Int. j. inf. tecnol. (March 2022) 14(2):657–665
https://doi.org/10.1007/s41870-021-00838-0

ORIGINAL RESEARCH

Euler totient function and fermat Euler theorem based an optimized key management scheme for securing mobile agents migration

Pradeep Kumar[1] · Niraj Singhal[2] · Dhiraj Pandey[1]

Received: 14 June 2021 / Accepted: 18 November 2021 / Published online: 24 January 2022

Abstract As the mobile agents move automatically on distributed networks, the security of agents and platforms is of prime concern. Over the year's mobile agents-based software applications have grown exceptionally. It also has increased the threats to the security of such applications. In the mobile agent paradigm, most of the protection schemes discuss the protection of platforms only and provides fewer directions on security of mobile agents which is still a complicated issue. In this paper, a mathematical key management technique has been presented to solve the problem of agent security and its authentication during the hoping of agents in different platforms. Here, a novel agent security approach based on the Euler totient function and Fermat Euler theorem is developed to secure a secret key among 'n' number of mobile agents. At the time of execution, mobile agent reconstructs the secret key based on the proposed technique. An evaluative judgment, comparing with various agent security schemes, has been presented, along with their complex nature. The proposed approach has been implemented and its various features are tested. The results indicate that the computing here is much more secure and easier in comparison to traditional client–server and code on-demand paradigms.

✉ Pradeep Kumar
pradeep8984@jssaten.ac.in

Niraj Singhal
drnirajsinghal@gmail.com

Dhiraj Pandey
dhirajpandey@jssaten.ac.in

[1] JSS Academy of Technical Education, Noida, Uttar Pradesh, India

[2] Shobhit Institute of Engineering and Technology (Deemed To-Be University), Meerut, India

Keywords Mobile agents · Degree of mobility · Euler totient function · Fermat Euler theorem · Distributing computing

1 Introduction

Mobile agent (MA) [1] based framework is an amendment on mobile agents for the distributed processing. It is a software process with intelligence that works on the behalf of its user. Mobile agent paradigm provides high degree of flexibility in processing. There are three main categories of computing i.e., Client server computing *(computing a server provides services to client),* Code on demand Computing *(sever sends executable code from a server to a client on the request from the client side*) and Agent Based computing *(An intelligent piece of code along with process form the mobile agents, which works on the bases of host and migrate automatically from one host to another host).*

In client sever approach, data moves from one user to another, but the movement of data takes more bandwidth of channel. In case of Mobile agent approach, instead of movement of data, a process moves from one host to another which takes less bandwidth of channel as compared to client–server approach. The mobile agent-based computing is shown in Fig. 1.

Mobile agent follows a life cycle during the communication to agent and platforn,as shown in Fig. 2

During lifetime of a mobile agent various phases that occur are Creation *(a newly agent is created and initializedstate of agent)*, Cloning*(duplicate mobile agent is created)*, Dispatch *(agent is dispatched and communicate to agent and platform)*, Deactivation*(agent is in sleep state save in to memory)*, Activation*(agent is activated from the memory)*, Retraction*(agent is ready to execute operation)*,

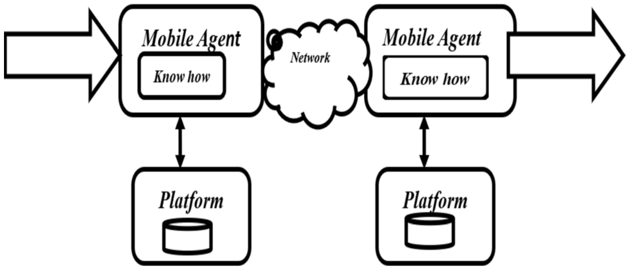

Fig. 1 Mobile Agent computing

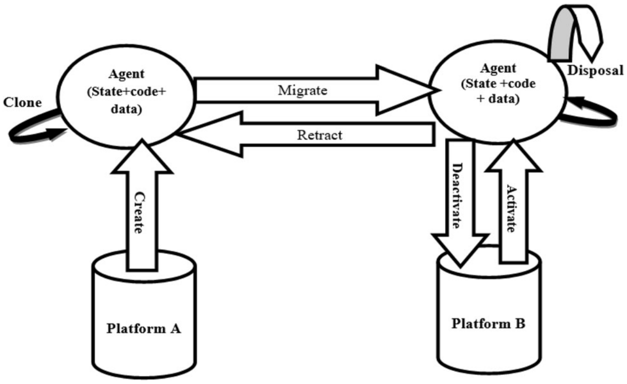

Fig. 2 Mobile Agent life cycle

Communication*(occur among agent and host)* and Disposal *(in the final sate a mobile agent terminated after completion of process).*

Security is a major concern for mobile agents [2] as they move from host to host. There are, Confidentiality *(in any developed frameworks, confidentiality should not be compromised during communications either by hoped agents or by different platforms under execution of agent process),* Data *integrity (data and information should be in original form not tampered by any third party. The integrity needs to maintain for any secure operation of mobile agents, both local as well as other platforms on which agent moves for execution.)* and Availability *(availability means data and information are required by platform or agents should be available. The agent platform will make it available to both local and remote agents).* Security parameters are shown in Fig. 3.

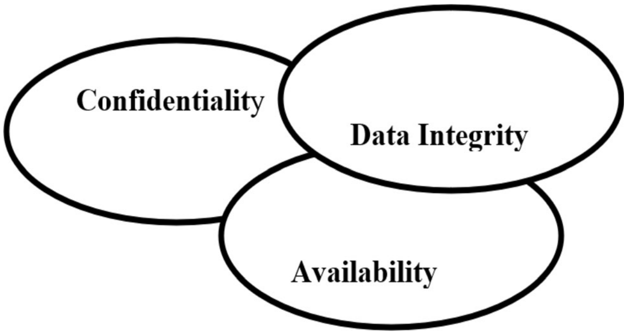

Fig. 3 Security parameters

Apart from these above mentioned parameters, authentication of identity of agents and platform is also needed for user actions [3].Major issue to design mobile agent paradigm [4]is security of mobile agent during the transmission on distributed networks. In the existing agent systems, most of the concentration was on deciding the working of agents [5] rather than security. Most of the schemes lack in implementation of an effective security for mobile agents. Some popular issues related to attacks on mobile agent frameworks are highlighted in Table1.

It shows that there is a need of securing mobile agents to support interoperability between agent's framework and security of paradigms. From centralized monolithic computing, Security of mobile agent has been less explored till date in research. So, there is a need of developing strategy that which can work in a secure distributed environment.

In this paper, a secure key management scheme of mobile agent in the agent-based environment has been proposed. Algorithm of proposed mathematical model is categorized into main three parts i.e., initialization, share creation and reconstruction of shares. In the initialization part, we select a secure key 'S' for authentication of mobile agent during execution and transformation of code. The creation of public share of every mobile agent $0 \le i \le n$ is done in second step of algorithm. At last, corresponding share is determined using threshold-based scheme (Si, pi).

The complete structure of article is as follows. Sections 2 and 3, have been describing mainly about the agent-based frameworks and problem statement. Section 4 presents proposed approach along with preliminaries. The performance evaluation of the approach along with implementation has been discussed in Sect. 5. Conclusions with future work has been highlighted in Sect. 6.

2 Related work

Disclosure of information [1], denial of service and, corruption of information are the main classes of threats to security. One can examine these classes of threats in greater detail as they apply to agent framework. Mobile agents simply offer a greater opportunity for abuse and misuse, broadening the scale of threats significantly.

A Verifiable Secret Sharing (VSS) technique based on Chinese Remainder Theorem (CRT) and Shamir's approach has been proposed by several researchers in the field of agent security. Piyanka Dadhich et al.[5] examines a variety of security techniques in depth. Security, particularly host-based attacks against visiting mobile agents (the malicious hosts problem), is a key roadblock to widespread

Table 1 Attacks on mobile agent frameworks

	Attacks of agent during execution on the platform	Attacks of agent during execution on agent	Attacks of platform-during execution on agent
Masquerading	✔	✔	✔
DOS	✔	✔	✔
Unauthorized access	✔	✔	
Repudiation		✔	
Eavesdropping			✔
Alteration			✔

use of mobile agent technology. The host, as the running environment for mobile agents, has complete control over them and can readily launch a variety of attacks against them.

Alfalayleh et al.[6] the main security challenges connected to the mobile agent paradigm are discussed in this study. Security risks, requirements, and approaches for keeping the mobile agent platform and the agent itself secure against each other are among the concerns addressed.

Kaur et al. [7] examines several forms of attacks, such as agent-based attacks, platform-based attacks, and so on. We then go over the many strategies that have been offered by researchers for both preventing and detecting attacks in detail. Finally, we offer a recommendation for an effective countermeasure.

Verma et al. [8] proposed idea of security using CRT which is helpful when shareholder is not honest and uses multiple secret shares in multilevel groups. All the participants are categorized into various levels and every level has dynamic threshold value. Reconstruction is done when sufficient numbers of shares are available. Iftene [9] proposed a scheme to support voting for person with multiple authority based on CRT. Each authority has different weight or threshold in the scheme. Lein et al. [10] proposed multiple level share designed based on CRT. Ersoy et al. [11] suggest a secret share based on the homomorphism aspect of CRT. Meng et al. [12] suggested a general access structure for secret sharing using CRT. It divides the secret in hierarchical structure in a way that higher level can access the lower-level share, to regenerate the secret. In Multilevel Secret Sharing Scheme (MTSS), only one secret is used in each level.

Sultanik et al. [13] describes the agents both the agent and the host level, the article explains the architectural technologies utilised in Secure wireless agent testing, the integration issues, as well as applications for group collaboration, network health monitoring, and system security. Mobile agents, security, and ad hoc networks are all becoming more important in next-generation computing and collaboration infrastructures.

A CRT based threshold RSA (Rivest, Shamir and Adleman) algorithm has been proposed by Sarkar et al. [14]. It generates consistent shares of the secret for the shareholder and provides security during reconstruction of key. Zou et al. [15] proposed a CRT based multiple secret key distribution technique using secure and fast electronic voting. Shyu et al.[16] described a secret image sharing technique based on CRT. A secret sharing technique based on sum of weights of shareholder involved in regeneration of secret has been suggested by Shi et al.[17]. This technique is based on fusion of Lagrange interpolation and CRT. Reddy et al. [2] proposed a protocol for general threshold based multiple secret sharing schemes that gives multiple secrets and recovery of secret is also visible in nature.

Deshmukh et al.[18] suggested a scheme of security using binary trees and boolean operation. The height of binary tree decides the security of scheme. Higher the height of tree, better is the security nature. In complete binary tree each node has random share and it does not reveal any data. If attacker has fewer shares than threshold value, data will not be revealed. A Scheme based on univariate and bi-variate symmetry polynomial to reduce coefficients of shares for each group is suggested by Meng et al. [19]. A sharing scheme based on robust reversible data hiding scheme suggested by Liu et al. [20] proposed a verifiable technique in which each shareholder is allocated a shadow share which provides the higher security during recovery phase. A combiner authenticates each shareholder before submitting information.

Cheating detection during generation of verifiable shares is also proposed by many researchers using different techniques. Xun Yi et al. [21] proposed two efficient (t,n) Threshold Password Authentication Secret Sharing (TPASS) technique for any ‘n’ greater than ‘t’ that provides security of secret of the user during the reconstruction of secret. Xiaotian Wu et al. [22] designed sharing scheme (k,n) Secret Image Sharing for Distributed Cloud Network (SISDCN). Where, ‘k’ shareholder can reconstruct share using Distributed Cloud Network (DCN) on images. Yanxiao Liu et al. [23] proposed two cheating detection scheme on (k, n) sharing technique of secret. First scheme identifies the cheating and second provides a higher authentication for secret regeneration.

Binue and Kumar [24] proposed a dynamic technique designed on elliptic curve, to provide higher security. This scheme also identifies cheating identification. A fast key exchange algorithm based on special key sharing technique has been proposed by Yi Sun et al. [25]. The regeneration of the group session key is completed by the shareholder itself. Xie et al. [26] suggested a modular arithmetic based scheme for security of key. Tong Zhang et al. [27] proposed a threshold based multi sharing secret scheme. Taihei Watanabe et al. [28] also proposed multiplication based on Shamir’s (k, n) using ‘k’ servers [29]. This technique provides higher secret evolution without alteration of threshold value.

The work carried out by above researchers in the direction of security of mobile agent paradigm is mainly concerned with the platform security rather than the mobile agent. During the hoping of agents from one hop to another, security of agent as well as platform security should be of concern to save it from any adversaries.

3 Proposed work

One need to have robust and efficient security mechanism to make the mobile agent framework more secure in nature. The popular encryption technique has been solving the security issues using traditional ways of generating key. The encryption techniques in turn leads to key handling issues and different attacks and its countermeasures required to make an agent secure enough. Security of mobile agent is entirely based on robustness of a key which an agent generated and its secure key management techniques. The problem in case of mobile agent security is beginning from source platform of agents that is completely trusted and secure. This trusted environment is difficult to carry to other agent platforms during the course of its hoping. When an agent is moved to another hop, its protection is minimal compared to source hop. Such agent protection scheme may be adequate based on few applications, but it is not optimal in nature. Execution tracing, Partial Result Encapsulation, and Mutual Itinerary Recording are few techniques for detecting unauthorized modifications of an agent’s behavior.

The proposed approach describes that describes a scheme in which an agent moves between the hops and maintains its security based on its key and threshold decided. The non-trusted hop station or hackers of agent code requires a proper threshold value to open the process associated with the key to the agent. For the security of mobile agent, proposed approach is based on Euler’s totient function and Fermat Euler theorem. A secret key for the execution and authentication of mobile agent has been generated and used at mobile host during the life cycle. Secret is divided in to ‘n’ number of shares based on proposed approach and it selects a random value between lower and upper bound. The mechanism uses a threshold-based decision for a particular key shared during movement for authenticating a mobile agent.

3.1 Preliminaries

The basic preliminaries used here such as, Shamir’s scheme, Euler’s theorem, Lagrange’s Interpolation and CRT. Euler totient function and Fermat Euler theorem has been used here to generate secure key for agent. Shamir’s threshold scheme along with CRT is used to compare the designed scheme on polynomial interpolation.

Chinese Remainder Theorem (CRT)[14] is used to generate the threshold value at each distributed at the second level of the threshold secret sharing scheme. Consider the co-prime integer p_1, p_2, $p_{3,}$ …p_n andα_1, α_2, α_3 …. α_n random integer ‘x’ system of simultaneous congruence relation,

$$\begin{aligned} x &\equiv \alpha_1 (\mathrm{mod}\ p_1) \\ x &\equiv \alpha_2 (\mathrm{mod}\ p_2) \\ &\cdots \\ x &\equiv \alpha_n (\mathrm{mod}\ p_n) \end{aligned} \tag{1}$$

has a unique solution modulo, P = p_1, p_2, $\cdots$ p_n, for any given integers $\alpha\ _1$, $\alpha\ _2$,..., $\alpha\ _n$. P = $p_* p_2 * \cdots * p_n$. $x \equiv \alpha\ _1 P_1 c1 + \alpha\ _2 P_2 c2 + \ldots + \alpha_n\ P_{ncn}$ (mod P). where P_{i-} = = P/n_i and $c_i \equiv P_i^{-1}$ (mod p_i).

Fermat Euler theorem: In number theory Euler is very important concept, Euler totient Function (φ (n)): For n ≥ 1, φ(n) represents the total number of positive integers less than n and co prime to n. If n is prime number, the Euler totient returns φ (n) = (n − 1).

If n is not prime φ(n) = n(1 − 1 /p_1)(1 − 1/p_2) .(1 – 1/p_n), $p_1 < p_2 < \ldots < p_n$ prime numbers.

Fermat Euler theorem based on Euler function theorem says that for positive integer n, and δ in such a way, gcd(δ, n) = 1, then

$$\delta^{\varphi(n)} \equiv 1 \mathrm{mod}\ n \tag{2}$$

where φ(n) is Euler’s totient function.

Shamir’s Secret Sharing [29]: Let us consider β_0, β_1, $\beta_{2,}$… $\beta_{k-1} \in$ GF (p)F(x) = ($\beta_0\ x^0 + \beta_1\ x^1 + \beta_2\ x^2 + \ldots + \beta_{t-1} x^{t-1}$) modp, F(0) = β_0 = session key and ‘p’ is a large prime number and β_1, β_2…, and β_{k-1} are randomly chosen real number from Z/PZ. On the basis of node identity generatre ‘n’ partial keys.At the reciver side,select ‘t’ randomly share out of ‘n’ partial share and generatre lagrange polynomail,

$$\mathbf{F}(\mathbf{x}) = \sum_{\mathbf{i}=1}^{\mathbf{k}} \Upsilon_{\mathbf{i}} \prod_{1\le \mathbf{j}\le \mathbf{k}, \mathbf{j}\ne \mathbf{i}} \frac{\chi - \chi_{\mathbf{j}}}{\chi_{\mathbf{i}} - \chi_{\mathbf{j}}}. \qquad (3)$$

Since f (0) = β_0 = S, the secret key evalute using

$$\mathbf{Secretkey}(\mathbf{S}) = \sum_{i=1}^{k} \boldsymbol{P_i \Upsilon_i} \qquad (4)$$

where

$$\boldsymbol{P_i} = \prod_{1\le j\le k, j\ne i} \frac{\chi_j}{\chi_j - \chi_i}$$

Secret share is genrated by using ‘t’ partial share by using F(0) = $\boldsymbol{\beta_0}$modp.

3.2 Countermeasure of agent security

To unlock process associated with the mobile agent, a security scheme based on robust key generation method is proposed here. The cryptographic condition is hidden through number of threshold and secret key divided among number of shares. For the security of mobile agent an approach based on Euler totient function and Fermat Euler theorem has been proposed. In this secure agent framework, a secret key for the execution and authentication of mobile agent is generated at mobile host during the life cycle. Secret is divided in to ‘n’ number of shares based on scheme shown in algorithm and select a random value between lower and upper bound. We have ‘n’ shares (as shown in Fig. 4), {S1, P1}, {S2, P2}, {S3, P3}, {S4, P4} ….. {S5, Pn}. Platforms at which mobile agents want to execute their task the on behalf of user reconstruct the secret key for execution and authentication of mobile agent.

Algorithm of proposed mathematical model is made in three parts initializations, share creation and reconstruction of share.

i. *Initialization:* Host (User) select n positive integer in such a way
- Select a secret key S ($0 \le S < p_0$) *for* authentication of mobile agent during execution and transformation of code.
- Select positive integer $p_0 < p_1 < p_2 < p_3 \ldots\ldots\ldots\ldots < p_n$
- *gcd (pi, pj) = 1 for every i* $\ne$ *j* $\le$ *n* $\prod_{i=1}^{t} \boldsymbol{p_i} > (p0 + 1)$.
- $\prod_{i=1}^{t-1} \boldsymbol{p_{n-t+i+1}}$

ii. *Share creation.*
- $\boldsymbol{S_i}$ **= (*S*+ α*p*$_0$) *mod p*$_i$public** share of every Mobile agent $0 \le i \le n$
- $\boldsymbol{\alpha\epsilon}\left(\frac{\prod_{i=1}^{t}(pn-i+1)}{p0}, \prod_{i=1}^{t}\frac{pi}{po} - 1\right)$

iii. *Reconstruction of Secret share S*
- ***gcd(A, B):***
- ***if (a = = 0)***
- *return b*
- *return gcd(B% A, A)*
- ***Euler totient φ (p$_j$):***
- *R = 1*
- *for i = 2 to n*
- *if (gcd(i, n) = = 1):*
- *R = R + 1*
- *return R*
- corresponding share are **(*S$_i$*, *p$_i$*)** *to respective modulo p$_i$ 0 ≤ i ≤ n*
- ***S’ =*** $\sum_{j=1}^{t} \boldsymbol{sj} * (\prod_{1\le k\le t, k\ne j} \boldsymbol{pk})^{\varphi} \;{}^{(}\boldsymbol{p_j}{}^{)}$ ***mod*** $\prod_{1\le k\le t,k} \boldsymbol{pj}$
- S = ***S’mod p$_0$ /* p$_0$ Is public parameter***

The algorithm makes a reliable environment in such a way that a malicious agent cannot uncover the agent process by reading straight the code of agent. The developed technique focuses on mobile agent security and evolves along the traditional lines of key generation security techniques of mobile agent framework. A simple example below demonstrates how the proposed scheme works in different chosen threshold values for ‘n’ number of distributed shares for a secret key generated using proposed approach.

Considering **n + 1 integer’s [*p0–p4*]** based on the algorithm. Select Secret(**S**) = 6, S < *p*0 and α = 19.

i. N (no. of shares) = 4, T(threshold) = 2
List of *p0, p1, p2, p3, p4***:** [7, 11–13, 17]
Enter the value of α = 19,
Enter the value of secret share **S = 6,** Generated shares: [3, 7, 7, 9]
*p1, p2, p3, p4***:** [11–13, 17]**,**
t = 2 take any two pair share 1 and share 4 (7, 11) and (3, 17)
S1 = (7(17)$^{\varphi(11)}$ + *3(11)*$^{\varphi(17))}$*%(11*17) = 139*
S = 139%7 = 6. Reconstructed secret share = 6.

ii. n = 4, t = 3, Enter the value of α = 19.
Enter the value of secret share **S = 6, Generated** shares: [3, 7, 7, 9], p*1, p2, p3, p4***:** [11–13, 17]
t = 3 take any three random pair (7, 11), (9, 13) and (3, 17),
S1 = (7(17*13)$^{\varphi(11)}$ +3(11*13) $^{\varphi(17)+}$ 9*(11*17) $^{\varphi\ (13))\ \%}$ (11*17*13) = 139
S = 139%7 = 6. Reconstructed secret share = 6.

iii. n = 4, t = 4 Enter the value of α = 19.

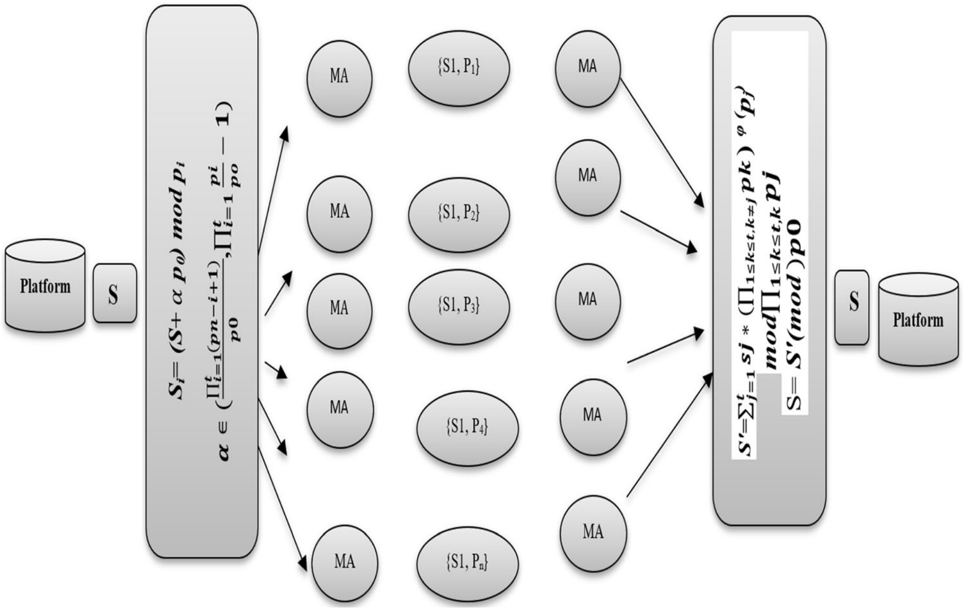
Platform
S
$S_i = (S + \alpha\, p_0) \bmod p_i$
$\alpha \in \left(\frac{\prod_{i=1}^{t}(pn-i+1)}{p0}, \prod_{i=1}^{t}\frac{pi}{po} - 1\right)$
MA
{S1, P1}
MA
{S1, P2}
MA
{S1, P3}
MA
{S1, P4}
MA
{S1, Pn}
MA
MA
MA
MA
MA
$S' = \sum_{j=1}^{t} sj * (\prod_{1\le k\le t, k\ne j} pk)^{\varphi(pj)} \bmod \prod_{1\le k\le t,k} pj$
$S = S'(mod)p0$
S
Platform

Fig. 4 Proposed framework based on Euler totient function and Fermat Euler theorem

Enter the value of secret share **S = 6,** Generated shares: [3, 7, 7, 9], *p1, p2, p3, p4*: [11–13, 17], t = 4 take all pair at a time (7, 11), (9, 13), (7, 12) and (3, 17),

$S1 = (7(17*13*12)^{\varphi(11)} + 3(11*13*12)^{\varphi(17)} + 9*(11*17*12)^{\varphi(13)} + 7*(13*11*17)^{\varphi(12)})\%(11*17*13*12) = 139$

S = 139%7 = 6. Reconstructed secret share = 6.

4 Implementation and results

The proposed approach based on Euler's theorem with n = 10 shares along with k = 2 onwards has been picked up for different threshold values. Table 2 represents the time taken by CRT to completely generate and reconstruct any integer range secret code value divided into chosen number of shares and different chosen threshold value as indicated. Table 3 shows the time elapsed to same process using proposed scheme using the same set conditions as set earlier.

It is observed from the results that the execution time of proposed approach is quite low as compared to CRT scheme for agent's key security. The best case and average case analysis of proposed scheme have also been checked over CRT based key generation and italic highlights in the tables indicate the obtained result in best case. Average case value has been picked up based on ten different results obtained on the same set conditions. Every 'n' number of shares created was distributed on different machine instances and implementation is done using python language. Here, 'n' number of shares generated on the basis of Euler theorem, out of 'n' share we want at least 't' share to regenerate the secret share for execution of agents on

Table 2 Time taken versus threshold for CRT

CRT t	2	4	6	8
1	0.0031	0.0055	0.0066	0.0059
2	0.0035	*0.0034*	0.0052	0.0069
3	0.0057	0.0054	0.0065	*0.005*
4	0.006	0.0057	0.0068	0.0081
5	0.0065	0.0065	*0.0041*	0.0069
6	*0.003*	0.0079	0.0047	0.0056
7	0.0067	0.0048	0.0058	0.0069
8	0.0061	0.0068	0.0087	0.0078
9	0.0046	0.0074	0.0073	0.0079
10	0.0065	0.0033	0.0049	0.0051
Avg	0.00517	0.00567	0.00606	0.00661

Table 3 Time taken versus threshold for Euler's totient

Euler t	2	4	6	8
1	0.0034	0.0042	0.0055	0.0071
2	*0.0027*	0.0033	0.0045	*0.0045*
3	0.0048	0.0051	*0.0035*	0.0057
4	0.0058	0.0055	0.0066	0.0059
5	0.0052	0.0041	0.0044	0.0056
6	0.0066	0.0055	0.0042	0.0057
7	0.0043	0.006	0.0046	0.0058
8	0.0035	*0.0029*	0.0043	0.0048
9	0.0047	0.004	0.0059	0.006
10	0.0049	0.0052	0.0042	0.0051
Avg	0.00459	0.00458	0.00477	0.00562

platform. If less than ‘t’ share wants to regenerate secret share they can’t generate. Security of this algorithm is based on the Euler totient function.

The graph shown below is based on the experimentation done with different settings as indicated above. Thresholds were introduced to check the recovery of dynamically decided value alternately to make the system more robust. At most of the tested value, the traditional key generation based on CRT was found to be low performer with respect to proposed scheme. It was observed that the performance of the proposed system was consistent in terms of time efficiency when compared to CRT. In such a view, it was found that the proposed system was much advantageous in detecting and counters measuring the attacks on agent-based security in any agent framework. Performance between threshold values versus total turnaround time taken to regenerate the session in average case is shown in Fig. 5. Performance between Threshold values versus total turnaround time taken to regenerate the session in Best case is shown in Fig. 6.

Table 4 represents the time taken by CRT to completely generate and reconstruct any integer range secret code value divided into different number of shares in plane. Table 5 shows the time elapsed to same process using proposed scheme using the same set conditions as set earlier. It is observed from the experimentation that the execution time of proposed approach is still better even in case of different numbers of share plain chosen as the compare to CRT scheme for agent’s key security. The best case and average case analysis of proposed scheme have also been checked over CRT based key generation and italic highlights in the tables indicate the obtained result in best case. Average case value has been picked up based on ten different results obtained on the same set conditions.

Figure 7 shows the performance between Numbers of mobile agent versus total turnaround time taken to regenerate the session in average case. Figure 8 shows the

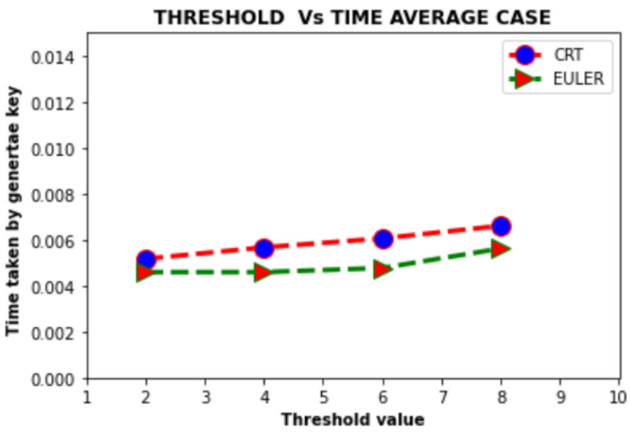

Fig. 5 Threshold value versus Time Average case

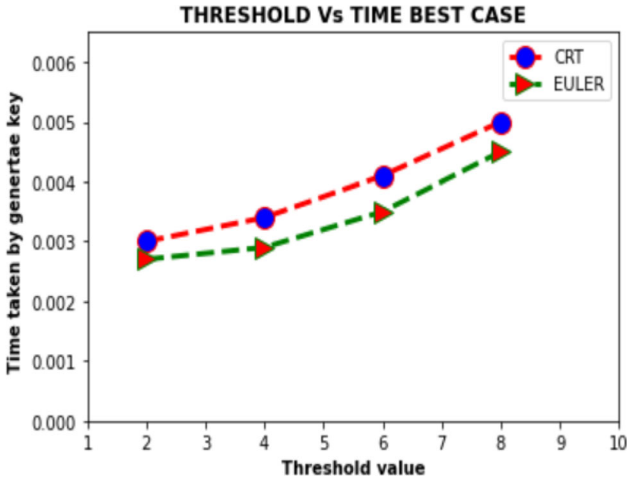

Fig. 6 Threshold value versus Time Best case

Table 4 Time taken versus number of mobile agent for CRT

CRT	No. of mobile agents					
Sr. no.	5	10	15	20	25	30
1	0.0065	0.0045	0.0084	0.011	0.0136	0.0079
2	0.0037	0.006	0.007	0.0096	0.0073	0.0081
3	0.0044	0.0068	0.0073	0.0127	0.0092	0.0086
4	*0.0033*	0.007	0.0083	0.0089	0.0081	0.0098
5	0.0041	*0.0042*	0.0076	*0.0068*	0.0103	0.0136
6	0.0052	0.0052	0.0048	0.007	*0.0061*	*0.0074*
7	0.0039	0.0057	0.0078	0.0087	0.0092	0.0128
8	0.0043	0.0055	0.0061	0.0084	0.007	0.0098
9	0.004	0.0062	0.0062	0.0076	0.0099	0.0087
10	0.0039	0.0094	*0.0042*	0.0075	0.0098	0.0097
Avg	0.0043	0.0061	0.0068	0.0088	0.0091	0.0096

Table 5 Time taken versus number of mobile agent for Euler’s totient

Euler	No. of mobile agents					
Sr. no.	5	10	15	20	25	30
1	0.003	0.004	0.006	*0.005*	0.009	*0.006*
2	0.004	0.007	0.008	0.01	0.006	0.008
3	0.004	0.005	0.006	0.006	0.016	0.013
4	0.004	0.004	0.007	0.007	0.01	0.009
5	0.004	0.005	0.008	0.007	0.008	0.007
6	0.003	0.005	*0.004*	0.01	0.011	0.01
7	0.004	0.006	0.007	0.017	0.01	0.01
8	*0.003*	*0.004*	0.007	0.009	*0.005*	0.016
9	0.004	0.006	0.004	0.008	0.01	0.009
10	0.004	0.007	0.01	0.008	0.006	0.008
Avg	0.004	0.005	0.006	0.009	0.009	0.009

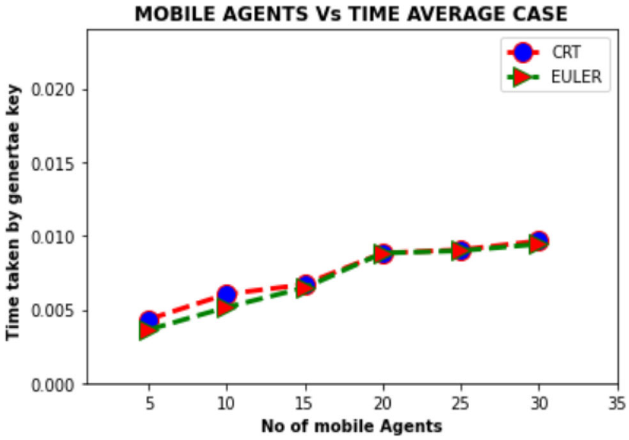

Fig. 7 No of Mobile agents Versus Time in Average case

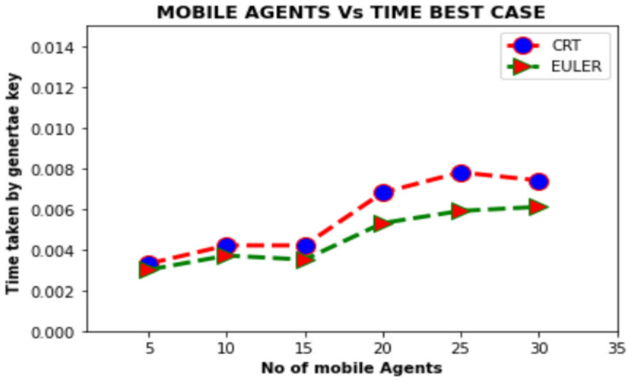

Fig. 8 No of Mobile agents Versus Time in Best case

performance between Numbers of mobile agent versus total turnaround time taken to regenerate the session in Best case. The graph shown is based on the experimentation done with different numbers of mobile agents as indicated above. Number of mobile agents was introduced to check the recovery of secret key from the scheme and observations were made to check the nature of complexity of algorithms based on increasing number of shares plane of secret information.

At most of the tested value, the proposed scheme generates optimal and fast result compared to traditional approaches. The analysis of key generated with proposed scheme and CRT along with Shamir's approach has been compared in terms of security features and time complexity. The analysis of scheme indicates that the secret is safe, and adversary cannot make any effect as threshold is chosen dynamic for different hops before reconstruction. The secret can be recovered in O (t) number of operation while O ($t\log^2 t$) number of operations is needed for Shamir's scheme which indicates that proposed scheme has linear complexity and gives better result.

Table 6 show the Comparison of proposed Euler theorem-based approach with another secret sharing technique. As a result, proposed scheme time complexity is better than another scheme. In case of proposed scheme threshold value has no limitation. If approach scheme has partial share more than threshold value 't'.

5 Conclusion and future scope

Securing the mobile agent during communication between different platforms is still a crucial issue. In this paper, an extension by fusion of Euler's totient and Fermat theorem has been designed to make it more secure. It gives the insurance of key security using threshold values and a robust key management scheme. It helps to monitor the security of access of agent only to legitimate number of users. Authentication of any agent by other hosts is equally important parameter of security. The model extends the key security as well as key management scheme. The time complexity of Euler totient-based framework is linear. It focuses towards improving the security aspects of agents and in turn, it opens up new directions of other research to meet other security requirements.

Table 6 Comparison of proposed Scheme with another Scheme

Technique	Share Generation complexity	Recreation time complexity	Threshold value 't'
Combinatorial Technique	Big oh(n)	Big oh(n)	t = 1 or t = 1
Strongly ideal secret sharing schemes	Big oh(n)	Big oh(n)	t = 2 or t = 1
Anonymous secret sharing schemes	Big oh(n)	Big oh(n)	t = 2 or t = 1
On the bound for anonymous secret sharing schemes	Big oh(n)	Big oh(n)	t = 2 or t = 1and other cases
Providing anonymity in unconditionally secure secret sharing scheme	Big oh(n)	Big oh(n)	t = 2 or t = 1and other cases
Proposed scheme	Big oh($t\log^2 t$)	O(t)	No limitation

In future, more generalized model may be developed to cater the need of agent's security as well as identifying the untrusted hosts that will be useful for critical application. This will help to save the computing paradigm of mobile agents from cheating and several new application areas will be benefited from the proposed work.

References

1. Nair MK (2011) a Pplying W Eb S Ervices W Ith M Obile a Gents. Comput Netw 3(2):125–144
2. Reddy LS, Prasad MVNK (2015) Multi-secret sharing threshold access structure. In: 2015 Int Conf Adv Comput Commun Informatics ICACCI 2015, pp. 1585–1590. https://doi.org/10.1109/ICACCI.2015.7275839.
3. Jansen W, Karygiannis T (1999) NIST special publication 800–19–mobile agent security computer. Nist Spec Publ 323(September):3–10
4. Xu H, Zhang Z, Shatz SM (2005) A security based model for mobile agent software systems. Int J Softw Eng Knowl Eng 15(4):719–746. https://doi.org/10.1142/S0218194005002518
5. Dadhich P, Govil MC, Dutta K (2010) Security measures to protect mobile agents. AIP Conf Proc 1324(December):298–302. https://doi.org/10.1063/1.3526218
6. Alfalayleh M, Brankovic L (2005) An overview of security issues and techniques in mobile agents. IFIP Adv Inf Commun Technol 175:59–78. https://doi.org/10.1007/0-387-24486-7_5
7. Kaur M, Saxena S (2017) A review of security techniques for mobile agents. Proc IEEE Int Conf Comput Commun Autom ICCCA 2017:807–812. https://doi.org/10.1109/CCAA.2017.8229906
8. Verma OP, Jain N, Pal SK (2020) A hybrid-based verifiable secret sharing scheme using Chinese remainder theorem. Arab J Sci Eng 45(4):2395–2406. https://doi.org/10.1007/s13369-019-03992-7
9. Iftene S (2007) General secret sharing based on the Chinese remainder theorem with applications in E-voting. Electron Notes Theo Comput Sci 186(SPEC. ISS.):67–84. https://doi.org/10.1016/j.entcs.2007.01.065
10. Harn L, Fuyou M (2014) Multilevel threshold secret sharing based on the Chinese remainder theorem. Inf Process Lett 114(9):504–509. https://doi.org/10.1016/j.ipl.2014.04.006
11. Ersoy O, Pedersen TB, Anarim E (2020) Homomorphic extensions of CRT-based secret sharing. Discret Appl Math 285:317–329. https://doi.org/10.1016/j.dam.2020.06.006
12. Meng K, Miao F, Huang W, Xiong Y (2020) Threshold changeable secret sharing with secure secret reconstruction. Inf Process Lett 157:105928. https://doi.org/10.1016/j.ipl.2020.105928
13. Sharma VP (2012) Secure mobile agents on ad hoc wireless networks 476444: 1–9
14. Sarkar S, Kisku B, Misra S, Obaidat MS (2009) Chinese remainder theorem-based RSA-threshold cryptography in MANET using verifiable secret sharing scheme. In: WiMob 2009—5th IEEE Int. Conf. Wirel. Mob. Comput. Netw. Commun, pp 258–262. https://doi.org/10.1109/WiMob.2009.51
15. Zou X, Maino F, Bertino E, Sui Y, Wang K, Li F (2011) A new approach to weighted multi-secret sharing. Proc. - Int. Conf. Comput. Commun. Networks, ICCCN, pp. 0–5, 2011. https://doi.org/10.1109/ICCCN.2011.6005766.
16. Shyu SJ, Chen YR (2008) Threshold secret image sharing by Chinese remainder theorem. Proc. 3rd IEEE Asia-Pacific Serv. Comput. Conf. APSCC 2008, pp. 1332–1337. https://doi.org/10.1109/APSCC.2008.223.
17. Shi G, Ci Y, Xie R, Wang H, Zeng J (2016) A dual threshold secret sharing scheme among weighted participants of special right. In: Proc. - 2016 IEEE 1st Int. Conf. Data Sci. Cyberspace, DSC 2016, pp. 104–108. https://doi.org/10.1109/DSC.2016.82.
18. Deshmukh M, Nain N, Ahmed M (2019) Secret sharing scheme based on binary trees and Boolean operation. Knowl Inf Syst 60(3):1377–1396. https://doi.org/10.1007/s10115-018-1268-9
19. Meng K, Miao F, Yu Y, Lu C (2018) A universal secret sharing scheme with general access structure based on CRT. In: Proc - 17th IEEE Int Conf Trust Secur Priv Comput Commun 12th IEEE Int Conf Big Data Sci. Eng. Trust. 2018, pp. 142–148. https://doi.org/10.1109/TrustCom/BigDataSE.2018.00031.
20. Liu Y, Chen L, Hu M, Jia Z, Jia S, Zhao H (2016) A reversible data hiding method for H.264 with Shamir's (t, n)-threshold secret sharing. Neurocomputing 188:63–70. https://doi.org/10.1016/j.neucom.2014.10.109
21. Yi X et al (2019) Efficient threshold password-authenticated secret sharing protocols for cloud computing. J Parallel Distrib Comput 128:57–70. https://doi.org/10.1016/j.jpdc.2019.01.013
22. Wu X, Yang C-N, Li J-M (2020) Secure image secret sharing over distributed cloud network. Signal Process 178:107768. https://doi.org/10.1016/j.sigpro.2020.107768
23. Liu Y, Yang C, Wang Y, Zhu L, Ji W (2018) Cheating identifiable secret sharing scheme using symmetric bivariate polynomial. Inf Sci (NY) 453:21–29. https://doi.org/10.1016/j.ins.2018.04.043
24. Binu VP, Sreekumar A (2017) Secure and efficient secret sharing scheme with general access structures based on elliptic curve and pairing. Wirel Pers Commun 92(4):1531–1543. https://doi.org/10.1007/s11277-016-3619-8
25. Sun Y, Wen Q, Sun H, Li W, Jin Z, Zhang H (2012) An authenticated group key transfer protocol based on secret sharing. Procedia Eng., vol. 29, no. Wenmin Li, pp. 403–408. https://doi.org/10.1016/j.proeng.2011.12.731.
26. Xie Q, Shen Z, Yu X (2008) Threshold signature scheme based on modular secret sharing. In: Proc 2008 Int Conf Comput Intell Secur CIS, vol. 2, pp. 442–445. https://doi.org/10.1109/cis.2008.78.
27. Zhang T, Ke X, Liu Y (2018) (t, n) multi-secret sharing scheme extended from Harn-Hsu's scheme. Eurasip J Wirel Commun Netw 2018(1):3. https://doi.org/10.1186/s13638-018-1086-5
28. Wang G, Yan W, Kankanhalli M (2017) Content based authentication of visual cryptography. Multimed Tools Appl 76(7):9427–9441. https://doi.org/10.1007/s11042-016-3549-1
29. Naor M, Shamir A (1995) Visual cryptography. In: Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), vol. 950, pp 1–12.https://doi.org/10.1007/bfb0053419