Josef Ressel Zentrum
für Blockchain-Technologien &
Sicherheitsmanagement

fh st.pölten

DLT4IT

eine neue DLT / Blockchain Technologie für bestehende IT-Anwendungen

Josef Ressel Zentrum für Blockchain-Technologien

Fachhochschule St. Pölten

Ernst Piller

Peter Kieseberg, Martin Pirker, Robert Haas, Dominik Sagl, Clemens Jung, Jürgen Wurzer

Status IT

- Speicherung und IT-Verarbeitung in Zentralen (Hosts), zunehmend in der Cloud, mit oft ungewisser Sicherheit
- Vertrauen in Hosts (Zentralen), vor allem Cloud, ist bei vielen Personen / Organisationen und Regionen mangelhaft
- **Gegenbewegung: die gesamte IT-Verarbeitung und Speicherung komplett oder zum Teil direkt beim Benutzer**
- PCs / Laptops / … haben heute Leistungskennzahlen in der Prozessorleistung und im Speichervolumen und das Internet (auch mobiles) ist so schnell geworden, dass dies möglich ist
- Blockchain Systeme und DLTs (Distributed Ledger Technologies) bekommen verstärktes Interesse

Kann man beide Welten vereinen?

- Aktuelle zentralistische IT hat andere Ziele / Orientierung als DLTs / Blockchains mit ihren eigenen Plattformen, Smart Contracts, Konsensverfahren etc. ➔ es bestehen erhebliche Unterschiede zwischen beiden „Welten“
- **Ein Hauptproblem von DLTs: alle Anwendungen müssen neu entwickelt werden (Kosten, Zeit, Qualität)**
- Frage: **„Kann man die hunderttausenden von existierenden und gut funktionierenden IT-Anwendungen, die heute auf Hosts ablaufen, auf Wunsch / auf Bedarf auch einfach DLT-fähig machen?“**

Anwendungen werden DLT-fähig

- **Ergebnisse des JRZ machen es möglich, wir befinden uns auf einem sehr guten Weg dorthin ➔ DLT4IT**
- **Dabei sollen nicht die existierenden IT-Anwendungen geändert werden, sondern es soll zusätzliche Software verwendet werden, die dies ermöglicht, dabei aber NICHT die heutigen DLT/Blockchain Plattformen verwenden**

➔ Viele der heutigen IT-Anwendungen können ohne oder mit nur geringen Änderungen rasch auf DLTs erweitert werden und damit von den Vorteilen von DLTs/Blockchains profitieren

➔ Statt in sicheren Zentralen laufen die Anwendungen in vielen einfachen Knoten, jeder Knoten kann alle Daten besitzen und die Daten sind verkettet (müssen es aber nicht)

Herausforderungen einer „DLT-fähigen Host-Anwendung

Dadurch soll Vertrauen in die IT-Verarbeitung zunehmen, aber auch Verfügbarkeit und Transparenz der Verarbeitung

Unsere großen Herausforderungen waren:

- die **IT-Sicherheit**, die in einem Host viel einfacher zu erzielen ist als in vielen einfachen Knoten mit Administratorrechten
- die **Berücksichtigung / Übernahme herkömmlicher Berechtigungssysteme**, die in Host-Systemen vorliegen
- der **Zugriffsschutz zu den Daten**: **nur über Kryptografie** mit neuen Ansätzen sicher möglich und wesentlich komplexer
- die **Replizierung und Synchronisation** der Datenbanken und Dateisysteme

DLT4IT Spezialitäten

Host-Anwendungen werden auf den Arbeitsplätzen etc. (bei den Benutzern) verarbeitet und nicht in „fernen“ Hosts

keine Neuentwicklung (Kosten, Zeit, Qualität), Verarbeitung und Daten bei Benutzern, Verkettung erhöht Transparenz

Parallelbetrieb Host-System und DLT möglich:

es gibt nicht nur DLT oder Nicht-DLT (Host-Lösung),

d.h. es ist möglich klein zu starten (kleine Benutzeranzahl und Budget),

es kann jeder Host-Benutzer zu DLT wechseln, ob und wann er möchte

und er kann auch wieder zurückwechseln, wenn ihm DLT nicht gefällt

DLT kann mit dem gleichen Berechtigungssystem arbeiten,

d.h. Benutzer haben die gleichen Zugriffsberechtigungen wie bisher

DLT4IT enthält kryptografisches Zugriffskontrollsystem

DLT4IT: Wichtige Komponenten

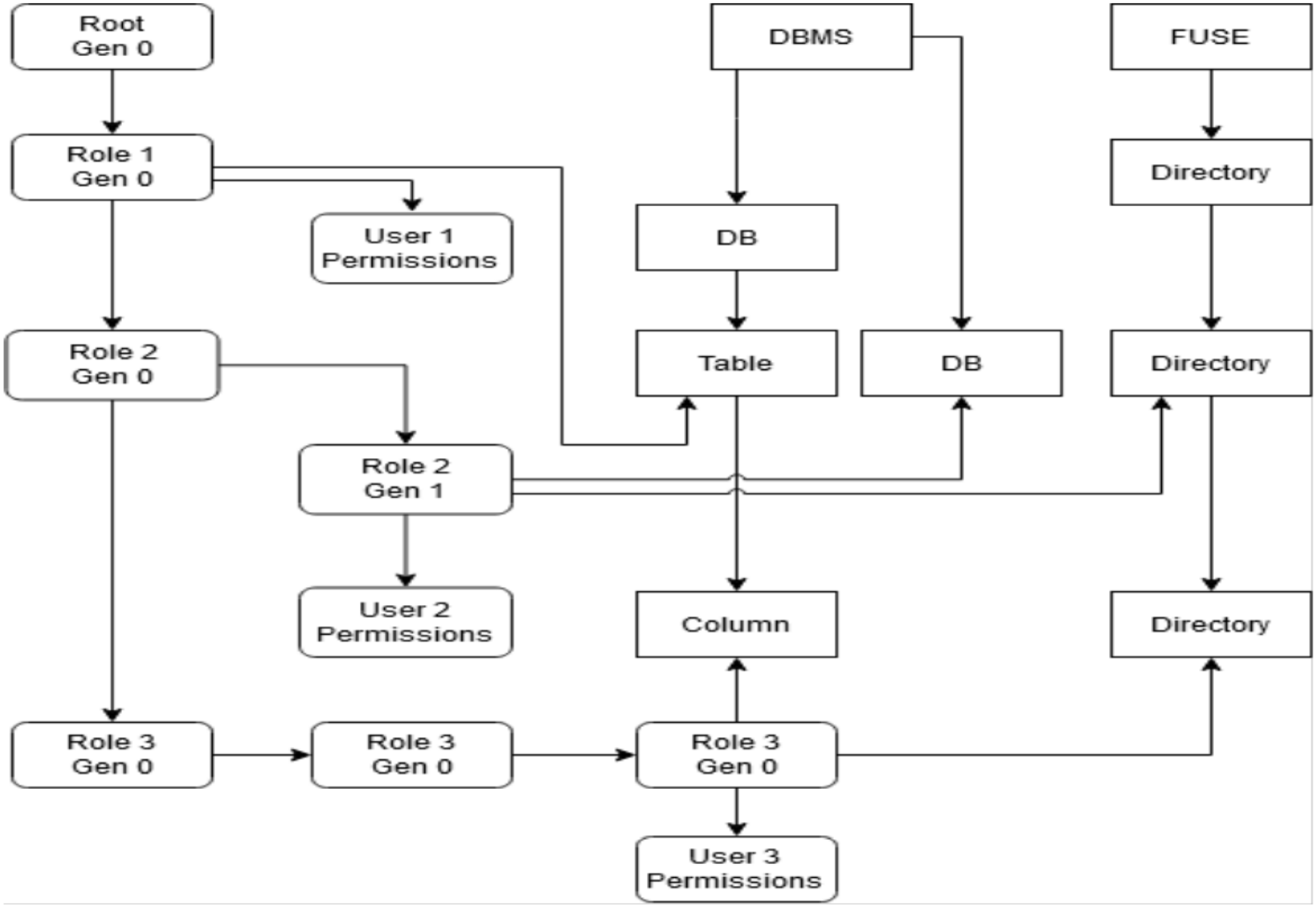
- Idee, Konzept, Spezifikation
- Verwaltungs-DAG (besteht aus mehreren DAGs)
- Schlüsselberechnung (erfolgt im Hardware-Token)
- Datenverschlüsselung (erfolgt im Node)
- Datenverkettung (erfolgt im Node)
- Middleware
- Datenreplizierung mit Schreibschutz (Fremd-SW)
- Hochsicherer Hardware Token (HWT)

Verwaltungs-DAG

Soll garantieren, dass alle Nodes rechtzeitig, sicher, gleiche Verwaltung- und Berechtigungsdaten erhält

- Enthält **Berechtigungs-DAG** (Lese-/Schreibbe-rechtigungen), **Zertifikats-DAG, DBMS-DAG**, etc.
- Jede Berechtigungen enthält Startzeitpunkt ➔ Gleichzeitigkeit in Nodes garantiert
- DAG-Daten sind jeweils durch Berechtigte signiert und verkettet, HWT und Nodes prüfen Signaturen ➔ alle Nodes und HWT haben gleiche Daten mit Nachvollziehbarkeit und Transparenz der Entstehung

CACHT: Berechtigungs-DAG
formale, rollenbasierte Darstellung



Kryptografisches Zugriffskontrollsystem CACHT

- Alle verschlüsselbaren Daten werden verschlüsselt
- Schlüssel zur Datenentschlüsselung spezifischer Daten erhalten nur Nodes mit einer Leseberechtigung
- Schlüsselberechnung für Node erfolgt durch hoch-sichern persönlichen Hardware-Token des Benutzers
- Zugriffsberechtigungen und sonstige wichtige Daten erhalten signiert alle Nodes durch Verwaltungs-DAG, sie garantiert Richtigkeit, Gleichzeitigkeit, Gleichheit
- Middleware liegt zwischen IT-Anwendungen und Dateisystem (FUSE-Schnittstelle) und DBMS
- Schreibschutz erfolgt durch Kontrolle anderer Nodes

CACHT: Schlüsselberechnung

- Erfolgt ausschließlich in Hardware-Token
- Erfolgt nach dem rollenbasierten Berechtigungs-DAG hierarchisch von Root-keys abwärts
- Root-keys erzeugen HWT gemeinsam
- Symmetrische Schlüssel für AES-256
- Zuerst erfolgt durch Hash-Berechnung nur den Rollen entlang ➔ für jede Rolle die Schlüsselgeneration 0
- Für jede Rolle von der jeweiligen Generation 0 mit Hash-Berechnung entsteht neue Schlüsselgeneration
- HWT gibt Node nur erlaubte Schlüssel des Benutzers

CACHT: Datenverschlüsselung

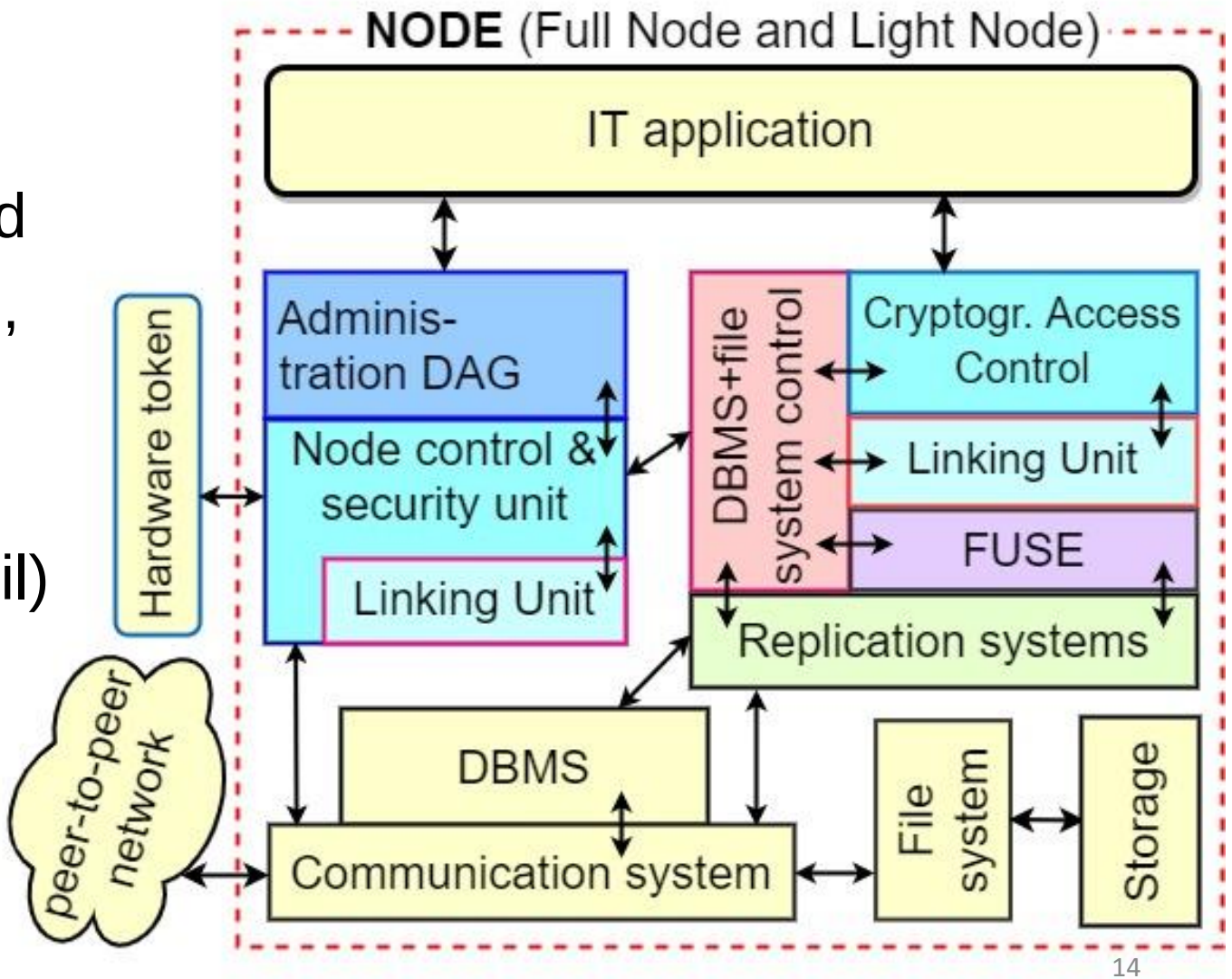
- Schlüsseln liefern hochsichere Hardware-Token
- Datenverschlüsselung erhält bei DBMS die Datenformate
- Zusammenfassung von Feldern (Elemente von Zeilen / Spalten in Datenbank-Feldern etc.) für Blockverlängerung
- Daten > 127 Bit werden mit AES XTS-Mode verschlüsselt
- Daten < 128 Bit werden mit FF1 verschlüsselt
- Where-Bedingungen in SQL-Kommandos (>, < etc.) werden bei der Verschlüsselung erhalten
- Bei Schlüsselwechsel erfolgt Neuverschlüsselung

Datenreplizierung und Schreibschutz

- Alle Dateien und Datenbanken müssen in allen Nodes „übereinstimmen“ ➔ geeignete Datenreplizierung
- Für DBMS „SymmetricDS“ ausgewählt und verwendet
- Tests - mit derzeit noch wenigen Nodes - mit dieser Software haben hohe Stabilität, Funktionalität, gute Synchronisierungseigenschaften und rasche Datenreplizierung ergeben
- Schreibschutz erfolgt, indem bei den empfangenen Daten die Schreibberechtigung der sendenden Nodes (Benutzers) überprüft wird

Middleware

Schnittstelle zwischen IT-Anwendung und Betriebssystem, DBMS, HWT, Netzwerk, ….. (nicht gelber Teil)



Hardware-Token (HWT)

- Standardisierte HWT nach ISO7816-3,-4,-8,-11,-15 in verschiedenen Bauformen (Chipkarte, USB-Stick, SD-Karte, Secure Element in Smartphones etc.), auch externe HSM (HW Security Module) möglich
- Software wird einmal hochgeladen und kann dann nicht mehr verändert werden

DLT4IT Vorteile

- Host-Anwendungen werden auf den Arbeitsplätzen der Benutzer inklusive aller Daten verarbeitet und nicht in „fernen“ Hosts bzw. einer Cloud ➔ Steigerung im Vertrauen in die IT-Verarbeitung, Datenschutz, Transparenz, Nachvollziehbarkeit, ……
- keine Neuentwicklung (Kosten, Zeit, Qualität) wie bei DLT
- Parallelbetrieb Host-System und DLT möglich
- Gleiches Berechtigungssystem wie im Host möglich
- Zugriffssicherheit: Kryptografisches Zugriffskontrollsystem

DLT4IT: Zusammenfassung und Ausblick

- Patent angemeldet, ein weiteres folgt in Kürze
- Prototyp für Basiskomponenten in Entwicklung
- Erste Tests (Funktionalität, Geschwindigkeit, etc.) durchgeführt, aktuell wichtige Aufgabe
- Am Ende soll ein Produkt entstehen, das über kommerzielles Unternehmen weltweit vermarket wird
- Josef Ressel Zentrum arbeitet an den noch offenen Forschungsfragen, Prototypenentwicklungen, Tests
- Suchen MitarbeiterInnen für verschiedenste Aufgaben