A Secure Service Key sharing Approach for Authorized Service Access in Wireless Network Services

D.V. Srihari Babu, Govardhan Reddy Kamatam, K. Ayyanna

Abstract— The development of Wireless Network Services (WNS) makes it simple for users to share information and computing competence with their hosts. User recognition is a fundamental access control method for client-server networking architectures. Recognition of single sign-on (SSO) agrees to rightful users to access other service providers on WNS by means of a one session key for a session. Currently, numerous user recognition approaches have been recommended for accessing WNS. Unfortunately, existing schemes cannot sustain user secrecy while the common expected attacks occur. In addition, for more period management mechanisms they utilize to be able to outcome in extensive overhead outlay. To prevail over these inadequacies, we propose a Secure Service Key (SSK) shared approach for authenticated service access in wireless network services. This approach initially implements a method for generating a secure encryption key to validate user login authentication and a mechanism for generating the SSK for service access at a later time. Generate a new security service key (N-SSK) for all service conversions to maintain reliability. Measure the execution time and overhead of various processes to assess on-the-fly and effects. The result is instant access to security service access.

Keywords: Wireless Network, Service key, Sharing, Authorized Service Access.

1. INTRODUCTION

In the real-world application, mobile users are able to utilize the particular session key to access many services, for instance, the downloading of music or videos, an e-mail receiving or replying, purchasing materials, or performing the online transactions etc., are from the various service providers in the Wireless Network Service (WNS). In a distinctive solution, users must register with every "service provider" and preserve dissimilar ID/password pairs to access every service provider. However, if users have to remain a lot of confidential information, security issues can arise and increase network overhead. A number of research studies [1], [2], [3], [4] are proposed in the past, which provides a user authentication protocol with the session's main practice in a WNS, an attacker's attack on those attacks that do not secure the secret session, an attacker Misuse of service provider by creating session key.

Many applications, platforms, and infrastructure [5], [6], [7], [8] have a horizontal authentication requirement. There are numerous security issues regarding wireless network users' identification plans and provide a lot of improvements in conducting attacks, which provide "session key," "user name", timestamp-based and various timing zones But do not provide effective results, or if there is an overcrowded network background with non-stable commitment, an additional time synchronization method is necessary. A replica set of data is at this time because all the organization's users have to be the group for the service provider's request.

The only sign in the model is the user's authentication and consent to a particular task that allocates the user to reach all the user's request resources [9], [10], [11]. There is no need to go into various passwords to access it. The conception of the same sign be able to be functional to international managerial relations, and users are liberated to find associated positions inside the "trust limit". However, for many sign-ins, every service provider's user has only one user's identification and password [12], [13]. For others, the only sign is that a user logs at all times, the user provides a set of custom interface and applications. Another preference for the writer is a single sign, whose purpose is to offer end users by means of simply individual confirmation through the same work period. Since dissimilar service contributor is completely beneath various managerial control, it is complicated to preserve a general SSO in the service. It creates extremely complicated to log in and access many services at once.

In this paper, we propose a Secure Service Key (SSK) sharing approach for authorized service access in wireless network services as a new secure service key (N-SSK) generation to prevent access to unauthorized service for mobile devices in the secure networks and distributed networks via random new generation. The aim of this proposal is to resolve the service accessing concern from numerous service providers. To access information a user has to repeatedly share their authentication details with multiple service providers or need to create many logins. This results in a cost of time and resources with additional overhead. The proposed SSK sharing approach target to solve the problem of unauthorized access to the service and repeatedly log in to minimize the cost and time.

The remaining papers are organized as follows: the discussion on background study in section 2, section 3 has presented the view of the contribution of the key to the proposed Secure Service Key (SSK) approach, section 4 presents the investigation results and Section 5 discuss the conclusion.

2. BACKGROUND

Revised Manuscript Received on March 10, 2019.

D.V. Srihari Babu, Assistant Professor, Department of ECE, GPREC, Kurnool, Andhra Pradesh, India.(E-mail: srihari2k1@gmail.com)

Govardhan Reddy Kamatam, Associate Professor, Department of CSE, GPREC, Kurnool, Andhra Pradesh, India.(E-mail: govardhan.cse@gprec.ac.in)

K. Ayyanna, Assistant Professor, Department of ECE, GPREC, Kurnool, Andhra Pradesh, India. (E-mail: ayyanna111@gmail.com)

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

STUDY

Improve the latest technology, create the most mobile services on the mobile, and easily access the Internet with high technology and versatility. However, a security issue that you must address to determine the user's validity, identify the service provider, and establish an appropriate session key to maintain the genuine user's private information. In the earlier, user recognition protocols that provide "session key sets" and "user-friendly" appear to attack without such attacks, and legitimate sessions could infringe the key to legal sessions that could fool service contributors [1], [4], [5], [14]. Later, many suggestions have been proposed to recognize the dangers of contagious attacks and to recommend enhancements to avoid these attacks [15], [16], [17], [18].

There are a number of approaches [2], [6], [19], [20], [21] that provide a mechanism for identifying dangerous public attacks and improving the way to prevent such attacks. Use of the exclusive central relation database and login procedures to solve the problem by handling authentication on other software systems. Handles access control for many interrelated but self-regulating applications. Users are able to log in to the same system by accepting the platform when they log back in without an authorization environment for all applications in the login environment. The proposal for SSO is based on other certifications carry out at dissimilar levels. To pass a signed token through the network access stage, the SSO system must be bootstrapped. SSO formed by Open Group to removes the encumber on system supervisor who requires to be restricted by the deployment system and gains access to multiple systems. Many passwords must be remembered. It offers an integrated mechanism for managing user authentication and implementing business rules to access applications and data.

The verification solution maintains a login to sustain an abundance of user IDs and passwords [9], [10], [11] but can provide a solution that can be applied to all applications at the same time. The SSO reduces a few security risks but increases additional security risks. For example, if a user is corrupted and disabled with the computer, then all authorized resources can be considered. Applying at least multiple times will only log on to one system at an instance, so merely individuality resources will get worse. By means of SSO, every one application able to utilize the central authentication service. This is an important target of the hacker to intrude the "denial of service" attacks.

The interesting "RSA-based SSO scheme" based on the "one-way hash function" for resolving time stamp weaknesses and reducing system overhead is presented by C. C. Chang et al. [9]. It is extremely effective in terms of accountability and communication costs. The parameters utilized at this time are account costs and communication costs. However, according to G. Wang et al. [10] suggest that certificates are not secure without credentials and it is not safe to implement for the "impersonation attacks". ". The main attack is "credential attack", which is a secret to interact with credentials, a corruption provider and has been delivered twice and twice after receiving the user's compensation. Users have individual access to individuals. Other attacks, such as attacking without identifying a counterfeit attack, provide an untrusted outsider as a legitimate user who can benefit freely from network services.

D. Davidson et al. [1] Protecting security rules can be discussed at the interface among app code and web substance and can be discussed without changing the OS. "Wi-Fi frames" allow you to define easy access policies to defend your apps and embedded web content. These policies control real and all communications between satisfied security principals such as apps and the web.

R. Peeters et al. [3] gather more personal data for discussion about "weak security", "user profiling", "user practices", and "mobile validation" concerns. In regardless of an interesting stage, mobile devices still do not contain sufficient features. It suggests an art of essential security step in opening up the complete probable of mobile devices to ensure user security and to ensure reliability by regarding the confidentiality of users. It unites numerous protected encryption technologies with the principles of secure HCI design to focus on achieving improved user knowledge.

S. D. Yalew et al. [4] proposed "TRUAPP" design, a software verification service that makes certain the stability and truthfulness of apps operation on mobile devices. The benefit of expanding the "ARM TrustZone hardware" of the operating system corruption ensures the expansion of security. Use techniques such as "Steady Watermarking", "Dynamic Watermarking", and "Cryptographic Heads" to authenticate the honesty of the application. This service is applied to the hardware embark that introduced the mobile device exploited to executed the service analytical evaluations.

J. Costa et al. [5] explains a lightweight "two-factor validation system" that uses mobile devices to access valid users. "Two Factor Verification (TFA)" is appropriate progressively more significant to users' security and identity. As cybercrime enhancing year by year and consumers implement the TFA mechanism to implement user risk of machinery every year, reduce the risk of misguided consumers, and ensure consumer confidence in the system.

Various existing proposals [22], [23], [27] offer common verification and key exchanging mechanisms, although they are discussed in some reliable core commitment protocols. A major error is a high cost. The RSA algorithm [24] is typically used for encryption and validation which consists of two sets of numbers, another set of public keys, and a set of private keys. It is necessary to inject or use "public and private keys", but simply the private key holder wants to recognize about them. Based on the "RSA" and "Duffy-Hallman Algorithm", mobile user identification and current user verification ensure efficient access to authentication and privacy plans [26] and loss of damage. New authentication mechanism.

3. PROPOSED SECURE SERVICE KEY SHARING APPROACH

We provide a Secure Service Key (SSK) contribution approach as shown in Fig. 1. It is commonly known as the "individual's personal individuality". It ensures that users have access to multiple resources once and provide a validation process that can be used to provide service to mobile users through accessing service providers using "Secure Service key (SSKey)" authentication.

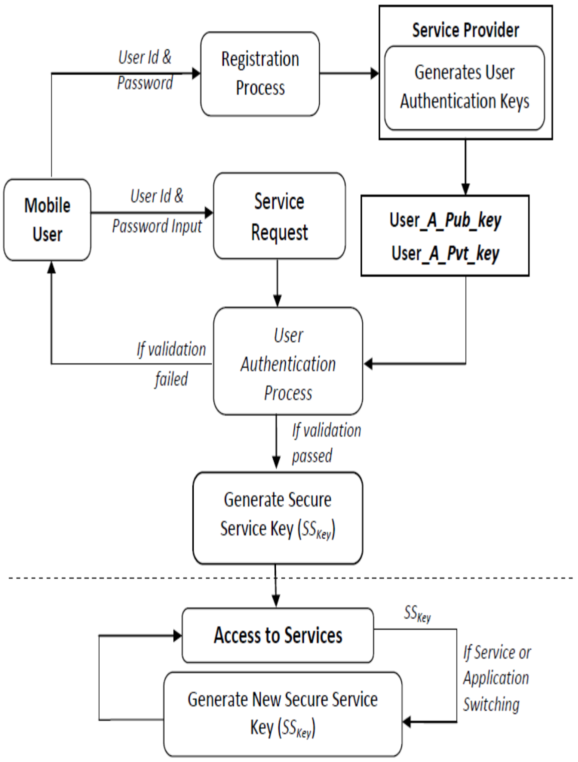


Fig.1- Proposed Secure Service Key sharing and Authentication Architecture

The first step in proving a user's authentication is validation when looking at the logging system of identity management. Entering the authentication secrets is done in comparison to search table data where personal information is stored and only users can access the service. In a "unidirectional authorization scheme", an agency reviews a variety of confidential information and recognizes a second party. In addition, the familiar identification protocol agrees among the two communication parties to verify each other. Therefore, there are major security issues that need to be conveyed to the user in the "user authorization scheme". Regardless of whether the user is rightful, the service provider has to be authenticated and properly configured to allow the generic SSKey to access a variety of services and retain the valid user's individual information.

The purpose of the method is able to be separated into three modules. The initial element provides the registration and authorization of the user, the second part provides the SSKey generation mechanism, the third component includes the SSKey 's contribution to the new service and finally accessing to the service using the new N-SSKey .

A. Registration and Login Authentication Process

One should enter their user's ID and password to specify the initial registration process for the user registration and login authentication process that the user provides. The identity and password of the entered user have been provided to the service provider to generate the service key provider as "User_A_Pub_Key" and "User_A_Pvt_Key" using a service provider public and private encryption key. The process flow of the mechanism is demonstrated in the Fig.2.

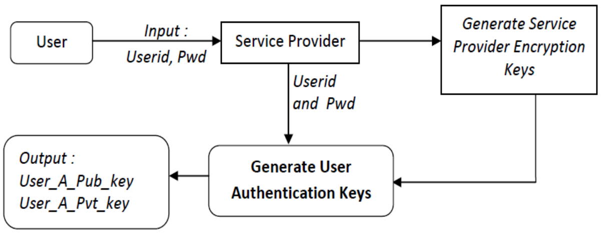


Fig.2- User secure Key Generation process

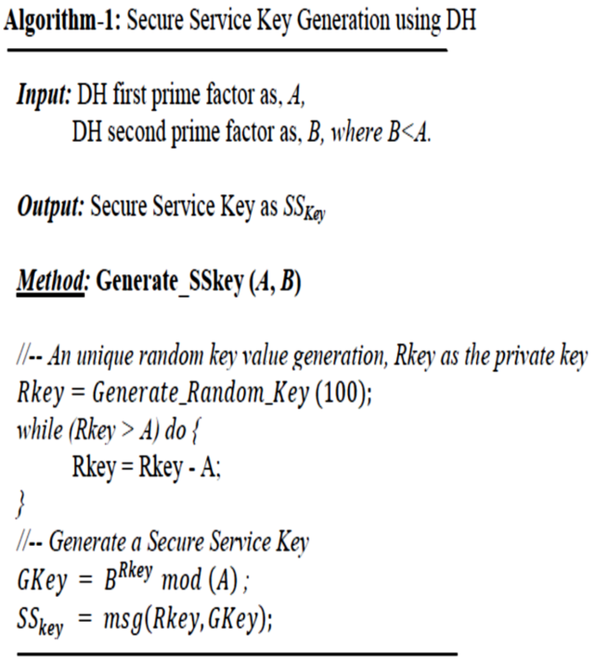
The process of registration provides the valid authentication keys to validate the user login to access the mobile services. The mechanism of authentication validates the user input key using the service provider encryption key and comparing with the key generated during registration. Controlling through the conventional mode of saving "usernames" and "passwords" in a straightforward query design that is sensitive to "Brute Force attacks". This presents a well-built relationship among users and service providers to improve the validation mechanism using an encryption key.

Service providers use a variety of methods to implement the validation process, but common industry standard IDs use server and web-supported solutions. In conventional systems, numerous systems are truly autonomous mechanisms, every user must refer to each system, and there is no unique interface each time the system is accessed. For mid- to large-sized service providers, the verification process can be called "proper failure" if it is not properly created. If the validation system is behind but the service provider still exists, users able to lose their keys because they can access any resources or applications stored on the authenticated system. It is even subject to verification procedures and cannot guarantee access to services for persistent and unauthenticated access.

To overcome the unauthorized services access we present a secure service key(SSKey) which will provide a valid service sharing key among the application an accessing session. In the case of switching between service within applications, it will provide a new service key(N-SSKey) as discussed in the next section.

B. Secure Service Key (SSKey) Utilization

Because the Internet is inaccessible, the software must validate all requests from the user's browser to establish whether the "resource" or "application" associated with the validation policy can access the user. However, each time a user clicks on a different URL, it possibly will result in further overload and crowded traffic. Devices can experience high resource losses with inadequate resources such as mobile, PDA, and so on. This is important because it is an important module that can provide the core of a secure service without having to check the verification process again. We recommend a project that uses the "Diffie Hellman (DH)" algorithm [25] as a solution by creating a Security Service Key (SSKey).The DH public key technique uses the "asymmetric key principles" to distribute "symmetric keys" to two parties in a communication network. Key involvement is a significant characteristic of existing algorithms and depends on deployment and overall security key updates. The algorithm-1 presented the SSKey generation utilizing the public key and private key of DH's asymmetric key cryptography [28]. The generated SSKey able to access services to provide through a user-based authenticated user service provider. The validation module for SSKey is implemented in an exchange with this major service provider, which is granted the user access. The next section presents the process of validation of SSKey and generating new N-SSkey while switching the application services.



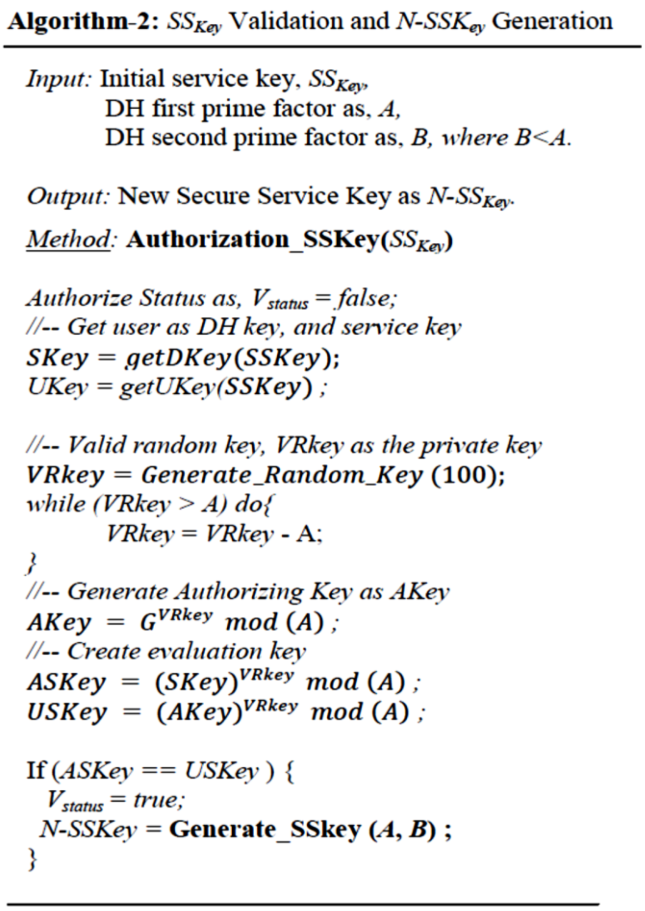
Algorithm-1: Secure Service Key Generation using DH

```
Input: DH first prime factor as, $A$,
       DH second prime factor as, $B$, where $B<A$.

Output: Secure Service Key as $SS_{Key}$

Method: Generate_SSkey (A, B)

//-- An unique random key value generation, Rkey as the private key
Rkey = Generate_Random_Key (100);
while (Rkey > A) do {
       Rkey = Rkey - A;
}
//-- Generate a Secure Service Key
$GKey = B^{Rkey} \bmod (A)$ ;
$SS_{key} = msg(Rkey, GKey)$;
```

C. Generating New SSKey on service Switching

The importance of exchanging key allows parties or services providers to share information and distribute information across multiple services. This authenticated key can work with valid code among the two parties to guarantees privacy and integrity. Generating SSKey when a user is loaded in a startup session and the user switches to that service context, a similar SSKey is the most common viewpoint. However, if a user takes the current service context to an extra service for an illustration, if a user at present browsing and download a movie from Site A and switching to Site B for melody download, this context change can potentially be reapplied. An overhead is added to the system to allow access to users it also may be delayed. By utilizing the SSKey improves the strength of service access with reduced latency to reduce collaboration and partner with service contexts and a new SSKey transformation processes outside the service context is recommended to improve the quality and secure access with low computation overhead and delay.

Because the key transformation in the cryptographic service is generated by the DH algorithm, the two service contexts have no knowledge of each other and therefore jointly identify unauthorized access to the service in collaborating on the unauthorized access of the new services. Algorithm-2 defines the authentication method of SSKey and N-SSKey generation processes.



Algorithm-2: SS_{Key} Validation and $N\text{-}SS_{Key}$ Generation

```
Input: Initial service key, $SS_{Key}$,
       DH first prime factor as, $A$,
       DH second prime factor as, $B$, where $B<A$.

Output: New Secure Service Key as $N\text{-}SS_{Key}$.

Method: Authorization_SSKey($SS_{Key}$)

Authorize Status as, $V_{status}$ = false;
//-- Get user as DH key, and service key
SKey = getDKey(SSKey);
UKey = getUKey(SSKey) ;

//-- Valid random key, VRkey as the private key
VRkey = Generate_Random_Key (100);
while (VRkey > A) do{
       VRkey = VRkey - A;
}
//-- Generate Authorizing Key as AKey
$AKey = G^{VRkey} \bmod (A)$ ;
//-- Create evaluation key
$ASKey = (SKey)^{VRkey} \bmod (A)$ ;
$USKey = (AKey)^{VRkey} \bmod (A)$ ;

If (ASKey == USKey ) {
   $V_{status}$ = true;
   N-SSKey = Generate_SSkey (A, B) ;
}
```

This randomly secured service guarantees the purpose of key generation and exchange so that the service key can be determined and set up so that unauthorized users can access the service unofficially. The authentication delivery mechanism experiment and the evaluation of the results are discussed in the next section.

4. EXPERIMENT RESULTS

This section illustrates the experimental evaluation of the procedures and techniques used in the proposed method. This includes two sections, the User section, and the Service Provider section. The user section is considered a mobile node for the user to set up, performs initial registration tasks, and then subscribes for services through the verification process. Java-based multi-nodes are evaluated for multiple nodes and the service provider section applies the listing server with the third-party "TA" through the "Apache web server". It moreover applies a separate module to estimate user authentication method all through login. The method is employed by means of the "Java Security API". The session construction module also applies when login is successful. To demonstrate the usefulness of our approach, it analyzed the performance of the underlying encryption algorithm, taking advantage of the key features of the overall architecture and architecture. For our experience requirements, it utilizes the nodes in the timer built into the module implemented using "JavaScript" [29].

4.1 Result Analysis

A. Analysis of Registration Process Time

We analyzed the computational time with varying the number of user requests from 1 to 10 for the registration process had to be completed on the server. The processing time utilized through the server was measured utilizing "Apache JMeter" [30]. It is implemented by the server along with the execution of the job code for each server application. The outcome of the result is show in Fig. 3.

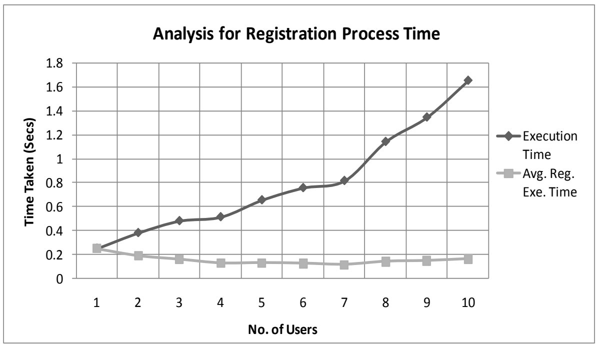


Fig.3: Registration and Avg. Execution Time Analysis

B. Analysis of Authentication Process Time

Here we analyze the encryption and decryption procedure of the key to authenticate the user login for the user verification mechanism. It repeats the process over 10 to 100 applications requests to measure the performance of this process. Fig. 4 displays the total amount of decryption time, with an increasing number of validation requests, and Fig. 5 displays full time taken performing the verification process.

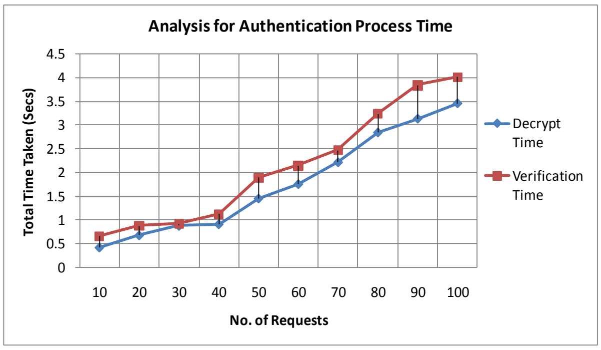


Fig.4: Decryption and Authentication Processing Analysis

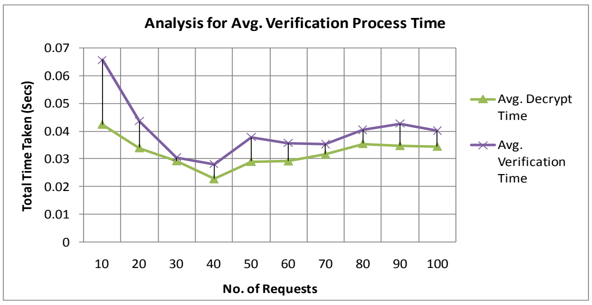


Fig.5: Analysis of Avg. Verification Process Time

C. Secure Service Key Generation Time

The total time spent on secure SSkey generation was taken for total time and overall authentication analysis. To do so, it will increase the number of user requests from 10 to 100 periodically for the evaluation. Fig. 6 shows the total time taken for complete authentication and key generation, and the average comparison of the verification with SSkey of the entire service shows in Fig.7.

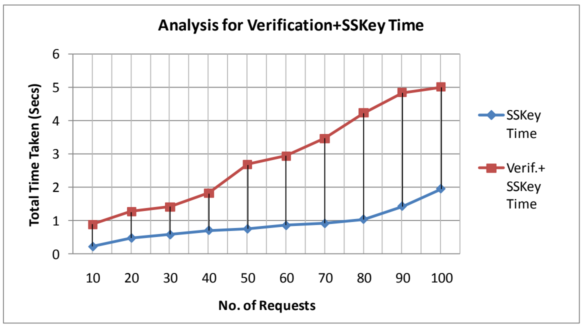


Fig.6: Analysis of verification and SSkey Processing Time

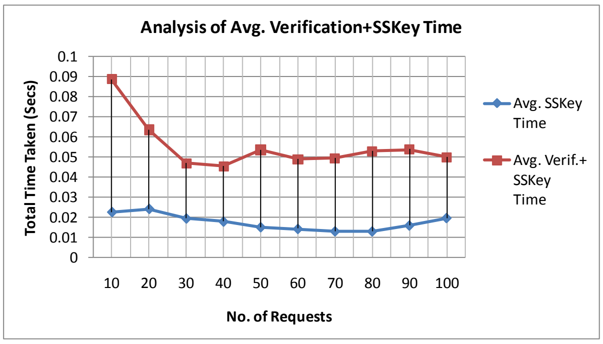


Fig.7: Analysis of Avg. verification and SSkey Processing Time

D. Service Request Performance

At this point, it measures service request performance for each request by the server related to the target node. Every second, it was dynamically passed through the node switch at the new URL to evaluate the server response time change. The resulting outcome for 10 users providing every 50 requests is presented in Fig.8.

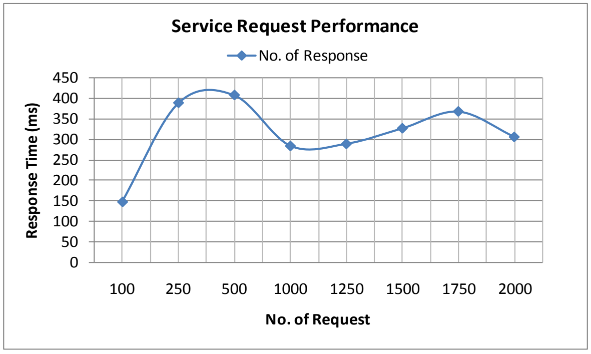


Fig.8: Analysis of Service Request Performance

In the observed result outcome from Fig. 3 to Fig. 8 verify the performance of the proposal through identifying the encryption and encryption process in the implementation and reducing the execution time for the key generation of the security service.

This demonstrates that it can help to reduce computational costs and save mobile device resources, which suggest the effectiveness of secure service key sharing.

5. CONCLUSION

In this paper, it presented the validation method of the current user for wireless network service. It provides a view of the Secure Service Key (SSKey) sharing for accessing authenticated services in wireless network services. This approach applies to user authentication, key generation of user-protected services, and service key exchange and validation methods. It will be useful to implement the SSKey mechanism to overcome possible errors in potential mobile devices. In observation of the comparing results analysis, the computational cost of the mobile users is lower with limited resources access and low communication overhead. It can also be applied to wireless networks where users can perform better on a variety of services without adding any time-consuming mechanisms in various service switching applications.

REFERENCES

1. D. Davidson, Y. Chen, F. George, L. Lu, S. Jha, "Secure Integration of Web Content and Applications on Commodity Mobile Operating Systems", In Proc. of the ACM on Asia Conf. on Computer and Comm. Security, pp. 652-665, 2017.
2. S. Ruoti, K. Seamons, End-to-End Passwords, In Proceedings of New Security Paradigm Workshop, Islamorada, Florida, USA, (NSPW'17), 14 pages, 2017.
3. R. Peeters, K. Grenman, "n-Auth: Mobile Authentication Done Right", In Proc. of ACSAC 2017, USA, pp. 4-8, Dec-15, 2017.
4. S. D. Yalew, P. Mendonça, G. McGuire, S. Haridi, M. Correia, "TruApp: A TrustZone-based Authenticity Detection Service for Mobile Apps", Proceedings of the 13th IEEE International Conf. on Wireless and Mobile Computing, Networking and Comm. (WiMob), 2017.
5. J. Costa and A. Michalas, "Middle Man: An Efficient Two-Factor Authentication Framework", In 3rd IEEE International Conf. on Computing, Comm., Control and Automation, 2017.
6. C.-H. Ling, C.-C. Lee, C.-C. Yang, and M.-S. Hwang1, "Secure and Efficient One-time Password Authentication Scheme for WSN", International Journal of Network Security, Vol.19, No.2, PP.177-181, Mar. 2017.
7. U. Shafique, A. Sher, R. Ullah, H. Khan, A. Zeb, et. al., "Modern Authentication Techniques in Smart Phones: Security and Usability Perspective", International Journal of Adv. Comp. Science and App. (IJACSA), Vol. 8, No. 1, 2017.
8. J.-Z. Lu, J. Zhou, "On the Security of an Efficient Mobile Authentication Scheme for Wireless Networks", IEEE 6th International Conference on Wireless Communications Networking and Mobile Computing (WiCOM), Pg. 1 - 3, 2010.
9. C.-C. Chang and C.-Y. Lee, • "A secure single sign-on mechanism for distributed computer networks", IEEE Transactions on Industrial Electronics, Vol. 59, Iss. 1, Pgs. 629-637, 2012.
10. G. Wang, J. Yu, Q. Xie, "Security Analysis of a Single Sign-On Mechanism for Distributed Computer Networks", IEEE Transactions on Industrial Info., vol. 9(1), pp. 294 - 302, 2013.
11. J. Yu, G. Wang, Y. Mu., • "Provably secure single sign-on scheme in distributed systems and networks. In. Proc. 11th IEEE International Conference On Trust, Security, and Privacy in Comp. and Comm. , pp 271-278, 2012.
12. J. Jacob, M. John, • "Security Enhancement Of Single Sign-On Mechanism for Distributed Computer Networks", International Journal of Modern Engineering Research (IJMER), Vol. 3, Issue. 3, pp-1811-1814, 2013.
13. J. Wang, G. Wang and W. Susilo, • "Anonymous Single Sign-on Schemes Transformed from Group Signatures", IEEE 5th International Conf. on Intelligent Networking and Collaborative Systems, Pages: 560 - 567, 2013.
14. P. Mutchler, A. Doupe, J. Mitchell, C. Kruegel, G. Vigna, "A Large-Scale Study of Mobile Web App Security", In Proceedings of the Mobile Security Technologies Workshop (MoST). 2015.
15. G. Yang, C. Tan, "Strongly secure certificateless key exchange without pairing", In 6th ACM Symposium on Information, Computer, and Communications Security, Page 71-79, 2011.
16. C. Tang and D. O. Wu, "An efficient mobile authentication scheme for wireless networks", IEEE Transactions on Wireless Communications, vol. 7, no. 4, pp. 1408-1416, 2008.
17. K. Mangipudi, R. Katti, "A Secure Identification and Key agreement protocol with user Anonymity(SIKA)", Elsevier Computers & Security, Vol. 25, Issue 6, September 2006, Pages 420-425, 2006.
18. L. Chen and C. Kudla, "Identity-based authenticated key agreement from pairings", IEEE Computer Security Foundations Workshop, pp. 219-233, 2003.
19. N. Naik, P. Jenkins, D. Newell, "Choice of suitable Identity and Access Management standards for mobile computing and communication", IEEE 24th International Conference on Tele. (ICT), Pgs. 1 - 6, 2017.
20. J. Zhang, Z. Wu, Y. Li, "An improved identity-based authenticated key agreement protocol using pairings", IEEE Proceedings of International Conference on Computer Science and Network Technology, Vol. 1, Pages: 45 - 49, 2011.
21. C. W. Lin, C. S. Tsai and M. S. Hwang, • "A New Strong-Password Authentication Scheme Using One-Way Hash Functions", Journal of Computer and Systems Sciences International, vol. 45, no. 4, pp. 623-626, 2006.
22. D. Pointcheval, M. Abdalla,D. Bernstein and T. Lange Eds, "Flexible Group Key Exchange with On-Demand Computation of Subgroup Keys", Springer-Verlag, LNCS 6055, pp.351-368, 2010.
23. R. Cramer and V. Shoup, "Design and analysis of practical public-key encryption schemes secure against adaptively chosen ciphertext attack", SIAM Journal of Computing, vol. 33, no. 1, pp. 167-226, 2003.
24. G. Sharma, S. Bala, A. K. Verma, "An improved RSA-based certificateless signature scheme for wireless sensor networks", International Journal of Network Security, vol. 18(1), pp. 82-89, 2016.
25. E. Bresson, O. Chevassut, and D. Pointcheval, "Provably-Secure Authenticated Group Diffie-Hellman Key Exchange", ACM Trans. on Info. and System Security, Vol. 10, No. 3. , Pp. 255-264, 2007.

26. M. Pirker and D. Slamanig, "A framework for privacy-preserving mobile payment on security-enhanced arm trust zone platforms", In Proceedings of the 11th IEEE International Conference on Trust, Security and Privacy in Computing and Communications, pp. 1155-1160, 2012.
27. R. Álvarez, L. Tortosa, "Analysis And Design Of A Secure Key Exchange Scheme", Elsevier Info. Sciences, Vol. 179, pp. 2014-2021, 2009.
28. D. Pointcheval, M. Abdalla, "Distributed Public-Key Cryptography from Weak Secrets", Springer-Verlag, LNCS 5443, pp.139-159, 2009.
29. JavaScript Runtime Process time, Link: "https://nodejs.org/api/process.html#process".
30. Apache jMeter, Link : "http://jmeter.apache.org".