AustriaPro & Blockchain Initiative Austria – DI Dr. Christian Baumann

„DatNoS“ – Daten Notarisierung – Beschreibung

V1.0 (für Private Sector Blockchain)

AUSTRIA PRO
Arbeitskreis Blockchain

BCI
Blockchain Initiative Austria

Inhalt

1. Einleitung

Dieses Dokument beschreibt das System „Daten-Notarisierungs Service“ (Arbeitstitel DatNoS), welches elektronische Informationen unterschiedlicher Art unter Nutzung der Blockchain Technologie dezentral und unveränderbar verspeichert und somit „notarisiert“. Damit kann zu einem späteren Zeitpunkt der Ursprungszeitpunkt und die Unverfälschtheit der betroffenen Informationen nachgewiesen werden.

Auf Blockchain-Technologie basierende Notarisierung kann auf zwei Varianten genutzt werden:

- Bei der hier beschriebenen „Daten-Notarisierung“ (DatNoS) werden Daten direkt im System abgelegt. Diese Daten sind entweder für alle Systemteilnehmer lesbar, oder nur für einen bestimmten Teilnehmerkreis.
- Alternativ dazu werden bei einer „Dokumenten-Notarisierung“ (DocNoS[1]) digitale Fingerabdrücke (Hashwerte) von beliebigen Dateien hinterlegt bzw. später verifiziert. D.h. es werden dabei keinerlei (lesbare) Daten (im Klartext) im System übertragen, verarbeitet oder gespeichert.

2. Systemaufbau – Varianten

2.1. Systemteilnahme / Zugriffsrechte

Ein DatNoS-Blockchainsystem kann auf unterschiedliche Varianten aufgebaut bzw. konfiguriert werden:

- **Öffentliches System**: Jeder Interessierte kann einen Blockchain Node betreiben, und damit (lesend) auf das System zugreifen.
- **Geschlossenes, privates System** („Konsortium-Chain“). Um am System teilnehmen zu können, muss man authentifiziert sein, z.B. durch Teilnahme an einem Konsortium. Ohne Authentifizierung kann man weder schreiben noch lesen.
- **Mischformen** aus diesen Varianten erlauben einen selektven Schreib-/Lesezugriff der einzelnen Nodes, siehe folgende Abbildung:

[1] Siehe gesonderte Beschreibung

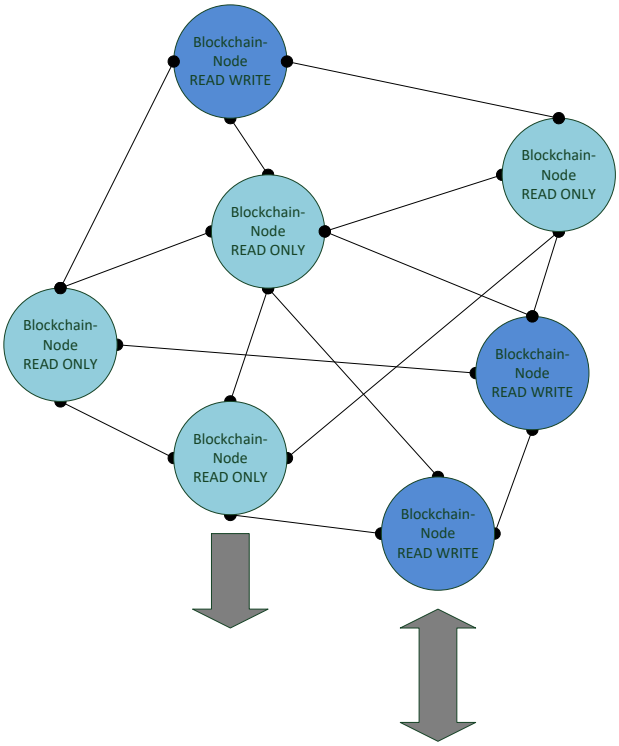

Abbildung 1: System mit "Read Only" und "Read/Write" Nodes

2.2. API-Zugriffe

Für externe Programme (Clients) werden die Zugriffe auf die Blockchain (Schreiben bzw. Suchen und Lesen von Daten) über APIs bereitgestellt. Abhängig von der Anzahl der Clients bzw. der Anzahl/Menge der zu übertragenden Daten können Systemteilnehmer eigene Nodes und APIs betreiben (siehe Teil (1) in folgender Abbildung) oder auf APIs zugreifen, die von anderen Anbietern bereitgestellt werden - siehe (2) in der Abbildung.

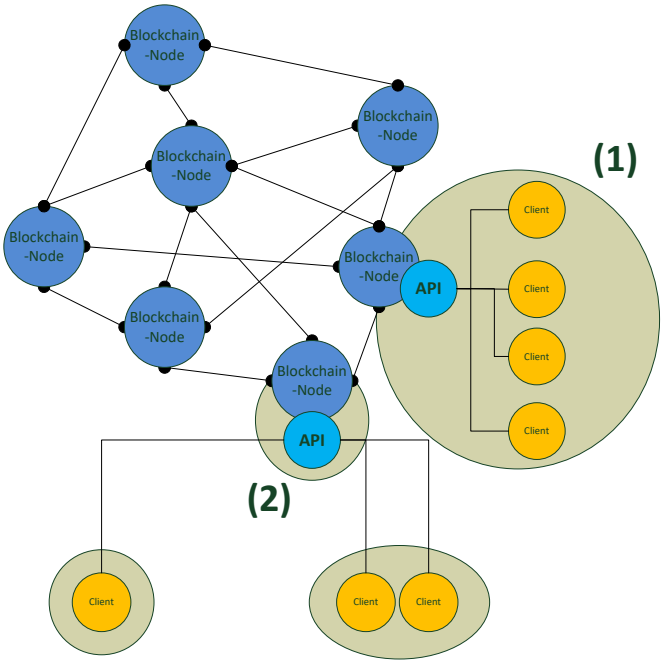

Abbildung 2: Varianten von API Zugriffen

Anmerkung: Die grauen Bereiche in der Grafik stellen z.B. einzelne Firmen/Institutionen dar.

3. Anwendungsschnittstelle (API)

Das REST-API stellt eine Funktionen für das Schreiben von Daten bereit. Es wird mit https POST angesprochen. Die Authentifizierung erfolgt über ein API-Token[2].

Die URLs des APIs lauten:

Testsystem	https://blockchains.web-lab.at/datnos-api/
Developmentsystem	https://test.baumann.at/dev9/datnos-api/

Ein DatNos Test Client in Python ist auf Github veröffentlicht:
https://github.com/austriapro/blockchain/tree/master/datnos-testclient

3.1. Request

Zur Authentifizierung des Clients am API muss folgender http Header gesetzt werden (Beispiel):

```
X-ApiToken: c235b4ff1d7a9ca8a8a7fa8740512bbfb6a8f13b
```

Die zu übermittelnden Nutzdaten sind im http Body als JSON zu übergeben und folgendermaßen aufgebaut (Beispiel):

```
{
   "keys":[
      "auto-capture-123",
      "VIE"
```

[2] Im derzeitigen Testsystem wird das API Token noch manuell vergeben, bitte dazu c.baumann@baumann.at kontaktieren.

```
   ],
   "data":{
      "timeStamp":"2022-05-06T09:52:56.850490",
      "device":"33123b9d-5f7c-4eb2-9344-b35943815ed5",
      "temp":22.24874556175432,
      "hum":9.139572439684985,
      "power":0.5684277811821341,
      "comment":"n/a"
   }
}
```

Die Elemente des Arrays „**keys**“ werden als Keys im Blockchain Stream eingetragen und ermöglichen (bei einer späteren Abfrage) eine flexible und performante Suche. Es können mehrere Elemente verwendet werden.

Im Element „**data**“ können beliebige Datenfelder mit beliebigen Datentypen verwendet werden (nach Json Spezifikation). Die Verwendung des Datenfeldes „timeStamp“ mit einer Zeitangabe im ISO8601 Format ist empfohlen.

Anmerkung: Derzeit besteht noch die Möglichkeit, Daten zu senden, die nicht der o.a. Struktur entsprechen. In diesem Fall werden die Daten vom API in obige Struktur gebracht, d.h. unter dem Element „data“ angehängt.

3.2. Response

Im Erfolgsfall retourniert das Service http Status 200 (OK) und folgenden JSON-Response:

```
{
   "success":"OK, data published in transaction ca5a42e99731e95ec632fb39bfb
3341199e516a978f0a7670cc5fffb199ba787",
   "timeStamp":"2022-05-06T09:52:57+02:00",
   "txid":
"ca5a42e99731e95ec632fb39bfb3341199e516a978f0a7670cc5fffb199ba787",
   "service":"DatNoS receiver v0.56 - (c) 2021 baumann.at"
}
```

Im Fehlerfall wird folgender Response gesendet:

```
{
      "error": "Error from config: 401",
      "system": "DatNoS receiver v0.56 - (c) 2021 baumann.at"
}
```

Zusätzlich wird der http Status laut folgender Tabelle gesetzt:

Statuscode	Bedeutung
401	Kein (gültiges) API-Token gesetzt
405	Method not allowed: Request ist kein Post-Request
400	Bad request: Keine Nutzdaten vorhanden oder Nutzdaten nicht (korrekt) JSON codiert
500	Fehler in der Konfiguration des Services

4. Datenstruktur im Blockchain Stream

4.1. Daten

Die übermittelten Daten werden in einem Blockchain Stream im Format JSON in folgender Struktur abgelegt (Beispiel, passend zu o.a. Request):


```
{
    "timeStamp": "2022-05-06T09:52:57+02:00",
    "client": "DatNoS-CB",
    "data": {
        "timeStamp": "2022-05-06T09:52:56.850490",
        "device": "33123b9d-5f7c-4eb2-9344-b35943815ed5",
        "temp": 22.248745561754,
        "hum": 9.139572439685,
        "power": 0.56842778118213,
        "comment": "n\/a"
    }
}
```


timeStamp	Zeitstempel des API
client	Bezeichnung (ID) des Clients, der das API genutzt hat (wird im Rahmen des API Tokens definiert)
data	Die vom Client übermittelten Nutzdaten im „data“ Element des Requests (siehe oben)

4.2. Schlüssel

Als Schlüsselfelder werden verwendet:

- Die vom Client im Element „keys“ übegebenen Werte
- Zuätzlich die Client-ID (die im Rahmen der Konfiguration des API-Tokens definiert wurde)

4.3. Beispiel

Folgender Screenshot zeigt ein Beispiel für einen Stream Eintrag, d.h. eine Blockchain Transaktion mit entsprechenden Daten:

Stream: datnos-test-1 – 1000 of 40933 items

Publishers	13VXwdarLRtV5fyP8qdWEFXebe6Ay45pgdY4Bb
Key 0	auto-capture-123
Key 1	VIE
Key 2	DatNoS-CB

JSON data	{
	"timeStamp": "2022-05-06T09:52:57+02:00",
	"client": "DatNoS-CB",
	"data": {
	"timeStamp": "2022-05-06T09:52:56.850490",
	"device": "33123b9d-5f7c-4eb2-9344-b35943815ed5",
	"temp": 22.248745561754,
	"hum": 9.139572439685,
	"power": 0.56842778118213,
	"comment": "n\/a"
	}
	}

Added	2022-05-06 07:52:59 GMT (confirmed)
Data location	on-chain, available

Abbildung 3: Beispiel für Abbildung der Daten in einem Blockchain Stream

5. Web-GUI

Zum Abfragen der Daten kann ein (sehr einfach gehaltenes) Web-GUI unter folgendem Url verwendet werden:

https://blockchains.web-lab.at/datnos/

Es zeigt die Datensätze des Blockchain Streams an, optional kann die Darstellung auf bestimmte Keys beschränkt werden.

Zusätzlich zu den Datensätzen (Nutzdaten plus Keys) werden die blockchainspezifischen Daten dargestellt:

Publisher/s	Public Key des Publishers = Betreiber des verwendeten APIs
Transaction	Transaktions-ID in der Blockchain
Blockhash	ID des Blockes
Blocktime	Zeitstempel des Blockes
Confirmations	Aktuelle (d.h. zum Zeitpunkt der Abfrage) Bestätigungen = Blöcke nach dem hier referenzierten Block

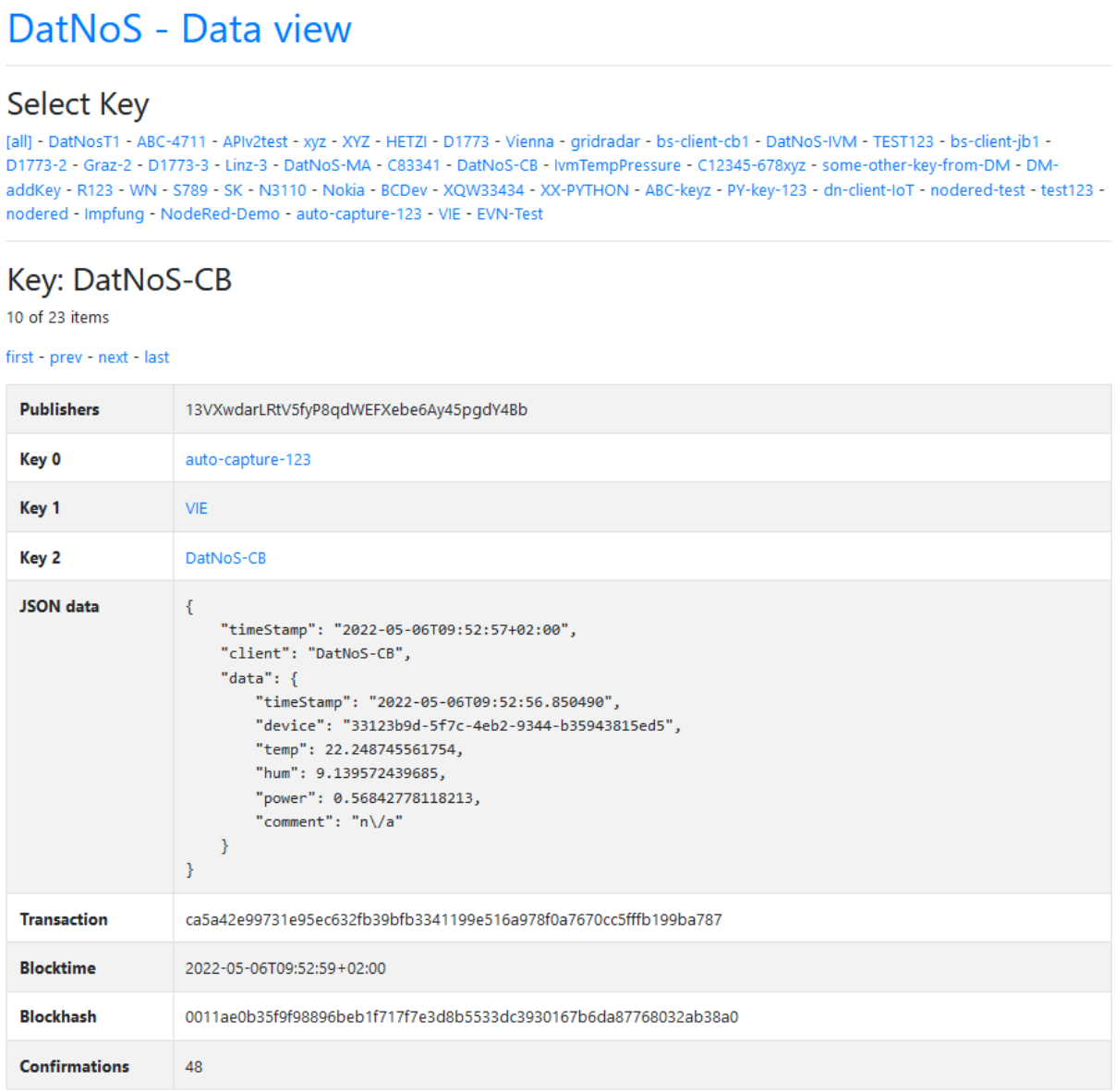

Abbildung 4: Web-GUI zur Abfrage von Daten

6. Systemumgebung

6.1. Blockchain Plattform

Als Blockchain Plattform wird MultiChain[3] eingesetzt, eine offene Plattform für Blockchain Anwendungen, die u.a. Data Streams und eine fein granulare Rechteverwaltung bietet. Die aktuelle Version 2.0 ist unter der Opensource Lizenz GPLv3[4] verfügbar. Alternativ wird eine kommerzielle Lizenz incl. SLA angeboten.

Die Parameter der Blockchainkonfiguration sind wie folgt:

	Aktuelles Testsystem	**Testsystem 2**
Bezeichnung Blockchain	mc2b1	zu definieren

[3] https://www.multichain.com
[4] https://github.com/MultiChain/multichain/blob/master/COPYING

Netzwerk-Port	7221	zu definieren
RPC-Port	7222 (durch Firewall gesperrt)	zu definieren
Bezeichnung Stream	datnos-test-1	zu definieren

6.2. Sicherheit

Alle Netzwerkverbindungen zwischen Clients und dem API-Service, sowie vom API-Service zum Blockchain-Node sind mittels https zu verschlüsseln. Zwischen API-Service und Blockchain-Node kann optional ein VPN zum Einsatz kommen.

Auf allen beteiligten Systemen sind Sicherheitseinrichtungen nach dem aktuellen Stand der Technik vorzusehen (Firewalls etc.).

7. Kontakt

Bei Fragen, Kommentaren etc. kontaktieren Sie mich unter

baumann.at - concepts & solutions
DI Dr. Christian Baumann
e-Mail: c.baumann@baumann.at
Tel.: +43 664 43 24 243
Web: https://www4.baumann.at