2020 2nd Annual International Conference on Information and Sciences (AiCIS) | 978-1-7281-9169-0/20/$31.00 ©2020 IEEE | DOI: 10.1109/AICIS51645.2020.00031

Implementing a Distributed Certificate Authority Using Elliptic Curve Cryptography for Big Data Environment

1st Omar Salah F. Shareef
University of Fallujah
Fallujah, Iraq
omar.alshareef@uofallujah.edu.iq

2nd Ali Makki Sagheer
Al-Qalam University College
Kirkuk, Iraq
prof.ali@alqalam.edu.iq

Abstract*—*Improving security is necessary to secure the exchange of confidential documents and protection against unauthorized accesses in the big data environment. Sharing data across a large-scale distributed system by a group of non-trusted parties can lead to illegal modification and unauthorized access. Thus, a secure and fast authentication service to authenticate the user connecting to the system is imperative. Based on Elliptic Curve Cryptography (ECC), a Distributed Certificate Authority (DCA) scheme has been designed and implemented which can efficiently improve the authentication of big data in a distributed environment. The role of authentication is distributed amongst the many nodes included in groups. Each group has its own revocation and signature list instead of each node, and this way reduces system overhead and avoids consuming much time when verifying the nodes without a centralized server. The proposed scheme is demonstrated on a big dataset of social networks. The scheme has been analyzed on the basis of security criteria and compare it with previous schemes to evaluate its performance. The experiment shows that the proposed scheme is capable of consumes less memory and reduces time consumption.

Keywords*— *Public Key Infrastructure, Distributed Certificate Authority, Elliptic Curve Cryptography, Big Data Security, Distributed Network

I. Introduction

In the era of information technology, data have become an essential part of people's daily lives. The information becomes available and plays an important part in the Internet, cloud computing and mobile appliances. Massive data are created from social networks, such as Facebook, Twitter, WhatsApp and YouTube [1]. The amount of data generated daily is increasing exponentially at a minimum of 2.5 quintillion bytes (10^{18}). These data come from several sources, including social networks, video and sound contents, websites and online sources [2]. Moreover, 7.6 billion mobile subscribers are currently in existence and are anticipated to reach 9 billion by 2020 [3]. The use and capabilities of mobile smartphones are increasing every day. For example, smartphones can be used to not only make phone calls but also for browsing the web, checking emails, playing games, shopping and transferring money. Thus, the dependence on using smartphones has become a large part of accessing, storing and processing various types of data (personal, medical and business), which are usually confidential and sensitive. The number of Internet users is approximately 3.4 billion, which is increasing rapidly at a rate of 150% each year [4]. Compared with 20 years ago, each second now has data more than what is on the whole Internet [5]. However, big and complicated datasets are difficult to manage using the classical system of database management. Thus, the term 'big data' has appeared and is used in people's lives at present. The term is developing and is used to characterize a large amount of data that can be semi-structured and unstructured and contain information. In particular, big data have the following characteristics: volume, velocity, and variety, which are primary attributes and are collectively called 3Vs; then value and veracity were added to 3Vs, all of which comprise the 5Vs [6] [7]. However, big data have resulted in new problems not only the characteristics of data but also their security. The occurrence of big data has brought new challenges regarding data security. The gathering, storage, manipulation and retention of huge quantities of data have led to critical security and privacy considerations [8].

The access to data needs to be controlled to ensure that non-eligible entities cannot tamper with or access the data. Improving the security and authentication of sensitive data can give companies new business opportunities. Thus, close attention has been drawn to secure data firmly from any unauthorized access.

Sharing big data is a major open and crucial issue [9][10]. In particular, sharing data stored on distributed cloud storage by a group of non-trusted parties can lead to illegal alteration and impersonated publication and retrieval and unauthorized access [11] [12].

Various technologies, such as Kerberos [13], X.509 [14] and access control policy [15], can be used to deal with big data authentication. However, two methods, namely, symmetric and asymmetric key are needed to design a secure system. The symmetric key has limitations in host-centric authentication and scalability and is thus inappropriate for the data-centric environment [11]. In the asymmetric method, the certificate should be restored from centralized servers before the process of authentication; thus, most of the communication is overhead. In this way, the delays in the authentication process increase. However, the efficiency of this function is difficult to achieve in a distributed environment.

The most challenge of big data that uses a complicated distributed system, is the complexity of managing wide implementation. Authentication should be managed by a flexible, robust and scalable system that denies a malicious user from the gain access to big data. Therefore, new approaches to security are required to overcome the security flaws in the existing implementation.

Based on elliptic curve cryptography (ECC), we design and implement a distributed certificate authority (DCA) scheme that can efficiently improve the authentication of big data in a distributed environment. The role of authentication is distributed amongst the many nodes included in groups.

978-1-7281-9169-0/20/$31.00 ©2020 IEEE
DOI 10.1109/AiCIS51645.2020.00031

Each group has its own revocation and signature list instead of each node, which reduces the requirement of storage of nodes and avoids consuming much time when verifying the nodes. This scheme is managed centrally and provides the capabilities for issuing, validating and revoking digital certificates. The proposed scheme can also present a verification process that can perform authentication between nodes without requiring a centralized server. When a node needs to connect with the other, they only have to verify the digital signatures of the other node (that are saved in a signature list), which are created from a previous verification process from the CA. The method shortens the steps and reduces time consumption.

II. RELATED WORK

[22] proposed self-organized public key management for MANETs similar to the PGP. Every node of MANET sign two types of certificates: one for itself and the other for other nodes it trusts. Each network node has its certification authority and issues certificates to other nodes by establishing chains of trust. The proposed approach is suitable for MANET. However, the approach relies on transitive trust relationships, which perhaps insufficient for some applications.

DCA has been used by many solutions to provide security services in mobile ad hoc networks (MANETs). In [19] a certificate authority cluster-based architecture by dividing the entire network into clusters is presented. Every head of the cluster can maintain a CA information table with details on the certification authority. The DCA information is updated amongst heads, and this way reduces the response delay and the overhead.

[21] used the cluster-based architecture of DCA to establish a secure connection that is adapted to the ad hoc networks. The secure network is established using the randomly generated session key.

Along with a redundancy technique, [20] use Shamir's secret sharing scheme to support certificate renewal and revocation. The scheme utilizes the routing with forwarding behavior to detect the problems with the routing protocols and data. The CA mentions the initiation and expiry times whilst approving certificates to the nodes.

A fully and dynamic DCA scheme based on ECC and trust graphs and threshold cryptography has been used for MANETs [23]. DCAs are the major approaches that can be used to issue, revoke and handle certificates in MANETs. The scheme takes less computational overhead and provides the same level of security as RSA.

In [16] a framework for security on G-Hadoop is suggested (a basic Hadoop extension that allows running MapReduce tasks on numerous clusters) that integrates a number of solutions for security. The framework is modeled on the basis of the concept of Signe Sign-On, which simplifies user authentication and computing functions of MapReduce. Thus, this framework gives users the ability to access various clusters with the same identifier of the account with a large-scale distributed system. In addition, the framework ensures the protection of privacy according to several security solutions, such as public-key encryption and the SSL protocol of authentication. As a result, this framework provides sufficient access control and protection from traditional attacks.

A token-based authentication, with the use of ECC particularly for a Hadoop distributed file system (HDFS), has been presented in [17]. The presented approach ensures the protection of private data in HDFS from attacks of impersonation. The clients are authenticated via "DataNode" through tokens of block access.

In-network big data-sharing environment, provision, and retrieval are susceptible to malicious requests and attacks of data poisoning. [11] researched the issue of unauthorized accessing, unauthorized modifications and attacks of impersonation. Moreover, they presented an innovative DAAS for enabling distributed verifiability and flexibility of authorization for users and publishers.

In [18] a blockchain-based approach is presented for enhancing big data authentication in distributed environments. They utilized the advantages of block-chain, such as anonymity, decentralization, open autonomy and unreadable information, to realize the zero-point failure system without password and anonymous authentication.

[24] proposed a secure and verifiable access control system that implements a Certificate Authority coupled with Role-Based Access Control to provide the permissions to the user to access data. The digital certificate is certified, issued, and revoked by a central administrator; as a result, the certificate is sent based on the role of the user. The proposed scheme has been demonstrated on a big dataset.

However, our literature investigation above reveals that no study applies a model in a comprehensive way for DCA in big data. In this work, we adopt the certificate authority management to improve the performance of big data authentication in a distributed environment.

III. PROPOSED SCHEME

In the proposed scheme, the DCA is applied in Friendster social network and ground-truth communities [27] over ECC. The dataset consists of many groups, each of which has many nodes that are connected with one another. Each node is a member of one or more groups, which implies it either belongs to at least one group or is connected to other nodes that belong to at least one group. All groups will form the entire network. This scheme is modeled on the distribution of nodes in these groups. Each group has its Certificate Revocation List (CRL) as each group has its own Digital Signature List (DSL). Thus, the communication process within the group can be managed. When the client requests access to a specific group, the CA verifies the user id and issues digital certificates. The CA generates a digital certificate using ECC. When the certificate is verified from the CA for the first time, its digital signature is stored in the DSL at a specific group where verification occurred. This step facilitates the mutual authentication between clients by checking the digital signature that is stored in the DSL in each group. Figure 1 illustrates the proposed system between the client-side and the issuer side.

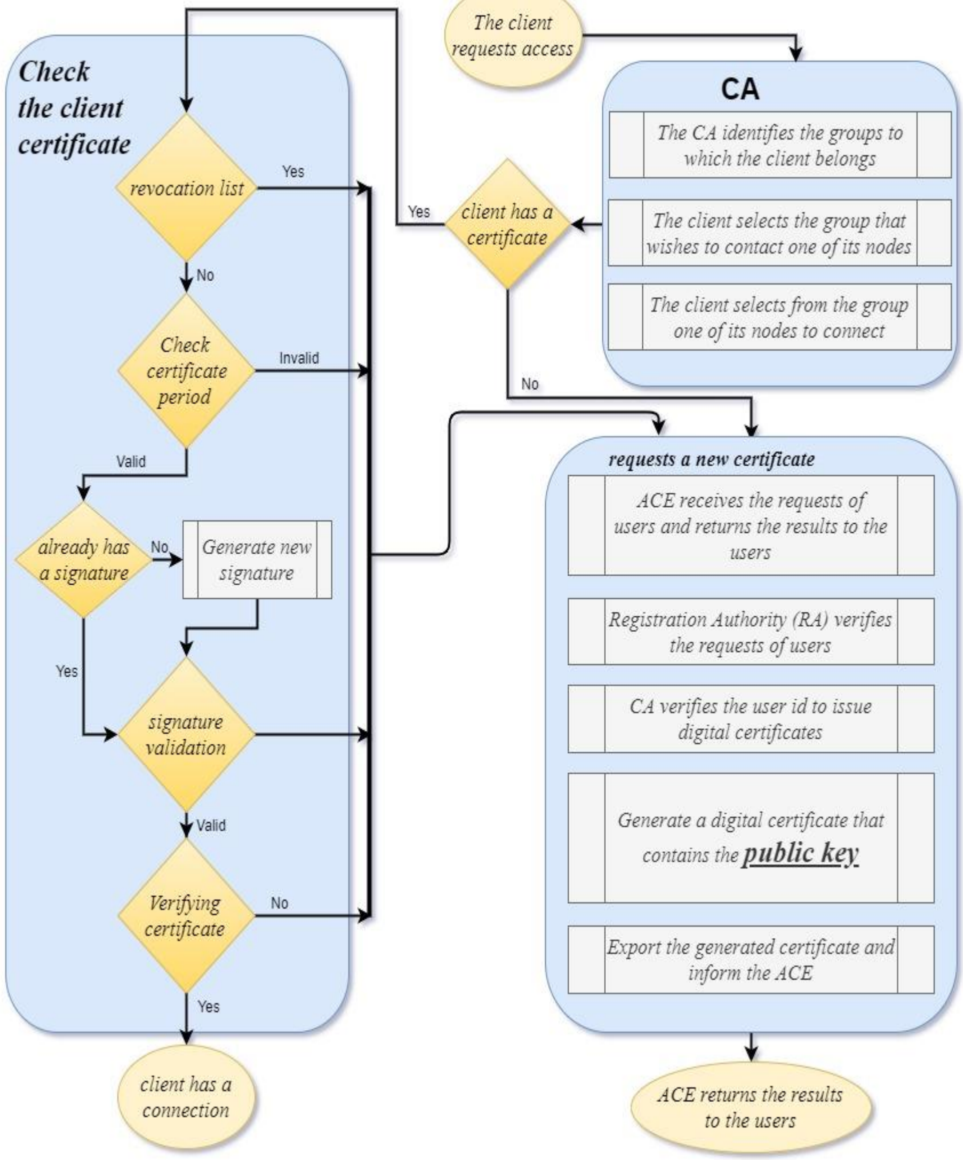
Check the client certificate
revocation list
Yes
No
Check certificate period
Invalid
Valid
already has a signature
No
Generate new signature
Yes
signature validation
Valid
Verifying certificate
No
Yes
client has a connection
The client requests access
CA
The CA identifies the groups to which the client belongs
The client selects the group that wishes to contact one of its nodes
The client selects from the group one of its nodes to connect
client has a certificate
Yes
No
requests a new certificate
ACE receives the requests of users and returns the results to the users
Registration Authority (RA) verifies the requests of users
CA verifies the user id to issue digital certificates
Generate a digital certificate that contains the public key
Export the generated certificate and inform the ACE
ACE returns the results to the users

Fig. 1. The Block Diagram of the Proposed Model

The operations of the proposed model are a set of statements between client and certificate issuer, these sequences of statements can be represented as follows:

A. Issuer Side

The system begins when the client requests a new certificate or request for accessing (where the client already has a certificate). When the client has a certificate, the issuer should verify this certificate to authenticate the client. These operations are described in detail as following:

1. Client request a new certificate: The client will request a new certificate for either new client entering or for the revoked certificate. The work can be divided into three blocks: the class to generate the CA's own certificate, the class to issue a certificate to the client, and the class to validate the client certificate:

- Certificate generation

  X.509v3 is used in this model to confirm the identification of users with a public key. the certificate includes information concerning the identity to which a certificate is issued and the identity which has issued it. The information in X.509 certificates are including the; Subject, Serial number, Signature algorithm, Issuer, Public key, Thumbprint algorithm, Validation, and Thumbprints.

  This class is responsible for generating a root certificate and PKCS#12 key store. Root certificate contains the public key of a CA, and PKCS#12 is a standard that specifies a portable format for storing the private key. Both certificate and private key are included in PKCS#12. The key parameters are shown in Table 1 are used in this class.

TABLE II. CLASS TO GENERATE CA CERTIFICATE

parameter	value
ECParameter	prime192v1
KeyPairGenerator	ECDSA
X509v3CertificateGenerator	subject of CA
Signature Algorithm	SHA256withECDSA
KeyStore	PKCS#12

- Certificate issuance

  The CA class issues user certificates to clients by using a secret key and subject information. The subject information of the purchaser with a pair of keys are joined to issue an X.509 certificate. The certificate is signed using the secret key of the CA with SHA256WithECDSA as the algorithm of signature and provided to the user. Certificate path (root certificate subject is the sub-certificate issuer) is made up of the certificate of the CA and the certificate of the user with its secret key and maybe encapsulated as PKCS#12.

2. The client has a certificate and requests accessing: The clients will ask for access when they already have a certificate, thus, the operations of the issuer are client certificate validation, certificate renewal, and certificate revocation.

- Certificate verification

  This class used to validate the client certificate for CA. The aim is to check the validity of the client digital certificate. In this sense, the CA reads the certificates of the client, including the path of the certificates, and checks the certificates in the following manner:

  a) Checking the validity duration; otherwise, it is not valid;

  b) Checking the certificate signature through utilizing the public key of root certificate; else, it is not authentic;

  c) Checking the serial number of the certificate, and if it is not within the recent Revocation List; the certificate is not valid in the case where it is true.

- Certificate renewal

  The given certificate is valid for a limited time, the client has to renew the certificate before expiry. The client requests from the issuer a certificate renewal, the issuer checks whether the certificate has not expired and not been revoked. The issuer ignores the request as long as the certificate has been revoked; or else, the request is granted. Then, the issuer produces a new certificate with limited time.

- Certificate revocation

  This mechanism is based on a distributed manner, the process of revocation of a certificate for a particular node will not be exhaustive in all communities but will be in a specific group. This feature indicates that the certificate node can work in certain groups and cannot work in other groups.

B. Client-Side

The client-side on the system used the proposed mechanism of Signature Verification. This method aims to mutually authenticate between the clients without connecting again to the centralized servers to facilitate authentication and ensure the privacy of communications between clients. However, this scheme is modeled on the distribution of nodes in these groups. Each group has its Certificate Revocation List (CRL) as each group has its own Digital Signature List (DSL). Thus, the communication process within the group can be managed. When the client requests access to a specific group, the CA verifies the user id and issues digital certificates.

The CA generates a digital certificate using ECC. When the certificate is verified from the CA for the first time, its digital signature will be stored in the DSL at a specific group where verification occurred. This step facilitates the mutual authentication between clients by checking the digital signature that is stored in the DSL in each group. Consequently, the client does not need to request a new authentication from CA and stores system CRL table at each group this will reduce time consumption and consumes less memory. Figure 2 illustrates the signature verification process.

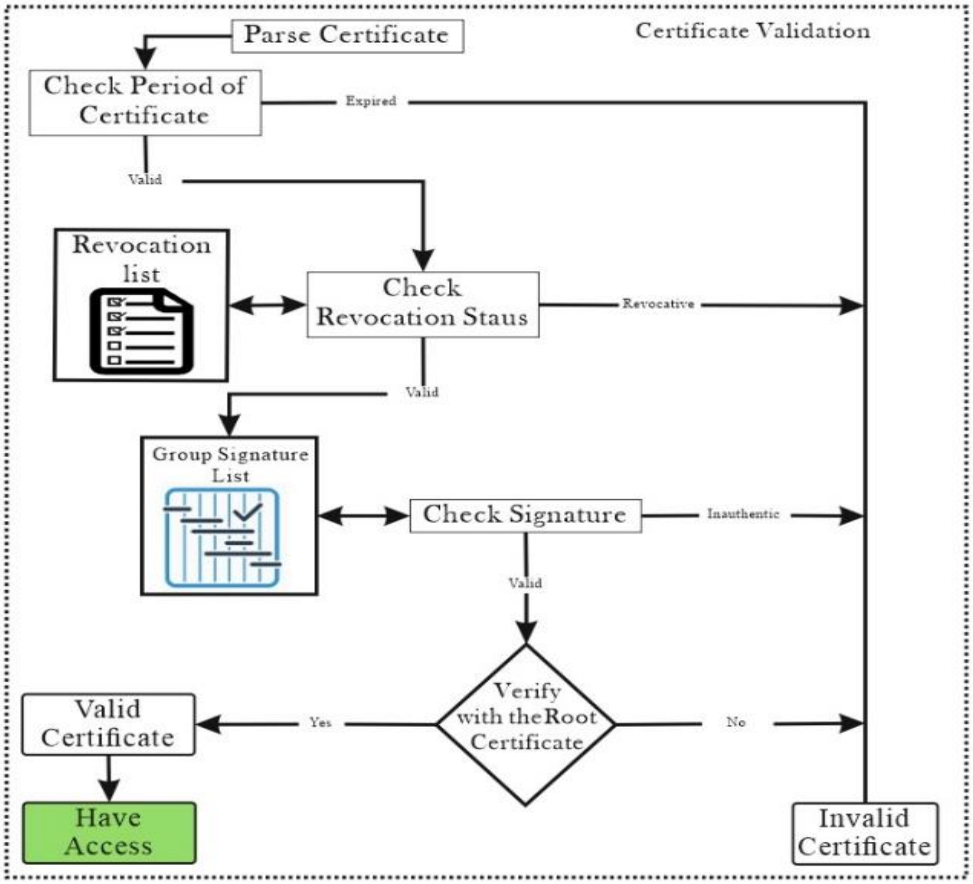

Fig. 2 Signature Verification Process

However, the dataset used for our model is named the Friendster social network and ground-truth communities [27]. Friendster is an online gaming network where users can form friendship edges with each other. The dataset consists of groups of nodes that either belongs to at least one community or are connected to other nodes that belong to at least one community. The dataset contains 65,608,366 nodes and 1,806,067,135 edges.

The proposed model is coded in the C# programming language. Moreover, the open code library "Bouncy Castle" is utilized to write the code. This library is developed by the Legion of Bouncy Castle and is a C# implementation of cryptographic algorithms. However, the algorithm of the proposed scheme is described in detail in Table 2.

TABLE II. THE ALGORITHM OF THE PROPOSED MODEL

```
Goal: Verify or generate new certificate
Input: Access request
Output: The client has access

Step1   The client request access
Step2   Check the client certificate
            o Parse the client certificate
                ▪ If the client has certificate go to the next condition
                ▪ Else the client does not have certificate and go to step3 or Cancelling request
            o Check the period of the client certificate
                ▪ If valid go to the next condition
                ▪ Else expired and go to Step3 or Cancelling request
            o Check the certificate signature
                ▪ If valid go to the next condition
                ▪ Else inauthentic and go to Step3 or Cancelling request
            o Check the revocation list
                ▪ If valid go to the next condition
                ▪ Else outmoded and go to Step3 or Cancelling request
            o Check the certificate path until the root certificate
                ▪ If it's root certificate it is valid and the client have access and Done
                ▪ Else invalid certificate and go to Step3 or Cancelling request
Step3   The client requests new certificate
Step4   The ACE receives the user requests and returns the results to the users.
Step5  Registration authority (RA) verifies user requests for a digital certificate and tells     the CA to issue it.
Step6   CA verify user id and define the client role to issues digital certificates.
Step7   Generate digital certificate that contain the client role and it is public key
Step8   Generated certificate exported to the server engine
Step9   Store the generated certificate at the certificate services database and inform the ACE
Step10  ACE returns the results to the users.
Step11   Go to step2
Step12   End
```

IV. SECURITY DISCUSSION AND ANALYSIS

The performance analysis is based on security criteria and comparison with previous schemes. The experiment shows that the availability and response time of the certificate management services is highly enhanced. In this section, we discussed the performance of using ECC with CA, the performance of verification, and the comparison with other methods.

A. Performance using ECC

The main advantages of the proposed scheme are its availability and that it's polynomial over the elliptic curve. The security of this research depends on the intractability of the elliptic curve discrete logarithm problem (ECDLP). Q can be easily calculated using the equation $Q = kp$, where $Q, p \in E$ (Fq) and $k < p$, given k and p. However, k is difficult to calculate given $Q\ and\ p$. This condition is called the discrete algorithm problem for ECDLP. This method builds a robust CA against some types of attacks.

The key sizes at equivalent security levels between the proposed scheme and the other schemes [25] [26] are compared in Table 3.

However, the powerful-per-key-bit of ECC is broadly known as better than DSA and RSA systems, namely, ECC has the ability to use smaller parameters than in DSA and RSA with corresponding levels of security.

Table 3 shows a comparison of the proposed scheme based on ECC with other schemes [25] [26] that based on RSA, our scheme has the ability to achieve equivalent security level with 192-bit ECC keys as in the other schemes with 1024-bit keys, as shown in the first row. However, the proposed scheme able to get a corresponding result with other security levels, in other words, the key size is at all times much smaller than other schemes.

TABLE III. TABLE 3. KEY LENGTH OF SIMILAR SECURITY LEVEL WITH [25] & [26]

Key sizes at equivalent security levels (bit)	**[25]**	**[26]**	**The proposed Scheme**	**The key size ratio of the proposed scheme to [25] & [26] schemes**
1	1024	1024	192	1:5
2	2048	2048	224	1:9
3	3072	3072	256	1:12
4	7680	7680	384	1:20
5	15360	15360	521	1:30

The security of ECC depends on the difficulty to determine k given k p and p. This condition is referred to as the elliptic curve logarithm problem. The RSA and ECC algorithms of comparable key sizes are compared in terms of computational effort for cryptanalysis. Significantly, a longer key size can be used for RSA than for ECC. Therefore, using ECC has a computational preference because of its shorter key length than that of a comparably secure RSA.

The comparisons in Table 3 display that a shorter key can be used in ECC than in RSA systems at a given security level. The gap between ECC and RSA systems grows rapidly as the key sizes increase in key pair generation. RSA system is more time consuming than the ECC system in obtaining high-security level and efficiency.

B. Performance of Verification

As we mentioned earlier, this system proposes a signature verification that mutually authenticates between the clients without connecting again to the centralized servers to facilitate authentication and ensure the privacy of communications between clients.

When the certificate is verified from the CA for the first time, its digital signature will be stored in the DSL at a specified group where verification occurred. Therefore, the new verification of this node will be in the same group. Consequently, the digital signature that is stored in the group will be used in the new verification process between the clients and thus save time and effort to extract this digital signature again.

Table 4 shows the estimated time of verification in both cases described above, also show different types of devices used to apply and test the system on them and the specifications of these devices.

TABLE IV. TIME CONSUMPTION OF VERIFICATION (MS)

	CPU Core i7, Frequency 2.2GHz, RAM 8GB, Harddisk 1TB		**CPU Core i5, Frequency 2.5GHz, RAM 4GB, Harddisk 512GB**	
	Verifying from Certificate (ms)	**Verifying from Signature (ms)**	**Verifying from Certificate (ms)**	**Verifying from Signature (ms)**
1	6	1	4	1
2	4	1	3	1
3	4	1	4	1
4	5	1	4	1
5	5	1	3	1

C. Comparison Between the proposed Scheme and Other Distributed Systems

In this section, we compare the proposed model with three schemes. The first method is “self-organized public key management”, which is a certificate-based authentication method proposed by Capkun et al. and is formed by certificate graphs [22]. The second method is an “enhanced DCA scheme for authentication in MANETs” [20]. The proposed scheme uses Shamir’s secret sharing scheme to support the renewal and revocation of the certificate. The third is a “fully DCA based on polynomial over elliptic curve for MANET” [23] (see Table 5).

TABLE V. COMPARISON OF CERTIFICATE-DISTRIBUTED SYSTEM

Requirements	*[22]*	*[20]*	*[23]*	*Proposed Scheme*
Resource awareness	Each node maintains two certificate repositories and thus incurs a high overhead	The generation and distribution of keys using complex polynomial functions are based on a finite field	The generation and distribution of keys using complex polynomial functions are done over the elliptic curve	The generation and distribution of keys use complex polynomial functions over the elliptic curve
Creation	Uses self-signed certificates and is thus more robust than a shared key-based mechanism	Requires at least k neighbors. The certificate comprises three basic fields: node ID, initiation time and expiry time	Requires at least k neighbors and is thus difficult	Uses self-signed certificates and is thus more robust than a shared key-based mechanism
Revocation	Explicit revocation causes a delay between faraway nodes in the network	Stores system trust counter table at each node and is thus memory-intensive	Stores system CRL table at each node and is thus memory-intensive	Stores system CRL table at each group and thus consumes less memory
Verification	A routine verification process that excludes shortening of steps or time	A routine verification process that excludes shortening of steps or time	A routine verification process that excludes shortening of steps or time	A verification process begins by taking the saved digital signature from a previous verification process and shortens the steps and reduces the time consumption

V. CONCLUSION

The increasing use of large networks, such as social networks, has made security vulnerabilities common particularly in a big data environment. New solutions are required in an era when the integration of data and scale of use rapidly increase.

In this work, we propose a DCA for big data in distributed environments by using ECC, thereby enabling the PKI-based scheme. This scheme is managed centrally and provides the capabilities for issuing, validating and revoking digital certificates.

We introduce an efficient signature verification scheme that provides mutual authentication between nodes without connecting again to the centralized servers unless the certificate has expired. Therefore, the needed effort of communication is minimized.

The main contributions of our work can be summarized as follows:

- We propose a DCA management system for big data in distributed environments.
- Using ECC with CA reduces the computations involved in cryptographic operations. Consequently, a robust and highly secure DCA over the large-scale system is produced.
- We present a verification process that can perform authentication between nodes without requiring a centralized server. This feature shortens the steps and reduces time consumption.

The proposed scheme is tested on a big dataset of social networks. The performance analysis is based on security criteria and comparison with previous schemes. The experiment shows that the availability and response time of the certificate management services is highly enhanced.

REFERENCES

[1] H. V. Jagadish, Johannes Gehrke, Alexandros Labrinidis, Yannis Papakonstantinou, Jignesh M. Patel, Raghu Ramakrishnan, and Cyrus Shahabi, “Big data and its technical challenges,” *Communications of the ACM*, vol. 57, no.7, pp. 86-94, 2014.

[2] Ganz, F., Barnaghi, P., and Carrez, F., “Information abstraction for heterogeneous real-world internet data,”. *IEEE Sensors Journal*, vol. 13, no. 10, pp. 3793-3805, 2013.

[3] Khan, Shafiullah, and Jaime Lloret Mauri, *Green Networking and communications: ICT for sustainability*, CRC press, 2013.

[4] Samdanis, Konstantinos, Manuel Paul, Thomas Kessler, Rolf Winter, P. Rost, A. Maeder, M. Meo, and C. Verikoukis, *Green Communications: Principles, Concepts and Practice,* 1st ed., Wiley, 2015.

[5] Andrew McAfee and Erik Brynjolfsson, A., Big Data : The Management Review. *Harvard Business Review*, 2012, pp. 1–12. Available: http://tarjomefa.com/wp-content/uploads/2017/04/6539-English-TarjomeFa-1.pdf

[6] Zaman, N., Seliaman, M. E., Hassan, M. F., and Márquez, F. P. G. (Eds.). *Handbook of Research on Trends and Future Directions in Big Data and Web Intelligence*. Information Science Reference, 2015.

[7] W. Wang, N. Hu and X. Liu, "BlockCAM: A Blockchain-Based Cross-Domain Authentication Model," *2018 IEEE Third International Conference on Data Science in Cyberspace (DSC)*, Guangzhou, 2018, pp. 896-901.

[8] Bhavani Thuraisingham, “Big data security and privacy”, In *Proceedings of the 5th ACM Conference on Data and Application Security and Privacy,* 2015. pp. 279-280.

[9] D. S. Terzi, R. Terzi and S. Sagiroglu, "A survey on security and privacy issues in big data," *2015 10th International Conference for Internet Technology and Secured Transactions (ICITST)*, London, 2015, pp. 202-207.

[10] Wu, Dapeng, Boran Yang, and Ruyan Wang, “Scalable privacy-preserving big data aggregation mechanism,” *Digital Communications and Networks*, vol. 2, no.3, pp. 122-129, 2016.

[11] Li, Ruidong, Hitoshi Asaeda, Jie Li, and Xiaoming Fu, “A distributed authentication and authorization scheme for in-network big data sharing,” *Digital Communications and Networks*, vol. 3, no.4, pp. 226-235, 2017.

[12] R. Li, H. Asaeda, J. Li and X. Fu, "A Verifiable and Flexible Data Sharing mechanism for Information-Centric IoT," *2017 IEEE*

International Conference on Communications (ICC), Paris, 2017, pp. 1-7.

[13] Neuman, C., Yu, T., Hartman, S., and Raeburn, K. "*The Kerberos network authentication service (V5)"*. RFC 4120, July 2005.

[14] Housley, R., Ford, W., Polk, W., and Solo, D., "Internet X. 509 public key infrastructure certificate and CRL profile," RFC 2459, pp. 1-3, January 1999.

[15] Cloud Security Alliance, "Expanded Top Ten Big Data Security and Privacy Challenges," *Cloud Security Alliance*,), pp. 1–39, April 2013.

[16] Zhao, J., Wang, L., Tao, J., Chen, J., Sun, W., Ranjan, R., and Georgakopoulos, D., "A security framework in G-Hadoop for big data computing across distributed Cloud data centres," *Journal of Computer and System Sciences*, vol. 80, no.5, pp. 994-1007, 2014.

[17] Y.-S. Jeong and Y.-T. Kim, "A token-based authentication security scheme for Hadoop distributed file system using elliptic curve cryptography.," *J. Comput. Virol. Hacking Tech.*, vol. 11, no. 3, pp. 137–142, 2015.

[18] N. Abdullah, A. Håkansson, and E. Moradian, 'Blockchain based approach to enhance big data authentication in distributed environment', in *2017 Ninth International Conference on Ubiquitous and Future Networks (ICUFN)*, 2017, pp. 887–892.

[19] Y. Dong, A.-F. Sui, S.-M. Yiu, V. O. Li, and L. C. Hui, "Providing distributed certificate authority service in cluster-based mobile ad hoc networks," *Computer Communications,* vol. 30, pp. 2442-2452, 2007.

[20] Rajaram Ayyasamy and Palaniswami Subramani, "An Enhanced Distributed Certificate Authority Scheme for Authentication in Mobile Ad-hoc Networks", *The International Arab Journal of Information Technology*, vol. 9, no. 3, May 2012.

[21] S.A. Hosseini Seno, R. Budiarto, and T-C.Wan, "A secure mobile ad hoc network based on distributed certificate authority," *Arabian Journal for Science and Engineering*, vol. 36, no. 2, pp. 245-257, 2011.

[22] S. Capkun, L. Buttyan and J. -. Hubaux, "Self-organized public-key management for mobile ad hoc networks," in *IEEE Transactions on Mobile Computing*, vol. 2, no. 1, pp. 52-64, Jan.-March 2003.

[23] A. Alomari, "Fully Distributed Certificate Authority Based on Polynomial over Elliptic Curve for MANET," *2013 14th ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing*, Honolulu, HI, 2013, pp. 96-100.

[24] Shareef, Omar Salah F., and Ali Makki Sagheer, "Using Role-based to Implement Certificate Authority Management for Big Data," *Journal of Al-Qadisiyah for computer science and mathematics*, vol. 11, no. 4, pp. 27-36, 2019.

[25] Tripathi, Shailendra Kumar, and Bhupendra Gupta, "A New Probabilistic Digital Signature Scheme Based on Integer Factorization Problemm" in *Information Science and Applications (ICISA) 2016*. Springer, Singapore, 2016. 613-621.

[26] U. Somani, K. Lakhani and M. Mundra, "Implementing digital signature with RSA encryption algorithm to enhance the Data Security of cloud in Cloud Computing," *2010 First International Conference On Parallel, Distributed and Grid Computing (PDGC 2010)*, Solan, 2010, pp. 211-216.

[27] Yang, Jaewon, and Jure Leskovec, "Defining and evaluating network communities based on ground-truth," *Knowledge and Information Systems*, vol. 42, no.1, pp. 181-213, 2015. Available: https://snap.stanford.edu/data/com-Friendster.html