HASH-BASED RANDOM SALT PASSWORD AUTHENTICATION IN TWO SERVERS

M. O. Ampomah[1], J. B. Hayfron-Acquah[1], F. Twum[1], J. K. Panford[1]

[1]*Department of Computer Science, Kwame Nkrumah University of Science and Technology (KNUST), Kumasi, Ghana*

Abstract - ***Internet has become the most integral part of our everyday life and those who use it for their work such as bank transaction and online shopping are also growing continuously. Services provided by these websites must also be authenticated. Such systems should allow users to create their own usernames and passwords with a reliable service so that only authorized users can access their account by password authentication.***

Usually, users keep the same usernames and passwords for the websites they use. This leads to the hacking of passwords by hawkers and can easily gain access to the contents of user information. Since most of the websites use low security suite, it stores the usernames and passwords in a clear text and this allows the hackers to hack them. These authentication problems can be resolved by using hardware tokens or client certificates. But these two methods are very costly and also are not user friendly. Many cryptographic hash functions are used for the development of secure password system.

The server system is particularly suitable for resource-constrained users due to its efficiency in terms of both computation and communication. The proposed work presented a user friendly secured password authentication system with two servers by applying Pseudo Random key model.

Key Words: *Hacking, Hash Function, Password Hash, Two Server Password, Authentication*

1. INTRODUCTION

In an internet based surroundings, two server authentication techniques are measured to be safe and sound for user validation. As the number of security measures online increases every day, users will have confidence in using variety of internet services. Since every service requires the user to record independently the transparency of identifying many user ID and password pairs will lead to the problem of having to memorize them. To deal with this, researchers have planned mechanisms for multi-server environment whereas the user requests to record with a single registration center by using one ID and password pair. By this means, they have the right to use all the services registered throughout the server. Since these mechanisms utilize textual passwords, they undergo many key drawbacks. In this study, a two-server password authenticated key mechanism has been proposed which improves communication efficiency compared to multi-server password authentication.

A two server password authentication is designed in a way that does not require a password table at the server side for verification. In addition, it also provides safe low calculation for mutual authentication and session key agreement. The proposed two server password authentication is computationally proficient and is expected to be secure against ID theft, inside attack, replay attack, shoulder surfing attack, server spooling attack, and guessing attack.

1.1. PROBLEM STATEMENT

Generally, user authentication systems that rely on passwords only tend to trust the dedicated verification server in a manner that weak/plaintext passwords verification data are stored. A system like that is susceptible to attacks and if an intruder succeeds in the attack then it will be easy to retrieve the password by the intruder and hence obtain the user's information. This issue can be resolved by using multi-server systems where a user can communicate with all the servers at the same time for authentication purpose [1]. This approach requires huge throughput channel and also needs end-user synchronization. Deploying and maintaining such systems is very difficult as costly protocols are needed. To resolve these problems the two server authentication approach that uses password and session keys only together with the two servers is proposed.

2. RELATED WORK

Internet is the biggest era to the whole world. To access the services offered through internet, several web services have been used .The users who are accessing the web services must create username and password for the website they had chosen. It is necessary to keep the password safe and secure. There may be a chance to hack the password by the outside onlookers to access the data provided by the user. So, it is necessary to follow techniques to preserve the password from onlookers to hack it.

To protect the server from offline dictionary attack, single server setting was used by Gong [2] in a hybrid, PKI-based model in which the users can identify the servers' public key along with their password. Bellovin and Merritt [3] were the very first to propose protocols designed for password-only authenticated key exchange, where the clients are required to store only a short password and no additional information. The initial works by Merritt and Bellovin [4], Jablon [5] and Lucks [6] were inefficient and it provides no security for the user. Recently, Bellovin and Merritt [3], Halevi and Krawczyk [7] followed a model for the password-only setting. Goldreich and Lindell [8] discussed associated protocols with proofs of security in the random oracle/ideal cipher models. Gennaro and Lindell [9], Jiang and Gong [10] assumed some public information which is available to all parties. Since this public information is coded into implementation of protocols, it is not necessary for the user to memorize high-entropy, cryptographic information as they are required to do in the PKI-based setting. A review will be conducted on the above mentioned techniques to find the merits and drawbacks of these system and how best to improve on the proposed two server password authentication system.

The approach by Ford and Kaliski [11] is believed to be the first multi-server password system which splits a password among multiple servers, and however, the servers need to use public keys. An improved version of the approach by Bellare and Rogaway [12] was proposed in Jablo [13] eradicated the use of public keys by the servers. Further and more rigorous extensions as presented by Mackenzie et al [14], where the former built an out of range threshold PAKE protocol and provided a formal security proof under the random oracle model proposed by Yang et al [15] and the latter presented two provably secure threshold PAKE protocols under the standard model. These protocols have very low efficiency and high operational overhead. In these multi server password systems, the servers which should be equally divided for the user or the users' will communicate with all those servers simultaneously or a gateway is introduced between the users and the servers for password authentication system.

3. METHODOLOGY

3.1 DATA COLLECTION

Primary and secondary data were collected using unstructured interviews with web application users and technical experts in web base applications such as system administrators, web security managers, and embedded system programmers to determine, analyze and improve on the ideas on the limitations of the model and the best hash protocol to suit the two server authentication model in web base application.

3.2 SYSTEM DESIGN

The generalized two-server split password construction and the applications justify the security model assumed. The CS is restricted by a passive adversary while the SS is controlled by an active adversary. The anticipated two servers split password authentication presented in the two server password system exploit the generalized architecture as depicted in Figure 1. The workload of the two servers in terms of computation and communication upon the user password authentication is less.
The user communicates only with the SS but both the CS and SS authenticate the user. The user has a password which is divided into two portions, one portion is stored by the SS and the other portion is stored by the CS. During the login process, the servers use their individual data portion for user authentication. To verify the password of the user in order to establish a connection with the SS, the servers' generated function is used to generate the user portion of the password.

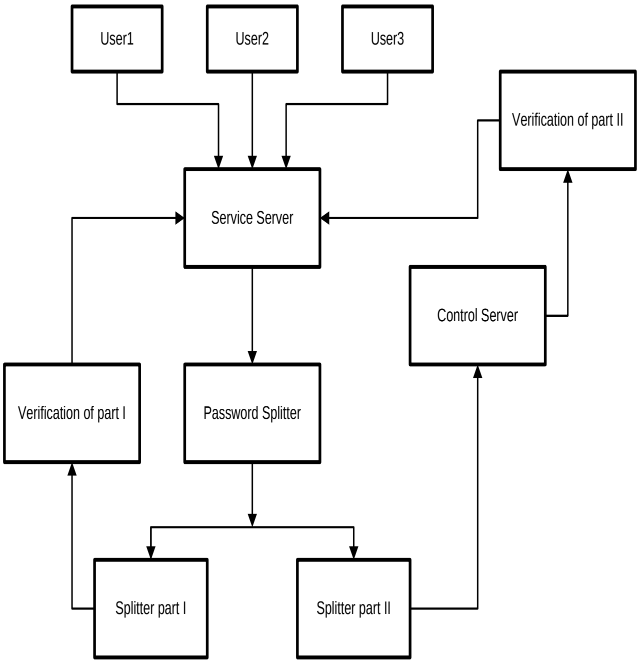


Fig 1: Generalized Two Server architecture of a single control server with service server

3.2.1 Algorithm for Hash Based Two Server Password Authentication

The algorithm for hash based Two Server Password Authentication Process is described in the following steps:

Input:

Pseudo Random Hash Password provided by user.

Output:

- Split password and authentic code.
- Authentication status to the user by Control Server and Service Server:

STEP 1:

- Input the Pseudo Random Hash Password.
- Initialize password to corresponding user.
- Listing all the user passwords

STEP 2:

- Verify the user registration.
- Service server authenticates user registration.

STEP 3:

- Authentication of the user at control server.
- Split the user hash password.
- Assign split passwords to the control and service servers.

STEP 4:

- Verification of user password.
- Initialize valid key between the user and service server.
- Split password exchange between service and control servers.
- Validate service server and control server exchange.
- Combine the split password.
- Matching of combined password to the user entered password.

STEP 5:

- Maintain status of the logged users and denied log users.
- Split Hash value exchange computation increase user security.

4 IMPLEMENTATION

The performance of hash based pseudo random password authentication in two server scheme presented in this work is measured in terms of its efficiency and effectiveness. The efficiency is evaluated between the two servers based on concurrent verification of two portions of the password by authenticated communication. The adversary resistance rate is improved as compared to that of single server models. The communication round on password authentication and verification is minimized. In addition, the effectiveness of the two server password authentication system can be measured in terms of web browser compatibility, storage capacity, computational and communication effectiveness of multiple web sites.

The proposed method proves to be effective irrespective of the speed of the client side computers as the work concentrate more on the server side and not on the client side. The password received from the user is sent to the SS known as the public server and the server is only responsible to communicate with the CS also known as back-end server. This is evident from the discussions made in the methodology, so whatever the client side computers speed may be, the proposed method is still effective for that scenario.

4.1 Simulation Tool

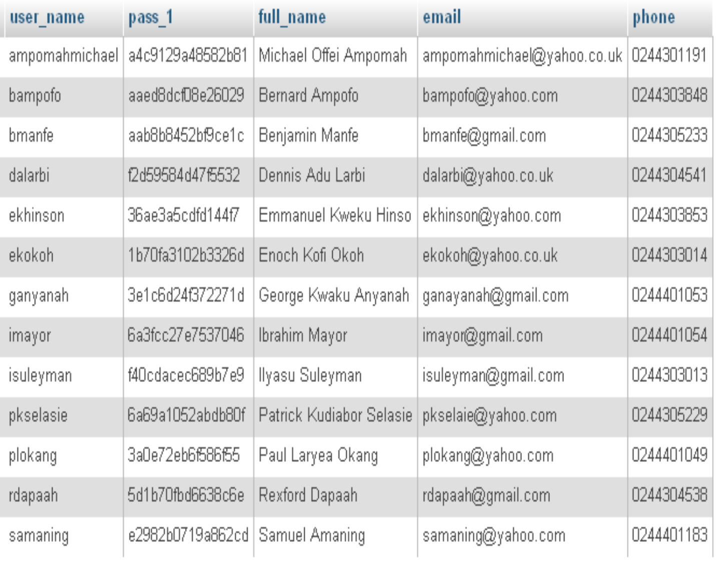
In this work the application development and simulation model were implemented using PHP 5.2.9 and MySQL 5.5.28 versions. A simple web application with authentication system and database management system has been developed. Xampp 1.8.1 was used for compiling the PHP code and developing the front-end GUI window.

4.1.1 Registration Phase

In the registration phase, the user has to input the name, email address, phone number, username, password and confirmed password. The password should be at least six characters and must contain alphabets, numbers and alphanumeric. The system automatically generate a random salt for the password entered by the user. The system map the salt to the password and hash it with MD5 standard cryptographic hash function of 64bits.

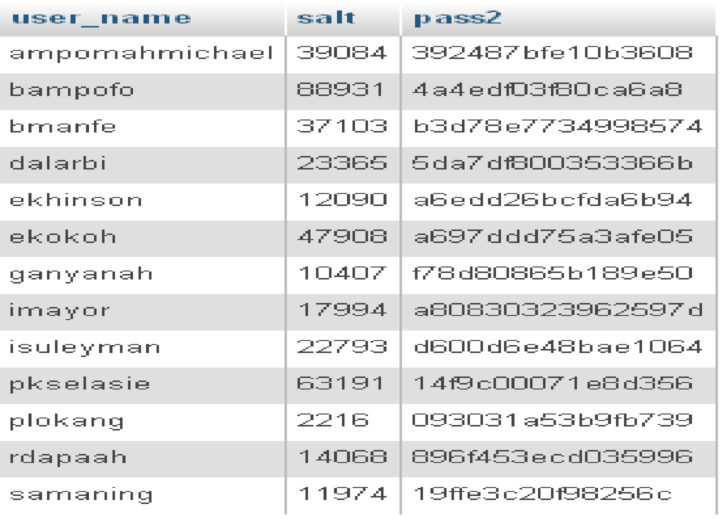
4.1.2 Database

The password is then split into two shares as Pwd1 and P2 by the system. Pwd1 together with the name, email, phone number and the username is saved in the user database record of the service server as indicated by figure 2. Pwd2 together with the username and the salt is saved in the user database record of the CS as indicated by figure 3.



user_name	pass_1	full_name	email	phone
ampomahmichael	a4c9129a48582b81	Michael Offei Ampomah	ampomahmichael@yahoo.co.uk	0244301191
bampofo	aaed8dcf08e26029	Bernard Ampofo	bampofo@yahoo.com	0244303848
bmanfe	aab8b8452bf9ce1c	Benjamin Manfe	bmanfe@gmail.com	0244305233
dalarbi	f2d59584d47f5532	Dennis Adu Larbi	dalarbi@yahoo.co.uk	0244304541
ekhinson	36ae3a5cdfd144f7	Emmanuel Kweku Hinso	ekhinson@yahoo.com	0244303853
ekokoh	1b70fa3102b3326d	Enoch Kofi Okoh	ekokoh@yahoo.co.uk	0244303014
ganyanah	3e1c6d24f372271d	George Kwaku Anyanah	ganayanah@gmail.com	0244401053
imayor	6a3fcc27e7537046	Ibrahim Mayor	imayor@gmail.com	0244401054
isuleyman	f40cdacec689b7e9	Ilyasu Suleyman	isuleyman@gmail.com	0244303013
pkselasie	6a69a1052abdb80f	Patrick Kudiabor Selasie	pkselaie@yahoo.com	0244305229
plokang	3a0e72eb6f586f55	Paul Laryea Okang	plokang@yahoo.com	0244401049
rdapaah	5d1b70fbd6638c6e	Rexford Dapaah	rdapaah@gmail.com	0244304538
samaning	e2982b0719a862cd	Samuel Amaning	samaning@yahoo.com	0244401183

Fig 2: Database Record for the Service Server



user_name	salt	pass2
ampomahmichael	39084	392487bfe10b3608
bampofo	88931	4a4edf03f80ca6a8
bmanfe	37103	b3d78e7734998574
dalarbi	23365	5da7df800353366b
ekhinson	12090	a6edd26bcfda6b94
ekokoh	47908	a697ddd75a3afe05
ganyanah	10407	f78d80865b189e50
imayor	17994	a80830323962597d
isuleyman	22793	d600d6e48bae1064
pkselasie	63191	14f9c00071e8d356
plokang	2216	093031a53b9fb739
rdapaah	14068	896f453ecd035996
samaning	11974	19ffe3c20f98256c

Fig 3: Database Record for the Control Server

4.1.3 Login Phase

The user, U, submits his identity, that is, username and password to the system in order to login on to the service provider server.

4.1.4 Authentication Phase

In this phase the entered password is divided into two shares in accordance with the random salt generated by the PRNG in the system and stored in the CS during the registration phase. The Pwd is divided into Pwd1 and Pwd2. Here the share of Pwd1 is authenticated by the SS. If Pwd1 matches with the stored value in the SS, Pwd1 is validated and pass on the session to the CS to also verify the share of Pwd2. If Pwd2 matches with the stored value in CS, the password is authenticated otherwise, it is rejected. Figure 2 and 3 show how each share of the splitted password are stored in the respective database records of both the SS and CS.

4.2 Security Analysis

To recall, the primary objective of this work is to eliminate single point of vulnerability, resist off-line dictionary attacks and on-line attacks such as man-in-the-middle attack and user impersonation attack, where CS is controlled by a passive adversary and SS is controlled by an active adversary. Accordingly, an examination of this protocol is performed against CS, SS and adversary on the servers.

The security analysis is discussed with respect to the security features which the proposed protocol should satisfy. It is desirable for a Hash-Based Random Salt Password Authentication to possess the following claims of the security attributes:

1. Resilience to single point of

2. Resilience to off-line dictionary attack

3. Resilience to on-line attack such as replay attack and guessing attack

4.2.1 Claim 1: Resilience to Single Point of Vulnerability

PROOF: A single point of vulnerability as in the single-server model is totally eliminated. In principle, without compromising both servers, no attacker can find user password through on-line attacks. Then again, as the control server is isolated from the public, the chance for it being attacked is greatly minimized, thus increasing server side security and in turn security of the overall system.

The two-server model depicted by Figure 1 is comprise of two servers at the server side (i.e.) the service (public) server which communicates directly with the users, and the control (back-end) server which stays behind the scene. Users only contact the public server, but the two servers cooperate to authenticate users. It is important to note the essential differences between the two-server model and the earlier multi-server model: in the two-server model, a user can establish a session key only with the public server, and the back-end server only plays a role in assisting the public server in user authentication; whereas in the multi-server models, a session key is established by a user either differently or same with each of the servers.

4.2.2 Claim 2: Resilience to Off-Line Dictionary Attack

PROOF: In off-line dictionary attack, the attacker can extract the secret data from the legitimate user and attempts to guess the user password in the registration phase. But, the attacker cannot guess the user password using the secret data extracted from the legitimate user information, because the attacker does not know the secret share of the CS. Therefore, the proposed scheme is secure for the off-line password guessing attack.

4.2.3 Claim 3: Resilience to Online Attack

The on-line attack comes in different forms such as replay attack and guessing attack. The proposed protocol deals with these attacks as follows:

4.2.3.1 Replay Attack

PROOF: Replay attack is also known as 'man-in-the-middle' attack. The attacker stays in between the user and the server and hacks the user credentials when the user communicates with the server. To overcome this, the user has to frequently change the credential randomly. But it is less probable to do that. The protocol implemented is strong and robust when the replay attack happens in between the two servers as the credentials are interpreted and divided into two parts. To perform replay attack, an attacker attempts to make the forged password with the legitimate password intercepted in the login phase. The forged password together with the legitimate password is intercepted in the authentication phase. But, the attacker cannot make the forged password, because the attacker is not able to compute these forged password without knowing the back-end server secret share of the legitimate user password Pwd2 and salt S, even if the attacker can obtain the secret share of the other half of the password stored in the public server. Hence, the attacker cannot perform man-in-the-middle attack while communicating between the server and the user.

4.2.3.2 Guessing Attack

PROOF: Guessing attack is when the attacker tries to communicate with the servers by randomly guessed credentials. The effective possibility to overcome this attack is to choose the password by maximum possible characters, so that the probability of guessing the correct password can be reduced. As the protocol implemented uses random generation of salt values not less than five characters together with a standard password set by the protocol which must be at least eight characters and must include alphanumeric, alphabets and numbers, it makes it more difficult for the attacker to guess the legitimate password.

5 CONCLUSION

The hash password system offers a secure hash function for password authentication. It is simple to use since the hash function is combined with pseudo random number to gives a challenge to the intruder to steal a password. It provides secure password authentication with a reliable service.

The hash password system splits the password into two components and these password components are stored in two different servers which include the service and control servers. Both the control server and service server are controlled by a passive and active adversary respectively. No attacker can find user password through offline dictionary attacks without compromising both servers. Since the control server is isolated from the public, the probability for it being attacked is significantly reduced, thereby increasing the security of the overall system. The system has no compatibility issue with the single-server model. The two server authentication exploit the advantages of the hash-based random salt model to progress the presentation of password sharing between the control server and service server. Compared with the single server model and the multi-server model, the two server model proposed, does not accept both replay and guessing attacks.

The hash is employed using a pseudo random function key. The cryptographic hash makes it complicated to work out hash from one website to another website. This technique deter password hacking and it can be used only in the domain in which they use. This builds the user confidentiality and this model can also be used under modern browsers for effective secure transaction.

6 RECOMMENDATIONS

By using hash based random salt password authentication in two servers, the web user efficiently handles the password for the various web sites in tandem. In addition the web user verifies their session logs and secure their data more effectively. However the session validation check of the user password, during new password updates produce complex adverse effect on the two server co-verification of split password portions.

To reduce the complex adverse effect, a cache update validation mechanisms between the two servers can be adopted in the future. In the hash key generation process, the pseudo random number may miss one or two hash keys for the websites, if the number of websites maintained by the user is high. The future concern is on the scalability of websites used by individual users to ensure higher password security levels.

7. REFERENCES

[1] T. Thangavel and a. A. Krishnan, “Integrated Quantum and Classical Key Scheme for Two Servers Password Authentication.,” Journal of Computer Science 6 (12), pp. pp 1396-1405, 2010.

[2] L. Gong, T. Lomas, R. Needham and a. J. Saltzer, “Protecting Poorly-Chosen Secrets from Guessing Attacks,” IEEE Journal on Selected Areas in Communications, Vol. 11, No. 5, pp. pp. 648-656, 1993.

[3] S. Bellovin and a. M. Merritt., “Encrypted Key Exchange: Password-Based Protocols Secure Against Dictionary Attacks,” Proceeding of the IEEE Computer Society Symposium on Research in Security and Privacy, Washington, pp. pp. 72-84, 1992.

[4] M. Merritt and a. S. Bellovin, “Augmented Encrypted Key Exchange: a Password- Based Protocol Secure against Dictionary Attacks and Password File Compromise,” Proceeding of ACM Conference on Computer and Communication Security.” ACM press, New York,, pp. pp. 244-250, 1993.

[5] D. Jablon, “Strong Password-Only Authenticated Key Exchange,” ACM Computer Communications Review Vol. 26, pp. pp. 5-20, 1996.

[6] S. Lucks, “Open Key Exchange: “How to Defeat Dictionary Attacks without Encrypting Public Keys,” Proceedings of the Security Protocols Workshop, Springer-Verlag, Vol. 1361, pp. pp. 79-90, 1997.

[7] S. Halevi and a. H. Krawczyk, “Public-Key Cryptography and Password Protocols,” ACM Transaction on Information and System Security Vol. 2, No. 3, pp. pp 230-268, 1999.

[8] O. Goldreich and a. Y. Lindell, “Session-Key Generation Using Human Memorable Passwords only in Crypto,” LNCS, Springer-Verlag, No. 2139, pp. pp. 408-432, 2001.

[9] R. Gennaro and a. Y. Lindell, “A Framework for Password-Based Authenticated Key Exchange,” Advances in Cryptology | Eurocrypt 2003, LNCS Springer-Verlag, Vol. 2656, pp. pp. 524-543, 2003

[10] S. Jiang and a. G. Gong, “Password Based Key Exchange with Mutual Authentication,” Proceedings of International Workshop on Selected Areas in Cryptography, pp. pp. 267-279, 2004.

[11] W. Ford and a. J. .. Kaliski, “Server-Assisted Generation of a Strong Secret from a Password,” Proceeding of IEEE Ninth International Workshop Enabling Technologies, Gaithersburg, USA, pp. pp. 176-180, 2000 .

[12] M. Bellare and a. P. Rogaway, “Random Oracles are Practical: A Paradigm for Designing Efficient Protocols,” Proceeding of ACM Computer and Communication Security, pp. pp. 62-73, 1993 .

[13] D. Jablo, “Password Authentication Using Multiple Servers,” RSA Cryptographers' Track 2001, Springer-Verlag, pp. pp. 344-360, 2001.

[14] P. T. Mackenzie, Shrimpton and a. M. Jacobson, “Threshold Password-Authenticated Key Exchange,” Proceedings on Advances in Cryptology, Springer-Verlag, pp. pp. 385-400, 2002.

[15] Y. Yang, F. Bao and a. R. Deng, “A New Architecture for Authentication and Key Exchange Using Password for Federated Enterprises,” Proceedings of International Federation for Information Processing International Information Security Conference, 2005.

M. O. Ampomah holds a first degree in Computer Science from Garden City University College, Kumasi, Ghana and MPhil in Information Technology from Kwame Nkrumah University of Science and Technology, Kumasi, Ghana. His research interest are in Computer Communications, Wireless Sensor Network, Computer Security, Computer Network, Networking, etc. He is a Field Service Engineer at MTN.

Dr J. B. Hayfron-Acquah received the BSc degree in Computer Science from the Kwame Nkrumah University of Science and Technology (KNUST), Kumasi, Ghana, his MSc Computer Science and Applications degree from Shanghai University of Science and Technology, Shanghai, China and his PhD from the Southampton University, Southampton, England. He is currently a Senior Lecturer at the Department of Computer Science, KNUST. He has over 40 publications to his credit. His research areas include Biometrics, Cloud Computing, Networking, Image Processing and Computer Security.

F.Twum received his B.Sc.(Hons) degree in Electrical and Electronic Engineering and MSc. Internet and Multimedia Engineering from London South Bank University in 2004, and 2007 respectively. He also received MSc. Degree in Information System from Roehampton University, London in 2011. He is currently pursuing his Ph.D at the Department of Computer Science, KNUST, where he also works as a Lecturer. His areas of research interest include: Computer Networks and Security, Cloud Computing, E-Commerce, Software Engineering.

J. K. Panford received his BSc degree in Computer Science from the Kwame Nkrumah University of Science and Technology (KNUST), Kumasi, Ghana, his MSc Software Technology degree from Stuttgart University, Stuttgart, Germany He is currently a Lecturer and pursing his PhD in Computer Science at the Department of Computer Science, KNUST. He has over 15 publications to his credit. His research areas include Cloud Computing, Networking, Image Processing and Computer Security, Software technology and Embedded Systems.