4.4 Fase 2: Ontwerpen

Uit het hoofdonderzoek zullen methoden en technieken geselecteerd worden die gerealiseerd zullen worden in een Proof of Concept. Alvorens deze gerealiseerd zal worden moet er nagedacht worden over hoe dit eruit zal komen te zien op technisch gebied. Het modelleren, implementeren en documenteren van een systeem vereist dat het systeem vanuit verschillende aspecten wordt bekeken. Er zal een keuze gemaakt worden betreft een methode voor het faciliteren van deze filosofie.

4.5 Fase 3: Realisatie

De uitgekozen technieken zullen gerealiseerd worden in een Proof of Concept. Dit zal in samenwerking zijn met de andere afstudeerder, die het lokale gedeelte van de Blockchain ontwikkeld. De onderdelen dienen samen te werken tot een functionele Blockchain implementatie, waarbij er overlap zal zijn in de keuzes binnen de pakketselectie en realisatie.

Requirements Er dienen criteria opgesteld te worden aan de hand van het resultaat van het onderzoek die van toepassing zijn op de realisatie van het Proof of Concept. Om te achterhalen wat de eisen en de toepassing waaraan het Proof of Concept moet voldoen zullen er informele interviews gehouden worden waarin requirements achterhaald worden.

Selecteren methoden Voor het opzetten van een development workflow en de technieken die daarbij te pas komen in overeenstemming met Quintor en de andere afstudeerder, zullen er beslissingen gemaakt worden op de manier waarop het Proof of Concept gerealiseerd gaat worden.

4.6 Planning

Er is een globale planning gemaakt die uitgaat van de drie gestelde fases: **onderzoek**, **ontwerp** en **realisatie**. Er wordt ervan uitgegaan dat elk van deze fases ongeveer $\frac{1}{3}$ de van de tijd in beslag zullen nemen. Aangezien de fases ontwerp en realisatie Agile uitgevoerd worden wordt er vanuit gegaan dat er niet veel tijd besteed zal worden aan het ontwerpen alvorens begonnen wordt aan de realisatie. Binnen deze drie fases zal er tijd gealloceerd worden voor het inventariseren, opzetten en opdoen van benodigde kennis en documentatie. In onderstaand tabel is een globale planning opgezet voor de drie fases.

Fase	Van	Tot en met
Onderzoek	Week 3	Week 9
Ontwerp	Week 10	Week 17
Realisatie	Week 11	Week 17
Afronding	Week 18	-

Tabel 4.1: Globale planning

Waarbij de werkzaamheden die uitgevoerd zullen worden binnen een fase er ongeveer als volgt uitzien:

- **Onderzoek**
 - Vooronderzoek
 - Selectie implementaties
 - Onderzoeksopzet
 - Onderzoek
 - Advies
- **Ontwerp**
 - Selecteren methode
- **Realisatie**
 - Ontwerp individuele views
 - Informatieplan
 - Realisatie in vorm van sprints
 - Testen
- **Afronding**
 - Overdracht
 - Afronding afstudeerverslag

5 | Oriëntatie

Dit hoofdstuk beschrijft het proces van de benodigde inventarisatie om duidelijkheid te krijgen over de criteria die gesteld zijn in de originele opdracht. Het doel van het uitvoeren van de inventarisatie is het opstellen van requirements en/of criteria die gesteld zijn aan het onderzoek en het uiteindelijke Proof of Concept.

Zoals eerder besproken bevat de opdrachtschrijving criteria die nader gespecificeerd dienen te worden. Deze criteria gaan over snelheid, beveiliging en toepassingsmogelijkheden waarbij elke invulling van deze criteria mogelijk invloed heeft op de planning en/of de aanpak van het project. Een van de belangrijkste criteria hierbij is de toepassingsmogelijkheid voor het Proof of Concept, waarbij er rekening gehouden moet worden met benodigde domeinkennis die het vooronderzoek en/of onderzoek kunnen uitbreiden.

Deze mogelijke implicaties zijn dan ook de aanleiding geweest tot een serie van gesprekken met de Blockchain expert en de bedrijfsbegeleider binnen Quintor, waarin er geprobeerd is een juiste toepassing te vinden die gerealiseerd kon worden binnen de beperkte tijd.

5.1 Omwenteling en implicaties

Uit deze gesprekken is voortgekomen dat de toepassing van het Proof of Concept los staat van de uitvoering van het onderzoek, aangezien de toepassing alleen maar dient als bewijs dat de gerealiseerde Blockchain kern functioneert. Dit heeft ertoe geleid dat er beslist is over veranderingen die impact hebben op de insteek van de opdracht zoals het origineel opgesteld is. Hieronder zijn kort de veranderingen weergegeven.

Generieke Blockchain

Er is besloten dat voor het Proof of Concept en het onderzoek de focus zal liggen op het creëren van een generieke Blockchain. Daarbij is ook het besluit genomen om de toepassing van het Proof of Concept bij het adviesrapport te betrekken. Zelf had ik hierbij mijn twijfels over de interpretatie van 'generiek'. Ik had hierbij het idee dat het een implementatie-eis is, en niet zozeer een type Blockchain dat te onderzoeken is.

Deze twijfel is dan ook aangekaart in de sprintreview, waarbij er besloten is dat deze eis inderdaad aan de implementatie gesteld is en niet aan het onderzoek. Dit heeft als gevolg dat de term 'generiek' niet voorkomt in het onderzoek en het niet wenselijk om een dergelijk aspect als criteria voor de selectie van de implementaties te gebruiken.

Criteria

Er is besloten dat de criteria behandeld zullen worden in het onderzoek, waardoor de grootte van het onderzoek zal toenemen. De toepassingsmogelijkheid zal hierbij geadviseerd worden in het adviesrapport. Hieruit zal beslist worden welke toepassing gerealiseerd kan worden binnen de beperkte tijd.

6 | Vooronderzoek en resultaten

In dit hoofdstuk wordt er een introductie gegeven in het Blockchain domein. Deze kennis is benodigd om het onderzoek uit te voeren en om het Proof of Concept te realiseren. Daarnaast zal deze kennis helpen om de uitvoering van de opdracht te begrijpen. Het vooronderzoek dient tevens om overeenstemming te krijgen met de opdrachtgever over de richting van het onderzoek. Zoals verteld in de aanpak zijn er weinig eisen gesteld aan de uitvoering en toepassing van de afstudeeropdracht, waardoor het wenselijk is om een gezamenlijke overeenstemming te krijgen van wat mogelijk is met het onderzoek.

Het vooronderzoek betreft kwalitatief-, exploratief onderzoek dat uitgevoerd wordt door middel van deskresearch. De reden dat ik hiervoor heb gekozen is omdat het Blockchain domein nieuw voor mij is en ik dus ook niet welke specifieke kennis ik nodig heb om de vragen te beantwoorden. De bronnen die ik gebruikt heb zijn zowel informeel als formeel waarbij er veel informatie afkomstig is uit het Bitcoin protocol, zoals beschreven door Nakamoto (2008). Dit is een van de meest gedocumenteerde Blockchain implementaties die publiekelijk in te zien is. Om de technische kennis te versterken voor de realisatie van het Proof of Concept is er een Coursera course gevolgd, Bitcoin and Cryptocurrency Technologies, waarin het Bitcoin protocol uitgelegd wordt. Dit is gevolgd omdat de beschrijving van het Bitcoin protocol niet meer toereikend is naar de huidige staat van de implementatie.

Er wordt ingegaan op de basis van Blockchain technologie waarna er gekeken wordt naar de mogelijke toepassingen. Vervolgens komt de architectuur van een Blockchain aan bod, waarbij de vraag "Uit welke componenten bestaat een Blockchain implementatie?" wordt behandeld. Om een afbakening te maken voor het onderzoek wordt er gekeken naar wat de onderdelen Distributed Network en Identity Management bevatten. Concreet staan de vragen die behandeld worden in het vooronderzoek hieronder weergegeven.

1. Wat is Blockchain technologie?
2. Waarvoor wordt Blockchain technologie gebruikt?
3. Uit welke onderdelen bestaat een Blockchain?
4. Waaruit bestaat het onderdeel Distributed Network binnen Blockchain technologie?
5. Waaruit bestaat het onderdeel Identity Management binnen Blockchain technologie?

6.1 Blockchain

In dit hoofdstuk wordt de vraag "Wat is Blockchain technologie" behandeld. Het betreft het verzamelen van kennis over de basis van het Blockchain begrip waarbij er ingegaan wordt op wat Blockchain is en welke eigenschappen het heeft. Door het beantwoorden van deze vraag wordt er een definitie vastgesteld van Blockchain technologie die gebruikt wordt in het gehele verslag.

Deze vraag heb ik opgesteld om een definitie van een Blockchain te stellen waarbij er onderzocht zal worden welke attributen deel uitmaken van de techniek. Er is allereerst gezocht naar een definitie van een Blockchain. Dit heb ik gedaan door te zoeken in informele databronnen zoals Wikipedia en de Bitcoin wiki. Uiteindelijk heb ik uit deze bronnen een definitie proberen te halen:

"Een blockchain is een gedistribueerde database die bestaat uit een keten van in de computer of op internet vastgelegde en samengevoegde gegevens genaamd blocks."

In zekere mate is dit correct maar het beschrijft niet compleet het meest vooraanstaande aspect van Blockchain, namelijk dat het gedecentraliseerd opereert. In veel van de definities die ik gevonden heb, wordt Blockchain vergeleken met een grootboek. Omdat ik dit een duidelijke analogie vindt heb ik dit ook gebruikt om het concept van decentralisatie uit te leggen. Hieronder is deze analogie te vinden:

"Het grootboek is in handen van één organisatie waarin transacties van of naar de organisatie vastgelegd worden. Dit betekent dat er een centrale autoriteit is die kan bepalen of er überhaupt wel transacties plaatsvinden, of erger, het systeem buiten gebruik kan stellen. Daarnaast is de centrale autoriteit ook in staat misbruik te maken hiervan door bijvoorbeeld valse transacties te registreren. Dit brengt een risico met zich mee die blockchain technologie oplost door het grootboek te verspreiden over een netwerk dat ervoor zorgt dat deze centrale autoriteit niet meer nodig is, aangezien elke participant in het netwerk verantwoordelijk is voor het valideren van een transactie."

Na een van de kernaspecten van de Blockchain technologie gevonden te hebben, namelijk decentralisatie, begon ik mij af te vragen of er een studie bestond die het totaalplaatje van Blockchain beschreef, inclusief de kernaspecten. Uiteindelijk ben ik via Google Scholar op de studie van Zheng, Xie, Dai, Chen en Wang (2017) terechtgekomen die een overzicht geeft van de Blockchain techniek. De paper is gepubliceerd door IEEE Xplore voor een Big Data congres, en heeft 82 citaties. Dit leek mij daarom een degelijke bron om de benodigde informatie uit te halen.

In de studie wordt gesteld dat er vier eigenschappen zijn die een Blockchain definiëren:

Decentralisatie In traditionele gecentraliseerde transactie systemen wordt iedere transactie gevalideerd door een centrale vertrouwde organisatie (e.g. banken), waardoor er een bottleneck gecreëerd wordt door de transacties te verwerken door centrale informatiesystemen. In contrast daarmee is een derde partij niet meer nodig in blockchain systemen. Consensus algoritmes zorgen ervoor dat data consistent is binnen het netwerk.

Persistentie Transacties kunnen snel gevalideerd worden en invalide transacties zullen niet toegelaten worden. Het is bijna onmogelijk om te transacties verwijderen of ongedaan te maken als ze zijn opgenomen in de blockchain.

Anonimiteit Elke gebruiker van het systeem kan interacteren zonder zijn ware identiteit kenbaar te maken.

Controleerbaarheid In bitcoin wordt de balans van een gebruiker opgeslagen door gebruik te maken van het Unspent Transaction Output (UTXO) model. Elke transactie refereert naar eerdere unspent transacties. Wanneer de huidige transactie is opgenomen in de blockchain, zal de staat van alle gerefereerde transacties verandert worden van "unspent" naar "spent". Hierdoor zijn transacties makkelijk te valideren en te traceren.

6.1.1 Conclusie

De uiteindelijke conclusie van de vraag "**Wat is Blockchain technologie?**" is dan ook als volgt:

Blockchain technologie gaat uit van vier kernprincipes: **decentralisatie, persistentie, anonimiteit en controleerbaarheid**. Een Blockchain is dan ook een gedistribueerde database die bestaat uit een keten van in computer of op internet vastgelegde en samengevoegde gegevens genaamd blocks. Doordat ieder van deze blocks met elkaar verbonden zijn is het niet mogelijk om informatie die reeds in het systeem is opgenomen aan te passen (persistentie). Het werkt decentraal doordat er geen één centrale vertrouwde organisatie bestaat die transacties valideren, maar alle participanten in het netwerk deel uitmaken van het validatie proces (decentralisatie). Dit gebeurt allemaal zonder je eigen identiteit bloot te stellen in het systeem (anonimiteit). Doordat alle transacties vastgelegd worden in het systeem en doorgaans publiekelijk in te zien zijn, en dus ook niet aan te passen zijn, is er de mogelijkheid om transacties makkelijk te traceren (controleerbaarheid).

6.2 Toepassing

In dit hoofdstuk wordt de vraag "Waarvoor wordt Blockchain technologie gebruikt?" beantwoord. Het antwoord op deze vraag dient om de opdrachtgever te informeren in wat er mogelijk is met Blockchain technologie om zo een toepassing te kiezen voor het te ontwikkelen Proof of Concept.

Er is in het bijzonder aandacht geschonken aan Blockchain als development platform aangezien de opdrachtomschrijving, bijlage I, spreekt over de realisatie van het onderdeel Smart Contract. Uitleg over het onderdeel Smart Contracts is beperkt gebleven aangezien het buiten de scope van de opdracht valt.

Deze vraag is opgesteld naar aanleiding van de omwenteling van de opdracht, zoals besproken in 5.1. Het betreft het opstellen van een lijst van toepassingsmogelijkheden die geadviseerd kunnen worden aan Quintor, waarbij er rekening gehouden wordt met de gelimiteerde tijd. Aangezien deze vraag al extra gesteld wordt voor het adviseren van een toepassing die volgens de beschrijving van de originele opdracht gegeven had moeten worden door Quintor, heb ik bij het beantwoorden van deze vraag een limiet gesteld van twee toepassingen. Dit is mede gedaan omdat de toepassing van Blockchain technologie steeds breder wordt.

Blockchain technologie wordt steeds vaker toegepast voor het opzetten van een gedecentraliseerd systeem. Aangezien de bekendste toepassing van Blockchain technologie een financieel systeem is, namelijk Bitcoin, wordt het vaak gezien als technologie die specifiek bedoeld is om financiële diensten te ondersteunen. In de literatuur wordt er echter veel geëxperimenteerd en gespeculeerd over andere mogelijke toepassingen van Blockchain technologie. Tijdens mijn zoektocht naar studies over de toepassingen van Blockchain buiten de financiële markt zijn er twee interessante studies gevonden.

6.2.1 Politiek

Een van deze studies is gedaan door Atzori (2015), waarin een interessant idee als toepassing van Blockchain technologie gepresenteerd wordt. Atzori spreekt namelijk over "*Decentralized Governance*", een idee om de autoriteit van de staat over te zetten naar het Blockchain domein. Alhoewel zij met de onderstaande passage aangeeft dat er geen academische ideeën zijn voor mogelijke Blockchain gebaseerde modellen, is er wel een verzameling gemaakt van principes in de huidige politiek die aangekaart kunnen worden in een Blockchain gebaseerd model.

"To date, a comprehensive discussion of possible blockchain-based models of governance does not yet exist at academic level. Since a coherent and consistent body of thought on this subject is missing, for the purpose of our paper we have collected information from a number of sources as accurately as possible, though probably in a non-exhaustive manner."

- Atzori (2015, p.7)

Met de passage *"For the first time in history, citizens can now reach consensus and coordination at global through cryptographically verified peer-to-peer procedures, without the intermediation of a third party."*, is ter interpretatie gesteld dat het mogelijk is om bijvoorbeeld het stemproces te veranderen, waarbij zoiets als een eerste of tweede kamer overbodig zou worden.

6.2.2 Ontwikkelpatform

Een andere insteek voor het gebruik van Blockchain, waar ik zelf als developer meer ervaring mee heb, is de mogelijkheid om Blockchain te gebruiken als ontwikkelplatform. Het leek mij toepasselijk om deze toepassing te analyseren gezien in deze toepassing Blockchain als bouwsteen functioneert. Dit betekent dat het los staat van een toepassing en dat er verschillende toepassingen mee gerealiseerd kunnen worden. Om deze reden leek het mij dan ook een goede reden om het te adviseren als mogelijke "toepassing" voor het Proof of Concept. Blockchains als Ethereum, EOS en HyperLedger bieden hun functionaliteit aan als development platform. Het stelt ontwikkelaars in staat om hun eigen toepassingen te realiseren, zogenaamde Distributed Applications (DApp).

Listing 6.1: Smart contract voor "The Greeter" geschreven in Solidity, zoals gepresenteerd in een tutorial voor Smart Contracts op het Ethereum netwerk (Ethereum, 2017).

```
contract Mortal {
    /* Define variable owner of the type address */
    address owner;

    /* This function is executed at initialization and sets the owner of the
       contract */
    function Mortal() { owner = msg.sender; }

    /* Function to recover the funds on the contract */
    function kill() { if (msg.sender == owner) selfdestruct(owner); }
}

contract Greeter is Mortal {
    /* Define variable greeting of the type string */
    string greeting;

    /* This runs when the contract is executed */
    function Greeter(string _greeting) public {
        greeting = _greeting;
    }

    /* Main function */
    function greet() constant returns (string) {
        return greeting;
    }
}
```

Door het gebruik van Smart Contracts, te zien in fig. 6.1, is het mogelijk om functionaliteit bij transacties te voegen om extra handelingen, die niet gerelateerd zijn tot de kern van een Blockchain implementatie, uit te voeren. Een bekende, controversiële toepassing die hiermee gerealiseerd is, is het spel genaamd **CryptoKitties**. Het is een spel dat gebruikt maakt van het Ethereum platform, bestaand uit verzamelbare en fokbare digitale katten. De uitwisseling en het fokken van CryptoKitties wordt vastgelegd in het Ethereum netwerk door middel van Smart Contracts. Wanneer twee CryptoKitties gefokt worden, wordt het uiterlijk en de eigenschappen van hun nageslacht bepaald door het 256-bits genoom van elke ouder en een toeval element, wat leidt tot 4 miljard mogelijke genetische variaties (Zen, A., 2017).

6.2.3 Conclusie

Het antwoord op de vraag “**Waarvoor wordt Blockchain technologie gebruikt?**” is dan ook als volgt:

Blockchain technologie wordt toegepast in veel domeinen. Tijdens de analyse van de vraag zijn er met name twee toepassingen bekeken, **politiek / maatschappij** en als **ontwikkelplatform**. In de politiek / maatschappij is de mogelijkheid voor het toepassen van Blockchain talloos, alleen over de daadwerkelijke toepassingen hebben de academi het hoofd nog niet gebogen. Wel zijn er principes gevonden binnen de politiek die aansluiten bij de kernprincipes van Blockchain. Daarnaast is er gekeken naar de mogelijkheid om Blockchain te gebruiken als ontwikkelplatform, waarbij er kort ingegaan is over Smart Contracts.

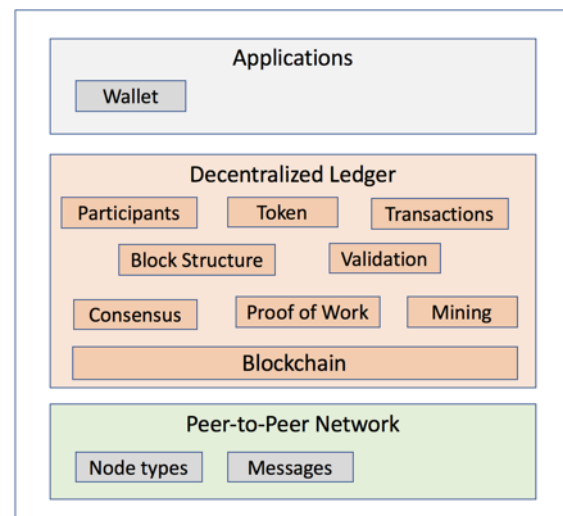
6.3 Architectuur

In dit hoofdstuk wordt de vraag "Uit welke componenten bestaat een Blockchain implementatie?" behandeld. Het antwoord op deze vraag dient om een duidelijk beeld te scheppen welke componenten betrekking hebben op de onderdelen Distributed Network en Identity Management en tevens gebruikt zal worden als afstemming met zowel de opdrachtgever als de medeafstudeerder, Kevin Bos.

In fig. 6.1 is een overzicht weergegeven van de onderdelen en componenten waaruit een Blockchain bestaat. In de applicatie laag is de wallet te vinden die een gebruiker van de Blockchain doorgaans gebruikt om transacties te verrichten. De onderliggende functionaliteit van de wallet doet niets meer als het bijhouden van public- en private keys van de gebruiker waarop de nog niet uitgegeven tokens (cryptocurrency, contracten, diensten) geregistreerd staan.

De Decentralized Ledger is de kern van de technologie en zorgt ervoor dat de globale blockchain consistent en fraudebestendig blijft. De fundamentele structuur achter de gehele technologie is de blockchain, waar transacties gegroepeerd worden in blokken en elk blok cryptografisch verbonden wordt met het vorige blok. Een transactie is een vorm van uitwisseling van tokens tussen deelnemers, ook wel nodes genoemd, van het systeem. Voordat transacties als valide worden beschouwd, ondergaan ze een validatie proces die uitgevoerd wordt door alle nodes in het systeem. Het proces van het groeperen van transacties in een blok dat toegevoegd wordt aan het einde van de blockchain wordt ook wel minen genoemd. Om er zeker van te zijn dat er overeenstemming is onder alle deelnemers over welke blockchain legitiem is, wordt er gebruik gemaakt van een Proof-of-Work algoritme tijdens het mining proces om te bepalen welke ketting de meeste inspanning vereist.

Het laatste component is het peer-to-peer netwerk, waarin verschillende node types gedefinieerd zijn. Zo heb je bijvoorbeeld de validatie node die transacties valideert en een mining node die het mining proces uitvoert. Om de Decentralized Ledger bij te werken en te onderhouden communiceren de nodes met elkaar door middel van het versturen van berichten.



Figuur 6.1: Blockchain architectuur

6.4 Gedistribueerd netwerk

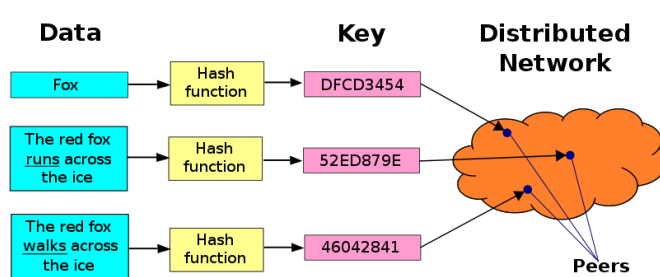
In dit hoofdstuk wordt de vraag "Waaruit bestaat het onderdeel gedistribueerd netwerk binnen Blockchain technologie?" behandeld. Bij deze vraag wordt er gekeken naar de geïdentificeerde onderdelen uit hoofdstuk 6.3. Er wordt een korte introductie gegeven in peer-to-peer netwerken en waarom het een belangrijk onderdeel is bij het realiseren van de eigenschappen, behandeld in hoofdstuk 6.1, van een Blockchain implementatie. Het antwoord op deze vraag zal helpen bij het selecteren van zoektermen die gebruikt worden om inventarisatie te doen op de onderdelen die het Distributed Network omvat.

Het onderdeel Distributed Network bestaat uit het verspreiden, uitbreiden en het behalen van consensus over de staat van de Blockchain tussen de deelnemers aan het netwerk. Om dit te doen wordt er gebruik gemaakt van een Peer-to-Peer (P2P) implementatie waarbij het mogelijk is om een lokale versie van de ketting aan te bieden aan andere nodes binnen het P2P netwerk, om zo de huidige chain up-to-date te houden met wijzigingen die gedaan zijn door de verschillende verbonden nodes. Dit leidt tot een complex probleem dat beschreven wordt als het Byzantine Generals Problem (Lamport et al., 1982), wat beschrijft aan de hand van een abstract voorbeeld dat het essentieel is voor een betrouwbaar computersysteem om te kunnen gaan met fouten die optreden in een of meer van de componenten, waardoor het kan voorkomen dat er conflicterende informatie verstuurd wordt naar de andere componenten van het systeem.

Peer-to-Peer

De term P2P betekend dat alle computers die deel uit maken van het netwerk, peers van elkaar zijn, gelijk aan elkaar zijn, er geen speciale "nodes" zijn en dat alle deelnemers in het netwerk de last delen van het leveren van netwerkdiensten (Antonopoulos, 2014, p.171). Het is een techniek die cruciaal is voor Blockchain en de doelen die het probeert te behalen. P2P systemen verdelen namelijk de kosten om data te delen – opslag voor bestanden en bandbreedte voor het versturen van de bestanden – over de deelnemers van het netwerk, waardoor applicaties kunnen schalen zonder krachtige, dure servers (Bawa et al., 2003).

Een van de bekendste toepassingen van een peer-to-peer netwerk is het creëren van een gedecentraliseerd file-sharing protocol. Implementaties hiervan zijn BitTorrent, LimeWire en Gnutella. Om een bestand te distribueren wordt het opgesplitst in delen, waarbij er een hash gecreëerd wordt voor elk deel. Wanneer een andere deelnemer van het netwerk een deel ontvangt wordt er gekeken aan de hand van de hash of het onderdeel geen fouten bevat. Bestanden worden geregistreerd in het netwerk door het opnemen van de hashes in een zogenaamde *tracker* die gebruik maakt van een Distributed Hash Table (DHT). Een voorbeeld van een DHT is te zien in fig. 6.2.



Figuur 6.2: Door het vertalen van data naar een cryptografische sleutel is het mogelijk om aan de hand van de sleutel de data op te vragen aan peers die de data bezitten.

Consensus

Consensus is een dynamische manier van het behalen van overeenstemming in een groep. In blockchain implementaties wordt het gebruikt om overeenstemming te behalen over de staat van het netwerk en de volgorde waarin transacties gedaan zijn. Met het consensus algoritme wordt er een zekere mate van veiligheid gewaarborgd, waardoor het voor een kwaadwillende

deelnemer (bijna) onmogelijk dient te zijn om het netwerk te beïnvloeden. Het kan voorkomen dat een kwaadwillende deelnemer probeert het netwerk te beïnvloeden waardoor er tegenstrijdige consensus kan optreden en een fork ontstaat in het netwerk.

Fork

Een fork is een splitsing in het netwerk die veroorzaakt is door een verandering in het protocol of door het toedoen van kwaadwillende deelnemer(s). Er zijn hiervoor twee categorieën forks, een Hard Fork en een Soft Fork.

Soft fork is een verandering in het netwerk die terugwaartse compatibiliteit heeft met eerdere versies van het protocol. Als voorbeeld kan er voor gekozen worden dat in plaats van blocks een limiet hebben van 1MB, de regel aangepast wordt zodat blocks een grootte van 500K moeten hebben. Als een Soft Fork verkeerd gaat is het nog steeds mogelijk dat er een Hard Fork optreedt (Castor, A. , 2017, Soft Fork).

Hard fork is een protocol update waarbij een nieuwe regel geïntroduceerd wordt, waardoor het netwerk geen compatibiliteit heeft met oudere versies. Dit zorgt ervoor dat deelnemers in het netwerk die een oudere versie hebben, de nieuwe transacties als invalide beschouwen. Een voorbeeld van een regel waarbij een Hard Fork ontstaat is bijvoorbeeld het ophogen van de block grootte naar 2MB in plaats van 1MB (Castor, A. , 2017, Hard Fork).

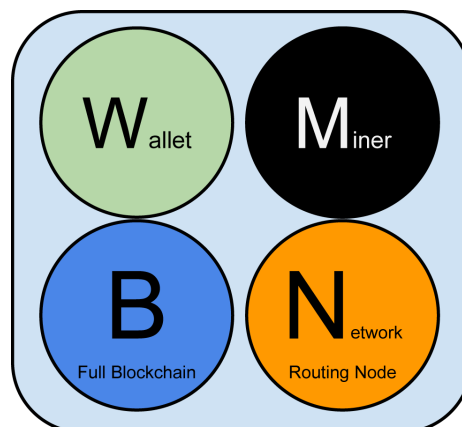
Nodes

Alhoewel de structuur van een Blockchain dezelfde structuur afdwingt voor de nodes in het netwerk, kunnen zij een verschillende rol spelen. Alle nodes binnen het netwerk valideren, verspreiden en ontdekken en onderhouden connecties met andere nodes binnen het netwerk. In fig. 6.3 is te zien welke services een full node in het Bitcoin netwerk aanbiedt.

Een **full node** is een collectie van functies, namelijk routing, de blockchain database, het mining proces en wallet services en bevat een gehele kopie van de actuele blockchain. Een **wallet (node)** is een deelnemer in het netwerk die een subset van de gehele blockchain bevat om transacties te versturen, verifiëren en ontvangen. De **mining nodes** concurreren voor het creëren van een nieuw block door het uitvoeren van het Proof-of-Work algoritme.

Alle nodes binnen het netwerk bieden gelijke diensten aan en kunnen gebruik maken van dezelfde diensten terwijl ze samenwerken door middel van een consensus protocol.

De verschillende services binnen het netwerk en de node types die hieraan meewerken is dan ook een architecturale keuze over de indeling van het P2P netwerk.



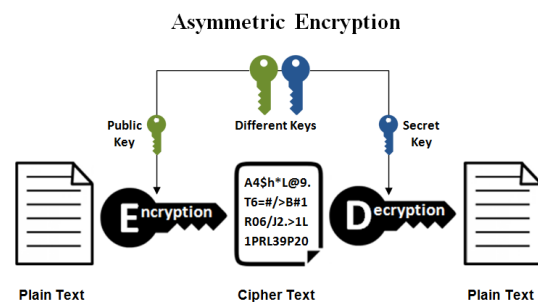
Figuur 6.3: Een bitcoin netwerk node die alle functies bevat: wallet, mining, blockchain database en netwerk routing, (Antonopoulos, 2014, p. 172).

6.5 Identiteit

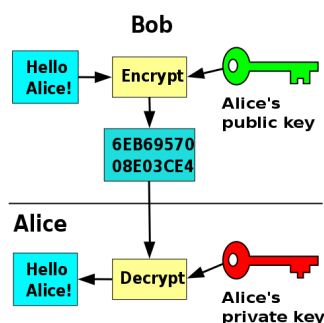
In dit hoofdstuk wordt de vraag “Waaruit bestaat het onderdeel Identity Management binnen Blockchain technologie?” behandeld. Zoals beschreven in hoofdstuk 6.1 zijn anonimiteit en controleerbaarheid belangrijke eigenschappen van een Blockchain implementatie. Allereerst zal er beschreven worden wat identiteit inhoud binnen een Blockchain en welke mogelijke vormen van management er zijn. Het antwoord op deze vraag wordt gebruikt om zoektermen op te stellen en een afbakening te creëren voor de te onderzoeken protocollen.

Identificatie wordt doorgaans gedaan aan de hand van de public key van een gebruiker. Public key cryptografie is een essentieel onderdeel van het Bitcoin protocol en wordt gebruikt voor verschillende doeleinden om de integriteit van berichten die verstuurd worden te waarborgen. Public key cryptografie bestaat uit twee onderdelen:

- **Public key**
Een key die verstuurd wordt om aan te tonen dat een bericht daadwerkelijk verstuurd is door de maker van het bericht, door het ondertekenen van het bericht.
- **Private key**
Een key die geheim wordt gehouden en gebruikt wordt om te valideren dat een public key valide is.



Figuur 6.4: Asymmetrische encryptie door middel van Public key cryptografie zoals in gebruik bij het Bitcoin protocol.



Figuur 6.5: Het gebruik van asymmetrische encryptie om berichten die verstuurd worden op het netwerk te versleutelen.

In het Bitcoin protocol is elke coin terug te leiden naar een eigenaar waarbij er gebruik gemaakt wordt van de public key. Wanneer er coins van eigenaar wisselen worden de coins overgezet naar de public key van de ontvanger en wordt het getekend met de private key van de verstuurder. Dit zorgt ervoor dat iedereen in het netwerk weet dat het bericht authentiek is (Bitcoin Wiki, 2010, "How bitcoin works").

6.5.1 Autorisatie

Zheng et al. (2017) deelt Blockchain implementaties op in drie categorieën, waarin de zichtbaarheid en participatie in het consensus proces gelimiteerd.

Public In een public Blockchain zijn alle transacties publiekelijk inzichtbaar en iedereen in het netwerk maakt onderdeel uit van het consensus proces. Dit wordt ook wel gezien als een permissionless Blockchain.

Consortium In een consortium Blockchain is er een groep van vooraf geselecteerde nodes die deel uitmaken van het consensus proces. De consortium Blockchain wordt meestal gebruikt door meerdere organisaties en is gedeeltelijk gedecentraliseerd. Omdat bepaalde nodes geïdentificeerd dienen te worden wordt dit type Blockchain gezien als een permissioned Blockchain.

Private In een private Blockchain worden alleen nodes van een specifieke organisatie toegelaten tot het consensus proces. Het wordt ook wel als een centraal netwerk gezien omdat het in volledige controle is van één organisatie. Omdat het hier gaat om volledige restrictie tot het Blockchain netwerk wordt dit type Blockchain gezien als een permissioned Blockchain.

In een consortium en private Blockchain dient de gebruiker zich te identificeren aan de hand van een identiteit. Bitcoin maakt gebruik van public- en private keys om de gebruiker te identificeren. Dit hanteert in zekere mate een permissie model waarbij de autorisatie van een gebruiker vastgelegd wordt aan de hand van de identificatie (i.e. de public key) die het netwerk gebruikt.

Privacy

Een van de doelen van Blockchain is totale anonimiteit, alleen zijn er een aantal problemen die volledige anonimiteit tegengaan. Om de terminologie duidelijk te maken wordt hieronder het verschil tussen pseudoniem en anoniem uitgelegd aan de hand van voorbeelden vanuit het Bitcoin protocol.

"Anonymity is the state of being not identifiable within a set of subjects, the anonymity set."
Pfitzmann en Köhntopp (2001).

Pseudoniem Een pseudoniem is een referentie naar je ware identiteit. Een voorbeeld hiervan is het burgerservicenummer (BSN). Door het geven van je BSN is niet direct je ware identiteit terug te leiden.

Anoniem Wanneer je anoniem bent, ben je niet meer te identificeren binnen een set van soortgelijke identiteiten. Een voorbeeld hiervan is anoniem bellen. Hierbij wordt er gebruik gemaakt van het maskeren van het pseudoniem, namelijk het telefoonnummer.

In feite komt het neer op de identificatie van de handelingen die gedaan worden door de gebruiker. Hiervoor bestaat een term, unlinkability, dat beschrijft wanneer een gebruiker meerdere keren interacteert met het systeem deze handelingen niet terug te leiden zijn naar elkaar.

Privacy in Bitcoin

In Bitcoin is de identiteit van de gebruiker, de public key(s), een pseudoniem. Aangezien Bitcoin een permissionless model heeft, waarbij iedereen elke transactie kan inzien, is het mogelijk om transacties die gedaan zijn door dezelfde public key terug te leiden naar elkaar.

Een analyse model geïntroduceerd door Reid en Harrigan (2013) maakt gebruik van twee modellen van het Bitcoin netwerk, waarbij er een model gemaakt wordt voor bitcoins tussen transacties, en bitcoins tussen gebruikers. Door het gebruik van de voorgestelde analyse is het mogelijk om meerdere public keys met elkaar te associëren.

6.6 Obstakels

In het vooronderzoek is er weinig gevonden over Identity Management en zelf wist ik dan ook niet goed wat dit onderdeel voor functionaliteiten bevat. In eerste instantie is er gekeken naar de verschillende types van Blockchain, waarbij gebruikers van het systeem autorisatie hebben tot bepaalde acties. Om een beter beeld te schetsen en de afbakening van het onderdeel compleet te maken voor het onderzoek is er besloten om een gesprek te houden met de Blockchain Expert.

Uit dit gesprek is naar voren gekomen dat het Identity Management gedeelte gaat over hoe de Blockchain implementatie met public keys (de identiteit van een gebruiker) omgaat. Als tip werd er gegeven om te kijken naar de wallet software indien beschikbaar. Dit is software die de public- en private key beheert voor een gebruiker. Daarnaast is er ook een tip gegeven over het onderdeel Distributed Network. Om een goed beeld te krijgen van de aanvallen waar het netwerk tegen bestand is, is het handig om een threat model op te stellen.

6.7 Conclusie

Naar aanleiding van de resultaten uit het vooronderzoek zijn er keuzes gemaakt die zich reflecteren in het onderzoek. Hieronder is per vraag beschreven over de implicaties die de resultaten gaven tegenover het onderzoek.

Uit de vraag “Waaruit bestaat het onderdeel Distributed Network binnen Blockchain technologie?” is gebleken dat het consensus proces invloed heeft op de structuur van het netwerk. Het soort consensus zal dan ook gebruikt worden om de verschillende type Distributed Networks te onderscheiden. Ook zullen de verschillende type van nodes onderzocht worden om te identificeren welke bijdrage een bepaald type node levert om het netwerk in stand te houden.

In de resultaten van de vraag “Waaruit bestaat het onderdeel Identity Management binnen Blockchain technologie?” is geïdentificeerd dat er twee manieren zijn waarop de privacy van de gebruiker gewaarborgd wordt, namelijk of het een permissionless of permissioned Blockchain is en de identificatie van de gebruiker binnen het netwerk. Er wordt aangegeven dat het niveau van privacy en autorisatie ligt aan de categorie van waar de Blockchain deel van uitmaakt.

1. Zichtbaarheid van acties die de gebruiker onderneemt op de Blockchain.
2. Autorisaties voor acties die de gebruiker wilt ondernemen op de Blockchain.

7 | Onderzoek

In dit hoofdstuk worden de werkzaamheden met betrekking tot het uitvoeren van het onderzoek beschreven. De aanleiding voor het onderzoek is te vinden in hoofdstuk 4, waarin wordt beschreven waarom dit onderzoek meerwaarde heeft binnen de opdracht.

Het onderzoek dient voor het opstellen van het adviesrapport waarin protocollen worden gepresenteerd aan Quintor die mogelijk geïmplementeerd kunnen worden tijdens de realisatie van het Proof-of-Concept. Het betreft exploratief onderzoek waarin case-study gebruikt wordt om een gedetailleerde omschrijving van de onderdelen Identity Management en Distributed Network op te stellen van Blockchain implementaties die geselecteerd zijn in hoofdstuk 7.2. De kennis die hiermee wordt opgebouwd kan eventueel gebruikt worden in vervolgonderzoek.

In het onderzoek staat de onderstaande hoofdvraag centraal:

Welke protocol implementaties kunnen toegepast worden om de onderdelen Distributed Network en Identity Management te realiseren voor een Blockchain implementatie?

Omdat de hoofdvraag te groot is om in een keer te beantwoorden is het opgesplitst in de volgende deelvragen:

1. "Welke soorten gedistribueerde netwerken worden er gebruikt?"
2. "Hoe werken de gedistribueerde netwerken en tegen welke gevaren zijn ze bestendig?"
3. "Hoe wordt er omgegaan met de identiteit van de gebruiker?"

Op de volgende pagina's is per deelvraag behandeld wat de bijdrage van het antwoord oplevert aan de doelstelling, hoe de vraag beantwoord is en wordt er concreet de bevindingen besproken.

7.1 Opzet

Om te achterhalen welke protocol implementaties toegepast kunnen worden om de onderdelen Distributed Network en Identity Management te realiseren voer ik kwalitatief, exploratief onderzoek uit. Door het uitgevoerde vooronderzoek heb ik wel basiskennis over de onderdelen, maar weet ik nog niet de details. Om die reden heb ik er dan ook voor gekozen om het exploratief aan te pakken. Het gevaar hierbij is dat er teveel informatie geanalyseerd wordt waardoor het onderzoek te uitgebreid wordt.

7.1.1 Dataverzameling

De benodigde data zal ik verzamelen door het uitvoeren van deskresearch, waarbij ik de geselecteerde Blockchain protocol implementaties zal analyseren. De data die ik hiervoor nodig hebt komt vooral uit de whitepapers die beschikbaar zijn voor elke implementatie. Daarnaast worden er zo veel mogelijk academische bronnen gebruikt. Om deze, mogelijk aanvullende, academische bronnen te vinden zal ik gebruik maken van het schoolportaal en Google Scholar. Hierbij zal er gelet worden op de kwaliteit van de studie, door te achterhalen of het gebruikt wordt in andere studies en te kijken of het peer reviewed is.

7.1.2 Dataomschrijving

De Blockchain implementaties zullen in eerste instantie geselecteerd worden op de aanwezigheid van het onderdeel Identity Management. Hierbij zal ik kijken of de implementatie actief iets onderneemt voor bijvoorbeeld het verhogen van de privacy van de gebruiker. Daarnaast zal er gekeken worden naar de beschikbare hoeveelheid informatie.

7.1.3 Analysemethode

De methode die ik zal gebruiken voor het analyseren van de data is het uitvoeren van cumulatieve case-studies. Hierbij zullen er meerdere bronnen bekeken worden van een Blockchain implementatie waarna een aggregatie gemaakt wordt van de benodigde data voor het beantwoorden van de opgestelde vragen.

7.2 Selectie protocollen

Om het onderzoek binnen de beschikbare tijd te houden is er in overleg met de Blockchain Expert voor gekozen om een initiële selectie van de top 20 verhandelde cryptocurrencies te bekijken, waarna er een selectie van vier implementaties gemaakt wordt gebaseerd op de beschikbare informatie, het type consensus en hoe het omgaat met de identiteit van de gebruiker. Deze vier implementaties zullen vervolgens uitvoerig onderzocht en beschreven worden op de werking van de onderdelen Distributed Network en Identity Management.


























7.2.1 Coinmarketcap

De selectie van de top 20 verhandelde cryptocurrencies wordt gedaan aan de hand van de website Coinmarketcap. Hierop zijn meerdere overzichten te zien die te maken hebben met de handelsvolume

Een van de overzichten is het maandelijks handelsvolume zoals te zien in fig. 7.1. Deze lijst is gebruikt voor het selecteren van de initiële top 20 van cryptocurrencies. Door de architecturen achter de meest verhandelde cryptocurrencies te gebruiken wordt ervoor gezorgd dat er robuuste en volwassen implementaties bekeken worden.

Hard forks

Om te voorkomen dat er soortgelijke implementaties bekeken worden is ervoor gekozen om de hard forks niet mee te nemen in de initiële selectie. Een voorbeeld hiervan is Bitcoin Cash ten opzichte van Bitcoin. Alhoewel Bitcoin Cash een aantal veranderingen doorgemaakt heeft sinds de afsplitsing van het Bitcoin protocol, wordt het niet meegenomen omdat er in zekere mate overeenkomsten aanwezig zijn.

#	Name	Symbol	Volume (1d)	Volume (7d)	Volume (30d)
1	 Bitcoin	BTC	\$5,728,630,000	\$47,024,950,272	\$239,937,765,688
2	 Ethereum	ETH	\$1,653,700,000	\$13,056,842,240	\$83,063,570,688
3	 Tether	USDT	\$1,958,830,000	\$16,378,062,592	\$84,926,233,088
4	 Ripple	XRP	\$357,324,000	\$4,156,748,960	\$39,850,125,664
5	 Litecoin	LTC	\$1,285,160,000	\$6,746,461,952	\$26,736,620,416
6	 Bitcoin Cash	BCH	\$376,441,000	\$2,973,709,568	\$20,046,743,072
7	 Ethereum Cla...	ETC	\$646,646,000	\$5,532,372,672	\$15,749,806,368
8	 EOS	EOS	\$188,493,000	\$1,532,097,920	\$15,605,236,864
9	 Cardano	ADA	\$266,120,000	\$1,252,818,648	\$13,445,324,382
10	 NEO	NEO	\$123,156,000	\$1,072,350,408	\$9,084,051,800
11	 Qtum	QTUM	\$80,423,900	\$847,290,136	\$8,218,878,888
12	 TRON	TRX	\$172,623,000	\$1,071,585,984	\$6,733,278,384
13	 Status	SNT	\$29,635,700	\$199,291,636	\$6,500,340,332
14	 Stellar	XLM	\$42,479,700	\$422,246,356	\$4,661,325,352
15	 Huobi Token	HT	\$105,096,000	\$910,826,360	\$3,566,953,088
16	 ATMCoin	ATMC	\$50,360,800	\$537,747,432	\$3,060,712,860
17	 Dash	DASH	\$68,725,100	\$594,435,760	\$2,883,398,448
18	 VeChain	VEN	\$82,665,100	\$599,619,288	\$2,767,577,812
19	 Lisk	LSK	\$38,802,300	\$569,389,712	\$2,547,525,232
20	 Zcash	ZEC	\$52,432,700	\$433,916,424	\$2,477,751,772
21	 Monero	XMR	\$44,197,000	\$530,603,416	\$2,265,885,760
22	 Hshare	HSR	\$55,382,500	\$412,009,240	\$2,249,269,776
23	 ICON	ICX	\$39,998,600	\$223,640,806	\$2,201,559,444
24	 Nano	NANO	\$125,487,000	\$1,053,026,588	\$1,879,299,038
25	 Binance Coin	BNB	\$45,055,600	\$312,276,692	\$1,835,435,524

Figuur 7.1: Meest verhandelde cryptocurrencies in de maand februari zoals gepresenteerd op de website van Coinmarketcap.

7.2.2 Attributen

Om een selectie te maken tussen de top 20 verhandelde cryptocurrencies is er gekeken naar attributen die nader beschreven zijn in onderstaand tabel.

Tabel 7.1: Attributen opgesteld voor initiële selectie implementaties.

Identity Management	Of de implementatie actief iets onderneemt dat te maken heeft met Identity Management, e.g. het vergroten van de privacy van de gebruiker.
Whitepaper	Of de implementatie een technische whitepaper beschikbaar heeft.
Open-source	Of er een referentie implementatie open-source beschikbaar is voor het bestuderen van de code.
In circulatie sinds	Een indicatie van de volwassenheid van de implementatie.
DApps platform	Of het gebruikt kan worden als development platform. Hierbij zal er een zekere mate van modulariteit nodig zijn in de broncode.
Consensus	Welk consensus algoritme gebruikt wordt. Dit is van invloed op de werking van het onderdeel Distributed Network.

Het doel van de attributen is een indicatie te krijgen over de hoeveelheid documentatie die een implementatie beschikbaar heeft. Dit is dan ook de doorslaggevende factor geweest bij het selecteren van vier implementaties die nader onderzocht zullen worden.

7.2.3 Totstandkoming

Hieronder wordt kort beschreven hoe de inventarisatie van de attributen gedaan is.

Identity Management Om vast te stellen of een implementatie iets onderneemt in de vorm van Identity Management wordt er gebruik gemaakt van bestaande literatuur over de implementatie. Door middel van het scannend lezen van de beschikbare literatuur wordt er vastgesteld of er beschrijvingen zijn van de identiteit binnen de Blockchain implementatie en hoe dit tot stand is gekomen.

Whitepaper Bijna elke Blockchain implementatie heeft een website waarin de functionaliteiten gepresenteerd worden voor de mogelijke gebruiker. Om erachter te komen of er een whitepaper beschikbaar is, is een scan van de website voldoende.

Open-source Om na te gaan of een implementatie open-source is wordt er gezocht op Github en Bitbucket op de aanwezigheid van de organisatie en/of protocol naam.

In circulatie sinds Om de circulatiedatum te achterhalen is gebruik gemaakt van Wikipedia. Hierbij is een schatting van de datum waarop de Blockchain implementatie actief is geworden al voldoende.

DApps platform Om na te gaan of de implementatie de ontwikkeling van gedistribueerde applicaties ondersteund is er gezocht naar development tutorials op de websites van de Blockchain implementatie.

Consensus Het type consensus dat gebruikt wordt is tevens te achterhalen uit de beschikbare documentatie en wordt achterhaald door scannend te lezen.

7.2.4 Resultaat

Aan de hand van deze attributen is een lijst opgesteld, te zien in de bijlage: “Bekeken implementaties uit de initiële selectie met de onderzochte attributen.”, waarin de initiële selectie te vinden is met bijbehorende attributen van de implementatie. Over sommige implementaties zoals VeChain is weinig informatie gevonden waardoor ze direct afvallen. Aan de hand van deze attributen zijn de volgende protocollen geselecteerd.

Cardano is een Blockchain protocol waarin onderzoek centraal staat. Het beweert dan ook het eerste blockchain platform te zijn die ontstaan is uit een filosofisch en onderzoek gedreven aanpak. De implementatie van het protocol is volledig open-source en er is een technische whitepaper beschikbaar. Daarnaast is er ook een platform om je eigen applicaties op het netwerk te creëren.

Monero is een implementatie die beweert dat de gebruiker volledig ontraceerbaar is. Het is net zoals Cardano een volledige open-source implementatie en maakt gebruik van egalitair Proof of Work. Daarnaast heeft het protocol een technische whitepaper.

Bitcoin is het originele protocol waarin de Blockchain technologie gerealiseerd is. Door de grote hoeveelheid onderzoek die gedaan is naar Bitcoin is er een overvloed van informatie, waarin niet alleen informatie over Bitcoin gegeven wordt maar ook over het Blockchain domein. De implementatie van het protocol is wederom volledig open-source en er is een technische whitepaper beschikbaar.

EOS is een relatief nieuw protocol die zojuist een test netwerk gelanceerd heeft. Ook deze implementatie is beschreven in een whitepaper, is volledig open-source en kan gebruikt worden als platform om applicaties op te ontwikkelen. Consensus binnen het protocol wordt bereikt door Delegated Proof of Stake.

7.2.5 Afwijking

In de originele opdrachtomschrijving is aangegeven dat er drie implementaties bekeken zou worden. In deze selectie is dat geëscaleerd tot vier implementaties. Elk van deze implementaties brengt een unieke architectuur met zich mee op het gebied van de onderdelen Identity Management of Distributed Network. Hierbij is Bitcoin een bijkomstigheid omdat er zoveel informatie over beschikbaar is. Dit zorgt er uiteraard wel voor dat het onderzoek uitgebreid wordt.

7.3 Soorten netwerken

In dit hoofdstuk wordt de vraag “Welke soorten gedistribueerde netwerken worden er gebruikt?” behandeld. Het doel van de vraag is om de architectuurkeuzes op het gebied van het Distributed Network onderdeel op te stellen, en waar mogelijk is de implicaties van de keuze tegenover het Identity Management onderdeel.

7.3.1 Aanpak

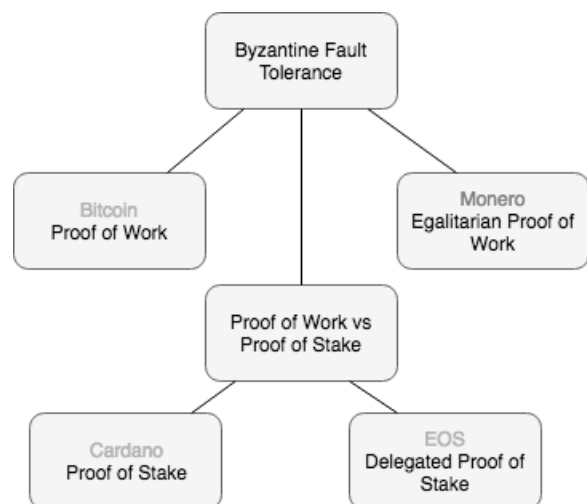
In fig. 7.2 is te zien welke (globale) termen er gebruikt zijn om de benodigde informatie te vinden. In de meeste gevallen is de beschikbare whitepaper van de implementatie voldoende geweest om de werking van het consensus te beschrijven. Hieronder is de werkwijze en denkwijze uitgeschreven per individueel onderdeel.

Byzantine Fault Tolerance

Om deze vraag te beantwoorden is allereerst gezocht naar achtergrondinformatie over consensus en wat het doel ervan is. In het vooronderzoek is er informatie gevonden over het Byzantine Generals Problem (Lamport et al., 1982), waarin, vertaald naar de IT-wereld, wordt gesteld dat het essentieel is voor een betrouwbaar computersysteem om te gaan met fouten in de componenten, waardoor het kan voorkomen dat er conflicterende informatie verstuurd wordt naar de andere componenten van het systeem.

Proof of Work

Hierna zijn bij de implementaties de manier waarop consensus behaald wordt onderzocht. Voor het beschrijven van het Proof of Work algoritme is er gebruik gemaakt van de originele presentatie van het Bitcoin protocol door Nakamoto (2008), hierin was alle informatie te vinden die benodigd was. Het egalitarian Proof of Work zoals in gebruik bij Monero is beschreven in Van Saberhagen (2013) waarin de verschillen en tekortkomingen van het Proof of Work zoals in gebruik bij Bitcoin uiteengezet wordt.



Figuur 7.2: Termen die als leidraad gebruikt zijn om het resultaat te beschrijven

Proof of Stake

Om een indicatie te geven van de grootste tekortkomingen op het gebied van Proof of Work en redeneren waarom Blockchain implementaties kiezen voor het implementeren van Proof of Stake is er gebruik gemaakt van de whitepaper van Cardano Kiayias et al. (2017), waarin de uitgelegd wordt waarom er voor Proof of Stake is gekozen in plaats van Proof of Work. Naar aanleiding van de primaire reden, namelijk dat Proof of Work enorm veel stroom verspilt, is er gezocht naar een studie die deze claim kan bevestigen, waarbij de studie van O'Dwyer en Malone (2014) gebruikt is om dit te bevestigen.

Bij het beschrijven van Delegated Proof of Stake zoals in gebruik bij EOS, bleek de whitepaper niet voldoende informatie te bevatten om het functioneel te beschrijven. Hiervoor is er dan ook een artikel gebruikt dat geschreven is door Roman, K. (2018), waarin het algoritme uitgelegd wordt.

7.3.2 Conclusie

Een gedistribueerd netwerk binnen Blockchain is getypeerd aan het consensus protocol dat gebruikt wordt. In het onderzoek zijn er twee primaire soorten geïdentificeerd, netwerken die gebruik maken van Proof of Stake of van Proof of Work, waarbij Proof of Work gebruik maakt van de rekenkracht van een node en Proof of Stake gebruik maakt van de stake van een node.

7.4 Functionaliteit en gevaren

In dit hoofdstuk wordt de vraag “Hoe werken de gedistribueerde netwerken en tegen welke gevaren zijn ze bestendig?” behandeld. Het doel van de vraag is om de werking van het gedistribueerd netwerk in kaart te brengen en tegenmaatregelen tegen aanvallen die in de functionaliteit verwerkt zitten te beschrijven.

Deze vraag is opgesteld naar aanleiding van de criteria “het moet resistent zijn tegen aanvallen” die gesteld is in de opdrachtformulering zoals gegeven door Quintor, in te zien in bijlage I. Het eerste idee om deze vraag te beantwoorden was om een vergelijking te maken tussen de implementaties op het gebied van veiligheid, waarbij er onderzocht zou worden of een aanval op de implementatie uitgevoerd was. Dit zou uiteindelijk een “beste” implementatie opleveren die geadviseerd zou worden in het adviesrapport. Uiteindelijk is dit idee niet gebruikt omdat er een aantal redenen zijn waarom dit niet zou werken:

- **Protocol volwassenheid**

Het Bitcoin protocol bestaat al sinds 2011, terwijl het Monero protocol sinds 2014 bestaat. In het begin heeft Bitcoin waarschijnlijk veel te verduren gehad qua aanvallen, waardoor het via bovenstaande vergelijking slecht uit zou komen. Daarentegen heeft Monero gedurende de drie jaar zowel verbeteringen als lessen getrokken uit het Bitcoin protocol.

- **Adoptie van de technologie**

Proof of Work implementaties zijn vatbaar voor een majority attack, waarbij een gebruiker 51% van de rekenkracht binnen het netwerk in handen moeten hebben om transacties in de Blockchain te registreren zonder dat er validatie te pas komt. Hoe minder gebruikers, hoe minder de benodigde rekenkracht om dit uit te voeren.

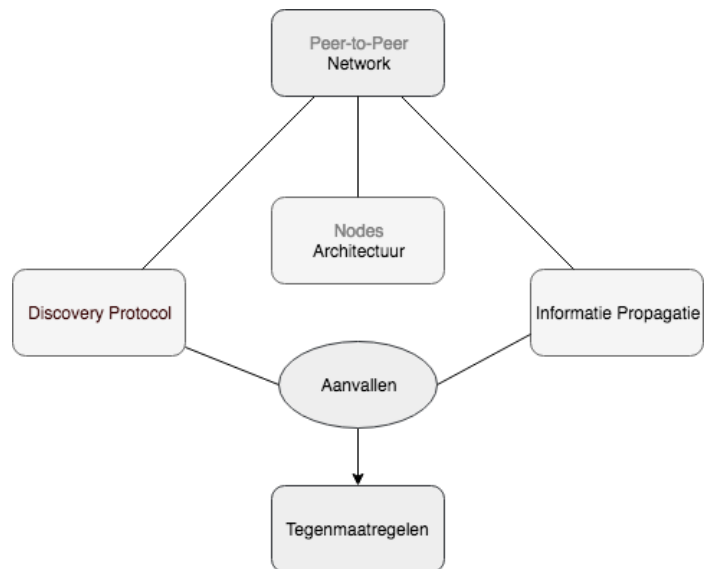
Om deze reden is er besloten om met de Blockchain expert over de aanpak en uiteindelijke doel van deze vraag te discussiëren, wat ertoe heeft geleid dat de focus van de vraag veranderd is van een vergelijking doen op basis van de veiligheid, het een meer beschrijvende vorm heeft gekregen waar er gekeken wordt naar componenten van het netwerk: discovery protocol, hoe informatie verstuurd wordt tussen twee nodes en wanneer mogelijk de knelpunten met betrekking tot aanvallen binnen deze componenten.

7.4.1 Aanpak

In fig. 7.3 is te zien welke componenten er gebruikt zijn om de benodigde informatie te vinden. Hieronder is de werkwijze en denkwijze uitgeschreven per individueel onderdeel.

Aanvallen

Allereerst is er begonnen met het zoeken naar de verschillende aanvallen die mogelijk zijn op Blockchain implementaties. Binnen het gesprek met de Blockchain Expert is hierbij het woord threat model gevallen, en zijn er een aantal aanvallen aan bod gekomen:



Figuur 7.3: Componenten en termen die als leidraad gebruikt zijn om het resultaat te beschrijven.

- **Eclipse attack**

Meer informatie en de definitie van een eclipse attack is gevonden in de studie van Heilman, Kendler, Zohar en Goldberg (2015).

- **Majority attack**

De majority attack staat beschreven op de wiki van Bitcoin, waarbij er uitgelegd wordt wat het is, wat er mee mogelijk is en waarom het bijna niet uit te voeren is.

- **Denial of Service**

Bij Denial of Service gaat het om meerdere manieren om de uitvoering van processen binnen het netwerk te verstoren, waardoor er niet een specifieke bron te vinden is voor alle mogelijke aanvallen.

- **Sybil attack**

Voor het beschrijven van de sybil attack in relatie tot Blockchain is er gebruik gemaakt van de studie gedaan door Conti, Lal, Ruj et al. (2017).

- **Double spending**

Informatie double spending is gevonden in de studie van Karame, Androuraki en Capkun (2012).

Een van de knelpunten bij het beschrijven van een aanval was een studie vinden die aantoonde wat het gevolg ervan was binnen een Blockchain implementatie.

Network

Voor het beschrijven van de verschillende netwerken is er gebruikt gemaakt van niet wetenschappelijke bronnen zoals wiki's of blogs. De reden hiervoor is dat een whitepaper van een Blockchain implementatie zelden de architectuur van het netwerk beschrijft. Deze informatie is dan ook gebruikt om de verschillende componenten van het netwerk te beschrijven.

7.4.2 Conclusie

Bitcoin Het netwerk van Bitcoin communiceert via TCP/IP en maakt gebruik van bootstrap nodes waarmee connectie wordt gemaakt op het moment dat een nieuwe deelnemer het netwerk wilt toetreden. Informatie wordt verstuurd door een voorafgedefinieerde set aan berichttypes: *inv*, *tx*, *block*, *getdata*, waarbij een *inv* bericht gebruikt wordt ter inventarisatie over de beschikbaarheid van data, *tx* bericht om een transactie te versturen, *block* bericht om een block te versturen, *getdata* bericht om data op te vragen.

Op het Bitcoin netwerk zijn meerdere aanvallen in de loop der jaren uitgevoerd en geïdentificeerd, een studie uit 2015 gedaan door Heilman et al. (2015) toont aan dat het Peer Discovery mechanisme vatbaar is voor een Sybil Attack. Nakamoto (2008) stelt dat de voordelen van het uitvoeren van een majority attack niet opweegt tegen de kosten voor de benodigde hardware om de rekenkracht te behalen. Eyal en Sirer (2014) beschrijft dat het niet nodig is om een merendeel van de rekenkracht te bezitten en introduceert de aanval selfish mining.

Cardano Het netwerk van Cardano communiceert via TCP/IP en maakt gebruik van het Kademlia protocol waardoor het maar nodig is om één bootstrap node te gebruiken om het netwerk toe te treden. De achterliggende structuur van Kademlia is een Binary Tree waarbij de positie van een deelnemer in de Binary Tree bepaald wordt door een unieke prefix van de identificatiecode. Het protocol garandeert dat een deelnemer in verbinding staat met ten minste één andere deelnemer. Informatie wordt uitgewisseld door drie abstracte berichttypes: *inv*, *req*, en *data*. Het *inv* bericht wordt gebruikt om aan te geven dat er data beschikbaar is, het *req* bericht wordt gebruikt om beschikbare data op te vragen en het *data* bericht wordt vervolgens gebruikt om de data te versturen.

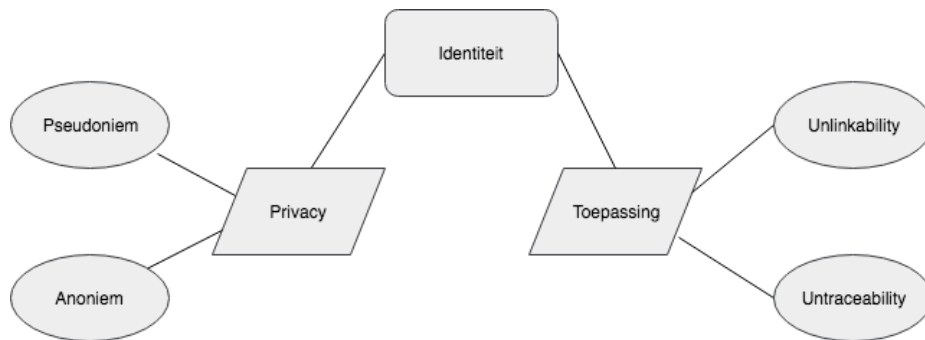
Implementaties die gebruik maken van PoS zijn afhankelijk van de manier waarop een leiderschapsverkiezing wordt gesimuleerd, waarbij er grote kans is dat het gevoelig is voor beïnvloedingen van kwaadwillende deelnemers in het netwerk in de vorm van een Sybil Attack. Cardano heeft een zwak punt in het Kademlia netwerk geïdentificeerd waardoor het mogelijk zou zijn om Eclipse Attack uit te voeren.

Monero Het netwerk van Monero maakt gebruik van het The Invisible Internet Project (I2P) protocol, dat zowel UDP/IP als TCP/IP ondersteund. Om het netwerk toe te treden wordt er gebruik gemaakt van bootstrap nodes die vastgelegd zijn in de broncode. Communicatie wordt gedaan door middel van Tunnels, waarbij elke deelnemer twee Tunnels, een inkomende en een uitgaande, heeft voor elke connectie.

EOS *Ten tijde van het onderzoek is er geen technische beschrijving beschikbaar over het netwerk component van EOS.*

7.5 Identiteit

In dit hoofdstuk wordt de vraag "Hoe wordt er omgegaan met de identiteit van de gebruiker binnen de implementatie?" behandeld. Het doel van de vraag is om de mogelijkheden en toepassingen van identiteit te beschrijven aan de hand van de geselecteerde implementaties.



Figuur 7.4: Termen die als leidraad gebruikt zijn om het resultaat te beschrijven

7.5.1 Aanpak

In fig. 7.4 is te zien welke componenten er gebruikt zijn om de benodigde informatie te vinden. In het vooronderzoek is er gevonden dat identiteit eigenlijk uit twee onderdelen bestaat binnen Blockchain implementaties. Het privacy gedeelte, wat bepaalt of de identiteit zoals in gebruik bij de implementatie pseudoniem of anoniem is. Daarnaast is de toepassing van de identiteit van belang. Een voorbeeld van de toepassing van de identiteit is bijvoorbeeld het ondertekenen van transacties.

Identiteit

In het vooronderzoek is er ook vastgesteld dat de identiteit van een gebruiker en hoe het gebruikt wordt is terug te leiden naar de architectuur van de Blockchain. De kern van Blockchain implementaties met betrekking tot identiteit en de toepassing daarvan is het gebruik van public- en private key cryptografie. Bij dit onderdeel was het belangrijk om niet teveel te beschrijven van de onderliggende cryptografie, omdat dit buiten de scope van de vraag valt.

7.5.2 Conclusie

Bitcoin is een public Blockchain waarbij de gehele historie van transacties publiekelijk in te zien is. Een deelnemer in het Bitcoin netwerk wordt geïdentificeerd aan de hand van zijn public key. Deze public key wordt onder andere opgenomen in transacties om de betaler en de ontvanger te registreren. In een studie gedaan door Reid en Harrigan (2013) wordt er een analyse model opgezet dat aantoonst dat het Bitcoin protocol niet aan de untraceability eis voldoet.

Cardano is een public Blockchain waarbij de gehele historie van transacties publiekelijk in te zien is. Cardano maakt gebruik van public- en private key cryptografie om pseudonimiteit te waarborgen. Deze keys worden gebruikt om een transactie van een bestemming te voorzien, waarbij er drie definities van adressen gebruikt worden: een public key address, een script address en een redeem address.

EOS is een consortium Blockchain waarbij gebruikers zichzelf identificeren met een unieke naam van maximaal twaalf karakters. Om te participeren binnen het netwerk dient er toegang verleent te worden door een authenticatie proces alvorens de deelnemer wordt toegelaten. Handeling binnen het netwerk worden gevalideerd door een Role Based Permissie systeem, waarbij permissies gekoppeld zijn aan actions die vastgelegd zijn in de lokale database.

Monero is een public Blockchain waarbij de gehele historie van transacties publiekelijk in te zien is. Binnen Monero heeft elke deelnemer een account die gebaseerd is op twee keys: Spend Key en een View Key. Door het afleiden van een eenmalige public key, ook wel een Stealth Address genoemd, uit de Spend Key en View Key garandeert het Monero protocol unlinkability. Untraceability wordt behaald door het gebruik van Ring Signatures. Hierbij worden meerdere Stealth Addresses toegevoegd aan een transactie, waarbij een afkomstig van de verstuurder van de transactie en de rest aangevuld door eerder gebruikte Stealth Addresses in de Blockchain. Hierdoor wordt de herkomst van een transactie gemaskeerd.

7.6 Conclusie

In het onderzoek is er een selectie van Blockchain implementaties onderzocht op de onderdelen Identity Management en Distributed Network. Door het uitvoeren van exploratief onderzoek waarin case-study gebruikt is om een gedetailleerde omschrijving van desbetreffende onderdelen op te stellen. De hoofdvraag *Welke protocol implementaties kunnen toegepast worden om de onderdelen Distributed Network en Identity Management te realiseren voor een Blockchain implementatie?* is opgedeeld in deelvragen:

1. "Welke soorten gedistribueerde netwerken worden er gebruikt?"
2. "Hoe werken de gedistribueerde netwerken en tegen welke gevaren zijn ze bestendig?"
3. "Hoe wordt er omgegaan met de identiteit van de gebruiker?"

Uit de resultaten van de deelvragen is uiteindelijk een antwoord op de hoofdvraag gekomen. Voor het onderdeel Distributed Network kan er gebruik gemaakt worden van:

Kademlia Een bestaand protocol gerealiseerd door Maymounkov en Mazières (2002). Dit protocol heeft een aantal wijzigingen binnen Cardano, zoals het versturen van informatie gaat over TCP/IP en er is een uitbreiding gemaakt op de manier waarop identificatiecodes toegekend worden aan deelnemers om een mogelijke Sybil Attack uit te sluiten.

Bitcoin Communicatie binnen het Bitcoin netwerk verloopt over TCP/IP waarbij informatie wordt verstuurd door inv, tx, block en getdata berichten. Het maakt gebruik van Proof of Work om consensus te bereiken over de staat van de Blockchain.

Monero De Monero implementatie is gefocust op het bevorderen van de privacy binnen Blockchain implementaties. Voor het netwerk is dat dan ook niet anders. Het maakt gebruik van The Invisible Project om anonimiteit in het netwerk te waarborgen.

Voor het onderdeel Identity Management is het mogelijk om de volgende protocollen toe te passen:

Bitcoin het Bitcoin protocol maakt gebruik van het UTXO-model, waarin public- en private keys gebruikt worden om de betaler en ontvanger te registreren binnen een transactie. Door het gebruik van het analysemodel gepresenteerd door Reid en Harrigan (2013) is aangetoond dat Bitcoin niet aan de untraceability en unlinkability eis voldoet.

EOS maakt gebruik van een account-model, waarin een gebruiker een unieke naam van maximaal twaalf karakters hanteert als identiteit. Daarnaast hanteert EOS een Role Based Permission Management systeem, waarbij het mogelijk is actions en handlers te definiëren.

7.7 Resultaat

Uit het onderzoek zijn waardevolle technieken gekomen die ieder valide zijn om gerealiseerd te worden in het Proof of Concept. De uiteindelijke uitvoering van het onderzoek is wat stroever gegaan. Gezien ik gebruik maak van exploratief onderzoek ben ik bepaalde aspecten tegengekomen in het onderzoek waar ik geen verstand van had. Deze aspecten heb ik achteraf nog uitgezocht en beschreven in het vooronderzoek. Met name door deze beslissing en het feit dat ik vier implementaties heb bekeken in plaats van drie, heeft ervoor gezorgd dat ik behoorlijk afgeweken ben van de originele planning om het onderzoek af te ronden.

In de globale planning, te vinden in het hoofdstuk “Aanpak”, zijn er 6 weken ingepland voor het onderzoek. Door al de bijkomstigheden zijn er uiteindelijk bijna 14 weken aan gespendeert.

8 | Advies

Aan het uitgeven van een officieel advies is niet toegekomen. Tijdens de laatste fase van het project is de tijd die bestemd was voor het adviesrapport, besteed aan het realiseren en ontwerpen van het Proof of Concept.

Zoals besproken in hoofdstuk 7.7 ben ik erg uitgelopen met het onderzoek, waardoor ik pas laat de mogelijkheid had om een adviesrapport op te stellen om die vervolgens voor te dragen aan Quintor. In overeenstemming met de bedrijfsbegeleider, Blockchain expert en medeafstudeerder Kevin Bos, is er dan ook het besluit genomen om de focus meer op de realisatie van het Proof of Concept te leggen. In dit gesprek is aangegeven dat zij graag de modulariteit van de Blockchain willen zien binnen het Proof of Concept. Dit zou er op neer komen dat het bijvoorbeeld gemakkelijk zou moeten zijn om de huidige topology van het netwerk te verwisselen met een andere structuur.

Uiteindelijk heb ik zelf besloten om voor het ontwerpen van het Proof of Concept gebruik te maken van het Kademlia protocol waarbij de berichtenstructuur van Cardano gerealiseerd zal worden. Aangezien de focus ligt op het aantonen van modulariteit zal hier tijdens het ontwerpen extra zorg aan besteed worden.

9 | Ontwerpen

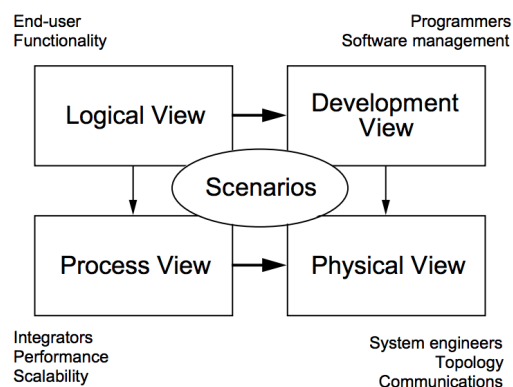
In dit hoofdstuk zal er nagedacht worden over hoe de architectuur van het Proof of Concept er uit gaat zien. Alvorens dit opgesteld wordt dient er een methode gekozen te worden om het ontwerp mee te creëren. Na de totstandkoming van de keuze zullen de user-stories uitgewerkt worden. Deze user-stories zullen gebruikt worden voor de planning van de verschillende sprints tijdens de realisatie van het Proof of Concept.

9.1 Methode

Het modelleren, implementeren en documenteren van een systeem vereist dat het systeem vanuit verschillende perspectieven wordt bekeken. Het is dan ook essentieel om een methode te selecteren die hiervan gebruik maakt. Er zullen twee methodes bekeken worden namelijk de methode van Rozanski en Woods (2012), waar ervaring mee opgedaan is tijdens de studie en de methode van Kruchten (1995), wat aangeraden is door mijn begeleider bij Quintor.

9.1.1 Kruchten

Het 4+1 view-model organiseert een beschrijving van een software-architectuur met behulp van vijf views, die elk een specifieke reeks van problemen adresseren. Het is een lichter model als de methode van Rozanski en Woods (2012) gezien het feit dat Kruchten (1995) zich alleen richt op vier viewpoints. Dit is goed te zien in het overzicht van de methode in fig. 9.1. Bij deze methode staan scenarios centraal en worden gebruikt tijdens het uitwerken van de vier viewpoints:



Figuur 9.1: Het 4+1 view-model volgens Kruchten (1995).

Development view beschrijft het systeem uit het perspectief van een ontwikkelaar en beschrijft aspecten die te maken hebben met de software matige indeling.

Logical view beschrijft hoe de eindgebruiker in staat zal zijn om de software te gebruiken. Hierbij worden vaak klassediagrammen en state diagrammen toegepast.

Process view beschrijft de dynamische aspecten van het systeem op gebied van schaalbaarheid, integratie en performance.

Physical view beschrijft hoe de softwarearchitectuur gaat werken op de benodigde hardware en focust zich met name op het gedistribueerde aspect ervan.

9.1.2 Rozanski en Woods

De methode van Rozanski en Woods (2012) beschrijft de architectuur in zogenaamde viewpoints en perspectives. In deze viewpoints en perspectives zijn categorieën te vinden die elk verantwoordelijk zijn voor een aspect van de architectuur. Zo is er bijvoorbeeld het context-viewpoint waarin de relaties, dependencies en interacties van het systeem met zijn omgeving wordt beschreven. De perspectives beschrijven de kwaliteitsattributen van een architectuur, waarbij het bijvoorbeeld gaat over accessibility of security.

Keuze

Door het aanraden van het 4+1 model door mijn begeleider en omdat het opstellen van de methode minder tijd zal kosten aangezien het minder omvang heeft als de methode van Rozanski en Woods (2012), is er besloten om gebruik te maken van het 4+1 model beschreven door Kruchten (1995).

9.2 Opstellen scenarios

Om een sprintplanning te maken is het benodigd om user-stories op te stellen voor het Proof of Concept. In de methode van Kruchten (1995) wordt dit gedaan door het opstellen van use cases in het onderdeel scenario's, dat ook wel de use case view wordt genoemd. Voor het opstellen van de scenario's zal ik allereerst requirements opstellen, aangezien die benodigd zijn voor het uitwerken van de use cases. Omdat er geen stakeholder is die de rol van eindgebruiker invult, is het niet mogelijk om een interview af te nemen voor het eliciteren van de requirements. Om deze reden heb ik dan ook de requirements uit de informatie van het gedane onderzoek gehaald die de werking van de segmenten beschrijft.

Bij het opstellen van de requirements is er onderscheid gemaakt tussen non-functional en functional requirements. De non-functional requirements beschrijven de kwaliteitsattributen van het systeem, en de functional requirements beschrijft de werking van het systeem. Hieronder is een voorbeeld gegeven van een non-functional requirement die opgesteld is:

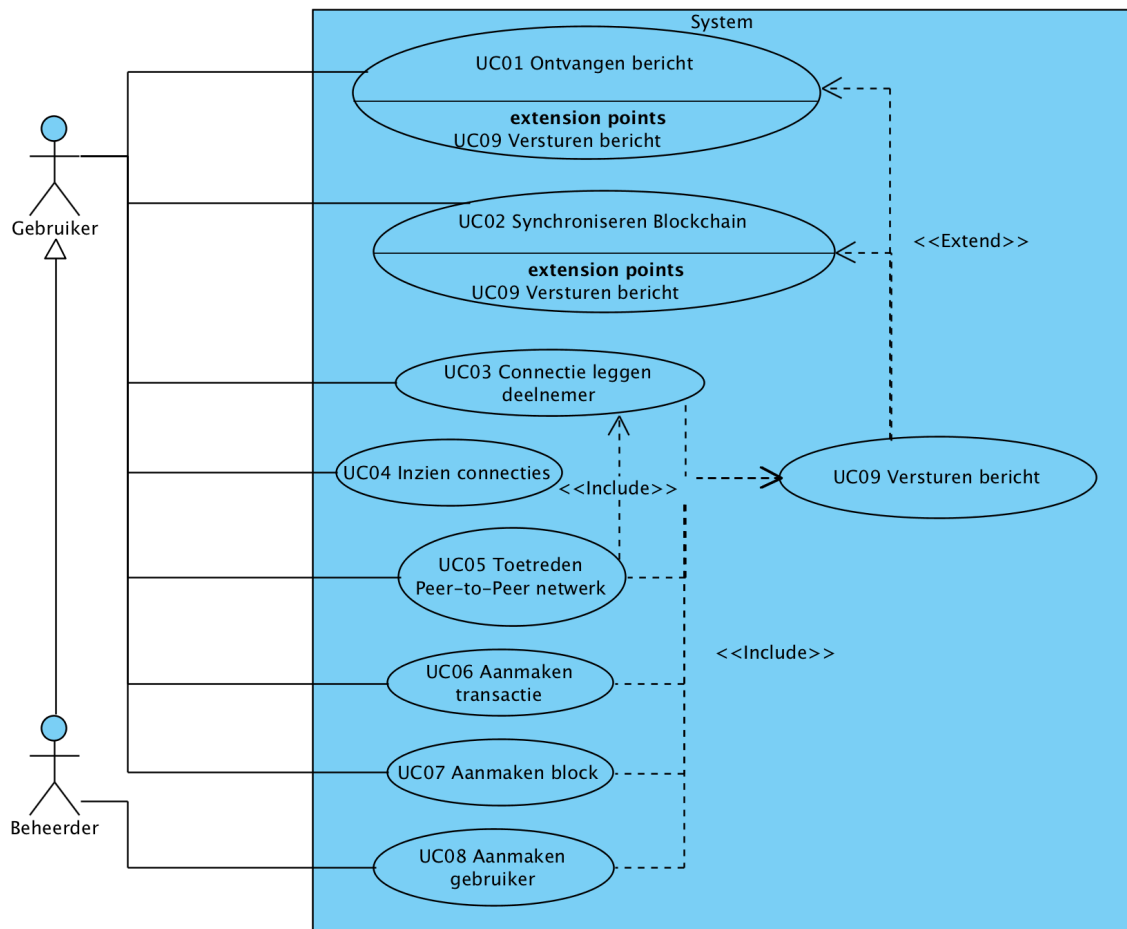
"Het systeem dient makkelijk uitgebreid te worden door de kerncomponenten modulair op te stellen."

Non-functional requirement om de eis "generiek" te beschrijven, te vinden in het Architectuurdokument.

Use-case	Connectie leggen deelnemer
Id	UCo3
Requirements	FRo3
Beschrijving	Gebruiker maakt connectie met een deelnemer uit het Peer-to-Peer netwerk
Primaire actor	Gebruiker
Secundaire actor	-
Precondition	De gebruiker is verbonden met het Peer-to-Peer netwerk
Main flow	<ol style="list-style-type: none">1. Systeem vraagt om adresgegevens(ip, poort) van deelnemer2. Actor vult informatie in3. Systeem valideert adresgegevens4. Systeem valideert dat deelnemer bereikbaar is5. Systeem creëert <i>auth</i> bericht6. Systeem voert <i>UCo3 - Versturen bericht</i> uit7. Use-case eindigt (Postconditie: Succes)
Postconditie	Succes: De gebruiker is verbonden met de deelnemer Failure: Systeem is ongewijzigd
Alternatieve flow	<ol style="list-style-type: none">1. Invalide adresgegevens (na MF3)<ol style="list-style-type: none">1.1. Use-case gaat verder bij MF12. Deelnemer is niet bereikbaar (na MF4)<ol style="list-style-type: none">2.1. Systeem toont foutmelding2.2. Use-case eindigt (Postconditie: Failure)3. Actor annuleert (Overall)

Figuur 9.2: Voorbeeld van een opgestelde use-case zoals te vinden in het Architectuurdokument.

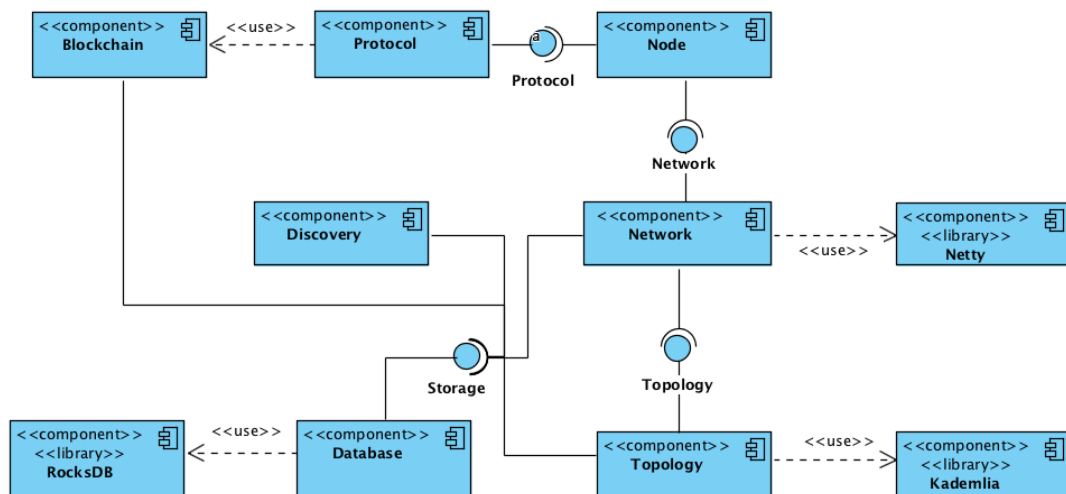
Uiteindelijk heb ik de use cases opgesteld voor de functional requirements. Een voorbeeld hiervan is te zien in 9.2. De format die voor de use case beschrijving is gebruikt komt uit een school project waarin les gegeven is door dhr. John Smeets, een uitstekend docent op het gebied van UML. In 9.3 is een overzicht te zien van alle gerealiseerde use cases en de relaties onderling. Zo wordt bijvoorbeeld de use case "bericht versturen" gebruikt in bijna elk andere use case omdat dit een van de kernfunctionaliteit is binnen het segment Distributed Network.



Figuur 9.3: Totaalplaatje opgestelde use case diagrams

9.3 Development View

Nu ik duidelijk weet welke functionaliteiten er in het systeem komen, kan ik beginnen aan de Development View. Hierin wordt het systeem beschreven in het perspectief van een ontwikkelaar waarin nagedacht wordt over de software matige indeling. Deze view zal ik als referentie gebruiken tijdens de realisatie van het Proof of Concept. Het uiteindelijke resultaat is te vinden in fig. 9.4. Elk component heeft een specifieke taak binnen de architectuur, waarbij voorbeeld het Discovery component verantwoordelijk is voor de Peer Discovery strategie die gebruikt zal worden. Een uitgebreide beschrijving van de componenten is te vinden in het Architectuurdokument document.



Figuur 9.4: Development view van de te realiseren applicatie, waarin de interacties tussen de componenten in de vorm van interfaces weergegeven worden.

10 | Ontwikkelen

Dit hoofdstuk beschrijft de werkzaamheden die uitgevoerd zijn om het Proof of Concept te realiseren. Voordat de realisatie uitgevoerd kan worden dienen er keuzes gemaakt te worden over de methoden en tools die gebruikt zullen worden tijdens de realisatie. Na dit gedaan te hebben zal er per sprint beschreven worden hoe het Proof of Concept gerealiseerd wordt.

10.1 Selecteren methoden, technieken en tools

Een ontwikkelstraat staat aan de basis van succesvolle softwareontwikkeling, en zorgt voor een duidelijke structuur tijdens de ontwikkeling van het Proof of Concept. Hierbij wordt er gebruik gemaakt van de OTAP aanpak, waarbij er voor elke fase van het ontwikkeltraject een omgeving beschikbaar is.

10.1.1 Programmeertaal

In de opdrachtformulering zoals beschreven door Quintor, te vinden in bijlage I, zijn er twee keuzes voor de programmeertaal voorgesteld, C# of Java, waarmee het Proof of Concept gerealiseerd dient te worden. Zelf heb ik met beide talen ervaring opgedaan tijdens de studie, waardoor beide talen een mogelijke keuze zijn. Aangezien mijn begeleider vanuit Quintor een Java ontwikkelaar is, neigt de keuze al snel naar desbetreffende taal. Dit omdat er altijd de mogelijkheid is om hem te benaderen wanneer er Java expertise benodigd is. Zelf ben ik niet enthousiast over Java en bleek de andere afstudeerder waarmee er een mate van overlap is tussen de keuze voor de taal waarin het Proof of Concept gerealiseerd wordt dat ook niet te zijn. Alhoewel Java de meest voor hand liggende keuze is, hebben wij een voorstel gedaan aan de begeleider om gebruik te maken van de programmeertaal Kotlin.

Kotlin

Kotlin is een programmeertaal ontwikkeld door JetBrains, een bedrijf dat bekend staat om hun wijde assortiment aan Integrated Development Environment (IDE)'s. Ze zochten een nieuwe programmeertaal die een verbetering op Java zou zijn, maar nog steeds compatible is voor migratiedoeleinden. Naar aanleiding hiervan heeft JetBrains een team opgezet dat zich bezig ging houden met het ontwikkelen van deze nieuwe programmeertaal. Deze programmeertaal is Kotlin geworden en heeft in februari 2016 een 1.0 release gehad. De programmeertaal is volledig open-source

en compileert naar de Java Virtual Machine (JVM), waardoor Java en Kotlin tegelijkertijd gebruikt kunnen worden. Dit is een belangrijk punt aangezien dit betekent dat alle libraries die beschikbaar zijn voor Java, ook gebruikt kunnen worden in Kotlin (Pieter Otten, 2017).

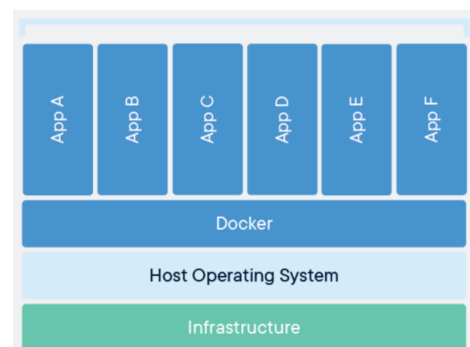
Het doel van het gebruiken van Kotlin is dan ook om de adoptiesnelheid, de werking, en de ervaring aan te tonen aan Quintor, zodat ze kunnen overwegen om deze programmeertaal in te zetten in toekomstige projecten. In overleg met de bedrijfsbegeleider is dit goed bevonden.

10.1.2 Versiebeheer

In het hoofdstuk “Infrastructuur” is er gesproken over de versiebeheer die gebruikt wordt binnen Quintor. Het is dan ook een voor de hand liggende keuze dat BitBucket gebruikt wordt voor het waarborgen van de kwaliteit van de software, aangezien ik gebruik mag maken van de infrastructuur binnen Quintor.

10.1.3 Deployment

Voor het inrichten van de OTAP omgeving zal er gebruik gemaakt worden van containerisation. Deze optie is gekozen omdat er geen centraal deployment punt bestaat waarop de applicatie gedraaid kan worden. Gezien het feit dat de Blockchain per segment ontwikkeld wordt, is er veel baat bij overdraagbaarheid. Aangezien containerisation platformneutraal is, en dus op verschillende operating systems kan draaien, lijkt het mij een gepaste keuze om deze techniek toe te passen. De meest bekende technologie die containerisation faciliteert is Docker.



Figuur 10.1: Overzicht van de werking van Docker.

Docker

Een Docker container is een lichtgewicht, alleenstaand, uitvoerbaar pakket van software die alles bevat om een applicatie uit te voeren: code, een runtime, systeem tools, systeem libraries en settings. In fig. 10.1 is een overzicht te zien hoe Docker werkt. Hierin is te zien dat Docker gebruik maakt van het operating system van de computer waardoor er geen operating systeem per applicatie nodig is. Dit zorgt ervoor dat de technologie zeer efficiënt is.

10.2 Configureren

Voor het opzetten van het project wordt er gebruik gemaakt van IntelliJ IDEA. Dit is een van de meest geavanceerde Integrated Development Environment (IDE) die beschikbaar is op de markt. Deze IDE maakt het moeiteloos om projecten die gebruik maken van onder andere Java en Kotlin op te zetten en te configureren. Bij het opzetten van een nieuw project is er de keuze om gebruik te maken van Maven of Gradle. Dit zijn allebei software project management tools die het mogelijk maken om externe libraries toe te voegen aan de codebase. In overleg met de andere afstudeerder hebben we ervoor gekozen om gebruik te maken van Maven, omdat we hier beide mee bekend zijn.

10.2.1 Maven

Tijdens de configuratie van Maven blijkt dat dit proces anders is als bij een Java project, waarbij je eigenlijk alleen een selectie hoeft te maken in de IDE. Gelukkig is de documentatie van het Kotlin project op orde en is er snel gevonden wat er aangepast moet worden. In fig. 10.2 is te zien welk stuk toegevoegd moet worden in het configuratiebestand. Na dit gedaan te hebben worden libraries correct geïmporteerd.

```
<build>
  <plugins>
    <plugin>
      <artifactId>kotlin-maven-plugin</artifactId>
      <groupId>org.jetbrains.kotlin</groupId>
      <version>${kotlin.version}</version>

      <executions>
        <execution>
          <id>compile</id>
          <goals> <goal>compile</goal> </goals>
        </execution>

        <execution>
          <id>test-compile</id>
          <goals> <goal>test-compile</goal> </goals>
        </execution>
      </executions>
    </plugin>
  </plugins>
</build>
```

Figuur 10.2: Configuratie Maven in een Kotlin project afkomstig van "Using Maven" - Kotlin (2018).

10.2.2 Docker

oor het opzetten van een Docker image is er voor nu gekozen voor een simpele aanpak, waarbij we besloten hebben dat naarmate het project vordert, en wanneer nodig, het uitgebreid zal worden.

```
1 FROM openjdk:8-jre-alpine
2
3 # Install dependencies, bash and su-exec for easy step-down from root
4 RUN apk add --no-cache bash libc6-compat && rm -rf /var/cache/apk/*
5
6 # Configure container
7 ARG JAR_FILE
8 COPY ${JAR_FILE} app.jar
9 ENTRYPOINT ["java", "-Djava.security.egd=file:/dev/./urandom", "-jar", "/app.jar"]
```

Figuur 10.3: Opgestelde configuratie voor de Dockerfile.

project naar de Docker omgeving, om het vervolgens op te starten.

In fig. 10.3 is te zien hoe de Docker configuratie is opgezet. Er wordt gebruik gemaakt van een AlpineLinux gebaseerde image waarin de Java Development Kit (JDK) en de Java Runtime Environment (JRE) voorgeconfigureerd zijn. Voorlopig kopieert het alleen maar de build artifact van het

10.3 Testen

Voor het uitvoeren en opstellen van de testen heb ik een selectie gemaakt van methodieken die toepasbaar zijn in een Agile project. Hierbij zijn er naar drie mogelijke testmethodieken gekeken: Test-driven Development (TDD), Acceptance Test Driven Development (ATDD) en Behaviour-driven development (BDD).

10.3.1 Test Driven Development

TDD verwijst naar een programmeerstijl waarin drie activiteiten nauw met elkaar verweven zijn: ontwikkeling, testen (in de vorm van unit-tests) en ontwerp (in de vorm van refactoring) (Janzen & Saiedian, 2005). Het kan worden beschreven door met de volgende stappen:

- Schrijf een unit-test die een aspect van het programma beschrijft
- Voer de test uit, deze faalt omdat het programma de functionaliteit mist
- Schrijf code die het eenvoudigst mogelijk de test laat slagen
- "Refactor" de code totdat deze voldoet aan de architectuur criteria
- Herhaal de stappen

Een belangrijk voordeel van TDD is dat het promoot om kleine stappen te nemen bij het realiseren van software. Stel dat bijvoorbeeld een nieuwe functionaliteit wordt toevoegt, gecompileerd en getest. De kans is groot dat bestaande testen falen door defecten in de nieuwe code. Het is veel gemakkelijker om deze gebreken te vinden en op te lossen als je twee nieuwe coderegels hebt geschreven in plaats van 2000. De implicatie is dat hoe sneller je ontwikkeld en testen uitvoert, hoe aantrekkelijker het is om in kleinere stappen te werk te gaan.

10.3.2 Acceptance Test Driven Development

ATDD is een methode waarbij het hele team samen discussieert over acceptatiecriteria met voorbeelden en deze vervolgens in een reeks concrete acceptatietests verwerkt voordat de ontwikkeling begint (Aggarwal & Singh, 2014).

Deze acceptatietests vertegenwoordigen de requirements van de gebruiker en functioneren als een vorm van vereisten om te beschrijven hoe het systeem zal functioneren. Tevens dienen ze ook als een manier om te controleren of het systeem functioneert zoals bedoeld.

10.3.3 Behaviour Driven Development

BDD is een methode die zich richt op het zakelijke gedrag dat de code implementeert: het 'waarom' achter de code (Wynne, Hellesoy & Tooke, 2017). BDD is een uitbreiding van TDD en ATDD. Net als bij TDD wordt er in BDD eerst de tests geschreven en daarna de applicatiecode. Het grote verschil dat te zien is:

- Tests zijn geschreven in duidelijke beschrijvende taal (Nederlands, Engels, etc..)
- Tests worden geschreven op de toepassing en zijn meer op de gebruiker gericht
- Aan de hand van voorbeelden om de vereisten te verduidelijken

10.3.4 Keuze

Uiteindelijk is de keuze gevallen op BDD. Aangezien er al scenario's zijn gemaakt is het gemakkelijk om test suites op stellen conform de de regels van Cucumber. Het is dan ook een kleine moeite om dit te doen en stelt de opdrachtgever in staat om makkelijk en overzichtelijk de geïmplementeerde use cases te zien.

10.3.5 Frameworks

Om BDD uit te voeren dienen er een aantal frameworks toegevoegd te worden aan het project. Hieronder is een overzicht gegeven van wat deze frameworks doen.

Cucumber

Cucumber is een test framework dat BDD ondersteunt (Wynne et al., 2017). Met Cucumber kan het applicatie gedrag in duidelijke, betekenisvolle tekst definiëren met behulp van een eenvoudige grammatica die wordt gedefinieerd door Gherkin. Cucumber zelf is geschreven in Ruby, maar het kan worden gebruikt om code geschreven in Ruby of andere talen te 'testen', inclusief maar niet beperkt tot Java/Kotlin, C# en Python.

JUnit

JUnit is een eenvoudig, open source framework voor het schrijven en uitvoeren van unit-tests. Dit framework zal worden ingezet om de unit-testen te schrijven voor de applicatie. JUnit is de defacto standaard voor het schrijven van tests in Java en heeft hierdoor een stabiele community die snel innoveert.

Continuous Integration

Om de testen te koppelen aan het gebruikte versiebeheer waardoor testen automatisch uitgevoerd worden, wordt er gebruik gemaakt van Continuous Integration (CI). Quintor maakt gebruik van Bamboo waarbij GitLab en Bamboo op elkaar ingesteld zijn. Om geen overbodige werkzaamheden uit te voeren is er dan ook voor gekozen om gebruik te maken van de beschikbare Bamboo omgeving.

10.4 Sprint een: Realiseren Peer to Peer netwerk

Scenarios	UC03 - Connectie leggen deelnemer
	UC09 - Versturen bericht
Componenten	Node, Network

Tabel 10.1: Betrokken scenarios en componenten bij de realisatie van het Peer-to-Peer netwerk.

Deze sprint staat in het teken van de basis van het Distributed Network, namelijk het opzetten van het netwerk. Een overzicht van de betrokken scenario's en componenten is te vinden in tabel 10.1. Het eerste wat ik gedaan heb is het vertalen van de scenario naar een Cucumber test suite. Deze test suite zal gebruikt worden om te valideren dat de gerealiseerde functionaliteit voldoet aan de opgestelde criteria. In listing 10.1 is de opgestelde test suite te vinden voor de scenario "Connectie leggen deelnemer".

Feature: Connect to participant
As a user
I want to connect to a participant
So that I can communicate with the participant

Scenario: Create connection
Given there is no connection yet
And I have entered the address
When I press enter
Then I should be connected with

Scenario: Create connection with unreachable address
Given there is no connection yet
And I have entered the address
When I press enter
Then I should receive a message "unreachable_address"

Listing 10.1: Scenario vertaald naar een Cucumber test suite.

10.4.1 Protocol

In het vooronderzoek is er geleerd dat een Peer-to-Peer (P2P) netwerk centraal staat in het segment Distributed Network. Het P2P netwerk kan gerealiseerd worden door gebruik te maken van twee mogelijke protocollen: het Transmission Control Protocol (TCP) en het User Datagram Protocol (UDP). Om hierin een keuze te maken is er een korte analyse gedaan over de aspecten van deze twee protocollen.

Transmission Control Protocol

Het Transmission Control Protocol (TCP) is het meest gebruikte protocol op het internet, het wordt namelijk gebruikt om data die benodigd is om een website te laden, te versturen. Een voordeel van TCP is dat het protocol de garantie geeft dat data in de juiste volgorde ontvangen wordt. Het protocol wacht namelijk op bevestiging dat een packet ontvangen is, alvorens een volgende packet verstuurd word. Tevens zorgt dit ervoor dat data nooit corrupt raakt of verloren gaat.

User Datagram Protocol

Het User Datagram Protocol (UDP) werkt hetzelfde als TCP alleen zit in dit protocol niet de controle of een packet correct is aangekomen. Packets worden achter elkaar verstuurd zonder na te gaan of de ontvanger ze daadwerkelijk ontvangen heeft. Dit zorgt ervoor dat de overhead van het controleren niet aanwezig is, waardoor het sneller is als het TCP protocol.

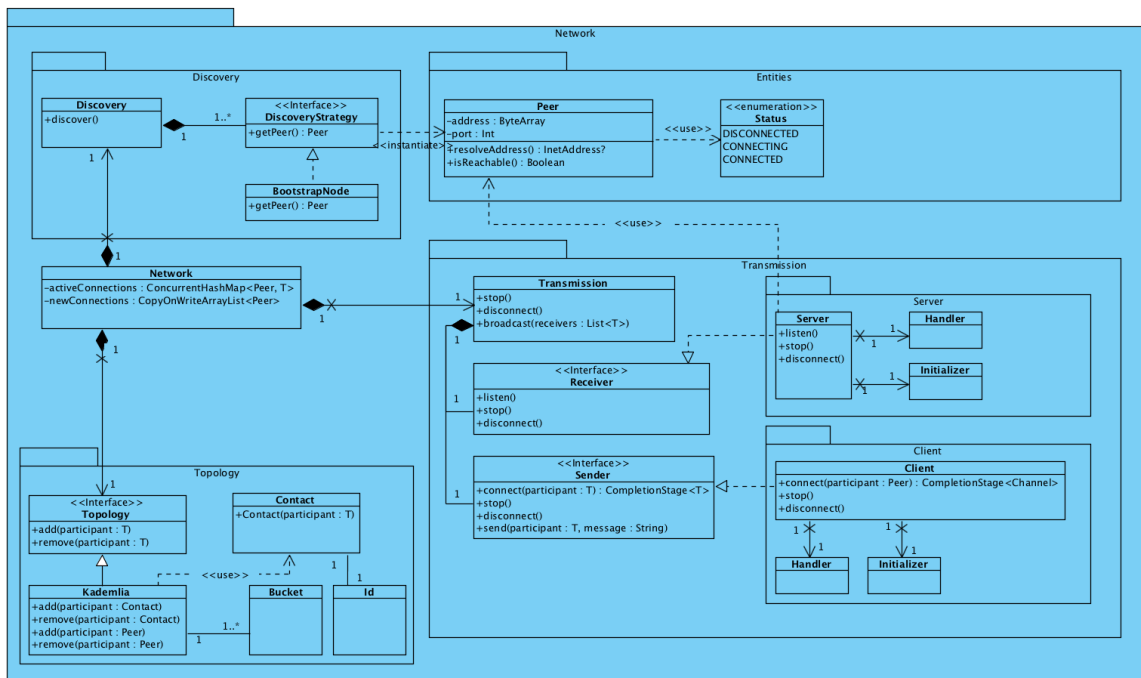
Omdat binnen een Blockchain implementatie garantie dat een transactie geregistreerd wordt zeer belangrijk is, zal er gebruik gemaakt worden van TCP/IP. De fail-safe mechaniek die in het protocol zit zal helpen om de Blockchain in een betrouwbare staat te houden.

Selectie framework

Aangezien het niet verstandig is om dit protocol zelf te realiseren ben ik op zoek gegaan naar frameworks die het TCP protocol ondersteunen. Hierbij viel mijn oog al snel op het Netty framework. Netty is een client-server framework die het mogelijk maakt om snel en simpel een netwerk applicatie op te zetten die gebruik maakt van een TCP socket server. De voorstaande reden voor het gebruik van Netty is de ondersteuning voor asynchroon programmeren, wat benodigd is om de hoge performance te behalen die nodig is voor een Blockchain implementatie. Daarnaast is Netty een de-facto standaard dat gebruikt wordt in prominente Java frameworks zoals Spring Boot, Google's GRPC en Minecraft.

10.4.2 Ontwerpen

Nu ik weet welk protocol gebruikt gaat worden voor het realiseren van het netwerk, heb ik een eerste draft gemaakt van het netwerk onderdeel dat conform de functionaliteiten van het Netty framework zijn. Het ontwerp is in fig. 10.4 te zien. Alle componenten zijn modulair opgebouwd conform de afspraak die gemaakt is over de realisatie de Blockchain. Zo zijn de functionaliteiten geïsoleerd in desbetreffende componenten waardoor ze gemakkelijk verwisseld kunnen worden.



Figuur 10.4: Gedetailleerd network component.

10.4.3 Versturen berichten

Voor de tweede scenario "Versturen bericht" is het benodigd om data om te zetten naar een formaat dat gemakkelijk verstuurd kan worden over het P2P netwerk. Een van de standaard technieken hiervoor is het omzetten van een entiteit naar bytes, door een proces genaamd serialisation. Dit is niet een van de meeste efficiënte manieren om data te versturen aangezien er een overhead is voor het omzetten en terugzetten van bytes naar entiteit en entiteit naar bytes. Om deze reden is er dan ook gezocht naar een efficiëntere manier om data te versturen. Hieronder zijn twee mogelijkheden beschreven die gebruikt kunnen worden op de Java Virtual Machine (JVM).

1. Java Serializable

De standaard manier van het implementeren van serialisatie binnen Java. Door het implementeren van een interface op een klasse is het mogelijk om een object out-of-the-box te serialiseren. Wanneer er gebruik wordt gemaakt van complexere entiteiten dient de programmeur de serialisatie zelf te implementeren.

2. Protobuf

Protocol buffers zijn Google's programmeertaal-neutrale implementatie van serialisatie. Door het eenmalig definiëren van de structuur door middel van een proto-bestand, is het mogelijk om via de library eenvoudig complexe entiteiten om te zetten naar bytes en van bytes naar entiteit. Een voordeel van Protobuf is dat het samenwerkt met alle talen die de library ondersteund. Op dit moment zijn dat Java, Python, Objective-C en C++ (Google, 2018).

De keuze hierbij is gevallen op Protobuf om entiteiten binnen de Blockchain applicatie te serialiseren. Het voordeel van Protobuf is namelijk dat het samenwerkt met alle talen die de library ondersteund. Dit zorgt ervoor dat het serialisatieproces toekomstbestendig blijft.

In fig. 10.5 is het Protobuf bestand te zien dat opgesteld is om de berichtenstructuur van Bitcoin te implementeren.

10.4.4 Resultaat

Aangezien mijn tijd beperkt was en de deze sprint in de laatste week gerealiseerd is ben ik over het algemeen tevreden met wat ik in deze sprint heb bereikt. Ik heb groter deel van de architectuur opgezet en de basis van het P2P netwerk gerealiseerd. Ook heb ik alvast een fundering neergezet voor het implementeren van de Protobuf entiteiten. Wel ben ik teleurgesteld dat ik niet verder ben gekomen met de realisatie van de functionaliteit. Daarnaast heb ik geen validaties kunnen doen door het uitvoeren van de opgestelde testen.

```
syntax = "proto3";
package com.quintor.protobuf;
import "google/protobuf/timestamp.proto";

message Message {
    string type = 1;
    string content = 2;
}

message Identification {
    uint32 port = 1;
    string genesisHash = 2;
    uint32 chainHeight = 3;
}

message Request {
    string type = 1;
    string hash = 2;
    google.protobuf.Timestamp timestamp = 3;
}

message Inventory {
    message Block {
        uint32 height = 1;
        string hash = 2;
    }
    repeated Block blocks = 1;
}
```

Figuur 10.5: Opgesteld Protobuf bestand.

11 | Evaluatie

In dit hoofdstuk worden de resultaten van de afstudeeropdracht geëvalueerd. Er wordt gekeken naar de opgeleverde producten en de kwaliteit hiervan. Vervolgens wordt de gekozen aanpak besproken en de mogelijke afwijkingen van het afstudeerplan. Als laatste wordt er gekeken naar de beroepstaken die uitgevoerd zijn.

11.1 Producten

11.1.1 Plan van Aanpak

Het plan van aanpak is te vinden in bijlage III en bevat een beschrijving van de aanpak zoals in het begin van het afstudeertraject is opgezet. Het is niet meegenomen in het iteratieve proces, waardoor het geen gedetailleerde beschrijving van de aanpakken bevat. Het plan van aanpak bevat tevens alle werkzaamheden die doorlopen zijn om tot het eindresultaat te komen en is essentieel geweest voor de uitvoering van het project. Het gaf me namelijk een goed beeld van wat er nodig was om het project tot een succesvol einde te brengen, zonder de focus van het project uit het oog te verliezen.

11.1.2 Onderzoeksrapport

Het onderzoeksrapport is te vinden in bijlage IV en bevat het resultaat van het onderzoek met als hoofdvraag "Welke protocol implementaties kunnen toegepast worden om de onderdelen Distributed Network en Identity Management te realiseren voor een Blockchain implementatie?". Ten tijde van het opzetten van het afstudeerplan was er nog geen kennis over het Blockchain domein, en ik ben dan ook zeer tevreden met de kennis die vergaard is met het gedane onderzoek. Helaas heb ik niet alle vragen kunnen behandelen die ik in eerste instantie in gedachten had, maar niettemin bevat het een goede fundering voor kennis voor de onderdelen Identity Management en Distributed Network in de Blockchain implementaties Bitcoin, EOS, Monero en Cardano.

11.1.3 Architectuurdocument

Het architectuurdocument is te vinden in bijlage V en bevat de ontwerpen van de te realiseren architectuur volgens de methode van (Kruchten, 1995). Ik vond het een duidelijke aanpak voor het benaderen van de architectuur van een systeem waarbij de meeste technieken overeen kwamen met de aangeleerde technieken vanuit school. Helaas heb ik niet de volledige architectuur kunnen implementeren, en dus ook niet kunnen testen of ik het correct opgesteld heb. Ik ben ervan overtuigd dat het een goede fundering beschrijft voor het realiseren van de segmenten Distributed Network en Identity Management.

11.1.4 Proof of Concept

Door het uitlopen van het onderzoek is er vrij weinig tijd over gebleven om het daadwerkelijke Proof of Concept te realiseren. Er is dan ook niet verder gekomen als het realiseren van het Peer to Peer netwerk met een opzet voor het versturen van berichten op de manier van Cardano. Het was in eerste instantie een ambitieuze implementatie, losstaand van het feit dat de complexiteit van de implementatie lag in de samenwerking met de onderdelen die gerealiseerd waren door Kevin Bos. Niettemin ben ik tevreden over wat ik bereikt waardoor ik toch de kern van het segment Distributed Network heb weten te realiseren.

11.2 Aanpak

Gedurende het afstudeertraject is de aanpak op Agile wijze uitgevoerd. Door de vele iteraties van zowel het vooronderzoek, de opzet van het onderzoek en het daadwerkelijke resultaat van het onderzoek is er veel tijd verloren gegaan, en zou ik in het vervolg ook niet voor een Agile aanpak kiezen bij het uitvoeren van onderzoek. Een groot valkuil waar ik mezelf op betrapte tijdens het onderzoeken van Blockchain protocollen is het feit dat ik teveel wil beschrijven en dat vervolgens ook wil uitlichten in het vooronderzoek om het concept duidelijk te maken voor de lezer.

11.2.1 Onderzoek

Het onderzoek is uitgevoerd door literatuurstudie waarbij geen toepassing bekend was. In andere afstudeerscripties komt het doorgaans voor dat er gebruik gemaakt wordt van toegepast onderzoek, waardoor het opzetten van de structuur in het gehele project nogal in de war kwam. Dit zorgde ervoor dat het onderzoeksrapport de grootste artefact van de afstudeeropdracht was en het Proof of Concept erbij kwam als bijkomstigheid. De keuzes die hiertoe geleid hebben konden in mijn ogen ook niet anders met hoe de opdracht vanuit Quintor gepresenteerd was, namelijk dat de insteek van de opdracht was om kennis op te doen van het Blockchain domein.

Daarnaast is het kiezen van EOS een fout geweest, dat oorzaak is geweest voor vele knelpunten in het onderzoeksproces. Dit is eigenlijk fout gegaan tijdens de selectie van implementaties en de bron die hiervoor gebruikt is. In het overzicht van coinmarketcap is mogelijk om alleen coins te tonen in plaats van de standaardweergave waarbij coins en tokens door elkaar heen weergegeven worden. Op het moment van schrijven is EOS een van de meest succesvolle Initial Coin Investment (ICOs) op het moment, waarbij het geen eigen coin heeft en dus ook geen eigen Blockchain implementatie. Hierdoor is de beschikbare documentatie van het protocol (bijna) beperkt tot een technische whitepaper die toch wel wat steken laat vallen voor het begrijp technisch.

Doordat er gekozen is voor exploratief onderzoek zijn er veel termen gevonden waar ik nog niet bekend mee was maar die wel nodig waren om het totaalplaatje van het onderzoek te begrijpen. Hierbij heb ik de keuze gemaakt om het vooronderzoek achteraf aan te vullen met informatie die geanalyseerd was voor de segmenten Identity Management en Distributed Network. Dit heeft ervoor gezorgd dat het onderzoek meer tijd in beslag heeft genomen dan gepland. Dit probleem komt neer op een duidelijke afbakening maken voor het onderzoek, wat ik in de toekomst dan ook zeker zal doen.

11.3 Beroepstaken

De beroepstaken die uitgevoerd zoals opgegeven in het afstudeerplan, in te zien in bijlage II, zijn hieronder verantwoord. De complexiteit is ingedeeld naar de beschrijving van beroepstaken zoals gepresenteerd in het document "Beroepstaken van de opleiding Informatica – Academie voor ICT & Media, uitgave juni 2009".

- **Selecteren, methoden, technieken en tools.**

Er zijn meerdere handelingen geweest die verantwoord kunnen worden onder deze beroepstaak. Tijdens het realiseren van het Proof of Concept zijn er zowel keuzes gemaakt op het gebied van de selectie van methoden, technieken en tools als bij het opstellen van de development workflow die gepaard ging met de realisatie.

De complexiteit van dit onderdeel komt neer op niveau 4, aangezien het zelfstandig is uitgevoerd en van voldoende complexiteit is in samenwerking met de inventarisatie van bestaande Blockchain technieken.

- **Ontwerpen systeemdeel.**

Het ontwerpen van het systeemdeel betrof het modelleren van de verschillende technologieën die uit de selectie van het adviesrapport gekomen zijn. De samenwerking tussen de technologieën dient modulair te zijn zodat elk losstaand deel vervangen kan worden. Het systeem dient tevens samen te werken met componenten die gerealiseerd zijn door een andere afstudeerder. Er is hierbij gebruik gemaakt van de ontwerpmethode 4+1 architectural view model zoals beschreven door Kruchten (1995).

De complexiteit van dit onderdeel komt neer op niveau 4, aangezien het systeem rekening dient te houden met de geïdentificeerde gevaren in het gedane onderzoek en het 4+1 architectural view model beschrijft de architectuur vanuit verschillende gezichtspunten.

- **Bouwen applicatie.**

De realisatie van het Proof of Concept betreft het bouwen van een applicatie die aansluit op een ander deel van de Blockchain dat gerealiseerd is door een afstudeerder. Er wordt hierbij gebruik gemaakt van frameworks waarbij er redenering aanwezig is voor gekozen frameworks. Er wordt gebruik gemaakt van versiebeheer dat gefaciliteerd is door Quintor, en er wordt containerization toegepast om een testomgeving te simuleren.

De complexiteit van dit onderdeel komt neer op niveau 4, aangezien het aansluit op bestaande software en er gebruik gemaakt wordt van een ontwikkelomgeving inclusief testomgeving en versiebeheertool.

- **Initiëren en plannen testproces.**

Helaas is het niet meer mogelijk geweest om de kwaliteit aan te tonen door het uitvoeren van een opgesteld testplan. Binnen het architectuurdokument is er wel rekening gehouden met criteria die gesteld is aan de implementatie in de vorm van non-functional requirements. Daarnaast is er wel inventarisatie gedaan naar de mogelijkheden met betrekking tot het testen van de applicatie. Dit is beschreven in hoofdstuk ??.

11.4 Functioneren binnen Quintor

Vanaf dag één voelde ik mij zeer welkom binnen Quintor. Het bedrijf is erg betrokken bij zijn medewerkers en daarbij ook de afstudeerders en stagiaires. Het bijzondere aan afstuderen bij Quintor is dat je in een groep met afstudeerders komt. Dit is erg fijn omdat je dan nog steeds als studenten onder elkaar bent, en waarbij nodig hulp kunt verlenen aan de andere. Daarnaast was de begeleiding die aangeboden werd meer als voldoende en waren de medewerkers altijd in om mee te

denken over problemen waar ik tegen aan liep. Het jammere aan afstuderen vind ik wel dat je geïsoleerd aan een opdracht werkt, waardoor je grotendeels op jezelf aangewezen bent en niet echt het bedrijfsleven ervaart. Niettemin doet Quintor er alles aan om je te betrekken bij het bedrijf. Dit doen zij door het organiseren van leuke evenementen zoals een uitje naar de verschillende festiviteiten die georganiseerd worden in Den Haag of een kennis avond over een van je favoriete onderwerpen. Tijdens de kennis avonden is er ook de gelegenheid om kennis te maken met de consultants die Quintor in dienst die normaal niet aanwezig zijn op kantoor.

11.5 Leerpunten

Hieronder heb ik een aantal leerpunten voor mijzelf neergezet die afkomstig zijn uit het afstudeertraject.

- **Zorg ervoor dat de scope van je onderzoek duidelijk is, zodat je een afbakening kan maken. Dit zorgt ervoor dat je rekening houdt met de onvoorspelbaarheid van je onderzoek en dat waar nodig geëscaleerd kan worden.** Omdat de scope niet duidelijk was heb ik achteraf het vooronderzoek aangepast met kennis die benodigd was om de deelvragen uit het hoofdonderzoek te beantwoorden.
- **Begin vanaf het begin al met het beschrijven van de uitvoering van het onderzoek, zodat je niet nogmaals je eigen onderzoek hoeft door te nemen om te achterhalen hoe je te werk bent gegaan. Analyseer hierbij je gebruikte bronnen op validiteit en toegevoegde waarde binnen de context.** De uitgevoerde stappen vertalen naar concrete processtappen binnen het hoofdonderzoek zijn niet duidelijk beschreven, waarbij een analyse op de gebruikte bronnen en welke informatie geaggregeerd is niet te achterhalen is.
- **Besteed meer tijd aan het duidelijk krijgen van je opdracht. Structureer hierbij je gesprekken en ga na wat je als resultaat verwacht. Wanneer dit niet zo is, bespreek dit met je begeleider.** Dit leerpunt komt voort uit de manier waarop de orientatie is gedaan in het begin van project. Hierbij zijn totaal andere resultaten gekomen als de specificatie van de criteria.
- **Stel een realistisch maar compleet Plan van Aanpak op. Schets hierbij de individuele stappen tot het eindresultaat, in plaats van diep in detail doorgaan op het uitwerken van de aanpak voor specifieke aspecten van het project.** Er is veel tijd besteed aan het opzetten van een onderzoeksplan die eigenlijk niet gebruikt is, en later een nieuwe revisie van is opgezet.

- **Leg afspraken over de opdracht vast met de opdrachtgever en laat het ondertekenen, zo kan de opdracht nooit veranderd worden halverwege de opdracht.** De insteek van de opdracht is naar mijn idee op twee plekken veranderd. Allereerst tijdens de orientatie, dat opgesteld was om duidelijkheid te krijgen over de niet gespecificeerde criteria. Het tweede moment is tijdens het advies gesprek, waarin wordt voorgesteld om twee verschillende aspecten van het systeem te ontwikkelen.

12 | Aanbevelingen

Tijdens het onderzoek is er helaas geen tijd geweest om de actualiteiten in het Blockchain domein te behandelen. Hiernaar is wel een korte inventarisatie gedaan, waardoor er verdergewerkt kan worden op de werkzaamheden zoals gepresenteerd in dit document.

12.1 Directed Acyclic Graph

De implementaties NANO en IOTA maken gebruik van een Directed Acyclic Graph (DAG), een nieuw soort architectuur die naar verluid de schaalbaarheid van Blockchain technologie dient te vergroten. Het is dan ook zeker interessant om te kijken of deze kennis van belang is voor Quintor.

12.2 Bitcoin Lightning Network

Bitcoin is al een tijd bezig met onderzoek naar verbetering van de transactie throughput. Dit netwerk is een tweede protocol laag bovenop een Blockchain implementatie die het mogelijk maakt om transacties direct door te zetten. Alhoewel het nog in de kinderschoenen staat en het nog vatbaar is voor bepaalde aanvallen zoals geïdentificeerd in dit onderzoek, is het een interessante techniek om te onderzoeken.

12.3 Ethereum Casper

Ethereum probeert van het Proof of Work consensus af te stappen, alleen willen ze hierdoor geen hard-fork veroorzaken. De eerste fase van Casper, Casper the Friendly Finality Gadget, is dan ook een hybride tussen Proof of Stake en Proof of Work die ingezet kan worden om het Ethereum netwerk te upgraden zonder een hard-fork te veroorzaken. Uiteindelijk zal Casper overgaan naar Casper the Friendly Ghost, een volledige implementatie van Proof of Stake.

12.4 EOS

Op 1 juni wordt EOS gepubliceerd waarbij het eindelijk mogelijk is om deel te nemen aan het netwerk. Aangezien Quintor Blockchain wilt inzetten als ontwikkelingsplatform, is het interessant om deze implementatie te volgen aangezien het primair gebouwd is om ingezet te worden als ontwikkelingsplatform.

12.5 Network Address Translators (NAT) Hole Punching

Ford, Srisuresh en Kegel (2005) presenteert een aantal technieken om Hole Punching toe te passen in P2P protocollen. In hoeverre dit gebruikt wordt in Blockchain implementaties is niet verder onderzocht waardoor deze studie waardevolle informatie kan bevatten. Tijdens het scannen van de tekst is er ook een stuk over IPv6 beschreven wat zeker interessant is voor toekomstige adoptie van het IPv6 adres.

Literatuur

- Castor, A. . (2017). *A short guide to bitcoin forks - coindesk*. Verkregen van <https://www.coindesk.com/short-guide-bitcoin-forks-explained/>
- Roman, K. . (2018). *Understanding eos and delegated proof of stake — steemit*. Verkregen van <https://steemit.com/eos/@eosgo/understanding-eos-and-delegated-proof-of-stake>
- Aggarwal, V. & Singh, M. (2014). Acceptance test driven development. *Journal of Advanced Computing and Communication Technologies (ISSN: 2347-2804) Volume(2)*.
- Antonopoulos, A. M. (2014). *Mastering bitcoin: Unlocking digital crypto-currencies* (1st dr.). O'Reilly Media, Inc.
- Atzori, M. (2015). Blockchain technology and decentralized governance: Is the state still necessary?
- Bawa, M., Cooper, B. F., Crespo, A., Daswani, N., Ganesan, P., Garcia-Molina, H., ... others (2003). Peer-to-peer research at stanford. *ACM SIGMOD Record*, 32(3), 23–28.
- Bitcoin Wiki. (2010). *Bitcoin wikio*. Verkregen van <https://en.bitcoin.it/wiki>
- Conti, M., Lal, C., Ruj, S. et al. (2017). A survey on security and privacy issues of bitcoin. *arXiv preprint arXiv:1706.00916*.
- Ethereum. (2017). *Create a hello world contract in ethereum*. Verkregen van <https://www.ethereum.org/greeter> ([Online; benaderd op 3 april, 2018])
- Eyal, I. & Sirer, E. G. (2014). Majority is not enough: Bitcoin mining is vulnerable. In *International conference on financial cryptography and data security* (pp. 436–454).
- Ford, B., Srisuresh, P. & Kegel, D. (2005). Peer-to-peer communication across network address translators. In *Usenix annual technical conference, general track* (pp. 179–192).
- Google. (2018). *Protocol buffers*. Verkregen van <https://developers.google.com/protocol-buffers/> ([Online; benaderd op 30 september, 2018])
- Heilman, E., Kendler, A., Zohar, A. & Goldberg, S. (2015). Eclipse attacks on bitcoin's peer-to-peer network. In *Usenix security symposium* (pp. 129–144).
- Janzen, D. & Saiedian, H. (2005). Test-driven development concepts, taxonomy, and future direction. *Computer*, 38(9), 43–50.
- Karame, G. O., Androulaki, E. & Capkun, S. (2012). Double-spending fast payments in bitcoin. In *Proceedings of the 2012 acm conference on computer and communications security* (pp. 906–917).

- Kiayias, A. et al. (2017). Ouroboros: A provably secure proof-of-stake blockchain protocol. In *Annual international cryptology conference* (pp. 357–388).
- Kotlin. (2018). *Using maven in kotlin*. Verkregen van <https://kotlinlang.org/docs/reference/using-maven.html> ([Online; benaderd op 30 september, 2018])
- Kruchten, P. B. (1995). The 4+ 1 view model of architecture. *IEEE software*, 12(6), 42–50.
- Lamport, L. et al. (1982). The byzantine generals problem. *ACM Transactions on Programming Languages and Systems (TOPLAS)*, 4(3), 382–401.
- Maymounkov, P. & Mazieres, D. (2002). Kademlia: A peer-to-peer information system based on the xor metric. In *International workshop on peer-to-peer systems* (pp. 53–65).
- Nakamoto, S. (2008). Bitcoin: A peer-to-peer electronic cash system.
- O'Dwyer, K. J. & Malone, D. (2014). Bitcoin mining and its energy footprint..
- Pfitzmann, A. & Köhntopp, M. (2001). Anonymity, unobservability, and pseudonymity—a proposal for terminology. In *Designing privacy enhancing technologies* (pp. 1–9).
- Pieter Otten. (2017). *Kotlin vs java - mediaan*. Verkregen van <https://www.mediaan.com/nl/kotlin-vs-java/> ([Online; benaderd op 10 april, 2018])
- Reid, F. & Harrigan, M. (2013). An analysis of anonymity in the bitcoin system. In *Security and privacy in social networks* (pp. 197–223). Springer.
- Rozanski, N. & Woods, E. (2012). *Software systems architecture: working with stakeholders using viewpoints and perspectives*. Addison-Wesley.
- Schwaber, K. & Sutherland, J. (2011). The scrum guide. *Scrum Alliance*, 21.
- Van Saberhagen, N. (2013). *Cryptonote v 2.0*.
- Wynne, M., Hellesoy, A. & Tooke, S. (2017). *The cucumber book: behaviour-driven development for testers and developers*. Pragmatic Bookshelf.
- Zen, A. (2017). *Cryptokitties | collect and breed digital cats*. Verkregen van <https://www.cryptokitties.co/press>
- Zheng, Z., Xie, S., Dai, H., Chen, X. & Wang, H. (2017). An overview of blockchain technology: Architecture, consensus, and future trends. In *Big data (bigdata congress), 2017 ieee international congress on* (pp. 557–564).

Bijlages

I Opdrachtformulering

ONTWIKKELING VAN EEN GEDISTRIBUEERDE BLOCKCHAIN

Bouw een blockchain implementatie zonder gebruik te maken van bestaande blockchain libraries.

Organisatie

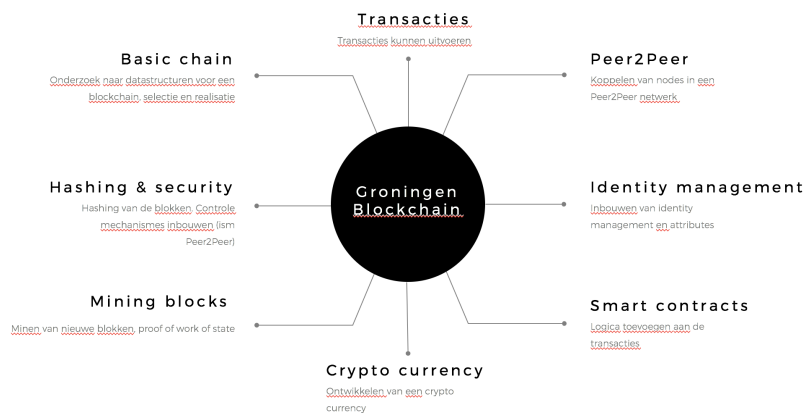
Quintor is een toonaangevend bedrijf op het gebied van Agile software development, Enterprise Java / .NET technologie en mobile development. Wij hebben sinds onze oprichting in 2005 een gezonde groei doorgemaakt en hebben inmiddels 150 personeelsleden. Vanuit onze vestigingen in Amersfoort, Groningen en Den Haag ondersteunen wij onze klanten bij de uitdagingen die grootschalige enterprise projecten met zich meebrengen. Quintor beschikt over een Software Factory waarin wij inhouse projecten voor onze klanten uitvoeren.

Probleemstelling

Voor de ontwikkeling van een blockchain dienen een aantal software componenten te worden ontwikkeld:

1. Een gegevensstructuur voor een node
2. Hashing functionaliteit
3. Mining van blokken
4. Synchronisatie van nodes (peer2peer)
5. Transacties
6. Identity management
7. Eventueel een crypto currency en smart contracts.

GRONINGEN BLOCKCHAIN



Deze opdracht richt zich op het ontwikkelen van een gedistribueerd netwerk(4) en identity management(6).

De overige onderdelen zullen in andere opdrachten gerealiseerd worden.

Aanpak

Allereerst worden in afstemming met de opdrachtgever (Johan Tillema) de uitgangspunten bepaald voor de te ontwikkelen blockchain. Deze gaan over snelheid, beveiligingsniveau en toepassingsmogelijkheden.

Vervolgens worden voor 4) en 6) de verschillende architectuur alternatieven in kaart gebracht. Dit wordt gedaan door het uitvoeren van literatuur onderzoek en door te kijken welke keuzes zijn gemaakt in andere blockchain implementaties zoals Ethereum, Hyperledger of BitCoin.

Samen met de opdrachtgever wordt een keuze gemaakt voor de toe te passen architectuur.

Vervolgens wordt een eerste gedistribueerde blockchain geïmplementeerd in Java of in .NET.

Achtergrond probleemstelling

Blockchains garanderen integriteit van data door gebruikmaking van cryptografische primitieven zoals hash functies en public-private key cryptografie. Door het ondertekenen van berichten wordt authenticiteit gegarandeerd en de hash functies zorgen voor een keten die mutaties van oude data onmogelijk maakt.

Op basis van deze onderzoeksvragen wordt een proof of concept verwacht. Op te leveren producten: een basaal blockchain implementatie die aan de volgende eisen voldoet;

- 1) er worden geen blockchain libraries gebruikt
- 2) het moet resistent tegen aanvallen zijn
- 3) het moet gedistribueerd zijn
- 4) er wordt op decentrale wijze consensus bereikt

II Afstudeerplan

Afstudeerplan

Informatie afstudeerder en gastbedrijf *(structuur niet wijzigen)*

Afstudeerblok: 2018-1.1 (start uiterlijk 5 februari 2018)

Startdatum uitvoering afstudeeropdracht:

Inleverdatum afstudeerdossier volgens jaarrooster: 1 juni 2018

Studentnummer: 14068265

Achternaam: van Hoven

Voorletters: J.

Roepnaam: Jeffrey

Adres: Rehobothplantsoen 14

Postcode: 2751BK

Woonplaats: Moerkapelle

Telefoonnummer: 0795932704

Mobiel nummer: 0646157795

Privé emailadres: jeffreyvanhoven@gmail.com

Opleiding: Informatica

Locatie: Zoetermeer

Variant: voltijd

Naam studieloopbaanbegeleider: Renate Vermeij

Naam begeleidend examiner: T. Cocx

Naam tweede examiner: D.R. Stikkolorum

Naam bedrijf: Quintor

Afdeling bedrijf: n.v.t

Bezoekadres bedrijf: Lange Vijverberg 4-5

Postcode bezoekadres: 2513 AC

Postbusnummer:

Postcode postbusnummer:

Plaats: Den Haag

Telefoon bedrijf: 070-2044037

Telefax bedrijf:

Internetsite bedrijf: <https://www.quintor.nl/>

Achternaam opdrachtgever: Tillema

Voorletters opdrachtgever: J.

Titulatuur opdrachtgever:

Functie opdrachtgever:

Doorkiesnummer opdrachtgever:

Email opdrachtgever: jtillema@quintor.nl

Achternaam bedrijfsmentor: dhr. Ooms

Voorletters bedrijfsmentor: B

Titulatuur bedrijfsmentor:

Functie bedrijfsmentor: Java Software Engineer

Doorkiesnummer bedrijfsmentor:

Email bedrijfsmentor: booms@quintor.nl

Doorkiesnummer afstudeerder:

Functie afstudeerder (deeltijd/duaal):

Titel afstudeeropdracht:

Ontwikkelen van een gedistribueerde Blockchain.

Opdrachtschrijving**Bedrijf**

Quintor is een toonaangevend bedrijf op het gebied van Agile software development, Enterprise Java/ .NET technologie en mobile development. Wij hebben sinds onze oprichting in 2005 een gezonde groei doorgemaakt en hebben inmiddels 150 personeelsleden. Vanuit onze vestigingen in Amersfoort, Groningen en Den Haag ondersteunen wij onze klanten bij de uitdagingen die grootschalige Enterprise projecten met zich meebrengen. Quintor beschikt over een Software Factory waarin wij inhouse projecten voor onze klanten uitvoeren.

Probleemstelling

Quintor is een bedrijf die klanten ondersteund bij het realiseren van grootschalige, uitdagende Enterprise projecten. Aangezien de toepassing en adoptie van Blockchain technologie steeds groter wordt wil het bedrijf de toepassingsmogelijkheden en technieken onderzoeken om zo inzicht te kunnen krijgen in hoe het gebruikt kan worden in de aangeboden vraagstukken vanuit klanten.

Sinds de opkomst van Bitcoin is de Blockchain technologie, de techniek die het mogelijk maakt om het op een gedecentraliseerde manier te laten werken, steeds populairder geworden. Alhoewel de Blockchain-technologie nog in de kinderschoenen staat, gaan de ontwikkelingen in het domein zeer snel. Zo worden er toepassingen bedacht die niet alleen voor de financiële markten interessant zijn, maar ook voor bijvoorbeeld het digitaliseren van contracten en contractbeheer.

De focus in deze opdracht ligt op het onderzoeken van de Blockchain onderdelen Identity Management en Distributed Network. Er zullen Blockchain implementaties onderzocht worden die de onderdelen geïmplementeerd hebben om een zo compleet mogelijk technisch overzicht te creëren van de technieken en protocollen die gebruikt zijn om de onderdelen te realiseren. Daarnaast wordt er onderzocht wat de toepassingen en de doelen van de bestaande implementaties zijn. Uiteindelijk zal er een advies uitgegeven worden aan de opdrachtgever, waarbij een keuze gemaakt zal worden op de manier waarop een Proof of Concept gerealiseerd gaat worden met als doel het toetsen van de gekozen technieken.

Doelstelling van de afstudeeropdracht

Het doel van deze opdracht is middels het opstellen van een Proof of Concept van de Blockchain onderdelen Network & Identity Management en Distributed Network, zonder gebruik te maken van bestaande oplossingen, kennis te ontwikkelen voor Quintor op het gebied van Blockchain technologie.

Resultaat

De opdracht zal een Proof of Concept van de Blockchain onderdelen Network & Identity Management en Distributed Network opleveren waarbij er gebruik gemaakt wordt van advies uit het literatuuronderzoek gedaan naar de onderdelen in bestaande Blockchain implementaties.

Uit te voeren werkzaamheden, inclusief een globale fasering, mijlpalen en bijbehorende activiteiten

5 dagen - Plan van Aanpak opstellen met behulp van J. Tillema.

- Opstart

15 dagen - Literatuuronderzoek naar Blockchain waarbij de volgende bekende architecturen worden onderzocht:

- Ethereum
- HyperLedger
- BitCoin.

Eventueel kennis uitwisselen met Blockchain experts van het Blockchain Fieldlab Education in Groningen waarvan Quintor medeoprichter van is.

35 dagen - Ontwikkelen, ontwerpen en testen.

- Waarbij geen gebruik gemaakt wordt van bestaande oplossingen.
- Door middel van Agile Software Development

Hierbij zullen de volgende mijlpalen behaald worden:

- Afronding implementatie Distributed Network.
- Afronding implementatie Network & Identity Management.

10 dagen – Testen.

5 dagen – Overdracht.

Op te leveren (tussen)producten

Product	
Plan van Aanpak	Een document met daarin de planning en de afspraken die gemaakt zijn met de opdrachtgever.
Sprint	Per twee weken zal er een sprintplanning plaatsvinden waarbij nieuwe taken worden ingedeeld. Aan het eind van een sprint zal er een presentatie gegeven worden over de voortgang van het project.
Adviesrapport	Een document met daarin de uitkomst van het literatuuronderzoek naar Ethereum, HyperLedger en BitCoin.
Proof of Concept	Als uiteindelijk resultaat een generieke implementatie van de onderdelen Network & Identity Management en een Distributed Network.

Te demonstreren competenties en wijze waarop

Kerntaak	
1.1	Selecteren, methoden, technieken en tools.
<p>Het juist uitzoeken van een development workflow en de technieken die daarbij te pas komen in overeenstemming met Quintor. Daarnaast zullen er beslissingen gemaakt moeten worden die invloed hebben op de manier waarop de Blockchain technologie geïmplementeerd zal worden:</p> <p>Bijvoorbeeld:</p> <ul style="list-style-type: none"> - Gaat het onderdeel Distributed Network webbased werken of met een ander protocol? <p>Concrete taken:</p> <ul style="list-style-type: none"> - Het opzetten van een OTAP-omgeving en/of DevOps toepassen om dit te simuleren. - Selecteren en adviseren over implementatie van de onderdelen Distributed Network en Network & Identity Management. 	
3.2	Ontwerpen systeemdeel.
<p>Voor het opstellen van een generieke Blockchain zal het nodig zijn om de complexe delen van de te ontwikkelen applicatie met behulp van UML uit te werken.</p> <p>Concrete taken:</p> <ul style="list-style-type: none"> - Een object georiënteerd ontwerp van de onderdelen Distributed Network en Network & Identity Management waarbij er rekening gehouden wordt met de generieke toepassing en samenwerking van de twee onderdelen. - Het identificeren, integreren en ontwerpen van de onderdelen die onderzocht zijn tijdens het literatuuronderzoek. - Rekening houdende met beveiligingseisen gesteld door Quintor. 	
3.3	Bouwen applicatie.
<p>Het ontwikkelen van de onderdelen Distributed Network en Network & Identity Management waarbij gebruikt wordt gemaakt van een object georiënteerde programmeertaal.</p> <p>Concrete taken:</p> <ul style="list-style-type: none"> - Het ontwikkelen van de onderdelen Distributed Network en Network & Identity Management door middel van de programmeertalen Java of C#. - Gebruik makend van de softwaremanagement tools die opgezet zijn voor de OTAP-omgeving en/of de DevOps toepassingen om deze omgeving te simuleren. 	
3.4	Initiëren en plannen testproces.
<p>Om de integriteit van de ontwikkelde software te waarborgen zullen er verschillende testen gemaakt worden. Een belangrijk aspect is het testen van de security bij het implementeren van het Network & Identity Management onderdeel.</p> <p>Concrete taken:</p> <ul style="list-style-type: none"> - Het onderzoeken van test-, soorten en strategieën die gebruikt worden voor een Blockchain implementatie. - Eventueel methodiek hanteren die Quintor gebruikt. 	

III Plan van Aanpak

Blockchain: Identity Management en Distributed Network

Plan van Aanpak

Jeffrey van Hoven
1 juni 2018

Inhoudsopgave

1 Aanleiding	3
2 Probleemanalyse	4
3 Doelstelling	5
4 Resultaten	6
4.1 Adviesrapport	6
4.2 Proof of Concept	6
5 Aanpak	7
5.1 Onderzoeksopzet	7
5.1.1 Dataverzameling	7
5.1.2 Dataomschrijving	7
Inclusie- en exclusiecriteria	8
Hard-forks	8
Consensus algoritme	8
5.1.3 Analysemethode	8
5.2 Adviesrapport	8
5.3 Proof-of-Concept	9
6 Planning	10

Inleiding

In dit document wordt de aanpak beschreven van de afstudeeropdracht “Ontwikkeling van de Blockchain onderdelen Distributed Network en Identity Management”, aangeboden door Quintor.

De resultaten van het uitgevoerde onderzoek naar de manier waarop Blockchain implementaties de onderdelen Distributed Network en Identity Management heeft als doel het bedrijf te adviseren over de mogelijkheden om een zo generiek mogelijke implementatie te realiseren van waarbij acties beperkt worden door het onderdeel Identity Management.

1 | Aanleiding

In 2017 heeft Quintor in samenwerking met DUO/MinOCW, Groningen Declaration Network (GDN), Stichting ePortfolio Support (StePS), TNO en Rabobank, het Blockchain Field-lab Education (BFE) gestart in Groningen. Het Blockchain-lab is opgezet om expertise en kennis uit te wisselen op regionaal, nationaal en internationaal gebied.

De oprichting van het Blockchain Field-lab Education heeft er mede voor gezorgd dat Quintor meer kennis wilt opdoen op het gebied van Blockchain. Daarnaast wil het bedrijf in de toekomst Blockchain technologie inzetten om vraagstukken vanuit klanten op te lossen. Door het aanbieden van een doorlopende afstudeeropdracht wil het bedrijf erachter komen wat er voor nodig is om een Blockchain implementatie te creëren.

2 | Probleemanalyse

Quintor is een bedrijf die klanten ondersteund bij het realiseren van grootschalige, uitdagende Enterprise projecten. Aangezien de toepassing en adoptie van Blockchain technologie steeds groter wordt wil het bedrijf de toepassingsmogelijkheden en technieken onderzoeken om zo inzicht te kunnen krijgen in hoe het gebruikt kan worden in de aangeboden vraagstukken vanuit klanten.

Sinds de opkomst van Bitcoin is de Blockchain technologie, de techniek die het mogelijk maakt om het op een gedecentraliseerde manier te laten werken, steeds populairder geworden. Alhoewel de Blockchain-technologie nog in de kinderschoenen staat, gaan de ontwikkelingen in het domein zeer snel. Zo worden er toepassingen bedacht die niet alleen voor de financiële markten interessant zijn, maar ook voor bijvoorbeeld het digitaliseren van contracten en contractbeheer.

De focus in deze opdracht ligt op het onderzoeken van de Blockchain onderdelen Identity Management en Distributed Network. Er zullen Blockchain implementaties onderzocht worden die de onderdelen geïmplementeerd hebben om een zo compleet mogelijk technisch overzicht te creëren van de technieken en protocollen die gebruikt zijn om de onderdelen te realiseren. Daarnaast wordt er onderzocht wat de toepassingen en de doelen van de bestaande implementaties zijn. Uiteindelijk zal er een advies uitgegeven worden aan de opdrachtgever, waarbij een keuze gemaakt zal worden op de manier waarop een Proof-of-Concept gerealiseerd gaat worden met als doel het toetsen van de gekozen technieken.

3 | Doelstelling

Aangezien de opdracht verspreid is over onderdelen van Blockchain technologie is er een globaal doel en een doel die specifiek voor deze opdracht geldt. Het streven naar het globale doel is het opdoen van kennis omtrent het realiseren van een Blockchain implementatie. Het doel van deze specifieke opdracht is middels het opstellen van een Proof-of-Concept van de Blockchain onderdelen Identity Management en Distributed Network, zonder gebruik te maken van bestaande oplossingen, kennis te ontwikkelen voor Quintor op het gebied van Blockchain technologie.

4 | Resultaten

4.1 Adviesrapport

Er zal een adviesrapport opgesteld worden die, met behulp van de informatie uit het onderzoek, technieken aanbeveelt om de Blockchain onderdelen Identity Management en Distributed Network te realiseren. Aan de hand van dit adviesrapport zal er in samenwerking met het bedrijf een besluit genomen worden over de technieken die geadviseerd zijn.

4.2 Proof of Concept

Het Proof-of-Concept zal de realisatie van de onderdelen Identity Management en Distributed Network bevatten met daarbij de opgestelde documentatie en ontwerpen. In het Proof-of-Concept worden de geselecteerde technieken uit het adviesrapport getoetst.

5 | Aanpak

De uitvoering van dit project zal bestaan uit meerdere delen. Allereerst zal er een literatuuronderzoek gedaan worden naar een selectie van Blockchain implementaties. Uit dit onderzoek zal een adviesrapport komen die aangeboden zal worden aan het bedrijf. Hieruit zal een keuze gemaakt worden op de manier waarop de onderdelen gerealiseerd zullen worden. Om uiteindelijk de geselecteerde technieken te toetsen zal er een Proof of Concept ontwikkeld worden.

5.1 Onderzoeksopzet

In de afstudeeropdracht wordt er een adviesrapport opgesteld waarin advies wordt gegeven over de realisatie van het Proof of Concept dat betrekking heeft tot de implementatie van een Blockchain implementatie met de onderdelen Distributed Network en Identity Management. Door kwalitatieve methodieken toe te passen wordt er een technische beschrijving opgesteld van de verschillende onderdelen in de geselecteerde Blockchain implementaties.

5.1.1 Dataverzameling

Er wordt onderzoek gedaan door middel van het uitvoeren van deskresearch. Er zullen specifieke cases, implementaties van de Blockchain technologie, geselecteerd worden aan de hand van de criteria die gesteld is in 'Inclusie- en exclusiecriteria'. Voor het opdoen van voorkennis zullen er gepubliceerde research papers, wiki's en beschikbare courses doorlopen worden. Hierna zal er een selectie van Blockchain implementaties gemaakt worden die bestudeerd zullen worden in het onderzoek.

5.1.2 Dataomschrijving

Om de scope van het onderzoek te beperken met betrekking tot de beschikbare tijd wordt er een selectie van drie Blockchain implementaties gemaakt. Om tot deze selectie te komen zal er een lijst van de top 20 cryptocurrencies opgesteld worden en onderzocht worden op de beschreven inclusie-en exclusiecriteria.

Inclusie- en exclusiecriteria

De implementaties zijn in eerste instantie geselecteerd op de aanwezigheid van het onderdeel Identity Management. Daarnaast spelen de attributen open-source, of er een technische White paper beschikbaar is en het gebruikte consensus algoritme een rol tijdens de selectie van de vijf implementaties. Om diverse implementaties in kaart te brengen voor het uitbrengen van een zo goed mogelijk advies is het van belang dat de onderdelen Identity Management en Distributed Network op diverse wijze zijn geïmplementeerd. Hiervoor zijn onderstaande criteria vastgesteld.

Hard-forks

Een hard fork ((blockchain), 2010) is in essentie een aftakking van een bestaande blockchain door wijzigingen in de huidige structuur van de blockchain. Dit komt bijvoorbeeld voor als er een fout in de Blockchain ontdekt of misbruikt wordt. Aangezien de implementaties hiervan niet afwijken van de originele Blockchain worden hard forks niet meegenomen in het onderzoek.

Consensus algoritme

Een van de bepalende factoren van de inrichting van het onderdeel Distributed Network is het gebruik van het consensus algoritme. Dit bepaalt in hoe de verschillende verbonden cliënten overeenstemming krijgen over de waarheid van de blockchain (Konstantopoulos, 2017). Om een compleet beeld te schetsen is het nodig om implementaties te selecteren met verschillende consensus algoritmes.

5.1.3 Analysemethode

Om te bepalen welke technieken gebruikt kunnen worden vanuit bestaande Blockchain implementaties zal er deskresearch uitgevoerd worden. Hierbij worden de werkingen van de onderdelen Distributed Network en Identity Management onderzocht en technisch beschreven.

5.2 Adviesrapport

Uit het onderzoek zal een adviesrapport komen over de manieren waarop de onderdelen Identity Management en Distributed Network opgesteld zijn binnen de onderzochte im-

plementaties. Door het overzichtelijk maken van de resultaten uit het onderzoek zal het makkelijker zijn voor het bedrijf om een keuze te maken over de manier waarop de onderdelen gerealiseerd zullen worden.

5.3 Proof-of-Concept

Om de geselecteerde keuze(s) te toetsen zal er uiteindelijk een proof-of-Concept van de onderdelen Identity Management en Distributed Network gerealiseerd worden. Het is belangrijk dat de integriteit van deze onderdelen zo goed mogelijk bewaakt worden, waardoor er veel tijd besteed zal worden aan het testen van de implementaties. Om kennis op te doen voor het testen, ontwikkelen en ontwerpen van een blockchain implementatie zal er een selectie gemaakt worden van de gebruikte methoden, toegepaste technieken en benodigde tools.

6 | Planning

Voor de uitvoering van het project is er een globale planning gemaakt die zowel de benodigde documenten en feedback momenten bevat als de werkzaamheden die verricht worden gedurende de opdracht. De planning is hieronder weergegeven in tabel 6.1.

Tabel 6.1: Planning

Mijlpaal	Duur in dagen	
Orientatie	10d	
Opstart	2d	
Vooronderzoek	4d	
Plan van Aanpak	4d	
Onderzoek	25d	
Selectie implementaties	2d	
Theoretisch kader	3d	
Implementatie #1	7d	
Implementatie #2	7d	
Implementatie #3	6d	
Adviesrapport	10d	
Orientatie indeling	2d	
Schrijven	7d	
Voorleggen	1d	
Selecteren methoden	5d	
Selectie taal	1d	
Ontwikkelomgeving	2d	
Testen	2d	
Ontwikkeling	25d	
Distributed Network	12d	
Identity Management	13d	
Testen	5d	
Integratie	5d	
Overdracht	5d	

Literatuur

(blockchain), F. (2010). *Fork (blockchain) wikipedia, the free encyclopedia*. Verkregen van [https://en.wikipedia.org/wiki/Fork\(blockchain\)](https://en.wikipedia.org/wiki/Fork(blockchain)) ([Online; geraadpleegd op 22 februari 2018])

Konstantopoulos, G. (2017). *Understanding blockchain fundamentals, part 2: Proof of work & proof of stake*. Verkregen van <https://medium.com/loom-network/understanding-blockchain-fundamentals-part-2-proof-of-work-proof-of-stake-b6ae907c7edb> ([Online; geraadpleegd op 22 februari 2018])

IV Onderzoeksrapport

Blockchain: Identity Management en Distributed Network

Onderzoeksrapport

Jeffrey van Hoven
31 mei 2018

Samenvatting

In dit onderzoeksrapport is het resultaat te vinden van het kwalitatieve onderzoek naar de Blockchain onderdelen Identity Management en Distributed Network. Binnen het onderzoek is er gekeken naar de Blockchain implementaties Bitcoin, Cardano, Monero en EOS. Aan de hand van de volgende hoofdvraag is er geïnventariseerd over de gebruikte protocollen binnen de implementaties.

Welke protocol implementaties kunnen toegepast worden om de onderdelen Distributed Network en Identity Management te realiseren voor een Blockchain implementatie?

Samenvattend zijn de volgende protocollen geïdentificeerd voor de individuele onderdelen:

Protocol	Toelichting
Kademlia	Een bestaand protocol gerealiseerd door Maymounkov en Mazieres (2002). Dit protocol heeft een aantal wijzigingen binnen Cardano, zoals het versturen van informatie gaat over TCP/IP en er is een uitbreiding gemaakt op de manier waarop identificatiecodes toegekend worden aan deelnemers om een mogelijke sybil attack uit te sluiten.
Bitcoin	Communicatie verloopt over TCP/IP waarbij informatie wordt verstuurd door <i>inv</i> , <i>tx</i> , <i>block</i> en <i>getdata</i> berichten.
Monero	Focust op de privacy van de gebruiker en maakt gebruik van The Invisible Internet Project (I2P) om deze anonimiteit binnen het netwerk te waarborgen.

Protocol	Toelichting
Bitcoin	Maakt gebruik van het UTXO-model, waarin public- en private keys gebruikt worden om de betaler en ontvanger te registreren binnen een transactie. Door het gebruik van het analysemodel gepresenteerd door Reid en Harrigan (2013) is aangetoond dat het Bitcoin niet aan de niet aan de untraceability en unlinkability eis voldoet.
Cardano	Maakt gebruik van het UTXO-model, waarin public- en key cryptografie gebruikt wordt. Er is hierbij geen studie gevonden die aantoont dat het voldoet aan de untraceability en unlinkability eis, maar heeft aanzienlijke overeenkomsten met hoe Bitcoin omgaat met de identiteit.
EOS	Maakt gebruikt van het Account-model, waarin een gebruiker een unieke naam van maximaal twaalf karakters hanteert als identiteit. Daarnaast hanteert EOS de volgende componenten: <ol style="list-style-type: none">1. Role Based Permission Management2. Actions & Handlers

Inhoudsopgave

1	Inleiding	1
2	Probleemstelling	2
3	Opzet	3
4	Resultaten	4
4.1	Soorten netwerken	4
4.1.1	Proof of Work	5
4.1.2	Proof of Stake	6
4.2	Gevaren	7
4.2.1	Eclipse Attack	7
4.2.2	Majority Attack	7
4.2.3	Denial of Service (DoS)	7
4.2.4	Sybil Attack	8
4.2.5	Double spending	8
4.2.6	Nothing at Stake	8
4.3	Identiteit	9
4.4	Bitcoin	10
4.4.1	Functionaliteit	10
4.4.2	Gevaren	11
4.4.3	Identiteit	12
4.5	Cardano	14
4.5.1	Functionaliteit	14
	Informatie propagatie	14
4.5.2	Gevaren	15
4.5.3	Identiteit	16
4.6	EOS	17
4.6.1	Functionaliteit	17
4.6.2	Gevaren	17
4.6.3	Identiteit	17
4.7	Monero	18
4.7.1	Functionaliteit	18
4.7.2	Gevaren	19
4.7.3	Identiteit	19
5	Conclusie	21
5.1	Deelvragen	21
5.2	Hoofdvraag	25

5.2.1	Distributed Network	25
5.2.2	Identity Management	26

Lijst van figuren

4.1	Proof-of-Work in Bitcoin	5
4.2	UTXO-model	10
4.3	Communicatie tussen deelnemers in Bitcoin	11
4.4	Kademlia Binary Tree	14

Woordenlijst

Symbolen

o-confirmation double spending Vorm van double spending waarbij het validatieproces omzijlt wordt. 7

A

account Een combinatie van public- en private keys waarbij de public key als identificatie gebruikt wordt. 10

B

block races Hiermee wordt de onderlinge competitie bedoeld van miners, waarbij ze “racen” om als eerste een block te produceren. 7

bloom filter . 12

bootstrap node Vaststaande nodes waarvan hun adressen geregistreerd zijn in de broncode van de Blockchain en gebruikt worden om als eerste connectiepunt te functioneren. 11

D

difficulty Een netwerk instelling dat beïnvloed hoe moeilijk het proof-of-work op te lossen is door het aanpassen van de nonce. 5

double spending Aanval waarbij geprobeerd wordt om reeds uitgegeven tokens nogmaals uit te geven. v, 10, 19

E

elector Groep van nodes die verantwoordelijk is voor het kiezen van slot leaders. 6

F

fork Splitsing in het netwerk dat veroorzaakt is door een kleine wijziging in het protocol. 8

K

key Image Structuur dat gebruikt om een transactie met ring signatures te valideren. 19

M

miner Een deelnemer of een groep van deelnemers die verantwoordelijk zijn voor de creatie en validatie van nieuwe blocks. Miners worden doorgaans beloond voor hun services in de vorm van tokens. v, 11

mining node Node als enige taak heeft om de rol van miner te vervullen. vi

minting Een benaming voor de manier waarop een nieuw block gegenereerd wordt bij een Proof of Stake algoritme. 6

N

node Computer dat in verbinding staat met het netwerk van de Blockchain. v, vi, 10, 11, 14, 18

nonce Een 4-byte veld waarvan de waarde ingesteld wordt zodat de hash van een block een reeks van nullen bevat. De rest van de inhoud van een block staat hierdoor vast. v, 5

P

peer Synoniem voor node. v, 18

peer list Lijst van peers waarmee connectie is gemaakt. 11

R

ring signature Een digitale handtekening dat gebruikt wordt om een bericht mee te ondertekenen, waarbij het niet mogelijk is om terug te leiden wie het heeft ondertekend. v, 19, 24

S

selfish mining Aanval waarbij er door een kwaadwillende mining node blocks achtergehouden worden. 7, 22

slot leader Block producer die verantwoordelijk is voor het creëren van een block. v, 6, 14

spend key Onderdeel van een account in Monero en is benodigd om ADA uit te geven.. vi, 19, 24

stake Investeren in de Blockchain proportioneel naar het type consensus, meestal gebruikt in PoS implementaties. vi, 6

stealth address Een eenmalig te gebruiken public-key die afgeleid wordt vanuit de view key en de spend key. 19, 24

sybil attack Aanval op het netwerkgedeelte van een Blockchain waarbij er virtuele deelnemers gecreëerd worden op het netwerk om ze de processen te beïnvloeden. i, 25

T

token Abstracte term voor data die verstuurd wordt over het netwerk van de Blockchain, meestal wordt hiermee een cryptocurrency bedoeld. 6, 8

tunnel . 17, 18, 19, 22

U

UTXO-model Gegevenstructuur voor transacties waarbij een transactie bestaat uit een lijst van inputs en outputs. 14, 17

V

view key Onderdeel van een account in Monero en wordt gebruikt om een derde partij inzicht te geven in gedane transacties. vi, 19, 24

voting power Hoeveel zeggenschap een node heeft in het netwerk gebaseerd op attributen als hashing power en stake. 5, 7

W

wallet Software die alle adressen en secret keys bijhoudt. Het wordt gebruikt om tokens te versturen, ontvangen en op te slaan. 19

wallet (node) Node die een gereduceerde staat van het Blockchain bevat, waarin alleen de transacties opgenomen worden die betrekking hebben op de public- en private key combinatie. 12

Afkortingen

B

BFT Byzantine Fault Tolerance. 4

D

DHT Distributed Hash Table. 14

DoS Denial of Service. ii, 7, 12

DPoS Delegated Proof of Stake. 6

I

I2NP I2P Network Protocol. 18, 19

I2P The Invisible Internet Project. i, 17, 18, 22, 25

P

PoS Proof of Stake. vi, 6, 15, 22

PoW Proof of Work. 5, 6, 7, 11

T

tx transactie. 11

1 | Inleiding

Dit onderzoeksrapport is opgesteld in het kader van een afstudeeropdracht gedaan in opdracht van Quintor. Het document betreft een onderzoek naar de Blockchain onderdelen Identity Management en Distributed Network. In dit document zijn de resultaten van het onderzoek naar hoe de onderdelen Distributed Network en Identity Management gerealiseerd zijn in de Blockchain protocollen EOS, Cardano, Bitcoin en Monero te vinden.

In hoofdstuk 4.1 is te vinden welke soorten gedistribueerde netwerken er gebruikt worden in de gekozen implementaties, waarna er in hoofdstuk 4.2 een introductie wordt gegeven over de mogelijke gevaren van gedistribueerde netwerken binnen Blockchain. In hoofdstuk 4.3 wordt er een introductie gegeven over identiteit binnen Blockchain implementaties. Vervolgens wordt er in hoofdstuk 4.4 het Bitcoin protocol behandeld, in hoofdstuk 4.5 het Cardano protocol, in 4.6 het EOS protocol en in 4.7 het Monero protocol. Uiteindelijk wordt er in de conclusie de hoofdvraag beantwoord.

2 | Probleemstelling

Aangezien de toepassing en adoptie van Blockchain technologie steeds groter wordt wil Quintor de toepassingsmogelijkheden en technieken onderzoeken om zo inzicht te kunnen krijgen in hoe het gebruikt kan worden in de aangeboden vraagstukken vanuit klanten. Dit brengt zich tot het probleem, namelijk dat Quintor onvoldoende kennis heeft van het Blockchain domein om de toepassing ervan te kunnen adviseren in vraagstukken vanuit klanten.

3 | Opzet

Het onderzoek dient voor het opstellen van het adviesrapport waarin protocollen worden gepresenteerd aan Quintor die mogelijk geïmplementeerd kunnen worden tijdens de realisatie van het Proof-of-Concept. Het betreft exploratief onderzoek waarin case-study gebruikt wordt om een gedetailleerde omschrijving van de onderdelen Identity Management en Distributed Network op op te stellen van de Blockchain implementaties Bitcoin, Cardano, EOS en Monero. De kennis die hiermee wordt opgebouwd kan eventueel gebruikt worden in vervolgonderzoek.

In het onderzoek staat de onderstaande hoofdvraag centraal:

Welke protocol implementaties kunnen toegepast worden om de onderdelen Distributed Network en Identity Management te realiseren voor een Blockchain implementatie?

Omdat de hoofdvraag te groot is om in een keer te beantwoorden is het opgesplitst in de volgende deelvragen:

1. "Welke soorten gedistribueerde netwerken worden er gebruikt?"
2. "Hoe werken de gedistribueerde netwerken en tegen welke gevaren zijn ze bestendig?"
3. "Hoe wordt er omgegaan met de identiteit van de gebruiker?"

4 | Resultaten

4.1 Soorten netwerken

In dit hoofdstuk wordt er onderzocht welke verschillende netwerken er gebruikt worden in bestaande implementaties. Hierbij wordt zowel de definitie van soorten en de selectie van implementaties gebruikt uit de resultaten van het vooronderzoek.

"De distributie van informatie en het probleem van wederzijdse overeenstemming over een consistente staat van het netwerk vormt een uitdaging, zeker in de aanwezigheid van zelfzuchtige en/of kwaadwillende deelnemers- en B. Scheuermann (2016). Het is een uitdaging die bekend staat als het Byzantine Generals' Problem, en is beschreven door Lamport et al. (1982). Het stelt dat het essentieel is voor een betrouwbaar computersysteem om te kunnen gaan met fouten die optreden in een of meer van de componenten, waardoor het kan voorkomen dat er conflicterende informatie verstuurd wordt naar de andere componenten van het systeem. In hoeverre een computersysteem hiermee om kan gaan wordt de Byzantine Fault Tolerance (BFT) genoemd en wordt aangeduid als: $f = \lfloor \frac{N-1}{t} \rfloor$ waarbij N componenten van een computersysteem zijn en t de foutieve componenten.

In blockchain implementaties zijn de componenten die onbetrouwbaar zijn de deelnemers van het peer-to-peer netwerk. Het soort netwerk is dan ook verbonden met de manier waarop consensus bereikt wordt tussen de deelnemers van het netwerk en is getypeerd als het consensus protocol dat geïmplementeerd is.

4.1.1 Proof of Work

De originele implementatie van Blockchain technologie is gepresenteerd door Nakamoto (2008) in "*Bitcoin: A peer-to-peer electronic cash system*". Het maakt gebruik van een algoritme genaamd Proof of Work (PoW) om consensus te bereiken. Hierbij gaat het om het oplossen van een wiskundig probleem $Y \in \mathbb{N} < f(X + n)$ waarbij f een hash functie is, n de nonce, X de data en Y de difficulty.

In het geval van Bitcoin is de Y waarde een getal die aangeeft wat de difficulty is om de hash te berekenen en wordt de X waarde incrementeel opgehoogd. Een voorbeeld is gegeven in fig. 4.1. Dit proces zorgt ervoor dat de integriteit van de data in een block op de Blockchain bewaakt wordt. Wanneer een kwaadwillende deelnemer aan het netwerk de data van een block wilt aanpassen die reeds opgenomen is in de Blockchain, kan er via het PoW makkelijk gevalideerd worden of het block invalide is.

```
"Hello, world!0" => 1312af178c253f84028d480
"Hello, world!1" => e9afc424b79e4f6ab42d99c
"Hello, world!2" => ae37343a357a8297591625e
...
"Hello, world!4248" => 6e110d98b388e77e9c6f
"Hello, world!4249" => c004190b822f1669cac8
"Hello, world!4250" => 0000c3af42fc31103f1f
```

Figuur 4.1: Werking Proof-of-Work, van Bitcoin Wiki (2010). Wanneer de eerste vier bits ($Y = 4$) van de hash 0 zijn is de proef opgelost.

Daarnaast beschrijft de bedenker van het protocol, Satoshi Nakamoto, het PoW algoritme als 'one-CPU-one-vote'. Aangezien het gebruikte hashing algoritme geen limitaties stelt tot de zogeheten voting power van een deelnemer in het netwerk creëert het gunstige omstandigheden voor high-end GPU eigenaren tegenover high-end CPU eigenaren (Van Saberhagen, 2013, p. 2).

Monero maakt gebruik van het CryptoNight algoritme (Noether et al., 2014), een implementatie gebaseerd op CryptoNote, waarin gebruik gemaakt wordt van een egalitair Proof of Work (Van Saberhagen, 2013, p. 11). In contrast met het Bitcoin protocol Proof of Work algoritme is het ontworpen om inefficiënt berekenbaar te zijn op een GPU, waardoor er gelijke kansen zijn voor de deelnemers van het netwerk die het mining proces uitvoeren.

4.1.2 Proof of Stake

"Een eerste overweging met betrekking tot de werking van blockchain protocollen gebaseerd op Proof of Work – zoals Bitcoin – is de energie benodigd voor hun uitvoering.- Kiayias et al. (2017). In een onderzoek gedaan door O'Dwyer en Malone in 2014 naar het energieverbruik van het Bitcoin mining netwerk is geschat dat onder redelijke omstandigheden het netwerk gelijk stond met het energiegebruik van Ierland. Om deze reden zijn er onderzoeken en experimenten gedaan naar alternatieve consensus algoritmes. Proof of Stake (PoS) is een consensus algoritme waarbij, in plaats van het verspillen van elektriciteit om zware rekenkundige problemen op te lossen, een deelnemer geselecteerd wordt om het volgende blok te genereren (doorgaans minting genoemd) op basis van willekeurige selectie en rijkdom of leeftijd (i.e., de stake).

Cardano maakt gebruik van PoS waarbij iedere deelnemer van het netwerk met een positief balans (e.g. stake) als stakeholders gezien worden. Om uitgekozen te worden om een nieuw blok te genereren moet een stakeholder geselecteerd worden als slot leader. De implementatie verdeelt de fysieke tijd in tijdvakken en elke tijdvak is verdeeld in slots. Voor elke slot wordt een slot leader verkozen, die verantwoordelijk is voor het produceren van één blok. Niet alle deelnemers van het netwerk, bijvoorbeeld die minder dan 2% van de totale circulatie van tokens hebben, worden geselecteerd om benoemd te worden tot slot leader. Deze groep van deelnemers maken deel uit van de electors groep. Electors kiezen nieuwe slot leaders gedurende het huidige tijdsvak, waarna er een selectie gemaakt wordt en de nieuwe slot leaders vaststaan voor het volgende tijdsvak. Hoe meer stake een deelnemer heeft, hoe groter de kans dat zij uitgekozen wordt om een slot leader te worden in het volgende tijdsvak. De slot leader luistert naar transacties die aangekondigd worden door andere nodes, bundelt ze in een nieuw blok, signeert het met zijn private key en publiceert het blok in het netwerk (Cardano Docs, 2013c).

EOS is een implementatie die gebruik maakt van Delegated Proof of Stake (DPoS) om consensus te bereiken. Het grote verschil tussen DPoS en PoS; in een PoS systeem is elke deelnemer die stake heeft maakt onderdeel uitmaken van het validatie- en consensusproces. Met DPoS kan elke deelnemer die stake heeft andere deelnemers verkiezen die onderdeel uitmaken van het validatie- en consensusproces (Roman, K., 2018). In contrast met het PoW algoritme is er geen competitie voor het produceren van een blok, maar wordt er samengewerkt om een blok te produceren.

4.2 Gevaren

Wanneer deelnemers uitmaken van een grootschalig netwerk die niet gecontroleerd wordt door een centrale autoriteit kan het voorkomen dat deelnemers zich misdragen. In juli 2016 is Ethereum opgesplitst in twee partities die dezelfde valuta hanteren; *Ethereum* en *Ethereum Classic*. Dit is veroorzaakt door een kwaadwillende deelnemer in het netwerk die door een bug in het systeem geld naar zichzelf toe kon sturen. Dit heeft ertoe geleid dat veel gebruikers mogelijk een aanzienlijk verlies geleden hebben, waaronder veel ontwikkelaars van Ethereum. Om dit verlies op te lossen werd er een hard-fork voorgesteld die Ethereums code aanpast waarbij de transacties van de kwaadwillende deelnemer teruggedraaid werden (Kiffer, Levin & Mislove, 2017).

Dit illustreert een van de mogelijke manieren waarop een kwaadwillende gebruiker het systeem kan ondermijnen. Om een duidelijk overzicht te geven van de gevaren binnen een gedecentraliseerd peer-to-peer systeem wordt er onderzocht welke technieken toegepast worden om aanvallen van een kwaadwillende deelnemer van het netwerk tegen te gaan.

4.2.1 Eclipse Attack

Een aanval op het peer-to-peer netwerk waarbij er controle over een deelnemer zijn toegang tot informatie gelimiteerd, of zelfs gemanipuleerd wordt. Met de juiste manipulatie van het peer-to-peer netwerk kan er informatie verduistert worden zodat een goedwillende deelnemer aan het netwerk alleen maar kan communiceren met kwaadwillende deelnemers. Dit kan leiden tot block races, selfish mining en o-confirmation double spending (Heilman, Kendler, Zohar & Goldberg, 2015).

4.2.2 Majority Attack

Een aanval waarbij één deelnemer de richting van het netwerk bepaald door het bezitten van 51% de voting power. In het geval van Proof of Work betekend dit dat de kwaadwillende deelnemer 51% van de totale rekenkracht nodig heeft om deze aanval uit te voeren. Dit stelt de kwaadwillende deelnemer in staat om het netwerk te manipuleren en kan leiden tot o-confirmation double spending.

4.2.3 Denial of Service (DoS)

Een algemene benaming voor een collectie van mogelijke oorzaken voor een bewuste verstoring van de services die het peer-to-peer netwerk faciliteert. Dit kan op meerdere ma-

nieren optreden, bijvoorbeeld door het invoegen van heel veel transacties in één block, zodat het lang duurt voordat het peer-to-peer netwerk het nieuwe block heeft opgenomen.

4.2.4 Sybil Attack

Een aanval waarbij een deelnemer meerdere virtuele deelnemers creëert in het netwerk waarbij de gecreëerde deelnemers het verkiezingsproces kunnen verstoren door verkeerde informatie door te geven in het netwerk, zoals positief stemmen voor een malafide transactie (Conti, Lal, Ruj et al., 2017).

4.2.5 Double spending

Bij Creditcard-gebaseerde betalingen wordt er eerlijkheid bereikt door het bestaan van een bank of een andere vertrouwde tussenpersoon (e.g. Paypal). Hierbij wordt de tussenpersoon vertrouwd om te controleren dat diegene die een betaling doet aan een derde partij het geld niet al heeft uitgegeven (G. Karame, Androulaki & Capkun, 2012). In gedecentraliseerde systemen, waarbij er geen vertrouwde tussenpersoon aanwezig is, staat dit bekend als het *double spending* probleem, waarbij het mogelijk is om tokens die reeds uitgegeven zijn (i.e. opgenomen in een block) nogmaals gebruikt wordt om een transactie uit te voeren.

4.2.6 Nothing at Stake

Wanneer er een fork ontstaat is de optimale strategie elke replica van de blockchain te valideren, zodat de diegene die het validatie proces uitvoert nog steeds uitbetaald krijgt, ongeacht of de fork geaccepteerd wordt of niet.

4.3 Identiteit

Blockchain kan een zeker mate van privacy garanderen door de public en private keys, wat ervoor zorgt dat een gebruiker niet zijn echte identiteit hoeft te hanteren om met het systeem te interacteren. Echter, Meiklejohn et al. (2013) toont aan dat blockchain niet de transactionele privacy kan waarborgen omdat de waarden van alle transacties en saldo van elke public key openbaar inzichtbaar zijn.

Okamoto (1992) beschrijft zes criteria waaraan de ideale implementatie van elektronisch geld moet voldoen. In het bijzonder worden er twee criteria genoemd:

- **Untraceability:** voor elke inkomende transactie hebben alle mogelijke afzenders gelijke kansen om geïdentificeerd te worden als verstuurder.
- **Unlinkability:** voor elke twee uitgaande transacties moet het onmogelijk zijn om aan te tonen dat ze naar dezelfde persoon verstuurd zijn.

4.4 Bitcoin

4.4.1 Functionaliteit

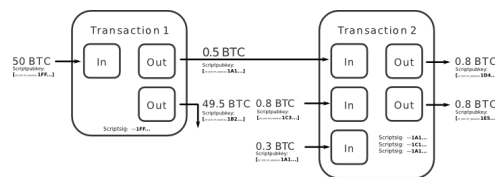
Architectuur Bitcoin is een netwerk waarin geen coördinerende rollen zijn. Elke deelnemer van het netwerk heeft een complete replica van alle informatie die benodigd is voor het verifiëren van de validiteit van binnenkomende transacties. Er zijn verschillende services die het netwerk faciliteert, twee daarvan zijn met name belangrijk voor de beschrijving van het netwerk: netwerk routing, en het mining proces. In de basis van het netwerk staan de transacties die op abstract niveau bitcoins van een of meer accounts naar een of meer bestemmingsaccounts overmaken. Een account, in de context van het bitcoin netwerk, is een combinatie van een public- en private key, waarbij de public key als identificatie van de account gebruikt wordt. Om een transactie te versturen wordt de transactie gesigneerd met de private key van de account die de transactie wilt uitvoeren.

Transacties bestaan uit een input en output. In plaats van het aggregeren van een balans voor elk account, wordt er bijgehouden wat de output van een transactie is. De balans is hierbij de som van alle openstaande outputs van het desbetreffende account. In fig. 4.2 is te zien hoe dit in zijn werk gaat.

Een onderdeel van de services die de nodes binnen het netwerk aanbieden is het valideren van transacties. Hierbij worden drie onderdelen gevalideerd:

- Een output mag maar één keer geclaimd zijn.
- Nieuwe outputs worden alleen gecreëerd door een transactie.
- De som van alle waarden van de geclaimde outputs moet groter zijn als de totale som van de nieuwe gecreëerde outputs.

Wanneer dit het geval is wordt de transactie geaccepteerd en opgenomen in de lokale replica van de blockchain. Over tijd kan het voorkomen dat de replica van verschillende nodes inconsistent worden, waarbij het kan voorkomen dat er twee of meer transacties dezelfde coin meerdere malen uitgeeft. Dit staat bekend als double spending (Decker & Wattenhofer, 2013).



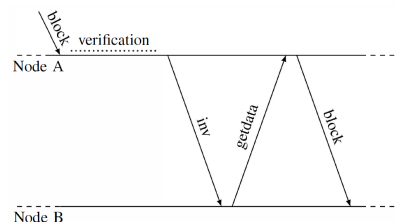
Figuur 4.2: Voorbeeld van het UTXO-model zoals in gebruik bij Bitcoin, bron: <http://news.8btc.com/thoughts-on-bytom-design-extension-of-utxo-structure>.

Een nieuw block wordt gecreëerd door het uitvoeren van het mining proces. Dit wordt uitgevoerd door zogenaamde miners node. Om te bepalen welke node verantwoordelijk is voor het volgende block moet er een oplossing gevonden worden voor het proof-of-work. Dit proces zorgt ervoor dat er een beslissing gemaakt wordt over de volgorde van de transacties, en dat de inhoud van een block niet aangepast kan worden omdat dit in directe verbinding staat met het gedane PoW.

Discovery protocol Om het het netwerk te betreden worden er DNS servers benaderd waarbij gebruik wordt gemaakt van het TCP protocol. Deze DNS servers worden in stand gehouden door vrijwilligers en geven een willekeurige set aan bootstrap nodes terug die actief zijn in het netwerk. Wanneer de node toegetreden is tot het netwerk wordt er een peer list bijgehouden met alle nodes waarmee er connectie is gelegd. Deze peer list wordt gebruikt om connectie te leggen bij een eerstvolgende toetreding tot het netwerk.

Informatie propagatie Voor het updaten en synchroniseren van de blockchain worden er transactie (tx) en block berichten verstuurd. Om tegen te gaan dat tx- en block berichten verstuurd worden naar nodes die al afweten van deze informatie, wordt er een *inv* bericht verstuurd wanneer een transactie of een block volledig geverifieerd is. Het *inv* bericht bevat een lijst van transactie- en block hashes die reeds ontvangen zijn door de verstuurder en die beschikbaar zijn om opgehaald te worden.

Wanneer een node deze informatie wilt ontvangen (bijv. omdat het de informatie nog niet heeft), wordt er een *getdata* bericht verstuurd naar de verstuurder van het *inv* bericht, met daarin de hashes van de informatie die de node wilt hebben. Fig. 4.3 visualiseert dit proces.



Figuur 4.3: Berichten die verzonden worden om informatie over een block uit te wisselen (Decker & Wattenhofer, 2013, p. 4).

4.4.2 Gevaren

Majority Attack Nakamoto (2008) stelt dat het uitvoeren van een majority attack op het netwerk onpraktisch is omdat het uitvoeren ervan niet opweegt tegen de kosten voor de benodigde hardware om de rekenkracht te behalen die hiervoor nodig is. Dit blijkt niet altijd het geval, Eyal en Sirer (2014) beschrijft namelijk een strategie genaamd Selfish Mining waarbij er gevalideerde blocks achtergehouden worden voor het netwerk waardoor er opzettelijk een fork wordt gecreëerd. De eerlijke miners zullen verder werken aan de publiekelijke blockchain terwijl de uitvoerder van het Selfish Mining strategie verder werkt op de achtergehouden blockchain. Als de uitvoerder meer blokken ontdekt ontstaat er een voorsprong op de publiekelijke blockchain en worden de blocks nog steeds achtergehouden. Wanneer de lengte van de publiekelijke blockchain de lengte van de achtergehouden

blockchain benaderd, zal de uitvoerder de blockchain publiceren. Dit leidt ertoe dat miners die het Bitcoin protocol volgen hun middelen verspillen aan het minen van cryptopuzzles die er niet toe doen.

Denial of Service Over de jaren heen zijn er kwetsbaarheden in het Bitcoin protocol geïdentificeerd die het mogelijk maken om een DoS aanval uit te voeren. De meest recente¹ aanval (NIST, 2013) exploiteert een zwakte in de implementatie van een Bloom filter, een filter die onder andere gebruikt wordt door wallets om alleen transacties binnen te halen waarbij de deelnemer betrokken is. Hierdoor was het mogelijk om een sequentie van berichten te sturen die ervoor zorgde dat een volledige node binnen het netwerk overbelast werd.

Eclipse Attack Heilman et al. heeft aangetoond dat Bitcoin's peer discovery mechanisme toegankelijk is voor een *Eclipse attack*. Door de manier waarop het Peer Discovery mechanisme werkt is het mogelijk om de lijst van connecties zo te manipuleren dat nieuwe deelnemers doorgestuurd worden naar kwaadwillende deelnemers.

Double spending G. O. Karame, Androutsaki en Capkun toont aan dat het in het beginstadium van het Bitcoin protocol mogelijk was om via zogenaamde 'fast payments' een double spending aanval uit te voeren.

4.4.3 Identiteit

Er zijn drie onderdelen van het Bitcoin systeem die interessant zijn voor het analyseren van het systeem in relatie tot de identiteit van de gebruiker. Ten eerste is de gehele historie van Bitcoin transacties publiekelijk in te zien. Zoals eerder vermeld is dit nodig om zonder centrale autoriteit validatie van de transacties te doen. Het tweede is het UTXO-model dat gebruikt wordt om uitgaves en inkomsten bij te houden. In dit model bestaat een transactie uit meerdere inputs en outputs, waarbij de input een eerdere output van een transactie is geweest. Ten derde zijn de betaler en de ontvanger van een transactie gekoppeld aan de transactie door middel van een public key.

Reid en Harrigan (2013) stelt dat deze drie onderdelen, met name de publieke toegankelijkheid van de Bitcoin transacties en de input-output relatie tussen transacties en public keys, ingedeeld kunnen worden in twee verschillende netwerken die samen opereren, het *transaction network* en het *user network*. Waarbij het *transaction network* de stroom van Bitcoins beschrijft tussen transacties over de tijd, en het *user network* tussen gebruikers over de tijd. Door het analyseren van de structuur van deze twee netwerken aan de hand

¹Er zijn recentere aanvallen op het Bitcoin protocol geweest waarbij er DoS aanval heeft plaatsgevonden maar deze zijn niet nader gespecificeerd, zie: "Common Vulnerabilities and Exposures - Bitcoin Wiki".

van de informatie uit het Bitcoin netwerk, is er geconcludeerd dat het mogelijk is om verschillende public keys met elkaar te associëren, en het met de juiste middelen het mogelijk is om de activiteit van een gebruiker gedetailleerd in kaart te brengen.

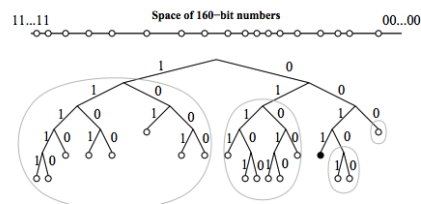
Hierbij voldoet het bitcoin protocol met name niet aan de de untraceability eis. Alle transacties die gedaan worden tussen de deelnemers van het netwerk zijn publiekelijk in te zien en elke transactie kan herleid worden naar de verstuurder en ontvanger. Ook is het indirect mogelijk om twee uitgaande transacties naar dezelfde persoon aan te tonen binnen het netwerk.

4.5 Cardano

4.5.1 Functionaliteit

Architectuur Net zoals bij Bitcoin zijn de transacties de kern van de implementatie, waarbij er wederom gebruik wordt gemaakt van het UTXO-model zoals beschreven bij de architectuur van Bitcoin. De architectuur van het Cardano netwerk bestaat uit drie soorten nodes die fundamenteel zijn voor de werking van het protocol: *core*, *relay* en *edge* nodes. *Core* nodes zijn de kern van het netwerk. Het zijn de enige nodes die geselecteerd kunnen worden om slot leader te worden, waardoor het de enige nodes zijn die een block kunnen creëren. *Relay* nodes worden gezien als de proxy tussen core nodes en het internet. Ze hebben geen stake in het netwerk, waardoor ze makkelijk te verplaatsen of veranderd kunnen worden. *Edge* nodes zijn de simpele nodes die iedereen kan uitvoeren. Deze nodes kunnen transacties aanmaken binnen het netwerk en aanbieden aan *core* nodes via de *relay* nodes (Cardano Docs, 2013a, Topology).

Discovery protocol Om het netwerk te betreden wordt er gebruik gemaakt van een bestaand protocol genaamd Kademlia, wat gebaseerd is op het gebruik van een Distributed Hash Table (DHT) architectuur. Elke node wordt behandeld als een tak in een Binary Tree waarbij de positie van een node bepaald wordt door een unieke prefix van de identificatie code van een node. In fig. 4.4 is de positie van een node met de prefix 0011 te zien. Het protocol garandeert dat elke node in verbinding staat met een andere node. Met deze garantie kan elke node een andere node lokaliseren aan de hand van de identificatie code (Maymounkov & Mazieres, 2002, p. 2).



Figuur 4.4: Binary Tree zoals in gebruik bij het Kademlia protocol, Maymounkov en Mazieres (2002).

Informatie propagatie

Berichten worden verstuurd voor het uitwisselen van informatie tussen deelnemers. Hierbij zijn drie abstracte types gedefinieerd: *inv*, *req* en *data*. Net zoals bij Bitcoin wordt de *inv* message gebruikt om aan te geven dat er data beschikbaar is. Het *req* bericht wordt vervolgens gebruikt om beschikbare data op te vragen. De data wordt vervolgens verstuurd

via een *data* message. Berichten die bijvoorbeeld een block versturen zijn nader gespecificeerde *data* berichten. Op deze drie types zijn alle berichten in het netwerk gebaseerd, bijvoorbeeld is het *MsgBlock* bericht, die block informatie uitwisselt, gebaseerd op een *data* bericht (Cardano Docs, 2013b). Een bericht kan verstuurd worden naar drie verschillende mediums: het versturen van een bericht naar een node, de burens, en het gehele netwerk. Naar welk medium het bericht wordt verstuurd is opgenomen in de header van een bericht.

4.5.2 Gevaren

Sybil Attack Een fundamenteel probleem bij een implementatie van PoS, zoals beschreven door (Kiayias et al., 2017), is het simuleren van een leiderschapsverkiezing. Om een eerlijke, willekeurige verkiezing onder deelnemers van het netwerk te hebben is het nodig om een zekere mate van wanorde te introduceren. Mechanismes die benodigd zijn om deze wanorde te introduceren zijn gevoelig voor beïnvloedingen van kwaadwillende deelnemers in het netwerk.

Eclipse attack In het Kademlia netwerk is het mogelijk om een eclipse attack uit te voeren, maar wel lastig. In Cardano Docs (2013b) wordt uitgelegd hoe dit mogelijk zou zijn. Door de manier waarop het netwerk ingedeeld is, is het mogelijk, indien het netwerk constant blijft, om door veel nodes in het netwerk aan te maken de IDs rondom een bestaande node te bezitten, waardoor de communicatie met deze node te manipuleren is. Om dit tegen te gaan heeft Monero een uitbreiding gerealiseerd op het Kademlia protocol, waarbij node IDs vervangen worden door HashIds.

Een HashId is een binaire reeks van 32 bytes bestaande uit twee onderdelen. De nonce, een willekeurige 14 reeks aan bytes binaire reeks, en hashing data dat gegenereerd wordt aan de hand van de zogenaamde DerivingKey, een PBKDF2 hash dat gebruik maakt van HMAC (Hash-based Message Authentication Code) en een Salt, een SHA-512 hash (Cardano Docs, 2013a, P2P Layer, Addressing).

4.5.3 Identiteit

De Cardano implementatie is een public Blockchain waarbij alle transacties inzichtbaar zijn en elke deelnemer deel kan uitmaken van het consensus proces. Het maakt gebruik van public- en private keys om pseudonimiteit te waarborgen, waarbij de elliptic curve cryptografie implementatie Curve25519 toegepast wordt om de public- en private key te genereren. Binnen Cardano worden er verschillende adressen gebruikt om transacties van een bestemming te voorzien (Cardano Docs, 2013a, "Addresses in Cardano SL"):

1. **public key address**

Een base58 gecodeerde string van de public key dat gebruikt wordt als bestemming van een transactie.

2. **script address**

Wordt gebruikt voor het Pay to Script Hash principe, waarbij er in plaats van de public key gebruikt wordt als bestemming, een validatie script verstuurd wordt die gebruik maakt van een zogenaamde redemption script. Om de waarde van de transactie te claimen dient het validatie script positief uit te vallen.

3. **redeem address**

Wordt gebruikt voor het Pay to Public Key Hash principe, waarbij er een hash gecreëerd wordt wat ervoor zorgt dat de public key alleen publiekelijk geregistreerd wordt wanneer de output van een transactie wordt uitgegeven.

4.6 EOS

4.6.1 Functionaliteit

De blockchain implementatie EOS werkt toe naar een operating systeem speciaal voor blockchain toepassingen. In eerste instantie zal er een Blockchain gerealiseerd worden die dient als proof-of-concept van het ontwerp. In dit proof-of-concept is er een eerste versie gerealiseerd die het mogelijk maakt voor developers om een eigen applicatie op het EOS netwerk te creëren. Hierbij is de focus gelegd het faciliteren van functionaliteiten die betrekking hebben op account permissies, authenticatie en de communicatie tussen het internet en het netwerk. Er wordt gespeculeerd dat EOS een sterke concurrent van Ethereum zal worden als het gaat om Blockchain als een developer platform (Steemit , 2017).

Architectuur EOS maakt gebruik van aanpak waarbij extensies op de basis componenten (e.g. het netwerk, de 'chain', etc.) gerealiseerd worden als plugins. Dit maakt het zodat het protocol makkelijk te wijzigen is in de toekomst.

4.6.2 Gevaren

EOS is op dit moment nog niet in productie en heeft dan ook nog geen historie van mogelijke gevaren en de acties die hiertegen genomen zijn.

4.6.3 Identiteit

EOS is een consortium Blockchain waarin de identiteit van een gebruiker vastgelegd wordt in een account model, waarbij een account identificeerbaar is door een unieke naam van maximaal twaalf karakters. Handeling zijn gerestricteerd door middel van een Role Based Permissie systeem. Om dit mogelijk te maken dient een gebruiker allereerst geautoriseerd te zijn alvorens deel te kunnen nemen aan het netwerk. Centraal in de implementatie staat de notie van Actions & Handlers. Elk account (i.e. deelnemer) heeft een eigen database die alleen toegankelijk is door gedefinieerde action handlers. Dit systeem is soortgelijk aan smart contracts zoals in gebruik bij Ethereum.

4.7 Monero

4.7.1 Functionaliteit

Architectuur Monero maakt gebruik van The Invisible Internet Project (I2P) protocol. Het I2P protocol stelt het netwerk in staat om deelnemers te beschermen tegen een zekere mate van verkeer; waarbij de identiteit van de verstuurder en ontvanger verborgen wordt, terwijl er gebruik gemaakt wordt van encryptiestandaarden om de inhoud van berichten te verbergen en te garanderen dat het bericht aankomt (Zantout & Haraty, 2011). Het protocol ondersteunt zowel TCP/IP als UDP/IP communicatie, waarbij de Transport laag in het netwerk van Monero gelimiteerd is aan de mogelijkheden die I2P ondersteunt (Monero, 2017b). De transport laag faciliteert de connectie tussen de verschillende deelnemers in het netwerk. Om vervolgens te kunnen communiceren wordt er gebruik gemaakt van een tunnel. Elke deelnemer in het netwerk heeft minimaal twee Tunnels, een voor uitgaand- en inkomend verkeer. Wanneer er communicatie plaatsvindt tussen twee deelnemers zullen er vier tunnels aangemaakt worden; twee voor uitgaand verkeer en twee voor inkomend verkeer (Monero, 2017e). Ook Monero maakt gebruik van het UTXO-model, waarbij er bij iedere transactie twee keys aanwezig zijn; een spend key en een view key. Beide keys zijn onderdeel van een account, waarbij de spend key gebruikt wordt om geld uit te geven, en de view key gebruikt wordt om permissie te geven om de transacties in te zien van een deelnemer. De keys spelen een belangrijke rol in de privacy van de deelnemer omtrent transacties (Monero, 2017a).

Discovery protocol Het discovery protocol in gebruik bij Monero is soortgelijk aan de manier waarop Bitcoin het discovery proces uitvoert. Om het netwerk te bootstrappen wordt er gebruik gemaakt van nodes die vastgelegd zijn in de broncode, waarna er een lijst van peers wordt teruggegeven aan de deelnemer en de centrale node vergeten wordt. Het is ook mogelijk om zelf deelnemers vast te leggen waarna geprobeerd wordt om connectie te maken.

Informatie propagatie Alles binnen het I2P netwerk wordt gecommuniceerd via berichten. In het onderdeel architectuur is er kort gesproken over Tunnel en de functionaliteiten die ermee gerealiseerd wordt. Er zijn twee soorten berichten die verzonden worden: Tunnel berichten en I2P Network Protocol (I2NP) berichten². Het proces, zoals beschreven in Monero (2017c):

- De Tunnel verzamelt I2NP berichten en verwerkt ze naar Tunnel berichten. Hierbij kan het voorkomen dat I2NP berichten gefragmenteerd worden omdat ze van variabele grootte zijn, terwijl Tunnel berichten een vaste grootte hebben.

²Zie "I2NP Specification - I2P | Overview" voor de verschillende types.

- De Tunnel encrypt de verwerkte data en stuurt het door in de vorm van Tunnel berichten.
- De deelnemer, en andere deelnemers die deel uitmaken van de Tunnel, pakken een laag van de encryptie uit en verifiëren dat het bericht geen duplicaat is en sturen het vervolgens door naar een volgende deelnemer.
- Met de tijd zullen de Tunnel berichten het eindpunt bereiken waarna ze terug worden gezet naar de originele I2NP berichten.

4.7.2 Gevaren

Double Spending door gebruik te maken van ring signatures wordt de herkomst van een transactie gemaskeerd door de handtekening van de verstuurder te groeperen met handtekeningen vanuit outputs die reeds gedaan zijn in de Blockchain. Een probleem dat hierbij optreedt is de mogelijkheid tot de uitvoering van double spending omdat een transactie lastiger is te valideren. Hierdoor maakt Monero gebruik van key Image. Een key Image wordt gebruikt om te valideren dat de private key die gebruikt is om de transactie te ondertekenen niet eerder gebruikt is, zonder te onthullen welke handtekening het is.

4.7.3 Identiteit

De Monero implementatie is een public Blockchain waarbij alle transacties inzichtbaar zijn en elke deelnemer deel kan uitmaken van het consensus proces. De focus van de implementatie ligt op het verhogen van de privacy van een gebruiker.

Een account (i.e. wallet) binnen Monero is gebaseerd op twee keys, spend key en een view key. De spend key is een speciale key die benodigd is om Monero effecten uit te geven, terwijl daarentegen de view key gebruikt kan worden om een derde partij inzicht te geven in de gedane transacties, bijvoorbeeld voor verificatie doeleinden (Monero, 2017d, "Account"). Bovenstaande keys zijn ook terug te vinden in de bestemmingsadres van output binnen een transactie, waarbij het bestaat uit een eenmalige public key die berekend wordt vanuit de view key en spend key (Monero, 2017d, "Transaction"). Door het gebruik van een eenmalige public key garandeert het Monero protocol unlinkability.

Om aan de untracability eis te voldoen maakt Monero gebruik van ring signatures. Ring signature groepeerd de handtekening (i.e. de eenmalige public key afgeleid uit de view key en spend key ook wel bekend binnen Monero als het stealth address) van een deelnemer binnen een transactie met handtekeningen vanuit eerdere gedane outputs van transacties (Monero, 2017d, "Ring Signature").

Wanneer Bob Monero wilt versturen naar Alice, met een ring size van vijf handtekeningen, wordt een van de inputs uit Bob zijn account gehaald, welke toegevoegd wordt aan de transactie. De andere vier inputs worden uit de historie van de Blockchain gehaald. Deze vier inputs maskeren de herkomst van de transactie.

5 | Conclusie

In dit onderzoek is er gezocht naar een antwoord op de vraag: "Welke protocol implementaties kunnen toegepast worden om de onderdelen Distributed Network en Identity Management te realiseren voor een Blockchain implementatie?". Hiervoor is kwalitatief onderzoek uitgevoerd naar de Blockchain implementaties, EOS, Cardano, Monero en Bitcoin.

5.1 Deelvragen

1. **"Welke soorten gedistribueerde netwerken worden er gebruikt in de implementaties?"**

Een gedistribueerd netwerk binnen Blockchain is getypeerd aan het consensus protocol dat gebruikt wordt. In het onderzoek zijn er twee soorten geïdentificeerd, netwerken die gebruik maken van Proof of Stake of van Proof of Work.

2. **“Hoe werken de gedistribueerde netwerken van de implementaties en tegen welke gevaren zijn ze bestendig?”**

In het onderzoek is de functionaliteit beschreven die ondersteund wordt door een gedistribueerde netwerk van een implementatie en is er aandacht besteed aan de oplossingen die het netwerk gebruikt om aanvallen tegen te gaan.

- **Bitcoin**

Het netwerk van Bitcoin communiceert via TCP/IP en maakt gebruik van bootstrap nodes waarmee connectie wordt gemaakt op het moment dat een nieuwe deelnemer het netwerk wilt toetreden. Informatie wordt verstuurd door een voorafgedefinieerde set aan berichttypes: *inv*, *tx*, *block*, *getdata*, waarbij een *inv* bericht gebruikt wordt ter inventarisatie over de beschikbaarheid van data, *tx* bericht om een transactie te versturen, *block* bericht om een block te versturen, *getdata* bericht om data op te vragen.

Op het Bitcoin netwerk zijn meerdere aanvallen in de loop der jaren uitgevoerd en geïdentificeerd, een studie uit 2015 gedaan door Heilman et al. (2015) toont aan dat het Peer Discovery mechanisme vatbaar is voor een Sybil Attack. Nakamoto (2008) stelt dat de voordelen van het uitvoeren van een majority attack niet opweegt tegen de kosten voor de benodigde hardware om de rekenkracht te behalen. Eyal en Sirer (2014) beschrijft dat het niet nodig is om een merendeel van de rekenkracht te bezitten en introduceert de aanval selfish mining.

- **Cardano**

Het netwerk van Cardano communiceert via TCP/IP en maakt gebruik van het Kademlia protocol waardoor het maar nodig is om één bootstrap node te gebruiken om het netwerk toe te treden. De achterliggende structuur van Kademlia is een Binary Tree waarbij de positie van een deelnemer in de Binary Tree bepaald wordt door een unieke prefix van de identificatiecode. Het protocol garandeert dat een deelnemer in verbinding staat met ten minste één andere deelnemer. Informatie wordt uitgewisseld door drie abstracte berichttypes: *inv*, *req*, en *data*. Het *inv* bericht wordt gebruikt om aan te geven dat er data beschikbaar is, het *req* bericht wordt gebruikt om beschikbare data op te vragen en het *data* bericht wordt vervolgens gebruikt om de data te versturen.

Implementaties die gebruik maken van PoS zijn afhankelijk van de manier waarop een leiderschapsverkiezing wordt gesimuleerd, waarbij er grote kans is dat het gevoelig is voor beïnvloedingen van kwaadwillende deelnemers in het netwerk in de vorm van een Sybil Attack. Cardano heeft een zwak punt in het Kademlia netwerk geïdentificeerd waardoor het mogelijk zou zijn om Eclipse Attack uit te voeren.

- **Monero**

Het netwerk van Monero maakt gebruik van het The Invisible Internet Project (I2P) protocol, dat zowel UDP/IP als TCP/IP ondersteund. Om het netwerk toe te treden wordt er gebruik gemaakt van bootstrap nodes die vastgelegd zijn in de broncode. Communicatie wordt gedaan door middel van Tunnels, waarbij elke deelnemer twee Tunnels, een inkomende en een uitgaande, heeft voor elke connectie.

- **EOS**

Ten tijde van het onderzoek is er geen technische beschrijving beschikbaar over het netwerk component van EOS.

3. **“Hoe wordt er omgegaan met de identiteit van de gebruiker binnen de implementatie?”**

- **Bitcoin**

Bitcoin is een public Blockchain waarbij de gehele historie van transacties publiekelijk in te zien is. Een deelnemer in het Bitcoin netwerk wordt geïdentificeerd aan de hand van zijn public key. Deze public key wordt onder andere opgenomen in transacties om de betaler en de ontvanger te registreren. In een studie gedaan door Reid en Harrigan (2013) wordt er een analyse model opgezet dat aantoonde dat het Bitcoin protocol niet aan de untraceability eis voldoet.

- **Cardano**

Cardano is een public Blockchain waarbij de gehele historie van transacties publiekelijk in te zien is. Cardano maakt gebruik van public- en private key cryptografie om pseudonimiteit te waarborgen. Deze keys worden gebruikt om een transactie van een bestemming te voorzien, waarbij er drie definities van adressen gebruikt worden: een public key address, een script address en een redeem address.

- **EOS**

EOS is een consortium Blockchain waarbij gebruikers zichzelf identificeren met een unieke naam van maximaal twaalf karakters. Om te participeren binnen het netwerk dient er toegang verleent te worden door een authenticatie proces alvorens de deelnemer wordt toegelaten. Handeling binnen het netwerk worden gevalideerd door een Role Based Permissie systeem, waarbij permissies gekoppeld zijn aan actions die vastgelegd zijn in de lokale database.

- **Monero**

Monero is een public Blockchain waarbij de gehele historie van transacties publiekelijk in te zien is. Binnen Monero heeft elke deelnemer een account die gebaseerd is op twee keys: spend key en een view key. Door het afleiden van een eenmalige public key, ook wel een stealth address genoemd, uit de spend key en view key garandeert het Monero protocol unlinkability. Untraceability wordt behaald door het gebruik van ring signatures. Hierbij worden meerdere Stealth Addresses toegevoegd aan een transactie, waarbij een afkomstig van de verstuurder van de transactie en de rest aangevuld door eerder gebruikte Stealth Addresses in de Blockchain. Hierdoor wordt de herkomst van een transactie gemaskeerd.

5.2 Hoofdvraag

Tezamen beantwoorden de deelvragen de hoofdvraag:

Welke protocol implementaties kunnen toegepast worden om de onderdelen Distributed Network en Identity Management te realiseren voor een Blockchain implementatie?

5.2.1 Distributed Network

Zowel het gedistribueerd netwerk van Monero, Cardano (Kademlia) en Bitcoin maken gebruik van bootstrap nodes om een deelnemer toe te laten treden, waarbij het Kademlia protocol kan functioneren met één bootstrap node.

Protocol	Toelichting
Kademlia	Een bestaand protocol gerealiseerd door Maymounkov en Mazieres (2002). Dit protocol heeft een aantal wijzigingen binnen Cardano, zoals het versturen van informatie gaat over TCP/IP en er is een uitbreiding gemaakt op de manier waarop identificatiecodes toegekend worden aan deelnemers om een mogelijke sybil attack uit te sluiten.
Bitcoin	Communicatie verloopt over TCP/IP waarbij informatie wordt verstuurd door <i>inv</i> , <i>tx</i> , <i>block</i> en <i>getdata</i> berichten.
Monero	Focus op de privacy van de gebruiker en maakt gebruik van The Invisible Internet Project (I2P) om deze anonimiteit binnen het netwerk te waarborgen.

5.2.2 Identity Management

Protocol	Toelichting
Bitcoin	Maakt gebruik van het UTXO-model, waarin public- en private keys gebruikt worden om de betaler en ontvanger te registreren binnen een transactie. Door het gebruik van het analysemodel gepresenteerd door Reid en Harrigan (2013) is aangetoond dat het Bitcoin niet aan de niet aan de untraceability en unlinkability eis voldoet.
Cardano	Maakt gebruik van het UTXO-model, waarin public- en key cryptografie gebruikt wordt. Er is hierbij geen studie gevonden die aantoont dat het voldoet aan de untraceability en unlinkability eis, maar heeft aanzienlijke overeenkomsten met hoe Bitcoin omgaat met de identiteit.
EOS	Maakt gebruik van het Account-model, waarin een gebruiker een unieke naam van maximaal twaalf karakters hanteert als identiteit. Daarnaast hanteert EOS de volgende componenten: <ol style="list-style-type: none">1. Role Based Permission Management2. Actions & Handlers

Literatuur

- Roman, K. . (2018). *Understanding eos and delegated proof of stake — steemit*. Verkregen van <https://steemit.com/eos/@eosgo/understanding-eos-and-delegated-proof-of-stake>
- Steemit . (2017). *Eos vs. ethereum for dummies!* Verkregen van <https://steemit.com/eos/@trogdor/eos-vs-ethereum-for-dummies>
- Bitcoin Wiki. (2010). *Proof of work*. Verkregen van https://en.bitcoin.it/wiki/Proof_of_work ([Online; benaderd op 29 maart, 2018])
- Cardano Docs. (2013a). *Cardano*. Verkregen van <https://cardanodocs.com/technical/protocols/p2p/#addressing>
- Cardano Docs. (2013b). *Csl application-level messaging - cardano*. Verkregen van <https://cardanodocs.com/technical/protocols/csl-application-level/>
- Cardano Docs. (2013c). *Ouroboros proof of stake algorithm - cardano*. Verkregen van <https://cardanodocs.com/cardano/proof-of-stake/>
- Conti, M., Lal, C., Ruj, S. et al. (2017). A survey on security and privacy issues of bitcoin. *arXiv preprint arXiv:1706.00916*.
- Decker, C. & Wattenhofer, R. (2013, Sept). Information propagation in the bitcoin network. In *Ieee p2p 2013 proceedings* (p. 1-10). doi: 10.1109/P2P.2013.6688704
- en B. Scheuermann, F. T. (2016, thirdquarter). Bitcoin and beyond: A technical survey on decentralized digital currencies. *IEEE Communications Surveys Tutorials*, 18(3), 2084-2123. doi: 10.1109/COMST.2016.2535718
- Eyal, I. & Sirer, E. G. (2014). Majority is not enough: Bitcoin mining is vulnerable. In *International conference on financial cryptography and data security* (pp. 436-454).
- Heilman, E., Kendler, A., Zohar, A. & Goldberg, S. (2015). Eclipse attacks on bitcoin's peer-to-peer network. In *Usenix security symposium* (pp. 129-144).
- Karame, G., Androulaki, E. & Capkun, S. (2012). Two bitcoins at the price of one? double-spending attacks on fast payments in bitcoin. *IACR Cryptology ePrint Archive*, 2012(248).
- Karame, G. O., Androulaki, E. & Capkun, S. (2012). Double-spending fast payments in bitcoin. In *Proceedings of the 2012 acm conference on computer and communications security* (pp.

906–917).

- Kiayias, A. et al. (2017). Ouroboros: A provably secure proof-of-stake blockchain protocol. In *Annual international cryptology conference* (pp. 357–388).
- Kiffer, L., Levin, D. & Mislove, A. (2017). Stick a fork in it: Analyzing the ethereum network partition. In *Proceedings of the 16th acm workshop on hot topics in networks* (pp. 94–100).
- Lamport, L. et al. (1982). The byzantine generals problem. *ACM Transactions on Programming Languages and Systems (TOPLAS)*, 4(3), 382–401.
- Maymounkov, P. & Mazieres, D. (2002). Kademlia: A peer-to-peer information system based on the xor metric. In *International workshop on peer-to-peer systems* (pp. 53–65).
- Meiklejohn, S., Pomarole, M., Jordan, G., Levchenko, K., McCoy, D., Voelker, G. M. & Savage, S. (2013). A fistful of bitcoins: characterizing payments among men with no names. In *Proceedings of the 2013 conference on internet measurement conference* (pp. 127–140).
- Monero. (2017a). *Account* | *moneropedia* | *monero - secure, private, untraceable*. Verkregen van <https://getmonero.org/resources/moneropedia/account.html>
- Monero. (2017b). *Kovri* | *moneropedia* | *monero - secure, private, untraceable*. Verkregen van <https://getmonero.org/resources/moneropedia/kovri.html>
- Monero. (2017c). *Message* | *moneropedia* | *monero - secure, private, untraceable*. Verkregen van <https://getmonero.org/resources/moneropedia/message.html>
- Monero. (2017d). *Monoreopedia*. Verkregen van <https://getmonero.org/resources/moneropedia/>
- Monero. (2017e). *Tunnel* | *moneropedia* | *monero - secure, private, untraceable*. Verkregen van <https://getmonero.org/resources/moneropedia/tunnel.html>
- Nakamoto, S. (2008). Bitcoin: A peer-to-peer electronic cash system.
- NIST. (2013). *Nvd - cve-2013-5700*. Verkregen van <https://nvd.nist.gov/vuln/detail/CVE-2013-5700> ([Online; benaderd op 6 april, 2018])
- Noether, Y. et al. (2014). Monero is not that mysterious. *Technical report*.
- O'Dwyer, K. J. & Malone, D. (2014). Bitcoin mining and its energy footprint..
- Okamoto, K., Tatsuaki en Ohta. (1992). Universal electronic cash. In *Proceedings of the 11th annual international cryptology conference on advances in cryptology* (pp. 324–337).

London, UK, UK: Springer-Verlag. Verkregen van <http://dl.acm.org/citation.cfm?id=646756.705374>

Reid, F. & Harrigan, M. (2013). An analysis of anonymity in the bitcoin system. In *Security and privacy in social networks* (pp. 197–223). Springer.

Van Saberhagen, N. (2013). *Cryptonote v 2.0*.

Zantout, B. & Haraty, R. (2011). I2p data communication system. In *Proceedings of icn* (pp. 401–409).

V Architectuurdocument

Blockchain: Identity Management en Distributed Network

Architectuurdocument

Jeffrey van Hoven
30 mei 2018

Inhoudsopgave

1 Inleiding	2
2 Systeem stakeholders en requirements	3
2.1 Stakeholders	3
2.2 Requirements	3
2.2.1 Business rules	3
2.2.2 Functional requirements	4
2.2.3 Non-functional requirements	4
3 Architectuur views	5
3.1 Logical view	5
3.2 Development view	7
3.3 Physical view	8
3.3.1 Software Dependencies	8
3.4 Process view	9
3.5 Scenarios	10

1 | Inleiding

Dit document is opgesteld ter behoeve van het ontworpen architectuur voor de onderdelen Identity Management en Peer-to-Peer netwerk. Het maakt gebruik van het 4+1 architectural view model om logischerwijs de verschillende geïnteresseerden te informeren over de keuzes die gemaakt zijn.

2 | Systeem stakeholders en requirements

2.1 Stakeholders

Er zijn meerdere stakeholders die baat hebben bij de realisatie van dit project:

Quintor De opdrachtgever en tevens de eigenaar van het project. De organisatie heeft baat bij het opdoen van kennis gedaan door dit project. Tevens zal het de eindgebruiker zijn van het systeem.

Kevin Bos Heeft belang bij de realisatie van het onderdeel Distributed Network en Identity Management gezien de het gedeelte dat gerealiseerd wordt hem samen dient te werken met de componenten die voorgesteld zijn binnen dit document.

2.2 Requirements

2.2.1 Business rules

BR01	Berichten dienen van type req(uest), inv(entory), data en auth(entication) te zijn.
BR02	Transactietypes zijn: account - om een account te registreren in het netwerk, data - arbitraire data dat nog niet gedefinieerd is.

2.2.2 Functional requirements

Id	Beschrijving	Prioritering
FR01	Als gebruiker wil ik een transactie kunnen aanmaken.	Must have
FR02	Als gebruiker wil ik mijn data kunnen synchroniseren.	Should have
FR03	Als gebruiker wil ik connectie kunnen leggen met een deelnemer uit het Peer-to-Peer netwerk.	Must have
FR04	Als gebruiker wil ik mijn openstaande connecties kunnen inzien.	Could have
FR05	Als gebruiker wil ik kunnen toetreden in het Peer-to-Peer netwerk.	Must have
FR06	Als gebruiker wil ik een block kunnen aanmaken.	Must have
FR07	Als beheerder wil ik een gebruiker kunnen aanmaken.	Must have

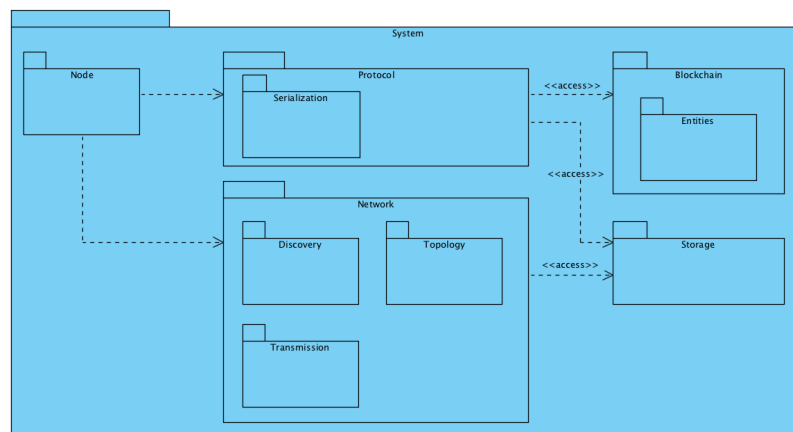
2.2.3 Non-functional requirements

Id	Beschrijving	ISO
NFR01	Het systeem dient om te kunnen gaan met deelnemers die de performance van het Peer-to-Peer netwerk proberen te verstoren.	Securability
NFR02	Het systeem dient om te kunnen gaan met het vervalsen van transacties.	Securability
NFR03	Het systeem dient makkelijk uitgebreid te worden door de kerncomponenten modulair op te stellen.	Maintainability
NFR04	Het systeem dient rekening te houden met protocol updates, en dient interactie met verouderde versies niet te ondersteunen.	Maintainability, Securability
NFR05	Het systeem dient makkelijk ingezet te kunnen worden.	Deployment

3 | Architectuur views

3.1 Logical view

In de logische weergave wordt de architectuur benaderd vanuit het oogpunt van de eindgebruiker. Hierin komen de functionaliteiten van de verschillende componenten aan bod om de functionaliteit te ondersteunen.

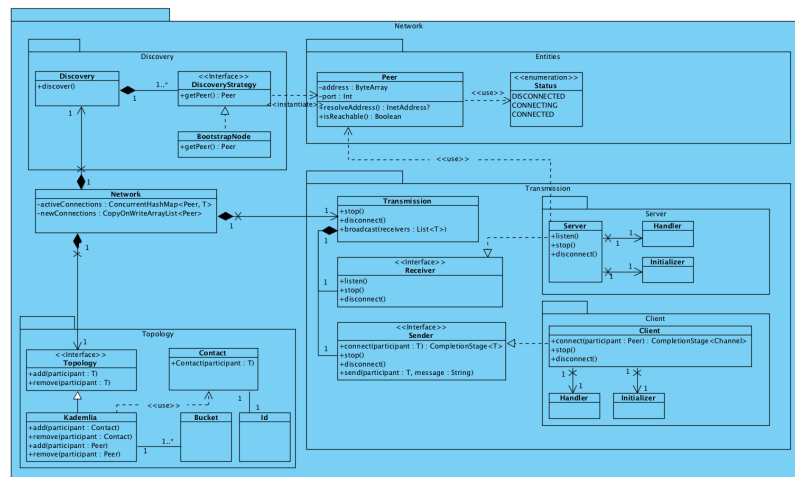


Figuur 3.1: Overzicht van het systeem

In fig. 3.1 is een overzicht te zien van de verschillende onderdelen van het systeem. De node is het startpunt van het systeem en maakt gebruik van een protocol specificatie om entiteiten uit de Blockchain op te maken in berichten die geschikt zijn voor het geïmplementeerde protocol.

Daarnaast maakt het gebruik van de netwerk specificatie om het toe te treden, de topologie te creëren en berichten die gemaakt zijn door het protocol te versturen.

Op de volgende pagina's zijn de verschillende componenten in detail gemodelleerd.



Figuur 3.2: Gedetailleerd overzicht van het Network component

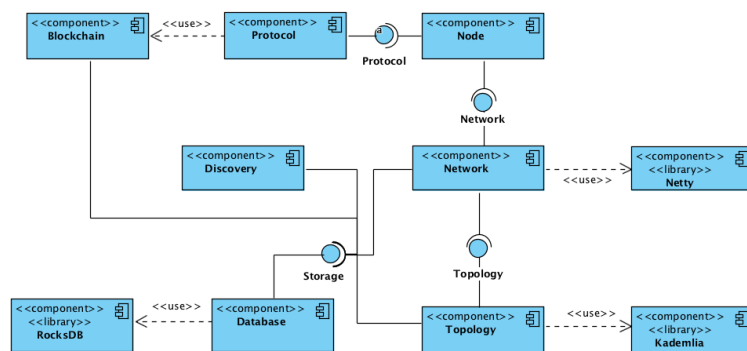
In fig. 3.2 is een gedetailleerd overzicht te van van het Network component. Het Discovery component is verantwoordelijk voor het uitvoeren van het Peer Discovery Protocol dat gebruikt wordt ten tijde van het toetreden van het netwerk. Er wordt hierbij gebruik gemaakt van een strategie, namelijk het opzoeken van een BootstrapNode.

Het Topology component is verantwoordelijk voor de structuur van het netwerk. Dit is modulair opgebouwd zodat het makkelijk gewisseld kan worden door een andere implementatie. De default topology is het Kademlia protocol.

Het Transmission component is verantwoordelijk voor het versturen en ontvangen van berichten. Dit is opgesplitst in een Receiver en Sender interface zodat het niet protocol specifiek geïmplementeerd hoeft te zijn.

3.2 Development view

De development weergave illustreert het systeem van een programmeur perspectief en omvat het Software Management gedeelte.



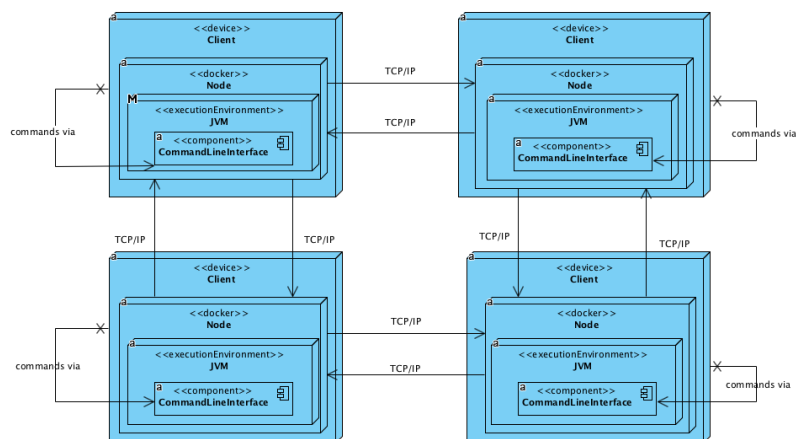
Figuur 3.3: Component Diagram waarin de diverse componenten en de samenwerking daartussen te zien is.

In fig. 3.3 is het component diagram te zien waarin de kerncomponenten van de applicatie staan. Hieronder zijn alle component individueel besproken:

- **Blockchain**
Het Blockchain component bevat de logica en cryptografie om de structuur van een Blockchain op te bouwen. Een belangrijk onderdeel van het Blockchain component zijn de identiteiten die benodigd zijn voor de communicatie tussen verschillende participanten van het netwerk.
- **Protocol**
Het Protocol component stelt de regels op met betrekking tot het gebruik van de Blockchain data.
- **Node**
Het Node component bevat de functionaliteit waarmee de eindgebruiker kan interacteren.
- **Network**
Het Network component is een encapsulatie van de verschillende componenten die hier deel van uitmaken. Het is verantwoordelijk voor het opzetten van het gehele Peer-to-Peer netwerk.

- **Discovery**
Het Discovery component bevat de Peer Discovery mechanisme die gebruikt om toe te treden in het netwerk.
- **Topology**
Het Topology component bepaald de infrastructuur van het Peer-to-Peer netwerk.
- **Database**
Het Database component bevat de logica om te interacteren met de gekozen database implementatie.

3.3 Physical view



Figuur 3.4: Deployment Diagram

In het Deployment Diagram is te zien dat er gebruik gemaakt wordt van Docker om de Blockchain client te draaien. Een vereiste hiervan is dat de Docker container beschikking heeft over de Java Virtual Machine. Communicatie tussen Blockchain clients gebeurt over TCP/IP waardoor een internetconnectie een vereiste is.

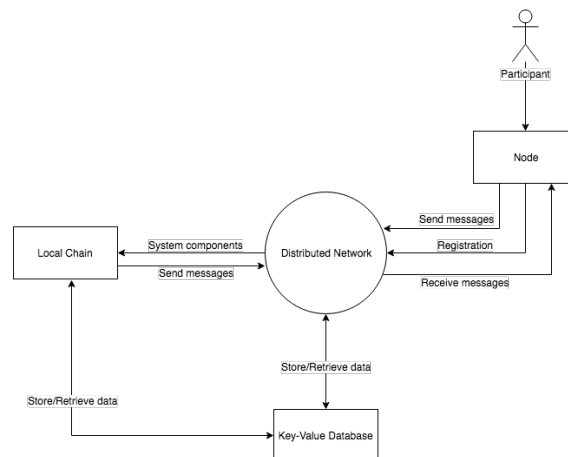
3.3.1 Software Dependencies

Om de applicatie te laten werken in de Docker container zijn er een aantal software modules nodig:

Module	Beschrijving
Maven	Wordt gebruikt om alle dependencies op te halen, en tevens het build proces uit te voeren.
RocksDB	Verzorgt de opslag binnen de applicatie.

3.4 Process view

De contextweergave van het systeem beschrijft de relaties, afhankelijkheden en interacties tussen het systeem en zijn omgeving (de mensen, systemen, en externe identiteiten waarmee het communiceert).

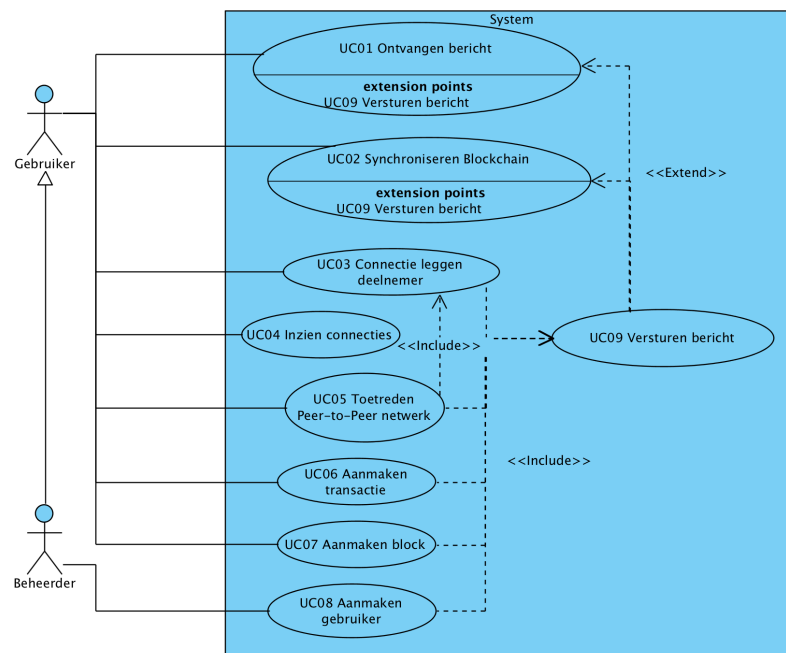


Figuur 3.5: Context Diagram waarin de interacties te zien is tussen het systeem en zijn omgeving.

De gebruiker draait een Node die gebruik maakt van het Peer-to-Peer netwerk om berichten te versturen. Een van de berichten is specifiek weergegeven aangezien het gaat om de registratie van een nieuwe gebruiker in het systeem. Het Distributed Network maakt gebruik van entiteiten uit de Local Chain om de benodigde data te versturen.

Zowel het Local Chain gedeelte als het Distributed Network maken gebruik van een Key-Value database om data op te slaan. In het geval van het Distributed Network gaat dit om informatie over connecties.

3.5 Scenarios



Figuur 3.6: Use-case diagram waarin de rollen binnen het systeem te zien zijn en de acties die zij kunnen uitvoeren.

Tabel 3.1: Use-case: Ontvangen bericht

Use-case	Ontvangen bericht
<i>Id</i>	UC01
<i>Requirements</i>	FR03, FR02, FR01
<i>Beschrijving</i>	Gebruiker ontvangt een bericht van een deelnemer uit het Peer-to-Peer netwerk
<i>Primaire actor</i>	Gebruiker
<i>Secundaire actor</i>	-
<i>Precondition</i>	De gebruiker is verbonden met het Peer-to-Peer netwerk
<i>Main flow</i>	<ol style="list-style-type: none"> 1. Systeem ontvangt bericht 2. Systeem valideert bericht type 3. Systeem deserialiseert bericht 4. Systeem controleert of er antwoord verstuurd dient te worden 5. Use-case eindigt (Postconditie: Success1)
<i>Postconditie</i>	Success1: Systeem heeft een bericht verstuurd naar verzender Failure1: Systeem is ongewijzigd
<i>Alternatieve flows</i>	<ol style="list-style-type: none"> 1. Bericht is van type <i>req</i> (na MF4) <ol style="list-style-type: none"> 1.1. Systeem valideert dat gevraagde data aanwezig is 1.2. Systeem creëert <i>data</i> bericht 1.3. Systeem voert <i>UC09 - Versturen bericht</i> uit 1.4. Use-case eindigt (Postconditie: Success1) 2. Bericht is van type <i>inv</i> (na MF4) <ol style="list-style-type: none"> 2.1. Systeem valideert dat aangeboden data niet aanwezig is 2.2. Systeem creëert <i>req</i> bericht 2.3. Systeem voert <i>UC09 - Versturen bericht</i> uit 2.4. Use-case eindigt (Postconditie: Success1)

Tabel 3.2: Use-case: Synchroniseren Blockchain

Use-case	Synchroniseren Blockchain
<i>Id</i>	UCo2
<i>Requirements</i>	FRo2
<i>Beschrijving</i>	Gebruiker haalt Blockchain informatie op van een verbonden deelnemer
<i>Primaire actor</i>	Gebruiker
<i>Secundaire actor</i>	-
<i>Precondition</i>	De gebruiker is verbonden met het Peer-to-Peer netwerk
<i>Main flow</i>	<ol style="list-style-type: none"> 1. Systeem maakt <i>req</i> bericht 2. Systeem voert <i>UCo9 - Versturen bericht</i> uit 3. Systeem voert <i>UCo1 - Ontvangen bericht</i> uit 4. Systeem hercreëert Blockchain van ontvangen data 5. Use case eindigt (Postconditie: Success1)
<i>Postconditie</i>	Success1: Gebruiker is up-to-date met de laatste Blockchain data

Tabel 3.3: Use-case: Connectie leggen deelnemer

Use-case	Connectie leggen deelnemer
<i>Id</i>	UCo3
<i>Requirements</i>	FRo3
<i>Beschrijving</i>	Gebruiker maakt connectie met een deelnemer uit het Peer-to-Peer netwerk
<i>Primaire actor</i>	Gebruiker
<i>Secundaire actor</i>	-
<i>Precondition</i>	De gebruiker is verbonden met het Peer-to-Peer netwerk
<i>Main flow</i>	<ol style="list-style-type: none"> 1. Systeem vraagt om adresgegevens(ip, poort) van deelnemer 2. Actor vult informatie in 3. Systeem valideert adresgegevens 4. Systeem valideert dat deelnemer bereikbaar is 5. Systeem creëert <i>auth</i> bericht 6. Systeem voert <i>UCo9 - Versturen bericht</i> uit 7. Use-case eindigt (Postconditie: Success1)
<i>Postconditie</i>	Success1: De gebruiker is verbonden met de deelnemer Failure1: Systeem is ongewijzigd
<i>Alternatieve flow</i>	<ol style="list-style-type: none"> 1. Invalide adresgegevens (na MF3) <ol style="list-style-type: none"> 1.1. Use-case gaat verder bij MF1 2. Deelnemer is niet bereikbaar (na MF4) <ol style="list-style-type: none"> 2.1. Systeem toont foutmelding 2.2. Use-case eindigt (Postconditie: Failure1) 3. Actor annuleert (Overall)

Tabel 3.4: Use-case: Toetreden Peer-to-Peer netwerk

Use-case	Toetreden Peer-to-Peer netwerk
<i>Id</i>	UCo5
<i>Requirements</i>	FRo5
<i>Beschrijving</i>	Gebruiker wilt deel uitmaken van het Peer-to-Peer netwerk
<i>Primaire actor</i>	Gebruiker
<i>Secundaire actor</i>	-
<i>Precondition</i>	Actor heeft een account tot zijn beschikking
<i>Main flow</i>	<ol style="list-style-type: none"> 1. Actor start systeem 2. Systeem controleert of de actor niet reeds connectie heeft gemaakt 3. Systeem zoekt bootstrap node op 4. Systeem verstuurd authenticatie bericht naar bootstrap node 5. Systeem voert <i>UCo1 - Ontvangen bericht</i> uit 6. Systeem ontvangt lijst van andere deelnemers die verbinding gemaakt hebben met het netwerk 7. Systeem voert <i>UCo3 - Connectie leggen deelnemer</i> uit 8. Systeem voert <i>UCo1 - Ontvangen bericht</i> uit 9. Systeem slaat adresgegevens (ip, port) op van deelnemer 10. Systeem voert <i>UCo2 - Synchroniseren Blockchain</i> uit 11. Use-case eindigt (Postconditie: Success1)
<i>Post conditie</i>	Success1: Actor is actief in het netwerk. Failure1: Systeem is ongewijzigd
<i>Alternatieve flows</i>	<ol style="list-style-type: none"> 1. AF1: Gebruiker heeft reeds connectie gemaakt (na MF2) <ol style="list-style-type: none"> 1.1. Systeem haalt lijst van opgeslagen deelnemers op 1.2. Systeem probeert verbinding te maken met deelnemers 1.3. Systeem voert <i>UCo1 - Ontvangen bericht</i> uit 1.4. Use-case eindigt (Postconditie: Success1) 2. AF2: Actor gebruikt verkeerde identificatie (na MF5) <ol style="list-style-type: none"> 2.1. Systeem toont foutmelding 2.2. Use-case eindigt (Postconditie: Failure1)

Tabel 3.5: Use-case: Aanmaken transactie

Use-case	Aanmaken transactie
<i>Id</i>	UCo6
<i>Requirements</i>	FRo1
<i>Beschrijving</i>	Gebruiker wilt een transactie opslaan in de Blockchain
<i>Primaire actor</i>	Gebruiker
<i>Secundaire actor</i>	-
<i>Precondition</i>	De gebruiker is verbonden met het Peer-to-Peer netwerk
<i>Main flow</i>	<ol style="list-style-type: none"> 1. Systeem vraagt om public key ontvanger 2. Actor vult public key in 3. Systeem valideert public key 4. Systeem vraagt om type transactie 5. Actor selecteert type transactie 6. Systeem vraag aanvullende informatie gebaseerd op geselecteerde type 7. Actor vult aanvullende informatie in 8. Systeem valideert aanvullende informatie 9. Systeem maakt transactie van geselecteerde transactietype aan 10. Systeem creëert een <i>inv</i> bericht 11. Systeem voert <i>UCog - Versturen bericht</i> uit 12. Use-case eindigt (Postconditie: Success1)
<i>Post conditie</i>	Success1: Systeem heeft een transactie aangemaakt
<i>Alternatieve flows</i>	<ol style="list-style-type: none"> 1. Actor annuleert (Overal)

Tabel 3.6: Use-case: Versturen bericht

Use-case	Versturen bericht
<i>Id</i>	UC09
<i>Requirements</i>	FR01
<i>Beschrijving</i>	Gebruiker verstuurd bericht over het netwerk
<i>Primaire actor</i>	Gebruiker
<i>Secundaire actor</i>	-
<i>Precondition</i>	De gebruiker is verbonden met het Peer-to-Peer netwerk
<i>Main flow</i>	<ol style="list-style-type: none"> 1. Systeem controleert bericht type 2. Systeem verstuurd bericht naar deelnemer 3. Use-case eindigt (Postconditie: Success1)
<i>Postconditie</i>	Success1: Systeem heeft bericht verstuurd naar deelnemer Success2: Systeem heeft bericht verstuurd naar alle verbonden deelnemers
<i>Alternatieve flows</i>	<ol style="list-style-type: none"> 1. Bericht is van type <i>inv</i> (na MF1) <ol style="list-style-type: none"> 1.1. Systeem haalt lijst van alle verbonden deelnemers op 1.2. Systeem verstuurd bericht naar alle verbonden deelnemers 1.3. Use-case eindigt (Postconditie: Success2)

VI Voortgangsverslag

Voortgangsverslag: Afstuderen Quintor Den Haag
28 maart 2018
Jeffrey van Hoven

In dit document wordt de voortgang besproken in het afstudeertraject van Jeffrey van Hoven bij het bedrijf Quintor in Den Haag. Het omvat werkzaamheden van 4 sprints waarin er gewerkt is aan het opstellen van het plan van aanpak, het doen van vooronderzoek, onderzoeksopzet en een start maken aan het uitvoeren van het onderzoek.

Plan van Aanpak

Het opstellen van het plan van aanpak duurde iets langer als ingepland. Uiteindelijk is er veel tijd besteed aan het beschrijven van de aanpak en het scherpstellen van de probleemstelling en doelstelling. Dit is in overleg gebeurd met de begeleider vanuit Quintor, Ben Ooms, en een mede afstudeerder, Kevin Bos, die het lokale onderdeel van de Blockchain onderzoekt. Hier zijn meerdere gesprekken over gehouden en daarom is het opstellen van het plan van aanpak ook een beetje uitgelopen.

Vooronderzoek

Gedurende de tijd die gebruikt werd om het plan van aanpak op te stellen is er ook vooronderzoek gedaan naar de onderdelen die deel uitmaken van mijn afstudeeropdracht, namelijk het Distributed Network en het Identity Management. Hierdoor is er tijds winst geboekt bij het doen van het vooronderzoek terwijl het plan van aanpak uitliep. De beschrijving van de werkzaamheden is hierbij wel achtergelopen voor het afstudeerverslag, wat weer ingehaald is in de afgelopen weken.

Onderzoeksopzet

Aan dit onderdeel is veel tijd besteed waardoor het uitgelopen is. Zo is er veel tijd verloren gegaan aan het opstellen van een selectiemethode voor de te onderzoeken implementaties. Daarnaast bleek het opstellen van de hoofdvraag en deelvragen redelijk lastig, aangezien de scope van het onderzoek niet is beperkt vanuit Quintor. Hier zijn ook meerdere gesprekken over geweest gedurende het project waaruit naar voren kwam dat de toepassing pas gegeven werd na het onderzoek.

Conclusie

In het algemeen loop ik achter op de initiële planning die ik gemaakt heb. Zoals besproken tijdens het bezoek van dhr. T. Cocx bij Quintor, is er ruim de tijd genomen om het adviesrapport op te stellen. Die tijd kan gelijktijdig gebruikt worden om het onderzoek uit te voeren. Over het algemeen ben ik tevreden met de voortgang die ik gemaakt heb, en ik hoop dat de obstakels die ik tegengekomen ben tijdens het opstellen van het plan van aanpak en het onderzoek duidelijk terug te lezen zijn in mijn afstudeerverslag.

VII Bezoekverslag

Bezoekverslag: Afstuderen Quintor Den Haag
27 maart 2018
Jeffrey van Hoven

Verslag

Op 19 maart 2018 is dhr. T. Cocx langsgeweest voor het benodigde bedrijfsbezoek bij Quintor Den Haag om kennis te maken met het bedrijf en meer inzicht te krijgen in de afstudeeropdracht die uitgevoerd wordt door de student. In dit document worden de belangrijkste afspraken, leerpunten en conclusies besproken.

Afspraken

Tijdens het bezoek is er kort verteld over de mogelijkheden tot het verkrijgen van feedback. Er werd nadruk gelegd op de tussentijdse beoordeling en dat het belangrijk is om deel te nemen aan het feedbackmoment dat beschikbaar is in de 10de week van de afstudeeropdracht. Hierbij is duidelijk verteld dat er verwacht wordt dat de student, indien hij gebruik wilt maken van het feedbackmoment, verwacht wordt om 60% van het verslag afgerond te hebben. Het is ook aan de student om de afspraak te maken indien hij er gebruik van wilt maken.

Leerpunten

Daarnaast is er gesproken over inhoudelijke werkzaamheden in relatie tot het verslag. Hierbij zijn een aantal punten genoemd over de formulering van het onderzoek. Er werd bijvoorbeeld gesproken over "technieken", wat een nogal vage term is en meerdere betekenissen kan hebben. Als suggestie werd er gegeven om het "protocol implementaties" te noemen.

Het tweede onderwerp was de stakeholder relatie met een andere afstudeerder die een onderdeel van de Blockchain gaat realiseren. Dit is totaal niet beschreven in het afstudeerverslag, maar toont wel de complexiteit van de opdracht. Er werd dan ook als tip gegeven om dit wel te beschrijven in het afstudeerverslag.

Conclusies

Uit het gesprek is waardevolle feedback gekomen. Aangezien er veel nadruk werd gelegd op het feedback moment in de 10de week, ook al is er aangegeven dat het niet benodigd is, zal er zeker naar toegewerkt worden om die datum als een deadline neer te zetten.

VIII Implementatie selectie

Tabel 1: Bekeken implementaties uit de initiële selectie met de onderzochte attributen.

Blockchain	Identity Management	Whitepaper	Open-source	In circulation since	Available Dapps development platform	Notes	Consensus	Website	Repository	Gebruikte talen	Whitepaper url
Bitcoin	No	Yes	Yes	4/27/11	No		Proof of Work	https://bitcoin.org/nl/	https://github.com/bitcoin/	C++	https://bitcoin.org/bitcoin.pdf
Ethereum	Yes	Yes	Yes	7/30/15	Yes		Proof of Work	https://www.ethereum.org/	https://github.com/ethereum/	Go, C++	https://github.com/ethereum/wiki/White-Paper
Tether	No	Yes	No	2015	Yes	Fork from Bitcoin.	Proof of Reserves	https://www.tether.to/	https://bitbucket.org/tether/		https://tether.torps.com/content/uploads/2016/06/tetherwhitepaper.pdf
Ripple	Not sure	Yes	Yes	2012	No	Not really a Blockchain	Ripple Consensus Algorithm	https://ripple.com/	https://github.com/ripple/	C++	https://ripple.com/files/ripple_consensus_whitepaper.pdf
EOS	Yes	Sort of	Yes	1/31/18	Yes		Delegated Proof of Stake	https://eos.io/	https://github.com/eosio/	C++	
Cardano	Not sure	Yes	Sort of	11/29/17	Yes	Only the Settlement layer is open-source available.	Proof of Stake		https://github.com/input-output-hk/cardano-sl	Haskell	
NEM	Yes	No	Yes	2014	Yes		Delegated Byzantine Fault Tolerance	https://nem.org/	https://github.com/nem-project/	Walter in C++	https://nem.org/uploads/files/ae772ef64dc8b1w396d48095b1.pdf
Quorum	Yes	Yes	Sort of	11/3/17	Yes	Adopts the UNO model from Bitcoin while utilizing the Ethereum network. Implements PoS 3.0 as sorted by the original Proof of Stake coin Bitcoin. Only the wallet is open-source.	Proof of Stake	https://quorum.org/en/	https://github.com/quorumproject/		
TRON	No	Yes	Sort of	2017	No	Badly translated whitepaper and website.	Proof of Stake	https://tron.network/en.html	https://github.com/tronprotocol/	Java	https://836f9e91.uns1.com/tron/whitebook/TronWhitepaper_en.pdf
Status	Yes	No	Yes		Yes	Identity Management in form of usernames. Mobile client voor Ethereum, een Dapp die het mogelijk maakt om te interacten met andere Dapps?	Proof of Work	https://status.im/	https://github.com/status-im/	Go	
Stellar	Not sure	Sort of	Yes	2014	Yes	Whitepaper only describes the consensus protocol, initially based on the Ripple protocol.	Stellar Consensus Protocol	https://www.stellar.org	https://github.com/stellar/	C	
Huobi Token	No	No	No		No	Loyalty Blockchain, users can buy but are awarded these tokens. Doesnt look like anything's available for this crypto.	? ?	https://www.huobi.pro			154
AMM Coin	Not sure	No	No		No			https://ammcoin.com/website/en/ico			https://ammcoin.com/content/docs/documents/ammcoin_whitepaper_en-us.pdf
Dash	No	Sort of	Yes	1/6/14	Not sure	Initially named Xeon (XC), renamed to Darkcoin and then rebranded as Dash. Fork from Litecoin. First self-funded blockchain. Transaction fees go to a treasury which funds development.	Proof of Service	https://www.dash.org/	https://github.com/dashpay/	C++	https://github.com/dashpay/dash/wiki/Whitepaper
VeChain	No	No	No	2015	No	Chinese, private blockchain for retail usage in combination with IoT.		https://www.vechain.com/#/			
LSK	Yes	No	Yes	9/22/17	Yes	Technical documentation available at https://lsk.io/documentation , albeit not in depth.	Delegated Proof of Stake	https://lsk.io/	https://github.com/LSK-HQ	JavaScript	https://lsk.io/documentation/
Monero	Yes	Yes	Yes	4/18/14	No	Claims to be one and only fully anonymized Blockchain implementation.	Proof of Work	https://getmonero.org/	https://github.com/monero-project/	C++	https://download.getmonero.org/whitepaper_a.moredated.pdf
Bitshare (BCN)	Yes	Yes	Yes			Uses black and white addresses.	PoW + PoS	https://bitshare.io/	https://github.com/bitshareIOg	C++	
Nano	No	Yes	Yes		Not sure	Previously known as Flabbits, second blockchain that uses a tangle instead of a chain.	Loop Fault Tolerance	https://nano.org/en	https://github.com/nanofoundation/		https://nanofoundation/resources/whitepaper/COINWhitepaper-EN-Draft.pdf