THE INTEGRITY IN BITCOIN: USING ELLIPTIC CURVE CRYPTOGRAPHY AND HASH FUNCTION

PRESENTED BY

OLUTAYO O. OJUAWO
DEPARTMENT OF COMPUTER SCICENCE
THE FEDERAL POLYTECHNIC, ILARO
olutayo.ojuawo@federalpolyilaro.edu.ng
08038417080

ABSTRACT

Elliptic Curve Cryptography has helped in securing data extensively most especially in a peer-to-peer system by ensuring transactions are free from attacks. Financial institutions are dependent upon as a trusted third party in processing electronic payments when trading on the internet whereas the Bitcoin has no involvement of the financial institutions to process transactions. The motive of this article to depict an application in which the combination of Elliptic Curve Cryptography and Hash function are utilized to maintain the integrity of data and transactions.This paper discusses the Elliptic Curve Cryptography (ECC), hash function, the blockchain technology, and shows how the application of ECDSA and Hash Function secure Bitcoin transaction.

Keywords— Bitcoin; ECC; ECDSA; Hash Function; Cryptography; Blockchain.

1.0 INTRODUCTION

Cryptography can be described as a branch of mathematical science that involve data transformation to render its meaning indecipherable by hiding its content semantically or prevent from unauthorized users. The security requirements controlled by cryptography protocols are confidentiality, availability, and integrity (Kessler, 2016).

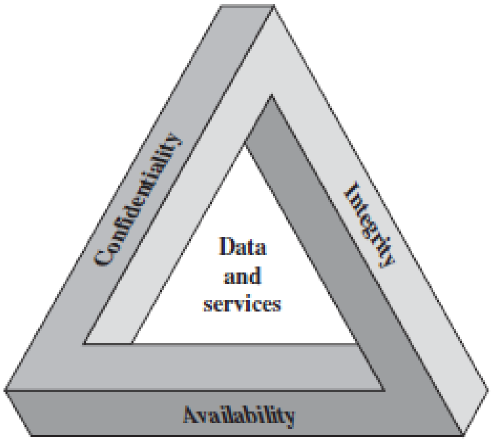

Fig 1: Security Goals

Confidentiality ensures that message cannot be accessed by a third party except the intended recipient. Availability ensures authorized users have access to assets at the right times. Integrity can be broken down into data integrity, non-repudiation origin integrity (authentication). Data integrity ensures data is not modified by unauthorized users. Origin integrity ensures the message origin. Non-repudiation ensures the non-denial of the message by the sender.

The three most important types of cryptographic algorithms are Symmetric Key Algorithm, Asymmetric Key Algorithm, and Hash Functions (Kessler, 2016). The symmetric key algorithms are algorithms that require the use of only one key to encrypt and decrypt data. Both users possess similar private key and must be a secret between then. The symmetric key algorithm is utilized for confidentiality and privacy (Kessler, 2016). Some examples are Blowfish, Advanced Encryption Standard (AES), and Data Encryption Standard (DES).

Asymmetric key algorithms are algorithms based on the public key and the private key, in which each user have both keys. The public key is made accessible to all users and the private key must be undisclosed during encryption. Most times, the public key is used for verification while the private key is for digital signature (Pfluegel, 2016a). The asymmetric key algorithm can be utilized for key exchange, authentication, and non-repudiation (Kessler, 2016).Some examples are RSA, DSA, ECC, Diffie-Hellman, and Elgama. The hash function accepts data as input and gives out an arbitrary fixed-length value called hash or digest (Pfluegel, 2016b). The encrypted information in a hash function is irreversible Hash function is used for message integrity (Kessler, 2016).Some examples are MD5, SHA-1, and SHA-256.

This paper mainly concentrates on how the integrity of transaction in Bitcoin is achieved with the use of ECC and Hash function and how Bitcoin blockchain works.

2.0 LITERATURE REVIEW

2.1 ELLIPTIC CURVE CRYPTOGRAPHY (ECC)

ECC is portrayed as a public key encryption technique in the light of elliptic curve theory that can be used to make faster, more diminutive, and more valuable cryptographic keys. Keys are created by ECC through elliptic curve equation properties instead of the conventional era strategy as the consequence of incomprehensible prime values. The innovation can be utilized with most public key encryption techniques e.g. RSA, and Diffie-Hellman. Some set of researchers describes the security level of ECC to yield a 164-bit key in which 1024-bit key is required by some other frameworks to accomplishing it (Kessler, 2016).The small key size of ECC serves as its main advantage (Anoop, 2007). Elliptic curve cryptography (ECC) can be described as an aspect of mathematics that deals with functions or curves with the format $y^2=x^3+ax+b$. ECDSA is a DSA variant that utilizes ECC. The two protocols commonly utilized by Elliptic Curve Cryptography are ECDH and ECDSA.

2.1.1 Elliptic Curve Discrete Logarithm Problem

The security of ECC is dependent upon the ECDLP difficulty. Given two points A and B on an elliptic curve, such that $mA = B$ and m is scalar, it easy to compute m, if m is adequately substantial then m is the discrete logarithm of B to the base A. Consequently, the fundamental operation required in ECC is the result of getting point B on the curve by multiplying point A with the scalar m(Anoop, 2007).

2.1.2 *Elliptic Curve Public-Key Pairs*

Given some parameters that incorporate a base point G, a prime number w, and an elliptic curve E/Fw on the order of n on E, an elliptic curve key pair (p,Y) comprises of a private key p, which is an arbitrarily chosen non-zero whole number modulo the group order n and a public key $Y = pG$. Point Y is an arbitrarily chosen point in the group generated by G in a multiple base point of G (Bos et al., 2014).

2.1.3 *Elliptic Curve Diffie-Hellman (ECDH)*

This can also be referred to as Elliptic Curve Key Exchange. ECDH is a protocol for key agreement between two users to institute a mutual secret key utilized for private key algorithm (Anoop, 2007). Two key pairs (p_a,Q_a) and (p_b,Q_b) are generated by Alice and Bob respectively to concede on a shared key. The public keys $Q_a$and $Q_b$are exchanged between the two such that point $V = p_aQb = p_bQ_a$ utilizing respective private keys. From point *V*, the shared secret can be gotten through a key derivation function usually connected to its x-coordinate (Bos et al., 2014).

2.1.4 *Elliptic Curve Digital Signature*

This is a variant of DSA in which is utilized by ECC whereby the sender generates a paired key (p,Q) a public verification key $Q = pG$ where p is a private key for digital signature. The sender picks a random integer z to sign message m, such that $1 \leq z \leq n-1$, calculates point $(X_1,Y_1) = zG$, transformation of X_1 to a number and $r = X_1 mod\ n$ is calculated. Message m is hashed to a bit string less than n bit length and transformed to and integer f. The pair (d, c) of integers modulo n is the signature of m where c = z-$1(f+pd)mod\ n$(Bos et al., 2014). It should be noted that z must not be revealed and shouldn't be used for more than a message (per-message secret), and values d and c should be greater than zero. The secret signing key p can be computedas $p \equiv d-1(zc-f)$ $mod\ n$ since d and $c$are specified in the signature and integer f is calculated from the signed message. If z is used to sign two messages m_1 and m_2 with the same sign key p and generating signatures (d,c_1) and (d,c_2), then $z \equiv(c_2-c_1)-1(f_1-f_2)mod\ n,$ that permits recuperation of the secret key (Bos et al., 2014).ECDSA does not work on the message itself, rather it works on the hashed message.

2.2 SECURE HASH FUNCTION

Cryptography hash functions are utilized by numerous cryptography protocols and algorithms, where they have some acute applications with regards to information security. The primary security goal of the hash function is to ensure data is protected from modification by unauthorized users and the origin of the data can be determined which are data integrity, and origin integrity respectively (Pfluegel, 2016a; Pfluegel, 2016b).

2.2.1 *Properties of Secure Hash Function*

- Mixed-transformation: Independent of the data input, the yield is computationally vague from uniform binary streams (Pfluegel, 2016b).
- Collision Resistance: This is the context of digital signing whereby it is difficult to find two distinct inputs, whose corresponding digest is identical, that map the same output. Given m_1 and m_2 and $m_1 \neq m_{2,}$ their output under the application of the hash function, $h(m_1)$ and $h(m_2)$ should not be the same (Pfluegel, 2016b). Discovering two inputs with varying lengths should be difficult such that when supplied to the hash function, it brings about comparatively figured hash values.
- Pre – image Resistance: when given q', it should be difficult to compute the value of q in such a way that $h(q) = q'$ (Pfluegel, 2016b). This denotes that the reversal of hash function must be difficult.
- Practical Efficiency: Given value p, the computation of hash function h can be achieved at a very fast time (polynomial) to achieve $h(p)$ (Pfluegel, 2016b).

2.2.2 *Utilization of Secure Hash Functions for Digital Signatures*

The process whereby the sender's private key is utilized to encrypt the message to be sent for sender's identity and message verification is called Digital Signature. According to Pfluegel(2016a), the procedure in using secure hash functions with a digital signature for message encryption are:

- A message digest $h(q)$ must be created before sending the message.

- The digital signature process whereby the private key of the sender is applied to the message digest i.e $E(K_{priv}, h(q))$.
- The receiver gets the digital signature along with the original message q i.e $q \parallel E(K_{priv}, h(q))$.
- The receiver decrypts the digital signature to get a message digest, and a hash function is applied to the real message to get another message digest, then the two message digests are compared (Pfluegel, 2016a).

2.3 BLOCKCHAIN TECHNOLOGY

The original the name given to the design supporting the operation of the digital currency Bitcoin is blockchain technology (Crosby et al, 2016). Bitcoin's block chain's operation is aimed at recording the transaction made between network members into a block with limited capacity and announcement of the transaction to other network users (Ammous, 2016; Wright & Filippi, 2015).A blockchain is a public of all transactions or distributed database of records,or digital events that have been executed and shared among partaking parties (Crosby et al, 2016;Pilkington, 2016).The consensus of most of the participants in the network verifies each transaction in the public ledger. Once information is entered, it can never be deleted(Crosby et al, 2016; Wright &De-Filippi, 2015). The block chain contains a certain and undeniable record of each and every exchange ever constructed. The First block in a blockchain is called Genesis. The figure 2 below displays a blockchain that consists of a block of at least one new transactions gathered into the information part of the transaction block. Transactions are likewise chained together.

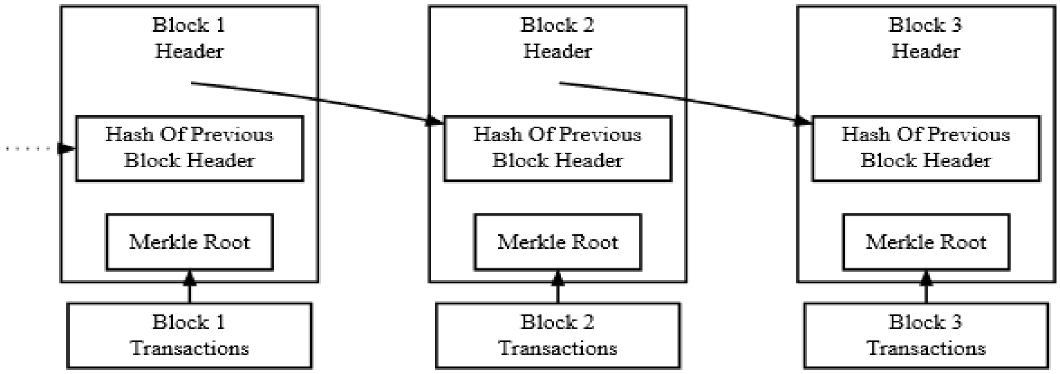

Fig 2: Bitcoin Block Chain (merkle, 2016)

Duplicates of every transaction are hashed, the hashes are matched, hashed again, and matched again (Bitcoin Development Guide, 2016). This process is done continuously until it remains a solitary hash which is the Merkle root of the Merkle tree. A Merkle tree is a tree developed by hashing combined data (the leaves), then matching and hashing the outcomes until a solitary hash remains called the Merkle root (Merkle, 2016). In Bitcoin, the leaves are quite often transactions from a single block (Merkle, 2016).

The block header stores the Merkle root and the hash of previous block header is stored in each block with chained blocks to each other. The process guarantees that a transaction cannot be altered unless the block that records it and all subsequent blocks are altered.

The software of Bitcoin wallet gives the imprint that bitcoin truly moves from transaction to transaction even though satoshis are transferred from wallet to wallet. A satoshi a Bitcoin value typically measured in fractions of a bitcoin which is equivalent to 0.000000001 Bitcoin i.e a Bitcoin is equal to 100,000,000 satoshis(Bitcoin Denominations, 2016). The satoshis formerly received through one or more previous transactions is spent by each transaction and the input of a transaction is the output of the previous transaction (Bitcoin Development Guide, 2016).

2.4 BITCOIN MINING

Mining can be described as a method of creating valid Bitcoin blocks that require exhibiting proof of work. In which devices that mine or individuals who possess devices that mine is called miners (Anoop, 2007;Crosby et al, 2016). Mining facilitates the addition of new blocks to the block chain which makes the transaction difficult to modify. The two forms of mining are discussed below:

a. *Solo Mining:* This is the process whereby the miner generates new blocks all alone with the returns from the block reward and transaction fees having both to himself and permitting the miner to receive huge payments having a higher variance between payments. The solo mining workflow is displayed in figure 3 below

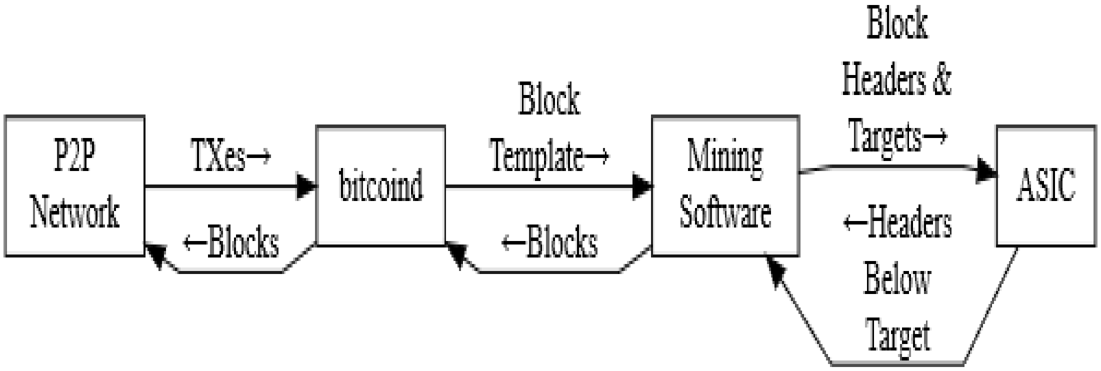

Fig. 3: Bitcoin Solo Mining Workflow (Crosby et al, 2016).

b. *Pool Mining*: This is when the miner puts resources together with other miners to discover blocks more frequently with the returns being shared among miners depending on the measure of the hash power contributed by each miner that permits the miner to get little payments with shorter time within payments (Crosby et al, 2016).

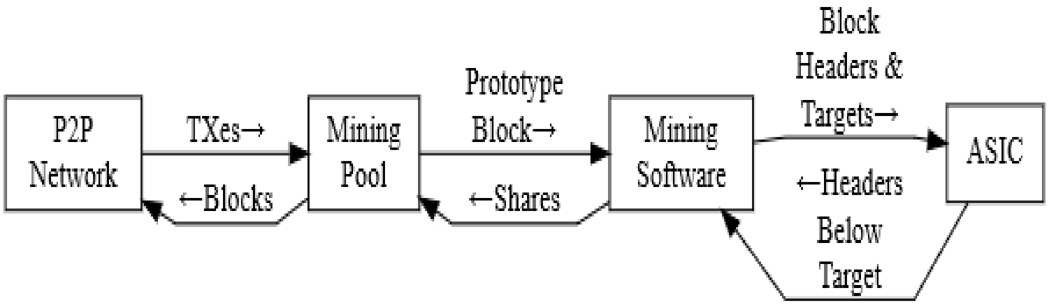

Fig. 4: Bitcoin Pool Mining Workflow (Crosby et el 2016).

2.5 BITCOIN CRYPTOGRAPHY

Bitcoin can be described as a virtual currency that has been contrived for anonymous payments made completely autonomously of governments and banks (Segendorf, 2014). Bitcoin is a decentralized digital currency comprised of encrypted data in block form which exists in a purely electronic form and is traded through a secure peer to peer file transfer framework (Banafa, 2014). Bitcoin is a cryptocurrency in which the existence of Bitcoin depends on cryptography. Bitcoin makes use of public key cryptography. The Electronic Curve Cryptography is utilized by Bitcoin to secure transaction and the cryptographic algorithm used by Bitcoin is the ECDSA. Bitcoin also utilizes Secure Hash function to ensure that a third party does not intercept transaction between Bitcoin users. The ECDSA is utilized by Bitcoin to make sure thatthe legitimate owners spend funds and to secure transactions by proving the sent message is from the sender and cannot be denied by the sender (non-repudiation). The sender is in possession of the private key and owns the Bitcoins and transaction can be verified on the network by any user (Antonopoulos, 2014).The main key secret behind Bitcoin protocol is digital signature and verification.The Security goal of Bitcoin is Integritywhich ensures that transactions are not tampered with by a third party.

A Bitcoin's private key is a number between 1 and 1.158x 10^{77}. This figure is created utilizing a secure random number generator that is then fed into the SHA-256 hashing calculation. The SHA-256 hashing algorithm takes a series of values and yields a 256-bit number which then must be verified whether it is less than 1.158x 10^{77}.

3.0 METHODOLOGY

This paper applies the Hash function techniques and the Elliptic Curve Digital Signature Algorithm (ECDSA) to prove the integrity of Bitcoin.

4.0 RESULTS

The Bitcoin protocol uses the secure hash function cryptographic technique ensure data and origin integrity. The transfer of data from a point to another point requires a sender and a receiver. In cryptography, Alice and Bob are names being used to represent the sender and the receiver respectively.

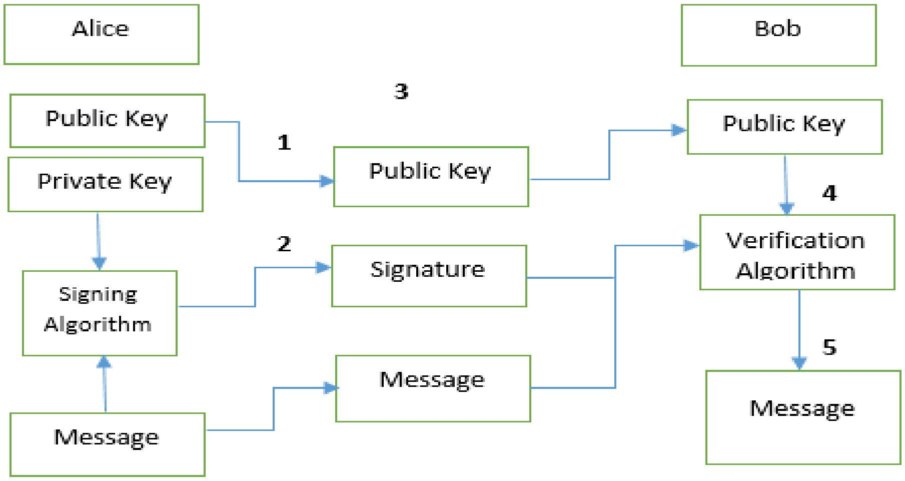

Fig 5: Bitcoin Public Key Cryptography.

From the above figure, the private key and the public key are generated by the Alice (the sender) and the message is signed with Alice's private key to generate the signature. Alice sends the signature, public key and the message to the peer-to-peer network. Bob (the receiver) now use the verification algorithm to verify that the sent message was signed by Alice, and this can be done by the private key holder to the sent public key.

The public key is created with the private key and the address is created with the public key using elliptic curves concept and modular arithmetic in finite fields. The elliptic curve for Bitcoin is defined by $y^2=x^3+7$ mod n where $n=1.158x10_{77}$.

The following scenario describes how Bitcoin transaction between Alice and Bob is sent and verified using Elliptic Curve Digital Signature Algorithm protocols and hash function:

Alice takes the message and converted to a number by hashing the message and multiply it by the generator point as shown in equation 1.1

$M_p\ (X_{m,}\ Y_m) = H(m) * G(X_g,Y_g)$………… (1.1)

A random number is chosen and a random point is created by multiplying the random number by the generator point.

$R_p\ (X_r, Y_r) = (R_n)* G(X_g, Y_g)$ …………….... (1.2).

The sender (Alice) takes a random point at the X coordinate and multiply by the public key as shown in equation 1.3. The public key is a point derived from the private key and the generator point.

$K_{Pub}(X_k, Y_k) = (X_m) * K_{Pub}(X_{pub}, Y_{pub}) = (X_m)*K_{Priv} * G(X_g, Y_g)$ …….….……. (1.3)

The abovedenotesequation that the public key o the coordinate is equal to the private key, $K_{Priv,}$ multiplied by the generator point, $G(X_g,Y_g)$.

$K_{Pub}(X_{pub}, Y_{pub}) = K_{Priv} * G(X_g, Y_g)$ …………………. (1.4)

The random point X coordinate, the random number for generating a random point, the signature factor using the Bitcoin private key and the hashed message, H(m), are created by Alice (the sender) as shown in equation 1.5 below.

$SF = (H(m) + Xr * Pr) / R_n mod n$ ………………. (1.5)

The sender (Alice) sends the public key, message and the signature (SF,X_r) to the receiver (Bob). Bob takes the following steps by making the following calculations:

Bob hash the received message and divide it by the signature factor as shown in equation 2.1

$U_1 = H(m)/SF$……………………….. (2.1)

The random number X_r is now divided by the signature function SF as shown below.

$U2 = X_r / SF$ ..……………………….. (2.2)

Bob then calculates the random point as calculated in equation 1.2 above.

$R_p(X_r, Y_r) = (U_1) * G(X_g, Y_g) + (U_2) * K_{Pub}(X_{pub}, Y_{pub})$ … (2.3)

The signature is valid if the $X_r$of Alice is equivalent to the X_r of Bob and the transaction was sent by Alice (the holder of the private key). Substituting equations 1.4, 15, 2.1, and 2.2 into equation 2.3, the result is shown below:

$R_p (X_r, Yr) = ((H(m)/ SF) * G(X_g, Y_g)) + ((X_r/ SF) * K_{Priv} * G(X_g, Y_g))$ ……………. (3.1)

$R_p (X_r, Yr) = ((H(m) + (X_r * K_{Priv})) * (G(X_g, Y_g))) / SF$ …………………………….. (3.2)

$R_p(X_r, Y_r) = [((H(m) + (X_r * K_{Priv})) / (H(m) + (X_r * K_{Priv}))] * R_n * G(X_g, Y_g) = R_n * G(X_g, Y_g)$ …….. (3.3)

Equation 3.3 above shows that the transaction is valid and the signature is from Alice (the holder of the private key).

6.0 CONCLUSION

This paper discussed the Elliptic Curve Cryptography, properties of ECC and Hash Function. Application of ECDSA, and the Bitcoin cryptographic technology which describes how Bitcoin uses ECDSA to send and verify transactions. The digital signature in the Bitcoin transaction cryptography makes it secure, safe, and instant high level of transparency. The Bitcoin blockchain technology was also explained in a simplified form and the Bitcoin mining which describes how a valid Bitcoin blocks are created. Transactions can be trusted by users since the Bitcoin protocol, gets rid of the need for a trusted third party. The elliptic curve cryptography keys are superior to the DSA and RSA keys because the ECC algorithm is much harder to break and can also utilize smaller length of keys. An ECC key of 256 bits is as secure as a RSA key with 3248 bits.

REFERENCES

Ammous, S.H. (2016) Blockchain Technology: What is it good for? Browser Download This Paper.

Anoop, M. (2007) Elliptic curve cryptography. An Implementation Guide.

Antonopoulos, A.M. (2014) Mastering Bitcoin: unlocking digital cryptocurrencies. : " O'Reilly Media, Inc."

Banafa, A. (2014) What is Bitcoin?

Bitcoin Denominations (2016) Available at:https://bitcoin.org/en/glossary/denominations Accessed at 17/11/2016.

Bitcoin Development (2016) GuideAvailable at:https://bitcoin.org/en/developer-guide#block-chain Accessed at 11/18/2016.

Bos, J., Halderman, J., Heninger, N., Moore, J., and Naehrig, M. 2014 Elliptic curve cryptography in practice.
International Conference on Financial Cryptography and Data Security: Springer.

Crosby, M., Pattanayak, P., Verma, S., and Kalyanaraman, V. (2016) BlockChain Technology: Beyond Bitcoin. Applied Innovation 2016:6.

Kessler, G.C. (2016) An overview of cryptography.

Merkle Tree. (2016); Available at:https://bitcoin.org/en/glossary/merkle-tree Accessed at: 11/17

Pfluegel, E. (2016a) Public Key Cryptography.; Available at: https://studyspace.kingston.ac.uk/bbcswebdav/pid-4032463-dt-content-rid-7646399_2/courses/CI7100-ALL_TB1_6/CI7100%20Public%20Key%20Cryptography%20Lecture.pdfAccessed at 12/01.

Pfluegel, E. (2016b) Secure Hash Functions.; Available at: https://studyspace.kingston.ac.uk/bbcswebdav/pid-4032461-dt-content-rid-7646395_2/courses/CI7100- ALL_TB1_6/CI7100%20Hash%20Functions%20Lecture.pdf Accessed at 11/17.

Pilkington, M. (2016) Blockchain technology: principles and applications. Research Handbook on Digital Transformations, edited by F.Xavier Olleros and Majlinda Zhegu.Edward Elgar.

Segendorf, B. (2014) What is bitcoin. Sveriges Riksbank Economic Review;2:71-87.

Wright, A., and De-Filippi, P. (2015) Decentralized blockchain technology and the rise of Lexcryptographia. Available atnSSRN 2580664.