Universal Labs White Paper

A protocol for the decentralized ownership exchange on the Ubbey network

February 25, 2018

v1.2

Universal Labs - Keda Che

keda.c@ulabs.technology

Abstract

This paper introduces a new blockchain network called "Ubbey" and the "OWP protocol", which enables peer-to-peer ownership exchange on the Ubbey network. The OWP protocol is not only designed for the exchange of ownership of digital assets such as data, images and tokens, but also for the ownership exchange of physical assets such as land, cars and luxury products. DApps built on top of the OWP protocol can connect the virtual world with the real world, allowing ownership to be transferred between the two worlds.

Table of Contents

1. Introduction

In 2016, the author of this paper was working on his first blockchain start-up, Ownership Technology. While attending to Boost VC Tribe 7's Demo Day, he had the opportunity to meet and discuss the potential of blockchain technology with the founder of Second Life, Philip Rosedale. Mr. Rosedale proposed a challenge to the author: "*VR technology could empower the creation of a complete virtual world, as exemplified by the movie "Matrix". However, will it be possible for people to transfer the ownership of something between the real world and the virtual world? For example, if someone has a Ferrari in Second Life, can he or she transfer the ownership of the Ferrari to the real world and get a Ferrari from a Ferrari dealer?*"

1.1 Cloud Storage

Cloud storage is a simple and scalable cloud computing model where data is stored, accessed and shared remotely over the internet. Cloud storage allows users to safely store data separately from the point of generation, while also facilitating the access to it from any connected device. The demand for cloud storage services has grown exponentially over the last decade, and two types of cloud storage seem discernible: Personal and Public.

Public clouds involve the complete outsourcing of data storage to trusted centralized providers (e.g.: DropBox, Google Drive); these provide the benefit of a low monthly fees and ease of use. However, public cloud providers present certain important weaknesses: Unpaid bills may result in lost data; pricing rises substantially as storage requirements increase; centralized providers are prone to hackers and government eavesdropping; and centralized providers themselves may collect information from the data being stored.

Private clouds involve the installation of a local storage system to which permissioned devices can be connect to in close proximity. Unlike public clouds, private clouds enjoy higher levels of privacy and security: They are harder to breach, and centralized third parties will not have access to the data itself. Although frequently personal clouds present a higher upfront cost, the amortized cost over the long run will be lower, and this will be achieved while retaining the strictest ownership of personal data.

The drawbacks of personal clouds up until now, however, where related to the difficulty of setting up the system and the risks of lost data due to device failure. Universal Labs aims to harness all of the benefits of personal clouds without any of the drawbacks through a very simple and effective solution: Blockchain technology.

1.2 Blockchain

In 2008, an article titled "Bitcoin: A Peer-to-Peer Electronic Cash System" [1] was published under the pseudonym Satoshi Nakamoto[1]. This paper led to the creation of Bitcoin, the world's first decentralized

[1] Inventor of Bitcoin.

digital currency. Bitcoin's decentralized nature is realized through blockchain technology, a form of distributed ledger.

Since the birth of the Internet in October 1969, there has been ongoing research on the feasibility of issuing a digital currency. However, it was not until 2008 when Satoshi created Bitcoin that the challenges of launching a decentralized digital currency were overcome. These challenges revolve around the nature of data, namely that data can spread easily and freely on the internet, making it difficult to manage and maintain data ownership. For example, if person A shares an image with person B, then person B can immediately share this picture with other people. Likewise, person A can continue to send the picture to other people. This model of free data flow works great for the spread of information, but not for digital currency. This is because for something to qualify as money, it must have the two basic attributes of not being able to be copied and not being able to be spent twice. Any digital currency must also satisfy these two conditions to qualify as money. The traditional model is to rely on a third-party intermediary to carry out transaction settlement to guarantee these conditions. The innovation of blockchain technology is that it allows any two individuals on the internet to transfer value to one another without the need for third-party institutions to intervene.

The Byzantine Generals' Problem is a fundamental problem in point-to-point information networks. In distributed computing, computers in the network reach consensus through information exchange, but system failures or erroneous information might compromise system consistency [2]. Aside from quantum communication, blockchain technology is also an effective solution to this problem. Blockchain technology is no longer simply a public ledger, but a public computer. This means that anyone in the world can not only store data on the immutable blockchain, but also run programs on the blockchain. The underlying logic of all programs running on the blockchain is unanimously approved by all users to ensure consistency, security, and fairness.

1.3 What is ownership?

We understand ownership as the right to possess, use, profit and dispose of property in accordance with the law. China's classical text The Book of Lord Shang[2] described ownership based on the theory of "Ding Fen Zhi Zheng", the fixing of rights and duties to prevent disputes of ownership. During the Qin Kingdom's reformation, Duke Xiao of Qin inquired legalist statesman Shang Yang regarding how to govern his kingdom. Shang Yang replied:

> "That a hundred men will chase after a single rabbit that runs away, is not for the sake of the rabbit; for when they are sold everywhere on the market, even a thief does not dare to take one away, because their legal title is definite. Thus, if the legal title is not definite, then even men like Yao, Shun, Yu or Tang would all rush to chase after it, but if the legal title is definite even a poor thief would not take it."

If legal ownership is definite, then property cannot be legally obtained through brute force, and social order will be established. If ownership is not definite, then properties can be obtained through brute force, and society will descend into chaos. In effect, Shang Yang's governance principles are based

[2] A renowned Legalist text written during the Spring and Autumn Warring States period.

on the basic idea of ownership [3]. This historical event shows that as early as the Spring and Autumn Warring States period, people understood the value and function of ownership.

1.4 Ownership in the Digital Age

From a general perspective, we can classify all digital assets as data. In the age of big data, data is just as valuable an asset as land, labor, capital, and the aforementioned "rabbit". But how is data ownership defined? Data ownership can be understood through the framework of property ownership, and thus include having the right to control the data and claim the profits generated from the data. Control over data means being able to add, delete, change, and query the data. The principle of data ownership is that data belongs to those who have ownership over it, and data that is not owned by anyone are public resource.

One major difference between data and the "rabbit" described in the excerpt above is that data is composed of 0s and 1s in the computer network, which makes it extremely difficult to clearly assign ownership. The first difference is that data can be easily copied by nature, with the copies being the same as the original, while the "rabbit" in The Book of Lord Shang cannot be duplicated. Secondly, anyone with a copy of the data now has the same rights over the data as the original owner, and the original owner of the data loses control of this data. As a result of these two challenges, data owners are reluctant to exchange data with each other and must rely on a trusted third party for such exchanges. However, the profit-seeking nature of these centralized third parties is calling their neutrality and credibility into question. These "trusted" third parties sometimes are storing data in the absence of explicit authorization, neglecting data privacy, and even engaging in data fraud in extreme cases.

According to Steven Levy's book Crypto[3], cryptographer and Turing Award winner Whitfield Diffie[4] put forth the concept of a "decentralized view of authority" in the 1960s with the aim of building cryptographic tools to solve the problem of data security during data exchange [5]. Bitcoin, the first decentralized digital currency in the history of mankind, has definitively shown the power of decentralization. In his speech on Bitcoin [6], Andreas Antonopoulos[5] summarized the power of Bitcoin:

> "Bitcoin is fundamentally different because in Bitcoin you don't owe anyone anything and no one owes you anything it is not a system based on that it is a system based on *ownership* and no one can censor it no one can seize it no one can freeze it."

The Bitcoin system is built on blockchain technology, a technology that presents the solution to data ownership.

[3] A novel written by Steven Levy published in 2002

[4] Renowned cryptographer and security expert, inventor of public key encryption

[5] Security expert and author, the author of "Mastering Bitcoin"

In society, ownership is the foundation of many human activities. The function of ownership is not to simply define property and wealth, but to provide basic social stability and social order. The Coase Theorem[6], proposed by Nobel Prize Winning Economist Ronald H. Coase, states that explicit property rights are necessary for optimal resource allocation through market forces. Ownership in the traditional sense has been clearly defined and is well understood, but how ownership should be defined in the Age of the Internet remains an open question.

2. The Ubbey network

Universal Labs was created to solve a series of shortcomings presented by incumbent infrastructures:

- Digital data ownership: It is hard to assign ownership on data. It is even harder to claim, enforce and monetize that ownership.
- Universal solution: Lack of a universal protocol for the transaction of *digital* assets.
- Verification of physical assets: Difficulty for the average consumer to differentiate original products from counterfeit ones (e.g. Luxury handbags, luxury watches, fine arts, wines, etc.)
- Limitations of current cloud model: Inability of the current cloud service model to service exponentially increasing throughput. This goes on top of the aforementioned costs of setting up personal cloud systems and the risks of data loss due to device failure.

2.1 Universal Labs solution: The Ubbey Network

The Ubbey network is a decentralized network powered by *Ubbey Box* and the *Ownership Protocol* (OWP), which enables ownership exchange of digital & physical assets in a peer-to-peer way. Once an asset has been registered, it could be transferred directly from one IP to another on the Ubbey Network almost instantly, just like transferring files over peer-to-peer protocols.

2.2 The Ubbey Box Advantage

Ubbey Box is a hardware device designed by Universal Labs. It can be used as a personal cloud storage for individuals and enterprises. It provides user with massive amount of secure data storage space, with 1TB or 2TB options available in the first version. It allows users to backup or store at one place and access the file from anywhere and anytime. Users can connect Ubbey Box with different devices including but not limited to media players, gaming consoles and smart TVs. With the arrival of Internet of Things, Ubbey box can also connect with various smart devices in future. Ubbey Box is also a full node for the Ubbey network, which means it acts like a "miner" in Bitcoin network. The Ubbey network block creator will be randomly selected from Ubbey box nodes based on Proof of

[6] An economic theorem that describes the economic efficiency of an economic allocation or outcome in the presence of externalities

Storage consensus algorithm. Users can mine YOU token through sharing their personal storage space and be rewarded with YOU token as an incentive to power and secure the network.

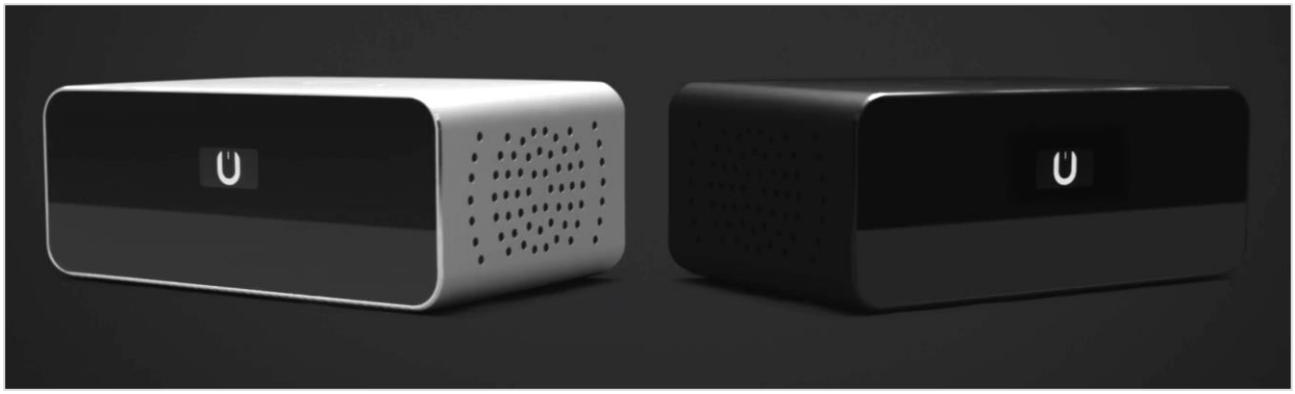

Figure 1. Ubbey Box

2.3 Survey of Consensus Algorithms

The success of Bitcoin has raised public awareness of the value of blockchain technology. Since the introduction of the Bitcoin blockchain, there have been many innovations such as colored coins, smart contracts, and new consensus algorithms. However, the technology still faces some fundamental challenges, including limitations in performance, high energy consumption, risks of forking, and a tendency to centralize.

The Bitcoin blockchain is currently experiencing some major issues. The Bitcoin blockchain produces a block every ten minutes, and it takes an hour or more to confirm a transaction. In addition, the proof-of-work (POW) consensus algorithm requires a large amount of wasted computing power and electricity. Moreover, the rise of large mining pools is leading to a centralization of authority, in opposition to the decentralized intent of cryptocurrencies. While innovative solutions such as side-chain technology and lightning network have been designed to overcome these issues, fundamental problems of computation power concentration and performance limitations are yet to be resolved.

Even the most active blockchain, Ethereum, requires 15 seconds to produce a block with throughput per second being in the single digits. In addition, there isn't necessarily just one globally-consistent Bitcoin blockchain. In fact, there often exists many forks of the blockchain containing the latest transaction data. For example, user A might see the block combination $\beta_1, \beta_2 \ldots \beta_k, \beta_{k+1}$ on the blockchain while user B sees the block combination $\beta_1, \beta_2 \ldots \beta_k, \beta_{k+1}^w, \beta_{k+2}^w$ on the blockchain. Only when the blocks $k+1, k+2$ have been verified and added to the chain can the block k be assumed to be consistent among all users. As a result of this uncertainty, the transactions in the latest blocks cannot immediately be assumed to be completed.

As blockchain technology moves beyond cryptocurrencies and into industry and consumer applications, the above challenges become serious limitations in widespread adoption of blockchain technology. Hence the need for a blockchain that is designed with real-world applications in mind, a blockchain that is democratic and offers high computing performance.

Name	Principle	Examples	Advantages	Disadvantages
PoW	Solve cryptographic problems to obtain the right to record transactions	*Bitcoin* *Litecoin* *Primecoin*	Simple and Secure	Computationally and energy intensive Inefficient and hard to scale Mining pool centralization
PoS	Ownership of the cryptocurrency confers the right to record transactions	*Peercoin* *Blackcoin* *Nxt*	Energy-saving Highly efficient	Susceptible to security threats such as nothing-at-stake Must rely on a security deposit
DPoS	Cryptocurrency owners vote for delegates who record the transactions	*BitShares*	Requires fewer nodes to reach consensus Consensus is reached quickly	Low participation leads to concentration of power
BFT	A majority of the nodes reach agreement	*HyperLedger*	Highly efficient	High cost of information propagation Not suitable for public blockchains Vulnerable to malicious attacks

2.4 Proof of Storage Consensus Algorithm

The Ubbey network adopts Proof of Storage/Space consensus algorithm and incentivizes Ubbey box users with YOU coin. Proof of Storage are motivated that users often have significant amounts of free disk. A Proof of Storage is a protocol between a prover A and a verifier B which has two distinct phases. After an initialization phase A is supposed to store some data D of size N, whereas B only holds some small piece of information. At any later time, B can initialize a proof execution phase, and at the end B outputs reject or accept [9].

3. The Ownership Protocol (OWP)

A typical public ownership registry is a publicly available record that associates an asset, either a digital asset such as token or a tangible physical asset such as a house.

3.1 Identity Management

Digital identity is critical to ownership transactions. It enables ways to interact with billions of users in the digital world. However, traditional identity systems are costly, disjointed and fallible. Digital identity is a growing issue in the digital economy. Even though there is only "one" physical person, digital clones often exist in data stored across the internet. To solve this issue, Universal Labs will at first collaborate with third parties such as blockchain digital ID service providers and then create a universal blockchain ID system to support all kinds of ownership transactions. For now we still need to reply on third parties for digital identity registration and verification, but eventually there will be a universal blockchain ID system adopted by most of people in both real world and virtual world.

3.2 Ownership Registry

There are four steps for which any ownership registration process needs to follow.

1). An asset must be demonstrated to be valid

2). The ownership must be verified (the identity of an owner)

3). The asset must be unimpeachable

4). Register

When this process is applied to blockchain registration, as mentioned above, for now a third party may be involved to set up the initial data when migrating to blockchain registration from existing off-blockchain technologies (no matter physical assets or virtual assets). Once the four processes have been completed, an ownership is generated on blockchain with the listed information:

Name	Data Type	Description
version	uint256	this will change each time the owner is updated.
registrant	address	address of an ownership account
owner	address	address of an ownership account

value	uint256	value of the asset (for reference only)
axn	hash	the asset's unique identity code
r	bytes32	signature of the registrant
t	time	registration time

3.3 Transfer

Once an asset is registered with the OWP protocol on the Ubbey network, the transfer of ownership of asset between two unique IDs can be done immediately. It is as easy as transferring files over peer-to-peer network.

3.4 YOU Coins

All computations carried out on the Ubbey network is subject to fees to prevent network abuse and to ensure the integrity of the consensus algorithm. The fees required for the transactions are specified in units of yo. Yo is used to pay the fees for all computations in the Ubbey network, including creating contracts, executing contracts, and conducting privacy-preserving ownership exchange on the OWP. The specific amount of yo required for each computation depends on the type of the computation and the amount of system resources required to execute it. YOU coins are used to purchase the yo used to conduct computations, so the yo price is quoted in the number of YOU coins. The yo price will fluctuate based on the amount of transactions on the network and the amount of system resources that are available.

YOU coins can also be used as the payment for sharing storage. Since we create a decentralized storage service, we will let all Ubbey network nodes to share profits from the decentralized storage service.

In short, YOU coins are used for the payment of:

- Creation and execution of smart contracts
- Exchange of digital asset ownership
- Storage of data
- Retrieval & recovery of data

The Proof-of-Storage consensus protocol reward users who provide storage capacity to the network with YOU coins.

4. Application Cases

See Appendixes.

5. Vision

The boundary between the real world and the virtual world will be blurry in the coming future. We want to use the OWP protocol to redefine the traditional OSI seven layers model and solve Phillip's challenge.

Layer	Protocols
1 Physical	Cabling protocols 802.3 Ethernet and 802.11 Wireless operate on both layers 1 and 2
2 Data Link	LLC, MAC, VLANs, PPP, L2TP, ATM and more
3 Network	IPv4, IPv6, ICMP, IPsec and more
4 Transport	TCP and UDP
5 Session	NetBIOS and PPTP
6 Presentation	ASCII, EBCDIC and MPEG
7 Application	DNS, DHCP, FTP, HTTP, SMTP, RDP and OWP (the OWP protocol)

5.1 Roadmap

- 2017-12 Xia: Launch of the Ubbey network project after 3 years of research and 1 year of blockchain consensus protocol development.
- 2018-02 Shang: Design and manufacture of the Ubbey Box prototype.
- 2018-05 Zhou: Shipment of the first generation of Ubbey Boxes.
- 2018-12 Qin: Launch of Main-net; moving the Ubbey Network towards a more mature stage of being fully decentralized and autonomous.
- 2019-03 Han: Integration of IPFS.

6. Acknowledgements

Here and now, we would like to extend our sincere thanks to all those who have helped us make this white paper possible. First and foremost, we are deeply grateful to the inventor of Bitcoin, Satoshi Nakamoto. His extraordinary work introduced blockchain technology to the world, set the foundation for a decentralized future, and served as the inspiration for the OWP protocol. Next, I want to thank the founder of Second Life and High-Fidelity Philip Rosedale. Finally, this project would not have been possible without the support of our friends.

7. References

[1] Nakamoto, Satoshi. "Bitcoin: A peer-to-peer electronic cash system." (2008): 28.

[2] Vukolić, Marko. "The Byzantine empire in the intercloud." ACM SIGACT News 41.3 (2010): 105-111.

[3] Duyvendak, Jan Julius Lodewijk. The Book of Lord Shang. Probsthain, 1928.

[4] Constine, Josh. “Facebook now has 2 billion monthly users… and responsibility.” TechCrunch, TechCrunch, 27 June 2017, techcrunch.com/2017/06/27/facebook-2-billion-users/. Accessed 17 July 2017.

[5] Levy, Steven. Crypto: secrecy and privacy in the new code war. London, Penguin, 2002.

[6] Antonopoulos, Andreas M. “Blockchain vs. Bullshit: Thoughts on the Future of Money.”

[7] https://en.wikipedia.org/wiki/Personal_cloud

[8] Computing, Fog. "the Internet of Things: Extend the Cloud to Where the Things Are." (2016)

[9] Dziembowski, Stefan, et al. "Proofs of space." Annual Cryptology Conference. Springer, Berlin, Heidelberg, 2015.

8. Disclaimer

This draft Universal Labs White Paper is for information purposes only. Universal Labs does not guarantee the accuracy of the conclusions reached in this paper, and the white paper is provided "as is" with no representations and warranties, express or implied, whatsoever, including, but not limited to: (i) warranties of merchantability, fitness for a particular purpose, title or non-infringement; (ii) that the contents of this white paper are free from error or suitable for any purpose; and (iii) that such contents will not infringe third-party rights. All warranties are expressly disclaimed. Universal Labs and its affiliates expressly disclaim all liability for and damages of any kind arising out of the use, reference to, or reliance on any information contained in this white paper, even if advised of the possibility of such damages. In no event will Universal Labs or its affiliates be liable to any person or entity for any direct, indirect, special or consequential damages for the use of, reference to, or reliance on this white paper or any of the content contained herein.

Appendix 1

Digital Assets and the OWP protocol

A new layer of the internet: The Ownership layer

September 12, 2017

v1.8

Keda Che, Dr. Hongyuan Yuan[7], Dr. Lei Ding[8]

info@ulabs.technology

Abstract

This paper presents the use case of the OWP protocol with digital assets. All digital assets can be classified as data. With the OWP protocol, a decentralized secure data computation and exchange platform ("Diffie") can be built. By incorporating zero-knowledge proof, homomorphic encryption, and secure multi-party computation, the platform delivers a secure and efficient data computation and exchange solution. The platform can reshape the traditional competitive landscape defined by zero-sum games, build a new paradigm for data sharing and collaborative computing, and usher in a new era of business cooperation and economic prosperity.

[7] Dr. Hongyuan Yuan is the Project Lead and Machine Learning Scientist at Adobe Systems Inc.

[8] Dr. Lei Ding is the Chief Data Scientist at Baidu Inc.

Introduction

Data as an Asset

In July of 2017, Mark Zuckerberg announced that Facebook's monthly active users exceeded 2 billion, which is close to a quarter of the world's population and more than half of all internet users [1]. With the development of social networks, e-commerce, and mobile internet, the amount of data has increased exponentially. In fact, the amount of data that is being generated is so large that we need a new unit of measurement called the Petabyte (PB)[9].

In the age of big data, data is just as valuable an asset as land, labor, capital, and rabbit. But how is data ownership defined? Data ownership can be understood through the framework of property ownership, and thus include having the right to control the data and claim the profits generated from the data. Control over data means being able to add, delete, change, and query the data. The principle of data ownership is that data belongs to those who have ownership over it, and data that is not owned by anyone are public resource.

One major difference between data and a physical asset is that data is composed of 0s and 1s in the computer network, which makes it extremely difficult to clearly assign ownership. The first difference is that data can be easily copied by nature, with the copies being the same as the original, while a physical asset cannot be duplicated. Secondly, anyone with a copy of the data now has the same rights over the data as the original owner, and the original owner of the data loses control of this data. As a result of these two challenges, data owners are reluctant to exchange data with each other and must rely on a trusted third party for such exchanges. However, the profit-seeking nature of these centralized third parties is calling their neutrality and credibility into question. These "trusted" third parties are storing data in the absence of authorization, neglecting data privacy, and even engaging in data fraud.

Diffie System

System Overview

On November 5, 2002, the famous novelist Neal Stephenson published Cryptonomicon[10], a novel thought by many to hold a "profound prophecy". This novel predicted the creation of cryptocurrencies in the beginning of the 21st century, that cryptocurrencies would have a profound impact on our society. The invention of Bitcoin seems to confirm the author's prediction. The author even boldly envisions a "data sanctuary" in Southeast Asia - a place where encrypted data can be freely stored and exchanged [4].

The design of the system combines Whitfield's "decentralized view of authority" with Stephenson's "data sanctuary". Whitfield's "decentralized view of authority" became a reality with the birth of blockchain technology. The OWP incorporates Whitfield's "decentralized view of authority through its

[9] A unit of data measurement equivalent to 10^15 bytes of data

[10] A novel by Neal Stephenson published in 2002

consensus algorithm while the Diffie Engine implements Stephenson's "data sanctuary" through privacy-preserving data computation.

Account System

Diffie's account system is one of the core components of the entire system, responsible for data control within the Diffie System. Unlike the single-password single-account system used in general information systems, or the pure dual-account system (internal and external accounts) adopted by Ethereum, the Diffie account system implements a dual account multi-password verification model that supports authorized access control of the system's data.

Account System Functionality: The functionalities of the Diffie account system are shown in the following list. These functionalities serve as the starting point for the design of the Diffie account system.

A. Subject of Ownership

B. Unique Data Controller

C. Smart Contract Authorization Controller

D. Multi-party Signature Access Authorization

Account System Components: Diffie's account system implements a dual account multi-password verification model that consists of four main components: the account lock system, account lock file, account lock interface, and client-side wallet. First, the account lock system grants authorization for transactions or data that need to be authorized. The account lock file is the data source that enables the account lock system. It consists of six variable groups, and serves as a unique ID: account system version, account authentication type, account public key, account private key, account expansion data, and operation sequence number. Client-side wallets are nodes in the OWP protocol, and is responsible for functionalities such as initiating data signing and creating and verifying transactions. The account lock interface is the communication interface between the account lock system and the client-side wallet, a framework for signature and encryption.

Diffie Engine

Overview

The Diffie Engine is a platform for running decentralized privacy-preserving applications. The core purpose of the Diffie Engine is to enable multi-party collaborative computing while allowing all parties to retain full control over data ownership. The Diffie Engine consists of three components:

(1) DVM (Diffie Virtual Machine): Smart contract execution engine, supports homomorphic encryption and other cryptographic primitives

(2) DOP (Diffie Oracle Protocol): Standardized data exchange protocol governing secure smart contract data exchange execution

(3) DAF (Diffie Application Framework): Decentralized privacy-preserving application development framework containing cryptographic function libraries

The Diffie Engine integrates blockchain technology with cryptographic technologies such as zero-knowledge proof, homomorphic encryption, and secure multi-party computation to enable the rapid development and deployment of privacy-preserving smart contracts and decentralized data computation applications.

Zero Knowledge Proofs

Introduced by S. Goldwasser and C. Rackoff in the 1980s [5], zero-knowledge proof is a cryptographic methodology by which one party (the prover) can prove to another party (the verifier) that a given statement is true, without conveying any information apart from the fact that the statement is indeed true. Zero knowledge proof is an effective cryptographic method of establishing a secure privacy-preserving protocol. To understand zero-knowledge proof, start by defining an interactive proof system.

Interactive Proof System: a pair of interactive machines <P,V>, where P and V represent the prover and the verifier respectively, can be considered as an interactive proof system of language L if they satisfy the following requirements:

(1) Machine V is polynomial time ;

(2) Completeness: for any $x \in L$, there exist an honest prover P, so that the system outputs "$x \in L$" when the verifier V completes the interaction with P;

(3) Soundness: for any $x \notin L$ and any prover P, the system output "$x \in L$" with negligible probability when the verifier V completes the interaction with P.

A zero-knowledge proof system is an interactive proof system that meets the requirements of zero knowledge proof and must possess the following four attributes:

(1) The verifier cannot obtain any information from the protocol ;

(2) The prover cannot deceive the verifier;

(3) The verifier cannot deceive the prover;

(4) The verifier cannot simultaneously disguise him/herself as a prover in another zero knowledge prove system;

Zero knowledge proof is particularly suitable for use cases which require privacy-preservation. Zerocash, a privacy-preserving version of bitcoin, is a classic example of the application of zero knowledge proof. Zerocash is the first blockchain system to integrate zero knowledge proof, providing full confidentiality in peer-to-peer payments. The system does not reveal the origin, destination, or amount of the transaction, while allowing authorized queries regarding transaction details through private keys.

The Diffie Engine provides a zero-knowledge proof security service layer at the architectural level for smart contracts and decentralized applications. This security service layer ensures privacy protection during data computation, such as for zero knowledge identification verification, transaction data security, etc.

Homomorphic Encryption

The topic of homomorphic encryption was first introduced by Ronald L. Rivest, Leonard Adleman, and Michael L. Dertouzos in 1978 [6], but it was only in 2009 that Craig Gentry proved the first homomorphic algorithm [7]. Homomorphic encryption is a form of encryption that allows computations to be carried out on ciphertext, thus generating an encrypted result which, when decrypted, matches the result of operations performed on the plaintext.

Let E(m) be the encrypted cyphertext of m, if E(a) and E(b) are known, then anyone can obtain the cyphertext of $a \oplus b$ through some sort of operation denoted by $E(a) \otimes E(b)$, and homomorphism can be generally expressed as:

$E(a \oplus b) = E(a) \otimes E(b)$,

where $\oplus$ and $\otimes$ represent the binary operations of plaintext and ciphertext spaces respectively.

Homomorphic encryption includes operations such as addition, subtraction, multiplication, and division. Achieving both additive homomorphism and multiplicative homomorphism means being able to run all operations, a state known as algebraic homomorphism.

Homomorphic encryption is a key cryptographic technology for the development of blockchain technology and the adoption of blockchain-based applications. Due to current security concerns, users do not perform computations on sensitive data directly on the blockchain. Therefore, the introduction of a homomorphic encryption technology suited for practical use can relieve user concerns over data security and drive the adoption of blockchain-based data computation.

Although current homomorphic encryption technology is computationally-prohibitive for large scale commercial use on large datasets, it is invaluable for the deployment of smart contracts in specific security-sensitive use cases involving small data sets. The Diffie Engine has a native homomorphic encryption operator built into the Diffie Virtual Machine to implement homomorphic cryptosystems including Palllier, Benaloh, RSA, and ElGamal. With this homomorphic encryption-powered Diffie Virtual Machine, the Diffie Engine enables the rapid development and deployment of decentralized privacy-preserving applications.

Secure Multi-Party Computation

The traditional method of completing a computation involving data from multiple sources/parties is to aggregate the data in a single location for centralized computation. While this method works for some cases, it fails in situations that lack a single entity with enough authority and credibility to obtain data from all participating parties. The following lists a couple of such situations:

(1) Alice suspects that she might have a genetic disease and knows that Bob has a DNA database that has categorized a wide variety of genetic diseases. Alice wants to find out if she indeed has the disease, but the only way to do so is to send her DNA sequence to Bob so that Bob can perform a diagnosis. However, Alice is concerned about her privacy and does not want to disclose her own DNA information and diagnostic results. It appears as if she must choose between her health and her privacy, both of which are important to Alice. With secure multi-party computation, she can get a reliable diagnosis without compromising her privacy.

(2) Company A is considering expanding into a certain geographical region, but is concerned that company B is also thinking of entering the same region. Both Company A and Company B want to avoid competing in the same region. Therefore, they want to verify that their market expansion plans do not overlap, but do not want to reveal the specific regions that they are targeting. Revealing such sensitive information can be very costly since Company A can preemptively act on Company B's plans and vice versa. Moreover, should such information leak, a third competitor can also execute on those plans, or the real estate developer can charge distortionary rates. With secure multi-party computation, Company A and Company B can achieve their goal of not competing in the same region without revealing their actual market expansion plans.

The above two examples share a common characteristic, namely that two or more parties want to engage in collaborative computation that requires the input of their private data, but none of the parties is willing to divulge their private data. The technical question becomes how to complete such computations while protecting the private information of all parties. This general problem is known as the secure multi-party computation problem.

Secure multi-party computation was proposed by the Turing Award winner A.C. Yao in the 1980s [8]. The main objective of secure multi-party computation is to complete the following computational task: In a trust-less distributed network, two or more parties can collaboratively perform agreed upon computations and retrieve the results of such computations, all while guaranteeing privacy-preservation. Secure multi-party computation has important applications in the fields of collaborative scientific computing, privacy-preserving database query, privacy-preserving data mining, privacy-preserving data analysis, and more.

Although O. Goldreich, S. Micali and A. Wigderson proposed a cryptographic computing protocol that can calculate any function [9], its real-world application is limited because the protocol runs a large number of zero-knowledge proofs which require users to transfer large amounts of data. Therefore, the key to improving the applicability of secure multi-party computation is to design protocols tailored to meet targeted use cases. The Diffie Engine categorizes the various use cases of secure multi-party calculation and build computation protocols into the blockchain infrastructure in order to meet the privacy-preservation and data computation needs of different industries and use cases.

Diffie Application Areas

Health Care Data

With the migration of medical data from paper records to electronic records since 2008, medical institutions and patients have accumulated a wealth of health care data. As data continues to accumulate, it becomes increasingly imperative that this data be analyzed and shared in order to improve health care services. This trend is resulting from a few major developments:

1. Patients are demanding a better and more individualized end-to-end health care experience

2. The low-hanging fruits in pharmaceutical R&D are gone

3. Patient concerns over data security and personal data privacy are increasing

In response to the above challenges, Diffie System provides a revolutionary solution. Take the case of a pharmaceutical company that is developing a new drug and in need of patient data for clinical trials. Under the current regime, the cost of doing so is high, not least because patients have concerns over their privacy and the security of their data. With guaranteed privacy and data security as well as autonomous control over their data, patients will be a lot more likely to share/sell their data to the pharmaceutical company. The smart contract can also simultaneously handle the payment attached to the sale of the patient's data. This pharmaceutical company is thus able to obtain the data that it needs at a much lower cost, from both a financial and non-financial perspective. This is a win-win situation in which the pharmaceutical company increased the efficiency of drug development by acquiring valuable patient data and the patient earned an income from their data without sacrificing privacy.

Financial Data

In the developing world, many people live without access to modern financial services, such as borrowing money from financial institutions. Traditional financial institutions base their lending decisions on previous financially related behavior, such as whether one has borrowed money or used credit cards and paid back the debt in time. This is a vicious circle to people who are excluded from traditional financial services, as it would be literally impossible for them to obtain the first credit card etc. and build a trustworthy credit record. However, this is a large population among which there are lots of otherwise qualified borrowers. Serving them is certainly of high economic value as well as social benefit.

Proper use of these individuals' personal data opens up a door to the aforementioned dilemma. For example, if a fresh college graduate has paid cell phone bills regularly in time, has participated in advanced classes, has often volunteered in his or her community, and is interested in events on software development, he may still be qualified and at low risk for lending a reasonably large amount of money, even his traditional credit record may be fairly insufficient.

With little doubt, leveraging big data or alternative data would be useful for a broad spectrum of financial applications. In fact, the use of telecom data and utility data to do credit rating has been reported in [10]. However, these companies typically hosting these volumes of data usually have a closed-door policy, making it hard for individuals to access their own records. Ownership blockchain and engine provide a means to this problem, in that individuals can authorize the use of their data to financial institutions via the ownership systems. The whole process guarantees that only the intended institutions would have access to the personal data, no intermediary or other parties would be able to

get such access. In this way, ownership blockchain and engines will enable the use of personal data in financial applications while preserving total safety and privacy.

One last point worth mentioning, the entire process works on end-to-end encryption and the only data owners and authorized users would have access. The elimination of additional parties in the loop not only improves safety and privacy, it also further brings down the cost incurred in the process, should third-party companies be needed to provide data services to financial institutions for credit evaluation.

Online Marketing

The exploding amount of data that internet companies are nowadays collecting and manipulating is fundamentally transforming the way in which marketing works. As of June 2017, 51% of the entire population on earth is using internet. Taking advantage of the large size and wide application of internet data, marketing is moving away from its traditional form of offline to online, realizing an unprecedented scalability and accuracy by artificial intelligence (AI). Today, online marketers not only know who the customers are and what the customers have done online but can also predict the customers next move fairly accurately using simple and robust models. The success of using data for prediction makes online marketing one of the fastest growing field in recent years, which is usually referred to as the beginning of the BIG DATA era.

However, simply increasing the quantity of data is far from reaching the full potential of value acquisition for online marketing. Currently, the data generated from any customer is being divided and collected separately by different companies he interacts with. For example, Walmart would only know if someone has shopped at Walmart's website, but would hardly know that the same person has also shopped at Amazon. This wall which prevents internet companies from realizing the full picture of their customers' online activity primarily results from legal issues regarding data privacy. Without the full picture, it is hard for internet companies to optimize their online marketing strategies, e.g. it is impossible to discover potential new customers using only the data from the same company.

Realizing the full picture of customers' online activity is the key path towards the post BIG DATA era. Currently, the temporal solution is through third party data providers. Internet companies purchase relevant data from these third-party data providers in the hope of adding value to their business. However, since third party data are collected in various ways and formats, e.g. fetching/copy data from websites directly, it usually suffers severely from limited categories and low quality, leaving the value of data unguaranteed. Even more problematic is the pricing. Since there are few alternatives, third party data providers usually have unfair advantage in pricing data.

In order to break the artificial "data wall" among different internet companies without the involvement of any third-party data providers, Diffie provides one promising solution; a decentralized data trading center built with the OWP protocol. Data can be traded freely and directly between any two interested companies on this platform without having to worry about leaking data to the un-authorized party. As a result, the data privacy is automatically protected. Besides, the quality of data is also guaranteed as being the first party data, which follows standard formats with completed meta data information and contents. Last but not the least, the data trading platform would serve as a fair market. The price of data would only be determined by the demand and supply relationship, neither the buyer nor the seller

alone. In summary, with the OWP protocol, the artificial "data wall" could be potentially overcome, leading to another transformative change in the field of marketing towards the post BIG DATA era.

Other

The above example is not the only use case of the Diffie System. In fact, the potential applications are only limited by imagination. Developers can develop various decentralized applications based on the OWP protocol and/or the Diffie Engine. Developers can also decide whether to incorporate privacy protection and data security functionalities into their decentralized applications.

References

[1] Constine, Josh. "Facebook now has 2 billion monthly users… and responsibility." TechCrunch, TechCrunch, 27 June 2017, techcrunch.com/2017/06/27/facebook-2-billion-users/. Accessed 17 July 2017.

[2] Levy, Steven. Crypto: secrecy and privacy in the new code war. London, Penguin, 2002.

[3] Antonopoulos, Andreas M. "Blockchain vs. Bullshit: Thoughts on the Future of Money ."

[4] Stephenson, Neal, and Jean Bonnefoy. Cryptonomicon. Paris: Payot & Rivages, 2000. Print.

[5] R.Riv est, A. Shamir and L. Adleman, "A Method for Obtaining Digital Signatures and Public-Key Cryptosystems," Communications of the ACM, Vol. 21, No 2, pp.120-126, 1978.

[6] Gentry C. Computing arbitrary functions of encrypted data[J]. Communications of the ACM, 2010, 53(3): 97-105.

[7] Goldwasser, Shafi, Silvio Micali, and Charles Rackoff. "The knowledge complexity of interactive proof systems." SIAM Journal on computing 18.1 (1989): 186-208.

[8] Yao, Andrew C. "Protocols for secure computations." Foundations of Computer Science, 1982. SFCS'08. 23rd Annual Symposium on. IEEE, 1982.

[9] Goldreich, Oded, Silvio Micali, and Avi Wigderson. "How to play any mental game." Proceedings of the nineteenth annual ACM symposium on Theory of computing. ACM, 1987.

[10] Turner, Michael A. "Predicting Financial Account Delinquencies with Utility and Telecom Payment Data." PERC Results and Solutions. 2015.

Appendix 2

Physical Assets and the OWP protocol

A new layer of the internet - the Ownership layer

January 01, 2018

v2.6

Keda Che[11]

keda.c@ulabs.technology

Abstract

This is a use case of the OWP protocol with physical assets. Luxury products are an ideal target for counterfeiters, and luxury brands subsequently face a significant problem with its goods being counterfeited. In fact, the quality of counterfeit goods is sometimes high enough that consumers cannot spot the difference. The prevalence of counterfeit goods on the market also results in brand dilution.

[11] The original draft was co-authored by Keda, Steven, Sergey and Marius in August 04, 2015.

EXECUTIVE SUMMARY

Fictitious Client Introduction

XYZ is the world's most valuable luxury brand with revenues of over $33 billion and a brand value of $30.9 billion. The multinational corporation makes luxury goods, which include leather goods, handbags, trunks, shoes, watches, jewelry and accessories. The distinctive products in this market are widely recognized and command premium prices in the marketplace.

Fictitious Consulting Team

IT Vendor YQTC is a Blockchain startup company based in Shenzhen, China, specialized in supply chain management system. YQTC is the main contractor for XYZ in a development project for checking the authenticity of manufactured luxury products.

Problem Statement

Being the market leader, XYZ is an ideal target for counterfeiters and subsequently faces a significant problem with its goods being counterfeited. In fact, the exterior of counterfeit goods is sometimes so good that consumers cannot spot the difference. On the other hand, the overall quality of the counterfeited product is bad. That's why customers who purchase counterfeit goods develop a negative perception of the XYZ brand and spread negative sentiment to other existing or potential clients. In addition, the large quantity of products being available on the market diminishes the "luxury appeal" of the brand. All these factors result in lost revenue and brand value for XYZ.

The company is well-aware of the severity of this problem and as a consequence employs a full-time dedicated team of 60 employees and spends $36 million annually to combat counterfeit goods. However, manual verification during the sale and return business processes is not a reliable and effective control mechanism. Current anti-counterfeit efforts are largely reactive.

Solution

YQTC has identified the primary problem as the lack of an integrated IT system capable of ensuring that only authentic products reach end customers.

The proposed IT platform introduces a new service that enables consumers to authenticate products against the manufacturer's production systems. Existing manual processes of product authentication will be automated to increase accuracy and consistency as well as decrease the amount of manual labor.

YQTC will establish a secure digital identification and validation platform with the OWP protocol to ensure that product authenticity is verified from manufacturer to consumer. The IT platform will generate unique digital product identification "stamps" that will be physically embedded into each product during the manufacturing process. The identification "stamp" can be used to reliably retrieve the product's

characteristics (brand, style) and the location it is authorized to be sold from. A mobile application is used to scan the product "stamp" and validate that the product is authentic.

Customers can use the mobile application to check authenticity before purchasing products. Retail employees can use the mobile application to check authenticity when receiving new product shipments and customer returns, as well as when selling a product in store.

1. Business Requirements

1.1 Key Business Objectives

- Reduce number of counterfeit goods sold in stores
- Make product verification easier, quicker and more consistent

1.2 "As-is" Business Process

The current business processes to validate product authenticity are manual. They are also highly vulnerable to subjective interpretation because of the high reliance on manual processes. In other words, there is a high degree of variability from the knowledge, experience, and training between XYZ employees. Therefore, the effectiveness of current business processes is extremely difficult to maintain.

Process 1 - Customer verifies a product before purchase

1. customer looks at the quality of the store and assesses whether it looks "real"
2. customer can check the corporate website to validate store location
3. customer enters a company store
4. customer browses the products available in-store
5. customer assesses the knowledge of the staff
6. customer evaluates the appearance of product
7. customer evaluates "materials and look and feel" of the product in order to confirm authenticity
8. customer makes a subjective decision on whether to purchase the product or raise a concern, based on subjective information collected above about the product

Process 2 - Employee verifies a product before sale

1. employee receives shipments from corporate office
2. employee makes a subjective decision that the shipment is valid
3. in the exceptional circumstances where an obvious problem is found, the employee would call corporate support

Process 3 - Employee processes a return

1. customer returns a product and requests a refund
2. employee checks the receipt
3. employee briefly validates the condition and physical appearance of the product to ensure its condition and whether the product matches the item present on the receipt
4. employee makes a subjective decision on whether the product is authentic
5. employee refunds the customer

Process 4 - Product is manufactured

1. exclusive material supplies are stocked in the factory
2. materials are assembled into a final product
3. products are packaged for individual retail distribution
4. products are recorded as inventory
5. products are packaged into large lots for store distribution
6. products are shipped to retail stores

1.3 "To-be" Business Process

The new IT system provides the ability to make the current manual evaluation processes required for customers and employees more precise. It is expected that the new system will decrease the average time spent on checking the authenticity of the products as well as significantly increasing the accuracy and consistency of the results.

Process 1 - Customer verifies a product before purchase

1. customer enters a company store

2. customer browses the products available in-store
3. customer scans a product with mobile phone and checks whether the product is new and authentic using the branded XYZ mobile app
4. customer makes a precise decision on whether to purchase the product or raise a concern, based on the precise information about the product

Process 2 - Employee verifies a product before sale

1. employee receives shipment from corporate office
2. employee validates each product is authentic before placing on retail shelves (optional - since this is duplication of effort)
3. employee validates that each product is authentic during the actual sale transaction (mandatory)
4. employee marks the product as "sold" using the app

Process 3 - Employee processes a return

1. customer requests a refund for product
2. employee checks the receipt
3. employee briefly validates the condition and physical appearance of the product to ensure its condition is adequate and whether the product matches the item sold on the returned receipt
4. employee scans product using the app and validates its authenticity and then makes a precise decision on whether to accept or decline the product
5. employee refunds the customer

Process 4 - Product is manufactured

1. exclusive material supplies are stocked in the factory
2. "stamps" are produced and shipped to the factory
3. materials are assembled into a final product
4. during assembly, "stamp" is physically embedded into product
5. products are packaged for individual retail distribution
6. products are recorded as inventory with assigned stamp

7. products are packaged into large lots for store distribution
8. products are shipped to retail stores

1.4 Required Functionality

Information system should fulfill the following requirements (grouped by component):

Mobile application for customers and mobile application for employees:

1. Mobile application for customer should have the ability to perform fast and reliant authenticity check of the product unique identifiers in read-only mode. The mobile application should natively support the NFC of the mobile device in order to scan the product's chips.
2. Mobile application for POS employee should have an ability to perform fast and reliant authenticity check of the product in read-only mode and should have the ability to mark product as sold in read-write mode. The mobile application should natively support the NFC of the mobile device in order to scan the product's chips.
3. Mobile applications should be able to interpret the scanned chip and convert it into the alphanumeric sequence that uniquely identifies the product in the system.
4. Mobile applications should only be able to connect to the single point of truth system from XYZ's internal network and provide the alphanumeric sequence as a request parameter.
5. Mobile applications should support an offline mode in order to avoid business disruption.
6. SSL-encryption technology should be used for ensuring the security of the communication between a mobile app and a backend system.
7. Customers mobile application should be published on the public application stores of the main mobile OS vendors.
8. Mobile applications should be created according to industry best-practice design guidelines.
9. Created mobile applications should be supported in multi-operating system environment (both iOS, Android are mandatory).
10. Mobile applications should generate the unique hardware/userID for future identification by the analytic system.
11. No user authentication should be required for customers using the mobile app.
12. User authentication should be mandatory for employees using the mobile app.

Component for generating stamps:

1. Component should contain the ability to generate unique identifiers (so called "stamps") using a hashing algorithm.

2. Component should be able to push the information with the OWP protocol and perform transactions between wallets.

3. Component should be integrated with legacy XYZ's Material Management SAP ERP using the natively supported protocols such as SOAP. Data exchange should include stamps and manufacturing metadata about the product.

Component for stamp checks:

1. All transactional requests and metadata from the applications should be recorded by the system. Data should include - mobile device type, user/device id, ip address, mac address, date/time of the request.

2. Component should support the role-based division of user rights.

3. Component should act as a backend application server for mobile clients and handle all requests from mobile applications.

4. Component should be integrated with legacy XYZ's Material Management SAP ERP using natively supported protocols such as SOAP. Data exchange should include stamps and manufacturing metadata about the product.

Component for maintaining the lifecycle of the employee's mobile application (enterprise application store):

1. Owner of the system should have a full control and visibility over the distribution of mobile application to POS employees.

2. Owner of the system should be able to perform remote device management of employee's mobile devices (wipe, block, install profiles, etc.).

Component for data analysis:

1. Component should provide the ability to aggregate data on transactional requests from mobile applications.

2. Component should be integrated with legacy XYZ's SAP Business warehouse using natively supported protocols such as SOAP. Data exchange should include aggregated information about mobile user requests.

2. Technical Specifications

2.1 Solution Overview

A complete list of applications involved in the proposed business solution and their usage types is presented in the table below:

#	System component	Application name	Application type	Existence
1	Mobile application for customer	StampCustomer	Mobile app for customers	New
2	Mobile application for employee	StampEmployee	Mobile app for POS employees	New
3	Enterprise application store	StampStore	XYZ's internal mobile application store	New
4	Component for stamps check	StampCheck	Back-end for mobile apps	New
5	Component for data analysis	StampDA	Business analytics of stamp usage	New
6	Component for generating stamps	StampGen	Stamp generator	New
7	Business warehouse		Data warehouse and processing	Legacy
8	Internal dashboards		Internal dashboard reports	Legacy
9	Enterprise resource planning system		Materials management, logistics, CRM	Legacy
10	Authentication system		Provides authentication services	Legacy

At a high level, the new solution will introduce several new services. These services are delivered through client-facing mobile applications as well as internal-facing web application (additional capability for future development of the project). The new system is broken down as individual modules in the diagrams to aid in the explanation and development process. These individual modules may be deployed independently as micro-services or together on a single server.

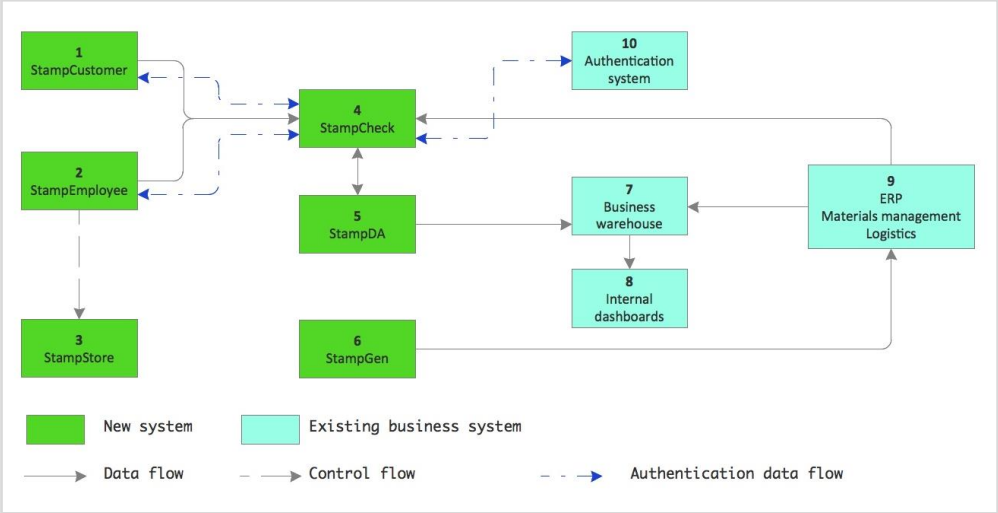

Figure 1. High-level presentation of system architecture.

Additional information regarding data flow and control flow is provided in the 2.2 and 2.3 chapters of the document.

2.2 Business Context

The proposed solution affects business processes in four primary areas: Customer, Retail, Manufacturing, and Fraud Prevention.

In the customer context, a new service will be made available which provides customers with an interface to validate the authenticity of goods. Retail staff will also have this ability in addition to gaining an interface for marking products sold/returned. In the manufacturing context, interfaces must be provided to read/assign "stamps" during manufacturing, and to enter product details into the ERP as inventory after manufacturing. Finally, the fraud prevention business group must have interfaces available to receive activity/efficacy reports, as well as a management interface that allows them to alter the sensitivity of "alarm" triggers.

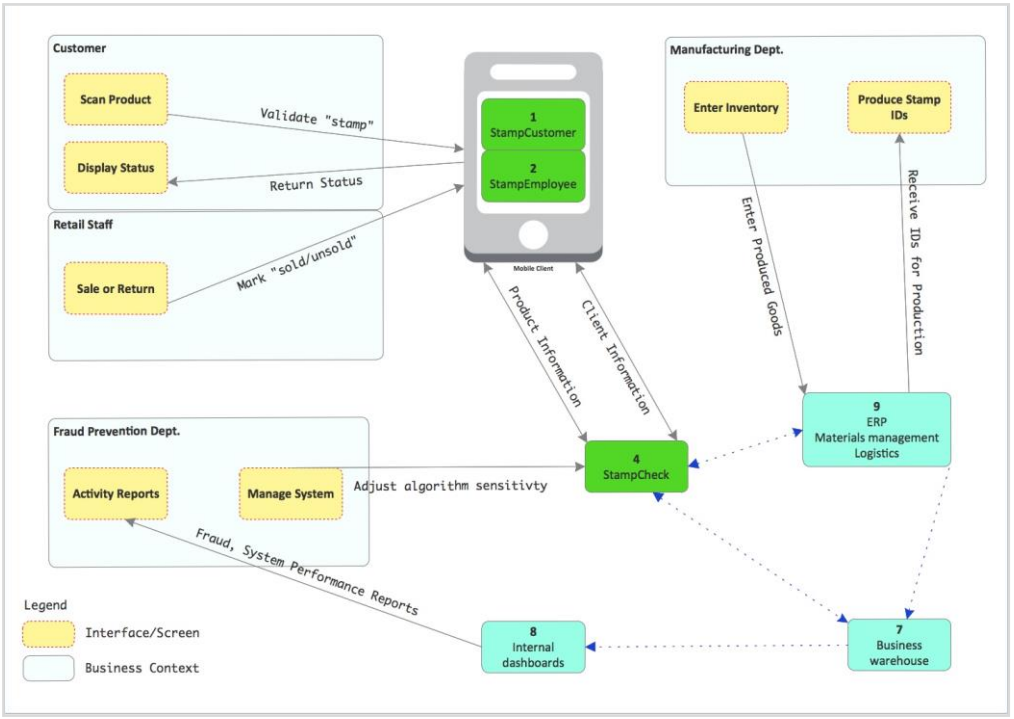

Figure 2. Business context diagram.

2.3 Vendor Selection

The four components of the system landscape (s. Table below) are built using existing 3rd party software and tools. The following section contains the vendor selection process for each component.

No	Application name	Development information	Vendor selection
1	StampCustomer	New development using 3rd party software and tools	Yes
2	StampEmployee	New development using 3rd party software and tools	Yes
3	StampStore	Configuration of predefined 3rd party software and tools	Yes
4	StampCheck	New development using open technologies	No
5	StampDA	New development using 3rd party software and tools	Yes
6	StampGen	New development using open technologies	No

(1) & (2) Mobile development packaging tool:

In order to build an application that can be deployed on all popular mobile operating systems without duplicating effort or code, YQTC has opted to use a 3rd party product that will allow them to develop the application once, using the widely used HTML5, CSS and JavaScript languages, and deploy instantly to any popular operating system. The products considered are listed below:

#	Metric name	Titanium Appcelerator	Adobe PhoneGap	NativeScript
1	Gartner's quadrant part	Leader	Leader	Visionary
2	Gartner's completeness of vision (rank)	1st	2nd	3rd
3	Available documentation*	High	High	Low

*Available documentation considers both trainings and tutorials available on the vendor's pages and related blogs and posts on specialty websites like StackOverflow

According to Gartner's business quadrant[12] Titanium and PhoneGap are in the "Leader's" part of the quadrant. NativeScript is in a "Visionaries" part and does not have huge amount of successful projects as above mentioned.

(3) StampStore

In order to identify the best Enterprise application store solution for XYZ's requirements the magic quadrant 2015 from Gartner has been used. Both Leaders and Challengers groups have been taken into consideration. As XYZ's requirements contain only mobile device management functionality only those metrics has been used. As metrics 1-3 are purely technical and metrics 4-5 are obvious all metrics have the same weight.

#	Metric name	SAP Afaria	VMware Airwatch	MobileIron	Citrix	Good Technology	IBM
1	Remote wipe	yes	yes	yes	yes	yes	yes

12

http://www.gartner.com/document/2835919?ref=QuickSearch&sthkw=titanium%20phonegap%20nativescript&refval=153604763&qid=03e436a6f09049e60ff1f57a89b1d3ee

2	OS configuration management	yes	yes	yes	yes	yes	yes
3	App provisioning and deprovisioning	yes	yes	yes	yes	yes	yes
4	Cost	$	$$$$	$$	$$	$	$$$
5	IT infrastructure compatibility	High	Good	None	Good	None	None

As can be seen in the table above, all products have the similar technical capabilities. From a non-technical perspective, SAP Afaria shows a better cost-it infrastructure compatibility balance. This product will be proposed to XYZ as the most suitable for Enterprise App Store functionality.

(5) StampDA

The main tasks of the StampDA is the processing and aggregation of captured application logs. Information should be aggregated for two main usages - fraud prevention analytics and marketing analytics, both tasks being suitable for a MapReduce implementation.

Elastic MapReduce, from Amazon AWS, is the product of choice for StampDA because of its low maintenance and licensing cost, easy integration with our components and Amazon's dominant position in the enterprise cloud solutions market.

#	Metric name	On-Premise	Amazon AWS	Microsoft Azure
1	Time to provision	High	Low	Low
2	Maintenance	High	Low	Low
3	Cost	High	Low	Low
4	Elasticity	N/A	Good	Good
5	Market Position	N/A	Leader	2nd
6	Existing skillset	Low	Good	Moderate

Computing hardware

In terms of hosting the new solution, the decision has been made to use public cloud infrastructure in order to reduce maintenance and costs, take advantage of the cloud's elasticity and reliability as well as its nightly automatic backups. The industry leaders in terms of public enterprise cloud solutions have been considered:

#	Metric name	Amazon AWS	Microsoft Azure
1	Cost	$52,000	$30,000
2	Current IT landscape succession	yes	no

Due to insignificant difference in cost (regarding to the total price of a product) AWS is chosen as the hardware for StampCheck & StampGen.

Product manufacturing hardware

In terms of physically embedding the newly generated stamp into XYZ's products, the following vendors have been considered:

#	Metric name	Seagull	Soldair
1	RFID supported	Yes	No
2	NFC supported	No	Yes
3	Cost	$$	Free
4	Ease of integration	Medium	Easy

2.4 Software Solution & Development Approach

The YQTC software platform is comprised of six components. Each component is explained in detail in terms of its function and how it fits into the technical architecture. All references to system components going forward will refer to them by name and system number as illustrated in figure 1.

(1) StampCustomer

StampCustomer is a mobile application which customers use to authenticate XYZ's products. The StampCustomer application functions as a client to the (4) StampCheck application server. This application will need to be distributed on the two most popular mobile operating systems, iOS & Android. The achieve this, we will use the Titanium platform to write one codebase, rather than writing separate native code for each platform. We will write the code in html 5, css and javascript using the Titanium framework to access the native API functionality such as NFC support for each platform, respectively.

As a universal app, StampCustomer will use the mobile device's NFC sensor to read product NFCs and decode them within the client-side application. The decoded NFC, a string of 2953-4296 alphanumeric characters[13], will be sent as part of the 'product authentication' REST API request over HTTPs to the (4) StampCheck server.

StampCustomer also allows customers to digitally "own" products. The mobile application will provide a user interface that prompts users for registration. The UI will collect the name, email address, and password of the user. After this information is entered, it is sent to the (4) StampCheck server as part of the 'registration' REST API request using HTTPS. Subsequent user logins from the mobile application will be transparent to user and will send the email address/password as part of the 'customer account' REST API to the StampCheck server over HTTPS. The mobile application will then display the data returned by the 'account' REST API within an 'account' user interface.

(2) StampEmployee

StampEmployee is the mobile application which XYZ store employees will use. It includes the same NFC product authentication functionality described in the previous paragraphs. In addition, the application will allow employees to transfer digital "ownership" of XYZ's products after a sale or return. StampEmployee also requires mandatory user authentication in order to use the application. A login screen will be presented which prompts employees for their credentials. A username/password is sent to StampCheck via a 'employee auth' REST API over HTTPs.

After a product chip is scanned by the mobile application, it will provide an additional transaction interface that allows a XYZ employee to mark that product as either "sold" or "returned". When marking the product as "sold", the application will prompt for the new owner's (customer) email address. The mobile application will then send the employee UID, customer email address, stamp, & new product status to (4) StampCheck using a 'transaction' REST API over HTTPS. When marking a product as "returned", the application will send the employee UID, stamp, & new product status to (4) StampCheck using a 'transaction' REST API over HTTPS.

[13] http://www.iso.org/iso/home/store/catalogue_ics/catalogue_detail_ics.htm?csnumber=62021

(3) StampStore

StampStore is an enterprise application store that hosts the (2) StampEmployee mobile application. XYZ's IT staff will be able to internally deploy the mobile application to employees using its existing corporate network. The StampStore application will also provide push application update services to deployed mobile employee applications using internal update protocol. When the mobile application package is updated on the StampStore server, mobile clients will receive an update automatically.

(4) StampCheck

StampCheck is a backend application server that responds to mobile applications' NFC product authentication requests and returns product status using REST APIs over HTTPs. In order to validate product authentication requests, StampCheck validates the received NFC data to check for a (matching) code within a product record table in the manufacturing ERP system using SOAP. Simultaneously, the StampCheck application validates that the received code exists in the OWP protocol by sending a query to the (6) StampGen module using a 'block verify' REST API. In addition, the StampCheck application sends the received code as a query to the (5) StampDA module using a 'flag check' REST API to check whether there are any warning flags logged against the respective code. The application uses a simple algorithm generate a single clear "pass/fail" result from the query data sources, and returns the result to the mobile clients.

When new users of the mobile application register, StampCheck is the server that collects their name, email address, and password. The StampCheck application then performs a public blockchain account generation process which is invisible to the end-user. This action is triggered upon receipt of the REST client registration request (section 2) when StampCheck uses a 'generate block account' REST API request to the (6) StampGen module. The module will return an alphanumeric value for the newly created customer blockchain account. This information (registration + account) is securely stored and cached locally to improve performance. It is also stored in a new customer record table XYZ's ERP system using SOAP.

The StampCheck application will also provide a user's' previously purchased products via the mobile application. The StampCheck application receives the account credentials via the 'customer account' REST API from the mobile application. StampCheck authenticates the credentials, and then looks up the client's public blockchain account number. StampCheck then sends a query request to (6) StampGen using a 'previous transactions' REST API. The API will return a list of transactions to StampCheck. The application will then return a list back to the client as a response to the 'customer account' REST API.

Finally, the StampCheck application acts as a "relay" from the mobile employee application clients to the (6) StampGen module. When the mobile employee applications send "sold/returned" status updates (see section 2), StampCheck is the recipient of the requests. As mentioned previously in the mobile client description, employee authentication must occur. StampCheck passes the mobile client credentials to XYZ's federated authentication server to get an "authorized/unauthorized" result. If the client is authorized, StampGen forwards the mobile employee client's transaction request to (6) StampGen for processing.

(5) StampDA

StampDA is a backend module that performs data analytics of mobile client activities. To do this, the StampDA receives log data from the (4) StampCheck server using a simple file transfer protocol such as SCP or RSYNC. The log data is then analyzed using MapReduce and Hive to produce a summary of client activity and detect any unusual usage trends.. An example of an unusual client usage trend would be detecting the same stamp authorization requests coming from multiple clients at multiple IP addresses. StampDA will 'flag' stamp values that are discovered through this analytic process. The StampDA application will provide this "flag data" as a service to the (4) StampCheck application using a 'flag check' REST API.

(6) StampGen

StampGen is a backend application that generates and manages "stamps". A stamp is the unique data identifier that proves the authenticity of XYZ products. The stamps are physically embedded into each XYZ product during manufacturing.

The StampGen application generates stamps by reading product serial numbers and metadata (manufacture date, etc) from the legacy (9) ERP Materials management logistics system using SOAP. The application creates a "salted hash" for each serial number + metadata to obfuscate the information. The StampGen application then creates a code of the salted hash by using the open source Soldair API for node.js. StampGen then stores the salted hash & codes in new product related data fields in the legacy (9) ERP Materials management logistics system, using SOAP. The StampGen component then uses the OWP protocol to perform a transaction. The transaction occurs between two public blockchain addresses (both owned and controlled by XYZ/StampGen) containing the "salted hash" aka "stamp". This creates a permanent record of authenticity in the public blockchain.

The StampGen application provides several services to the StampCheck application server as REST APIs:

- 'generateblockaccount': StampGen receives a customer ID from the (4) StampCheck application via REST. StampGen uses the OWP protocol to create a new public blockchain account and stores the blockchain account number associated with that customer ID within the legacy ERP using SOAP. The API returns a success/fail code back to (4) StampCheck via REST.

- 'block verify': StampGen receives a stamp ID from the (4) StampCheck application via REST. StampGen uses the OWP protocol to query its blockchain account to verify that a public transaction has been validated for that stamp ID. The API returns a pass/fail code back to (4) StampCheck via REST.

- 'transactionstatus': StampGen receives a customer ID & transaction status from the (4) StampCheck application via REST. If the transaction status is set to "sold" StampGen queries the legacy ERP for the public blockchain address associated with the customer ID using SOAP. It then performs a transaction transferring the stamp ID from the XYZ blockchain account to the customer's blockchain account. Alternatively, if the transaction status is set to "return" StampGen performs the same process in reverse. The application will query for the customer blockchain account and then perform

a transaction transferring the stamp ID from the customer to XYZ. The API will return a success/fail code back to (4) StampCheck via REST.

- 'previous transactions': StampGen receives a customer ID from the (4) StampCheck application via REST. StampGen makes a query for the blockchain account number associated with that customer ID from the legacy ERP using SOAP. The application then uses the OWP protocol to query public transactions for the customer's blockchain account. The API returns a list of any stamps owned by the account as serialized JSON data via REST upon success. Upon error, the API shall return a fail code via REST.

2.5 Deployment

The deployment model chosen for this solution is the hybrid cloud model. This model uses a mix between on-premise and cloud-based services, with a seamless and secure integration between the two environment. This combines the flexibility, elasticity and redundancy one gets from the cloud with the low lag, complete control and better security specific to on-prem deployments. The tradeoff however is greater complexity.

(1) StampCustomer

For the consumer end users, the mobile application (1) StampCustomer - will be published to the publicly accessible "Google Play" & "Apple App store". XYZ will have to start a marketing initiative to educate consumers on the availability of the new application and use techniques such as in-store promotion to spread awareness and drive installations.

(2) StampEmployee

In order for corporate users to install the privately distributed (2) StampEmployee mobile application, their mobile devices should be subscribed to the (3) StampStore. Once the mobile application code is completed and stored on the StampStore for distribution, the XYZ IT system administrators push the application to their employees. The instructions must direct users on how to sign the mobile device and then install the application. In addition, XYZ system administrators must configure (10) authentication server with appropriate permissions to allow for both types of mobile clients to authenticate.

(3) StampStore

StampStore is hosted in the cloud and works out-of-the box. The employee's mobile devices should be connected to StampStore using the internal SAP protocols over the public internet.

(4) StampCheck

The mobile applications will connect to the (4) StampCheck application over the public internet via wifi or cellular data. Therefore, the decision was made to host (4) StampCheck in the cloud to leverage the scaling and availability strengths of cloud infrastructure. In this case, we will be using a IAAS (infrastructure as a service) model. Virtual machines will be used to host the (4) StampCheck application. StampCheck will connect the legacy systems using virtual private cloud (VPC).

(5) StampDA

StampDA is hosted in the cloud and works out-of-box. No additional configuration of the SaaS is necessary. EMR will perform its functions after loading the developed java program. StampDA is connected to StampCheck and BW via VPC.

(6) StampGen

To leverage the scaling and availability strengths of cloud infrastructure StampGen is installed in the cloud. In this case, we will be using a IAAS (infrastructure as a service) model. Virtual machines will be used to host the (6) StampGen application.

2.5.1 Cloud-Deployed Systems

The StampCheck application will be hosted on a cloud-based environment. For deployment purposes this has several implications:

A. Load balancers must be configured to distribute traffic among compute nodes

B. Auto-scaling groups must be configured to increase capacity as-needed and reduce (elasticity) capacity during non-peak loads

C. A VPC (virtual private cloud) must be configured between the cloud provider and the on-premise network. This will require a cloud admin to configure the virtual cloud network, and a XYZ network admin to configure the linkage back to the enterprise network including routers, firewalls, and so on. (The VPC is needed since the (4) StampCheck application must communicate with the internal legacy systems)

2.5.2 Legacy Systems

Several enterprise systems will need to be configured as well. Therefore, it is important that a thorough needs assessment is performed, including risk assessment and test planning, including the pilot phase. This may help ensure that ongoing business processes are not negatively affected. Further, it is equally important to validate that the assumptions (configuration changes) will meet system integration

objectives. The following legacy systems and business processes will need to be managed during deployment:

A. (7) Business warehouse must be configured to store new table data provided by (5) StampDA.

B. (8) Internal dashboards must be configured to generate new report queries. The data for new reports will be generated using (4) StampDA. Further - the Fraud prevention business group must be trained on how to access and use the new reports (internal education) and to configure the StampDA fraud sensitivity.

C. (9) ERP Materials management system must be configured with new table record fields for "stamp" data. A WS interface and any necessary authentication details will need to be made available - this will require XYZ's IT team to extend the tables. Further, the manufacturing business group will need to be trained on the changes made to the ERP data fields. These changes will be minor since only one new data field will be added to the existing product records. However, the changes to the product assembly process may be much more disruptive and may require business and manufacturing specialists to be involved with IT as a cross-functional team. It is recommended that this team is formed as an agile team early in the design process in order to gain accurate project requirements and adjust as-needed.

D. (10) Authentication system will need to be configured by XYZ system administrators to create secure "zones" separating user groups with appropriate types of permissions: (Consumers, Employees).

2.6 Integration with existing systems

All system integrations are displayed below in figure 5. Each system integration is illustrated with the communication method or protocol which will be used to tie the systems together.

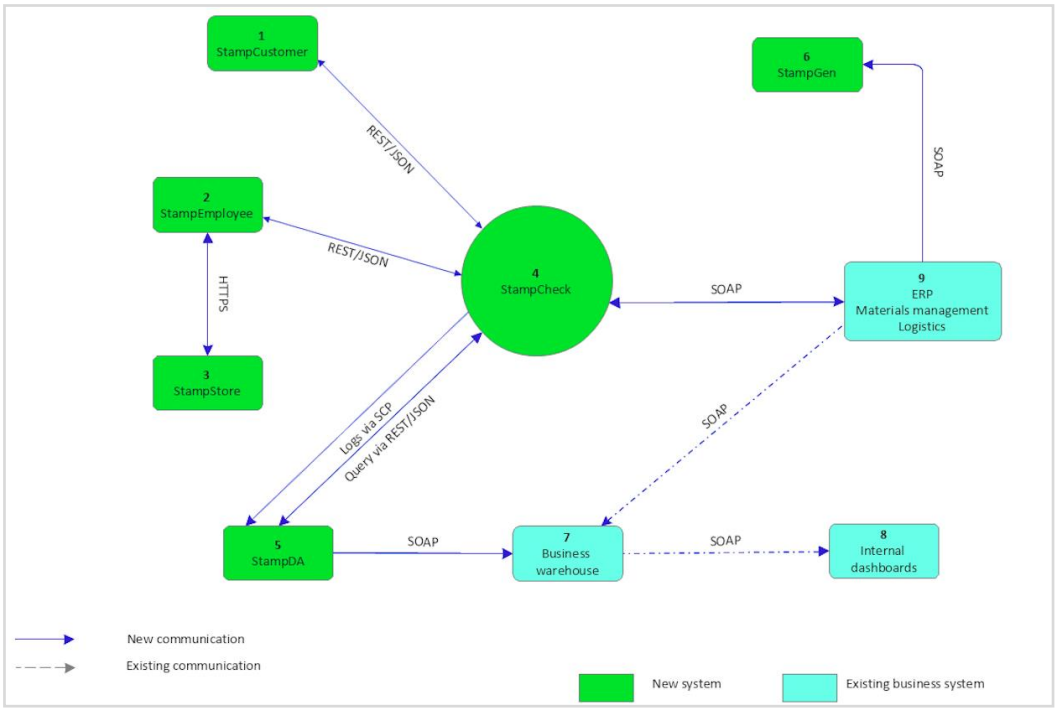

Figure 3. Integration diagram showing communication methods.

The following applications will integrate with existing legacy systems:

(4) StampCheck

Integrates with legacy (9) ERP Materials Management Logistics system. No customization required. A configuration change will be required to add a new data field ("Stamp" ID #) to the product record table. Communication between the (4) StampCheck application and the (9) ERP Materials Management Logistics system will occur by using a SOAP WS. StampCheck must also integrate with the legacy (10) Authentication system. No customization is required, but configuration changes will need to be made to allow mobile application user registrations to be stored and used for (1) StampCustomer authentication. Additionally, store employee records would be used to authenticate (2) StoreEmployee users.

(5) StampDA

Integrates with legacy (7) Business warehouse system. No customization required. Configuration changes will need to be made to the (7) Business warehouse system as follows: New infocubes added to store analytics generated using customer request data. Additionally, the legacy (7) Business warehouse system feeds data to the legacy (8) Internal dashboard system for reporting purposes. Since XYZ expects to receive reports of the newly integrated solution using their existing tools, we must also make some configuration changes to (8) as well in order to generate new reports using the "StampDA" data. In both cases the integration efforts should be possible by making configuration changes and created new infocubes using a BW consultant. All of the data exchange that will occur through integration will be enabled by using SOAP messaging between the existing legacy systems and (5) StampDA.

(6) StampGen

Integrates with (9) ERP Materials Management Logistics system. The (6) StampGen application uses SOAP to read product serial numbers and customer records from the legacy (9) ERP materials management system. After creating stamps, it again uses SOAP to store the generated product "stamp" IDs and public blockchain addresses in the legacy (9) ERP materials management system. This will require minor configuration changes to add a data fields to existing tables. No ERP customization will be required.

2.7 Data Design and Management

The data collected and used by the application has been split by change rate into master data (also called reference data) and transactional data. The master data is data that changes infrequently and contains the attributes and characteristics of each product. Most of the master data is sourced from the (9) ERP Materials Management Logistics system. The transactional data a contains large volumes of data collected via logs and describing all the requests made by the mobile applications to the (4) StampCheck application server. Both the master data and the transactional data are detailed in Figure C of the Appendix.

2.8 Enterprise Data flow

The following diagram describes the flow and capturing of the data between the components of the new system and the existing XYZ systems.:

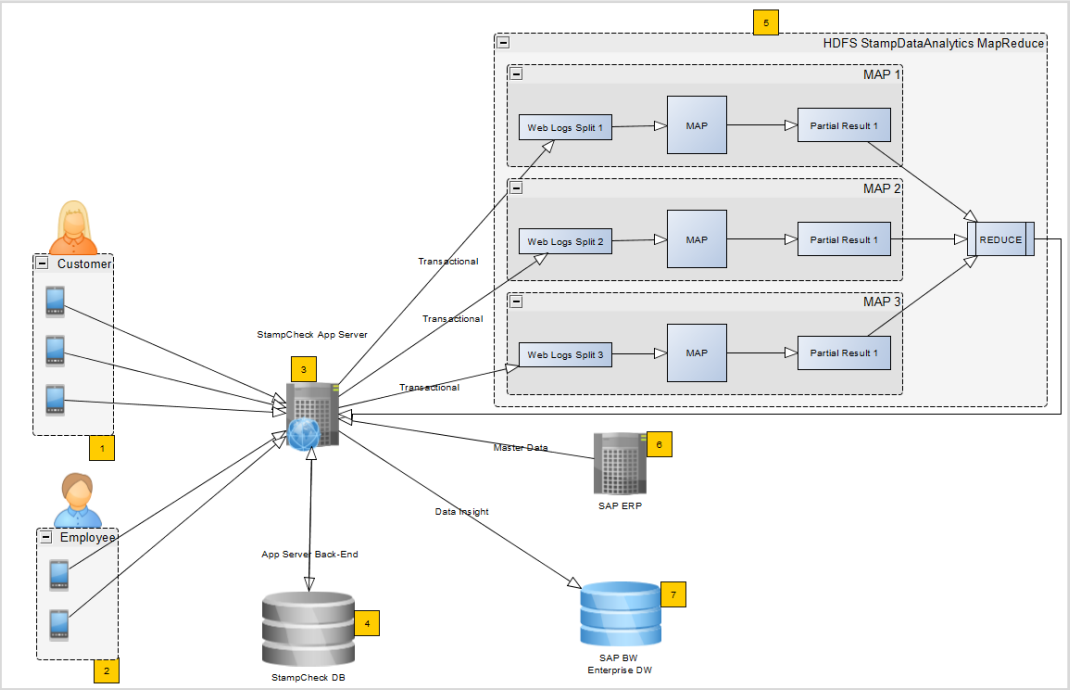

Figure 4. Illustration of data flow between systems.

1. Customer sending request for authenticating a product to the StampCheck application server. Data collected includes: Timestamp of the request, GEO coordinates of the request, Product chip, Mobile device identifier.

2. Employee sending request to authenticate and sell or accept the return of a product to the StampCheck application server. Data collected includes: Timestamp of the request, GEO coordinates of the request, Store and Employee ID of the employee, Product Chip, type of transaction attempted (check, sell, return)

3. StampCheck application server will forward the logs resulting from the (1) and (2) request to the Hadoop MapReduce cluster for processing. The application server will store and combine the results from the Hadoop MapReduce cluster with relevant master data from the SAP ERP system in order to analyze the data and rapidly detect and prevent fraud.

4. Back-end database for the StampCheck web application. Will persist the data required for the authentication and fraud prevention processes.

5. Hadoop MapReduce cluster will consume the web logs and from the StampCheck app server and transform them into actionable information.

6. Existing ERP system that server as the golden source for the product master data

7. Existing Enterprise Data Warehouse and Business Intelligence System that will use the new system as a new source of reporting data.

2.8 The Role of the OWP protocol in the System Architecture

There are multiple blockchains in existence today. In this project, we focus on the OWP protocol and Ubbey network. Distributed nodes participating in the blockchain act to mathematically confirm transactions that have occurred.

Besides the exchange of monetary value, blockchain transactions can store information as well. This information then becomes confirmed and preserved as part of the blockchain, resulting in a very secure means of information archival and data integrity.

From a business context, the OWP protocol allows us to extend significant capabilities to the consumer of XYZ. Using the proposed IT solution, a client running the StampCustomer mobile application can authenticate a product and gain a simple, confident answer on the product's validity. Under the hood, 3 levels of validation are occurring:

A) Mobile application validates that the "Stamp" (salted hash of the serial number) via the StampCheck application.

B) Mobile application validates that no unusual activity has been flagged against the "Stamp" via the StampCheck application. This allows XYZ to add an additional layer of business intelligence and anti-fraud countermeasures.

C) Mobile application receives digital confirmation of product "ownership" after a purchase. This is possible because the company/employee can mark the product as "sold" and transfer ownership (via backend) to the customer's blockchain address. (the customer app will automatically generate an address when installed without user intervention required). This creates an immutable digital ownership record that is linked to the physical good.

The OWP protocol plays a key role in our system architecture. As illustrated below, we present one use-case showing a simplified model focusing on how blockchain is leveraged by our software architecture. Figure 3 displays the "stamp" creation process as well as the validation process. The stamp creation process begins when product serial numbers are read from the (9) ERP system. These numbers are then hashed & "salted" for additional layers of security and privacy. The "salted hashes" are the "stamps". StampGen then conducts a blockchain transaction using two of its own public blockchain addresses. The "stamp" is placed into a protocol header field of the block transaction. When the transaction is completed the "stamp" becomes a permanent record in the blockchain. Moving to the client validation scenario in figure 3 - you will notice that the client checks the "stamp" against the (4) StampCheck application server. The application server checks the validity of the stamp against the OWP protocol and returns a result to the user. Finally, when a product is sold - (2) StampEmployee can authorize StampCheck to transfer ownership to the customer. StampGen then performs another transaction from public address B to public address C.

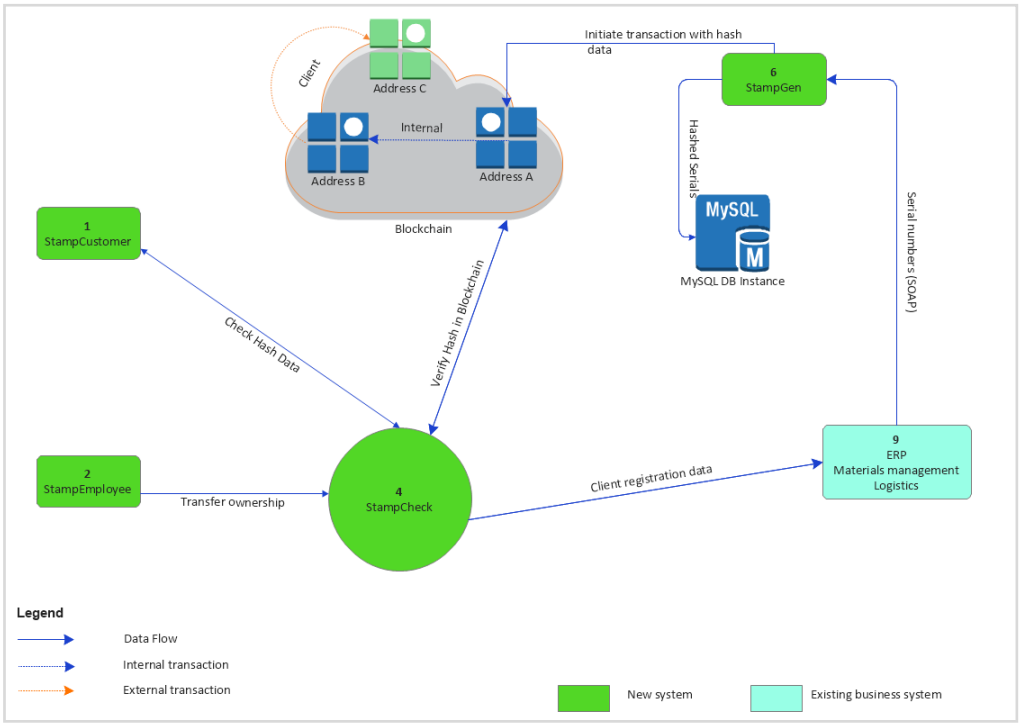

Figure 5. Illustration of a transaction and validation.

As illustrated in the model above, figure 4 demonstrates how the OWP protocol is leveraged during a system use-case. In this scenario, the customer has initiated a product return. XYZ now has a need to recapture the digital asset in addition to the physical product. When (2) StampEmployee scans/marks the product as 'returned', StampCheck authorizes StampGen to perform another blockchain transaction. This time, the transaction originates from public address C back to public address B. The product may not be validated again as being authentic and available for sale.

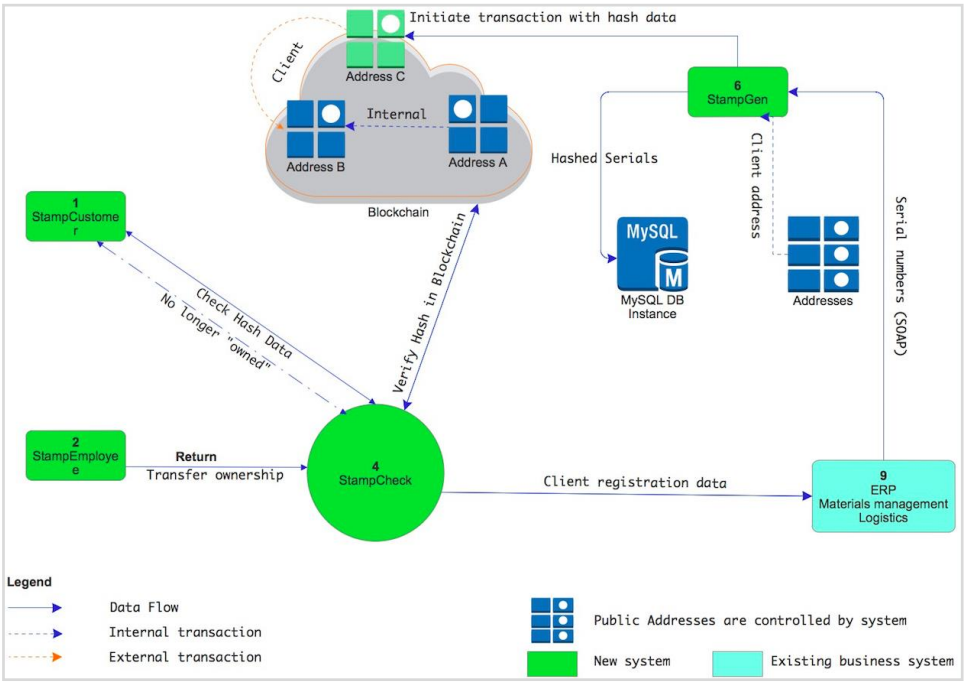

Figure 6. Illustration of a transaction reversal and subsequent validation.

2.9 System Development Metrics

The implementation effort for each system component will be monitored using the following system KPIs:

System component	KPI	Target
(1&2) Mobile applications	Number users (after 24 months, customers-only)	100,000
(4) StampCheck	Maximum response time	5000ms
(4) StampCheck	Average response time	1000ms
(4) StampCheck	# Transactions/hr	5,707*
(5) StampDA	# Transactions/hr	17,121**
(6) StampGen	Time to produce one stamp	< 1sec
(6) StampGen	Price per stamp	< $50

*Estimated at 10x annual number of sales per year (5m/8760*10) to account for load

** Triple StampCheck volume to account for reselling and returns

Part 3. Implementation Plan

The implementation of YQTC's stamp project is divided into six phases. The first four phases are dedicated to system development, and the last two are dedicated to a pilot and go-live phase, respectively. Go-live is divided into two components to minimize the risks of manufacturing interruption.

The project team uses a blended waterfall/agile approach for project phases. In general, project phases are run in a waterfall methodology with parallel execution of phases 1,2 and 3,4. Inside each phase tasks are multiple iterations using the agile methodology.

3.1 Solution Development

The YQTC project's six phases were structured to group related development and integration efforts that should occur logically using the same development resources. The following diagram illustrates the

major milestones due during each phase. Each milestone is numbered (s. Figure below) for the purpose of illustrating its position in the milestone timeline diagram to follow.

3.1.1 Milestone Diagram

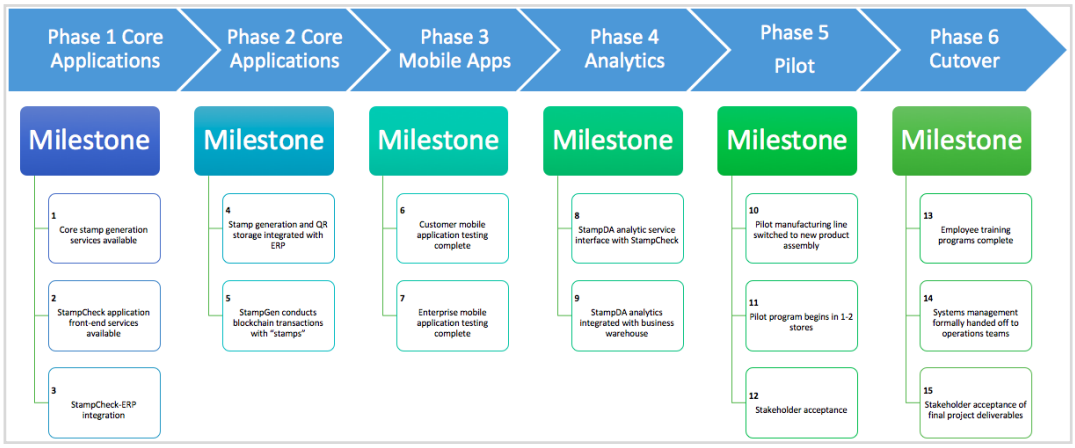

Figure 7. Major project milestones by phase

3.1.2 Deliverables

The following table shows a high-level view of project deliverables, individual task detail and assignment is not included in this table - these would be broken down in the project plan using Microsoft project or similar project management software per XYZ's existing organizational standards on project management and resource scheduling.

Phase / WBS	**Deliverable**
Phase 1: StampGen+StampCheck	
1.1.1	StampGen with an ability to generate the salted hashes (stamps) and generate NFCs from "stamps".
1.1.2	StampCheck with an ability to get data from ERP and present it via web service protocols
1.1.3	QA confirmation of components readiness

Phase 2: StampGen

Customer blockchain addresses are created by StampGen for each customer record in ERP
Initial transactions performed between public addresses A & B for each “stamp”
StampGen is integrated with a ERP
QA confirmation of components readiness

Phase 3: StampCustomer & StampEmployee & StampStore
Customer mobile application
Enterprise mobile application
Enterprise store is created
Enterprise mobile application is uploaded to the enterprise store
QA confirmation of components readiness

Phase 4: StampDA
Provision, install & configure mapreduce cluster
Build mapreduce jobs
Test application service interface between StampCheck and StampDA
StampDA integrated with business warehouse
QA confirmation of components readiness

Phase 5: Pilot
Developer team sign off - system ready to be used in productive mode for business processes
Internal training for beta users
Single manufacturing line in selected BU begins using new system
POS in selected BU begins using new system
Key-user group acceptance of system metric deliverables

Phase 6: "Go-live" cutover
Employee training * (this is expanded in section 3.5 "Training deliverables")
Documentation for mobile applications, StampCheck, StampGen, StampDA
Customer mobile application published to public app store
Marketing and education outreach programs for new product capabilities
YQTC system administration and maintenance is transitioned to XYZ IT operations team

3.1.4 Go-live Strategy

The goal of the pilot phase is to prove the readiness of created IT system of going live using bounded business unit and to prepare XYZ IT team to receive a system for support. The newly created IT system should demonstrate the ability of support of the following business processes:

1. Manufacture the product with a sealed stamp.
2. Check the status of a stamped product.
3. Purchase a stamped product.

4. Return a stamped product.

To identify the results of the pilot phase and make a decision of going live the group of key-users should be created. This group should contain representatives of business process owner, IT Service of XYZ, YQTC team members, BU representatives. This group should be informed about all incidents during the pilot phase.

In order to have a smooth start of the pilot phase those prerequisites should be met:

1. All development should be finished; no critical incidents should remain in work by a development team.
2. Following technical business processes should be tested, and lack of errors should be confirmed by the QA:
    2.1. Create the stamp and push it into the manufacturing system.
    2.2. Check the status of a product using mobile applications.
    2.3. Install the StampEmployee application via Enterprise store.
    2.4. Purchase the product using the StampEmployee.
    2.5. Check the validity of the product using the StampCustomer.
    2.6. Return the product using the StampEmployee.

To fulfill the pilot process following preparations should be done:

1. Identify the Business unit (separate region, country, city or product line) for a pilot phase.
2. Identify the pilot go-live date.
3. Provide the training for manufacturing personnel about changes in the manufacturing business process.
4. Provide the training for POS employees about the StampEmployee and its usage.
5. Provide the Service desk support for the pilot BU.
6. Provide the marketing materials for end customers of the BU.
7. Create the users for POS employees in Authentication system.
8. Install the application to the POS employee's devices using StampStore.

During the pilot phase the system is supported by both YQTC and XYZ teams. The pilot phase should be 1 full collections lifecycle length (approx. 6 months).

After the confirmation of key-users that pilot phase ended successfully the go-live phase starts. The preparations for the go-live phase are the same as for the pilot except the amount of affected business units and supporting personnel of XYZ. During the go-live phase the system is supported fully by XYZ's internal IT team. YQTC employees officially closes the contract. YQTC employees can only participate for warranty purposes.