Volume: 10 Issue: 06 Pages: 4070-4075 (2019) ISSN: 0975-0290

Self-Protected Mobile Agent Paradigm for DDoS Threats Using Block Chain Technology

Mona Nasr
Department of Information Systems, Faculty of Computers & Information
Helwan University, Egypt
Email : m.nasr@helwan.edu.eg

ABSTRACT

This paper describes Mobile Agents paradigm for tracking and tracing the effects of Denial of Service security threat in Mobile Agent System, an implementation of this paradigm has been entirely developed in java programming language. The proposed paradigm considers a range of techniques that provide high degree of security during the mobile agent system life cycle in its environment.

This paper highlights the spot to two main design objectives: The importance of including various supportive types of agents within a system e.g., police agents, service agents, …etc. Second: Evaluation analysis and number of checks to be done to trace the Mobile Agents if denial of the provided services during its path. Evaluation analysis for detecting tolerance differences for the calculated agent's route before and during its journey, storing agent transactions, storing snapshots of agent state information, checking from time to time agent status and task completeness and lastly guard agent checks the changed variables of migrated agent. During tracing and monitoring Mobile Agents, the initiator node may destroy it and continue with another. In this paper a new paradigm is presented that detects and eliminate with high probability, any degree of tampering within a reasonable amount of time, also provide the ability of scalability of security administration.

Keywords: - Mobile Agents, Denial of Service DDoS, Security threats, Block Chain Technology BCT, Trust

Date of Submission: Feb 13, 2019 Date of Acceptance: May 25, 2019

I. INTRODUCTION

Due to its salient properties, mobile agent technology has received a rapidly growing attention. Many developments of mobile agent systems in both academic and industrial environments. Security is the most important principle when talking about distributed systems. Security in distributed systems can roughly be divided into two parts. [2] One-part concerns with the communication between users and processes, possibly residing in different machines. The principle mechanism for ensuring secure communication is that of a secure channel. Secure channels, and more specifically, authentication, message integrity, and confidentiality. [1] The other part concerns authorization, which deals with ensuring that a process gets only those access rights to the resources in a distributed system it is entitled to. Most of the researches work into security is concentrating on the malicious agent issue, by advancing techniques that isolate the execution of agents from the rest of the system. However, isolation on its own is only a first step for security. [4] Since code mobility is subject to intense research, code mobility creates new security problems, for the executing host as well as the program to execute. Today we must acknowledge that we do not have good answers for all these security problems, such as those related to security of mobile agent systems. Recent studies show that binding Mobile Agents and blockchain technologies BCT are gathering theoretical contributions from many fields. [5]

Recent studies focus on the integration with the expectation of providing BCT features in use-cases where agent systems require them. [14,18]

II. MOBILE CODE SECURITY

Migration various in three main levels for migrating code, as shown in table (1) mobile code, mobile object, and mobile agent.

Sn.	Equation	Result Concerns
1	Migrating Code	= Mobile Code
2	Migrating Code + Data	= Mobile Object
3	Migrating Code + Data + Execution State	= ***Mobile Agent***

Table 1: Levels of Migration

The common hostile actions affecting the agent are masquerading, eavesdropping, repudiation of action, alteration of carried results, and denial of service attacks.

Another way on looking at security in computer systems is that we attempt to protect the services and data it offers against security threats. There are four types of security threats: interception, interruption, modification, and fabrication. [12,13]

Platform to Agent (P2A)
Masquerading, denial of service, eavesdropping, and alteration

Volume: 10 Issue: 06 Pages: 4070-4075 (2019) ISSN: 0975-0290

Agent to Agent (A2A)
Masquerading, denial of service, repudiation, and unauthorized access.

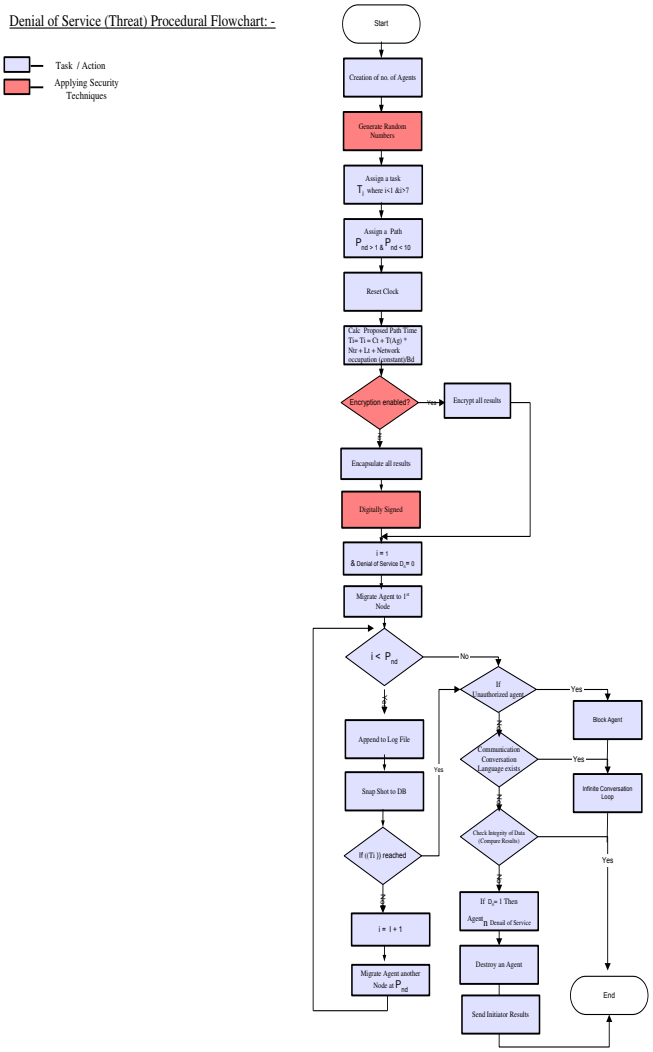

Fig. 1 Denial of Service Threat Procedural Flowchart

III. THE PROPOSED CAT'S EYE PARADIGM

The proposed paradigm enforces applying security in mobile agent system. [15] Its security mechanism reduces threats that affect systems' resources and accessing vulnerability. Any system must guard itself from any accessing that happened to its available resources. [6,7,8] These resources can be divided to built-in resources and indirect resources, example of the Built-in resources such as screen, the file system, memory, real time, and CPU time. Knowing that building all kinds of security mechanisms into a system does not really make sense unless it is known how these mechanisms are to be used. [9,19]

The security between host and agent is twofold: from one side, hosts must be protected against malicious agents, on the other side; agents have also to be protected against malicious hosts. The first direction, protection against agents, can be solved using existing technology known as Java Applets and SafeTcl programs, since there the same problem exists, the execution of unknown programs. Both systems use an approach, the so-called *Sandbox* security model, where all potentially dangerous procedure calls are restricted by special security control components that decide which programs can use these procedures. [21]

Cat's eye Mobile Agents-based paradigm is a simulator that has been entirely developed in java programming language, which consists of various types of classes and routines. Java programming language has been chosen mainly due to inheritance capability, rich class hierarchy and dynamic class loading capability.

Giving an example from real life e.g.: Assuming we visit Bee Network, there are two types of bees where found the queen bee and the servant bees. The "servants" bees are many, but the queen is one.

Police/ Guard agents = Queen Bee
Service agents = Servant bee(s)

Police/ Guard agent features (Queen Bee)

a) Check the trusty of the host.

b) Monitoring various nodes, to detect agent parameters and variables at each node and order for destruct of an agent if unexpected modification had been happened and notify the owner machine.

Service agent features (Servant bee(s))

a) Perform multiple tasks.
b) It assumed to travel to un-trusted and trusted nodes.
c) Don't hold sensitive information
d) Its credentials being checked from the visited node.

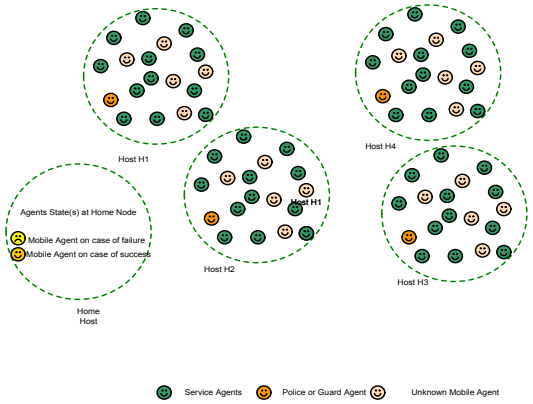

Fig. 2 Detailed Scope of the Agent Community and the behavior of the agent at visited node

IV. CAT'S EYE SCENARIO

In our paradigm Blockchain technology is suggested as an immutable, and transparent append-only register of all the actions that have happened to all the agents in the

Volume: 10 Issue: 06 Pages: 4070-4075 (2019) ISSN: 0975-0290

network. [5,17] It is secured using cryptographic primitives such as hash function, digital signature, and encryption.

During the agent life cycle from creation time till its destruction, detailed steps for each functionality are shown in Fig. 1. We try in this model to simulate some of the threats that may attack the agent during its life in a real environment.

If the agent has visited a node that can attacked it, doing modification to its state, values the next visited node can detect the destruction happen to the agent body, also Cat' Eye mobile agent with its guarding eye execute from home node to know if any modification has been happened to the agent this routine has all the agent input parameter and used variables, so any changes can be easily detected.

We supposed a combination of some techniques that provide high degree of security during the mobile agent system life cycle and its environment in addition to suggesting some new security goals represented in this paper.

The Agent is created at creation time at its agent owner node. The creation time here we assumed that it is a milestone of agent creation, after that the agent migrates from its agent owner node through its route (path) between various nodes. The mobile agent may be faced by various threats, it may face with extra tasks make it live locked or may attacked by a malicious host/node as a way of a threat (denial of Service), they may steel, corrupt or divert an agent from it right route. The agent may override the threat and arrive to its owner node successfully at a return time. This return time must be pre-calculated in-order to detect the delay due to prioritization, or network delay.

Security is the most important principle when talking about distributed systems. This complexity is due mainly to the size of the problem, moreover, in open networks, security problems and costs of potential solutions might outweigh the benefits of mobility. The previous sections discuss Cat's Eye Mobile Agent paradigm; this section shows the security aspects that need to be considered for Cat's Eye Mobile Agents-Paradigm: As security is the most critical issue in a mobile-agent system. We must try to detect tampering as soon as the agent migrates to/from a malicious machine during its path and back to its owner machine. On case of tampering of an agent we can terminate, fix, and destruct the agent.

Authentication: Verify the identity of an agent's owner.

Authorization and enforcement: Assign resource limits to the agent based on the identity and enforce those resource limits.
Mobile agent not migrate to nodes had lower degree of trustworthy.
Sensitive migrated data must be in an encryption form.
Audit log be created on case of failure, for keeping and comparing the inconsistency that may happen during the journey.
The scope of work is for protecting agent in mobile agent systems.

V. CASE STUDY

Mobile agent is a program that moves from machine to machine and executes on each. As an example of Cat's Eye paradigm, As shown in the following figure Cat's Eye simulation environmemt network consists of three entrance points(domains) to many disricts in real life network at taxing authority (Ministry of Finance in Arab Republic of Egypt).

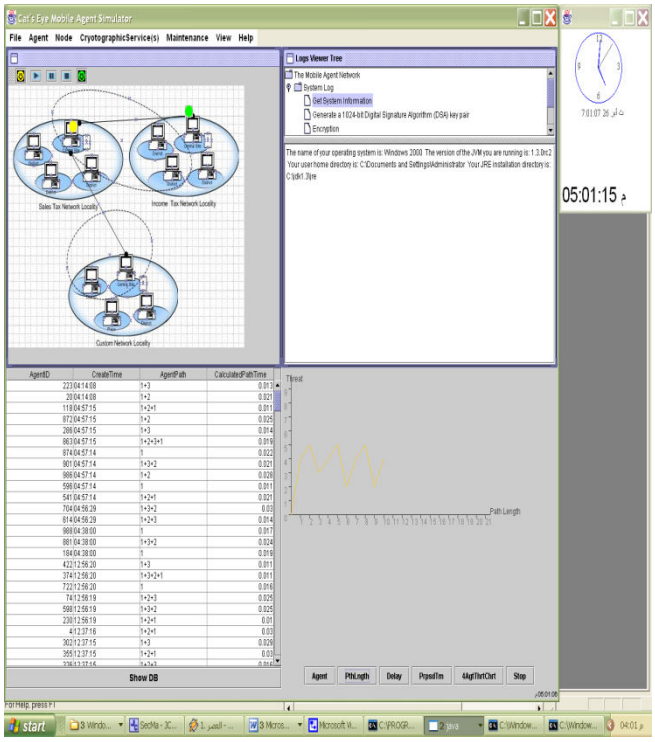

Fig. 3 Deployment of Agent though the Network

Cat's eye within simulation environment supposed to work within the network shown in the above diagram for three domains of machines and entries (Sales Tax, Income Tax, and Custom). Cat's Eye Simulator supposed to be placed at sales tax main entrance machine (Central Site and more than 130+ district/region/office).

According to the deployment scenario of the agent is generated at sales node then migrate to take it's pass randomly depends upon the node confidence ratio after that the agent may face an attack agent that it may divert it's pass again.

Volume: 10 Issue: 06 Pages: 4070-4075 (2019) ISSN: 0975-0290

Symbol	Meaning	Dimension
Bd	The network bandwidth	Bytes/Sec.
T_R	The total time elapsed in a single hop.	Seconds.
Ag	Agent (Code + Data + State)	Bytes.
N	Number of Nodes	Integer no.
Ntr	Number of trips	Integer no.
T(Ag)	The time needed to migrate an agent between nodes	Seconds.
Ct	Creation Time	Seconds.
Mt	Migrate Time	Seconds.
Rt	Return Time	Seconds.
Ti	Real agent route time	Seconds.
Lt	Latency	Seconds.
Ri	Pre- Calc. Agent Route time	Seconds.
No	Network Occupation	Bytes.
tk	Assigned Task to Agent	Integer no.

Table 1 Used Notation(s)

The Constant with asterisk at the beginning has been taken from a similar working environment: ((Local Network))

The Average of Create Time within our research: 0.3 sec
* Minimal Agent Migration on the Local Network
: 35ms. ((0.035 Sec.))
* Latency on the Local Network
: 0.6 ms ((0.0006 Sec.))
* Bandwidth on the Local Network: 6.736 Mbits/s
Network Occupation : 6 Bytes

$T_{i=}$ Rt - Ct

T_{Ri} = Ti + Dt

Delay Time (Dt) = Mt – Ct + Network Delay +Network occupation(constant)/Bd

Ri =

Ct + T(Ag) * Ntr + Lt + Network occupation (constant)/Bd

Ri =

0.3sec +(0.035* 4) + 0.6ms/1000+ 6/6.736Mbits/s =3.3Seconds.

0.3 + 0.14 + 0.0006 + 0.8907 = 1.331Seconds.

Calculated Path = 0.022Minutes.

All the generated agent(s) and its calculated path(s) are within 0.022 Minutes. This time considered to be an appropriate time after applying Cat's Eye Mobile Agent t paradigm techniques compared to similar Mobile Agents Simulators.

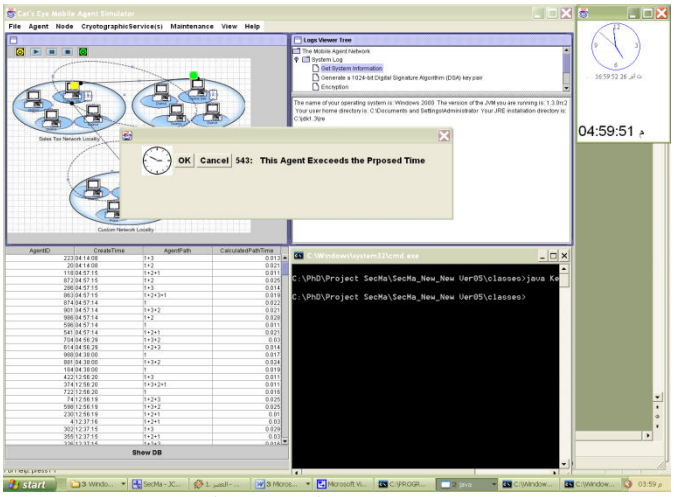

Fig. 4 Tracing Agents

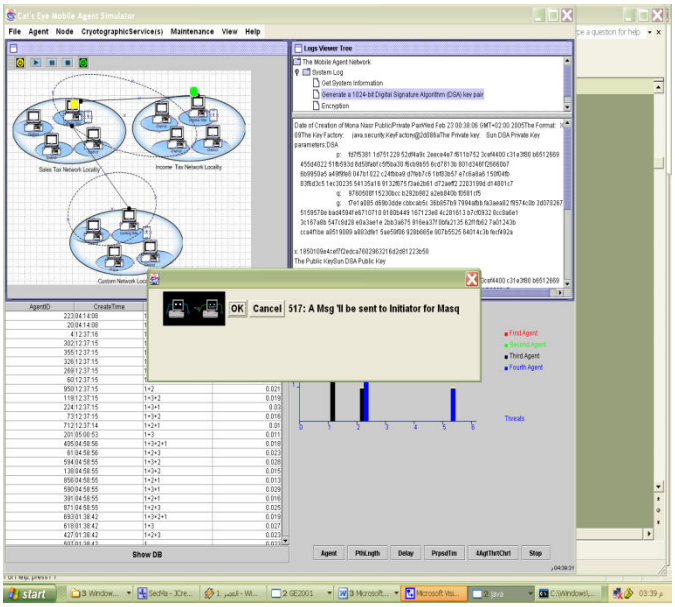

Figure 5 Threat Security Response

Cat's Eye Mobile Agent Simulator as shown in Fig(s) (4,5) an evaluation analysis for detecting tolerance differences for the calculated agent's route before and during its journey, storing agent transactions, storing snapshots of agent state information, checking from time to time agent status and task completeness.

Cat's Eye Mobile Agent can block messages from unauthorized agents but even this task requires some processing by the agent or its communication proxy for preventing undue burden on the message handling routine of the receipt agent.

Cat's Eye Mobile Agent can prevent receiving false or useless information that can lead to delay in completing task in timely manner, by designing a trustworthy tree for nodes, where agent can travel too more confidence node.

Cat's Eye Mobile Agent through its graphical interface and pre-calculation of the agent rout(path) can prevent the

Volume: 10 Issue: 06 Pages: 4070-4075 (2019) ISSN: 0975-0290

delay that may happen during agent journey and checking for conversation policies that the agent doesn't engage in an infinite conversation loop.

VI. WORKING ENVIRONMENT

Since agents are mainly intended to be used for applications over large-scale network.[10,11] Here mentioned below the simulation environment as a step towards large scale environment.

Simulation Environment

In our simulation model we supposed that there are three main entrance nodes to the nework. Sales Tax centeral site, Income central site, and custom cetral site.

Large-Scale Environmet

In large scale these three-entrance nodes are within domain each of the main entrance node relate to number of machines. For example, the sales tax central site is connected more than 130+ district /regional /office, the 245+ to the income and hundred of customs offices but with different number of connections.

On the large-scale environment, the factor of network bandwidth must be taken in consideration, where in our simulation environment we have given it fixed number.

Service	Meaning
Execution	Execute Agent instructions
Transaction	Enables grouping of agent actions into atomic transactions
Authentication	Provides mechanisms
Cryptographic	Provides mechanisms for secure authentication & access.
Agents	Number of Agents
DataBase	Provide multipurpose database service
Messaging	Encodes/decodes and sends/receives messages.
CheckPoint	Stores snapshots of agent state information.

Table 2: A Summary table for all the possible services provided by Cat's Eye Mobile Agent Simulation

VII CONCLUSION & PERSPECTIVE WORK

CONCLUSION

Knowing that mobile agent is a new technology worldwide, which attracts more attention, and as mobile agent importance increase day after day, security of mobile agent became a must. Mobile agent goes through many applications through the web like auction rooms, job finder [2], which attract the attention to all security topics related to mobile agent systems. Also, mobile agent technology may solve problems of many enterprises. By studying the tools, languages used for issuing software mobile agent and the approaches for securing agent journey we found the following:

Many approaches deal with the agent protection against host only, other deals with protection against the agent only.

Many security issues for data and information are constrained and don't provide efficient solution for the complex problems.

Many approaches not realistic in practical and operational use.

Many approaches deal with agent security without using the agent as a tool for applying the security.

Applying mobile agent technology has many benefits for example the mobile agent can apply security and help in reduction of network bandwidth.

The proposed paradigm combines the benefits of the black box approach with trustworthy nodes structure approach.

The proposed solution covers the security of mobile agent from various sides (against the visited hosts and against other agents).

Ease and scalability of security administration of the proposed solution.

The proposed paradigm directs the lights to the survivability of agent and the believability where the available information in mobile agent system where survivability appears to be better. [1]

In designing the proposed security mechanism, the security architectures should not ask whether mobile agents are a component of the architecture, but rather what type of agents the architecture uses so we use service agent for un-necessary information and police (guard) agents for monitoring.

The proposed solution provides the ease of integration with existing applications/products.

The proposed solution provides reduction of network bandwidth, as agent holds on its body versioning of states inform the owner at the end of its journey. We place journey details on agent.

This solution is greatly feasible for operation. As this solution present volatile store for agents so that an agent can restart after a machine failure.

PERSPECTIVE WORK

We suggest in addition to the presented functions in this paper the following items to be added on future work on real environment.

Appropriate servlet classes: en-order to ensure the availability of access the application via web. In this case a threat is appeared, so we must ensure about applet downloaded from un-trusted server or applet from trusted server but contains bugs.

Volume: 10 Issue: 06 Pages: 4070-4075 (2019) ISSN: 0975-0290

Trace mobile transaction: Enhance cat's eye simulator adding:

-Network sensing-tools.
-Resource Manager …with graphical user Interface.
-Guard the access for Screen, Network & Disk.
-Decide which actions an agent can perform based on the Authenticated identity of the agent's owner.

Debugger (for Keep tracking of an agent as it moves through the network, monitors its communication with other agents, and provides traditional debugger features such as breakpoints, watch conditions and line-at-time execution).

Docking system that allows an agent to transparently migrate to or from a mobile computer, even if the mobile computer is not currently connected to the network.

REFERENCES

1. Ahila, S. Sobitha, and K. L. Shunmuganathan. "Overview of mobile agent security issues—Solutions." In Information Communication and Embedded Systems (ICICES), International Conference on, pp. 1-6. IEEE, 2014.
2. Aloui, Imene, Okba Kazar, Laid Kahloul, and Sylvie Servigne. "A new Itinerary planning approach among multiple mobile agents in wireless sensor networks (WSN) to reduce energy consumption." International Journal of Communication Networks and Information Security Vol.7, no.2, 2015.
3. Bagga, Pallavi, and Rahul Hans. "Applications of mobile agents in healthcare domain: a literature survey." International Journal of Grid Distribution Computing Vol.8, no. 5, pp.55-72, 2015.
4. Bhaskar, B., Kumar T. Jagadish., Kamal, M.V., "A Security Determination-Reaction Architecture for Heterogeneous Distributed Network.", IJSRCSE, Vol.5, Issue.5, pp.22-34, 2017.
5. Calvaresi, D., et al., Multi-agent systems and blockchain: Results from a systematic literature review, Conference: Swiss eHealth Summit 2018.
6. Chowhan, R., Mobile Agent Programming Paradigm and its Application Scenarios, International Journal of Current Microbiology and Applied Sciences, Volume 7 Number 05, 2018.
7. Chowhan, R., Dayya, P., Itinerary and Mobile Code Patterns for Emerging Mobile Agent Systems in Large Scale Distributed Environments, International Journal of Computer Sciences and Engineering, Volume 6, Issue 5, May 2018.
8. Dinh, Hoang T., Chonho Lee, Dusit Niyato, and Ping Wang. "A survey of mobile cloud computing: architecture, applications, and approaches." Wireless communications and mobile computing Vol.13, no.18 pp.1587-1611, 2013.
9. Dounya, K., et al., A new approach based mobile agent system for ensuring secure big data transmission and storage, 2017 International Conference on Mathematics and Information Technology (ICMIT).
10. Gavalas, Damianos, Ioannis E. Venetis, Charalampos Konstantopoulos, and Grammati Pantziou. "Mobile agent itinerary planning for WSN data fusion: considering multiple sinks and heterogeneous networks" International Journal of Communication Systems Vol.30, no.8, 2017.
11. Hasan, Ragib, Md Mahmud Hossain, and Rasib Khan. "Aura: An iot based cloud infrastructure for localized mobile computation outsourcing." In Mobile Cloud Computing, Services, and Engineering (MobileCloud), 3rd IEEE International Conference on, pp. 183-188. IEEE, 2015.
12. Karim, M., Security for Mobile Agents and Platforms: Securing the Code and Protecting its Integrity, Journal of Information Technology & Software Engineering, Volume 8, Issue 1, 2018.
13. Kilari, N., Sridaran, R., An Overview of DDoS Attacks in Cloud Environment, International Journal of Advanced Networking and Applications IJANA, Special Issue pp: 124-126.
14. Miller, Naomi Liora, Harold Roy Miller, and Warren Stableford. "Translation of user requests into itinerary solutions." U.S. Patent 9,659,099, issued May 23, 2017.
15. Rahul Singh Chowhan and Rajesh Purohit, "Study of mobile agent server architectures for homogeneous and heterogeneous distributed systems." International Journal of Computer Applications Vol.156, No. 4, pp.32-37, 2016.
16. Rahul Singh Chowhan, Amit Mishra, and Ajay Mathur, "Aglet and kerrighed as a tool for load balancing and scheduling in distributed environment." In Recent Advances and Innovations in Engineering (ICRAIE), International Conference on, pp.1-6. IEEE, 2016.
17. Shermin, V.: Disrupting governance with blockchains and smart. Strategic Change Vol 26, Issue 5, 2017. pp499–509.
18. Shiraz, Muhammad, Abdullah Gani, Rashid Hafeez Khokhar, and Rajkumar Buyya. "A review on distributed application processing frameworks in smart mobile devices for mobile cloud computing." IEEE Communications Surveys & Tutorials Vol.15, no. 3, pp.1294-1313, 2013.
19. Singla, Annie., Jain, Kamal., Gairola, Ajay., "Delving into Security of Networks-Time's Need" IJSRNSC, Vol.2, Issue.3, pp.1-8, 2014.
20. Tapscott, D., Tapscott, A.: Blockchain Revolution: How the technology behind Bitcoin is changing money, business, and the world. Penguin, 2016.
21. Singla, Annie., Jain, Kamal., Gairola, Ajay., "Delving into Security of Networks-Time's Need" IJSRNSC, Vol.2, Issue.3, pp.1-8, 2014.