Austrian Public Service Blockchain
Technische Spezifikation

Doku-Klasse:
verbindlich

Kurzbezeichnung:
APSB-Tech-1.0

Kurzbeschreibung: Dieses Dokument spezifiziert die technischen Rahmenbedingungen zum Betrieb eines Blockchain Knotens im Rahmen der „APSB" – „Austrian Public Service Blockchain".

Verfasst von: DI Dr. Christian Baumann

Beiträge von:

Projektteam/Arbeitsgruppe: Austrian Public Service Blockchain

Version / Datum: V1.0 / 6.12.2021

Doku-Stadium: Ergebnis der AG

Gültig seit: **5.12.2022**

Nächste Überprüfung am: 5.12.2024

Inhaltsverzeichnis

Management Summary

Dieses Dokument spezifiziert die technischen Rahmenbedingungen zum Betrieb eines Blockchain Knotens im Rahmen der „APSB" – „Austrian Public Service Blockchain".

Einleitung

In Kapitel 1 werden die Anforderungen an die Systemumgebung beschrieben, die Anforderungen an die Server, die Blockchain Umgebung Multichain und Details zum IP-Netzwerk (Adressierung) incl. VPN.

Kapitel 2 beschreibt Details zur Blockchain selbst, zu den verwendeten "Streams" (Bereiche zur Speicherung), die Datenstruktur der Transaktionen, die Verwendung von Keys zur Optimierung und die einzusetzenden Hash-Verfahren.

1 Systemumgebung

1.1 Anforderungen Server

An einen Server zum Betrieb eines Blockchain Nodes werden folgende Anforderungen gestellt:

Die Leistungsanforderungen hängen u.a. vom Einsatzzweck des Nodes ab, pro Instanz sind etwa folgende Ausstattungsmerkmale empfohlen:

Ziel ist, dass die Texte unformatiert übernommen werden und dann nochmals die Formate im Nachhinein zu erstellen. Die Vorgangsweise ist

- „Minimaler" Node: nur Synchronisieren der Blockchain, kein POA-Mining, kein API, d.h. keine Transaktionen erstellen: 1 CPU/Core, 2GB RAM
- Maximale Funktionalität: 2 CPU/Core, 8GB RAM

Allgemeine Anforderungen

- Disk (empfohlen SSD) >= 50 GB
- Betriebssystem
    - Empfohlen: Ubuntu >= LTS 18.04
    - Alternativ möglich:
        - CentOS 6.2+, Debian 7+, Fedora 15+, RHEL 6.2+
        - Windows: 64-bit, supports Windows 7, 8, 10, Server 2008 or later.
        - Mac: 64-bit, supports OS X 10.11 or later

1.2 Blockchainumgebung „Multichain"

Als Blockchainumgebung wird das System „Multichain[1]" eingesetzt - https://www.multichain.com/

- Multichain Community Edition
    - Aktuelle Version 2.1.2
- Installation entweder native oder als Docker Image möglich

1.3 VPN „tinc"

Abbildungen und Tabellen sind automatisch durchnummeriert. Zusätzlich sollte auch ein Abbildungs- und Tabellenverzeichnis erstellt werden, dies erleichtern in umfassenderen Dokumenten die Navigation.

- Teilnahme an der APSB erfordert VPN "tinc" - https://www.tinc-vpn.org/
    - Version 1.0.36
- Installation siehe z.B. https://florianjensen.com/2018/03/30/set-up-tinc-on-ubuntu/

1.3.1 Netzwerk Details

IPv6 Netzwerk Bereich: fccc:0412:B10C:574b::x:y

y/y errechnen sich aus der IPv4 Adresse des Nodes, z.B.:

Tabelle 1 - Netzwerk

Host	**wkobcc01.rss.kapper.net**
IPv4	94.136.13.114
IPv6 im VPN	fccc:0412:B10C:574b::5e88:0d72

Vereinbarter Port: 9875

1.3.2 Organisatorisches

Nach Konfiguration des VPN ist das eigene Keyfile an die anderen Partner zu übermitteln, im Gegenzug erhält man alle anderen Keyfiles.

Tabelle 2 – Teilnehmer APSB

Teilnehmer	**Ansprechstelle**
Bundesrechenzentrum	help-desk@brz.gv.at
Wien	post@ma01.wien.gv.at
WKO und nic.at/cert.at	austriapro@wko.at
WU Wien	hotline@wu.ac.at

2 Streams, Datenstruktur

2.1 Blockchains und Streams

Im Rahmen des ersten Usecases „Notarisierung" (WKO: Daten-Zertifizierung) werden die Daten in einem gemeinsamen „Stream" gehalten. Es stehen derzeit zwei Systeme zur Verfügung:

Tabelle 3 - Streams

Umgebung	**Chain-Bezeichnung**	**In Betrieb seit**	**Knoten dzt.**	**DocNoS-Stream**	**Anmerkung**
Test	test	26.11.2018	ca. 5	dataStream	
Produktiv	APSB-20191017	17.10.2019	ca. 7	blockstempel	

2.2 Datenstruktur

2.2.1 Daten

Die Daten werden im JSON-Format in den Stream eingetragen und sind gemäß folgender Struktur aufgebaut:


```
{
   "metadataInternal": {
      "app": "unknown",
      "time": "1636376015000",
      "storageType": "JSON"
},
   "metadataExternal": {
      "additionalMetadata": null,
      "user": "wko-client-v2",
      "dataType": "Blockstempel-v2",
      "tags": [

         "Blockstempel-v2",
         "id:ec8857040dae02cf96d8c104d293a052",
"hash:sha256:42f27b9b5f947e318ad8391f590c48dc950d430ce438c75192578c543a3dfe58",
"hash:sha512:397be5e98e95d89a1b4358e71b3e119a41e0914fd49c1952b04ae2b433573fa5f99fba4cd
1b2a39329fc0968b8205ca07361da6bf21e7734a4024302e0e08e79"
   ]
   },
   "data": {
      "id": "ec8857040dae02cf96d8c104d293a052",
      "time": "2021-11-08T13:53:35+01:00",
      "hashes": {
         "sha256": "42f27b9b5f947e318ad8391f590c48dc950d430ce438c75192578c543a3dfe58",
         "sha512":
"397be5e98e95d89a1b4358e71b3e119a41e0914fd49c1952b04ae2b433573fa5f99fba4cd1b2a39329fc0
968b8205ca07361da6bf21e7734a4024302e0e08e79"
      },
      "optional": {
         "size": null
      }
   }
}
```

Tabelle 4 - Struktur der Daten

Feld	Beschreibung	Beispiel
metadataInternal		
app	Bezeichnung der internen Anwendung (oder „unknown“	
time	Zeitstempel der Verarbeitung	
storageType	Art der Speicherung der „data“	JSON
metadataExternal		
additionalMetadata	Zusätzliche Metadaten das Anwendung (oder null)	
user	Bezeichnung der externen Anwendung, optional	wko-client-v2
dataType	Verwendetes Datenformat; mandatory	Blockstempel-v2
tags	Array mit den gesetzten Keys (s.u.)	
data		
id	Id des Dokumentes; optional	
time	Zeitstempel nach ISO 8601; mandatory	2020-10-23T09:52:34+02:00
hashes	Ein oder mehrere Hashwerte des Dokumentes mit Bezeichnung des verwendeten Verfahrens. Mindestens ein Hashwert (sha256) ist mandatory.	"sha256": "211852…8526ea"
optional	weitere optionale Datenfelder	

Zur Bezeichnung von Hash-Verfahren werden die entsprechenden Token laut folgender Tabelle verwendet (Kleinschreibung):

Tabelle 5 - Hash-Verfahren

Hash-Verfahren	Token
SHA2– prefix „sha“	
Beispiel SHA2 256 Bit	„sha256“ (muss mindestens vorhanden sein)
Beispiel SHA2 512 Bit	„sha512“
SHA3 – prefix „sha3/“	
Beispiel SHA3 512 Bit	„sha3/512“ [2]

[2] Bitte beachten: Beim Codieren nach JSON wird dies zu „sha3\/512“

2.2.2 Keys

In der eingesetzten MultiChain Umgebung können Keys verwendet werden, um Informationen effizienter suchen zu können (vergleichbar mit indizierten Feldern einer Datenbank). Für die zu verwendenden Keys gelten folgende Regeln:

0) dataType, z.B:

Key 0: Blockstempel-v2

1) Für die „id" muss ein key angelegt werden, z.B.

`Key 1: id:ec8857040dae02cf96d8c104d293a052`

2) Jeder verwendete Hashwert kann als key eingetragen werden, z.B.

`Key 2: hash:sha256:42f27b9b5f947e318ad8391f590c48dc950d430ce438c75192578c543a3dfe58`

Es muss mindestens der sha256 Hash eingetragen werden.

Beispielhafte Darstellung des gesamten Datensatzes incl. Keys:

Abbildung 1 - Datensatz incl. Keys

Key 0	Blockstempel-v2
Key 1	id:ec8857040dae02cf96d8c104d293a052
Key 2	hash:sha256:42f27b9b5f947e318ad8391f590c48dc950d430ce438c75192578c543a3dfe58
Key 3	hash:sha512:397be5e98e95d89a1b4358e71b3e119a41e0914fd49c1952b04ae2b433573fa5f99fba4cc

JSON data	{
	"metadataInternal": {
	"app": "unknown",
	"time": "1636376015000",
	"storageType": "JSON"
	},
	"metadataExternal": {
	"additionalMetadata": null,
	"user": "wko-client-v2",
	"dataType": "Blockstempel-v2",
	"tags": [
	"Blockstempel-v2",
	"id:ec8857040dae02cf96d8c104d293a052",
	"hash:sha256:42f27b9b5f947e318ad8391f590c48dc950d430ce438c
	"hash:sha512:397be5e98e95d89a1b4358e71b3e119a41e0914fd49c1
	]
	},
	"data": {
	"id": "ec8857040dae02cf96d8c104d293a052",
	"time": "2021-11-08T13:53:35+01:00",
	"hashes": {
	"sha256": "42f27b9b5f947e318ad8391f590c48dc950d430ce438c75
	"sha512": "397be5e98e95d89a1b4358e71b3e119a41e0914fd49c195
	},
	"optional": {
	"size": null
	}
	}
	}

Added	2021-11-08 12:53:49 GMT (confirmed)
Data location	on-chain, available

Abbildungsverzeichnis

Tabellenverzeichnis

Änderungen