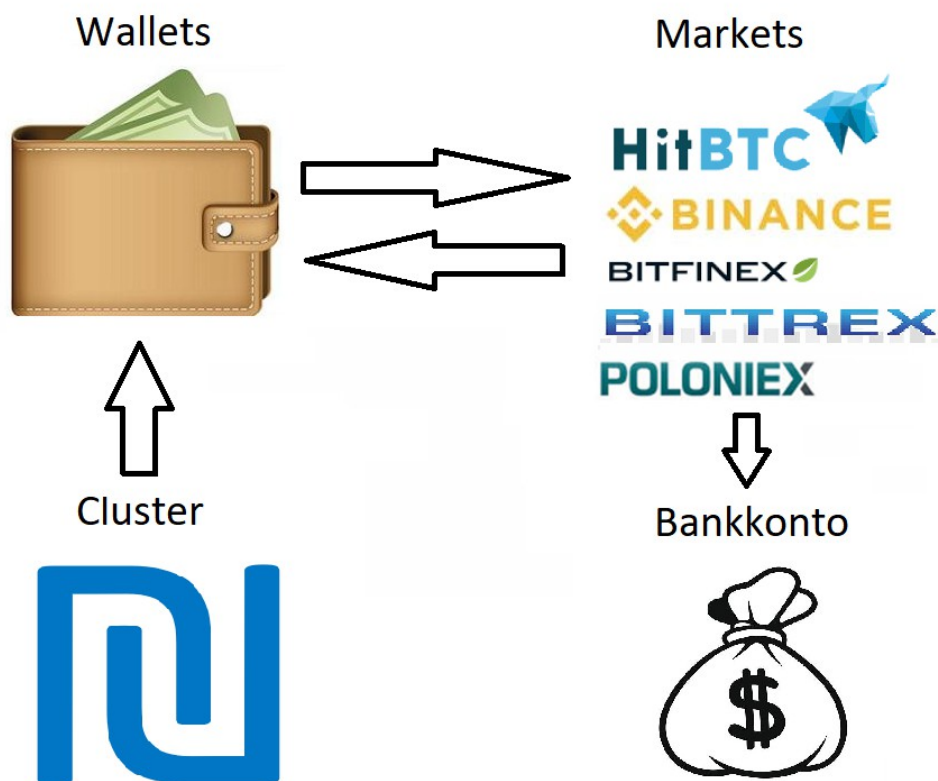


Nebula: Mining Cluster

Christoph Amrein



Management Summary

Inhaltsverzeichnis

1	Initialisierung	1
1.1	Ausgangslage	1
1.1.1	Weshalb soll das Projekt realisiert werden?	1
1.1.2	Für wen ist das Projekt gedacht?	1
1.2	Situationsanalyse	1
1.2.1	Stärken	2
1.2.2	Schwächen	2
1.3	Ziele	2
1.3.1	Vorgehensziele	2
1.3.2	Projektziele	3
1.3.3	Lieferobjekte	4
1.3.4	Rahmenbedingungen	4
1.3.5	Abgrenzungen	4
1.4	Lösungsbeschreibung	5
1.5	Kosten	5
1.5.1	Einmalige Kosten	5
1.5.2	Betriebskosten (repetitiv)	6
1.5.3	Gesamtkosten	6
1.6	Wirtschaftlichkeit	7
1.6.1	Spekulation	7
1.6.2	Infrastruktur	7
1.7	Planung	8
1.7.1	Grober Projektplan	8
1.7.2	Termine	9
1.8	Ressourcen	9
1.8.1	Budget	9
1.8.2	Sachmittel	9
1.9	Organisation	10
1.9.1	Projektorganisation	10
1.9.2	Projektablage	10
1.10	Cluster-Software Evaluation	11
1.10.1	Cluster Software Kriterien	11
1.10.2	Informationsbeschaffung	11
1.11	Lösungsvarianten	12
1.11.1	Variantenübersicht	12
1.11.2	Variante V1 «OpenHPC»	12

1.11.3	Variante V2 «TinyTitan»	12
1.11.4	Variante V3 «Minimale Lösung»	13
1.11.5	Anforderungsabdeckung der Varianten	13
1.12	Bewertung der Varianten	14
1.13	Variantenentscheid	14
1.14	Risiken	14
2	Konzept	16
2.1	Physikalischer Überblick	16
2.2	Physikalische Verbindungen	17
2.2.1	Stromversorgung Management Node	17
2.2.2	Compute Nodes	17
2.2.3	Übrige Geräte	17
2.2.4	Netzwerkverbindungen	17
2.3	Technischer Überblick	18
2.3.1	Verwendete Protokolle	19
2.4	Technische Verbindungen & Kommunikation	19
2.5	Komponentenbeschreibung	20
2.5.1	Router	20
2.5.2	PC	20
2.5.3	Management Node	20
2.5.4	Netzteil Management Node	20
2.5.5	NAS	20
2.5.6	Switch	20
2.5.7	Compute Nodes	21
2.5.8	Schaltnetzteil Compute Nodes	21
2.6	Tests	21
2.6.1	Testobjekte	21
2.6.2	Testarten	22
2.6.3	Testvoraussetzungen	22
2.6.4	Fehlerklassen	23
2.6.5	Testhilfsmittel	23
2.6.6	Testfazit	23
2.7	Monitoring	24
2.7.1	Service Monitoring - Nagios	24
2.7.2	Performance Monitoring - Ganglia	24
2.8	Mining	24
2.8.1	Kryptowährungen	25
2.9	Hostnamen	25
2.9.1	Management Node Name	25
2.9.2	Reserve Node Name	25

2.9.3	Compute Node Namen	26
3	Realisierung	27
3.1	Physischer Aufbau	27
3.1.1	Komponenten Platzierung	27
3.1.2	Kühlung	28
3.1.3	Stromversorgung	28
3.1.4	Kommunikation	28
3.2	Technischer Aufbau	28
3.2.1	Betriebssystem	28
3.2.2	Vorbereitungen	29
3.2.3	Installation	29
4	Schlussbetrachtung	29
4.1	Arbeiten nach dem Projekt	30
4.2	Persönliche Betrachtung	30
4.3	Danksagung	30
5	Authentizität	30
A	Anhang	31
B	Vorbereitungen RPI's	31
B.1	Betriebssystem installieren	31
B.2	RPI für den Netzwerkboot vorbereiten	34
B.3	Hostname und IP der MAC Adresse zuweisen	34
B.4	Netzwerkshare einrichten	36
C	Installation Management Node	37
C.1	Variablen Definition	37
C.2	Basiskonfiguration	41
C.3	OpenHPC Komponenten installieren	41
C.4	Netzwerkboot einrichten	42
C.5	Netzwerkshare einbinden	47
C.6	Registration Compute Nodes	47
C.7	Monitoring installieren	48
C.8	Miner installieren	49
D	Projektplan	54
E	Diplomeingabe	55
F	Testfälle	58
F.1	Komponententests	58

Inhaltsverzeichnis

F.2	Integrationstests	74
F.3	Systemtests	74
G	Testprotokoll	77
G.1	Komponententests	77
G.2	Integrationstests	101
G.3	Systemtests	105
H	Protokoll	110
I	Arbeitsjournal	112
J	Quellenverzeichnis	125
	Abbildungsverzeichnis	126
	Tabellenverzeichnis	127

1 Initialisierung

1.1 Ausgangslage

Die aktuelle Umgebung ist nicht auf das Schürfen von Kryptowährungen ausgelegt. Das System hat eine Uptime von maximal 40%. Auch der Standort, der Lärm und die Hitze des Systems und Raumes werden als störend empfunden. Deswegen wurden hauptsächlich Tokens auf Börsen gekauft, welche nicht geschürft werden können. Die Sicherung der Daten ist ebenfalls nicht gewährleistet. Zudem existieren keine Monitoring Tools, welche den Status des Schürfens und der Hardware zu erkennen geben. Dabei wird das Analysieren von Problemen als schwierig erachtet, da keine Logdaten existieren oder diese mit viel Aufwand zusammengesucht werden müssen.

1.1.1 Weshalb soll das Projekt realisiert werden?

Es soll eine stabile Lösung zum Schürfen von Kryptowährungen auf CPU Basis erschaffen werden, welche permanent in Betrieb sein kann und Profit generiert.

1.1.2 Für wen ist das Projekt gedacht?

Das Projekt wird in eigenem Interesse aufgebaut. Es existieren demnach keine Kunden und Abhängigkeiten zu anderen Personen oder Unternehmen.

1.2 Situationsanalyse

Für das Schürfen von Kryptowährungen werden folgende Komponenten eingesetzt:

Nr.	Typ	Komponente	Modell Version
1	HW	Prozessor (CPU)	Intel Core i7-4700, 3.40 GHz Quad Core
2	HW	Grafikkarte (GPU)	NVIDIA GeForce GTX 1070 Ti
3	HW	Festplatte (HDD)	TOSHIBA DT01ACA200
4	SW	Schürf-Software	Minergate, Version 7.2
5	SW	Betriebssystem (OS)	Windows 10 EDU, Version 1709

Tabelle 1: Situationsanalyse Komponenten

Legende: HW = Hardware, SW = Software

1 Initialisierung

1.2.1 Stärken

Nr.	Kategorie	Beschreibung
1	Bedienbarkeit	Das Schürfen der Währungen kann über ein GUI gestartet werden.
2	Wartung	Es existieren keine Umsysteme

Tabelle 2: Situationsanalyse Stärken

1.2.2 Schwächen

Nr.	Kategorie	Beschreibung
1	Flexibilität	Während des Schürfens ist der Computer für andere Tätigkeiten blockiert.
2	Kosten	Die Betriebskosten sind höher als der Ertrag.
3	Betriebszeit	Es können nicht durchgehend Kryptowährungen geschürft werden.

Tabelle 3: Situationsanalyse Schwächen

1.3 Ziele

1.3.1 Vorgehensziele

Zeitplan

Während der Initialisierungsphase wurde eine Projektplanung mit den Aufgaben und den vorgesehenen Aufwänden während des Projektes erstellt. Die definierten Soll-Aufwände sollen mit den stetig nachgeführten IST-Aufwänden verglichen werden. Die Abweichungen werden im Projektplan direkt errechnet.

Meilensteine

Die Meilensteine wurden in der Zeitplanung des Projektes berücksichtigt und definiert. Die Aufwände werden jeweils im Projektplan nachgeführt, dies ermöglicht einen Ist- und Soll-Aufwand-Vergleich.

Arbeitsjournal

Das Arbeitsjournal wird alle 2 Wochen an die Experten versendet. Diese haben die Möglichkeit, die Aufwände und investierte Zeit zu überprüfen.

Beweiserbringung

Alle geleisteten Arbeiten sollen in dokumentarischer Form, Präsentation oder einem Gespräch bewiesen werden können.

1.3.2 Projektziele

Nr.	Ziel	Messgrösse	Kat.	Prio.
01	Die CPU des Clusters soll zu 90% zum Schürfen von Kryptowährung beansprucht werden.	Log und Monitoring Auswertungen nach dem Testlauf.	LZ	M
02	Die Daten werden auf einem NAS mit RAID I gesichert.	Die Festplatten werden einzeln überprüft, der Datenbestand muss identisch sein.	BZ TZ	M
03	Der Cluster soll eine Verfügbarkeit von 98% aufweisen.	Dies kann erst nach dem Testlauf durch ein Monitoring der Laufzeit gemessen werden.	LZ BZ TZ	M
04	Es können während des Betriebs neue Compute Nodes hinzugefügt werden & ausfallende Compute Nodes verursachen keinen Unterbruch des Betriebs.	Während der Testphase werden neue Compute Nodes hinzugefügt und Compute Nodes vom Cluster getrennt.	LZ BZ TZ	M
05	Der Cluster kann für verschiedene Anwendungsgebiete eingesetzt werden.	Während der Testphase werden andere Applikationen, welche die Cluster Ressourcen verwenden sollen, installiert.	BZ	M
06	Das Betriebssystem soll über das Netzwerk an die Compute Nodes verteilt werden, um SD-Karten zu sparen und ein Betriebssystem zentral verwalten zu können.	Wird während der Installation über Systemlogdateien ausgelesen und mit SSH-Zugriffen getestet.	WZ BZ TZ	M
07	Das Schürfprogramm soll automatisch die gewinnbringendste Währung abbauen.	Nach der Testphase werden die Logdateien und Wallets ausgewertet und mit Daten der Währungskurse abgeglichen.	LZ WZ	K
08	Mit der geschürften Währung soll auf Börsen gehandelt werden können.	Kann nach der Realisierung durch Transaktionslogdaten gemessen werden.	LZ WZ	K
09	Die Wartungsarbeiten sollen pro Monat nicht mehr als 3 Stunden betragen.	Wird durch ein Eingriffsprotokoll nach der Realisierungsphase festgehalten.	BZ	K
10	Der Cluster soll einfach transportierbar und wiederaufbaubar sein.	Der Cluster wird nach der Testphase physisch verschoben und neu aufgebaut, dabei wird die Zeit des Wiederaufbaus gemessen.	TZ	K

Tabelle 4: Projektziele

Legende: LZ = Leistungsziel, WZ = Wirtschaftsziel, BZ = Betriebsziel, TZ = Technisches Ziel, M = Muss-Kriterium, K = Kann-Kriterium

1.3.3 Lieferobjekte

Folgende Dokumente werden während des Projektes erstellt und geliefert:

Nr.	Dokument	Phase	Termin
1	Projektplan	Initialisierung	15.02.2018
2	Projektlogo	Initialisierung	20.02.2018
3	Projektauftrag	Initialisierung	20.02.2018
4	Studie	Initialisierung	25.02.2018
5	Detaillkonzept	Konzept	17.03.2018
6	Testkonzept	Konzept	22.03.2018
7	Testprotokoll	Realisierung	03.05.2018
8	Installationshandbuch	Realisierung	10.05.2018
9	Betriebshandbuch	Realisierung	13.05.2018
10	Diplombericht	Einführung	22.05.2018

Tabelle 5: Lieferobjekte

1.3.4 Rahmenbedingungen

Folgende Bedingungen gelten für die Durchführung des Projektes:

- Dem Projekt stehen 294 Stunden Arbeitszeit zur Verfügung.
- Es wird nach der Projektmethode HERMES gearbeitet.
- Der Stundenansatz der involvierten Personen ist auf 120.00 CHF angesetzt.
- Neue Anforderungen werden erst nach dem Projektabschluss berücksichtigt.

1.3.5 Abgrenzungen

- Das Projekt wird für den privaten Nutzen durchgeführt.
- Die Experten sind in den Kostenrechnungen nicht berücksichtigt.
- Der Cluster wird aus Kosten- und Leistungsgründen nicht redundant aufgebaut.
- Neue Anforderungen können während des Projektes nicht berücksichtigt werden.
- Das Projektbudget kann aus finanziellen Gründen nicht erhöht werden.
- Defekte Computenodes werden während des Projektes nicht ersetzt.

1.4 Lösungsbeschreibung

Es wird physisch ein Cluster aus mindestens 40 aktiven RPI's aufgebaut. Der Cluster soll aus finanziellen Gründen mit möglichst wenig Komponenten, wie Netzteile, USB Kabel & Speicherkarten in Betrieb genommen werden. Dabei wird das Betriebssystem zentral verwaltet und über das Netzwerk an die einzelnen Raspberry PI's verteilt. Zugleich wird zur Datensicherheit ein Netzwerkshare (NAS) mit RAID I installiert. Zusätzlich soll die rentabelste Währung automatisch für eine Woche geschürft werden, bevor eine erneute Prüfung auf die rentabelste Währung geschieht. Die geschürften Kryptowährungen werden jeweils in die entsprechenden verschlüsselten Wallets transferiert. Durch die geschürften Währungen soll an Börsen gehandelt werden können, welche es ermöglichen sollen, Profit zu erzielen. Durch eine Monitoring-, Alarming- und Logdaten-Lösung soll auf Missstände des Clusters aufmerksam gemacht werden. Die Tools bieten sich sogleich für eine Analyse der Probleme an und sind über einen Webbrowser aufrufbar.

1.5 Kosten

1.5.1 Einmalige Kosten

Beschaffungskosten

Anzahl	Komponente	Stückpreis(CHF)	Gesamtwert(CHF)
40	Raspberry PI Model B+	33.00	1'320.00
1	Schaltnetzteil	229.00	229.00
1	TTL Serial Kabel	30.00	30.00
40	Patchkabel Cat. 5e	1.00	40.00
1	TP-Link Switch	220.00	220.00
1	Synology NAS DS216	600.00	600.00
1	Diverse Kabel, Schrauben	50.00	50.00
-	Total	-	2'489.00

Tabelle 6: Beschaffungskosten

Aufwandskosten

Stunden	Phase	Stundenansatz(CHF)	Gesamtkosten(CHF)
30	Initialisierung	120.00	3'600.00
41	Konzept	120.00	6'000.00
171	Realisierung	120.00	17'040.00
25.5	Einführung	120.00	2'640.00
25	Periodische Arbeiten	120.00	3'000.00
1.5	Reserve / Administration	120.00	3'000.00
294	Total	120.00	35'280.00

Tabelle 7: Aufwandskosten

1.5.2 Betriebskosten (repetitiv)

Folgende Voraussetzungen sind für die folgenden Berechnungen definiert:

- 1000 Watt die Stunde kostet 0.2894 CHF
- Ein Monat hat 30 Tage
- Ein Jahr hat 360 Tage

Wartungskosten

Pro Monat sind 3 Stunden Wartungsaufwand einzuberechnen, dadurch ergeben sich mit dem definierten Stundenansatz jährliche Wartungskosten von **4'320.00 CHF**.

Stromkosten Der Strom wird durch die BKW über den Vertrag Energy Blue bezogen.

Anzahl	Leistung	Kosten in CHF			
RPI	kW	Stunde	Tag	Monat	Jahr
40	0.4	0.11576	2.78	83.35	1'000.20

Tabelle 8: Stromkostenrechnung

1.5.3 Gesamtkosten

Jährliche Kosten

Das Projekt wird Anfang Juni abgeschlossen sein. Deshalb belaufen sich die Wartungs- und Stromkosten im 1. Jahr auf die Hälfte gegenüber den Folgejahren. Der Cluster soll innerhalb von 3 Jahren gewinnbringend wirken. Die jährlichen sowie täglichen Kosten (in CHF) sind der nachfolgenden Tabelle zu entnehmen:

Kostengrund	Kosten 1. Jahr	Kosten 2. Jahr	Kosten 3. Jahr
Beschaffung	2'489.00	-	-
Aufwand	35'280.00	-	-
Wartungskosten	2'160.00	4'320.00	4'320.00
Stromkosten	500.00	1'000.00	1'000.00
Total	40'429.00	45'749.00	51'069.00

Tabelle 9: Gesamtkosten

Tägliche Kosten

Auf 3 Jahre ausgerechnet, muss täglich ein Ertrag von 47.29 CHF erwirtschaftet werden, um die investierten Aufwände und Kosten zu decken.

1.6 Wirtschaftlichkeit

Das Projekt wird für den privaten Nutzen und aus eigenem Interesse aufgebaut. Aus diesem Grunde ist die Wirtschaftlichkeit kein Kernpunkt des Clusters.

1.6.1 Spekulation

Das Ziel des Clusters ist es, täglich **30 CHF** zu erwirtschaften. Dieser Wert ist nicht deckungsgleich mit den täglichen Kosten, soll aber über Marktspekulationen gedeckt werden. Durch die volatilen Märkte sind Kursschwankungen in beide Richtungen möglich. Es ist jedoch davon auszugehen, dass die Währungen in Zukunft noch an Wert zulegen werden, sobald diese einmal als geltende Zahlungsmittel aufgenommen werden. Durch reine Betriebskosten ohne Spekulation **47.29 CHF** ergibt sich ein tägliches Defizit von **17.29 CHF**, welches einem Verlust von **36.56%** entspricht.

1.6.2 Infrastruktur

Beim Projekt wird der Fokus der Wirtschaftlichkeit hauptsächlich auf den Aufbau gelegt. Hier gilt es, möglichst wenige überflüssige Komponenten zu benutzen. Es wurde darauf geachtet, dass die Komponenten durch eine zentrale Stelle versorgt werden. Dabei werden die Raspberry PI's mit nur einem Netzteil versorgt und das Betriebssystem wird über das Netzwerk verteilt, was Speicherkarten einspart.

Anzahl	Komponente	Stückpreis in CHF	Gesamtwert in CHF
Standardlösung			700.00
4	USB-HUB 10 Ports	35.00	140.00
40	Mini-USB Kabel	6.00	240.00
40	MicroSD Karten	8.00	320.00
Projektlösung			268.00
1	Netzteil	230.00	230.00
1	MicroSD Karte	8.00	8.00
-	Diverse Stromkabel	30.00	30.00
Differenz der Lösungen			432.00

Tabelle 10: Wirtschaftlichkeit Hardware

Durch die vorgesehene Hardwarelösung können 432.00 CHF eingespart werden. Dies entspricht einer Einsparung von **261%**.

1.7 Planung

1.7.1 Grober Projektplan

Die folgende Tabelle zeigt die Arbeiten an, welche während des Projektes erledigt werden sollen. Ein grafischer und detaillierter Projektplan ist über das eingerichtete Confluence¹ zugänglich.

Aufgabe	Start	Ende	Dauer in Stunden		
			Soll	Ist	Abw.
0.0 Initialisierung			30	38.5	+8.5
0.1 Projektplan erstellen	06.02.2018	15.02.2018	4	6	+2
0.2 Projektlogo erstellen	06.02.2018	20.02.2018	2	0.5	-1.5
0.3 Studie durchführen	06.02.2018	20.02.2018	18	24	+6
0.4 Projektauftrag erstellen	20.02.2018	25.02.2018	2	4	+2
0.5 Diplombericht erstellen	25.02.2018	28.02.2018	4	4	0
1.0 Konzept			41	38.5	-2.5
1.1 Zwischen-Meeting	01.03.2018	01.03.2018	6	5.5	-0.5
1.2 Detailkonzept erstellen	05.03.2018	17.03.2018	12	18	+6
1.3 Testkonzept erstellen	18.03.2018	22.03.2018	11	12	+1
1.4 Dokumenten Review	24.03.2018	26.03.2018	12	3	-9
2.0 Realisierung			171	176	+5
2.1 Physischer Aufbau	03.04.2018	07.04.2018	20	32	+12
2.2 Stromversorgung einrichten	08.04.2018	09.04.2018	8	7	-1
2.3 Raspberry PI's vorbereiten	17.04.2018	17.04.2018	4	6	+2
2.4 Netzwerkboot einrichten	21.04.2018	23.04.2018	25	25	0
2.5 Cluster Software installieren	24.04.2018	25.04.2018	25	20	-5
2.6 Schürf Software installieren	26.04.2018	26.04.2018	12	10	-2
2.7 Entwickeln von Tools und Automatismen	02.04.2018	28.04.2018	30	17	-13
2.8 Monitoring einrichten	01.05.2018	10.05.2018	14	7	-7
2.9 Periodische Systemtests	10.04.2018	13.05.2018	7	30	+23
3.0 Installationshandbuch erstellen	02.04.2018	10.05.2018	8	10	+2
3.1 Testprotokoll erstellen	02.05.2018	03.05.2018	9	9	0
3.2 Betriebshandbuch	01.05.2018	13.05.2018	9	3	-6
3.3 Freigabe zur Einführung	07.05.2018	15.05.2018	0	0	0
4.0 Einführung			25.5	27	+1.5
4.1 Abschlussbericht	17.05.2018	22.05.2018	9.5	15	+5.5
4.2 Management Summary	20.05.2018	24.05.2018	8	2	-6
4.3 Vorbereitung Abschluss Meeting	22.05.2018	27.05.2018	3	5	+2
4.4 Drucken und Binden	24.05.2018	01.06.2018	2	2	0
4.5 Abschluss-Meeting	02.06.2018	02.06.2018	2	2	0
4.6 Projektabschluss	03.06.2018	03.06.2018	1	1	0

Tabelle 11: Grober Projektplan

¹<http://wiki.influ.ch/download/attachments/327735/NEBULA-Projektplan-v1.xlsx?api=v2>

1 Initialisierung

1.7.2 Termine

Ereignis	Datum	Teilnehmer	Standort
Einmalige Ereignisse			
Kick-Off Meeting	05.02.2018	Projektleiter & Experten	Post IT, Zollikofen
Zwischenmeeting	01.03.2018	Projektleiter & Experten	GIBB (TSBE), Bern
Abgabe des Diplomberichts	01.06.2018	Projektleiter	-
Abschlussmeeting	07.06.2018	Projektleiter & Experten	GIBB (TSBE), Bern
Periodische Ereignisse			
Statusbericht	Monatlich	Projektleiter	-
Arbeitsjournal	2x Wöchentlich	Projektleiter	-

Tabelle 12: Termine

1.8 Ressourcen

1.8.1 Budget

Dem privaten Projekt steht ein Budget von 3'250 CHF zu. Die Aufwände werden hierbei nicht berücksichtigt, da keine Löhne bezahlt werden müssen.

Nr.	Verwendungszweck	Budget in CHF
1	Beschaffungen	3'000.00
2	Apéro	150.00
3	Drucken & Binden	100.00
Total		3'250.00

Tabelle 13: Projektbudget

1.8.2 Sachmittel

Die aufgelisteten Komponenten werden für die Lösung benötigt.

Nr.	Anzahl	Komponenten	Modell / Spezifikationen
1	40	Mini Computer	Raspberry PI 3 Model B+
2	1	Schaltnetzteil	RSP-750-5, Mean Well
3	1	USB zu TTL Serial-Kabel	Adafruit USB zu TTL Serial Kabel, 75cm
4	40	Ethernetkabel	FTP Cat.5e Patchkabel
5	1	Switch	TL-SL3452 48-Port 10/100, TP-Link
6	1	Datenspeicher	Synology NAS DS218
7	*	Diverse Kabel & Befestigungsmaterialien	*

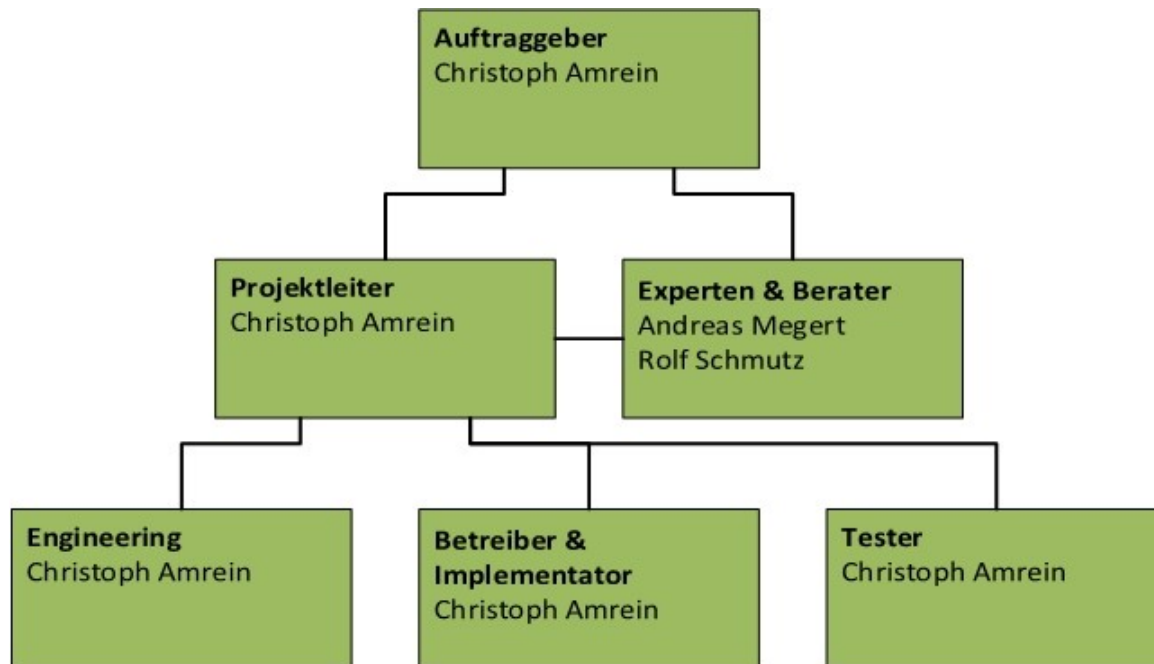
Tabelle 14: Sachmittel

* Anzahl und Hersteller unbekannt. Die Artikel wurden in lokalen Baumärkten eingekauft.

1.9 Organisation

1.9.1 Projektorganisation

Die Projektorganisation ist wie folgt aufgebaut:



Rolle	Verantwortlichkeit
Auftraggeber	Erstellt den Auftrag und übergibt diesen an den Projektleiter
Projektleiter	Organisiert die Planung, Durchführung und präsentiert das Projekt
Experte	Stehen in Kontakt mit dem Projektleiter und beraten ihn bei Schwierigkeiten
Engineering	Stellt dem Betreiber zu implementierende Applikationen zur Verfügung
Betreiber	Setzt die Lösung technisch um
Tester	Testet die Lösung auf Fehler

Tabelle 15: Organisation

1.9.2 Projektablage

Nr.	Was	Wo
1	Allgemeine Ablage	wiki.influ.ch
2	Dokumentation	https://github.com/amreinch/Nebula_AMC
3	Snapshots	Lokal, D:\Diplomarbeit\CentOS_works
4	Skripte, Entwürfe	https://github.com/amreinch/OpenHPC_Install_Nebula

Tabelle 16: Projektablage

1.10 Cluster-Software Evaluation

1.10.1 Cluster Software Kriterien

Es wurden drei Cluster Software Produkte evaluiert. Dabei mussten die Muss-Kriterien erfüllt werden, um in die Auswahl zu kommen. Diese Kriterien werden für den Entscheid der Software nicht berücksichtigt, grenzen aber die Auswahlmöglichkeit der Produkte ein.

Nr.	Anforderung	Prio.
01	Ist die Software HPC tauglich?	M
02	Kann das Produkt innerhalb des vorgesehenen Zeitraumes installiert werden?	M
03	Ist die Lösung skalierbar?	M
04	Existieren Dokumentationen?	S
05	Kann Support beansprucht und bezogen werden?	S
06	Ist die Lösung benutzerfreundlich?	S
07	Existieren Verwaltungstools?	S
08	Fallen zusätzliche Kosten an?	S

Tabelle 17: Software Kriterien

1.10.2 Informationsbeschaffung

Es wurde nach einer Lösung gemäss der oben definierten Kriterien gesucht. Dabei bin ich auf einen Wikipedia Eintrag² gestossen, welcher die verschiedenen Cluster Software Angebote auflistet. Dabei wurde nur nach einer HPC Lösung gefiltert. Durch diese Analyse hat sich die OpenHPC Lösung der Linux Foundation herauskristalisiert. Weiterhin wurden Suchbegriffe wie "HPC Raspberry PI" über Suchmaschinen eingegeben, da die Compute Nodes des Clusters Raspberry PI's sein sollen. Den Artikel über TinyTitan³ habe ich als interessant erachtet und deshalb genauer untersucht. Mit einer weiteren Suche(hpc cluster software raspberry) bin ich auf einen Guide⁴ gestossen, der relativ simpel aussieht und einfach umzusetzen ist. Es gab durchaus noch weitere Guides und Softwarelösungen, welche ich aber nach einer genaueren Analyse der Installationsanleitung verworfen habe, da diese mir zum Teil zu wenig Informationen lieferten. Während der Informationsbeschaffung wurden alle Installationsskripte und Anleitungen sorgfältig durchgelesen, um diese als mögliche Variante zu empfehlen. Während der Informationsbeschaffung bin ich auf zwei Fachbegriffe (MPI & SLURM), welche meistens in Zusammenhang mit HPC stehen, gestossen. Diese musste ich ebenfalls noch in Erfahrung bringen.

²https://en.wikipedia.org/wiki/Comparison_of_cluster_software

³<http://www.hpctoday.com/best-practices/tinytitan-a-raspberry-pi-computing-based-cluster/>

⁴<http://thundaxsoftware.blogspot.ch/2016/07/creating-raspberry-pi-3-cluster.html>

1.11 Lösungsvarianten

1.11.1 Variantenübersicht

Die Informationen wurden über die folgenden Produkte gesammelt und zusammengestellt:

Nr.	Variante	Bezeichnung
01	OpenHPC	HPC Lösung entwickelt von der Linux Foundation
02	TinyTitan	Open Source Lösung entwickelt von Oak Ridge Leadership Computing Facility
03	Minimale Lösung	Simple 32-Bit Architekturlösung

Tabelle 18: Variantenübersicht

1.11.2 Variante V1 «OpenHPC»

Beschreibung

OpenHPC gilt als vorangeschrittenes OpenSource Projekt der Linux Foundation. Das Produkt steht in direkter Verbindung mit diversen grossen IT Unternehmen weltweit. Das Ziel der Linux Foundation ist es, durch OpenHPC eine kostengünstige sowie schnell zu installierende HPC-Umgebung aufzubauen. Durch viele zusätzliche OpenSource Tools rundet sich das Produkt ab und gilt als ernstzunehmender Konkurrent gegenüber kostenpflichtiger Software.

Installation und Betrieb

Es existieren diverse Guides, Foren und Chats sowie eine E-Mail-Liste zu OpenHPC. Dadurch scheint die Unterstützung bei allfälligen Problemen vorhanden zu sein. Die Installationsanleitung, welche von der Linux Foundation geschrieben wurde, liest sich sehr gut und ist absolut ausreichend für die Installation. Der Betriebsaufwand wird als gering eingeschätzt, da es ein sehr ausgereiftes Produkt, welches stetig weiterentwickelt wird, ist.

Voraussetzungen, Abhängigkeiten

Für die Cluster Software werden mindestens ein Masternode und 4 Computenodes vorausgesetzt. Das Betriebssystem bezieht sich hierbei auf ein CentOS7x. Jeder Computenode benötigt 2 Netzwerkschnittstellen. Das eine Interface wird für den Standard Ethernet Zugriff verwendet und das zweite Interface wird für die Kommunikation zu jedem BMC Host verwendet. Es werden zusätzliche Intel Bibliotheken benötigt. Dazu müssen Lizenzen für Parallel Studio XE von Intel besorgt werden. Die Lizenzen können mit einer offiziellen E-Mail-Adresse der Schule gratis bezogen werden. Die Linux Foundation erwähnt in ihrem Guide, dass sie die «Bring your own Licence» Strategie verfolgt.

1.11.3 Variante V2 «TinyTitan»

Beschreibung

Das Produkt wurde von der Firma «Oak Ridge Facility» entwickelt. Die Software ist unter anderem

1 Initialisierung

für RPI's entwickelt worden. TinyTitan wurde für das Durchführen wissenschaftlicher Berechnungen entworfen. Jedoch wurde seit geraumer Zeit an dem Produkt nicht mehr weitergearbeitet, wie dem offiziellen GitHub Repository zu entnehmen ist. Die Community selbst erweist sich ebenfalls als sehr klein.

Installation und Betrieb

Für die Installation des Produktes wird ein XServer vorausgesetzt, da empfohlen wird Tiny-Titan über ein GUI zu installieren. Der Installationsanleitung ist ebenfalls zu entnehmen, dass sich die Entwickler viele Gedanken über das Look a Like des Clusters gemacht haben, zum Beispiel wird ein Thema dem Einbinden von LED's gewidmet. Die Installation findet ausschliesslich durch vordefinierte Scripts statt. Durch die kleine Community und nicht mehr gepflegte Software kann nichts über den Betriebsaufwand in Erfahrung gebracht werden.

Voraussetzungen, Abhängigkeiten

Laut Guides werden lediglich 2 RPI's benötigt.

1.11.4 Variante V3 «Minimale Lösung»

Beschreibung

Die Minimale Installation ist eine zum Teil Eigenbau Lösung, welche sich nahe an diverse Guides aus dem Internet bezieht. Jedoch wird diese auf eigene Bedürfnisse angepasst.

Installation und Betrieb

Da es bei dieser Lösung selber zu entscheiden gibt, was und wie die Lösung installiert und umgesetzt werden soll, kann während der Installation darauf geachtet werden, was den grössten Vorteil für den Betrieb danach mit sich bringt. Während dem Projekt soll die Installation aber klein gehalten werden und nur das nötigste wird umgesetzt.

Voraussetzungen, Abhängigkeiten

Es werden 2 RPI's benötigt.

1.11.5 Anforderungsabdeckung der Varianten

Nr.	Kriterium	Gewichtung	Begründung
1	Installation	40%	Die Installation soll keine Hürden aufweisen, da der Zeitplan ansonsten nicht eingehalten werden kann.
2	Partner	10%	Je mehr Partner vorhanden sind, desto grösser und innovativer ist die Software. Die Software hat dadurch einen fixen Standpunkt auf dem Markt und wird weiterentwickelt.
3	Aktualität	20%	Fördert den LifeCycle und die Sicherheit der Cluster-Software.
4	Tools	30%	Mitgelieferte Tools

Tabelle 19: Anforderungsabdeckung

1 Initialisierung

Nr.	Kriterium	Note	Begründung
1	Installation	0/3/5	5 = Kann gemäss Anleitung direkt installiert werden. 3 = Veraltete Anleitung, es wird mit Kompatibilitätsproblemen gerechnet. 0 = Keine Anleitung vorhanden.
2	Partner	0/2	2 = Viele Partner vorhanden. 0 = Keine Partner vorhanden
3	Aktualität	0/3/5	5 = Releases in den letzten 2 Monaten. 3 = Releases in den letzten 6 Monaten 0 = Keine Releases seit einem Jahr.
4	Tools	0/5	5 = Es werden Tools angeboten 0 = Es werden keine Tools angeboten.

Tabelle 20: Notenskala der Kriterien

1.12 Bewertung der Varianten

Kriterium	Gewicht	Variante 1	Variante 2	Variante 3
Installation	40%	5x40 = 200	4x40 = 160	5x40 = 200
Partner	10%	2x10 = 20	0x10 = 0	0x10 = 0
Aktualität	20%	5x20 = 100	0x20 = 0	5x20 = 100
Tools	30%	5x30 = 150	0x30 = 0	0x30 = 0
Total	100%	470	160	300

Tabelle 21: Bewertung der Varianten

1.13 Variantenentscheid

Anhand der Bewertung wird empfohlen, die OpenHPC Lösung der Linux Foundation zu verwenden. Die Installation kann gemäss Anleitung in kürzester Zeit umgesetzt werden. Die Releases können mit kleinerem Aufwand installiert werden. Zudem runden die Möglichkeiten der Schnittstellen und Komponenten den Entscheid ab. Es ist möglich, Administrations- sowie Performance Monitoring Tools einzusetzen, welche mit der Lösung harmonieren. Als Hürde sehe ich die möglichen anfallenden Lizenzen und das zweite Netzwerk Interface, welches man für die Kommunikation unter den RPI's benötigt.

1.14 Risiken

Der untenstehenden Abbildung kann entnommen werden, welche Risiken während des Projekts existieren. Dabei sind Eintrittswahrscheinlichkeit und Auswirkungen tabularisch aufgelistet.

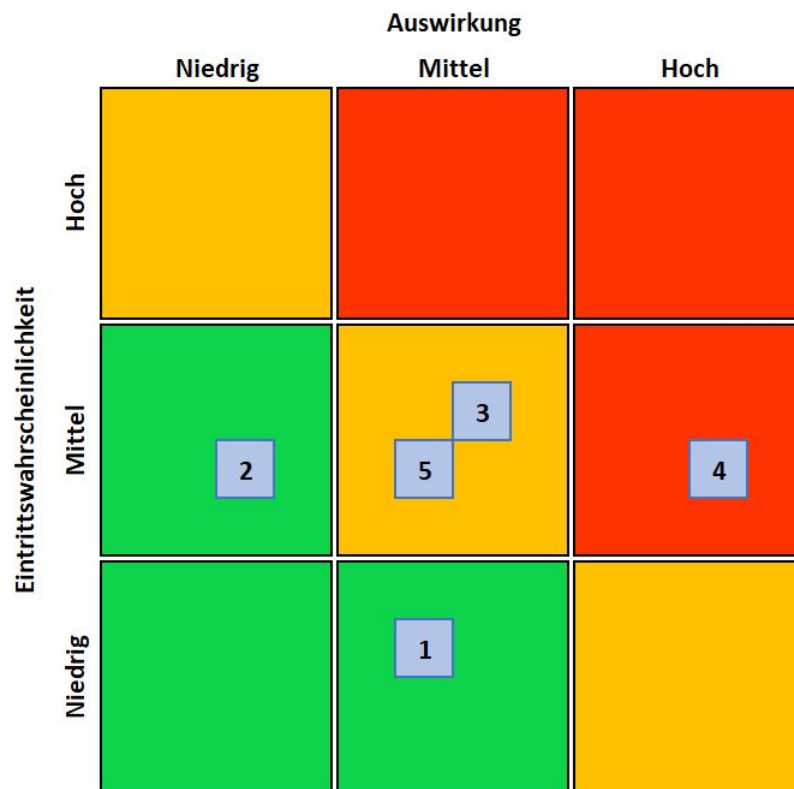


Abbildung 1: Risiken

Nr.	Beschreibung	Massnahmen zur Problemlösung
1	Der Terminplan kann nicht eingehalten werden	- Zeitplan anpassen - Experten informieren und nach einer Lösung suchen
2	Ausfall durch Unfall	Experten informieren und nach einer Lösung suchen
3	Technische Umsetzungsprobleme	- Informieren der Experten - Hilfe der Experten einholen - Alternative Lösung umsetzen
4	Defekte Hardware (Switch, Netzteil)	Hardware muss umgehend neu beschafft werden
5	Softwarefehler	Patches einspielen, Kontakt mit Lieferanten aufnehmen

Tabelle 22: Risiken

2.1 Physikalischer Überblick

```

graph TD
    Internet((Internet)) --- Router[Router]
    Router --- PC[PC]
    Router --- Switch[Switch]
    NAS[NAS FileShare] --- Switch
    Switch --- Managementnode[Managementnode]
    Switch --- Computenodes[Computenodes<br/>40 aktive<br/>5 passive]
    Managementnode --- 1((1)) --- NetzteilManagement[Netzteil Managementnode]
    Computenodes --- 2((2)) --- Schaltnetzteil[Schaltnetzteil<br/>Computenodes]
  
```

Beschreibung

Verbindung 1

Verbindung 2

Die Compute Nodes werden über ein Schaltnetzteil über die GPIO Pins mit Strom betrieben.

2.2 Physikalische Verbindungen

2.2.1 Stromversorgung Management Node

Der Management Node wird über den Micro USB Anschluss mit Strom versorgt. Dabei muss darauf geachtet werden, dass ein mindest Strom von 2 Ampere fließt. Zudem wird eine konstante Spannung von 5 Volt benötigt. Deshalb wird ein Netzteil mit einer Leistung von 10 Watt verwendet. Das Netzteil wird über eine Stromschiene an das Stromnetz angeschlossen.

2.2.2 Compute Nodes

Die Compute Nodes werden über die GPIO Pins via Jumperkabel über ein gemeinsames Netzteil mit Strom versorgt. Da es sich hierbei um eine Anzahl von mindestens 45 Raspberry's handelt, ist ein Netzteil mit einer Leistung von 500W vorgesehen. Das Netzteil wird über die Stromschiene an das Stromnetz angeschlossen.

2.2.3 Übrige Geräte

Die übrigen Geräte werden über den herkömmlichen Weg mit Strom über eine Stromschiene versorgt.

2.2.4 Netzwerkverbindungen

Die folgenden Komponenten sind über den Switch in das lokale Netzwerk eingebunden. Die nicht aufgelisteten Geräte werden direkt über Powerline oder WLAN mit dem Router verbunden:

- Management Node
- Compute Nodes
- NAS

2.3 Technischer Überblick

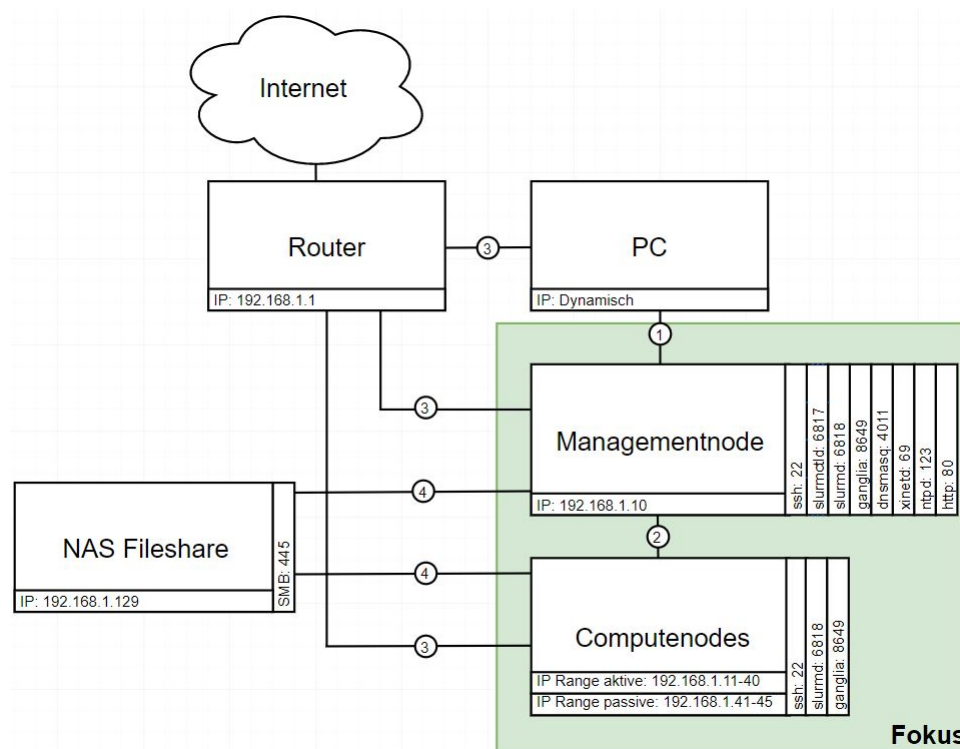


Abbildung 3: Technischer Überblick

Beschreibung

Der grün markierte Teil beinhaltet das Vorhaben. Diese Komponenten werden neu in das Netzwerk eingebunden und aufgebaut. Die Komponenten ausserhalb des grünen Bereiches existieren bereits und es müssen für die Umsetzung Konfigurationen vorgenommen werden.

Verbindung 1

Der PC kann mit dem **SSH Protokoll** auf den Management Node zugreifen. Dadurch kann die Installation vorgenommen werden. Zugleich wird über **HTTP** via Webbrowser der Zugriff auf diverse Applikationen, wie z.B. Nagios & Ganglia, ermöglicht.

Verbindung 2

Der Management Node verteilt via **dnsmasq** und **TFTP** das Betriebssystem an die Compute Nodes über das Netzwerk. Sogleich ist auch der **Slurm Controller** für die Jobsteuerung auf dem Management Node installiert, welcher mit den **Slurm Daemons** auf den Compute Nodes kommuniziert. Weiterhin sind die Monitoring Komponenten **Ganglia** und **Nagios** auf dem Management Node installiert, welche Monitoringdaten der Compute Nodes sammeln und zur Auswertung verarbeiten.

Verbindung 3

Der Router verteilt via **DHCP** statische IP Adressen und Hostnamen, welche über die MAC Adressen definiert sind.

Verbindung 4

Der NAS Share wird über **SMB** auf den Compute Nodes und dem Master Node angehängt.

2.3.1 Verwendete Protokolle

Verbindung	Protokoll	Protokollfamilie	Ports
1	SSH	TCP	22
2	SMTP	TCP	25
3	DHCP	UDP	67 / 78
4	TFTP	UDP	69
5	HTTP	TCP	80
6	SMB	TCP	445

Tabelle 23: Protokolle

2.4 Technische Verbindungen & Kommunikation

Nr.	Quelle	Ziel	Betrifft	Beschreibung
1	NAS	Mgmt	Datenablage	Der NAS Share wird über das SMB Protokoll angehängt.
2	NAS	Compute	Datenablage	Der NAS Share wird über das SMB Protokoll angehängt.
3	Router	Mgmt	IP Adresse	Anhand der MAC Adresse wird eine statische IP Adresse zugewiesen.
4	Router	Compute	IP Adressen	Anhand der MAC Adressen werden statische IP Adressen zugewiesen.
5	Router	Mgmt	Hostname	Es wird über den Router ein definierter Hostname verteilt.
6	Router	Compute	Hostnamen	Es werden über den Router definierte Hostnamen verteilt.
7	Mgmt	Compute	Netzwerkboot	Der Management Node beliefert die Compute Nodes über das TFTP Protokoll mit dem Betriebssystem
8	Internet	Mgmt	Zeitserver	Die aktuelle Zeit wird mit NTP über das Internet synchronisiert.
9	Mgmt	Compute	Zeitserver	Die Compute Nodes beziehen die aktuelle Zeit über NTP.
10	Internet	Compute	Internetzugriff	Die Compute Nodes können über ein Routing über den Mgmt auf das Internet zugreifen.
11	PC	Mgmt	Zugriff	Verbindungen über den PC können mit dem SSH Protokoll aufgebaut werden.
12	PC	Compute	Zugriff	Verbindungen über den PC können mit dem SSH Protokoll aufgebaut werden.

Tabelle 24: Verbindungen & Kommunikation

Legende: Mgmt = Management Node, Compute = Compute Nodes, PC = Home Computer

2.5 Komponentenbeschreibung

2.5.1 Router

Bei dem Router handelt es sich um eine Internet-Box Plus von Swisscom. Das Admin Interface ist über <http://internetbox> aufrufbar.

2.5.2 PC

Der PC ist selbst zusammengestellt und wird für den SSH Zugriff auf den Management Node und für Zugriffe auf die Webanwendungen des Clusters benötigt.

2.5.3 Management Node

Der Management Node dient der Jobsteuerung sowie Clusterverwaltung. Alle zentralen Programme sind auf diesem Node installiert.

Hostname	nebula
Modell	Raspberry PI 3 B
Betriebssystem	Centos 7.4
CPU	Quad Core 1.2GHz Broadcom BCM2837 64bit CPU
RAM	1GB

Tabelle 25: Komponente Management Node

2.5.4 Netzteil Management Node

Das Netzteil liefert eine konstante Spannung von 5V und Strom von mindestens 2 Ampere. Dabei handelt es sich um ein Noname Netzteil, welches eine Mindestleistung von 10 Watt aufbringen muss.

2.5.5 NAS

Das NAS ist von der Firma Synology. Das Modell lautet DS216 und wird als redundanter Datenspeicher benutzt.

2.5.6 Switch

Der Managed Switch TL-SL3428 von TP-Link wird für die Kommunikation zwischen NAS, Router, Management Node und den Compute Nodes benötigt. Auf die Managed Funktion wird allerdings während des Aufbaus und Betriebes verzichtet.

2.5.7 Compute Nodes

Die Compute Nodes erhalten über das Netzwerk das Betriebssystem durch den Management Node zugestellt. Dabei sind alle Hostnamen der Compute Nodes mit dem Prefix "c" versehen und werden aufnummeriert. Dabei sind die Compute Nodes in aktiv und passiv (Fallback, Reserve) aufgeteilt. Die passiven Compute Nodes sollen ausgefallene aktive Compute Nodes ersetzen und deren Arbeiten übernehmen und die Leistung des Clusters konstant halten.

Aktiv

Hostname	c[1-40]
Modell	Raspberry PI 3 B
Betriebssystem	Centos 7.4
CPU	Quad Core 1.2GHz Broadcom BCM2837 64bit CPU
RAM	1GB

Tabelle 26: Komponente aktive Compute Nodes

Passiv

Hostname	c[41-45]
Modell	Raspberry PI 3 B
Betriebssystem	Centos 7.4
CPU	Quad Core 1.2GHz Broadcom BCM2837 64bit CPU
RAM	1GB

Tabelle 27: Komponente passive Compute Nodes

2.5.8 Schaltnetzteil Compute Nodes

Das Schaltnetzteil RSP-750-5 von Mean Well liefert konstante 5 Volt als Ausgangsspannung und kann eine Leistung bis zu 500 Watt aufbringen. Daraus können 100 Ampere auf die Nodes verteilt werden.

2.6 Tests

2.6.1 Testobjekte

Die folgende Hardware ist für die Tests der Funktionsfähigkeit des Clusters im Scope vorgesehen:

2 Konzept

Nr.	Objekt	Beschreibung
1	Management Node	Raspberry PI 3
2	Compute Nodes	Raspberry PI 3
3	NAS	Synology NAS DS216
4	Switch	TP-Link TL-SL3428

Tabelle 28: Testobjekte

2.6.2 Testarten

Die Tests werden in folgende Kategorien eingestuft:

Nr.	Testart	Beschreibung
1	Komponententest	Die Lauffähigkeit und Erreichbarkeit der einzelnen Hardware Komponenten wird überprüft.
2	Integrationstest	Es wird die Zusammenarbeit der aktiven und neu integrierten abhängigen Komponenten überprüft.
3	Systemtest	Das System wird als Komplettlösung getestet. Hierbei soll geprüft werden, ob die Lösung den Anforderungen der Anwendbarkeit und Nutzbarkeit dem Auftrag entspricht.

Tabelle 29: Testarten

2.6.3 Testvoraussetzungen

Startbedingungen

Für den Start der Tests muss der Cluster aufgebaut sein und die einzelnen Komponenten müssen mit Strom versorgt sein.

Abbruchbedingungen

Die Tests werden abgebrochen, sobald Fehler auftauchen, welche Folgetests verhindern.

2.6.4 Fehlerklassen

Nr.	Fehlerklassen	Beschreibung
1	Fehlerfrei	Die Erwartungen sind erfüllt.
2	Harmloser Mangel	Es sind keine Betriebsverhinderungen zu erkennen. Die Erwartungen sind erfüllt.
3	Kleiner Mangel	Der Betrieb kann aufgenommen werden. Das Problem sollte aber über einen Zeitraum von 6 Monaten behoben werden.
4	Schwerer Mangel	Der Cluster kann nur teilweise in Betrieb genommen werden. Der Mangel muss innerhalb zwei Wochen behoben werden.
5	Kritischer Mangel	Der Cluster kann nicht in Betrieb genommen werden. Die Mängel müssen umgehend behoben werden.

Tabelle 30: Fehlerklassen

2.6.5 Testhilfsmittel

Die Dokumentation der Tests wird im Testprotokoll nachgeführt. Damit die Tests durchgeführt werden können wird ein PC oder Notebook als Testclient benötigt. Dieser Client muss sich im selben Netzwerk wie der Cluster befinden.

2.6.6 Testfazit

Die Tests wurden gemäss Testkonzept durchgeführt worden. Alle Testfälle sind im Anhang E dokumentiert. Das Testprotokoll befindet sich im Anhang F und beinhaltet alle Resultate der durchgegangenen Testfälle. Dem Testprotokoll ist zu entnehmen das ein gutes Resultat erzielt wurde. Dadurch sollten keine Hindernisse bei der Inbetriebnahme des Clusters anzutreffen sein. Während der Tests sind jedoch kleinere Probleme aufgefallen, welche behoben werden müssen.

K-037: Compute Node c35

Der Compute Node c35 ist nicht mehr funktionsfähig und muss innerhalb von 6 Monaten ausgetauscht werden. Während dieser Zeit wird auf einen unbestimmten Reserve Node c[41-45] zugegriffen. Der Compute Node c35 soll vom Netzwerk und von der Stromversorgung abgekoppelt werden um Energie zu sparen.

S-004: Ganglia Monitoring

Es wurde festgestellt, dass Ganglia jeweils alle Daten zusammenfasst und auf einem Node darstellt. Die Ursache ist noch nicht bekannt und es müssen Analysen stattfinden. Da Ganglia als Performance Messungs Tool benutzt wird, ist dies nicht kritisch. Eine Lösung sollte dennoch innerhalb von 6 Monaten gefunden werden.

2.7 Monitoring

Als Monitoring-Lösungen werden die Applikationen Nagios und Ganglia eingesetzt. Die Einsatzgebiete sind wie folgt definiert:

- Nagios = Service Monitoring
- Ganglia = Performance Monitoring

2.7.1 Service Monitoring - Nagios

Sämtliche Service Tests werden vom Management Node aus automatisch in definierten Intervallen ausgeführt. Fehlgeschlagene Tests sowie Statusänderungen der Überwachungsstatus generieren eine Benachrichtigungs-E-Mail, welche an den Systemadministrator versendet wird. Nagios ist über den Browser via <http://nebula/nagios> zu erreichen.

Die folgenden Überwachungen sollen während der Realisierungsphase implementiert werden. Die Intervalle sind in Minuten angegeben. Weitere Überwachungen können nach dem Projektabschluss implementiert werden:

Nr.	Überwachung	Schweregrad	Intervall	Beschreibung
1	Erreichbarkeit	Kritisch	5	Es wird mittels Ping eine Statusüberprüfung der Nodes durchgeführt.
2	SSH Zugriff	Mittel	60	Die Zugriffe auf die Compute Nodes sollen über den Management Node stattfinden.
3	CPU Last	Hoch	5	Die CPU's sollen ständig ausgelastet sein.

Tabelle 31: Service Monitoring

2.7.2 Performance Monitoring - Ganglia

Die Ganglia Applikation ist auf dem Management Node installiert und kommuniziert mit den Ganglia Daemons auf den Compute Nodes. Dabei werden die übermittelten Daten als Grafen dargestellt. Ganglia ist über <http://nebula/ganglia> aufrufbar.

2.8 Mining

Die Kryptowährungen werden über die Miningpools von Minergate.com geschürft. Dafür wird die cpuminer Version von [tkinjo1985](https://github.com/tkinjo1985) verwendet. Diese Version unterstützt die ARMv8 Prozessoren und bietet alle gängigen Algorithmen für das Schürfen der Währungen an. Zudem werden nur Währungen geschürft, welche auf Börsen resp. Märkten gehandelt werden können.

2.8.1 Kryptowährungen

Folgende Kryptowährungen werden über die Minergate Pools mit dem CryptoNight Algorithmus geschürft:

Nr.	Währung	Kürzel	Märkte
1	Bytecoin	BCN	HitBTC, Poloniex
2	Monero	XMR	HitBTC, Binance, Bitfinex, Poloniex
3	Monero Original	XMO	HitBTC
4	DigitalNote	XDN	HitBTC, Bittrex
5	Quazar Coin	QCN	HitBTC
6	DashCoin	DSH	HitBTC
7	FantomCoin	FCN	HitBTC

Tabelle 32: Kryptowährungen

2.9 Hostnamen

Die Compute Node Namen wurden nach einem überschaulichen Konzept definiert. Jeder Compute Node trägt den Prefix "c". Dies soll bei der Behebung von Problemen auf physischer Ebene, z.B. Austauschen, eines Nodes dienen. Zudem werden alle Hostnamen immer in kleinen Buchstaben geschrieben.

2.9.1 Management Node Name

Name	IP	MAC
nebula	192.168.1.10	B8:27:EB:32:A9:1C

Tabelle 33: Management Node Name

2.9.2 Reserve Node Name

Die Reservenodes sind als Fallback für ausgefallene Compute Nodes vorgesehen.

Nr.	Name	IP	MAC
1	c41	192.168.1.51	B8:27:EB:DE:C9:69
2	c42	192.168.1.52	B8:27:EB:7E:6F:48
3	c43	192.168.1.53	B8:27:EB:5D:DD:FE
4	c44	192.168.1.54	B8:27:EB:A6:6D:4D
5	c45	192.168.1.55	B8:27:EB:0C:63:10

Tabelle 34: Reserve Node Namen

2.9.3 Compute Node Namen

Nr.	Name	IP	MAC
1	c1	192.168.1.11	B8:27:EB:32:39:A7
2	c2	192.168.1.12	B8:27:EB:2E:A3:D1
3	c3	192.168.1.13	B8:27:EB:50:45:3F
4	c4	192.168.1.14	B8:27:EB:0D:E6:25
5	c5	192.168.1.15	B8:27:EB:3E:96:B5
6	c6	192.168.1.16	B8:27:EB:EE:77:DA
7	c7	192.168.1.17	B8:27:EB:21:63:E6
8	c8	192.168.1.18	B8:27:EB:2E:2E:CC
9	c9	192.168.1.19	B8:27:EB:17:32:96
10	c10	192.168.1.20	B8:27:EB:B2:1C:A9
11	c11	192.168.1.21	B8:27:EB:AF:63:1F
12	c12	192.168.1.22	B8:27:EB:43:00:2C
13	c13	192.168.1.23	B8:27:EB:13:7B:18
14	c14	192.168.1.24	B8:27:EB:43:CD:29
15	c15	192.168.1.25	B8:27:EB:FF:C7:56
16	c16	192.168.1.26	B8:27:EB:CE:98:66
17	c17	192.168.1.27	B8:27:EB:5D:63:34
18	c18	192.168.1.28	B8:27:EB:91:3E:0F
19	c19	192.168.1.29	B8:27:EB:F4:65:EC
20	c20	192.168.1.30	B8:27:EB:3E:AB:DC
21	c21	192.168.1.31	B8:27:EB:66:60:F6
22	c22	192.168.1.32	B8:27:EB:37:3F:74
23	c23	192.168.1.33	B8:27:EB:18:5E:F0
24	c24	192.168.1.34	B8:27:EB:B0:23:B8
25	c25	192.168.1.35	B8:27:EB:BE:C4:94
26	c26	192.168.1.36	B8:27:EB:FB:FF:57
27	c27	192.168.1.37	B8:27:EB:4E:EC:CE
28	c28	192.168.1.38	B8:27:EB:43:1C:35
29	c29	192.168.1.39	B8:27:EB:DC:74:5F
30	c30	192.168.1.40	B8:27:EB:D1:DE:2F
31	c31	192.168.1.41	B8:27:EB:5E:90:34
32	c32	192.168.1.42	B8:27:EB:DE:80:24
33	c33	192.168.1.43	B8:27:EB:A4:79:6F
34	c34	192.168.1.44	B8:27:EB:0A:4D:C7
35	c35	192.168.1.45	B8:27:EB:5C:53:5F
36	c36	192.168.1.46	B8:27:EB:F7:AF:C2
37	c37	192.168.1.47	B8:27:EB:CE:BA:ED
38	c38	192.168.1.48	B8:27:EB:59:38:3C
39	c39	192.168.1.49	B8:27:EB:99:BB:8E
40	c40	192.168.1.50	B8:27:EB:8F:7A:0D

Tabelle 35: Compute Node Namen

3 Realisierung

Dieses Kapitel beschreibt, in welcher Reihenfolge der Cluster aufgebaut wurde. Einen tieferen Einblick des Aufbaus und der verwendeten Installationsbefehle kann dem Anhang entnommen werden.

3.1 Physischer Aufbau

3.1.1 Komponenten Platzierung

Der Cluster ist in einem Gestell, welches 4 Ebenen hat, implementiert. Die Ebenen sind wie folgt aufgeteilt:

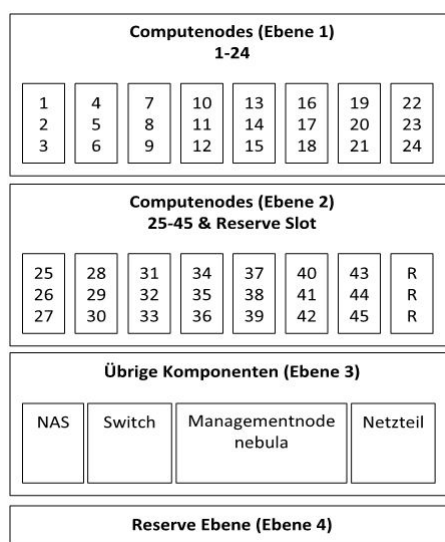


Abbildung 4: Physischer Aufbau

Ebene 1

24 RPI's sind auf dieser Ebene befestigt worden.

Ebene 2

Es befinden sich 21 RPI's auf dieser Ebene. Es können noch 3 weitere platziert werden.

Ebene 3

Hier wurden alle übrigen Komponenten befestigt. Darunter ist das NAS, der Switch, der Management Node und das Netzteil zu finden.

Ebene 4

Auf dieser Ebene wurde nichts installiert. Sie kann als Reserve Ebene betrachtet werden.

3.1.2 Kühlung

Die CPU, RAM und GPU der RPI's wurden mit Aluminium-Kühlkörpern ausgestattet. Die passive Kühlung soll die übertakten CPU's der RPI's am laufen halten.

3.1.3 Stromversorgung

Compute Nodes

Die Compute Nodes wurden über die GPIO Pins 2 (5 Volt Anschluss) und 6 (GND Anschluss) über Jumperkabel und weiteren Leiterkabeln, welche zur Verlängerung dienen, mit dem Netzteil verbunden. Es wurde darauf geachtet, dass der Kabeldurchschnitt für eine Anzahl von mindestens 25 Ampere ausreicht, so dass diese nicht durchbrennen.

Netzteil

Am Netzteil wurden Kabelschuhe befestigt, welche es ermöglichen, eine Verbindung mit den Leitern in Richtung RPI's herzustellen. Das Netzteil ist an einer gewöhnlichen Stromschiene angeschlossen.

Generelle Verkabelung

Die Leiter wurden hauptsächlich mit Lüsterklemmen verlängert und auf das zusammenlöten der Komponenten wurde deswegen verzichtet. Dies bietet für neue Verkabelungen eine grössere Flexibilität.

3.1.4 Kommunikation

Alle Komponenten, welche eine Netzwerkverbindung benötigen, sind über den Switch mit Patchkabeln zusammengeschlossen worden. Es wurde dabei keine spezielle Slotzuweisung des Switches berücksichtigt.

3.2 Technischer Aufbau

3.2.1 Betriebssystem

Es existiert zur Zeit kein 64-Bit CentOS Kernel, welcher mit den RPI's kompatibel ist. Deswegen wurde mit einer alternativen Lösung das Gentoo Image vom Github Repository⁵ von Sakaki heruntergeladen und auf die SD-Karte geschrieben. Dabei wurden zwei Partitionen erstellt: Die Boot-Partition, welche den Kernel und die Bootbefehle beinhaltet, und die Dateisystem-Partition. Diese beinhaltet die Ordnerstruktur und Dateien des Betriebssystems. Diese Partition muss mit der eines CentOS Dateisystems überschrieben werden. Dabei wurde das Dateisystem aus dem offiziellen Repository von CentOS⁶ heruntergeladen und auf die Dateisystem Partition kopiert.

⁵<https://github.com/sakaki-/gentoo-on-rpi3-64bit>

⁶<http://mirror.centos.org/altarch/7/isos/aarch64/CentOS-7-aarch64-rootfs-7.4.1708.tar.xz>

3.2.2 Vorbereitungen

Netzwerk

Die IP- und Hostnamenzuweisung wurde über die Internetbox von Swisscom eingerichtet. Dabei wurden alle RPI's an das Netzwerk und den Strom angeschlossen. Nach ca. 2 Minuten wurden alle angeschlossenen Geräte im Interface aufgelistet und konnten anhand des Hostnamenkonzeptes eingerichtet werden.

Netzwerkboot

Der Management Node dient als Provider und verteilt das Betriebssystem an alle angeschlossenen Compute Nodes. Diese mussten vorgängig bearbeitet werden und benötigen bei einem Power On einen OTP Eintrag. Dadurch wird eine Anfrage von jedem Compute Node (Client) an den Management Node gestellt, ob es das Betriebssystem erhalten darf. Zeitgleich wurden noch alle MAC-Adressen ausgelesen, diese werden später für die statische Zuweisung von IP Adressen und Hostnamen benötigt.

3.2.3 Installation

Während der Realisierung wurde nach Erfolgserlebnissen jeweils ein aktueller Snapshot der SD-Karte mit dem Programm Win32DiskImager erstellt. Dadurch war es möglich, ein rasches Vorantreiben der Installation zu gewährleisten. Falls zuviele Änderungen am System vorgenommen wurden, welche nicht mehr rasch rückgängig gemacht werden konnten und Probleme verursachten, wurde als Wiederherstellung eines funktionierenden Systems der letzte Snapshot wieder eingespielt.

Zudem wurde für die Installation des Clusters und dessen Komponenten die Installationsanleitung (Install guide with Warewulf + Slurm) von OpenHPC verwendet. Dabei wurde der Warewulf Part zu einem grossen Teil übersprungen. Dieser hätte es ermöglicht, einen vereinfachten Netzwerkboot der Compute Nodes einzurichten. Es ist aber leider nicht möglich, die RPI's damit zu managen, da Warewulf iPXE benutzt und dies nicht kompatibel mit den RPI's ist.

4 Schlussbetrachtung

Der Cluster kann parallele Berechnungen durchführen. Leider ist es nicht möglich, Kryptowährungen mit nur einem Prozess über den Cluster hinweg zu schürfen. Jedoch ergibt es hierbei keinen Unterschied, ob jeweils neue Prozesse pro Node gestartet werden oder ob diese in einen Prozess zusammengeführt sind. Dies würde erst eine Wichtigkeit erlangen, wenn es darum ginge, Währungen zu schürfen, welche in den Blockchains diverse Bountys bereithalten, welche Belohnungen für das Schürfen beinhalten.

4.1 Arbeiten nach dem Projekt

Es gibt noch kleinere Bugs, welche behoben werden müssen. Diese verhindern aber die Inbetriebnahme des Clusters nicht. Zudem wird nach einer Virtualisierungssoftware gesucht, welche es ermöglicht, verschiedene Instanzen für andere Anwendungsgebiete in Betrieb zu nehmen.

4.2 Persönliche Betrachtung

Generell ist es mir gelungen, eine grössere Anzahl von verschiedenen Komponenten zu einem einheitlichen Produkt zu verbinden. Dadurch habe ich nun private CPU Ressourcen, welche abgekoppelt von meinem PC sind. Das Produkt kann ich für meine nächsten persönlichen Vorhaben weiterhin benutzen und muss mir keine Webserver mieten.

4.3 Danksagung

An dieser Stelle möchte ich mich speziell bei den unten aufgeführten Personen für die Unterstützung meiner Diplomarbeit bedanken:

Monika Amrein

Vielen Dank für die Überprüfung der Satzstellungen und das Korrigieren der Schreibfehler.

Stefan Rätz

Vielen Dank für die Beratung der Stromversorgung der Compute Nodes.

5 Authentizität

Mit meiner Unterschrift bestätige ich, die vorliegende Diplomarbeit selbstständig, ohne Hilfe Dritter und nur unter Benutzung der angegebenen Quellen ohne Copyright-Verletzung, erstellt zu haben.

Schüpfen, 27.05.2018

Christoph Amrein

A Anhang

Die folgenden Kapitel gehören zum Anhang und ergänzen den Diplombericht.

B Vorbereitungen RPI's

B.1 Betriebssystem installieren

Für die Installation des CentOS 7.4 ist kein boot fähiger Kernel vorhanden. Das RPI kann aber dennoch mit einem Centos 7.4 betrieben werden. Dafür sind die folgenden Schritte vorzunehmen. Die Installation des Betriebssystems ist von einem Fedora Linux Client aus beschrieben.

1. Gentoo 64 Bit Image herunterladen aus dem Github Repository⁷ von Sakaki
2. Das Archiv wird mit dem Fedora Media Writer auf die SD Karte geschrieben. Dies ist anhand der Screenshots beschrieben.



Abbildung 5: FMW: Fedora Media Writer starten

⁷<https://github.com/sakaki-/gentoo-on-rpi3-64bit>

B Vorbereitungen RPI's

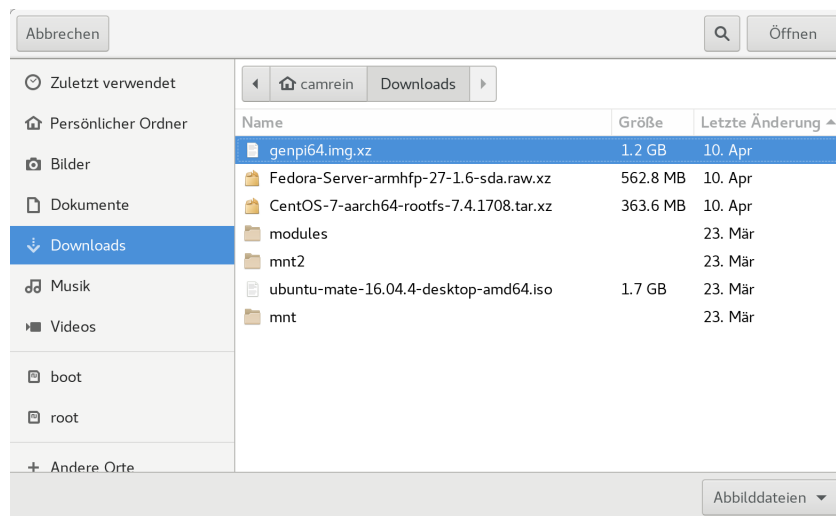


Abbildung 6: FMW: Archiv auswählen

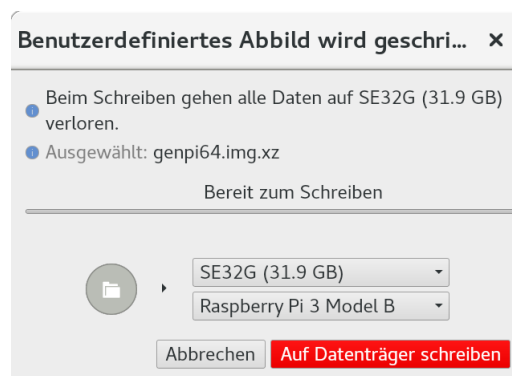


Abbildung 7: FMW: Abbild schreiben

3. Durch das Schreiben des Archivs wurden zwei Partitionen (boot & rootfs) auf der SD Karte erstellt. Diese werden wie folgt ausgelesen:

```
1 [camrein@wifibridge ~]\$ lsblk
2 NAME MAJ:MIN RM SIZE RO TYPE MOUNTPOINT
3 sda 8:0 0 238.5G 0 disk
4  └─sda1 8:1 0 200M 0 part /boot/efi
5  └─sda2 8:2 0 1G 0 part /boot
6  └─sda3 8:3 0 237.3G 0 part
7     └─fedora─root 253:0 0 50G 0 lvm /
8     └─fedora─swap 253:1 0 7.8G 0 lvm [SWAP]
9     └─fedora─home 253:2 0 179.5G 0 lvm /home
10 mmcblk0 179:0 0 29.7G 0 disk}
11  └─mmcblk0p1 179:1 0 43.1M 0 part /run/media/camrein/boot
12  └─mmcblk0p2 179:2 0 29.7G 0 part /run/media/camrein/rootfs
13 [camrein@wifibridge ~]\$
```

B Vorbereitungen RPI's

4. Die Dateisystem-Partition muss mit der von CentOS überschrieben werden. Dazu wird die Partition auf dem Linux Client angehängt. Bei Schritt 5-7 wird überprüft, ob die Partition wirklich leer ist.

```
1 [root@wifibridge Downloads]# mkdir mnt
2 [root@wifibridge Downloads]# mount /dev/mmcbk0p2 mnt
3 [root@wifibridge Downloads]# cd mnt
4 [root@wifibridge mnt]# rm -rf *
5 [root@wifibridge mnt]# ls -lrtha
6 drwxr-xr-x. 5 camrein camrein 20K 15. Mai 17:39 ..
7 drwxr-xr-x. 2 root root 4.0K 16. Mai 17:58 .
```

Die Dateisystem-Partition ist nun leer und kann mit der von CentOS überschrieben werden.

5. Das Dateisystem aus dem offiziellen Centos Repository⁸ beziehen.

6. Die geleerte Partition wird nun mit dem Dateisystem von Centos7.4 überschrieben. Zugleich soll bei Schritt 2 überprüft werden, ob die Daten wirklich auf die Partition geschrieben wurden.

```
1 [root@wifibridge mnt]# tar --numeric-owner -xpf ../CentOS-7-aarch64-rootfs-7.4.1708.tar.xz
2 [root@wifibridge mnt]# ls -lrtha
3 insgesamt 84K
4 drwxr-xr-x. 2 root root 4.0K 23. Nov 2016 srv
5 drwxr-xr-x. 2 root root 4.0K 23. Nov 2016 opt
6 drwxr-xr-x. 2 root root 4.0K 23. Nov 2016 mnt
7 drwxr-xr-x. 2 root root 4.0K 23. Nov 2016 media
8 drwxr-xr-x. 2 root root 4.0K 23. Nov 2016 home
9 drwxr-xr-x. 2 root root 4.0K 12. Sep 2017 dev
10 drwxr-xr-x. 2 root root 4.0K 12. Sep 2017 proc
11 drwxr-xr-x. 2 root root 4.0K 12. Sep 2017 run
12 drwxr-xr-x. 2 root root 4.0K 12. Sep 2017 sys
13 lrwxrwxrwx. 1 root root 7 12. Sep 2017 bin -> usr/bin
14 lrwxrwxrwx. 1 root root 8 12. Sep 2017/sbin -> usr/sbin
15 lrwxrwxrwx. 1 root root 9 12. Sep 2017 lib64 -> usr/lib64
16 lrwxrwxrwx. 1 root root 7 12. Sep 2017 lib -> usr/lib
17 drwxr-xr-x. 13 root root 4.0K 12. Sep 2017 usr
18 drwxr-xr-x. 19 root root 4.0K 12. Sep 2017 var
19 dr-xr-xr-x. 17 root root 4.0K 12. Sep 2017 .
20 drwxr-xr-x. 82 root root 4.0K 12. Sep 2017 etc
21 dr-xr-xr-x. 3 root root 4.0K 12. Sep 2017 boot
22 drwxrwxrwt. 7 root root 4.0K 12. Sep 2017 tmp
23 dr-xr-x---. 2 root root 4.0K 12. Sep 2017 root
24 drwxr-xr-x. 5 camrein camrein 20K 15. Mai 17:39 ..
25 [root@wifibridge mnt]#
```

Die SD Karte kann nun mit den RPI's verwendet werden. Diese starten jeweils mit dem Hostnamen centos.

⁸<http://mirror.centos.org/altarch/7.4.1708/isos/aarch64/CentOS-7-aarch64-rootfs-7.4.1708.tar.xz>

B.2 RPI für den Netzwerkboot vorbereiten

Für das Vorbereiten der Clients für den Netzwerkboot wurde der Guide NETWORK BOOT YOUR RASPBERRY PI von raspberrypi.org⁹ verwendet. Die RPI's werden wie folgt vorbereitet:

1. Die config.txt Datei im /boot Verzeichnis benötigt einen OTP Eintrag, dieser sagt aus, dass das RPI ohne SD Karte nach einem Betriebssystem anfragen soll.

```
1 echo program_usb_boot_mode=1 | sudo tee -a /boot/config.txt
```

2. RPI neu starten.
3. Prüfen, ob die Änderung aktiv ist.

```
1 vcgencmd otp_dump | grep 17:
```

Erwartetes Ergebnis:

```
1 17:3020000a
```

4. Den Eintrag in der /boot/config.txt wieder entfernen.

MAC-Adressen auslesen

Um Zeit zu sparen können alle MAC-Adressen der RPI's während der Vorbereitung der Clients auf den Netzwerkboot ausgelesen werden.

5. nmap Scan auf die IP Range 192.168.1.0-255 von einem Linux Client aus durchführen.

```
1 nmap -sP 192.168.1.0/24
```

Erwartetes Ergebnis für ein RPI:

```
1 Nmap scan report for centos.home (192.168.1.11)
2 Host is up (0.00055s latency).
3 MAC Address: B8:27:EB:32:39:A7 (Raspberry Pi Foundation)
```

Die Vorbereitung der RPI(Compute Nodes) ist somit abgeschlossen und es kann mit der Installation des Management Node fortgefahren werden.

B.3 Hostname und IP der MAC Adresse zuweisen

über <http://internetbox/> wird die IP Adresse sowie der Hostname der MAC Adresse der RPI's zugewiesen. Alle jemals angeschlossenen Geräte werden unter der Geräteliste aufgelistet. Dort können ebenfalls die Hostnamen den Geräten zugewiesen werden.

⁹https://www.raspberrypi.org/documentation/hardware/raspberrypi/bootmodes/net_tutorial.md

Abbildung 8: Hostnamen editieren

Abbildung 9: Übersicht der Hostnamen Zuweisung

Die fixe IP Adresse wird unter den Netzwerkeinstellungen des Routers direkt vorgenommen. Hierbei kann den bereits erkannten und definierten RPI's per Hostname eine IP zugewiesen werden.

Statische IP-Adresse automatisch zuweisen ⓘ

Gerätename	Statische IP-Adresse	MAC-Adresse	Aktionen
nebula	192.168.1.10	B8:27:EB:32:A9:1C	
c1	192.168.1.11	B8:27:EB:32:39:A7	
c2	192.168.1.12	B8:27:EB:2E:A3:D1	

Abbildung 10: Statische IP vergeben

B.4 Netzwerkshare einrichten

Der Netzwerkshare wird über das Synology NAS mit dem NFS Protokoll bereitgestellt. Der Share wird gemäss den aufgelisteten Schritten erstellt.

1. Über den Browser auf das NAS via <http://influbox/> verbinden.
2. Anmeldedaten eingeben.
3. Über die Systemsteuerung die Einstellungen für Dateidienste aufrufen.
4. Unter dem Reiter SMB/AFP/NFS soll die Funktion “NFS aktivieren“aktiviert werden.
5. Der NFS Dienst ist nun aktiviert und es kann unter “Gemeinsam Ordner“ein neues Verzeichnis angelegt werden. Dieses Verzeichnis wird der Mountpoint für die Nodes. Der Mountpoint wurde nebula genannt.
6. Für den erstellten Ordner müssen nun noch die Berechtigungen für Zugriffe eingerichtet werden. Dazu bearbeitet man unter Einstellungen die Rechte auf das Verzeichnis.
7. Damit nicht jeder das Verzeichnis mounten kann, wurde auf 192.168.10/24 eingegrenzt. Dies kann über das GUI von Synology direkt eingerichtet werden.

Freigegebenen Ordner nebula bearbeiten					
Allgemein	Verschlüsselung	Berechtigungen	Erweiterte Berechtigungen	NFS-Berechtigungen	
<div>Erstellen</div> <div>Bearbeiten</div> <div>Löschen</div>					
Client	Privileg	Squash	Asynchron	Nicht-privilegier...	Über Kreuz anh...
≡ 192.168.1.10/...	Lesen/Schreiben	Keine Zuordnung	Ja	Abgelehnt	Abgelehnt

Abbildung 11: NFS: Erlaubte Clients

8. Der Netzwerkshare kann von nun an auf dem Management Node und den Computes eingebunden werden.

C Installation Management Node

Die Serverinstallation des Management Nodes wird nach dem offiziellen OpenHPC Guide mit einigen Abweichungen durchgeführt. Die Installation basiert auf der Anleitung CentOS 7.4 aarch64 Install guide with Warewulf + Slurm ¹⁰. Die Installation wird hauptsächlich mit dem ROOT-Benutzer durchgeführt, falls nichts anderes erwähnt wird.

C.1 Variablen Definition

Durch die Installation hinweg werden die folgenden Variablen verwendet:

Nr.	Variable	Beschreibung	Werte
1	sms_name	Hostname des Managementhosts	nebula
2	sms_ip	IP Adresse des Managementhosts	192.168.1.10
3	sms_eth_internal	Ethernet Interface	eth0
4	ntp_server[0] ntp_server[1] ntp_server[2] ntp_server[3]	Zeitserver (Array)	server 0.ch.pool.ntp.org server 1.ch.pool.ntp.org server 2.ch.pool.ntp.org server 3.ch.pool.ntp.org
5	num_compute	Anzahl Compute Nodes	45
6	c_ip[0] c_ip[1] c_ip[2] c_ip[3] c_ip[4] c_ip[5] c_ip[6] c_ip[7] c_ip[8] c_ip[9] c_ip[10] c_ip[11] c_ip[12] c_ip[13] c_ip[14] c_ip[15] c_ip[16] c_ip[17] c_ip[18]	IP Adressen der Compute Nodes (Array)	192.168.1.11 192.168.1.12 192.168.1.13 192.168.1.14 192.168.1.15 192.168.1.16 192.168.1.17 192.168.1.18 192.168.1.19 192.168.1.20 192.168.1.21 192.168.1.22 192.168.1.23 192.168.1.24 192.168.1.25 192.168.1.26 192.168.1.27 192.168.1.28 192.168.1.29

¹⁰<http://openhpc.community/downloads/>

Nr.	Variable	Beschreibung	Werte
6	c_ip[19]	IP Adressen der Compute Nodes (Array)	192.168.1.30
	c_ip[20]		192.168.1.31
	c_ip[21]		192.168.1.32
	c_ip[22]		192.168.1.33
	c_ip[23]		192.168.1.34
	c_ip[24]		192.168.1.35
	c_ip[25]		192.168.1.36
	c_ip[26]		192.168.1.37
	c_ip[27]		192.168.1.38
	c_ip[28]		192.168.1.39
	c_ip[29]		192.168.1.40
	c_ip[30]		192.168.1.41
	c_ip[31]		192.168.1.42
	c_ip[32]		192.168.1.43
	c_ip[33]		192.168.1.44
	c_ip[34]		192.168.1.45
	c_ip[35]		192.168.1.46
	c_ip[36]		192.168.1.47
	c_ip[37]		192.168.1.48
	c_ip[38]		192.168.1.49
	c_ip[39]		192.168.1.50
	c_ip[40]		192.168.1.51
	c_ip[41]		192.168.1.52
	c_ip[42]		192.168.1.53
	c_ip[43]		192.168.1.54
	c_ip[44]		192.168.1.55
7	c_name[0]	Hostnamen der Compute Nodes (Array)	c1
	c_name[1]		c2
	c_name[2]		c3
	c_name[3]		c4
	c_name[4]		c5
	c_name[5]		c6
	c_name[6]		c7
	c_name[7]		c8
	c_name[8]		c9
	c_name[9]		c10
	c_name[10]		c11
	c_name[11]		c12
	c_name[12]		c13

Nr.	Variable	Beschreibung	Werte
7	c_name[13]	Hostnamen der Compute Nodes (Array)	c14
	c_name[14]		c15
	c_name[15]		c16
	c_name[16]		c17
	c_name[17]		c18
	c_name[18]		c19
	c_name[19]		c20
	c_name[20]		c21
	c_name[21]		c22
	c_name[22]		c23
	c_name[23]		c24
	c_name[24]		c25
	c_name[25]		c26
	c_name[26]		c27
	c_name[27]		c28
	c_name[28]		c29
	c_name[29]		c30
	c_name[30]		c31
	c_name[31]		c32
	c_name[32]		c33
	c_name[33]		c34
	c_name[34]		c35
	c_name[35]		c36
	c_name[36]		c37
	c_name[37]		c38
	c_name[38]		c39
	c_name[39]		c40
	c_name[40]		c41
	c_name[41]		c42
	c_name[42]		c43
	c_name[43]		c44
	c_name[44]		c45
8	c_mac[0]	MAC Adressen der Compute Nodes (Array)	B8:27:EB:32:39:A7
	c_mac[1]		B8:27:EB:2E:A3:D1
	c_mac[2]		B8:27:EB:50:45:3F
	c_mac[3]		B8:27:EB:0D:E6:25
	c_mac[4]		B8:27:EB:3E:96:B5
	c_mac[5]		B8:27:EB:EE:77:DA
	c_mac[6]		B8:27:EB:21:63:E6

Nr.	Variable	Beschreibung	Werte
8	c_mac[7]	MAC Adressen der Compute Nodes (Array)	B8:27:EB:2E:2E:CC
	c_mac[8]		B8:27:EB:17:32:96
	c_mac[9]		B8:27:EB:B2:1C:A9
	c_mac[10]		B8:27:EB:AF:63:1F
	c_mac[11]		B8:27:EB:43:00:2C
	c_mac[12]		B8:27:EB:13:7B:18
	c_mac[13]		B8:27:EB:43:CD:29
	c_mac[14]		B8:27:EB:FF:C7:56
	c_mac[15]		B8:27:EB:CE:98:66
	c_mac[16]		B8:27:EB:5D:63:34
	c_mac[17]		B8:27:EB:91:3E:0F
	c_mac[18]		B8:27:EB:F4:65:EC
	c_mac[19]		B8:27:EB:3E:AB:DC
	c_mac[20]		B8:27:EB:66:60:F6
	c_mac[21]		B8:27:EB:37:3F:74
	c_mac[22]		B8:27:EB:18:5E:F0
	c_mac[23]		B8:27:EB:B0:23:B8
	c_mac[24]		B8:27:EB:BE:C4:94
	c_mac[25]		B8:27:EB:FB:FF:57
	c_mac[26]		B8:27:EB:4E:EC:CE
	c_mac[27]		B8:27:EB:43:1C:35
	c_mac[28]		B8:27:EB:DC:74:5F
	c_mac[29]		B8:27:EB:D1:DE:2F
	c_mac[30]		B8:27:EB:5E:90:34
	c_mac[31]		B8:27:EB:DE:80:24
	c_mac[32]		B8:27:EB:A4:79:6F
	c_mac[33]		B8:27:EB:0A:4D:C7
	c_mac[34]		B8:27:EB:5C:53:5F
	c_mac[35]		B8:27:EB:F7:AF:C2
	c_mac[36]		B8:27:EB:CE:BA:ED
	c_mac[37]		B8:27:EB:59:38:3C
	c_mac[38]		B8:27:EB:99:BB:8E
	c_mac[39]		B8:27:EB:8F:7A:0D
	c_mac[40]		B8:27:EB:DE:C9:69
	c_mac[41]		B8:27:EB:7E:6F:48
	c_mac[42]		B8:27:EB:5D:DD:FE
	c_mac[43]		B8:27:EB:A6:6D:4D
	c_mac[44]		B8:27:EB:0C:63:10

Tabelle 36: Variablen Definition

C.2 Basiskonfiguration

Dem Management Node muss ein eindeutiger Hostname zugewiesen werden. Hierbei werden die untenstehenden Kommandos abgesetzt:

```
1 echo ${sms_ip} ${sms_name} >> /etc/hosts
2 hostnamectl set-hostname ${sms_name}
```

Für die folgenden Arbeiten müssen die Firewall und SELinux deaktiviert werden. DHCP wird aufgrund der Verwendung von dnsmasq deaktiviert.

```
1 systemctl disable firewalld
2 systemctl stop firewalld
3 systemctl disable dhcpd
4 systemctl stop dhcpd
5 echo 0 > /selinux/enforce
```

C.3 OpenHPC Komponenten installieren

Es wird das OpenHPC Repository für die Installation der OpenHPC Komponenten benötigt. Dieses muss installiert werden:

```
1 yum install http://build.openhpc.community/OpenHPC:/1.3/CentOS_7/aarch64/ohpc-release-1.3-1.el7.aarch64.
  rpm
```

Von nun an können die benötigten Pakete über den RPM-Paketmanager installiert werden.

Durch das neu angehängte Repository können nun die folgenden Pakete installiert werden.

```
1 yum -y install ohpc-base
2 yum -y install ohpc-warewulf
```

Weiterhin spielt die Zeit eine wichtige Rolle zwischen der Kommunikation des Managementhosts und der Compute Nodes. Dazu müssen die bereits vorhandenen Einträge der Zeitserver in der ntp.conf entfernt werden.

```
1 systemctl disable chronyd.service
2 systemctl stop chronyd.service
3 systemctl enable ntpd.service
4 sed -i '/^server/ d' /etc/ntp.conf
5 echo "server ${ntp_server[0]}" >> /etc/ntp.conf
6 echo "server ${ntp_server[1]}" >> /etc/ntp.conf
7 echo "server ${ntp_server[2]}" >> /etc/ntp.conf
```

C Installation Management Node

```
8 echo "server ${ntp_server[3]}" >> /etc/ntp.conf
9 systemctl restart ntpd
```

Nun wird der Slurmcontroller installiert. Dieser dient dazu, die Jobs auf die Compute Nodes zu verteilen. Dieser steuert die Jobverwaltung und muss installiert und eingerichtet werden. Hierbei wird die Konfiguration für 45 Computenodes vorgenommen.

```
1 yum -y install ohpc-slurm-server
2 perl -pi -e "s/ControlMachine=\S+/ControlMachine=${sms_name}/" /etc/slurm/slurm.conf
3 sed -i '/^NodeName/ d' /etc/slurm/slurm.conf
4 sed -i '/^Partition/ d' /etc/slurm/slurm.conf
5 echo "NodeName=c[1-45] CoresPerSocket=1 ThreadsPerCore=1 SocketsPerBoard=4 State=UNKNOWN" >> /etc/
  slurm/slurm.conf
6 echo "PartitionName=normal Nodes=c[1-45] Default=YES MaxTime=INFINITE State=UP" >> /etc/slurm/slurm.
  conf
```

C.4 Netzwerkboot einrichten

Das Dateisystem und die Boot-Partition werden in zwei verschiedenen Verzeichnissen erstellt und abgelegt.

Definieren des Dateisystem-Verzeichnisses und das Centos7.4 Basis-Dateisystem erstellen.

```
1 export CHROOT=/opt/ohpc/admin/images/centos7.4
2 wwmkchroot centos-7 $CHROOT
```

Anstelle von DHCPD muss der networking Service verwendet werden. Deshalb wird dieser aktiviert und gestartet.

```
1 systemctl enable networking
2 systemctl start networking
```

Nun wird das Paket dnsmasq installiert. Dadurch wird die Verteilung des Betriebssystems an eine gewisse IP Range und an ein Subnetz gewährleistet. Geräte ausserhalb der Range und des Subnetzes sollen das Betriebssystem nicht erhalten dürfen.

```
1 yum -y install dnsmasq
```

Nach der Installation kann die Konfiguration von dnsmasq vorgenommen werden. Dazu werden die Datei /etc/dnsmasq.conf angepasst und die Datei /etc/dnsmasq.d/host.allow erstellt.

In der dnsmasq.conf Datei wird das Konfigurationsverzeichnis für die erlaubten Hosts erstellt, zudem wird eine IP Range angegeben, alle Geräte innerhalb dieser Range dürfen den Kernel des Betriebssystems aus dem Verzeichnis /tftboot erhalten.

```
1 [root@nebula etc]# cat /etc/dnsmasq.conf
2 conf-dir=/etc/dnsmasq.d
3 port=0
4 dhcp-range=192.168.1.10,192.168.1.50,12h
5 log-dhcp
6 enable-tftp
7 tftp-root=/tftpboot
8 pxe-service=0,"Raspberry Pi Boot"
```

Falls sich andere Geräte in dieser IP Range befinden, wird dies durch die `/etc/dnsmasq.d/host.allow` Datei abgefangen. Dort sind alle Compute Nodes mit MAC Adresse, Hostnamen und IP Adresse hinterlegt.

Die `host.allow` Datei kann in einer for Schleife mit den definierten Variablen einfach abgefüllt werden.

```
1 for ((i=0; i<45; i++)); do echo "dhcp-host=${c_mac[$i]};${c_name[$i]};${c_ip[$i]}" >> /etc/dnsmasq.d/host.
    allow; done
```

Das Ergebnis muss wie folgt aussehen:

```
1 [root@nebula etc]# cat /etc/dnsmasq.d/host.allow
2 dhcp-host=B8:27:EB:32:39:A7,c1,192.168.1.11
3 dhcp-host=B8:27:EB:2E:A3:D1,c2,192.168.1.12
4 dhcp-host=B8:27:EB:50:45:3F,c3,192.168.1.13
5 dhcp-host=B8:27:EB:0D:E6:25,c4,192.168.1.14
6 dhcp-host=B8:27:EB:3E:96:B5,c5,192.168.1.15
7 dhcp-host=B8:27:EB:EE:77:DA,c6,192.168.1.16
8 dhcp-host=B8:27:EB:21:63:E6,c7,192.168.1.17
9 dhcp-host=B8:27:EB:2E:2E:CC,c8,192.168.1.18
10 dhcp-host=B8:27:EB:17:32:96,c9,192.168.1.19
11 dhcp-host=B8:27:EB:B2:1C:A9,c10,192.168.1.20
12 dhcp-host=B8:27:EB:AF:63:1F,c11,192.168.1.21
13 dhcp-host=B8:27:EB:43:00:2C,c12,192.168.1.22
14 dhcp-host=B8:27:EB:13:7B:18,c13,192.168.1.23
15 dhcp-host=B8:27:EB:43:CD:29,c14,192.168.1.24
16 dhcp-host=B8:27:EB:FF:C7:56,c15,192.168.1.25
17 dhcp-host=B8:27:EB:CE:98:66,c16,192.168.1.26
18 dhcp-host=B8:27:EB:5D:63:34,c17,192.168.1.27
19 dhcp-host=B8:27:EB:91:3E:0F,c18,192.168.1.28
20 dhcp-host=B8:27:EB:F4:65:EC,c19,192.168.1.29
21 dhcp-host=B8:27:EB:3E:AB:DC,c20,192.168.1.30
22 dhcp-host=B8:27:EB:66:60:F6,c21,192.168.1.31
23 dhcp-host=B8:27:EB:37:3F:74,c22,192.168.1.32
24 dhcp-host=B8:27:EB:18:5E:F0,c23,192.168.1.33
25 dhcp-host=B8:27:EB:B0:23:B8,c24,192.168.1.34
26 dhcp-host=B8:27:EB:BE:C4:94,c25,192.168.1.35
27 dhcp-host=B8:27:EB:FB:FF:57,c26,192.168.1.36
28 dhcp-host=B8:27:EB:4E:EC:CE,c27,192.168.1.37
```


C Installation Management Node

```
29 dhcp-host=B8:27:EB:43:1C:35,c28,192.168.1.38
30 dhcp-host=B8:27:EB:DC:74:5F,c29,192.168.1.39
31 dhcp-host=B8:27:EB:D1:DE:2F,c30,192.168.1.40
32 dhcp-host=B8:27:EB:5E:90:34,c31,192.168.1.41
33 dhcp-host=B8:27:EB:DE:80:24,c32,192.168.1.42
34 dhcp-host=B8:27:EB:A4:79:6F,c33,192.168.1.43
35 dhcp-host=B8:27:EB:0A:4D:C7,c34,192.168.1.44
36 dhcp-host=B8:27:EB:5C:53:5F,c35,192.168.1.45
37 dhcp-host=B8:27:EB:F7:AF:C2,c36,192.168.1.46
38 dhcp-host=B8:27:EB:CE:BA:ED,c37,192.168.1.47
39 dhcp-host=B8:27:EB:59:38:3C,c38,192.168.1.48
40 dhcp-host=B8:27:EB:99:BB:8E,c39,192.168.1.49
41 dhcp-host=B8:27:EB:8F:7A:0D,c40,192.168.1.50
42 dhcp-host=B8:27:EB:DE:C9:69,c41,192.168.1.51
43 dhcp-host=B8:27:EB:7E:6F:48,c42,192.168.1.52
44 dhcp-host=B8:27:EB:5D:DD:FE,c43,192.168.1.53
45 dhcp-host=B8:27:EB:A6:6D:4D,c44,192.168.1.54
46 dhcp-host=B8:27:EB:0C:63:10,c45,192.168.1.55
```

Boot-Verzeichnis erstellen. Aus diesem Verzeichnis wird der Kernel den Compute Nodes angeboten und übermittelt. Dabei wird das /boot Verzeichnis des Management- Nodes in dieses Verzeichnis kopiert, da es sich um das selbe Betriebssystem handelt.

```
1 [root@nebula /]# mkdir /tftboot
2 [root@nebula /]# chmod 777 /tftboot
3 [root@nebula /]# cp -r /boot /tftboot
```

Dabei muss darauf geachtet werden, dass die cmdline.txt Datei im /tftboot Verzeichnis folgenden Eintrag erhält. Dieser sagt aus, dass der Compute Node sein /root Verzeichnis vom Management Node unter dem Pfad /opt/ohpc/admin/images/centos7.4 beziehen soll.

```
1 [root@nebula etc]# cat /tftboot/cmdline.txt
2 dwc_otg.lpm_enable=0 root=/dev/nfs nfsroot=192.168.1.10:/opt/ohpc/admin/images/centos7.4,vers=3 rw ip=dhcp
  rootwait elevator=deadline
```

Zudem wird die config.txt angepasst. Hierbei wird die Taktrate der CPU auf 1400 MHz hochgetaktet. Dadurch steigt die Mining Performance von einer Hashrate 1.6 H/s auf 1.9 H/s.

```
1 [root@nebula /]# echo "arm_freq=1400" > /tftboot/config.txt
```

Den dnsmasq Service aktivieren und starten.

```
1 [root@nebula /]# systemctl enable dnsmasq.service
2 [root@nebula /]# systemctl restart dnsmasq.service
```

OpenHPC Pakete für die Compute Nodes installieren.

```
1 [root@nebula /]# yum -y --installroot=$CHROOT install ohpc-base-compute
```

C Installation Management Node

Damit die Compute Nodes über den Hostnamen angesprochen werden können, wird eine DNS Konfiguration benötigt. Die Auflösung findet über folgende Einstellung statt. Dabei wird vom Management Node die resolv.conf Datei übernommen und auf das Dateisystem der Compute Nodes kopiert:

```
1 [root@nebula /]# cp -p /etc/resolv.conf $CHROOT/etc/resolv.conf
```

Die zusätzlichen Pakete werden für die Kommunikation mit dem Slurmcontroller und der Zeitsynchronisierung benötigt.

```
1 [root@nebula /]# yum -y --installroot=$CHROOT install ohpc-slurm-client
2 [root@nebula /]# yum -y --installroot=$CHROOT install ntp
```

Beim Booten der Compute Nodes muss das Dateisystem gemountet werden. Dafür werden fstab Einträge erstellt. Ausserdem wird auf der letzten Zeile der gemeinsame Netzwerkshare mit NFS angehängt.

```
1 [root@nebula /]# echo "${sms_ip}:/home /home nfs nfsvers=3,nodev,nosuid,noatime 0 0" >> $CHROOT/etc/fstab
2 [root@nebula /]# echo "${sms_ip}:/opt/ohpc/pub /opt/ohpc/pub nfs nfsvers=3,nodev,noatime 0 0" >> $CHROOT/
  etc/fstab
3 [root@nebula /]# echo "192.168.1.129:/volume1/nebula /media/nebula_data nfs" >> $CHROOT/etc/fstab
```

Zudem müssen gewisse Verzeichnisse vom Management Node aus exportiert werden.

```
1 [root@nebula /]# echo "/home *(rw,no_subtree_check,fsid=10,no_root_squash)" >> /etc/exports
2 [root@nebula /]# echo "echo "/opt/ohpc/pub *(ro,no_subtree_check,fsid=11)" >> /etc/exports
3 [root@nebula /]# echo "/opt/ohpc/admin/images/centos7.4 *(rw,sync,no_subtree_check,no_root_squash)" /etc/
  exports
4 [root@nebula /]# exportfs -a
5 [root@nebula /]# systemctl restart nfs-server
6 [root@nebula /]# systemctl enable nfs-server
```

Damit die Zeitsynchronisierung über den Management Node läuft, wird folgendes umgesetzt:

```
1 [root@nebula /]# chroot $CHROOT systemctl enable ntpd
2 [root@nebula /]# echo "server ${sms_ip}" >> $CHROOT/etc/ntp.conf
```

Die Compute Nodes können nun mit Strom versorgt werden. Dabei wird auf dem Management Node auf die /var/log/messages Logdatei geachtet. Dort sind alle Einträge des Netzwerkboots zusammengeschlossen. Dabei müssen folgende Einträge erscheinen:

```
1 May 17 20:31:34 nebula dnsmasq-dhcp[382]: 653460281 next server: 192.168.1.10
2 May 17 20:31:34 nebula dnsmasq-dhcp[382]: 653460281 sent size: 1 option: 53 message-type 2
3 May 17 20:31:34 nebula dnsmasq-dhcp[382]: 653460281 sent size: 4 option: 54 server-identifier 192.168.1.10
4 May 17 20:31:34 nebula dnsmasq-dhcp[382]: 653460281 sent size: 4 option: 51 lease-time 12h
5 May 17 20:31:34 nebula dnsmasq-dhcp[382]: 653460281 sent size: 4 option: 58 T1 6h
6 May 17 20:31:34 nebula dnsmasq-dhcp[382]: 653460281 sent size: 4 option: 59 T2 10h30m
7 May 17 20:31:34 nebula dnsmasq-dhcp[382]: 653460281 sent size: 4 option: 1 netmask 255.255.255.0
8 May 17 20:31:34 nebula dnsmasq-dhcp[382]: 653460281 sent size: 4 option: 28 broadcast 192.168.1.255
9 May 17 20:31:34 nebula dnsmasq-dhcp[382]: 653460281 sent size: 9 option: 60 vendor-class 50:58:45:43:6c:69:65:6e
  :74
```


C.5 Netzwerkshare einbinden

Der Netzwerkshare wird über NFS auf dem Management Node eingebunden. Dies wird mit einem Eintrag in `/etc/fstab` erledigt. Dadurch ist bei einem Neustart der Netzwerkshare automatisch wieder verbunden.

```
1 [root@nebula /]# echo "192.168.1.129:/volume1/nebula /media/nebula_data nfs" >> /etc/fstab
```

C.6 Registration Compute Nodes

Die Basisinstallation ist soweit abgeschlossen. Nun werden die Compute Nodes registriert. Ab diesem Zeitpunkt kennt der Management Node jeden Compute Node. Die Compute Nodes werden wie folgt in den Datastore aufgenommen:

```
1 [root@nebula /]# echo "GATEWAYDEV=${eth_internal}" > /tmp/network.$$
2 [root@nebula /]# wwsh -y file import /tmp/network.$$ --name network
3 [root@nebula /]# wwsh -y file set network --path /etc/sysconfig/network --mode=0644 --uid=0
4 [root@nebula /]# for ((i=0; i<$num_computes; i++)) ; do
5 [root@nebula /]# wwsh -y node new ${c_name[i]} --ipaddr=${c_ip[i]} --hwaddr=${c_mac[i]} -D ${
    eth_internal}
6 done
```

C.7 Monitoring installieren

Das Nagios Monitoring wird wie folgt installiert. Dabei werden alle Grundüberwachungen installiert und können Out of the Box verwendet werden:

```
1 [root@nebula /]# yum -y install ohpc-nagios
2 [root@nebula /]# yum -y --installroot=$CHROOT install nagios-plugins-all-ohpc nrpe-ohpc
3 [root@nebula /]# chroot $CHROOT systemctl enable nrpe
4 [root@nebula /]# perl -pi -e "s/^allowed_hosts=/# allowed_hosts="/" $CHROOT/etc/nagios/nrpe.cfg
5 [root@nebula /]# echo "nrpe 5666/tcp # NRPE" >> $CHROOT/etc/services
6 [root@nebula /]# echo "nrpe : ${sms_ip} : ALLOW" >> $CHROOT/etc/hosts.allow
7 [root@nebula /]# echo "nrpe : ALL : DENY" >> $CHROOT/etc/hosts.allow
8 [root@nebula /]# chroot $CHROOT /usr/sbin/useradd -c "NRPE user for the NRPE service" -d /var/run/nrpe -r
   -g nrpe -s /sbin/nologin nrpe
9 [root@nebula /]# chroot $CHROOT /usr/sbin/groupadd -r nrpe
10 [root@nebula /]# mv /etc/nagios/conf.d/services.cfg.example /etc/nagios/conf.d/services.cfg
11 [root@nebula /]# mv /etc/nagios/conf.d/hosts.cfg.example /etc/nagios/conf.d/hosts.cfg
12 [root@nebula /]# for ((i=0; i<$num_computes; i++)) ; do
13 perl -pi -e "s/HOSTNAME$((i+1))/${c_name[i]}/ || s/HOST$((i+1))_IP/${c_ip[i]}/" \
14 [root@nebula /]# /etc/nagios/conf.d/hosts.cfg
15 done
16 [root@nebula /]# perl -pi -e "s/ \bin\mail/ \usr\bin\mailx/g" /etc/nagios/objects/commands.cfg
17 [root@nebula /]# perl -pi -e "s/nagios\@localhost/root\@${sms_name}/" /etc/nagios/objects/contacts.cfg
18 [root@nebula /]# echo command[check_ssh]=usr/lib64/nagios/plugins/check_ssh localhost >> $CHROOT/etc/
   nagios/nrpe.cfg
19 [root@nebula /]# htpasswd -bc /etc/nagios/passwd nagiosadmin ${Eigenes_Passwort}
20 [root@nebula /]# chkconfig nagios on
21 [root@nebula /]# systemctl start nagios
22 [root@nebula /]# chmod u+s 'which ping'
```

Nagios kann nun eingesetzt werden und ist über <http://nebula/nagios> erreichbar.

Das Performance Monitoring wird mit Ganglia realisiert, welches sich wie folgt installieren lässt:

```
1 [root@nebula /]# yum -y install ohpc-ganglia
2 [root@nebula /]# yum -y --installroot=$CHROOT install ganglia-gmond-ohpc
3 [root@nebula /]# cp /opt/ohpc/pub/examples/ganglia/gmond.conf /etc/ganglia/gmond.conf
4 [root@nebula /]# perl -pi -e "s/<sms>/${sms_name}/" /etc/ganglia/gmond.conf
5 [root@nebula /]# cp /etc/ganglia/gmond.conf $CHROOT/etc/ganglia/gmond.conf
6 echo "gridname MySite" >> /etc/ganglia/gmetad.conf
7 [root@nebula /]# systemctl enable gmond
8 [root@nebula /]# systemctl enable gmetad
9 [root@nebula /]# systemctl start gmond
10 [root@nebula /]# systemctl start gmetad
11 [root@nebula /]# chroot $CHROOT systemctl enable gmond
12 [root@nebula /]# systemctl try-restart httpd
```

Ganglia kann nun eingesetzt werden und ist über <http://nebula/ganglia> erreichbar.

C.8 Miner installieren

Es wird die cpuminer-multi Version von tkinjo verwendet. Diese ist mit dem ARMv8 Prozessor kompatibel und der Miner kann kompiliert werden.

Die folgenden Schritte sind für die Installation notwendig:

Verzeichnis erstellen, in dem der Miner installiert werden soll.

```
1 [root@nebula /]# mkdir -p /opt/miners
```

Das tkinjo cpuminer-multi Repository klonen. Dabei wird das /opt/miners/tkinjo Verzeichnis erstellt.

```
1 [root@nebula /]# cd /opt/miners
2 [root@nebula miners]# git clone https://github.com/tkinjo1985/cpuminer-multi.git tkinjo
3 Klonen nach 'tkinjo '...
4 remote: Counting objects: 3805, done.
5 remote: Total 3805 (delta 0), reused 0 (delta 0), pack-reused 3805
6 Empfangen Objekte: 100% (3805/3805), 18.98 MiB | 3.77 MiB/s, done.
7 Löse Unterschiede auf: 100% (2589/2589), done.
```

Die geklonte Version wurde noch modifiziert, so dass die jeweiligen Hostnamen beim Schürfen angegeben werden. Dazu wird die /opt/miners/tkinjo/cpu-miner.c wie folgt angepasst. Dies kann mit git apply cpu-miner.c.diff installiert werden:

```
1 cat /opt/miners/tkinjo/cpu-miner.c.diff
2 diff --git a/cpu-miner.c b/cpu-miner.c
3 index 0620d15..0d8c5e2 100644
4 --- a/cpu-miner.c
5 +++ b/cpu-miner.c
6 @@ -281,6 +281,11 @@ char *opt_api_allow = NULL;
7  int opt_api_remote = 0;
8  int opt_api_listen = 4048; /* 0 to disable */
9
10 + #ifndef MAX_HOST_LEN
11 + #define MAX_HOST_LEN 0xff
12 + char local_hostname[MAX_HOST_LEN];
13 + #endif
14 +
15 + #ifdef HAVE_GETOPT_LONG
16 + #include <getopt.h>
17 + #else
18 @@ -1011,6 +1016,7 @@ out:
19  #define YAY "yay!!!"
20  #define BOO "booooo"
21
22 +
23  static int share_result(int result, struct work *work, const char *reason)
24  {
```

```
25     const char *flag;
26 @@ -1052,14 +1058,14 @@ static int share_result(int result, struct work *work, const char *reason)
27     case ALGO_PLUCK:
28     case ALGO_SCRIPTJANE:
29         sprintf(s, hashrate >= 1e6 ? "%.0f" : "%.2f", hashrate);
30 -         applog(LOG_NOTICE, "accepted: %lu/%lu (%s), %s H/s %s",
31 -             accepted_count, accepted_count + rejected_count,
32 +         applog(LOG_NOTICE, "[" CL_LBL "%s" CL_N "] accepted: %lu/%lu (%s), %s H/s %s",
33 +             local_hostname, accepted_count, accepted_count + rejected_count,
34             suppl, s, flag);
35         break;
36     default:
37         sprintf(s, hashrate >= 1e6 ? "%.0f" : "%.2f", hashrate / 1000.0);
38 -         applog(LOG_NOTICE, "accepted: %lu/%lu (%s), %s kH/s %s",
39 -             accepted_count, accepted_count + rejected_count,
40 +         applog(LOG_NOTICE, "[" CL_LBL "%s" CL_N "] accepted: %lu/%lu (%s), %s kH/s %s",
41 +             local_hostname, accepted_count, accepted_count + rejected_count,
42             suppl, s, flag);
43         break;
44     }
45 @@ -2388,12 +2394,12 @@ static void *miner_thread(void *userdata)
46         case ALGO_CRYPTONIGHT:
47         case ALGO_PLUCK:
48         case ALGO_SCRIPTJANE:
49 -             applog(LOG_INFO, "CPU # %d: %.2f H/s", thr_id, thr_hashrates[thr_id]);
50 +             applog(LOG_INFO, "[" CL_LBL "%s" CL_N "] CPU # %d: %.2f H/s",
51                 local_hostname, thr_id, thr_hashrates[thr_id]);
52             break;
53         default:
54             sprintf(s, thr_hashrates[thr_id] >= 1e6 ? "%.0f" : "%.2f",
55                 thr_hashrates[thr_id] / 1e3);
56 -             applog(LOG_INFO, "CPU # %d: %s kH/s", thr_id, s);
57 +             applog(LOG_INFO, "[" CL_LBL "%s" CL_N "] CPU # %d: %s kH/s",
58                 local_hostname, thr_id, s);
59             break;
60     }
61     tm_rate_log = time(NULL);
62 @@ -2409,11 +2415,11 @@ static void *miner_thread(void *userdata)
63     case ALGO_AXIOM:
64     case ALGO_SCRIPTJANE:
65         sprintf(s, "%.3f", hashrate);
66 -         applog(LOG_NOTICE, "Total: %s H/s", s);
67 +         applog(LOG_NOTICE, "[" CL_LBL "%s" CL_N "] Total: %s H/s",
68             local_hostname, s);
69         break;
70     default:
71         sprintf(s, hashrate >= 1e6 ? "%.0f" : "%.2f", hashrate / 1000);
72 -         applog(LOG_NOTICE, "Total: %s kH/s", s);
73 +         applog(LOG_NOTICE, "[" CL_LBL "%s" CL_N "] Total: %s kH/s",
74             local_hostname, s);
```

C Installation Management Node

```
71         break;
72     }
73     global_hashrate = (uint64_t) hashrate;
74 @@ -3338,6 +3344,8 @@ int main(int argc, char *argv[]) {
75     long flags;
76     int i, err;
77
78 + gethostname(local_hostname, MAX_HOST_LEN*sizeof(char));
79 +
80     pthread_mutex_init(&applog_lock, NULL);
81
82     show_credits();
```

Ohne Anpassung sieht die Ausgabe wie folgt aus:

```
1 [2018-05-19 14:09:03] CPU #3: 1.86 H/s
2 [2018-05-19 14:09:03] CPU #2: 1.86 H/s
3 [2018-05-19 14:09:03] CPU #3: 1.86 H/s
4 [2018-05-19 14:09:03] CPU #1: 1.86 H/s
5 [2018-05-19 14:09:03] CPU #3: 1.86 H/s
6 [2018-05-19 14:09:03] CPU #0: 1.86 H/s
7 [2018-05-19 14:09:03] CPU #2: 1.86 H/s
8 [2018-05-19 14:09:03] CPU #2: 1.85 H/s
9 [2018-05-19 14:09:03] CPU #2: 1.86 H/s
10 [2018-05-19 14:09:03] CPU #3: 1.85 H/s
11 [2018-05-19 14:09:03] CPU #3: 1.86 H/s
12 [2018-05-19 14:09:03] CPU #0: 1.85 H/s
13 [2018-05-19 14:09:03] CPU #0: 1.85 H/s
14 [2018-05-19 14:09:03] CPU #0: 1.85 H/s
15 [2018-05-19 14:09:03] CPU #2: 1.85 H/s
16 [2018-05-19 14:09:03] CPU #1: 1.85 H/s
17 [2018-05-19 14:09:03] CPU #2: 1.85 H/s
```

Durch die Modifikation werden die Computernamen ausgegeben:

```
1 [2018-05-19 14:06:33] [c26] CPU #2: 1.86 H/s
2 [2018-05-19 14:06:33] [c14] CPU #1: 1.86 H/s
3 [2018-05-19 14:06:33] [c1] CPU #1: 1.86 H/s
4 [2018-05-19 14:06:33] [c30] CPU #1: 1.85 H/s
5 [2018-05-19 14:06:33] [c9] CPU #3: 1.86 H/s
6 [2018-05-19 14:06:33] [c21] CPU #1: 1.86 H/s
7 [2018-05-19 14:06:33] [c9] CPU #2: 1.85 H/s
8 [2018-05-19 14:06:33] [c21] CPU #2: 1.86 H/s
9 [2018-05-19 14:06:33] [c14] CPU #3: 1.85 H/s
10 [2018-05-19 14:06:33] [c1] CPU #2: 1.85 H/s
11 [2018-05-19 14:06:33] [c39] CPU #2: 1.85 H/s
12 [2018-05-19 14:06:33] [c39] CPU #0: 1.85 H/s
13 [2018-05-19 14:06:33] [c13] CPU #2: 1.84 H/s
14 [2018-05-19 14:06:33] [c13] CPU #0: 1.84 H/s
15 [2018-05-19 14:06:33] [c4] CPU #3: 1.85 H/s
```


C Installation Management Node

Die geklonte Version kann nun kompiliert werden. Dabei kann die Warnung: “Implizite Deklaration der Funktion“ ignoriert werden. Zur Übersicht wurde die Ausgabe unten gekürzt:

```
1 [root@nebula miners]# cd tkinjo/
2 [root@nebula tkinjo]# ./build.sh
3 make: *** Keine Regel, um >>clean<< zu erstellen. Schluss.
4 clean
5 configure.ac:15: installing './compile'
6 configure.ac:4: installing './config.guess'
7 configure.ac:4: installing './config.sub'
8 configure.ac:9: installing './install-sh'
9 configure.ac:9: installing './missing'
10 Makefile.am: installing './depcomp'
11 checking build system type... aarch64-unknown-linux-gnu
12 checking host system type... aarch64-unknown-linux-gnu
13 checking target system type... aarch64-unknown-linux-gnu
14 checking for a BSD-compatible install... /usr/bin/install -c
15 checking whether build environment is sane... yes
16 checking for a thread-safe mkdir -p... /usr/bin/mkdir -p
17 In file included from algo/../../scryptjane/scrypt-jane-romix.h:2:0,
18      from algo/scrypt-jane.c:24:
19 algo/../../scryptjane/scrypt-jane-chacha.h: In Funktion >>scrypt_test_mix<<:
20 algo/../../scryptjane/scrypt-jane-chacha.h:125:20: Warnung: Implizite Deklaration der Funktion >>detect_cpu<< [-Wimplicit-function-declaration]
21     size_t cpuflags = detect_cpu();
22                        ~~~~~
23 make[2]: Leaving directory '/opt/miners/tkinjo'
24 make[1]: Leaving directory '/opt/miners/tkinjo'
```



D Projektplan

Aktivität		Start	Ende	Dauer [h]	Wer	2018																	
						Februar			März				April				Mai - Juni						
							6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
Phasen		Datum	Datum	Soll	Ist	Abw.																	
Vorarbeiten				8.2	7.2	-1																	
1.0	Ablage erstellen	05.02.2018	05.02.2018	0.2	0.2	0	CA																
2.0	Projektinitialisierungsauftrag schreiben	05.02.2018	05.02.2018	3	2	-1	CA																
3.0	Kickoff-Meeting	05.02.2018	05.02.2018			0	CA																
3.1	Kickoff-Meeting Vorbereitung (Präsentation, Raum, Apero)	05.02.2018	05.02.2018	2	2	0	CA																
3.2	Kickoff-Meeting Durchführen	05.02.2018	05.02.2018	2	2	0	CA,AM,RS																
4.0	Kickoff-Dokumente nachführen, anpassen, fertigstellen	06.02.2018	08.02.2018	1	1	0	CA																
5.0	Abnahme Projektinitialisierungsauftrag durch Experten [M1]	08.02.2018	15.02.2018	0	0	0	AM,RS	M1															
Initialisierung				30	38.5	8.5	Stunden																
6.0	Projektplan erstellen	06.02.2018	15.02.2018	4	6	2	CA	Initialisierungsphase															
7.0	Projektlogo erstellen	06.02.2018	20.02.2018	2	0.5	-1.5	CA																
8.0	Studie durchführen	06.02.2018	20.02.2018				CA																
8.1	Studie: Informationsbeschaffung	06.02.2018	20.02.2018	5	10	5	CA																
8.2	Studie: verschiedene Varianten	06.02.2018	20.02.2018	5	5	0	CA																
8.3	Studie: Analyse der Varianten	06.02.2018	20.02.2018	4	6	2	CA																
8.4	Studie: Wirtschaftlichkeit	06.02.2018	20.02.2018	2	2	0	CA																
8.5	Studie: Realisierungsvorschlag	06.02.2018	20.02.2018	2	1	-1	CA																
9.0	Projektauftrag erstellen	20.02.2018	25.02.2018				CA																
9.1	Projektauftrag: Ist- und Sollzustand	20.02.2018	25.02.2018	0.5	2	1.5	CA																
9.2	Projektauftrag: Ziele und deren Messbarkeit	20.02.2018	25.02.2018	1	1.5	0.5	CA																
9.3	Projektauftrag: Projektorganisation	20.02.2018	25.02.2018	0.5	0.5	0	CA																
10.0	Diplombericht der Phase Initialisierung schreiben	25.02.2018	28.02.2018	4	4	0	CA																
Konzept				41	38.5	-2.5	Stunden																
11.0	Zwischen-Meeting	01.03.2018	01.03.2018			0	CA	Konzeptphase															
11.1	Zwischen-Meeting Vorbereitung(Präsentation, Raum, Apero)	01.03.2018	01.03.2018	2	0.5	-1.5	CA																
11.2	Zwischen-Meeting durchführen	01.03.2018	01.03.2018	2	2	0	CA,AM,RS																
12	Zwischen-Meeting Dokumente nachführen, anpassen, fertigstellen	02.03.2018	04.03.2018	2	3	1	CA																
13	Detaillkonzept erstellen	05.03.2018	17.03.2018	12	18	6	CA																
14	Testkonzept schreiben	18.03.2018	22.03.2018	11	12	1	CA																
15	Dokumenten Review der Phase Konzept	24.03.2018	26.03.2018	12	3	-9	CA																
Realisierung				171	176	5	Stunden																
16	Physischer Aufbau des Clusters	03.04.2018	07.04.2018	20	32	12	CA																
17	Stromversorgung des Clusters [M2]	08.04.2018	09.04.2018	8	7	-1	CA									M2							
18	Raspberry PI's vorbereiten	17.04.2018	17.04.2018	4	6	2	CA																
19	PXE Boot / DHCP / DNS / Netzwerk	21.04.2018	23.04.2018	25	25	0	CA																
20	Cluster Software installieren und Konfigurieren	24.04.2018	25.04.2018	25	20	-5	CA																
21	Mining Software installieren und Konfigurieren	26.04.2018	26.04.2018	12	10	-2	CA																
22	Entwickeln von Tools und Automatismen	02.04.2018	28.04.2018	30	17	-13	CA																
23	Monitoring einrichten	01.05.2018	10.05.2018	14	7	-7	CA																
24	Periodische Tests	10.04.2018	13.05.2018	7	30	23	CA																
25	Installationshandbuch erarbeiten	02.04.2018	10.05.2018	8	10	2	CA																
26	Testprotokoll gemäss Testkonzept	02.05.2018	03.05.2018	9	9	0	CA																
27	Betriebshandbuch	01.05.2018	13.05.2018	9	3	-6	CA																
Einführung / Abschluss				25.5	27	1.5	Stunden																
28	Abschlussbericht	17.05.2018	22.05.2018	9.5	15	5.5	CA																
29	Management Summary	20.05.2018	24.05.2018	8	2	-6	CA																
30	Vorbereitung Abschluss-Meeting	22.05.2018	27.05.2018	3	5	2	CA																
31	Drucken & Binden	24.05.2018	01.06.2018	2	2	0	CA																
32	Abschluss-Meeting	02.06.2018	02.06.2018	2	2	0	CA																
33	Projektabschluss [M3]	03.06.2018	03.06.2018	1	1	0	CA,AM,RS															M3	
Weitere Tasks				25	33	8	Stunden																
34	Periodische Prüfung der Dokumente auf Rechtschreibfehler	05.02.2018	03.06.2018	8	18	10	CA																
35	Logbuch / Journal / Protokolle	05.02.2018	03.06.2018	17	25	8	CA																
Reserve für unvorhergesehene Arbeiten (25h eingeplant)				1.5	2	0.5	Stunden																
36	Wiki aufsetzen	06.02.2018	08.02.2018	1	1	0	CA																
37	Projektnamen definieren	05.02.2018	03.06.2018	0.5	1	0.5	CA																

E Diplomeingabe

Projekt: Mining Cluster

Christoph Amrein TSBE 16B

Praktische Diplomarbeit 2018

Ausgangslage

Gemäss der Webseite coinmarketcap.com, einer Webseite zur Verfolgung der Kurse von Kryptowährungen, gibt es zurzeit 1349¹ Kryptowährungen und täglich werden es mehr. Aus diesem Grund habe ich anfangs Jahr selbst mit dem Erzeugen von Coins diverser Kryptowährungen begonnen. Jedoch bin ich mit meiner aktuellen Ausrüstung, einem leistungsstarken Computer mit einer guten Grafikkarte, nicht zufrieden. Die Stromkosten sind zu hoch und die Erträge zu gering. Aus diesem Grund will ich ein Projekt durchführen, welches meine Erträge auf ein neues Niveau heben soll. Dazu soll auf einem Cluster, der aus Raspberry PI's besteht, ein Miner² installiert werden, welcher alle verfügbaren Ressourcen in diesem Cluster nutzt. Ich habe vor, dieses Projekt im Rahmen meiner Diplomarbeit durchzuführen. Das Projekt soll durch den Handel mit Kryptowährungen finanziert werden.

Begründung

Es soll ein skalierbarer Cluster aus ca. 40 Raspberry PI's installiert werden, auf dem eine Mining-Software betrieben wird. Das Einsatzgebiet des Clusters soll jederzeit ohne viel Aufwand angepasst werden können. Nach der Umsetzung sollen durch den Cluster möglichst effizient Coins erzeugt werden. Dabei soll der Ertrag aus den erzeugten Coins den Stromkosten gegenübergestellt werden. Ein weiteres Ziel der Projektarbeit ist es, mit diesen auf den üblichen Handelsplattformen tätig zu sein. Aus Kostengründen habe ich als optionales Ziel vorgesehen, Grafikkarten in den Cluster einzubinden und mit diesen ebenfalls möglichst effizient Coins zu erzeugen. Der Fokus der Arbeit liegt hierbei aber hauptsächlich auf dem Cluster und dessen CPU Nutzung. Die Kosten für das Projekt sollen innerhalb von 2 Jahren amortisiert werden. Danach will ich damit über längere Zeit Geld verdienen.

Themenbereiche

Es wird ein fundamentales Wissen in **Elektrotechnik** benötigt, da die Raspberry PI's durch eine gemeinsame Stromquelle versorgt werden sollen. Damit die Kommunikation zwischen Raspberry PI's eingerichtet werden kann, wird Wissen in **Linux** und **Netzwerktechnik** benötigt. Nach der Installation des Clusters sind Kenntnisse in **Entwicklung**, **Programmieren**, **Monitoring** und **Systempflege** von Nöten.

¹ Stand 13.12.2017

² Software zum Minen der Kryptowährung

Projektziele

Operationelles Ziel

Der Cluster soll durchgehend und selbstständig funktionieren. Die Arbeiten auf dem Cluster sollen sich auf das Patching und Updaten des Betriebssystems sowie der Miner- und Cluster-Software beziehen.
Bei Problemen ausserhalb der oben genannten Aufgaben soll automatisch durch die Systemüberwachung Alarm ausgelöst und eine Nachricht versendet werden.

Abwicklungsziele

- Zeitplan einhalten
 - o Es sollen keine grossen Abweichungen zum Zeitplan entstehen.
 - o Die Meilensteine müssen eingehalten werden.
- Arbeitsjournal führen
 - o Es wird ein lückenloses und verständliches Arbeitsjournal geführt.
 - o Das Arbeitsjournal soll zeitnahe geschrieben und ergänzt werden.

Wirkungs- und Nutzenziele

- Es soll das Maximale an Ressourcen aus den Raspberry PI's herausgeholt werden.
- Der Cluster kann schnell für andere Anwendungsgebiete konfiguriert werden.
- Durch die Lösung sollen verschiedene Coins diverser Kryptowährungen erzeugt werden.
- Es soll mit den erzeugten Coins auf Handelsplattformen gehandelt werden.

Lieferobjekte

Initialisierung

- Detaillierter Projektplan
- Projektauftrag
- Dokumente zum Kick-Off-Meeting

Voranalyse

- Diplombericht
- Initiale Voranalyse
- Präsentation

Konzept

- Hostnamenkonzept der einzelnen Raspberry PI's
- Backupkonzept
- Miningkonzept
- Überwachungskonzept

Realisierung

- Dokumentation der Arbeit
- Abnahmetests, Testprotokoll
- Cluster aus Raspberry PI's
- Transaktionsauszug der erzeugten Währung (Wallet zu Wallet)

Projektabschluss

- Management Summary
- Abschlussbericht
- Präsentation der Arbeit
- Arbeitsjournal

Projektplan

Monat	Januar				Februar				März					April				Mai				Juni				
Kalenderwoche	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26
Phasen																										
Initialisierung																										
Voranalyse																										
Konzept																										
Realisierung									Ms				Ms						Ms							
Abschluss																										
Meetings			Me									Me												Me		
Dokumentation																										

Organisation

Infrastruktur

Das Projekt wird bei mir zu Hause durchgeführt.

Beteiligte Personen

Funktion	Name
Auftraggeber	Christoph Amrein
Projektleiter	Christoph Amrein
Ausführender	Christoph Amrein
Begleitender Dozent	Andreas Megert, TSBE
Begleitender Experte	Rolf Schmutz, Post CH AG

F Testfälle

F.1 Komponententests

Bezeichnung	K-001	Management Node	Hostnamen / IP / MAC
Beschreibung	Der Management Node wird auf die zugewiesene IP Adresse, MAC Adresse und den Hostnamen geprüft.		
Testvoraussetzung	Der Management Node und der Testclient befinden sich im selben Netzwerk.		
Testschritte	<ul style="list-style-type: none"> - Management Node starten (Strom anschliessen). - 30 Sekunden warten. - Auf dem Testclient über Putty oder Shell mit dem Befehl "nmap -sn 192.168.1.10" eingeben. - Prüfen, ob der die Zuweisung gemäss Hostnamenkonzept richtig ist. 		
Erwartetes Ergebnis	Die Rückgabewerte (Hostnamen, IP- und MAC-Adresse) sollen identisch mit denen im Hostnamenkonzept sein.		

Tabelle 37: Testfall K-001

Bezeichnung	K-002	Management Node	Anmelden
Beschreibung	Es wird getestet ob der Management Node über das SSH Protokoll erreichbar ist.		
Testvoraussetzung	Der Management Node und der Testclient befinden sich im selben Netzwerk.		
Testschritte	<ul style="list-style-type: none"> - Management Node starten (Strom anschliessen). - 30 Sekunden warten. - Auf dem Testclient über Putty oder Shell den Befehl "ssh root@nebula" eingeben. - Passwort eingeben. 		
Erwartetes Ergebnis	Der Zugriff auf den Management Node funktioniert		

Tabelle 38: Testfall K-002

Bezeichnung	K-003	Compute Node c1	Hostnamen / IP / MAC
Beschreibung	Der Compute Node wird auf die zugewiesene IP Adresse, MAC Adresse und den Hostnamen geprüft.		
Testvoraussetzung	Der Compute Node und der Testclient befinden sich im selben Netzwerk.		
Testschritte	<ul style="list-style-type: none"> - Compute Node starten (Strom anschliessen) - 3 Minuten warten - Auf dem Testclient über Putty oder Shell mit dem Befehl "nmap -sn 192.168.1.11" eingeben - Prüfen ob die Zuweisung gemäss Hostnamenkonzept richtig ist. 		
Erwartetes Ergebnis	Die Rückgabewerte (Hostnamen, IP- und MAC-Adresse) sollen identisch mit denen im Hostnamenkonzept sein.		

Tabelle 39: Testfall K-003

F Testfälle

Bezeichnung	K-004	Compute Node c2	Hostnamen / IP / MAC
Beschreibung	Der Compute Node wird auf die zugewiesene IP Adresse, MAC Adresse und den Hostnamen geprüft.		
Testvoraussetzung	Der Compute Node und der Testclient befinden sich im selben Netzwerk.		
Testschritte	<ul style="list-style-type: none"> - Compute Node starten (Strom anschliessen) - 3 Minuten Sekunden warten - Auf dem Testclient über Putty oder Shell mit dem Befehl "nmap -sn 192.168.1.12" eingeben - Prüfen ob die Zuweisung gemäss Hostnamenkonzept richtig ist. 		
Erwartetes Ergebnis	Die Rückgabewerte (Hostnamen, IP- und MAC-Adresse) sollen identisch mit deren im Hostnamenkonzept sein.		

Tabelle 40: Testfall K-004

Bezeichnung	K-005	Compute Node c3	Hostnamen / IP / MAC
Beschreibung	Der Compute Node wird auf die zugewiesene IP Adresse, MAC Adresse und den Hostnamen geprüft.		
Testvoraussetzung	Der Compute Node und der Testclient befinden sich im selben Netzwerk.		
Testschritte	<ul style="list-style-type: none"> - Compute Node starten (Strom anschliessen) - 3 Minuten Sekunden warten - Auf dem Testclient über Putty oder Shell mit dem Befehl "nmap -sn 192.168.1.13" eingeben - Prüfen ob die Zuweisung gemäss Hostnamenkonzept richtig ist. 		
Erwartetes Ergebnis	Die Rückgabewerte (Hostnamen, IP- und MAC-Adresse) sollen identisch mit deren im Hostnamenkonzept sein.		

Tabelle 41: Testfall K-005

Bezeichnung	K-006	Compute Node c4	Hostnamen / IP / MAC
Beschreibung	Der Compute Node wird auf die zugewiesene IP Adresse, MAC Adresse und den Hostnamen geprüft.		
Testvoraussetzung	Der Compute Node und der Testclient befinden sich im selben Netzwerk.		
Testschritte	<ul style="list-style-type: none"> - Compute Node starten (Strom anschliessen) - 3 Minuten Sekunden warten - Auf dem Testclient über Putty oder Shell mit dem Befehl "nmap -sn 192.168.1.14" eingeben - Prüfen ob die Zuweisung gemäss Hostnamenkonzept richtig ist. 		
Erwartetes Ergebnis	Die Rückgabewerte (Hostnamen, IP- und MAC-Adresse) sollen identisch mit deren im Hostnamenkonzept sein.		

Tabelle 42: Testfall K-006

F Testfälle

Bezeichnung	K-007	Compute Node c5	Hostnamen / IP / MAC
Beschreibung	Der Compute Node wird auf die zugewiesene IP Adresse, MAC Adresse und den Hostnamen geprüft.		
Testvoraussetzung	Der Compute Node und der Testclient befinden sich im selben Netzwerk.		
Testschritte	<ul style="list-style-type: none"> - Compute Node starten (Strom anschliessen) - 3 Minuten Sekunden warten - Auf dem Testclient über Putty oder Shell mit dem Befehl "nmap -sn 192.168.1.15" eingeben - Prüfen ob die Zuweisung gemäss Hostnamenkonzept richtig ist. 		
Erwartetes Ergebnis	Die Rückgabewerte (Hostnamen, IP- und MAC-Adresse) sollen identisch mit deren im Hostnamenkonzept sein.		

Tabelle 43: Testfall K-007

Bezeichnung	K-008	Compute Node c6	Hostnamen / IP / MAC
Beschreibung	Der Compute Node wird auf die zugewiesene IP Adresse, MAC Adresse und den Hostnamen geprüft.		
Testvoraussetzung	Der Compute Node und der Testclient befinden sich im selben Netzwerk.		
Testschritte	<ul style="list-style-type: none"> - Compute Node starten (Strom anschliessen) - 3 Minuten Sekunden warten - Auf dem Testclient über Putty oder Shell mit dem Befehl "nmap -sn 192.168.1.16" eingeben - Prüfen ob die Zuweisung gemäss Hostnamenkonzept richtig ist. 		
Erwartetes Ergebnis	Die Rückgabewerte (Hostnamen, IP- und MAC-Adresse) sollen identisch mit deren im Hostnamenkonzept sein.		

Tabelle 44: Testfall K-008

Bezeichnung	K-009	Compute Node c7	Hostnamen / IP / MAC
Beschreibung	Der Compute Node wird auf die zugewiesene IP Adresse, MAC Adresse und den Hostnamen geprüft.		
Testvoraussetzung	Der Compute Node und der Testclient befinden sich im selben Netzwerk.		
Testschritte	<ul style="list-style-type: none"> - Compute Node starten (Strom anschliessen) - 3 Minuten Sekunden warten - Auf dem Testclient über Putty oder Shell mit dem Befehl "nmap -sn 192.168.1.17" eingeben - Prüfen ob die Zuweisung gemäss Hostnamenkonzept richtig ist. 		
Erwartetes Ergebnis	Die Rückgabewerte (Hostnamen, IP- und MAC-Adresse) sollen identisch mit deren im Hostnamenkonzept sein.		

Tabelle 45: Testfall K-009

F Testfälle

Bezeichnung	K-010	Compute Node c8	Hostnamen / IP / MAC
Beschreibung	Der Compute Node wird auf die zugewiesene IP Adresse, MAC Adresse und den Hostnamen geprüft.		
Testvoraussetzung	Der Compute Node und der Testclient befinden sich im selben Netzwerk.		
Testschritte	<ul style="list-style-type: none"> - Compute Node starten (Strom anschliessen) - 3 Minuten Sekunden warten - Auf dem Testclient über Putty oder Shell mit dem Befehl "nmap -sn 192.168.1.18" eingeben - Prüfen ob die Zuweisung gemäss Hostnamenkonzept richtig ist. 		
Erwartetes Ergebnis	Die Rückgabewerte (Hostnamen, IP- und MAC-Adresse) sollen identisch mit deren im Hostnamenkonzept sein.		

Tabelle 46: Testfall K-010

Bezeichnung	K-011	Compute Node c9	Hostnamen / IP / MAC
Beschreibung	Der Compute Node wird auf die zugewiesene IP Adresse, MAC Adresse und den Hostnamen geprüft.		
Testvoraussetzung	Der Compute Node und der Testclient befinden sich im selben Netzwerk.		
Testschritte	<ul style="list-style-type: none"> - Compute Node starten (Strom anschliessen) - 3 Minuten Sekunden warten - Auf dem Testclient über Putty oder Shell mit dem Befehl "nmap -sn 192.168.1.19" eingeben - Prüfen ob die Zuweisung gemäss Hostnamenkonzept richtig ist. 		
Erwartetes Ergebnis	Die Rückgabewerte (Hostnamen, IP- und MAC-Adresse) sollen identisch mit deren im Hostnamenkonzept sein.		

Tabelle 47: Testfall K-011

Bezeichnung	K-012	Compute Node c10	Hostnamen / IP / MAC
Beschreibung	Der Compute Node wird auf die zugewiesene IP Adresse, MAC Adresse und den Hostnamen geprüft.		
Testvoraussetzung	Der Compute Node und der Testclient befinden sich im selben Netzwerk.		
Testschritte	<ul style="list-style-type: none"> - Compute Node starten (Strom anschliessen) - 3 Minuten Sekunden warten - Auf dem Testclient über Putty oder Shell mit dem Befehl "nmap -sn 192.168.1.20" eingeben - Prüfen ob die Zuweisung gemäss Hostnamenkonzept richtig ist. 		
Erwartetes Ergebnis	Die Rückgabewerte (Hostnamen, IP- und MAC-Adresse) sollen identisch mit deren im Hostnamenkonzept sein.		

Tabelle 48: Testfall K-012

F Testfälle

Bezeichnung	K-013	Compute Node c11	Hostnamen / IP / MAC
Beschreibung	Der Compute Node wird auf die zugewiesene IP Adresse, MAC Adresse und den Hostnamen geprüft.		
Testvoraussetzung	Der Compute Node und der Testclient befinden sich im selben Netzwerk.		
Testschritte	<ul style="list-style-type: none"> - Compute Node starten (Strom anschliessen) - 3 Minuten Sekunden warten - Auf dem Testclient über Putty oder Shell mit dem Befehl "nmap -sn 192.168.1.21" eingeben - Prüfen ob die Zuweisung gemäss Hostnamenkonzept richtig ist. 		
Erwartetes Ergebnis	Die Rückgabewerte (Hostnamen, IP- und MAC-Adresse) sollen identisch mit deren im Hostnamenkonzept sein.		

Tabelle 49: Testfall K-013

Bezeichnung	K-014	Compute Node c12	Hostnamen / IP / MAC
Beschreibung	Der Compute Node wird auf die zugewiesene IP Adresse, MAC Adresse und den Hostnamen geprüft.		
Testvoraussetzung	Der Compute Node und der Testclient befinden sich im selben Netzwerk.		
Testschritte	<ul style="list-style-type: none"> - Compute Node starten (Strom anschliessen) - 3 Minuten Sekunden warten - Auf dem Testclient über Putty oder Shell mit dem Befehl "nmap -sn 192.168.1.22" eingeben - Prüfen ob die Zuweisung gemäss Hostnamenkonzept richtig ist. 		
Erwartetes Ergebnis	Die Rückgabewerte (Hostnamen, IP- und MAC-Adresse) sollen identisch mit deren im Hostnamenkonzept sein.		

Tabelle 50: Testfall K-014

Bezeichnung	K-015	Compute Node c13	Hostnamen / IP / MAC
Beschreibung	Der Compute Node wird auf die zugewiesene IP Adresse, MAC Adresse und den Hostnamen geprüft.		
Testvoraussetzung	Der Compute Node und der Testclient befinden sich im selben Netzwerk.		
Testschritte	<ul style="list-style-type: none"> - Compute Node starten (Strom anschliessen) - 3 Minuten Sekunden warten - Auf dem Testclient über Putty oder Shell mit dem Befehl "nmap -sn 192.168.1.23" eingeben - Prüfen ob die Zuweisung gemäss Hostnamenkonzept richtig ist. 		
Erwartetes Ergebnis	Die Rückgabewerte (Hostnamen, IP- und MAC-Adresse) sollen identisch mit deren im Hostnamenkonzept sein.		

Tabelle 51: Testfall K-015

F Testfälle

Bezeichnung	K-016	Compute Node c14	Hostnamen / IP / MAC
Beschreibung	Der Compute Node wird auf die zugewiesene IP Adresse, MAC Adresse und den Hostnamen geprüft.		
Testvoraussetzung	Der Compute Node und der Testclient befinden sich im selben Netzwerk.		
Testschritte	<ul style="list-style-type: none"> - Compute Node starten (Strom anschliessen) - 3 Minuten Sekunden warten - Auf dem Testclient über Putty oder Shell mit dem Befehl "nmap -sn 192.168.1.24" eingeben - Prüfen ob die Zuweisung gemäss Hostnamenkonzept richtig ist. 		
Erwartetes Ergebnis	Die Rückgabewerte (Hostnamen, IP- und MAC-Adresse) sollen identisch mit deren im Hostnamenkonzept sein.		

Tabelle 52: Testfall K-016

Bezeichnung	K-017	Compute Node c15	Hostnamen / IP / MAC
Beschreibung	Der Compute Node wird auf die zugewiesene IP Adresse, MAC Adresse und den Hostnamen geprüft.		
Testvoraussetzung	Der Compute Node und der Testclient befinden sich im selben Netzwerk.		
Testschritte	<ul style="list-style-type: none"> - Compute Node starten (Strom anschliessen) - 3 Minuten Sekunden warten - Auf dem Testclient über Putty oder Shell mit dem Befehl "nmap -sn 192.168.1.25" eingeben - Prüfen ob die Zuweisung gemäss Hostnamenkonzept richtig ist. 		
Erwartetes Ergebnis	Die Rückgabewerte (Hostnamen, IP- und MAC-Adresse) sollen identisch mit deren im Hostnamenkonzept sein.		

Tabelle 53: Testfall K-017

Bezeichnung	K-018	Compute Node c16	Hostnamen / IP / MAC
Beschreibung	Der Compute Node wird auf die zugewiesene IP Adresse, MAC Adresse und den Hostnamen geprüft.		
Testvoraussetzung	Der Compute Node und der Testclient befinden sich im selben Netzwerk.		
Testschritte	<ul style="list-style-type: none"> - Compute Node starten (Strom anschliessen) - 3 Minuten Sekunden warten - Auf dem Testclient über Putty oder Shell mit dem Befehl "nmap -sn 192.168.1.26" eingeben - Prüfen ob die Zuweisung gemäss Hostnamenkonzept richtig ist. 		
Erwartetes Ergebnis	Die Rückgabewerte (Hostnamen, IP- und MAC-Adresse) sollen identisch mit deren im Hostnamenkonzept sein.		

Tabelle 54: Testfall K-018

F Testfälle

Bezeichnung	K-019	Compute Node c17	Hostnamen / IP / MAC
Beschreibung	Der Compute Node wird auf die zugewiesene IP Adresse, MAC Adresse und den Hostnamen geprüft.		
Testvoraussetzung	Der Compute Node und der Testclient befinden sich im selben Netzwerk.		
Testschritte	<ul style="list-style-type: none"> - Compute Node starten (Strom anschliessen) - 3 Minuten Sekunden warten - Auf dem Testclient über Putty oder Shell mit dem Befehl "nmap -sn 192.168.1.27" eingeben - Prüfen ob die Zuweisung gemäss Hostnamenkonzept richtig ist. 		
Erwartetes Ergebnis	Die Rückgabewerte (Hostnamen, IP- und MAC-Adresse) sollen identisch mit deren im Hostnamenkonzept sein.		

Tabelle 55: Testfall K-019

Bezeichnung	K-020	Compute Node c18	Hostnamen / IP / MAC
Beschreibung	Der Compute Node wird auf die zugewiesene IP Adresse, MAC Adresse und den Hostnamen geprüft.		
Testvoraussetzung	Der Compute Node und der Testclient befinden sich im selben Netzwerk.		
Testschritte	<ul style="list-style-type: none"> - Compute Node starten (Strom anschliessen) - 3 Minuten Sekunden warten - Auf dem Testclient über Putty oder Shell mit dem Befehl "nmap -sn 192.168.1.28" eingeben - Prüfen ob die Zuweisung gemäss Hostnamenkonzept richtig ist. 		
Erwartetes Ergebnis	Die Rückgabewerte (Hostnamen, IP- und MAC-Adresse) sollen identisch mit deren im Hostnamenkonzept sein.		

Tabelle 56: Testfall K-020

Bezeichnung	K-021	Compute Node c19	Hostnamen / IP / MAC
Beschreibung	Der Compute Node wird auf die zugewiesene IP Adresse, MAC Adresse und den Hostnamen geprüft.		
Testvoraussetzung	Der Compute Node und der Testclient befinden sich im selben Netzwerk.		
Testschritte	<ul style="list-style-type: none"> - Compute Node starten (Strom anschliessen) - 3 Minuten Sekunden warten - Auf dem Testclient über Putty oder Shell mit dem Befehl "nmap -sn 192.168.1.29" eingeben - Prüfen ob die Zuweisung gemäss Hostnamenkonzept richtig ist. 		
Erwartetes Ergebnis	Die Rückgabewerte (Hostnamen, IP- und MAC-Adresse) sollen identisch mit deren im Hostnamenkonzept sein.		

Tabelle 57: Testfall K-021

F Testfälle

Bezeichnung	K-022	Compute Node c20	Hostnamen / IP / MAC
Beschreibung	Der Compute Node wird auf die zugewiesene IP Adresse, MAC Adresse und den Hostnamen geprüft.		
Testvoraussetzung	Der Compute Node und der Testclient befinden sich im selben Netzwerk.		
Testschritte	<ul style="list-style-type: none"> - Compute Node starten (Strom anschliessen) - 3 Minuten Sekunden warten - Auf dem Testclient über Putty oder Shell mit dem Befehl "nmap -sn 192.168.1.30" eingeben - Prüfen ob die Zuweisung gemäss Hostnamenkonzept richtig ist. 		
Erwartetes Ergebnis	Die Rückgabewerte (Hostnamen, IP- und MAC-Adresse) sollen identisch mit deren im Hostnamenkonzept sein.		

Tabelle 58: Testfall K-022

Bezeichnung	K-023	Compute Node c21	Hostnamen / IP / MAC
Beschreibung	Der Compute Node wird auf die zugewiesene IP Adresse, MAC Adresse und den Hostnamen geprüft.		
Testvoraussetzung	Der Compute Node und der Testclient befinden sich im selben Netzwerk.		
Testschritte	<ul style="list-style-type: none"> - Compute Node starten (Strom anschliessen) - 3 Minuten Sekunden warten - Auf dem Testclient über Putty oder Shell mit dem Befehl "nmap -sn 192.168.1.31" eingeben - Prüfen ob die Zuweisung gemäss Hostnamenkonzept richtig ist. 		
Erwartetes Ergebnis	Die Rückgabewerte (Hostnamen, IP- und MAC-Adresse) sollen identisch mit deren im Hostnamenkonzept sein.		

Tabelle 59: Testfall K-023

Bezeichnung	K-024	Compute Node c22	Hostnamen / IP / MAC
Beschreibung	Der Compute Node wird auf die zugewiesene IP Adresse, MAC Adresse und den Hostnamen geprüft.		
Testvoraussetzung	Der Compute Node und der Testclient befinden sich im selben Netzwerk.		
Testschritte	<ul style="list-style-type: none"> - Compute Node starten (Strom anschliessen) - 3 Minuten Sekunden warten - Auf dem Testclient über Putty oder Shell mit dem Befehl "nmap -sn 192.168.1.32" eingeben - Prüfen ob die Zuweisung gemäss Hostnamenkonzept richtig ist. 		
Erwartetes Ergebnis	Die Rückgabewerte (Hostnamen, IP- und MAC-Adresse) sollen identisch mit deren im Hostnamenkonzept sein.		

Tabelle 60: Testfall K-024

F Testfälle

Bezeichnung	K-025	Compute Node c23	Hostnamen / IP / MAC
Beschreibung	Der Compute Node wird auf die zugewiesene IP Adresse, MAC Adresse und den Hostnamen geprüft.		
Testvoraussetzung	Der Compute Node und der Testclient befinden sich im selben Netzwerk.		
Testschritte	<ul style="list-style-type: none"> - Compute Node starten (Strom anschliessen) - 3 Minuten Sekunden warten - Auf dem Testclient über Putty oder Shell mit dem Befehl "nmap -sn 192.168.1.33" eingeben - Prüfen ob die Zuweisung gemäss Hostnamenkonzept richtig ist. 		
Erwartetes Ergebnis	Die Rückgabewerte (Hostnamen, IP- und MAC-Adresse) sollen identisch mit deren im Hostnamenkonzept sein.		

Tabelle 61: Testfall K-025

Bezeichnung	K-026	Compute Node c24	Hostnamen / IP / MAC
Beschreibung	Der Compute Node wird auf die zugewiesene IP Adresse, MAC Adresse und den Hostnamen geprüft.		
Testvoraussetzung	Der Compute Node und der Testclient befinden sich im selben Netzwerk.		
Testschritte	<ul style="list-style-type: none"> - Compute Node starten (Strom anschliessen) - 3 Minuten Sekunden warten - Auf dem Testclient über Putty oder Shell mit dem Befehl "nmap -sn 192.168.1.34" eingeben - Prüfen ob die Zuweisung gemäss Hostnamenkonzept richtig ist. 		
Erwartetes Ergebnis	Die Rückgabewerte (Hostnamen, IP- und MAC-Adresse) sollen identisch mit deren im Hostnamenkonzept sein.		

Tabelle 62: Testfall K-026

Bezeichnung	K-027	Compute Node c25	Hostnamen / IP / MAC
Beschreibung	Der Compute Node wird auf die zugewiesene IP Adresse, MAC Adresse und den Hostnamen geprüft.		
Testvoraussetzung	Der Compute Node und der Testclient befinden sich im selben Netzwerk.		
Testschritte	<ul style="list-style-type: none"> - Compute Node starten (Strom anschliessen) - 3 Minuten Sekunden warten - Auf dem Testclient über Putty oder Shell mit dem Befehl "nmap -sn 192.168.1.35" eingeben - Prüfen ob die Zuweisung gemäss Hostnamenkonzept richtig ist. 		
Erwartetes Ergebnis	Die Rückgabewerte (Hostnamen, IP- und MAC-Adresse) sollen identisch mit deren im Hostnamenkonzept sein.		

Tabelle 63: Testfall K-027

F Testfälle

Bezeichnung	K-028	Compute Node c26	Hostnamen / IP / MAC
Beschreibung	Der Compute Node wird auf die zugewiesene IP Adresse, MAC Adresse und den Hostnamen geprüft.		
Testvoraussetzung	Der Compute Node und der Testclient befinden sich im selben Netzwerk.		
Testschritte	<ul style="list-style-type: none"> - Compute Node starten (Strom anschliessen) - 3 Minuten Sekunden warten - Auf dem Testclient über Putty oder Shell mit dem Befehl "nmap -sn 192.168.1.36" eingeben - Prüfen ob die Zuweisung gemäss Hostnamenkonzept richtig ist. 		
Erwartetes Ergebnis	Die Rückgabewerte (Hostnamen, IP- und MAC-Adresse) sollen identisch mit deren im Hostnamenkonzept sein.		

Tabelle 64: Testfall K-028

Bezeichnung	K-029	Compute Node c27	Hostnamen / IP / MAC
Beschreibung	Der Compute Node wird auf die zugewiesene IP Adresse, MAC Adresse und den Hostnamen geprüft.		
Testvoraussetzung	Der Compute Node und der Testclient befinden sich im selben Netzwerk.		
Testschritte	<ul style="list-style-type: none"> - Compute Node starten (Strom anschliessen) - 3 Minuten Sekunden warten - Auf dem Testclient über Putty oder Shell mit dem Befehl "nmap -sn 192.168.1.37" eingeben - Prüfen ob die Zuweisung gemäss Hostnamenkonzept richtig ist. 		
Erwartetes Ergebnis	Die Rückgabewerte (Hostnamen, IP- und MAC-Adresse) sollen identisch mit deren im Hostnamenkonzept sein.		

Tabelle 65: Testfall K-029

Bezeichnung	K-030	Compute Node c28	Hostnamen / IP / MAC
Beschreibung	Der Compute Node wird auf die zugewiesene IP Adresse, MAC Adresse und den Hostnamen geprüft.		
Testvoraussetzung	Der Compute Node und der Testclient befinden sich im selben Netzwerk.		
Testschritte	<ul style="list-style-type: none"> - Compute Node starten (Strom anschliessen) - 3 Minuten Sekunden warten - Auf dem Testclient über Putty oder Shell mit dem Befehl "nmap -sn 192.168.1.38" eingeben - Prüfen ob die Zuweisung gemäss Hostnamenkonzept richtig ist. 		
Erwartetes Ergebnis	Die Rückgabewerte (Hostnamen, IP- und MAC-Adresse) sollen identisch mit deren im Hostnamenkonzept sein.		

Tabelle 66: Testfall K-030

F Testfälle

Bezeichnung	K-031	Compute Node c29	Hostnamen / IP / MAC
Beschreibung	Der Compute Node wird auf die zugewiesene IP Adresse, MAC Adresse und den Hostnamen geprüft.		
Testvoraussetzung	Der Compute Node und der Testclient befinden sich im selben Netzwerk.		
Testschritte	<ul style="list-style-type: none"> - Compute Node starten (Strom anschliessen) - 3 Minuten Sekunden warten - Auf dem Testclient über Putty oder Shell mit dem Befehl "nmap -sn 192.168.1.39" eingeben - Prüfen ob die Zuweisung gemäss Hostnamenkonzept richtig ist. 		
Erwartetes Ergebnis	Die Rückgabewerte (Hostnamen, IP- und MAC-Adresse) sollen identisch mit deren im Hostnamenkonzept sein.		

Tabelle 67: Testfall K-031

Bezeichnung	K-032	Compute Node c30	Hostnamen / IP / MAC
Beschreibung	Der Compute Node wird auf die zugewiesene IP Adresse, MAC Adresse und den Hostnamen geprüft.		
Testvoraussetzung	Der Compute Node und der Testclient befinden sich im selben Netzwerk.		
Testschritte	<ul style="list-style-type: none"> - Compute Node starten (Strom anschliessen) - 3 Minuten Sekunden warten - Auf dem Testclient über Putty oder Shell mit dem Befehl "nmap -sn 192.168.1.40" eingeben - Prüfen ob die Zuweisung gemäss Hostnamenkonzept richtig ist. 		
Erwartetes Ergebnis	Die Rückgabewerte (Hostnamen, IP- und MAC-Adresse) sollen identisch mit deren im Hostnamenkonzept sein.		

Tabelle 68: Testfall K-032

Bezeichnung	K-033	Compute Node c31	Hostnamen / IP / MAC
Beschreibung	Der Compute Node wird auf die zugewiesene IP Adresse, MAC Adresse und den Hostnamen geprüft.		
Testvoraussetzung	Der Compute Node und der Testclient befinden sich im selben Netzwerk.		
Testschritte	<ul style="list-style-type: none"> - Compute Node starten (Strom anschliessen) - 3 Minuten Sekunden warten - Auf dem Testclient über Putty oder Shell mit dem Befehl "nmap -sn 192.168.1.41" eingeben - Prüfen ob die Zuweisung gemäss Hostnamenkonzept richtig ist. 		
Erwartetes Ergebnis	Die Rückgabewerte (Hostnamen, IP- und MAC-Adresse) sollen identisch mit deren im Hostnamenkonzept sein.		

Tabelle 69: Testfall K-033

F Testfälle

Bezeichnung	K-034	Compute Node c32	Hostnamen / IP / MAC
Beschreibung	Der Compute Node wird auf die zugewiesene IP Adresse, MAC Adresse und den Hostnamen geprüft.		
Testvoraussetzung	Der Compute Node und der Testclient befinden sich im selben Netzwerk.		
Testschritte	<ul style="list-style-type: none"> - Compute Node starten (Strom anschliessen) - 3 Minuten Sekunden warten - Auf dem Testclient über Putty oder Shell mit dem Befehl "nmap -sn 192.168.1.42" eingeben - Prüfen ob die Zuweisung gemäss Hostnamenkonzept richtig ist. 		
Erwartetes Ergebnis	Die Rückgabewerte (Hostnamen, IP- und MAC-Adresse) sollen identisch mit deren im Hostnamenkonzept sein.		

Tabelle 70: Testfall K-034

Bezeichnung	K-035	Compute Node c33	Hostnamen / IP / MAC
Beschreibung	Der Compute Node wird auf die zugewiesene IP Adresse, MAC Adresse und den Hostnamen geprüft.		
Testvoraussetzung	Der Compute Node und der Testclient befinden sich im selben Netzwerk.		
Testschritte	<ul style="list-style-type: none"> - Compute Node starten (Strom anschliessen) - 3 Minuten Sekunden warten - Auf dem Testclient über Putty oder Shell mit dem Befehl "nmap -sn 192.168.1.43" eingeben - Prüfen ob die Zuweisung gemäss Hostnamenkonzept richtig ist. 		
Erwartetes Ergebnis	Die Rückgabewerte (Hostnamen, IP- und MAC-Adresse) sollen identisch mit deren im Hostnamenkonzept sein.		

Tabelle 71: Testfall K-035

Bezeichnung	K-036	Compute Node c34	Hostnamen / IP / MAC
Beschreibung	Der Compute Node wird auf die zugewiesene IP Adresse, MAC Adresse und den Hostnamen geprüft.		
Testvoraussetzung	Der Compute Node und der Testclient befinden sich im selben Netzwerk.		
Testschritte	<ul style="list-style-type: none"> - Compute Node starten (Strom anschliessen) - 3 Minuten Sekunden warten - Auf dem Testclient über Putty oder Shell mit dem Befehl "nmap -sn 192.168.1.44" eingeben - Prüfen ob die Zuweisung gemäss Hostnamenkonzept richtig ist. 		
Erwartetes Ergebnis	Die Rückgabewerte (Hostnamen, IP- und MAC-Adresse) sollen identisch mit deren im Hostnamenkonzept sein.		

Tabelle 72: Testfall K-036

F Testfälle

Bezeichnung	K-037	Compute Node c35	Hostnamen / IP / MAC
Beschreibung	Der Compute Node wird auf die zugewiesene IP Adresse, MAC Adresse und den Hostnamen geprüft.		
Testvoraussetzung	Der Compute Node und der Testclient befinden sich im selben Netzwerk.		
Testschritte	<ul style="list-style-type: none"> - Compute Node starten (Strom anschliessen) - 3 Minuten Sekunden warten - Auf dem Testclient über Putty oder Shell mit dem Befehl "nmap -sn 192.168.1.45" eingeben - Prüfen ob die Zuweisung gemäss Hostnamenkonzept richtig ist. 		
Erwartetes Ergebnis	Die Rückgabewerte (Hostnamen, IP- und MAC-Adresse) sollen identisch mit deren im Hostnamenkonzept sein.		

Tabelle 73: Testfall K-037

Bezeichnung	K-038	Compute Node c36	Hostnamen / IP / MAC
Beschreibung	Der Compute Node wird auf die zugewiesene IP Adresse, MAC Adresse und den Hostnamen geprüft.		
Testvoraussetzung	Der Compute Node und der Testclient befinden sich im selben Netzwerk.		
Testschritte	<ul style="list-style-type: none"> - Compute Node starten (Strom anschliessen) - 3 Minuten Sekunden warten - Auf dem Testclient über Putty oder Shell mit dem Befehl "nmap -sn 192.168.1.46" eingeben - Prüfen ob die Zuweisung gemäss Hostnamenkonzept richtig ist. 		
Erwartetes Ergebnis	Die Rückgabewerte (Hostnamen, IP- und MAC-Adresse) sollen identisch mit deren im Hostnamenkonzept sein.		

Tabelle 74: Testfall K-038

Bezeichnung	K-039	Compute Node c37	Hostnamen / IP / MAC
Beschreibung	Der Compute Node wird auf die zugewiesene IP Adresse, MAC Adresse und den Hostnamen geprüft.		
Testvoraussetzung	Der Compute Node und der Testclient befinden sich im selben Netzwerk.		
Testschritte	<ul style="list-style-type: none"> - Compute Node starten (Strom anschliessen) - 3 Minuten Sekunden warten - Auf dem Testclient über Putty oder Shell mit dem Befehl "nmap -sn 192.168.1.47" eingeben - Prüfen ob die Zuweisung gemäss Hostnamenkonzept richtig ist. 		
Erwartetes Ergebnis	Die Rückgabewerte (Hostnamen, IP- und MAC-Adresse) sollen identisch mit deren im Hostnamenkonzept sein.		

Tabelle 75: Testfall K-039

F Testfälle

Bezeichnung	K-040	Compute Node c38	Hostnamen / IP / MAC
Beschreibung	Der Compute Node wird auf die zugewiesene IP Adresse, MAC Adresse und den Hostnamen geprüft.		
Testvoraussetzung	Der Compute Node und der Testclient befinden sich im selben Netzwerk.		
Testschritte	<ul style="list-style-type: none"> - Compute Node starten (Strom anschliessen) - 3 Minuten Sekunden warten - Auf dem Testclient über Putty oder Shell mit dem Befehl "nmap -sn 192.168.1.48" eingeben - Prüfen ob die Zuweisung gemäss Hostnamenkonzept richtig ist. 		
Erwartetes Ergebnis	Die Rückgabewerte (Hostnamen, IP- und MAC-Adresse) sollen identisch mit deren im Hostnamenkonzept sein.		

Tabelle 76: Testfall K-040

Bezeichnung	K-041	Compute Node c39	Hostnamen / IP / MAC
Beschreibung	Der Compute Node wird auf die zugewiesene IP Adresse, MAC Adresse und den Hostnamen geprüft.		
Testvoraussetzung	Der Compute Node und der Testclient befinden sich im selben Netzwerk.		
Testschritte	<ul style="list-style-type: none"> - Compute Node starten (Strom anschliessen) - 3 Minuten Sekunden warten - Auf dem Testclient über Putty oder Shell mit dem Befehl "nmap -sn 192.168.1.49" eingeben - Prüfen ob die Zuweisung gemäss Hostnamenkonzept richtig ist. 		
Erwartetes Ergebnis	Die Rückgabewerte (Hostnamen, IP- und MAC-Adresse) sollen identisch mit deren im Hostnamenkonzept sein.		

Tabelle 77: Testfall K-041

Bezeichnung	K-042	Compute Node c40	Hostnamen / IP / MAC
Beschreibung	Der Compute Node wird auf die zugewiesene IP Adresse, MAC Adresse und den Hostnamen geprüft.		
Testvoraussetzung	Der Compute Node und der Testclient befinden sich im selben Netzwerk.		
Testschritte	<ul style="list-style-type: none"> - Compute Node starten (Strom anschliessen) - 3 Minuten Sekunden warten - Auf dem Testclient über Putty oder Shell mit dem Befehl "nmap -sn 192.168.1.50" eingeben - Prüfen ob die Zuweisung gemäss Hostnamenkonzept richtig ist. 		
Erwartetes Ergebnis	Die Rückgabewerte (Hostnamen, IP- und MAC-Adresse) sollen identisch mit deren im Hostnamenkonzept sein.		

Tabelle 78: Testfall K-042

F Testfälle

Bezeichnung	K-043	Compute Node c41	Hostnamen / IP / MAC
Beschreibung	Der Compute Node wird auf die zugewiesene IP Adresse, MAC Adresse und den Hostnamen geprüft.		
Testvoraussetzung	Der Compute Node und der Testclient befinden sich im selben Netzwerk.		
Testschritte	<ul style="list-style-type: none"> - Compute Node starten (Strom anschliessen) - 3 Minuten Sekunden warten - Auf dem Testclient über Putty oder Shell mit dem Befehl "nmap -sn 192.168.1.51" eingeben - Prüfen ob die Zuweisung gemäss Hostnamenkonzept richtig ist. 		
Erwartetes Ergebnis	Die Rückgabewerte (Hostnamen, IP- und MAC-Adresse) sollen identisch mit deren im Hostnamenkonzept sein.		

Tabelle 79: Testfall K-043

Bezeichnung	K-044	Compute Node c42	Hostnamen / IP / MAC
Beschreibung	Der Compute Node wird auf die zugewiesene IP Adresse, MAC Adresse und den Hostnamen geprüft.		
Testvoraussetzung	Der Compute Node und der Testclient befinden sich im selben Netzwerk.		
Testschritte	<ul style="list-style-type: none"> - Compute Node starten (Strom anschliessen) - 3 Minuten Sekunden warten - Auf dem Testclient über Putty oder Shell mit dem Befehl "nmap -sn 192.168.1.52" eingeben - Prüfen ob die Zuweisung gemäss Hostnamenkonzept richtig ist. 		
Erwartetes Ergebnis	Die Rückgabewerte (Hostnamen, IP- und MAC-Adresse) sollen identisch mit deren im Hostnamenkonzept sein.		

Tabelle 80: Testfall K-044

Bezeichnung	K-045	Compute Node c43	Hostnamen / IP / MAC
Beschreibung	Der Compute Node wird auf die zugewiesene IP Adresse, MAC Adresse und den Hostnamen geprüft.		
Testvoraussetzung	Der Compute Node und der Testclient befinden sich im selben Netzwerk.		
Testschritte	<ul style="list-style-type: none"> - Compute Node starten (Strom anschliessen) - 3 Minuten Sekunden warten - Auf dem Testclient über Putty oder Shell mit dem Befehl "nmap -sn 192.168.1.53" eingeben - Prüfen ob die Zuweisung gemäss Hostnamenkonzept richtig ist. 		
Erwartetes Ergebnis	Die Rückgabewerte (Hostnamen, IP- und MAC-Adresse) sollen identisch mit deren im Hostnamenkonzept sein.		

Tabelle 81: Testfall K-045

F Testfälle

Bezeichnung	K-046	Compute Node c44	Hostnamen / IP / MAC
Beschreibung	Der Compute Node wird auf die zugewiesene IP Adresse, MAC Adresse und den Hostnamen geprüft.		
Testvoraussetzung	Der Compute Node und der Testclient befinden sich im selben Netzwerk.		
Testschritte	<ul style="list-style-type: none"> - Compute Node starten (Strom anschliessen) - 3 Minuten Sekunden warten - Auf dem Testclient über Putty oder Shell mit dem Befehl "nmap -sn 192.168.1.54" eingeben - Prüfen ob die Zuweisung gemäss Hostnamenkonzept richtig ist. 		
Erwartetes Ergebnis	Die Rückgabewerte (Hostnamen, IP- und MAC-Adresse) sollen identisch mit deren im Hostnamenkonzept sein.		

Tabelle 82: Testfall K-046

Bezeichnung	K-047	Compute Node c45	Hostnamen / IP / MAC
Beschreibung	Der Compute Node wird auf die zugewiesene IP Adresse, MAC Adresse und den Hostnamen geprüft.		
Testvoraussetzung	Der Compute Node und der Testclient befinden sich im selben Netzwerk.		
Testschritte	<ul style="list-style-type: none"> - Compute Node starten (Strom anschliessen) - 3 Minuten Sekunden warten - Auf dem Testclient über Putty oder Shell mit dem Befehl "nmap -sn 192.168.1.55" eingeben - Prüfen ob die Zuweisung gemäss Hostnamenkonzept richtig ist. 		
Erwartetes Ergebnis	Die Rückgabewerte (Hostnamen, IP- und MAC-Adresse) sollen identisch mit deren im Hostnamenkonzept sein.		

Tabelle 83: Testfall K-047

Bezeichnung	K-048	NAS	Erreichbarkeit
Beschreibung	Das NAS soll auf die Erreichbarkeit geprüft werden.		
Testvoraussetzung	Das NAS ist am Netzwerk angeschlossen.		
Testschritte			
Erwartetes Ergebnis	Das NAS antwortet auf den Ping befehl mit: "2 packets transmitted, 2 received, 0% packet loss"		

Tabelle 84: Testfall K-048

F.2 Integrationstests

Bezeichnung	I-001	Compute Nodes	Internet Zugang
Beschreibung	Es wird getestet ob alle Computenodes auf das Internet zugreifen können.		
Testvoraussetzung	Das Routing ist eingerichtet.		
Testschritte	- Den Befehl "pdsh -w c[1-45] ping -c 1 google.de" auf dem Management Node eingeben.		
Erwartetes Ergebnis	Der Rückgabewert muss pro Compute Node die Ausgabe "1 packets transmitted, 1 received, 0% packet loss" beinhalten. Damit der Test erfolgreich war.		

Tabelle 85: Testfall I-001

Bezeichnung	I-002	NAS	Mountpoint
Beschreibung	Es wird getestet ob der Management Node und die Compute Nodes automatisch bei einem Systemstart sich mit dem Netzwerkshare verbinden.		
Testvoraussetzung	Der fstab-Eintrag muss vorhanden sein.		
Testschritte	<ul style="list-style-type: none"> - Management Node starten. - auf dem Management Node anmelden. - Den Befehl "ls -l /media/nebula_data/ wc -l " eingeben. 		
Erwartetes Ergebnis	Die Ausgabe des Befehls muss einen grösseren Wert als "1" zurückgeben.		

Tabelle 86: Testfall I-002

F.3 Systemtests

Bezeichnung	S-001	Compute Nodes	CPU Last
Beschreibung	Es wird geprüft ob die CPU während des Schürfens der Kryptowährungen zu über 90% beansprucht wird.		
Testvoraussetzung	Der Cluster muss am Schürfen einer Kryptowährung sein. Dazu muss er bereits 15 Minuten am schürfen sein, bevor mit dem Test begonnen werden kann.		
Testschritte	Die Tests finden vom Management Node aus statt. - Den Befehl "pdsh -w c[1-45] uptime" eingeben. - Alternativ kann man sich auf jeden Compute Node anmelden und den Befehl "top" eingeben.		
Erwartetes Ergebnis	Die Loadaverage-Werte überschreiten jeweils den Wert 1, die Ausgabe sollte in etwa so aussehen: load average: 4.18, 2.44, 1.16		

Tabelle 87: Testfall S-001

F Testfälle

Bezeichnung	S-002	Nodes	Nagios Monitoring
Beschreibung	Es wird geprüft ob alle Nodes von Nagios überwacht werden.		
Testvoraussetzung	Alle Nodes müssen einmal in Betrieb gewesen sein		
Testschritte	<ul style="list-style-type: none"> - http://nebula/nagios aufrufen. - Bei Nagios anmelden. - Auf den Reiter "Hosts" klicken. 		
Erwartetes Ergebnis	Es werden alle Nodes aufgelistet. Unabhängig des Status ob der Node in Betrieb ist oder nicht.		

Tabelle 88: Testfall S-002

Bezeichnung	S-003	Nodes	Nagios Alarmierung
Beschreibung	Bei aufgetretenen Problemen soll Nagios eine E-Mail versenden.		
Testvoraussetzung	Der Cluster muss in Betrieb sein.		
Testschritte	<ul style="list-style-type: none"> - Strom von Node c1 trennen. - 5 Minuten warten. - Email prüfen (christoph.amrein86@gmail.com) 		
Erwartetes Ergebnis	Es trifft eine Alarmierungs-Email ein.		

Tabelle 89: Testfall S-004

Bezeichnung	S-004	Nodes	Ganglia Monitoring
Beschreibung	Es wird geprüft ob alle Nodes von Ganglia gemonitored werden.		
Testvoraussetzung	Alle Nodes müssen einmal in Betrieb gewesen sein		
Testschritte	<ul style="list-style-type: none"> - http://nebula/ganglia aufrufen. - Quelle "Nebula" auswählen. 		
Erwartetes Ergebnis	Es werden alle Nodes aufgelistet.		

Tabelle 90: Testfall S-004

Bezeichnung	S-005	Nodes	Cluster Jobs
Beschreibung	Es wird geprüft ob Jobs auf dem Cluster laufen.		
Testvoraussetzung	Der Cluster muss in Betrieb sein.		
Testschritte	Auf dem Management Node sind folgende Befehle abzusetzen: <ul style="list-style-type: none"> - "mpicc -O3 /opt/ohpc/pub/examples/mpi/hello.c" - "srun -n 8 -N 2 -pty /bin/bash" - "squeue" 		
Erwartetes Ergebnis	Der Job wird in der Queue angezeigt und ist 2 Nodes zugewiesen.		

Tabelle 91: Testfall S-005

F Testfälle

Bezeichnung	S-006	Nodes	Schürfen
Beschreibung	Es wird geprüft ob über den Cluster Kryptowährung über einen Job geschürft werden kann.		
Testvoraussetzung	Der Cluster muss in Betrieb sein.		
Testschritte	Auf dem Management Node sind folgende Befehle als root abzusetzen: - "cd /opt/miners/tkinjo" - "srun -nodes=40-45 -ntasks=40 -cpus-per-task=4 ./cpuminer -a cryptonight -o stratum+tcp://xdn.pool.minergate.com:45620 -u x -p x"		
Erwartetes Ergebnis	Es soll direkt in der Shell die Ausgabe des Miners ausgegeben werden.		

Tabelle 92: Testfall S-006

G Testprotokoll

G.1 Komponententests

Test-ID:	K-001
Testobjekt	Management Node
Testschritte	<ul style="list-style-type: none"> - Management Node starten (Strom anschliessen). - 30 Sekunden warten. - Auf dem Testclient über Putty oder Shell mit dem Befehl "nmap -sn 192.168.1.10" eingeben. - Prüfen, ob die Zuweisung gemäss Hostnamenkonzept richtig ist.
Erwartetes Ergebnis	Hostname = nebula IP = 192.168.1.10 MAC = B8:27:EB:32:A9:1C
Tatsächliches Ergebnis	Nmap scan report for nebula.home (192.168.1.10) MAC Address: B8:27:EB:32:A9:1C (Raspberry Pi Foundation) Nmap done: 1 IP address (1 host up) scanned in 0.26 seconds
Tester	Christoph Amrein
Datum des Tests	21.05.2018
Testergebnis (Fehlerklasse)	1 - Fehlerfrei, die Erwartungen sind erfüllt.
Fehlerbeschreibung	

Tabelle 93: K-001 Protokoll

Test-ID:	K-002
Testobjekt	Management Node
Testschritte	<ul style="list-style-type: none"> - Management Node starten (Strom anschliessen). - 30 Sekunden warten. - Auf dem Testclient über Putty oder Shell den Befehl "ssh root@nebula" eingeben. - Passwort eingeben.
Erwartetes Ergebnis	Die SSH Verbindung auf den Management Node hat funktioniert und man ist als root-Benutzer angemeldet.
Tatsächliches Ergebnis	login as: root root@nebula's password: Last login: Fri May 18 17:40:34 2018 from desktop-rrq1k7v.home
Tester	Christoph Amrein
Datum des Tests	21.05.2018
Testergebnis (Fehlerklasse)	1 - Fehlerfrei, die Erwartungen sind erfüllt.
Fehlerbeschreibung	

Tabelle 94: K-002 Protokoll

G Testprotokoll

Test-ID:	K-003
Testobjekt	Compute Node c1
Testschritte	<ul style="list-style-type: none"> - Management Node starten (Strom anschliessen). - 3 Minuten warten. - Auf dem Testclient über Putty oder Shell mit dem Befehl "nmap -sn 192.168.1.11" eingeben. - Prüfen, ob die Zuweisung gemäss Hostnamenkonzept richtig ist.
Erwartetes Ergebnis	Hostname = c1 IP = 192.168.1.11 MAC = B8:27:EB:32:39:A7
Tatsächliches Ergebnis	Nmap scan report for c1.home (192.168.1.11) MAC Address: B8:27:EB:32:39:A7 (Raspberry Pi Foundation) Nmap done: 1 IP address (1 host up) scanned in 0.09 seconds
Tester	Christoph Amrein
Datum des Tests	21.05.2018
Testergebnis (Fehlerklasse)	1 - Fehlerfrei, die Erwartungen sind erfüllt.
Fehlerbeschreibung	

Tabelle 95: K-003 Protokoll

Test-ID:	K-004
Testobjekt	Compute Node c2
Testschritte	<ul style="list-style-type: none"> - Management Node starten (Strom anschliessen). - 3 Minuten warten. - Auf dem Testclient über Putty oder Shell mit dem Befehl "nmap -sn 192.168.1.12" eingeben. - Prüfen, ob die Zuweisung gemäss Hostnamenkonzept richtig ist.
Erwartetes Ergebnis	Hostname = c2 IP = 192.168.1.12 MAC = B8:27:EB:2E:A3:D1
Tatsächliches Ergebnis	Nmap scan report for c2.home (192.168.1.12) MAC Address: B8:27:EB:2E:A3:D1 (Raspberry Pi Foundation) Nmap done: 1 IP address (1 host up) scanned in 0.09 seconds
Tester	Christoph Amrein
Datum des Tests	22.05.2018
Testergebnis (Fehlerklasse)	1 - Fehlerfrei, die Erwartungen sind erfüllt.
Fehlerbeschreibung	

Tabelle 96: K-004 Protokoll

G Testprotokoll

Test-ID:	K-005
Testobjekt	Compute Node c3
Testschritte	<ul style="list-style-type: none"> - Management Node starten (Strom anschliessen). - 3 Minuten warten. - Auf dem Testclient über Putty oder Shell mit dem Befehl "nmap -sn 192.168.1.13" eingeben. - Prüfen, ob die Zuweisung gemäss Hostnamenkonzept richtig ist.
Erwartetes Ergebnis	Hostname = c3 IP = 192.168.1.13 MAC = B8:27:EB:50:45:3F
Tatsächliches Ergebnis	Nmap scan report for c3.home (192.168.1.13) MAC Address: B8:27:EB:50:45:3F (Raspberry Pi Foundation) Nmap done: 1 IP address (1 host up) scanned in 0.09 seconds
Tester	Christoph Amrein
Datum des Tests	22.05.2018
Testergebnis (Fehlerklasse)	1 - Fehlerfrei, die Erwartungen sind erfüllt.
Fehlerbeschreibung	

Tabelle 97: K-005 Protokoll

Test-ID:	K-006
Testobjekt	Compute Node c4
Testschritte	<ul style="list-style-type: none"> - Management Node starten (Strom anschliessen). - 3 Minuten warten. - Auf dem Testclient über Putty oder Shell mit dem Befehl "nmap -sn 192.168.1.14" eingeben. - Prüfen, ob die Zuweisung gemäss Hostnamenkonzept richtig ist.
Erwartetes Ergebnis	Hostname = c4 IP = 192.168.1.14 MAC = B8:27:EB:0D:E6:25
Tatsächliches Ergebnis	Nmap scan report for c4.home (192.168.1.14) MAC Address: B8:27:EB:0D:E6:25 (Raspberry Pi Foundation) Nmap done: 1 IP address (1 host up) scanned in 0.10 seconds
Tester	Christoph Amrein
Datum des Tests	22.05.2018
Testergebnis (Fehlerklasse)	1 - Fehlerfrei, die Erwartungen sind erfüllt.
Fehlerbeschreibung	

Tabelle 98: K-006 Protokoll

G Testprotokoll

Test-ID:	K-007
Testobjekt	Compute Node c5
Testschritte	<ul style="list-style-type: none"> - Management Node starten (Strom anschliessen). - 3 Minuten warten. - Auf dem Testclient über Putty oder Shell mit dem Befehl "nmap -sn 192.168.1.15" eingeben. - Prüfen, ob die Zuweisung gemäss Hostnamenkonzept richtig ist.
Erwartetes Ergebnis	Hostname = c5 IP = 192.168.1.15 MAC = B8:27:EB:3E:96:B5
Tatsächliches Ergebnis	Nmap scan report for c5.home (192.168.1.15) MAC Address: B8:27:EB:3E:96:B5 (Raspberry Pi Foundation) Nmap done: 1 IP address (1 host up) scanned in 0.09 seconds
Tester	Christoph Amrein
Datum des Tests	22.05.2018
Testergebnis (Fehlerklasse)	1 - Fehlerfrei, die Erwartungen sind erfüllt.
Fehlerbeschreibung	

Tabelle 99: K-007 Protokoll

Test-ID:	K-008
Testobjekt	Compute Node c6
Testschritte	<ul style="list-style-type: none"> - Management Node starten (Strom anschliessen). - 3 Minuten warten. - Auf dem Testclient über Putty oder Shell mit dem Befehl "nmap -sn 192.168.1.16" eingeben. - Prüfen, ob die Zuweisung gemäss Hostnamenkonzept richtig ist.
Erwartetes Ergebnis	Hostname = c6 IP = 192.168.1.16 MAC = B8:27:EB:EE:77:DA
Tatsächliches Ergebnis	Nmap scan report for c6.home (192.168.1.16) MAC Address: B8:27:EB:EE:77:DA (Raspberry Pi Foundation) Nmap done: 1 IP address (1 host up) scanned in 0.09 seconds
Tester	Christoph Amrein
Datum des Tests	22.05.2018
Testergebnis (Fehlerklasse)	1 - Fehlerfrei, die Erwartungen sind erfüllt.
Fehlerbeschreibung	

Tabelle 100: K-008 Protokoll

G Testprotokoll

Test-ID:	K-009
Testobjekt	Compute Node c7
Testschritte	<ul style="list-style-type: none"> - Management Node starten (Strom anschliessen). - 3 Minuten warten. - Auf dem Testclient über Putty oder Shell mit dem Befehl "nmap -sn 192.168.1.17" eingeben. - Prüfen, ob die Zuweisung gemäss Hostnamenkonzept richtig ist.
Erwartetes Ergebnis	Hostname = c7 IP = 192.168.1.17 MAC = B8:27:EB:21:63:E6
Tatsächliches Ergebnis	Nmap scan report for c7.home (192.168.1.17) MAC Address: B8:27:EB:21:63:E6 (Raspberry Pi Foundation) Nmap done: 1 IP address (1 host up) scanned in 0.09 seconds
Tester	Christoph Amrein
Datum des Tests	22.05.2018
Testergebnis (Fehlerklasse)	1 - Fehlerfrei, die Erwartungen sind erfüllt.
Fehlerbeschreibung	

Tabelle 101: K-009 Protokoll

Test-ID:	K-010
Testobjekt	Compute Node c8
Testschritte	<ul style="list-style-type: none"> - Management Node starten (Strom anschliessen). - 3 Minuten warten. - Auf dem Testclient über Putty oder Shell mit dem Befehl "nmap -sn 192.168.1.18" eingeben. - Prüfen, ob die Zuweisung gemäss Hostnamenkonzept richtig ist.
Erwartetes Ergebnis	Hostname = c8 IP = 192.168.1.18 MAC = B8:27:EB:2E:2E:CC
Tatsächliches Ergebnis	Nmap scan report for c8.home (192.168.1.18) MAC Address: B8:27:EB:2E:2E:CC (Raspberry Pi Foundation) Nmap done: 1 IP address (1 host up) scanned in 0.09 seconds
Tester	Christoph Amrein
Datum des Tests	22.05.2018
Testergebnis (Fehlerklasse)	1 - Fehlerfrei, die Erwartungen sind erfüllt.
Fehlerbeschreibung	

Tabelle 102: K-010 Protokoll

G Testprotokoll

Test-ID:	K-011
Testobjekt	Compute Node c9
Testschritte	<ul style="list-style-type: none"> - Management Node starten (Strom anschliessen). - 3 Minuten warten. - Auf dem Testclient über Putty oder Shell mit dem Befehl "nmap -sn 192.168.1.19" eingeben. - Prüfen, ob die Zuweisung gemäss Hostnamenkonzept richtig ist.
Erwartetes Ergebnis	Hostname = c9 IP = 192.168.1.19 MAC = B8:27:EB:17:32:96
Tatsächliches Ergebnis	Nmap scan report for galaxy-a5-2017.home (192.168.1.19) MAC Address: B8:27:EB:17:32:96 (Raspberry Pi Foundation) Nmap done: 1 IP address (1 host up) scanned in 0.09 seconds
Tester	Christoph Amrein
Datum des Tests	22.05.2018
Testergebnis (Fehlerklasse)	1 - Fehlerfrei, die Erwartungen sind erfüllt.
Fehlerbeschreibung	

Tabelle 103: K-011 Protokoll

Test-ID:	K-012
Testobjekt	Compute Node c10
Testschritte	<ul style="list-style-type: none"> - Management Node starten (Strom anschliessen). - 3 Minuten warten. - Auf dem Testclient über Putty oder Shell mit dem Befehl "nmap -sn 192.168.1.20" eingeben. - Prüfen, ob die Zuweisung gemäss Hostnamenkonzept richtig ist.
Erwartetes Ergebnis	Hostname = c10 IP = 192.168.1.20 MAC = B8:27:EB:B2:1C:A9
Tatsächliches Ergebnis	Nmap scan report for c10.home (192.168.1.20) MAC Address: B8:27:EB:B2:1C:A9 (Raspberry Pi Foundation) Nmap done: 1 IP address (1 host up) scanned in 0.09 seconds
Tester	Christoph Amrein
Datum des Tests	22.05.2018
Testergebnis (Fehlerklasse)	1 - Fehlerfrei, die Erwartungen sind erfüllt.
Fehlerbeschreibung	

Tabelle 104: K-012 Protokoll

G Testprotokoll

Test-ID:	K-013
Testobjekt	Compute Node c11
Testschritte	<ul style="list-style-type: none"> - Management Node starten (Strom anschliessen). - 3 Minuten warten. - Auf dem Testclient über Putty oder Shell mit dem Befehl "nmap -sn 192.168.1.21" eingeben. - Prüfen, ob die Zuweisung gemäss Hostnamenkonzept richtig ist.
Erwartetes Ergebnis	Hostname = c11 IP = 192.168.1.21 MAC = B8:27:EB:AF:63:1F
Tatsächliches Ergebnis	Nmap scan report for c11.home (192.168.1.21) MAC Address: B8:27:EB:AF:63:1F (Raspberry Pi Foundation) Nmap done: 1 IP address (1 host up) scanned in 0.09 seconds
Tester	Christoph Amrein
Datum des Tests	22.05.2018
Testergebnis (Fehlerklasse)	1 - Fehlerfrei, die Erwartungen sind erfüllt.
Fehlerbeschreibung	

Tabelle 105: K-013 Protokoll

Test-ID:	K-014
Testobjekt	Compute Node c12
Testschritte	<ul style="list-style-type: none"> - Management Node starten (Strom anschliessen). - 3 Minuten warten. - Auf dem Testclient über Putty oder Shell mit dem Befehl "nmap -sn 192.168.1.22" eingeben. - Prüfen, ob die Zuweisung gemäss Hostnamenkonzept richtig ist.
Erwartetes Ergebnis	Hostname = c12 IP = 192.168.1.22 MAC = B8:27:EB:43:00:2C
Tatsächliches Ergebnis	Nmap scan report for c12.home (192.168.1.22) MAC Address: B8:27:EB:43:00:2C (Raspberry Pi Foundation) Nmap done: 1 IP address (1 host up) scanned in 0.10 seconds
Tester	Christoph Amrein
Datum des Tests	22.05.2018
Testergebnis (Fehlerklasse)	1 - Fehlerfrei, die Erwartungen sind erfüllt.
Fehlerbeschreibung	

Tabelle 106: K-014 Protokoll

G Testprotokoll

Test-ID:	K-015
Testobjekt	Compute Node c13
Testschritte	<ul style="list-style-type: none"> - Management Node starten (Strom anschliessen). - 3 Minuten warten. - Auf dem Testclient über Putty oder Shell mit dem Befehl "nmap -sn 192.168.1.23" eingeben. - Prüfen, ob die Zuweisung gemäss Hostnamenkonzept richtig ist.
Erwartetes Ergebnis	Hostname = c13 IP = 192.168.1.23 MAC = B8:27:EB:13:7B:18
Tatsächliches Ergebnis	Nmap scan report for c13.home (192.168.1.23) MAC Address: B8:27:EB:13:7B:18 (Raspberry Pi Foundation) Nmap done: 1 IP address (1 host up) scanned in 0.09 seconds
Tester	Christoph Amrein
Datum des Tests	22.05.2018
Testergebnis (Fehlerklasse)	1 - Fehlerfrei, die Erwartungen sind erfüllt.
Fehlerbeschreibung	

Tabelle 107: K-015 Protokoll

Test-ID:	K-016
Testobjekt	Compute Node c14
Testschritte	<ul style="list-style-type: none"> - Management Node starten (Strom anschliessen). - 3 Minuten warten. - Auf dem Testclient über Putty oder Shell mit dem Befehl "nmap -sn 192.168.1.24" eingeben. - Prüfen, ob die Zuweisung gemäss Hostnamenkonzept richtig ist.
Erwartetes Ergebnis	Hostname = c14 IP = 192.168.1.24 MAC = B8:27:EB:43:CD:29
Tatsächliches Ergebnis	Nmap scan report for c14.home (192.168.1.24) MAC Address: B8:27:EB:43:CD:29 (Raspberry Pi Foundation) Nmap done: 1 IP address (1 host up) scanned in 0.09 seconds
Tester	Christoph Amrein
Datum des Tests	22.05.2018
Testergebnis (Fehlerklasse)	1 - Fehlerfrei, die Erwartungen sind erfüllt.
Fehlerbeschreibung	

Tabelle 108: K-016 Protokoll

G Testprotokoll

Test-ID:	K-017
Testobjekt	Compute Node c15
Testschritte	<ul style="list-style-type: none"> - Management Node starten (Strom anschliessen). - 3 Minuten warten. - Auf dem Testclient über Putty oder Shell mit dem Befehl "nmap -sn 192.168.1.25" eingeben. - Prüfen, ob die Zuweisung gemäss Hostnamenkonzept richtig ist.
Erwartetes Ergebnis	Hostname = c15 IP = 192.168.1.25 MAC = B8:27:EB:FF:C7:56
Tatsächliches Ergebnis	Nmap scan report for c15.home (192.168.1.25) MAC Address: B8:27:EB:FF:C7:56 (Raspberry Pi Foundation) Nmap done: 1 IP address (1 host up) scanned in 0.09 seconds
Tester	Christoph Amrein
Datum des Tests	22.05.2018
Testergebnis (Fehlerklasse)	1 - Fehlerfrei, die Erwartungen sind erfüllt.
Fehlerbeschreibung	

Tabelle 109: K-017 Protokoll

Test-ID:	K-018
Testobjekt	Compute Node c16
Testschritte	<ul style="list-style-type: none"> - Management Node starten (Strom anschliessen). - 3 Minuten warten. - Auf dem Testclient über Putty oder Shell mit dem Befehl "nmap -sn 192.168.1.26" eingeben. - Prüfen, ob die Zuweisung gemäss Hostnamenkonzept richtig ist.
Erwartetes Ergebnis	Hostname = c16 IP = 192.168.1.26 MAC = B8:27:EB:CE:98:66
Tatsächliches Ergebnis	Nmap scan report for c16.home (192.168.1.26) MAC Address: B8:27:EB:CE:98:66 (Raspberry Pi Foundation) Nmap done: 1 IP address (1 host up) scanned in 0.10 seconds
Tester	Christoph Amrein
Datum des Tests	22.05.2018
Testergebnis (Fehlerklasse)	1 - Fehlerfrei, die Erwartungen sind erfüllt.
Fehlerbeschreibung	

Tabelle 110: K-018 Protokoll

G Testprotokoll

Test-ID:	K-019
Testobjekt	Compute Node c17
Testschritte	<ul style="list-style-type: none"> - Management Node starten (Strom anschliessen). - 3 Minuten warten. - Auf dem Testclient über Putty oder Shell mit dem Befehl "nmap -sn 192.168.1.27" eingeben. - Prüfen, ob die Zuweisung gemäss Hostnamenkonzept richtig ist.
Erwartetes Ergebnis	Hostname = c17 IP = 192.168.1.27 MAC = B8:27:EB:5D:63:34
Tatsächliches Ergebnis	Nmap scan report for c17.home (192.168.1.27) MAC Address: B8:27:EB:5D:63:34 (Raspberry Pi Foundation) Nmap done: 1 IP address (1 host up) scanned in 0.09 seconds
Tester	Christoph Amrein
Datum des Tests	22.05.2018
Testergebnis (Fehlerklasse)	1 - Fehlerfrei, die Erwartungen sind erfüllt.
Fehlerbeschreibung	

Tabelle 111: K-019 Protokoll

Test-ID:	K-020
Testobjekt	Compute Node c18
Testschritte	<ul style="list-style-type: none"> - Management Node starten (Strom anschliessen). - 3 Minuten warten. - Auf dem Testclient über Putty oder Shell mit dem Befehl "nmap -sn 192.168.1.28" eingeben. - Prüfen, ob die Zuweisung gemäss Hostnamenkonzept richtig ist.
Erwartetes Ergebnis	Hostname = c18 IP = 192.168.1.28 MAC = B8:27:EB:91:3E:0F
Tatsächliches Ergebnis	Nmap scan report for c18.home (192.168.1.28) MAC Address: B8:27:EB:91:3E:0F (Raspberry Pi Foundation) Nmap done: 1 IP address (1 host up) scanned in 0.08 seconds
Tester	Christoph Amrein
Datum des Tests	22.05.2018
Testergebnis (Fehlerklasse)	1 - Fehlerfrei, die Erwartungen sind erfüllt.
Fehlerbeschreibung	

Tabelle 112: K-020 Protokoll

G Testprotokoll

Test-ID:	K-021
Testobjekt	Compute Node c19
Testschritte	<ul style="list-style-type: none"> - Management Node starten (Strom anschliessen). - 3 Minuten warten. - Auf dem Testclient über Putty oder Shell mit dem Befehl "nmap -sn 192.168.1.29" eingeben. - Prüfen, ob die Zuweisung gemäss Hostnamenkonzept richtig ist.
Erwartetes Ergebnis	Hostname = c19 IP = 192.168.1.29 MAC = B8:27:EB:F4:65:EC
Tatsächliches Ergebnis	Nmap scan report for c19.home (192.168.1.29) MAC Address: B8:27:EB:F4:65:EC (Raspberry Pi Foundation) Nmap done: 1 IP address (1 host up) scanned in 0.09 seconds
Tester	Christoph Amrein
Datum des Tests	22.05.2018
Testergebnis (Fehlerklasse)	1 - Fehlerfrei, die Erwartungen sind erfüllt.
Fehlerbeschreibung	

Tabelle 113: K-021 Protokoll

Test-ID:	K-022
Testobjekt	Compute Node c20
Testschritte	<ul style="list-style-type: none"> - Management Node starten (Strom anschliessen). - 3 Minuten warten. - Auf dem Testclient über Putty oder Shell mit dem Befehl "nmap -sn 192.168.1.30" eingeben. - Prüfen, ob die Zuweisung gemäss Hostnamenkonzept richtig ist.
Erwartetes Ergebnis	Hostname = c20 IP = 192.168.1.30 MAC = B8:27:EB:3E:AB:DC
Tatsächliches Ergebnis	Nmap scan report for c20.home (192.168.1.30) MAC Address: B8:27:EB:3E:AB:DC (Raspberry Pi Foundation) Nmap done: 1 IP address (1 host up) scanned in 0.10 seconds
Tester	Christoph Amrein
Datum des Tests	22.05.2018
Testergebnis (Fehlerklasse)	1 - Fehlerfrei, die Erwartungen sind erfüllt.
Fehlerbeschreibung	

Tabelle 114: K-022 Protokoll

G Testprotokoll

Test-ID:	K-023
Testobjekt	Compute Node c21
Testschritte	<ul style="list-style-type: none"> - Management Node starten (Strom anschliessen). - 3 Minuten warten. - Auf dem Testclient über Putty oder Shell mit dem Befehl "nmap -sn 192.168.1.31" eingeben. - Prüfen, ob die Zuweisung gemäss Hostnamenkonzept richtig ist.
Erwartetes Ergebnis	Hostname = c21 IP = 192.168.1.31 MAC = B8:27:EB:66:60:F6
Tatsächliches Ergebnis	Nmap scan report for c21.home (192.168.1.31) MAC Address: B8:27:EB:66:60:F6 (Raspberry Pi Foundation) Nmap done: 1 IP address (1 host up) scanned in 0.09 seconds
Tester	Christoph Amrein
Datum des Tests	22.05.2018
Testergebnis (Fehlerklasse)	1 - Fehlerfrei, die Erwartungen sind erfüllt.
Fehlerbeschreibung	

Tabelle 115: K-023 Protokoll

Test-ID:	K-024
Testobjekt	Compute Node c22
Testschritte	<ul style="list-style-type: none"> - Management Node starten (Strom anschliessen). - 3 Minuten warten. - Auf dem Testclient über Putty oder Shell mit dem Befehl "nmap -sn 192.168.1.32" eingeben. - Prüfen, ob die Zuweisung gemäss Hostnamenkonzept richtig ist.
Erwartetes Ergebnis	Hostname = c22 IP = 192.168.1.32 MAC = B8:27:EB:37:3F:74
Tatsächliches Ergebnis	Nmap scan report for c22.home (192.168.1.32) MAC Address: B8:27:EB:37:3F:74 (Raspberry Pi Foundation) Nmap done: 1 IP address (1 host up) scanned in 0.09 seconds
Tester	Christoph Amrein
Datum des Tests	22.05.2018
Testergebnis (Fehlerklasse)	1 - Fehlerfrei, die Erwartungen sind erfüllt.
Fehlerbeschreibung	

Tabelle 116: K-024 Protokoll

G Testprotokoll

Test-ID:	K-025
Testobjekt	Compute Node c23
Testschritte	<ul style="list-style-type: none"> - Management Node starten (Strom anschliessen). - 3 Minuten warten. - Auf dem Testclient über Putty oder Shell mit dem Befehl "nmap -sn 192.168.1.33" eingeben. - Prüfen, ob die Zuweisung gemäss Hostnamenkonzept richtig ist.
Erwartetes Ergebnis	Hostname = c23 IP = 192.168.1.33 MAC = B8:27:EB:18:5E:F0
Tatsächliches Ergebnis	Nmap scan report for c23.home (192.168.1.33) MAC Address: B8:27:EB:18:5E:F0 (Raspberry Pi Foundation) Nmap done: 1 IP address (1 host up) scanned in 0.09 seconds
Tester	Christoph Amrein
Datum des Tests	22.05.2018
Testergebnis (Fehlerklasse)	1 - Fehlerfrei, die Erwartungen sind erfüllt.
Fehlerbeschreibung	

Tabelle 117: K-025 Protokoll

Test-ID:	K-026
Testobjekt	Compute Node c24
Testschritte	<ul style="list-style-type: none"> - Management Node starten (Strom anschliessen). - 3 Minuten warten. - Auf dem Testclient über Putty oder Shell mit dem Befehl "nmap -sn 192.168.1.34" eingeben. - Prüfen, ob die Zuweisung gemäss Hostnamenkonzept richtig ist.
Erwartetes Ergebnis	Hostname = c24 IP = 192.168.1.34 MAC = B8:27:EB:B0:23:B8
Tatsächliches Ergebnis	Nmap scan report for c24.home (192.168.1.34) MAC Address: B8:27:EB:B0:23:B8 (Raspberry Pi Foundation) Nmap done: 1 IP address (1 host up) scanned in 0.09 seconds
Tester	Christoph Amrein
Datum des Tests	22.05.2018
Testergebnis (Fehlerklasse)	1 - Fehlerfrei, die Erwartungen sind erfüllt.
Fehlerbeschreibung	

Tabelle 118: K-026 Protokoll

G Testprotokoll

Test-ID:	K-027
Testobjekt	Compute Node c25
Testschritte	<ul style="list-style-type: none"> - Management Node starten (Strom anschliessen). - 3 Minuten warten. - Auf dem Testclient über Putty oder Shell mit dem Befehl "nmap -sn 192.168.1.35" eingeben. - Prüfen, ob die Zuweisung gemäss Hostnamenkonzept richtig ist.
Erwartetes Ergebnis	Hostname = c25 IP = 192.168.1.35 MAC = B8:27:EB:BE:C4:94
Tatsächliches Ergebnis	Nmap scan report for c25.home (192.168.1.35) MAC Address: B8:27:EB:BE:C4:94 (Raspberry Pi Foundation) Nmap done: 1 IP address (1 host up) scanned in 0.09 seconds
Tester	Christoph Amrein
Datum des Tests	22.05.2018
Testergebnis (Fehlerklasse)	1 - Fehlerfrei, die Erwartungen sind erfüllt.
Fehlerbeschreibung	

Tabelle 119: K-027 Protokoll

Test-ID:	K-028
Testobjekt	Compute Node c26
Testschritte	<ul style="list-style-type: none"> - Management Node starten (Strom anschliessen). - 3 Minuten warten. - Auf dem Testclient über Putty oder Shell mit dem Befehl "nmap -sn 192.168.1.36" eingeben. - Prüfen, ob die Zuweisung gemäss Hostnamenkonzept richtig ist.
Erwartetes Ergebnis	Hostname = c26 IP = 192.168.1.36 MAC = B8:27:EB:FB:FF:57
Tatsächliches Ergebnis	Nmap scan report for c26.home (192.168.1.36) MAC Address: B8:27:EB:FB:FF:57 (Raspberry Pi Foundation) Nmap done: 1 IP address (1 host up) scanned in 0.09 seconds
Tester	Christoph Amrein
Datum des Tests	22.05.2018
Testergebnis (Fehlerklasse)	1 - Fehlerfrei, die Erwartungen sind erfüllt.
Fehlerbeschreibung	

Tabelle 120: K-028 Protokoll

G Testprotokoll

Test-ID:	K-029
Testobjekt	Compute Node c27
Testschritte	<ul style="list-style-type: none"> - Management Node starten (Strom anschliessen). - 3 Minuten warten. - Auf dem Testclient über Putty oder Shell mit dem Befehl "nmap -sn 192.168.1.37" eingeben. - Prüfen, ob die Zuweisung gemäss Hostnamenkonzept richtig ist.
Erwartetes Ergebnis	Hostname = c27 IP = 192.168.1.37 MAC = B8:27:EB:4E:EC:CE
Tatsächliches Ergebnis	Nmap scan report for c27.home (192.168.1.37) MAC Address: B8:27:EB:4E:EC:CE (Raspberry Pi Foundation) Nmap done: 1 IP address (1 host up) scanned in 0.10 seconds
Tester	Christoph Amrein
Datum des Tests	22.05.2018
Testergebnis (Fehlerklasse)	1 - Fehlerfrei, die Erwartungen sind erfüllt.
Fehlerbeschreibung	

Tabelle 121: K-029 Protokoll

Test-ID:	K-030
Testobjekt	Compute Node c28
Testschritte	<ul style="list-style-type: none"> - Management Node starten (Strom anschliessen). - 3 Minuten warten. - Auf dem Testclient über Putty oder Shell mit dem Befehl "nmap -sn 192.168.1.38" eingeben. - Prüfen, ob die Zuweisung gemäss Hostnamenkonzept richtig ist.
Erwartetes Ergebnis	Hostname = c28 IP = 192.168.1.38 MAC = B8:27:EB:43:1C:35
Tatsächliches Ergebnis	Nmap scan report for c28.home (192.168.1.38) MAC Address: B8:27:EB:43:1C:35 (Raspberry Pi Foundation) Nmap done: 1 IP address (1 host up) scanned in 0.09 seconds
Tester	Christoph Amrein
Datum des Tests	22.05.2018
Testergebnis (Fehlerklasse)	1 - Fehlerfrei, die Erwartungen sind erfüllt.
Fehlerbeschreibung	

Tabelle 122: K-030 Protokoll

G Testprotokoll

Test-ID:	K-031
Testobjekt	Compute Node c29
Testschritte	<ul style="list-style-type: none"> - Management Node starten (Strom anschliessen). - 3 Minuten warten. - Auf dem Testclient über Putty oder Shell mit dem Befehl "nmap -sn 192.168.1.39" eingeben. - Prüfen, ob die Zuweisung gemäss Hostnamenkonzept richtig ist.
Erwartetes Ergebnis	Hostname = c29 IP = 192.168.1.39 MAC = B8:27:EB:DC:74:5F
Tatsächliches Ergebnis	Nmap scan report for c29.home (192.168.1.39) MAC Address: B8:27:EB:DC:74:5F (Raspberry Pi Foundation) Nmap done: 1 IP address (1 host up) scanned in 0.09 seconds
Tester	Christoph Amrein
Datum des Tests	22.05.2018
Testergebnis (Fehlerklasse)	1 - Fehlerfrei, die Erwartungen sind erfüllt.
Fehlerbeschreibung	

Tabelle 123: K-031 Protokoll

Test-ID:	K-032
Testobjekt	Compute Node c30
Testschritte	<ul style="list-style-type: none"> - Management Node starten (Strom anschliessen). - 3 Minuten warten. - Auf dem Testclient über Putty oder Shell mit dem Befehl "nmap -sn 192.168.1.40" eingeben. - Prüfen, ob die Zuweisung gemäss Hostnamenkonzept richtig ist.
Erwartetes Ergebnis	Hostname = c30 IP = 192.168.1.40 MAC = B8:27:EB:D1:DE:2F
Tatsächliches Ergebnis	Nmap scan report for c30.home (192.168.1.40) MAC Address: B8:27:EB:D1:DE:2F (Raspberry Pi Foundation) Nmap done: 1 IP address (1 host up) scanned in 0.09 seconds
Tester	Christoph Amrein
Datum des Tests	22.05.2018
Testergebnis (Fehlerklasse)	1 - Fehlerfrei, die Erwartungen sind erfüllt.
Fehlerbeschreibung	

Tabelle 124: K-032 Protokoll

G Testprotokoll

Test-ID:	K-033
Testobjekt	Compute Node c31
Testschritte	<ul style="list-style-type: none"> - Management Node starten (Strom anschliessen). - 3 Minuten warten. - Auf dem Testclient über Putty oder Shell mit dem Befehl "nmap -sn 192.168.1.41" eingeben. - Prüfen, ob die Zuweisung gemäss Hostnamenkonzept richtig ist.
Erwartetes Ergebnis	Hostname = c31 IP = 192.168.1.41 MAC = B8:27:EB:5E:90:34
Tatsächliches Ergebnis	Nmap scan report for c31.home (192.168.1.41) MAC Address: B8:27:EB:5E:90:34 (Raspberry Pi Foundation) Nmap done: 1 IP address (1 host up) scanned in 0.10 seconds
Tester	Christoph Amrein
Datum des Tests	22.05.2018
Testergebnis (Fehlerklasse)	1 - Fehlerfrei, die Erwartungen sind erfüllt.
Fehlerbeschreibung	

Tabelle 125: K-033 Protokoll

Test-ID:	K-034
Testobjekt	Compute Node c32
Testschritte	<ul style="list-style-type: none"> - Management Node starten (Strom anschliessen). - 3 Minuten warten. - Auf dem Testclient über Putty oder Shell mit dem Befehl "nmap -sn 192.168.1.42" eingeben. - Prüfen, ob die Zuweisung gemäss Hostnamenkonzept richtig ist.
Erwartetes Ergebnis	Hostname = c32 IP = 192.168.1.42 MAC = B8:27:EB:DE:80:24
Tatsächliches Ergebnis	Nmap scan report for c32.home (192.168.1.42) MAC Address: B8:27:EB:DE:80:24 (Raspberry Pi Foundation) Nmap done: 1 IP address (1 host up) scanned in 0.09 seconds
Tester	Christoph Amrein
Datum des Tests	22.05.2018
Testergebnis (Fehlerklasse)	1 - Fehlerfrei, die Erwartungen sind erfüllt.
Fehlerbeschreibung	

Tabelle 126: K-034 Protokoll

G Testprotokoll

Test-ID:	K-035
Testobjekt	Compute Node c33
Testschritte	<ul style="list-style-type: none"> - Management Node starten (Strom anschliessen). - 3 Minuten warten. - Auf dem Testclient über Putty oder Shell mit dem Befehl "nmap -sn 192.168.1.43" eingeben. - Prüfen, ob die Zuweisung gemäss Hostnamenkonzept richtig ist.
Erwartetes Ergebnis	Hostname = c33 IP = 192.168.1.43 MAC = B8:27:EB:A4:79:6F
Tatsächliches Ergebnis	Nmap scan report for c33.home (192.168.1.43) MAC Address: B8:27:EB:A4:79:6F (Raspberry Pi Foundation) Nmap done: 1 IP address (1 host up) scanned in 0.09 seconds
Tester	Christoph Amrein
Datum des Tests	22.05.2018
Testergebnis (Fehlerklasse)	1 - Fehlerfrei, die Erwartungen sind erfüllt.
Fehlerbeschreibung	

Tabelle 127: K-035 Protokoll

Test-ID:	K-036
Testobjekt	Compute Node c34
Testschritte	<ul style="list-style-type: none"> - Management Node starten (Strom anschliessen). - 3 Minuten warten. - Auf dem Testclient über Putty oder Shell mit dem Befehl "nmap -sn 192.168.1.44" eingeben. - Prüfen, ob die Zuweisung gemäss Hostnamenkonzept richtig ist.
Erwartetes Ergebnis	Hostname = c34 IP = 192.168.1.44 MAC = B8:27:EB:0A:4D:C7
Tatsächliches Ergebnis	Nmap scan report for c34.home (192.168.1.44) MAC Address: B8:27:EB:0A:4D:C7 (Raspberry Pi Foundation) Nmap done: 1 IP address (1 host up) scanned in 0.09 seconds
Tester	Christoph Amrein
Datum des Tests	22.05.2018
Testergebnis (Fehlerklasse)	1 - Fehlerfrei, die Erwartungen sind erfüllt.
Fehlerbeschreibung	

Tabelle 128: K-036 Protokoll

G Testprotokoll

Test-ID:	K-037
Testobjekt	Compute Node c35
Testschritte	<ul style="list-style-type: none"> - Management Node starten (Strom anschliessen). - 3 Minuten warten. - Auf dem Testclient über Putty oder Shell mit dem Befehl "nmap -sn 192.168.1.45" eingeben. - Prüfen, ob die Zuweisung gemäss Hostnamenkonzept richtig ist.
Erwartetes Ergebnis	Hostname = c35 IP = 192.168.1.45 MAC = B8:27:EB:5C:53:5F
Tatsächliches Ergebnis	Note: Host seems down. If it is really up, but blocking our ping probes, try -Pn Nmap done: 1 IP address (0 hosts up) scanned in 0.45 seconds
Tester	Christoph Amrein
Datum des Tests	22.05.2018
Testergebnis (Fehlerklasse)	3 - Das Problem sollte innerhalb 6 Monaten behoben werden.
Fehlerbeschreibung	Der Compute Node ist defekt und kann nicht mehr in Betrieb genommen werden.

Tabelle 129: K-037 Protokoll

Test-ID:	K-038
Testobjekt	Compute Node c36
Testschritte	<ul style="list-style-type: none"> - Management Node starten (Strom anschliessen). - 3 Minuten warten. - Auf dem Testclient über Putty oder Shell mit dem Befehl "nmap -sn 192.168.1.46" eingeben. - Prüfen, ob die Zuweisung gemäss Hostnamenkonzept richtig ist.
Erwartetes Ergebnis	Hostname = c36 IP = 192.168.1.46 MAC = B8:27:EB:F7:AF:C2
Tatsächliches Ergebnis	Nmap scan report for c36.home (192.168.1.46) MAC Address: B8:27:EB:F7:AF:C2 (Raspberry Pi Foundation) Nmap done: 1 IP address (1 host up) scanned in 0.09 seconds
Tester	Christoph Amrein
Datum des Tests	22.05.2018
Testergebnis (Fehlerklasse)	1 - Fehlerfrei, die Erwartungen sind erfüllt.
Fehlerbeschreibung	

Tabelle 130: K-038 Protokoll

G Testprotokoll

Test-ID:	K-039
Testobjekt	Compute Node c37
Testschritte	<ul style="list-style-type: none"> - Management Node starten (Strom anschliessen). - 3 Minuten warten. - Auf dem Testclient über Putty oder Shell mit dem Befehl "nmap -sn 192.168.1.47" eingeben. - Prüfen, ob die Zuweisung gemäss Hostnamenkonzept richtig ist.
Erwartetes Ergebnis	Hostname = c37 IP = 192.168.1.47 MAC = B8:27:EB:CE:BA:ED
Tatsächliches Ergebnis	Nmap scan report for c37.home (192.168.1.47) MAC Address: B8:27:EB:CE:BA:ED (Raspberry Pi Foundation) Nmap done: 1 IP address (1 host up) scanned in 0.09 seconds
Tester	Christoph Amrein
Datum des Tests	22.05.2018
Testergebnis (Fehlerklasse)	1 - Fehlerfrei, die Erwartungen sind erfüllt.
Fehlerbeschreibung	

Tabelle 131: K-039 Protokoll

Test-ID:	K-040
Testobjekt	Compute Node c38
Testschritte	<ul style="list-style-type: none"> - Management Node starten (Strom anschliessen). - 3 Minuten warten. - Auf dem Testclient über Putty oder Shell mit dem Befehl "nmap -sn 192.168.1.48" eingeben. - Prüfen, ob die Zuweisung gemäss Hostnamenkonzept richtig ist.
Erwartetes Ergebnis	Hostname = c38 IP = 192.168.1.48 MAC = B8:27:EB:59:38:3C
Tatsächliches Ergebnis	Nmap scan report for c38.home (192.168.1.48) MAC Address: B8:27:EB:59:38:3C (Raspberry Pi Foundation) Nmap done: 1 IP address (1 host up) scanned in 0.09 seconds
Tester	Christoph Amrein
Datum des Tests	22.05.2018
Testergebnis (Fehlerklasse)	1 - Fehlerfrei, die Erwartungen sind erfüllt.
Fehlerbeschreibung	

Tabelle 132: K-040 Protokoll

G Testprotokoll

Test-ID:	K-041
Testobjekt	Compute Node c39
Testschritte	<ul style="list-style-type: none"> - Management Node starten (Strom anschliessen). - 3 Minuten warten. - Auf dem Testclient über Putty oder Shell mit dem Befehl "nmap -sn 192.168.1.49" eingeben. - Prüfen, ob die Zuweisung gemäss Hostnamenkonzept richtig ist.
Erwartetes Ergebnis	Hostname = c39 IP = 192.168.1.49 MAC = B8:27:EB:99:BB:8E
Tatsächliches Ergebnis	Nmap scan report for c39.home (192.168.1.49) MAC Address: B8:27:EB:99:BB:8E (Raspberry Pi Foundation) Nmap done: 1 IP address (1 host up) scanned in 0.09 seconds
Tester	Christoph Amrein
Datum des Tests	22.05.2018
Testergebnis (Fehlerklasse)	1 - Fehlerfrei, die Erwartungen sind erfüllt.
Fehlerbeschreibung	

Tabelle 133: K-041 Protokoll

Test-ID:	K-042
Testobjekt	Compute Node c40
Testschritte	<ul style="list-style-type: none"> - Management Node starten (Strom anschliessen). - 3 Minuten warten. - Auf dem Testclient über Putty oder Shell mit dem Befehl "nmap -sn 192.168.1.50" eingeben. - Prüfen, ob die Zuweisung gemäss Hostnamenkonzept richtig ist.
Erwartetes Ergebnis	Hostname = c40 IP = 192.168.1.50 MAC = B8:27:EB:8F:7A:0D
Tatsächliches Ergebnis	Nmap scan report for c40.home (192.168.1.50) MAC Address: B8:27:EB:8F:7A:0D (Raspberry Pi Foundation) Nmap done: 1 IP address (1 host up) scanned in 0.10 seconds
Tester	Christoph Amrein
Datum des Tests	22.05.2018
Testergebnis (Fehlerklasse)	1 - Fehlerfrei, die Erwartungen sind erfüllt.
Fehlerbeschreibung	

Tabelle 134: K-042 Protokoll

G Testprotokoll

Test-ID:	K-043
Testobjekt	Compute Node c41
Testschritte	<ul style="list-style-type: none"> - Management Node starten (Strom anschliessen). - 3 Minuten warten. - Auf dem Testclient über Putty oder Shell mit dem Befehl "nmap -sn 192.168.1.51" eingeben. - Prüfen, ob die Zuweisung gemäss Hostnamenkonzept richtig ist.
Erwartetes Ergebnis	Hostname = c41 IP = 192.168.1.51 MAC = B8:27:EB:DE:C9:69
Tatsächliches Ergebnis	Nmap scan report for c41.home (192.168.1.51) MAC Address: B8:27:EB:DE:C9:69 (Raspberry Pi Foundation) Nmap done: 1 IP address (1 host up) scanned in 0.09 seconds
Tester	Christoph Amrein
Datum des Tests	22.05.2018
Testergebnis (Fehlerklasse)	1 - Fehlerfrei, die Erwartungen sind erfüllt.
Fehlerbeschreibung	

Tabelle 135: K-043 Protokoll

Test-ID:	K-044
Testobjekt	Compute Node c42
Testschritte	<ul style="list-style-type: none"> - Management Node starten (Strom anschliessen). - 3 Minuten warten. - Auf dem Testclient über Putty oder Shell mit dem Befehl "nmap -sn 192.168.1.52" eingeben. - Prüfen, ob die Zuweisung gemäss Hostnamenkonzept richtig ist.
Erwartetes Ergebnis	Hostname = c42 IP = 192.168.1.52 MAC = B8:27:EB:7E:6F:48
Tatsächliches Ergebnis	Nmap scan report for c42.home (192.168.1.52) MAC Address: B8:27:EB:7E:6F:48 (Raspberry Pi Foundation) Nmap done: 1 IP address (1 host up) scanned in 0.09 seconds
Tester	Christoph Amrein
Datum des Tests	22.05.2018
Testergebnis (Fehlerklasse)	1 - Fehlerfrei, die Erwartungen sind erfüllt.
Fehlerbeschreibung	

Tabelle 136: K-044 Protokoll

G Testprotokoll

Test-ID:	K-045
Testobjekt	Compute Node c43
Testschritte	<ul style="list-style-type: none"> - Management Node starten (Strom anschliessen). - 3 Minuten warten. - Auf dem Testclient über Putty oder Shell mit dem Befehl "nmap -sn 192.168.1.53" eingeben. - Prüfen, ob die Zuweisung gemäss Hostnamenkonzept richtig ist.
Erwartetes Ergebnis	Hostname = c43 IP = 192.168.1.53 MAC = B8:27:EB:5D:DD:FE
Tatsächliches Ergebnis	nmap scan report for c43.home (192.168.1.53) MAC Address: B8:27:EB:5D:DD:FE (Raspberry Pi Foundation) Nmap done: 1 IP address (1 host up) scanned in 0.09 seconds
Tester	Christoph Amrein
Datum des Tests	22.05.2018
Testergebnis (Fehlerklasse)	1 - Fehlerfrei, die Erwartungen sind erfüllt.
Fehlerbeschreibung	

Tabelle 137: K-045 Protokoll

Test-ID:	K-046
Testobjekt	Compute Node c44
Testschritte	<ul style="list-style-type: none"> - Management Node starten (Strom anschliessen). - 3 Minuten warten. - Auf dem Testclient über Putty oder Shell mit dem Befehl "nmap -sn 192.168.1.54" eingeben. - Prüfen, ob die Zuweisung gemäss Hostnamenkonzept richtig ist.
Erwartetes Ergebnis	Hostname = c44 IP = 192.168.1.54 MAC = B8:27:EB:A6:6D:4D
Tatsächliches Ergebnis	Nmap scan report for c44.home (192.168.1.54) MAC Address: B8:27:EB:A6:6D:4D (Raspberry Pi Foundation) Nmap done: 1 IP address (1 host up) scanned in 0.10 seconds
Tester	Christoph Amrein
Datum des Tests	22.05.2018
Testergebnis (Fehlerklasse)	1 - Fehlerfrei, die Erwartungen sind erfüllt.
Fehlerbeschreibung	

Tabelle 138: K-046 Protokoll

G Testprotokoll

Test-ID:	K-047
Testobjekt	Compute Node c45
Testschritte	<ul style="list-style-type: none"> - Management Node starten (Strom anschliessen). - 3 Minuten warten. - Auf dem Testclient über Putty oder Shell mit dem Befehl "nmap -sn 192.168.1.55" eingeben. - Prüfen, ob die Zuweisung gemäss Hostnamenkonzept richtig ist.
Erwartetes Ergebnis	Hostname = c45 IP = 192.168.1.55 MAC = B8:27:EB:0C:63:10
Tatsächliches Ergebnis	Nmap scan report for c45.home (192.168.1.55) MAC Address: B8:27:EB:0C:63:10 (Raspberry Pi Foundation) Nmap done: 1 IP address (1 host up) scanned in 0.09 seconds
Tester	Christoph Amrein
Datum des Tests	22.05.2018
Testergebnis (Fehlerklasse)	1 - Fehlerfrei, die Erwartungen sind erfüllt.
Fehlerbeschreibung	

Tabelle 139: K-047 Protokoll

Test-ID:	K-048
Testobjekt	NAS
Testschritte	- Den Befehl "pdsh -w c[1-45] ping -c 1 google.de" auf dem Management Node eingeben.
Erwartetes Ergebnis	Der Rückgabewert muss pro Compute Node die Ausgabe "1 packets transmitted, 1 received, 0% packet loss"beinhalten.
Tatsächliches Ergebnis	PING 192.168.1.129 (192.168.1.129) 56(84) bytes of data. 64 bytes from 192.168.1.129: icmp_seq=1 ttl=64 time=0.539 ms 64 bytes from 192.168.1.129: icmp_seq=2 ttl=64 time=0.434 ms — 192.168.1.129 ping statistics — 2 packets transmitted, 2 received, 0% packet loss, time 1001ms rtt min/avg/max/mdev = 0.434/0.486/0.539/0.056 ms
Tester	Christoph Amrein
Datum des Tests	22.05.2018
Testergebnis (Fehlerklasse)	1 - Fehlerfrei, die Erwartungen sind erfüllt.
Fehlerbeschreibung	

Tabelle 140: K-048 Protokoll

G.2 Integrationstests

Test-ID:	I-001
Testobjekt	Compute Nodes
Testschritte	- Den Befehl “pdsh -w c[1-45] ping -c 1 google.de“ auf dem Management Node eingeben.
Erwartetes Ergebnis	Die Ausgabe des Befehls muss einen grösseren Wert als “1“zurückgeben.
Tatsächliches Ergebnis	Die folgende Auflistung wurde aus Platzgründen gekürzt: — google.de ping statistics — c1: 1 packets transmitted, 0 received, 100% packet loss, time 0ms c2: 1 packets transmitted, 0 received, 100% packet loss, time 0ms c3: 1 packets transmitted, 0 received, 100% packet loss, time 0ms c4: 1 packets transmitted, 0 received, 100% packet loss, time 0ms c5: 1 packets transmitted, 0 received, 100% packet loss, time 0ms c6: 1 packets transmitted, 0 received, 100% packet loss, time 0ms c7: 1 packets transmitted, 0 received, 100% packet loss, time 0ms c8: 1 packets transmitted, 0 received, 100% packet loss, time 0ms c9: 1 packets transmitted, 0 received, 100% packet loss, time 0ms c10: 1 packets transmitted, 0 received, 100% packet loss, time 0ms c11: 1 packets transmitted, 0 received, 100% packet loss, time 0ms c12: 1 packets transmitted, 0 received, 100% packet loss, time 0ms c13: 1 packets transmitted, 0 received, 100% packet loss, time 0ms

Tatsächliches Ergebnis	c14: 1 packets transmitted, 0 received, 100% packet loss, time 0ms c15: 1 packets transmitted, 0 received, 100% packet loss, time 0ms c16: 1 packets transmitted, 0 received, 100% packet loss, time 0ms c17: 1 packets transmitted, 0 received, 100% packet loss, time 0ms c18: 1 packets transmitted, 0 received, 100% packet loss, time 0ms c19: 1 packets transmitted, 0 received, 100% packet loss, time 0ms c20: 1 packets transmitted, 0 received, 100% packet loss, time 0ms c21: 1 packets transmitted, 0 received, 100% packet loss, time 0ms c22: 1 packets transmitted, 0 received, 100% packet loss, time 0ms c23: 1 packets transmitted, 0 received, 100% packet loss, time 0ms c24: 1 packets transmitted, 0 received, 100% packet loss, time 0ms c25: 1 packets transmitted, 0 received, 100% packet loss, time 0ms c26: 1 packets transmitted, 0 received, 100% packet loss, time 0ms c27: 1 packets transmitted, 0 received, 100% packet loss, time 0ms c28: 1 packets transmitted, 0 received, 100% packet loss, time 0ms c29: 1 packets transmitted, 0 received, 100% packet loss, time 0ms c30: 1 packets transmitted, 0 received, 100% packet loss, time 0ms c31: 1 packets transmitted, 0 received, 100% packet loss, time 0ms c32: 1 packets transmitted, 0 received, 100% packet loss, time 0ms c33: 1 packets transmitted, 0 received, 100% packet loss, time 0ms c34: 1 packets transmitted, 0 received, 100% packet loss, time 0ms c35: ssh: connect to host c35 port 22: No route to host c36: 1 packets transmitted, 0 received, 100% packet loss, time 0ms c37: 1 packets transmitted, 0 received, 100% packet loss, time 0ms c38: 1 packets transmitted, 0 received, 100% packet loss, time 0ms c39: 1 packets transmitted, 0 received, 100% packet loss, time 0ms c40: 1 packets transmitted, 0 received, 100% packet loss, time 0ms c41: 1 packets transmitted, 0 received, 100% packet loss, time 0ms c42: 1 packets transmitted, 0 received, 100% packet loss, time 0ms c43: 1 packets transmitted, 0 received, 100% packet loss, time 0ms c44: 1 packets transmitted, 0 received, 100% packet loss, time 0ms c45: 1 packets transmitted, 0 received, 100% packet loss, time 0ms
Tester	Christoph Amrein
Datum des Tests	22.05.2018
Testergebnis (Fehlerklasse)	1 - Fehlerfrei, die Erwartungen sind erfüllt.
Fehlerbeschreibung	Der Compute Node c35 wurde bereits bei den Komponententests als Fehlerhaft erkannt. Deshalb ist das Ergebnis wie erwartet.

Tabelle 141: I-001 Protokoll

Test-ID:	I-002
Testobjekt	NAS / Mountpoint / Management Node
Testschritte	<ul style="list-style-type: none">- Management Node starten.- auf dem Management Node anmelden.- Den Befehl "ls -l /media/nebula_data/ wc -l " eingeben.
Erwartetes Ergebnis	Die Ausgabe des Befehls muss einen grösseren Wert als "1" zurückgeben.
Tatsächliches Ergebnis	17
Tester	Christoph Amrein
Datum des Tests	22.05.2018
Testergebnis (Fehlerklasse)	1 - Fehlerfrei, die Erwartungen sind erfüllt.
Fehlerbeschreibung	

Tabelle 142: I-002 Protokoll

G Testprotokoll

Test-ID:	I-003
Testobjekt	NAS / Mountpoint / Compute Nodes
Testschritte	<ul style="list-style-type: none"> - Compute Nodes starten. - auf dem Management Node anmelden. - Den Befehl " pdsh -w c[1-45] ls -l /media/nebula_data/ wc -l" eingeben.
Erwartetes Ergebnis	Die Ausgabe des Befehls muss jeweils einen grösseren Wert als "1" zurückgeben.
Tatsächliches Ergebnis	<p>Alle Compute Nodes haben folgendes ausgegeben. Die Ausgabe ist gekürzt.</p> <pre>c1: 17 c2: 17 c3: 17 c4: 17 c5: 17 c6: 17 c7: 17 .. c35: ssh: connect to host c35 port 22: No route to host pdsh@nebula: c35: ssh exited with exit code 255 c36: 17 .. c43: 17 c44: 17 c45: 17</pre>
Tester	Christoph Amrein
Datum des Tests	22.05.2018
Testergebnis (Fehlerklasse)	1 - Fehlerfrei, die Erwartungen sind erfüllt.
Fehlerbeschreibung	Der Compute Node c35 wurde bereits bei den Komponententests als Fehlerhaft erkannt. Deshalb ist das Ergebnis wie erwartet.

Tabelle 143: I-003 Protokoll

G.3 Systemtests

Test-ID:	S-001
Testobjekt	Compute Nodes / CPU Last
Testschritte	<p>Die Tests finden vom Management Node aus statt.</p> <ul style="list-style-type: none"> - Den Befehl "pdsh -w c[1-45] uptime" eingeben. - Alternativ kann man sich auf jeden Compute Node anmelden und den Befehl "top" eingeben.
Erwartetes Ergebnis	<p>Die Loadaverage-Werte überschreiten jeweils den Wert 1, die Ausgabe sollte in etwa so aussehen:</p> <p>load average: 4.18, 2.44, 1.16</p>
Tatsächliches Ergebnis	<p>c1: 16:42:23 up 1:02, 0 users, load average: 4.26, 4.41, 3.29</p> <p>c2: 16:42:23 up 1:29, 0 users, load average: 4.25, 4.26, 3.20</p> <p>c3: 16:42:23 up 1:02, 0 users, load average: 4.38, 4.32, 3.25</p> <p>c4: 16:42:23 up 1:29, 0 users, load average: 4.50, 4.35, 3.25</p> <p>c5: 16:42:23 up 1:02, 0 users, load average: 4.42, 4.30, 3.22</p> <p>c6: 16:42:23 up 1:29, 0 users, load average: 4.26, 4.38, 3.25</p> <p>c7: 16:42:23 up 1:29, 0 users, load average: 5.01, 4.46, 3.25</p> <p>c8: 16:42:23 up 1:02, 0 users, load average: 4.54, 4.42, 3.29</p> <p>c9: 16:42:24 up 1:02, 0 users, load average: 4.68, 4.38, 3.23</p> <p>c10: 16:42:23 up 1:29, 0 users, load average: 4.84, 4.54, 3.31</p> <p>c11: 16:42:24 up 1:02, 0 users, load average: 4.36, 4.38, 3.23</p> <p>c12: 16:42:23 up 1:02, 0 users, load average: 4.30, 4.22, 3.18</p> <p>c13: 16:42:23 up 1:29, 0 users, load average: 4.42, 4.36, 3.28</p> <p>c14: 16:42:23 up 1:29, 0 users, load average: 4.38, 4.28, 3.18</p> <p>c15: 16:42:23 up 1:29, 0 users, load average: 4.61, 4.53, 3.31</p> <p>c16: 16:42:23 up 1:02, 0 users, load average: 4.43, 4.39, 3.22</p> <p>c17: 16:42:23 up 1:29, 0 users, load average: 4.29, 4.28, 3.21</p> <p>c18: 16:42:23 up 1:29, 0 users, load average: 4.47, 4.34, 3.24</p> <p>c19: 16:42:23 up 1:29, 0 users, load average: 4.69, 4.42, 3.26</p> <p>c20: 16:42:23 up 1:29, 0 users, load average: 4.71, 4.42, 3.26</p> <p>c21: 16:42:23 up 1:29, 0 users, load average: 4.91, 4.46, 3.22</p> <p>c22: 16:42:23 up 1:29, 0 users, load average: 4.64, 4.34, 3.23</p> <p>c23: 16:42:23 up 1:29, 0 users, load average: 4.47, 4.32, 3.25</p>

Tatsächliches Ergebnis	c24: 16:42:23 up 1:29, 0 users, load average: 4.36, 4.30, 3.23 c25: 16:42:23 up 1:02, 0 users, load average: 4.34, 4.31, 3.24 c26: 16:42:23 up 1:29, 0 users, load average: 4.54, 4.43, 3.26 c27: 16:42:23 up 1:02, 0 users, load average: 4.82, 4.45, 3.27 c28: 16:42:23 up 1:02, 0 users, load average: 4.48, 4.38, 3.24 c29: 16:42:23 up 1:29, 0 users, load average: 4.28, 4.32, 3.22 c30: 16:42:23 up 1:29, 0 users, load average: 4.37, 4.31, 3.24 c31: 16:42:23 up 1:02, 0 users, load average: 4.48, 4.36, 3.24 c32: 16:42:23 up 1:29, 0 users, load average: 4.27, 4.26, 3.19 c33: 16:42:27 up 1:02, 0 users, load average: 4.62, 4.42, 3.30 c34: 16:42:28 up 1:02, 0 users, load average: 4.52, 4.27, 3.16 c35: ssh: connect to host c35 port 22: No route to host c36: 16:42:23 up 1:02, 0 users, load average: 4.82, 4.45, 3.27 c37: 16:42:23 up 1:02, 0 users, load average: 4.48, 4.38, 3.24 c38: 16:42:28 up 1:29, 0 users, load average: 4.28, 4.18, 3.12 c39: 16:42:28 up 1:02, 0 users, load average: 4.28, 4.40, 3.29 c40: 16:42:28 up 1:02, 0 users, load average: 4.58, 4.51, 3.34 c41: 16:42:28 up 20 min, 0 users, load average: 4.70, 4.40, 3.20 c42: 16:42:28 up 1:02, 0 users, load average: 4.55, 4.38, 3.26 c43: 16:42:28 up 1:29, 0 users, load average: 4.82, 4.49, 3.30 c44: 16:42:23 up 1:29, 0 users, load average: 4.47, 4.34, 3.24 c45: 16:42:23 up 1:29, 0 users, load average: 4.82, 4.45, 3.27
Tester	Christoph Amrein
Datum des Tests	22.05.2018
Testergebnis (Fehlerklasse)	1 - Fehlerfrei, die Erwartungen sind erfüllt.
Fehlerbeschreibung	Der Compute Node c35 wurde bereits bei den Komponententests als Fehlerhaft erkannt. Deshalb ist das Ergebnis wie erwartet.

Tabelle 144: S-001 Protokoll

G Testprotokoll

Test-ID:	S-002
Testobjekt	Nodes / Nagios Monitoring
Testschritte	<ul style="list-style-type: none"> - http://nebula/nagios aufrufen. - Bei Nagios anmelden. - Auf den Reiter "Hosts" klicken.
Erwartetes Ergebnis	Es werden alle Nodes aufgelistet. Unabhängig des Status ob der Node in Betrieb ist oder nicht.
Tatsächliches Ergebnis	Alle Nodes sind aufgelistet und haben den Status UP, mit Ausnahme des Compute Nodes c35.
Tester	Christoph Amrein
Datum des Tests	22.05.2018
Testergebnis (Fehlerklasse)	1 - Fehlerfrei, die Erwartungen sind erfüllt.
Fehlerbeschreibung	Der Compute Node c35 wurde bereits bei den Komponententests als Fehlerhaft erkannt. Deshalb ist das Ergebnis wie erwartet.

Tabelle 145: S-002 Protokoll

Test-ID:	S-003
Testobjekt	Nodes / Nagios Alarmierung
Testschritte	<ul style="list-style-type: none"> - Strom von Node c1 trennen. - 5 Minuten warten. - Email prüfen (christoph.amrein86@gmail.com)
Erwartetes Ergebnis	Es trifft eine Alarmierungs-Email ein.
Tatsächliches Ergebnis	<p>E-Mail ist eingetroffen.</p> <p>***** Nagios *****</p> <p>Notification Type: PROBLEM</p> <p>Host: c1</p> <p>State: DOWN</p> <p>Address: 192.168.1.11</p> <p>Info: CRITICAL - Host Unreachable (192.168.1.11)</p> <p>Date/Time: Sat May 26 17:59:10 CEST 2018</p>
Tester	Christoph Amrein
Datum des Tests	26.05.2018
Testergebnis (Fehlerklasse)	1 - Fehlerfrei, die Erwartungen sind erfüllt.
Fehlerbeschreibung	

Tabelle 146: S-003 Protokoll

G Testprotokoll

Test-ID:	S-004
Testobjekt	Nodes / Ganglia Monitoring
Testschritte	- http://nebula/ganglia aufrufen. - Quelle "Nebula" auswählen.
Erwartetes Ergebnis	Es werden alle Nodes aufgelistet.
Tatsächliches Ergebnis	Alle Nodes sind aufgelistet, jedoch nicht wie gewünscht gemonitort.
Tester	Christoph Amrein
Datum des Tests	26.05.2018
Testergebnis (Fehlerklasse)	3 - Kleiner Mangel, muss innerhalb von 6 Monaten gelöst werden.
Fehlerbeschreibung	Es wird nur jeweils ein Node mit aktuellen Aufzeichnungen dargestellt. Es sollten aber alle Nodes aktive Aufzeichnungen haben.

Tabelle 147: S-004 Protokoll

Test-ID:	S-005
Testobjekt	Nodes / Cluster Jobs
Testschritte	Auf dem Management Node sind folgende Befehle abzusetzen: - "mpicc -O3 /opt/ohpc/pub/examples/mpi/hello.c" - "srun -n 8 -N 2 -pty /bin/bash" - "squeue"
Erwartetes Ergebnis	Der Job wird in der Queue angezeigt und ist 2 zufälligen Compute Nodes zugewiesen.
Tatsächliches Ergebnis	Die Spalten wurden gekürzt. [root@nebula tmp]# squeue JOBID USER ST TIME NODES NODELIST(REASON) 123 root R 6:14 2 c[36-37]
Tester	Christoph Amrein
Datum des Tests	26.05.2018
Testergebnis (Fehlerklasse)	1 - Fehlerfrei, die Erwartungen sind erfüllt.
Fehlerbeschreibung	

Tabelle 148: S-005 Protokoll

G Testprotokoll

Test-ID:	S-006
Testobjekt	Nodes / Schürfen
Testschritte	Auf dem Management Node sind folgende Befehle als root abzusetzen: - “cd /opt/miners/tkinjo“ - “srun -nodes=40-45 -ntasks=40 -cpus-per-task=4 ./cpuminer -a cryptonight -o stratum+tcp://xdn.pool.minergate.com:45620 -u x -p x“
Erwartetes Ergebnis	Es soll direkt in der Shell die Ausgabe des Miners ausgegeben werden.
Tatsächliches Ergebnis	Es erscheint die gewünschte Ausgabe, es sind 40 aktive Nodes zu sehen, welche am Schürfen sind.
Tester	Christoph Amrein
Datum des Tests	26.05.2018
Testergebnis (Fehlerklasse)	1 - Fehlerfrei, die Erwartungen sind erfüllt.
Fehlerbeschreibung	

Tabelle 149: S-006 Protokoll

H Protokoll

Diplomarbeit Mining Cluster

Dokumentenart	Sitzungsprotokoll
Titel	Kick-Off Meeting Mining Cluster
Nummer	1
Autor/-in	Christoph Amrein
Kontaktangaben	christoph.amrein86@gmail.com
Ausgabestelle	Christoph Amrein, Privat
Geltungsbereich	Im Rahmen der Diplomarbeit
Klassifizierung	Nicht klassifiziert
Ausgabedatum	5. Februar 2018
Teilnehmer/-innen	Christoph Amrein CA, Andreas Megert, AM, Rolf Schmutz RS
Sachgeschäfte / Referenten	
Verteiler	Christoph Amrein (christoph.amrein86@gmail.com), Andreas Megert (andreas.megert@gibb.ch), Rolf Schmutz (rolf.schmutz@post.ch)
Entschuldigt	
Sitzungsort	Webergutstrasse 12, 3052 Zollikofen
Sitzungsdatum	05.02.2018

Themen / Notizen	Termin / Betrifft
Arbeitsjournal Rapportierung	Alle 2 Wochen / CA
CA soll alle 2 Wochen an den Verteiler das aktuelle Arbeitsjournal senden	
Arbeitsjournal & Hilfe anfordern	CA
Im Arbeitsjournal soll klar ersichtlich sein, ob Hilfe bei einem Task benötigt wird, oder ob man bei einem Task ansteht. Hilfe sollte dennoch direkt per E-Mail Anfrage angefordert werden.	
Feedback, Antworten, Entscheide	CA
Feedback soll aktiv eingeholt werden. Falls offene Punkte bestehen, dann soll nochmals nachgefragt und nicht selbst entschieden werden. NOGO: z.B. wenn ich innerhalb 2 Wochen keine Antwort erhalte... dann mache ich es so ...	
Projektinitialisierungsauftrag	CA, AM, RS
Der Projektinitialisierungsauftrag wird per E-Mail an den Verteiler versendet (CA) und per Antwort quittiert (AM, RS). Die Quittierung darf bei der Abgabe der Diplomarbeit im Anhang verwendet werden und ersetzt die physische Unterschrift.	
Projektname	08.02.2018 / CA
Es soll ein Projektname definiert werden.	
E-Mail Subject	CA
Durch das Projekt hinweg, soll für die Kommunikation per E-Mail folgender Betreff verwendet werden. Projektname: Inhalt, z.B. Mining Cluster: Arbeitsjournal	
Informationsplattform	08.02.2018 / CA
Es wurde entschieden, dass ein Wiki (Confluence) installiert wird. Alle erarbeiteten Dokumente inklusive Journal werden auf dieser Plattform abgelegt oder erfasst und können von allen beteiligten Personen zu jeder Zeit aufgerufen werden. die Domain wird noch bekannt gegeben.	
Zwischenmeeting	08.02.2018 / CA, AM
Das Zwischenmeeting soll am 05.03.2018 in den Räumen der GIBB stattfinden. AM hat sich dazu bereit erklärt einen Raum zu organisieren. CA soll möglichst rasch den Termin und die Uhrzeit festlegen und danach die Einladungen versenden.	
Konzeptphase	Konzeptphase / CA
CA soll mindestens 7-8 Muss-Kriterien für die Erfüllung des Ziels während der Konzeptphase liefern.	
Testkonzept	Konzeptphase / ALLE

Sitzungsprotokoll 1 Kick-Off Meeting Mining Cluster
D:\Download\KickOff Meeting Protokoll.docx

Ausgabedatum 5. Februar 2018

1 / 2

Die Experten legen viel Wert auf ein gut erstelltes Testkonzept, dabei soll klar vorab definiert werden, welche Komponenten und Funktionen getestet werden sollen.	
--	--

Protokollführer
Christoph Amrein

I Arbeitsjournal

Tag: 1	Datum: 05.02.2018	Aufwand:
Erledigte Arbeit	<ul style="list-style-type: none"> - Es wurde ein Share für die Ablage der entstehenden Dokumente und Arbeiten erstellt. - Projektinitialisierungsauftrag wurde geschrieben. - Kick-Off Meeting geplant und durchgeführt. 	
Aufgetretene Probleme	-	
Tag: 2	Datum: 06.02.2018	Aufwand:
Erledigte Arbeit	<p>Confluence für das Projekt eingerichtet (Wurde am Kick-Off Meeting entschieden).</p> <ul style="list-style-type: none"> - Protokoll des Kick-Off Meetings verfeinert. - Projektinitialisierungsauftrag gemäss Feedback aus Kick-Off Meeting angepasst. - Projektnamen definiert. - Beginn mit der Projektplanung. 	
Aufgetretene Probleme	-	
Tag: 3	Datum: 07.02.2018	Aufwand:
Erledigte Arbeit	<ul style="list-style-type: none"> - Zusammenfassung des Kick-Off Meetings an die Anwesenden versendet. - Administrative Arbeiten (Organisation). 	
Aufgetretene Probleme	-	
Tag: 4	Datum: 08.02.2018	Aufwand:
Erledigte Arbeit	- An der Projektplanung gearbeitet.	
Aufgetretene Probleme	-	
Tag: 5	Datum: 09.02.2018	Aufwand:
Erledigte Arbeit	-	
Aufgetretene Probleme	-	
Tag: 6	Datum: 10.02.2018	Aufwand:
Erledigte Arbeit	-	
Aufgetretene Probleme	-	
Tag: 7	Datum: 11.02.2018	Aufwand:
Erledigte Arbeit	-	
Aufgetretene Probleme	-	
Tag: 8	Datum: 12.02.2018	Aufwand:
Erledigte Arbeit	<ul style="list-style-type: none"> - Terminplan und Projektübersicht im Confluence erstellen. - An der Projektplanung weitergearbeitet. - Beginn mit dem Projektauftrag 	
Aufgetretene Probleme	-	
Tag: 9	Datum: 13.02.2018	Aufwand:

Erledigte Arbeit	- Am Projektauftrag weitergearbeitet.	
Aufgetretene Probleme	-	
Tag: 10	Datum: 14.02.2018	Aufwand:
Erledigte Arbeit	-	
Aufgetretene Probleme	-	
Tag: 11	Datum: 15.02.2018	Aufwand:
Erledigte Arbeit	- Projektauftrag fertiggestellt. - Administrative Arbeiten	
Aufgetretene Probleme	-	
Tag: 12	Datum: 16.02.2018	Aufwand:
Erledigte Arbeit	- Krank.	
Aufgetretene Probleme	-	
Tag: 13	Datum: 17.02.2018	Aufwand:
Erledigte Arbeit	- Krank.	
Aufgetretene Probleme	-	
Tag: 14	Datum: 18.02.2018	Aufwand:
Erledigte Arbeit	- Krank.	
Aufgetretene Probleme	-	
Tag: 15	Datum: 19.02.2018	Aufwand:
Erledigte Arbeit	- Krank.	
Aufgetretene Probleme	-	
Tag: 16	Datum: 20.02.2018	Aufwand:
Erledigte Arbeit	- Krank.	
Aufgetretene Probleme	-	
Tag: 17	Datum: 21.02.2018	Aufwand:
Erledigte Arbeit	- Krank.	
Aufgetretene Probleme	-	
Tag: 18	Datum: 22.02.2018	Aufwand:
Erledigte Arbeit	- Krank.	
Aufgetretene Probleme	-	
Tag: 19	Datum: 23.02.2018	Aufwand:
Erledigte Arbeit	- Krank.	
Aufgetretene Probleme	-	
Tag: 20	Datum: 24.02.2018	Aufwand:
Erledigte Arbeit	- Krank. - Mit der Studie begonnen. - Informationsbeschaffung der Varianten.	
Aufgetretene Probleme	-	
Tag: 21	Datum: 25.02.2018	Aufwand:

Erledigte Arbeit	<ul style="list-style-type: none"> - Studie ausarbeiten. - Informationsbeschaffung. - IST Zustand beschreiben. - Ziele beschreiben. 	
Aufgetretene Probleme	-	
Tag: 22	Datum: 26.02.2018	Aufwand:
Erledigte Arbeit	<ul style="list-style-type: none"> - Studie abschliessen. - Ziele erweitern. - Varianten beschreiben. - Variantenentscheid fällen. - Empfehlung der Variante beschreiben. 	
Aufgetretene Probleme	-	
Tag: 23	Datum: 27.02.2018	Aufwand:
Erledigte Arbeit	- Projektauftrag mit den neuen Erkenntnissen der Studie ergänzen.	
Aufgetretene Probleme	-	
Tag: 24	Datum: 28.02.2018	Aufwand:
Erledigte Arbeit	<ul style="list-style-type: none"> - Projektauftrag fertigstellen. - Studie fertigstellen. - Zwischenmeeting vorbereiten. 	
Aufgetretene Probleme.	Das vorgesehene Zeitkontingent der Studie wurde überschossen.	
Tag: 25	Datum: 01.03.2018	Aufwand:
Erledigte Arbeit	<ul style="list-style-type: none"> - Zwischenmeeting vorbereiten. - Finale Version Projektauftrag. - Finale Version Studie. - Powerpoint Präsentation (Agenda) erstellen. - Projektplan updaten. 	
Aufgetretene Probleme	-	
Tag: 26	Datum: 02.03.2018	Aufwand:
Erledigte Arbeit	-	
Aufgetretene Probleme	-	
Tag: 27	Datum: 03.03.2018	Aufwand:
Erledigte Arbeit	-	
Aufgetretene Probleme	-	
Tag: 28	Datum: 04.03.2018	Aufwand:
Erledigte Arbeit	-	
Aufgetretene Probleme	-	
Tag: 29	Datum: 05.03.2018	Aufwand:
Erledigte Arbeit	-	
Aufgetretene Probleme	-	
Tag: 30	Datum: 06.03.2018	Aufwand:

Erledigte Arbeit	-	
Aufgetretene Probleme	-	
Tag: 31	Datum: 07.03.2018	Aufwand:
Erledigte Arbeit	Erneute Informationsbeschaffung. Es sind Probleme bei der Installation der Variante OpenHPC aufgetreten. Als Zwischenlösung habe ich entschieden einen Laptop der keinen AMD Prozessor verwendet zu benutzen, da dort das Produkt einfach installiert werden kann. Die Computenodes sollen aber weiterhin via Raspberry PI betrieben werden.	
Aufgetretene Probleme	Zuwenige Informationen über OpenHPC gesammelt.	
Tag: 32	Datum: 08.03.2018	Aufwand:
Erledigte Arbeit	Hostnamen und IP's definiert und in Konzept aufgenommen.	
Aufgetretene Probleme	-	
Tag: 33	Datum: 09.03.2018	Aufwand:
Erledigte Arbeit	- Ich habe versucht verschiedene CentOS Images auf die Raspberry PI's zu installieren, keines hat funktioniert. Es wird weiterhin nach einer alternative gesucht.	
Aufgetretene Probleme	CentOS Images können nicht auf Raspberry PI's installiert werden.	
Tag: 34	Datum: 10.03.2018	Aufwand:
Erledigte Arbeit	<ul style="list-style-type: none"> - Neudefinition der Hostnamen. - OpenHPC wurde auf einem Laptop installiert. - Die MAC Adressen der Nodes wurden noch nicht ausgelesen deshalb können die Computenodes noch nicht dem Cluster zugewiesen werden. - Die Warewulf Komponente funktioniert noch nicht, somit kann das Betriebssystem noch nicht auf die Computenodes verteilt werden. 	
Aufgetretene Probleme	WareWulf Komponente blockiert die PXE Boot Installation.	
Tag: 35	Datum: 11.03.2018	Aufwand:
Erledigte Arbeit	- 10 MAC Adressen der RPI's ausgelesen, dafür musste ich eine SD Karte welche ein kompatibles OS für die RPI's besitzt verwenden. Danach habe ich über nmap die MAC Adressen jeweils ausgelesen.	
Aufgetretene Probleme	-	
Tag: 36	Datum: 12.03.2018	Aufwand:
Erledigte Arbeit	<ul style="list-style-type: none"> - Ich habe mit dem physischen Aufbau des Clusters begonnen. - Distanzbolzen mit Raspberry PI's verbunden. - Patchkabel and Raspberry PI's und Switch angeschlossen. - Netzteil via Jumper Kabel an Raspberry PI's angeschlossen. 	

Aufgetretene Probleme	Die Lösung sieht instabil aus.	
Tag: 37	Datum: 13.03.2018	Aufwand:
Erledigte Arbeit	<ul style="list-style-type: none"> - Stromtest des Clusters durchgeführt. 24 Stunden lang den Cluster mit Strom versorgt und beobachtet ob am Schluss Raspberry PI's ausgefallen sind. - Informationen über das Bedienen von OpenHPC eingeholt. 	
Aufgetretene Probleme		
Tag: 38	Datum: 14.03.2018	Aufwand:
Erledigte Arbeit	<ul style="list-style-type: none"> - ein weiterer Versuch PXE Boot mit Warewulf umzusetzen. Dabei bin ich auf Informationen gestossen, dass dies mit den ARMv8 Prozessoren aufgrund der Architektur nicht möglich ist. 	
Aufgetretene Probleme	Warewulf ist nicht mit Raspberry PI's kompatibel.	
Tag: 39	Datum: 15.03.2018	Aufwand:
Erledigte Arbeit	<ul style="list-style-type: none"> - Weitere Versuche PXE Boot mit Warewulf einzurichten, da ich es nicht glauben kann, dass dies nicht gehen soll. 	
Aufgetretene Probleme	Warewulf ist nicht mit Raspberry PI's kompatibel	
Tag: 40	Datum: 16.03.2018	Aufwand:
Erledigte Arbeit	<ul style="list-style-type: none"> - Weitere Versuche PXE Boot mit Warewulf einzurichten, da ich es nicht glauben kann, dass dies nicht gehen soll. 	
Aufgetretene Probleme	Warewulf ist nicht mit Raspberry PI's kompatibel.	
Tag: 41	Datum: 17.03.2018	Aufwand:
Erledigte Arbeit	<ul style="list-style-type: none"> - Weitere Versuche PXE Boot mit Warewulf einzurichten, da ich es nicht glauben kann, dass dies nicht gehen soll. 	
Aufgetretene Probleme	Warewulf ist nicht mit Raspberry PI's kompatibel.	
Tag: 42	Datum: 18.03.2018	Aufwand:
Erledigte Arbeit	<ul style="list-style-type: none"> - Den PXE Boot mit einem NOOBS Betriebssystem aufgesetzt. Dies hat funktioniert. - Dies habe ich versucht in Warewulf zu implementieren. Leider erfolglos. Entschieden den TFTPBOOT / PXE ohne WareWulf auf dem Managementnode einzurichten. 	
Aufgetretene Probleme		
Tag: 43	Datum: 19.03.2018	Aufwand:
Erledigte Arbeit	<ul style="list-style-type: none"> - Installationsscript geschrieben, welches alles automatisch installiert. 	
Aufgetretene Probleme	Es bestehen noch Probleme mit der Installation von CentOS auf den Raspberry PI's.	
Tag: 44	Datum: 20.03.2018	Aufwand:

Erledigte Arbeit	<ul style="list-style-type: none"> - Den Managementnode mehrfach neu aufgesetzt. - Weitere Abklärungen betreffend, PXE, CentOS und Warewulf. 	
Aufgetretene Probleme		
Tag: 45	Datum: 21.03.2018	Aufwand:
Erledigte Arbeit	<ul style="list-style-type: none"> - Ich habe versucht diverse Images auf die SD Karte zu schreiben und die Raspberry PI's zu betreiben. - Die Images stammen von http://mirror.centos.org/altarch/7/isos/aarch64/. 	
Aufgetretene Probleme	<ul style="list-style-type: none"> - Leider hat kein Image funktioniert. - Vermutlich kann der Kernel nicht richtig geladen werden oder ist nicht kompatibel. 	
Tag: 46	Datum: 22.03.2018	Aufwand:
Erledigte Arbeit	<ul style="list-style-type: none"> - Weitere fremde Images versucht zu installieren, z.B. Gentoo 64 Bit für Raspberry PI's, Fedora, usw. 	
Aufgetretene Probleme	-	
Tag: 47	Datum: 23.03.2018	Aufwand:
Erledigte Arbeit	<p>- Ich habe nach einer alternativen 64 Bit Version für RPI's gesucht. Dabei bin ich auf diverse Images gestossen:</p> <p>Fedora https://fedoraproject.org/wiki/Architectures/ARM, die Installation hat einwandfrei funktioniert. (CentOS nahe) Gentoo https://github.com/sakaki-/gentoo-on-rpi3-64bit, die Installation haut auf antrieb funktioniert. Ich bin auf diesen Guide gestossen: https://github.com/umiddelb/aarch64/wiki/Install-CentOS-7-on-your-favourite-ARMv8-ARM64-AArch64-board, durch diesen Guide habe ich den Durchbruch geschafft. Die Installation habe ich mit der Fedora Boot Partition leider nicht geschafft, mit der Gentoo Lösung ging es aber. Ich bin dabei wie folgt vorgegangen: Beim Schreiben des Gentoo Images auf die SD Karte, werden zwei Partitionen erstellt (Boot & File-System). Die Boot Partition habe ich nicht angepasst. Wichtig dabei ist, dass das RPI ein Kernel8.img für ARMv8 in der Boot Partition benötigt. Dies musste ich also stehen lassen. Als zweiten Schritt habe ich aus dem CentOS Repos das Archiv CentOS-7-aarch64-rootfs-7.4.1708.tar.xz heruntergeladen. Darin ist das komplette FileSystem enthalten. Dies habe ich auf der FileSystem Partition mit dem Befehl <code>tar -numeric-owner -xpJf .../CentOS-7-aarch64-rootfs-7.4.1708.tar.xz -C /home/camrein/-Downloads/mnt2</code> niedergeschrieben. (Achtung die Partition wurde nach /home/camrein/Downloads/mnt2 gemountet) Danach hatte ich ein funktionsfaehiges CentOS 64 Bit auf dem Raspberry PI.</p>	

Aufgetretene Probleme	-	
Tag: 48	Datum: 24.03.2018	Aufwand:
Erledigte Arbeit	- Das CentOS nun erfolgreich auf dem RPI betrieben werden kann, habe ich versucht in RPI als Management Node zu verwenden. Bei der Installation des Provisioning Programmes Warewulf bin ich jeweils auf Fehler gestossen. Leider konnte ich in keinem Log inkl Systemlogs keinen Eintrag zum Fehler finden. Das RPI ist jedoch immer wieder eingefroren. Deshalb habe ich versucht einen üblichen PXE Boot mit TFT einzurichten. Dies hat einwandfrei funktioniert. Bei jedem Erfolg habe ich die SD Karte erneut kopiert, so dass beim Aufsetzen des RPI's wieder möglichst rasch mit einem stabilen Ausgangspunkt weitergemacht werden kann.	
Aufgetretene Probleme	- Beim PXE Boot trat ich auf diverse Probleme. Jedoch habe ich durch lesen von mehreren Anleitungen und Guides dies beheben können. Ich hatte nicht alle Dateien von der Boot Partition im entsprechenden tftboot Ordner drin.	
Tag: 49	Datum: 25.03.2018	Aufwand:
Erledigte Arbeit	- Administratives, Dokumente nachführen. Entscheidungen treffen. Ich habe mich entschieden es nochmals mit dem Laptop als Managementnode zu versuchen. Dieser hat einen Intel Prozessor und Warewulf kann ohne Probleme auf dem Managementnode installiert werden. Ich habe mich dazu entschlossen als nächsten Schritt es nochmals zu versuchen das vorhandene tftboot Image in Warewulf zu implementieren. Dies würde die Installation enorm vereinfachen, da die Raspberry PI's direkt über die MAC Adresse eine IP und einen Hostnamen zugewiesen werden.	
Aufgetretene Probleme	-	
Tag: 50	Datum: 26.03.2018	Aufwand:
Erledigte Arbeit	- Zur Erkenntnis genommen, das die MS Office Tools für mich nicht zu gebrauchen sind. Ich habe mich entschieden die Dokumentation mit LaTeX zu schreiben.	
Aufgetretene Probleme	-	
Tag: 51	Datum: 27.03.2018	Aufwand:
Erledigte Arbeit	- Studieren von LaTeX: Ich habe noch nie mit LaTeX gearbeitet, es begeistert mich aber, deshalb habe ich mir Beispiele von Dokumentationen und Befehlen angeschaut.	
Aufgetretene Probleme	-	
Tag: 52	Datum: 28.03.2018	Aufwand:

Erledigte Arbeit	- Versuche mit LaTeX: Ich habe versucht selbst ein LaTeX Dokument zu erstellen und mich mit Freunden darüber unterhalten, dabei habe ich erfahren, dass es bereits sehr gute vordefinierte Templates gibt, welche frei im Internet beziehbar sind.	
Aufgetretene Probleme	-	
Tag: 53	Datum: 29.03.2018	Aufwand:
Erledigte Arbeit	- Ich habe mich für ein LaTeX Template von Macke entschieden und mit der Migration von Word nach LaTeX begonnen.	
Aufgetretene Probleme	-	
Tag: 54	Datum: 30.03.2018	Aufwand:
Erledigte Arbeit	- Ausgangslage und Projektziele nach LaTeX migriert.	
Aufgetretene Probleme	-	
Tag: 55	Datum: 31.03.2018	Aufwand:
Erledigte Arbeit	- Termine und Projektorganisation nach LaTeX migriert.	
Aufgetretene Probleme	-	
Tag: 56	Datum: 01.04.2018	Aufwand:
Erledigte Arbeit	- Überarbeitung der Darstellung der bereits migrierten Inhalte.	
Aufgetretene Probleme	-	
Tag: 57	Datum: 02.04.2018	Aufwand:
Erledigte Arbeit	- Kapitel Ressourcen nach LaTeX migriert.	
Aufgetretene Probleme	-	
Tag: 58	Datum: 03.04.2018	Aufwand:
Erledigte Arbeit	- Überarbeitung bisheriger Dokumentation in LaTeX.	
Aufgetretene Probleme	-	
Tag: 59	Datum: 04.04.2018	Aufwand:
Erledigte Arbeit	- Design Anpassungen des Diplomberichts.	
Aufgetretene Probleme	-	
Tag: 60	Datum: 05.04.2018	Aufwand:
Erledigte Arbeit	-	
Aufgetretene Probleme	-	
Tag: 61	Datum: 06.04.2018	Aufwand:
Erledigte Arbeit	-	
Aufgetretene Probleme	-	
Tag: 62	Datum: 07.04.2018	Aufwand:
Erledigte Arbeit	- Abschliessen der Migration des Projektauftrags (LaTeX). -Informationen über OpenHPC Warewulf und ARMv8 sammeln, da ich noch Probleme mit dem Provisioning des Betriebssystems habe.	
Aufgetretene Probleme	-	

Tag: 62	Datum: 08.04.2018	Aufwand:
Erledigte Arbeit	<ul style="list-style-type: none"> - Studie nach LaTeX migrieren. - Informationen über OpenHPC Warewulf und ARMv8 sammeln, da ich noch Probleme mit dem Provisioning des Betriebssystems habe. 	
Aufgetretene Probleme	-	
Tag: 63	Datum: 09.04.2018	Aufwand:
Erledigte Arbeit	<ul style="list-style-type: none"> - Überarbeitung bisheriger Dokumentation und Konzept Dokumentation migrieren. 	
Aufgetretene Probleme	-	
Tag: 64	Datum: 10.04.2018	Aufwand:
Erledigte Arbeit	<ul style="list-style-type: none"> - Entschieden Warewulf auszulassen und mit dnsmasq und pxe boot fortzufahren. PXE Boot mit Centos eingerichtet, alle Raspberry PI's können gestartet werden und beziehen das Betriebssystem über das Netzwerk. 	
Aufgetretene Probleme	-	
Tag: 65	Datum: 11.04.2018	Aufwand:
Erledigte Arbeit	<ul style="list-style-type: none"> - Nagios einrichten. Es existiert ein Bug mit Centos 7.4. der Ordner /var/run/nagios wird nicht automatisch erstellt. Debuggen bislang erfolglos. Wenn ich in der Konfiguration ein anderes Verzeichnis für die PID Ablage erstelle funktioniert dies noch nicht. Standard Monitoring für alle Nodes eingerichtet. Es müssen aber noch spezifiziertere Überwachungen geschrieben werden. 	
Aufgetretene Probleme	-	
Tag: 66	Datum: 12.04.2018	Aufwand:
Erledigte Arbeit	<ul style="list-style-type: none"> - Installationsskript anpassen und updaten. Ich habe eine Kopie der SD Karte erstellt. Die SD Karte wurde danach gelöscht und ich habe mit dem automatisierten Installationsskript den Cluster wieder versucht zu installieren. Dabei sind folgende Probleme noch vorhanden: - NTP Sync von Master zu Computenodes. - Slurmd startet nicht automatisch auf den Computes. - Nagios PID Fehler. - Ganglia Errors in /var/log/messages (Keine Nodes werden aufgeführt). 	
Aufgetretene Probleme	-	
Tag: 67	Datum: 13.04.2018	Aufwand:
Erledigte Arbeit	<ul style="list-style-type: none"> - Slurm Konfiguration manuell angepasst. Fehler gefunden und behoben. Versucht ersten Job zu erstellen. Jedoch noch erfolglos. 	

Aufgetretene Probleme	- Jobs können mit Slurm noch nicht erstellt werden.	
Tag: 68	Datum: 14.04.2018	Aufwand:
Erledigte Arbeit	- Herausgefunden wie Jobs erstellt werden müssen. Anstatt ein sbatch Script muss mit dem Befehl srun gearbeitet werden. Job erstellt und einen Testlauf vollzogen.	
Aufgetretene Probleme	-	
Tag: 69	Datum: 15.04.2018	Aufwand:
Erledigte Arbeit	- Dokumentation nachgeführt.	
Aufgetretene Probleme	-	
Tag: 70	Datum: 16.04.2018	Aufwand:
Erledigte Arbeit	- Bei einem erneuten Testlauf ist ein Raspberry PI beschädigt worden. Ich wollte es austauschen und habe bemerkt dass der Aufwand für den Austausch zu viel Zeit kostet. Deshalb habe ich nochmals den Aufbau überdacht. - Dokumentation nachführen, Heatsinks installieren	
Aufgetretene Probleme	-	
Tag: 71	Datum: 17.04.2018	Aufwand:
Erledigte Arbeit	- Neuen Aufbau des Clusters in Angriff genommen. Ich habe mir ein passendes Gerüst / Gestell in einem Warenhaus gekauft.	
Aufgetretene Probleme	-	
Tag: 72	Datum: 18.04.2018	Aufwand:
Erledigte Arbeit	- Schrauben für die Montage der Raspberry PI's bestellt. - Dokumentation nachgeführt.	
Aufgetretene Probleme	-	
Tag: 73	Datum: 19.04.2018	Aufwand:
Erledigte Arbeit	- Überarbeitung des Diplomberichts.	
Aufgetretene Probleme	-	
Tag: 74	Datum: 20.04.2018	Aufwand:
Erledigte Arbeit	- Diplombericht erweitern.	
Aufgetretene Probleme	-	
Tag: 75	Datum: 21.04.2018	Aufwand:
Erledigte Arbeit	- Diplombericht erweitern.	
Aufgetretene Probleme	-	
Tag: 76	Datum: 22.04.2018	Aufwand:
Erledigte Arbeit	- Diplombericht erweitern.	
Aufgetretene Probleme	-	
Tag: 77	Datum: 23.04.2018	Aufwand:
Erledigte Arbeit	- Schrauben sind angekommen. - Ich habe den Cluster neu zusammengestellt.	

Aufgetretene Probleme	-	
Tag: 78	Datum: 24.04.2018	Aufwand:
Erledigte Arbeit	- Wackelkontakte waren vorhanden. Ich musste nochmals die Verkabelung stabiler gestalten. Die Raspberry PI's wurden nicht konstant mit 5 Volt versorgt.	
Aufgetretene Probleme	-	
Tag: 79	Datum: 25.04.2018	Aufwand:
Erledigte Arbeit	-	
Aufgetretene Probleme	-	
Tag: 80	Datum: 26.04.2018	Aufwand:
Erledigte Arbeit	- Dokumentation erweitert.	
Aufgetretene Probleme	-	
Tag: 81	Datum: 27.04.2018	Aufwand:
Erledigte Arbeit	- Monitoring Programme optimieren und einrichten. Nagios konnte auf nicht alle Ports eine Verbindung aufbauen. Dies wurde gefixt. Die Ursprüngliche Konfiguration war nicht für die Umgebung eingerichtet.	
Aufgetretene Probleme	-	
Tag: 82	Datum: 28.04.2018	Aufwand:
Erledigte Arbeit	-	
Aufgetretene Probleme	-	
Tag: 83	Datum: 29.04.2018	Aufwand:
Erledigte Arbeit	-	
Aufgetretene Probleme	-	
Tag: 84	Datum: 30.04.2018	Aufwand:
Erledigte Arbeit	- Ganglia Monitoring konfiguriert, XMLParser Fehler waren vorhanden. Ganglia wurde für RPI Tests optimiert.	
Aufgetretene Probleme	-	
Tag: 85	Datum: 01.05.2018	Aufwand:
Erledigte Arbeit	- Mining Tests absolviert. Alle gewünschten Währungen wurden für eine Testdauer von jeweils 30 Minuten geschürft. - Die Raspberry PI's müssen noch übertaktet werden.	
Aufgetretene Probleme	-	
Tag: 86	Datum: 02.05.2018	Aufwand:
Erledigte Arbeit	- Dokumentation nachgeführt. - Raspberry PI's übertaktet (PXE Boot Image).	
Aufgetretene Probleme	-	
Tag: 87	Datum: 03.05.2018	Aufwand:
Erledigte Arbeit	- Dokumentation erweitert.	

Aufgetretene Probleme	-	
Tag: 88	Datum: 04.05.2018	Aufwand:
Erledigte Arbeit	- Dokumentation erweitert.	
Aufgetretene Probleme	-	
Tag: 89	Datum: 05.05.2018	Aufwand:
Erledigte Arbeit	-	
Aufgetretene Probleme	-	
Tag: 90	Datum: 06.05.2018	Aufwand:
Erledigte Arbeit	-	
Aufgetretene Probleme	-	
Tag: 91	Datum: 07.05.2018	Aufwand:
Erledigte Arbeit	-	
Aufgetretene Probleme	-	
Tag: 92	Datum: 08.05.2018	Aufwand:
Erledigte Arbeit	- Dokumentation erweitert.	
Aufgetretene Probleme	-	
Tag: 93	Datum: 09.05.2018	Aufwand:
Erledigte Arbeit	- Dokumentation erweitert.	
Aufgetretene Probleme	-	
Tag: 94	Datum: 10.05.2018	Aufwand:
Erledigte Arbeit	- Dokumentation erweitert.	
Aufgetretene Probleme	-	
Tag: 95	Datum: 11.05.2018	Aufwand:
Erledigte Arbeit	- Dokumentation erweitert.	
Aufgetretene Probleme	-	
Tag: 96	Datum: 12.05.2018	Aufwand:
Erledigte Arbeit	- Dokumentation erweitert.	
Aufgetretene Probleme	-	
Tag: 97	Datum: 13.05.2018	Aufwand:
Erledigte Arbeit	- Dokumentation erweitert.	
Aufgetretene Probleme	-	
Tag: 98	Datum: 14.05.2018	Aufwand:
Erledigte Arbeit	- Dokumentation erweitert.	
Aufgetretene Probleme	-	
Tag: 99	Datum: 15.05.2018	Aufwand:
Erledigte Arbeit	- Dokumentation erweitert.	
Aufgetretene Probleme	-	
Tag: 100	Datum: 16.05.2018	Aufwand:
Erledigte Arbeit	- Dokumentation erweitert.	

Aufgetretene Probleme	-	
Tag: 101	Datum: 17.05.2018	Aufwand:
Erledigte Arbeit	- Dokumentation erweitert.	
Aufgetretene Probleme	-	
Tag: 102	Datum: 18.05.2018	Aufwand:
Erledigte Arbeit	- Dokumentation erweitert.	
Aufgetretene Probleme	-	
Tag: 103	Datum: 19.05.2018	Aufwand:
Erledigte Arbeit	- Dokumentation erweitert.	
Aufgetretene Probleme	-	
Tag: 104	Datum: 20.05.2018	Aufwand:
Erledigte Arbeit	- Dokumentation abgeschlossen. - Die Dokumentation ist nun bereit für Korrekturen.	
Aufgetretene Probleme	-	
Tag: 105	Datum: 21.05.2018	Aufwand:
Erledigte Arbeit	- Überarbeitung der Dokumentation.	
Aufgetretene Probleme	-	

Tabelle 150: Arbeitsjournal

J Quellenverzeichnis

Namen der Quelle	Titel und Bemerkung
Wikipedia https://en.wikipedia.org/wiki/Comparison_of_cluster_software	Cluser Software Vergleichstabelle.
HPC Today http://www.hpctoday.com/best-practices/tinytitan-a-raspberry-pi-computing-based-cluster/	Installationsanleitung und Beschreibung der HPC Lösung TinyTitan
Jordi Corbilla von Thundax Software http://thundaxsoftware.blogspot.ch/2016/07/creating-raspberry-pi-3-cluster.html	Komplette Installationsanleitung einer Noname Cluster Lösung
Benutzer Sakaki auf Github https://github.com/sakaki-/gentoo-on-rpi3-64bit	Repository des Gentoo Images und Installationsanleitung
CentOS http://mirror.centos.org/altarch/7.4.1708/isos/aarch64/	Image Repository von CentOS
raspberrypi.org https://www.raspberrypi.org/documentation/hardware/raspberrypi/bootmodes/net_tutorial.md	Installationsanleitung zu PXE / Netzwerkboot
Fedora https://fedoraproject.org/wiki/Architectures/ARM	Fedora Image für Raspberry PI's und Installationsanleitung dazu.
Fedora https://fedoraproject.org/wiki/Architectures/ARM	Fedora Image für Raspberry PI's und Installationsanleitung dazu.
Benutzer Uli Middelberg auf Github https://github.com/umiddelb/aarch64/wiki/Install-CentOS-7-on-your-favourite-ARMv8-ARM64-AArch64-board	Beschreibung und Anleitung der Umgehungslösung für die Installation von CentOS auf den Raspberry PI's

Abbildungsverzeichnis

1	Risiken	15
2	Physikalischer Überblick	16
3	Technischer Überblick	18
4	Physischer Aufbau	27
5	FMW: Fedora Media Writer starten	31
6	FMW: Archiv auswählen	32
7	FMW: Abbild schreiben	32
8	Hostnamen editieren	35
9	Übersicht der Hostnamen Zuweisung	35
10	Statische IP vergeben	35
11	NFS: Erlaubte Clients	36
12	Projektplan detailliert	54

Tabellenverzeichnis

1	Situationsanalyse Komponenten	1
2	Situationsanalyse Stärken	2
3	Situationsanalyse Schwächen	2
4	Projektziele	3
5	Lieferobjekte	4
6	Beschaffungskosten	5
7	Aufwandskosten	5
8	Stromkostenrechnung	6
9	Gesamtkosten	6
10	Wirtschaftlichkeit Hardware	7
11	Grober Projektplan	8
12	Termine	9
13	Projektbudget	9
14	Sachmittel	9
15	Organisation	10
16	Projektablage	10
17	Software Kriterien	11
18	Variantenübersicht	12
19	Anforderungsabdeckung	13
20	Notenskala der Kriterien	14
21	Bewertung der Varianten	14
22	Risiken	15
23	Protokolle	19
24	Verbindungen & Kommunikation	19
25	Komponente Management Node	20
26	Komponente aktive Compute Nodes	21
27	Komponente passive Compute Nodes	21
28	Testobjekte	22
29	Testarten	22
30	Fehlerklassen	23
31	Service Monitoring	24
32	Kryptowährungen	25
33	Management Node Name	25
34	Reserve Node Namen	25
35	Compute Node Namen	26
36	Variablen Definition	41
37	Testfall K-001	58
38	Testfall K-002	58
39	Testfall K-003	58

Tabellenverzeichnis

40	Testfall K-004	59
41	Testfall K-005	59
42	Testfall K-006	59
43	Testfall K-007	60
44	Testfall K-008	60
45	Testfall K-009	60
46	Testfall K-010	61
47	Testfall K-011	61
48	Testfall K-012	61
49	Testfall K-013	62
50	Testfall K-014	62
51	Testfall K-015	62
52	Testfall K-016	63
53	Testfall K-017	63
54	Testfall K-018	63
55	Testfall K-019	64
56	Testfall K-020	64
57	Testfall K-021	64
58	Testfall K-022	65
59	Testfall K-023	65
60	Testfall K-024	65
61	Testfall K-025	66
62	Testfall K-026	66
63	Testfall K-027	66
64	Testfall K-028	67
65	Testfall K-029	67
66	Testfall K-030	67
67	Testfall K-031	68
68	Testfall K-032	68
69	Testfall K-033	68
70	Testfall K-034	69
71	Testfall K-035	69
72	Testfall K-036	69
73	Testfall K-037	70
74	Testfall K-038	70
75	Testfall K-039	70
76	Testfall K-040	71
77	Testfall K-041	71
78	Testfall K-042	71
79	Testfall K-043	72
80	Testfall K-044	72

Tabellenverzeichnis

81	Testfall K-045	72
82	Testfall K-046	73
83	Testfall K-047	73
84	Testfall K-048	73
85	Testfall I-001	74
86	Testfall I-002	74
87	Testfall S-001	74
88	Testfall S-002	75
89	Testfall S-004	75
90	Testfall S-004	75
91	Testfall S-005	75
92	Testfall S-006	76
93	K-001 Protokoll	77
94	K-002 Protokoll	77
95	K-003 Protokoll	78
96	K-004 Protokoll	78
97	K-005 Protokoll	79
98	K-006 Protokoll	79
99	K-007 Protokoll	80
100	K-008 Protokoll	80
101	K-009 Protokoll	81
102	K-010 Protokoll	81
103	K-011 Protokoll	82
104	K-012 Protokoll	82
105	K-013 Protokoll	83
106	K-014 Protokoll	83
107	K-015 Protokoll	84
108	K-016 Protokoll	84
109	K-017 Protokoll	85
110	K-018 Protokoll	85
111	K-019 Protokoll	86
112	K-020 Protokoll	86
113	K-021 Protokoll	87
114	K-022 Protokoll	87
115	K-023 Protokoll	88
116	K-024 Protokoll	88
117	K-025 Protokoll	89
118	K-026 Protokoll	89
119	K-027 Protokoll	90
120	K-028 Protokoll	90
121	K-029 Protokoll	91

Tabellenverzeichnis

122	K-030 Protokoll	91
123	K-031 Protokoll	92
124	K-032 Protokoll	92
125	K-033 Protokoll	93
126	K-034 Protokoll	93
127	K-035 Protokoll	94
128	K-036 Protokoll	94
129	K-037 Protokoll	95
130	K-038 Protokoll	95
131	K-039 Protokoll	96
132	K-040 Protokoll	96
133	K-041 Protokoll	97
134	K-042 Protokoll	97
135	K-043 Protokoll	98
136	K-044 Protokoll	98
137	K-045 Protokoll	99
138	K-046 Protokoll	99
139	K-047 Protokoll	100
140	K-048 Protokoll	100
141	I-001 Protokoll	103
142	I-002 Protokoll	103
143	I-003 Protokoll	104
144	S-001 Protokoll	106
145	S-002 Protokoll	107
146	S-003 Protokoll	107
147	S-004 Protokoll	108
148	S-005 Protokoll	108
149	S-006 Protokoll	109
150	Arbeitsjournal	124