BIG DATA: CRYPTOGRAPHICALLY ENFORCED ACCESS CONTROL AND SECURE COMMUNICATION

[1]AKASH GUPTA, [2]ALOK SHUKLA, [3]S. VENKATESAN

[1,2,3]Indian Institute of Information Technology, Allahabad
Email: er.akash2011@gmail.com, alok.shukla76@gmail.com, venkat@iiita.ac.in

Abstract— The evolution of big data has come with a lot of opportunities, but it also showcases new challenges to provide security in big data applications. Implementation of proper authentication and access control mechanism in big data environment is very important because a wide range of different users access massive amount of data. In this paper, we are proposing a modification in Secure Remote Password protocol to provide secure authentication and access control in big data environment and its benefits over some traditional methods of security implementation being used in current big data environments.

Key Terms—Big Data, Secure Authentication, SRP protocol, labels, Kerberos, Access control.

I. INTRODUCTION

As we all know that information technology is used extensively in our day-to-day life. We are using a lot of devices and other systems that are producing a very large amount of data every minute. So to manage this overwhelming need of data management, a new word is in the air- Big Data. These days, big data is being used in various fields such as in Genomics, Meteorology, complex physics simulation, medical research, business informatics, finance and internet search. And this counter of services is increasing every day. Because of all these reasons, security of big data is a very important aspect. The security of current big data applications are basically based on secure authentication methods. In this paper, we are proposing the use of Secure Remote Password (SRP) protocol instead of more traditional authentication protocols such as Kerberos.

We have organized this paper as follows: we discuss about big data and the various fields where it is being used in section 2. In section 3, we discuss the security problems, caused by the current authentication systems, using Kerberos as a reference model. In section 4, we propose our solution to these problems using the SRP protocol in which we made slight modification including the Labeled verifier into the SRP Protocol to provide the Access control. Finally, we conclude this paper and propose future works in this field.

II. OVERVIEW OF BIG DATA

A. Introduction of Big Data

The big data is the next big thing of the computer science. In simple language, big data is nothing but the collection of very large and complex data-sets that become very tedious and problematic task in terms of processing using the normal RDBMS or traditional applications for data processing.

B. Big Data Characteristics

The Big Data depends on three "V" factor i.e., Velocity, Volume and variety:

- This big data can come from anywhere at any flow **(Velocity)** . The data flow rate is very high in the organizations, even sometimes exceeds the capacity of current IT systems.
- At a very large amount **(Volume)**, most of the current RDBMS systems are unable to handle this much volume of data
- The type of the data, an organization captures these days is becoming extremely diverse **(Variety)** such as audio, videos, scientific data, complex simulation data etc.

Today, Big Data is being measured from Petabytes to Exabyte. The motivation behind the collection of Big Data is that, analysing a set of large amount of data instead of small sets of data of same quantity, can give some additional information which can be used in nearly every aspects of our modern society: to predict future business trends, to accurately forecast weathers, determine quality of research, to prevent wars or limit their consequences, Share-Market prediction etc.

According to a research, performed by CISCO, by the end of the year 2015, the global internet traffic will be reached at 4.8 Zettabytes a year that is 4.8 billion Terabytes per year. This growth indicates both the challenges ahead of big data as well as a large number of new opportunities.

C. Applications of Big Data

Most of the big organizations either they are government or private ventures; they are transforming their business policies towards the results generated by big data analysis. The most notable areas where the big data can and is actually playing a major role are Genomics, Meteorology, complex physics simulation, medical research, business informatics, finance and internet search.

D. Challenges in big data authentication and access control

Most of the popular big data solutions are using authentication as a primary means for their architecture security. As we know that the popular big data solutions such as Apache Hadoop, are based on cluster computing. It is very important that only the authenticated nodes of the clusters can communicate with each other. Simply we refer these devices and applications as clients. It is very important to properly authenticate the clients before they can make any interaction with our big data application. We should also consider the problems, the big data environments will face while implementing access control mechanism because the data is being captured by a wide range of different clients such as remote sensing satellites, mobile devices, logs generated by software applications, microphones, RFID tags and wireless sensor networks. So the access rights of these clients must be defined in such a way that they can interact with only those parts of big data environments that falls under their privilege. The Figure-1 is representing a general architecture of big data environment:

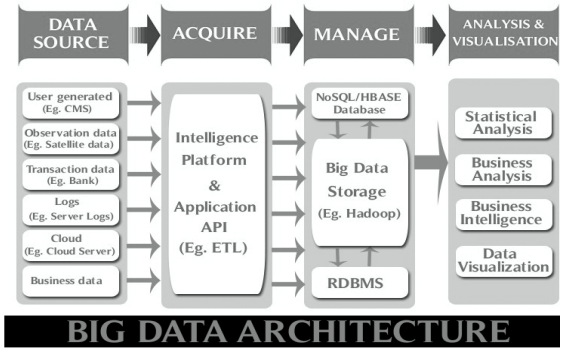


Figure-1: Big Data Architecture

E. Existing authentication techniques

The most popular big data solutions, such as Apache Hadoop are currently using symmetric key authentication system i.e. Kerberos. In this type of authentication system, the client authenticate itself using a User Id and password, known to both, the client and the big data solution. The user credentials are normally stored either in a server, in normal cases, that can be the big data solution platform itself or by a trusted third party, known as a key distribution center. When the credentials provided by the client are approved only then he is allowed to interact with big data application. For example, the most popular big data software framework, Apache Hadoop uses Kerberos protocol as the basis of its security model, for authenticating clients to Hadoop framework and to authenticate the Hadoop services to each other.

F. Security issues with traditional authentication

In traditional authentication systems, at login time the user's identity is checked using the password provided by the user. The system records the identity and determines what action or operation is to be taken. There is various security threats associated with the traditional authentication mechanism. Some of them are listed below:

1. Replay Attacks
2. Password-Guessing Attacks
3. Spoofing Logins
4. Inter-Session Chosen Plaintext Attacks (Kerberos Specific Attack)
5. Session Key's Exposure

These attacks are the most common and lethal ones, applicable on any of the traditional authentication systems. Even one of the most secure authentication security suit, Kerberos**,** is also susceptible for these types of attacks. So it is very important to implement such system which can easily deter these attacks.

Implementation of access control mechanism in big data environment is always a big challenge for the computer scientists. In a simple big data environment, the number of clients interacting with the big data application may range between a few hundred to multi-million users. Similar to any other environment, it is very critical in a big data environment to ensure that only the authorized users can access the information and the unauthorized users can't.

There are three basic problems while implementing access controls in big data environment:

1. Determining the security needs for individual users.
2. Monitoring the user's roles and authorities.
3. Proper implementation of secrecy requirements in big data environments.

To address these security problems in big data environments we are proposing a modification in Secure Remote Password Protocol to accommodate the access control of the clients in authentication level. For this purpose, we will assign the access labels to the big data users to define their access rights in big data environment.

III. SECURE AUTHENTICATION AND LABELED ACCESS CONTROL USING SRP PROTOCOL

Simple Remote Password Protocol is a secure password based authentication and key management protocol. This protocol authenticates the clients to the server using a password-like secret. This secret must be known to the client only. No other secret information is needed to remember by the client. The server stores the verifiers for every user to authenticate the client but if this verifier is compromised to an attacker, it cannot be used to by the attacker to impersonate as a client. The major advantage of SRP protocol over other authentication mechanisms is that there is no need to store any

password equivalent data and the systems are immune to the password attacks. When the client is verified by the server, a cryptographically strong secret is exchanged by the SRP protocol between the communication parties to communicate securely.

A. Advantage of SRP Protocols

The main advantage of SRP protocols over other authentication mechanisms are:

- An attacker cannot perform snooping attacks because there is no need to send password in any form, over the network.
- Replay attacks are not possible in SRP protocol because an attacker cannot reuse any of the information, exchanged between both parties during the authentication process, to get the server access.
- There is no need of any trusted-third party servers in SRP protocols.
- This protocol is used to provide mutual authentication to both parties.
- Neither the client nor the server store password in an form so the password attacks such as dictionary or brute-force attacks are not useful.

B. Our Contribution

SRP protocol is sufficiently secure when it comes to implement authentication mechanism in big data environments. With a slight modification in this protocol, we can achieve a high degree of access control in our big data environment. For this purpose, we use an access label which is associated with each of the big data client. In SRP protocol, the server stores a verifier value for each of the user instead of the user's password to verify their authenticity. In our access control model, we store these verifiers in different tables according to their specified access labels. These access labels are actually fixed numerical values that are used to distinguish the users from each other as their privilege to access the big data environment. For example, users associated with label A may have the read access only privilege while the users associated with label B may have both read/write access in big data environment. When a client requests to authenticate itself in a big data environment using the SRP protocol, the server verifies its credentials and the associated label. If the user is verified, then access to the data resources is provided to it according to the access rules defined in the associated label.

This method is very simple to implement with existing big data environments. The SRP authentication server has to maintain different tables to store user credentials according to their respective access labels only.

C. The modified implementation of the protocol

In this section, we describe that how we can slightly modify the SRP protocol so that we can explicitly implement access controls on different types of big data clients in addition to the secure authentication. The mathematical notations used for the protocol implementation are given below:

Notation	Description
s	Salt value
P	Client's password
v	Password verifier of the host
H()	Hash function, probably SHA-256
K	Session keys generated after the authentication
L1, L2..., Ln	Access Labels, fixed numeric values are assigned to these labels. These numeric values are chosen by the SRP protocol server.
LV	Labeled verifier.

Table-1: Notations

This whole process is completed in two steps. In first step, the client, who needs to access the big data environment, register itself to the SRP protocol server. The beauty of SRP protocol is that it does not store the password or any password equivalent data in any form at server. Instead it stores a password verifier which is generated by the client, if compromised; it does not reveal the original password.

For authenticating itself to the SRP server, a client chooses a random salt value s and then it computes a hash x using the password P and the salt value s:

$$x = H(P, s)$$

Now it computes the password verifier v:

$$V = g^x$$

After computing the verifier, the client sends it with the salt value s to the server.

Till now everything is just like the traditional SRP protocol. In our proposed model, we are using different tables to store these user's credentials according to their privilege rights. Each of these tables has a fixed numerical value associated with table. This numerical value is used to compute LV, labeled password verifier:

$$LV = H(L, v)$$

This LV is stored in the labeled table instead of v with Username and Salt.

The reason, we are doing this because it is much easier to define the access roles on the tables storing the user credentials and creating a well-defined session management rather than explicitly defining the access rules for individual users. Some of the access rules examples can be read, write, and read/write access to the assets of big data environment. We can also define the types of big data resource that can be accessed by the users defined in these tables.

In second step, the authentication process takes place. The complete steps of this process are given as follows:

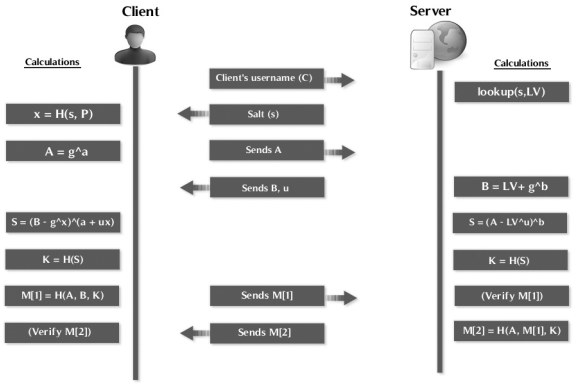


Figure-2: Modified SRP Protocol Implementation

1. The client sends his username to the SRP authentication server, hosted on the big data environment.

2. Now the server searches the client's labeled verifier LV and the salt value s. Now this salt value is send back to the client where the client computes its private-key **x** using its original password and the salt **s.**

3. A number **a,** randomly generated by the client, is such that, **1 < a < n**, and then this number is used to compute the client's public key $\mathbf{A = g^a}$. This public key is then transmitted to the server.

4. On the server side, a random number **b** is generated such that, **1 < b < n**. Then the server computes its own public key using this random number $\mathbf{B = LV + g^b}$**.** The **B** and a random parameter u are then sent to the server.

5. Both the server and the client calculate a common exponential value **S** with the help of commonly available values. If the client's password **P** matches with the value which was previously used to generate the password verifier **v,** then the values of the both **S** will also match.

6. Now this **S** is hashed by both client and server to generate a strong session key.

7. The client sends a message **M[1]** to the server as an evidence of the possession of the correct session key. On the server side, the server itself computes the value of **M[1]** to verify that the client sends him the right message .

8. The server also sends a message **M[2]** to the client as an evidence of the possession of the correct session key. On the client side, the client itself computes the value of M**[2]** to verify that the server sends him the right message .

Now when the both parties are verified to each other, the client can start its interaction with the big data environment. Now because the authenticated user is labeled with the access permissions on the big data environment, it will only interact with those parts of the environment that fell under its privilege area.

Label 1 (R)			Label 1 (RW)			.....	Label n (R...)		
Username	Salt	LV	Username	Salt	LV		Username	Salt	LV
.....	.....	.....	.....	.....	.....		.....	.....	.....
.....	.....	.....	.....	.....	.....		.....	.....	.....
.....	.....	.....	.....	.....	.....		.....	.....	.....
.....	.....	.....	.....	.....	.....		.....	.....	.....

Table-2: Sample LV Table

This simple modification in SRP protocol will help us to easily define the access roles of each big data user according to the labels that are assigned to them. Because the labels assigned to the tables, which store the LV, are unique then the session key will only generated if the user belongs to that particular table only. This will help to implement restricted access policy in big data environment and the clients can only interact with those parts of data resources that fall under their privilege level.

IV. COMPARISON WITH KERBEROS AUTHENTICATION PROTOCOL FOR BIG DATA

	Kerberos	SRP
Single point failure	Possible	Not Possible
	If Kerberos server fails then no user can login	KDC is not required
Time Stamp Sync	Required	Not required
	The clock of the host requesting for login should be synchronized	The clock is not required to be synchronized as the protocol is independent of time.
Access Control	Not implemented	Not Implemented
	Provide authentication only.	Original version provides authentication only. Access labels are included in modified SRP protocol to implement Cryptographically Enforced Access Control.
Authentication Delay	Possible	Not Possible
	As no. of user increases or delay due to KDC server overload leads to delay in authentication	Just like the modified SRP, delay would be negligible or impossible.
Implementation	Complex	Easy
	Implementation of Kerberos is complex in nature because of the management of the KDC and the tight synchronization of time between hosts and the Kerberos server.	This protocol is simple. No special requirement such as KDC or time synchronization is needed.

Table-3: Comparison of Modified SRP with Kerberos

CONCLUSION AND FUTURE WORK

In this paper, we discussed the authentication and access control issues related to big data application's security. We discussed the current authentication mechanisms and their demerits. Then we proposed our solution to modify the SRP protocol for the client authentication and implementing access controls in the big data application. Now we are trying to implement this algorithm with mandatory and role based access control systems. Also, we will study to implement it with the attribute based encryption system to achieve security in data nodes of the Apache Hadoop.

REFERENCES

[1] Thomas Wu (Sat Nov 22, 1997). Competitive Analysis of SRP (Online). Available: http://srp. stanford. edu/analysis. html (Accessed on 27/02/2014).

[2] Alvero A. Cardenas, Pratyusa K. Manadhata, Sreeranga P. Rajan et al., "Big Data Analytics for Security" IEEE Security&Privacy, Nov.-Dec. 2013 (vol. 11 no. 6), pp. 74-76.

[3] Wenrong Zeng, Yuhao Yang, Bo Luo et al., "Access Control for Big Data using Data Content" Big Data, 2013 IEEE International Conference, 6-9 Oct. 2013, pp. 45-47.

[4] Judith S. Hurwitz, Alan F. Nugent, Fern Halper, Marciaa Kaufman et al., "Security and Governance in Big Data Environments" in Big Data for Dummies, John Wiley & Sons, Hoboken, New Jersey: Wiley, 2013, pp. 226-234.

[5] Wikipedia (February 2014). Kerberos Protocol (Online). Available:http://en/wikipedia.org/wiki/kerberos_protocol (Accessed on 16/02/2014).

[6] IBM Corporation (October 2012). Top Tips for Securing Big Data Environments [Online]. Available: http://public. dhe.ibm. com/common.ssi /ecm/en/imb14137 usen/IMB14137USEN.PDF.

[7] Thomas Wu (Sat Nov 22, 1997). The Secure Remote Password Protocol [Online]. Available: http://srp. stanford.edu/ndss.html.

[8] Steven M. Bellovin and Michael Merritt, AT&T Bell Laboratories (January, 1991). Limitations of the Kerberos Authentication System [Online]. Available: http:/ /www.eecs.berkeley.edu/~fox/summaries/glomop/kerb_limi t.html.

[9] Wikipedia (February 2014). Big Data [Online]. Available: http:// en.wikipedia.org/wiki/Big_data (Accessed on 11/02/2014).

★★★