JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

A Brief Review on Mobile-Agent Security in Peer-to-Peer Networking

Km. Sapna
Department of Computer Science & Engineering
JPIET, Meerut

Ayan Rajput
Department of Computer Science and Engineering
JPIET, Meerut

Abstract- Mobile agents are a popular mobile computing technique that can be found in a wide range of information technology applications. Computer programs and data that can move from peer-to-peer computers in a network to interact with other agents and resources are known as mobile agents. They possess intelligence, autonomy, mobility, and communicative abilities. Mobile agents not only move data, but they also move code from one host to another, allowing the execution to continue on a different network site. Security risks such as agent to platform attacks, agent to agent attacks, agent to other agent attacks, and other security challenges arise as a result of the continuous use of mobile agents in dispersed computer networks. Many researchers have offered solutions to these problems in order to ensure data security, integrity, confidentiality, and availability in mobile agents when data is transmitted between hosts in a computer network. As a result, we provided a complete analysis of mobile agents in this work, as well as a summary of several current experiments.

***Keywords*-** *mobile agents, security, authentication, network*

I. INTRODUCTION

Mobile agents [1] are self-contained, intelligent computer programs that can transport data, mobile code, and their present state of execution from one system to another, so that the code can be run on another computer using distributed computing technology [2] after migrating from a previous machine on a network. In cryptography applications such as e-commerce, robotics, traffic control, and data-intensive systems, mobile agents are essential. When transmitting data and code from the agent to the platform, mobile agents, on the other hand, have security difficulties [3]. There are a variety of strategies and models that can be used to protect data from damaging network attacks. Before we get into the security challenges, let's go over the different sorts of mobile agents. There are two types [4] of mobile agents in mobile computing.

- Static Mobile Agents - Because they already have a host-to-destination path defined, mobile agents with predetermined paths are often referred to as static migration agents.

- Roaming mobile agents - These agents operate without being aware of a specific path, hence they are unaware of the migration path. It must be determined in light of the network's current state. As a result, these agents are referred to as roamer or mobile agents on an ad hoc basis.

II. FEATURES OF MOBILE AGENT

The categorization and definition of the term "agents" was driven by the invasion of numerous approaches under the banner of "agents." However, previous contacts with the AI community and the vague concept of intelligence revealed that everyone had their own meaning of intelligence. Many other definitions for agents have been presented, but most of them include a set of distinguishing features that each agent must possess. The weak agents [5] recommended by should be autonomous, reactive, and sociable, for example. Autonomous, reactive, communicative, adaptable, mobile, flexible, goal-oriented, continuous, and possessing a personality or emotion are all characteristics of mobile agents.

• Autonomous - While human involvement may be required on occasion, an agent should be able to function without it. Mobile Agents are self-contained units. It means that internal events, as well as external actions taken by users or the system, shape the agents' performance and behavior. While selecting a node, mobile agents can make autonomous decisions.

• Mobility - Mobile agents are those who travel from one location to another. The agent isn't tied to any particular node. While conducting activities, they can move from one node to the next. Load processing and balancing are distributed across the network with this function. This functionality also has the advantage of allowing the agents to continue working even if the user is not connected to the internet.

• Intelligence - Agents capable of reasoning, learning, and adapting throughout time. Within their domain, Mobile Agents can learn and seek for information. Because they have some domain competence, intelligent agents are referred regarded be such. They can also move from one setting to another without losing their previous data and function effectively in the new one.

• Adaptive - Adaptive agents can change their behavior over time in response to internal knowledge or environmental changes.

• Communicative - An agent must be able to communicate effectively with other agents. The Knowledge Query and Manipulation Language is the most extensively utilized protocol for agent communication (KQML). Mobile Agents are capable of effectively communicating with other agents, users, and systems. For inter-agent communication, the mobile agents use a communication language.

• Responsive - An agent must be capable of detecting and reacting to changes in its environment.

• Goal-oriented - These agents have a well-defined internal strategy for achieving a specific goal or set of goals.

• Persistent - Persistent agents maintain a constant internal state over time.

• Emotion - Agents capable of displaying emotion or mood in a human-like manner. Anthropomorphic personalities or appearances are possible for this type of agent.

• Integrity - Agents who are certain that the information they pass on is correct.

III. SECURITY THREATS

Mobile agents are a rapidly evolving technology in distributed computer systems that offer a variety of benefits such as autonomy, fault tolerance, and so on. However, there are various security dangers when data and code are transferred from one computer to another. Some instances of security threats [6] are listed below:

1. Agent - to - Platform Attacks

It is a series of threats in which agents attack an agent platform by exploiting security flaws in it. Impersonation, denial of service, and unlawful access are only a few of the threats.

- Denial of service - By utilizing a quantity of the computing resources of agent's platform, mobile agents can perform this type of attacks. They can be carried out on purpose, with attack scripts exploiting system faults, or unintentionally, with programming errors. A rogue agent could be carrying harmful code with the intent of disrupting the agent platform's services, degrading its performance, or stealing information it doesn't have access to.

- Unauthorized access – An agent having uncontrolled access to a platform and its services may jeopardize the platform and other agents. Agents representing different users and organizations are not allowed to read or write data over which they do not have permission, such as residual data stored in a cache or other temporary storage. The platform or the mobile agent must first validate its identity before it can be instantiated on the platform and employ the requisite access control mechanisms.

2. Agent - to - Agent Attacks

• Repudiation: Repudiation [7] occurs when a party to a transaction or communication later alleges that communication activity has never took place. If suitable countermeasures are not in place, repudiation can lead to severe arguments that are difficult to overcome. While an agent platform cannot prevent a transaction from being cancelled, it can ensure that sufficient evidence exists to support the resolution of objections. Although fake documents are rare, they are regularly misplaced, manufactured without authorization, or modified without adequate review. Because a transaction may be cancelled due to a misunderstanding, the agents and platforms engaged in the transaction must preserve records to aid in the resolution of any disputes.

•Masquerade[8]: Agent-to-agent communication is possible between two agents. An agent may hide its true identity to deceive the actual party. Masquerading as another agent is dangerous for both the misled agent and the agent whose identity has been assumed, especially in agent societies where reputation is prized and used to build trust.

• Unauthorized access – By performing its public methods (e.g., buffer overflow, reset to beginning state, etc.) or by using and manipulating the data or code of another agent, an agent might directly interfere with it. Modification of an agent's code has the potential to significantly alter the agent's behavior (for example, making a trusted agent hostile. An agent can learn about the activity of other agents by using platform services to snoop on their discussions.

3. Platform – to – Agent Attacks

These threats encompass all threats involving untrustworthy platforms to which an agent is delivered. Impersonation, denial of service, eavesdropping, and alteration are all risks.

- Alteration - This hazard emerges when the integrity of an agent's data on the new agent platform to which it is transferred is compromised. The contents of the information carried by the agent can be changed by an untrusted platform. It is currently unable to detect malicious changes to an agent's state during execution or the data an agent produces when contacting the compromised platform. For example, without the agent's awareness, the agent platform could be running a customized virtual machine, resulting in erroneous results.

- Masquerade - To fool a mobile agent as to its true destination, one agent platform can impersonate another. By appearing as a trustworthy third party, unsuspecting agents could be attracted to an agent platform, allowing the platform to obtain important data from them. Because of the messages they exchange and the actions they do as a result of these communications, an agent mimicking another agent can only harm other agents.

IV. LITERARURE REVIEW

In recent years, there has been a lot of research into the security of mobile agents, and the process is still ongoing. [9] proposed a three-layer security model for mobile agents that safeguards the agent's path, code, and itinerary while preserving the agent's autonomy and flexibility. Although there is a pre-determined itinerary list of nodes in this architecture, migration of agents to the next node is decided at execution time to provide for flexibility. They assume a trusted key server S in our case. Each host platform must first register with a trusted key server and obtain an identity certificate before participating in an agent-based transaction. Private and public keys should be separate for each hosting platform. In a nutshell, the PKI infrastructure is considered to be trustworthy. The names and descriptions of the keys used in the model are listed in Table 1. Figure 1 depicts the division of itinerary nodes into registered and non-registered nodes. The concept employs a number of techniques to ensure the security of the mobile agent.

Table 1. Names and Abbreviations Used

Name	Description
M	Mobile Agent
K	Random number (act as a secret key)
Pub $_{A}$	Public key of host A (agent's owner or home platform)
Pub $_{N}$	Public key of host N
Pri $_{N}$	Private key of host N
T1	Task Agent
T2	Clone creator Agent
R $_{N}$	Result generated at host N
Pri $_{pubA}$	Secret key of agent's owner which is made public but in secure manner
S $_{N}$	Encrypted result generated at host N
S $_{M}$	Encrypted Mobile agent

[10] proposed a multi-layer mobile agent-based framework for performing various cloud operations. This approach assists both parties in building trust and confidence in one another so that cloud services can be used efficiently. The suggested multi-layer security system has four tiers which are presented below:

1. **Customer to Cloud Service Provider Layer (CSP):** It's the first layer that establishes the relationship between the client and service provider. Multiple assignments are agreed upon by clients and service providers, so communication is two-way. In Fig. 1, Client's relationship with Cloud Service Provider is displayed.

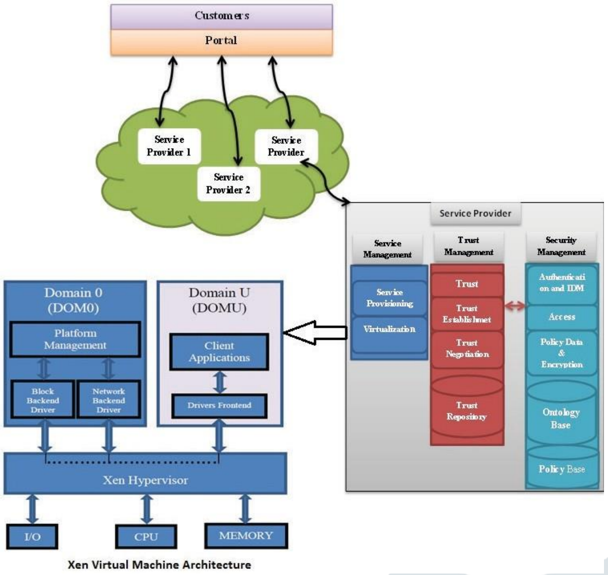

Figure 1. Customer to Cloud Service Layer

2. **Client Authentication Layer:** For ensuring proper communication and access to essential services, the client and service provider must authenticate each other to achieve security when conducting various services with mobile agents. As seen in Figure 3, authentication is achieved by establishing a trusted connection between two entities using the SSL key exchange protocol. Fig. 2 display the exchange process.

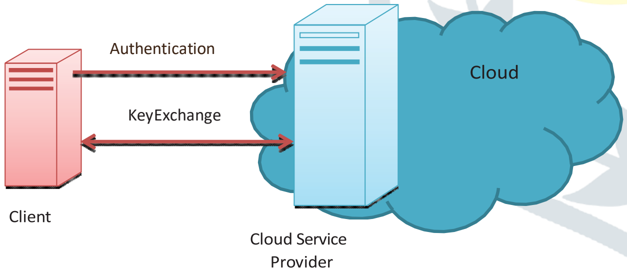

Figure 2. Key Exchange Process between Client and Cloud Server

3. **Mobile Agent Integrity and Authenticity Verification Layer:** Mobile Agent 1 (MA1) will be generated on the client and delivered to the CSP site when the client and CSP have finished the authentication procedure. The servers deployed on both organizations must validate the legitimacy and integrity of Mobile Agent, as shown in Fig. 3. (MA1). MA1 is initiated, and the client is given a new session key. This key, which is used for secure communication and to mask data from CSP, is kept hidden from CSP. This approach assures the client that the mobile agent and the tasks performed by MA1 are secure, without sharing any sensitive data with CSP.

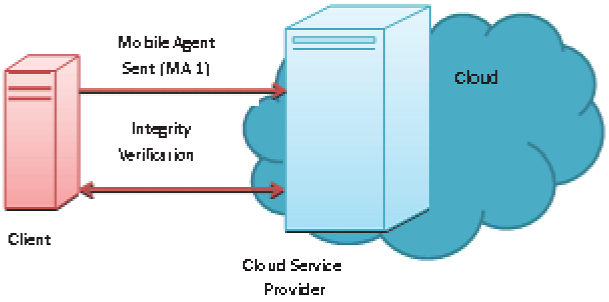

Figure 3. Mobile Agent Verification

4. **Resource Allocation Layer:** Based on the client's needs and the workload of virtual machines controlled by the task manager, MA1 requests resources from the CSP on behalf of the client. MA1 also monitors resource usage and verifies the CPS for erroneous service requests. Based on the request, CSP assigns virtual machines and other resources. If necessary, a new mobile agent MA2 will be developed and deployed on a platform with additional resources to do the scheduled duties more efficiently and effectively. In the same way, many mobile agents can be designed and installed in distinct VMs assigned to the client.

[11] presented an intelligent mobile agent-based traffic control system (MITS). It employs a video camera to detect excessive traffic, and when the number of cars on a certain path reaches a predetermined threshold, it sends an alarm signal to the MITS' smart traffic control module, signaling traffic congestion. A signal transmission and communication module are used to send control packets. The proposed system's operation is depicted in Figure 5.

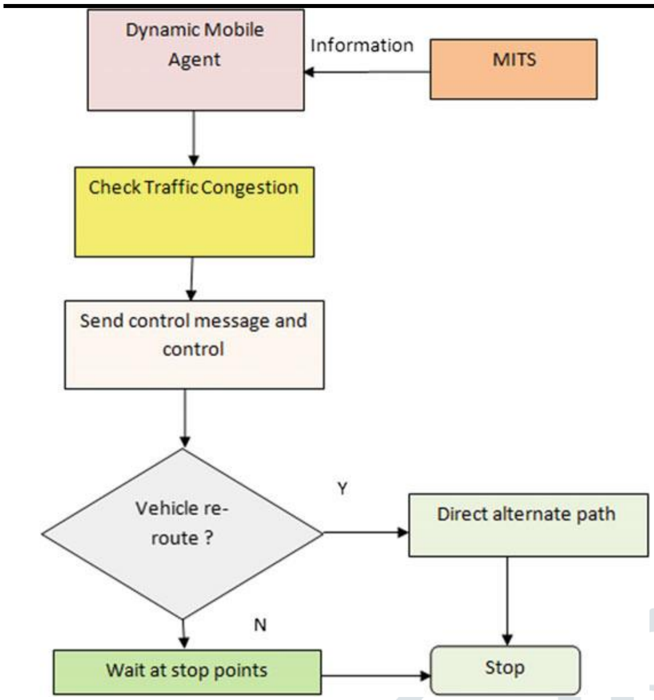

Figure 4. Flowchart of the Proposed Work

[12] presented a security standard for mobile agents to protect them from various types of security attacks. It presents a Petri net-based formal description of our protocol to reinforce the belief of distributed applications in order to demonstrate the correctness of our technique in resisting various types of security attacks. Furthermore, we implement our method on the JADE (Java agent development framework) [14] platform and compare it to a well-known self-protection methodology in terms of computational complexity and services provided.

[13] described a two-part strategy for improving the security of mobile agents: The first describes our contribution to ensuring stronger authentication through the use of a fixed and upgraded Diffie-Hellman key exchange. The second explains how we use an enhanced Discretionary Access Control model (DAC) in conjunction with a Shamir-threshold sharing mechanism to regulate access to platform resources.

[14] have suggested in this work an overview of security difficulties, classifications of threats related with mobile agent technology, and then solutions to security challenges using assessment criteria. The fundamental goal of host protection is to reduce the total power of the execution environment while also reducing a host's overall vulnerability to harmful mobile agents. In the paper, the following strategies are discussed.

1) Using a sandbox (Software-Based Fault Isolation)

Untrusted programs written in an unsafe language like C can be safely executed within an application's single virtual address space using this approach. In untrusted machine interpretable code modules, all memory accesses are limited to code and data segments within the fault domain. A domain-specific identity can also be used to restrict access to system resources. When modules communicate frequently between failure zones, the sandboxing strategy is far more efficient than deploying hardware page tables to keep module address spaces distinct. It's especially useful in situations when the majority of the code is confined within a single trusted domain, because modules in trusted domains don't have any execution cost.

2) Interpretation of Safe Codes

An interpreted script or programming language is typically used to create agent systems. The primary motivation is to enable agent platforms to run on a wide range of computer systems. Safe code interpretation entails making potentially risky directives safe or denying them to an agent. The command to run any string of data as a program segment, for example, is an excellent candidate for denial. Computational entities can be distributed, installed, and executed on remote servers using mobile code systems. Among the programming languages used to create such systems are Java, Telescript, Obliq, Tcl, Scheme, ML, and Python.

3) Code that has been signed

Signing code or other items with a digital signature is a basic way for protecting an agent system. A digital signature verifies the validity, provenance, and integrity of an object. The code signer is usually the person who signs the code.

[15] summarized the security challenges raised by the mobile agent concept. He presents a hybrid encryption approach that incorporates HES and Function Composition techniques (FnC). Mobile Agent Encryption (MAE), an encryption programme that encrypts the operands of the three-address code with HES and the codes with FnC, will intercept the three-address code. MAE will encrypt sensitive data stored in the operands of three address codes, such as credit card numbers and personal information, and scramble the mobile agent's code to fool untrustworthy hosts. The method encrypts executable mobile agents without the need for decryption, and it inherits the majority of the benefits of mobile cryptography.

[16] proposed a Fuzzy-based Mobile Agent migration approach working on Fuzzy Logic System (FLS). It exhibits human intelligence behavior using non-linear input-output scenario for handling unexpected conditions. The FLS is used to calculate the probability of node’s sequences between the two intermediate nodes. The FLS uses three input parameters in this research: node’s remaining energy, distance to the source node and number of node’s neighbors. The performance evaluation is agent’s code, agent’s result, confidentiality and integrity.

[17] presented a reliable approach for migrating mobile agent from the search engine side to the web server side in the form of web crawling agent. The main focus of the research is to reduce the network load on the server side. The performance evaluation is done terms of data integrity, confidentiality and network load consumption.

[18] introduced the concept of mobile agent in Intelligent Transportation System (ITS) by securing the authentication part of mobile agent. The authentication process is done with the help of notification prosecco of the system. The

performance of the system is validated by assessing authentication part of the mobile agent code.

[19] proposed two security mechanisms to enhance mobile agent security. On one hand, they used a cryptographic trace to ensure mobile agent integrity and origin authentication, and on the other hand an SOS agent monitor's model is proposed to protect the mobile agent system against malicious hosts and DOS attacks.

[7] presented an efficient encryption protocol is developed to restrict the execution of malicious code in mobile agents. The integrated bloom filter (IBF) is proposed in this work to filter the agent code and break the code into byte array blocks and generates the initial keys. For an efficient encryption and decryption, a robust and flexible key is needed, the elliptic curve is defined in this paper to generate the private and public key for the encryption process. The Cipher text Policy Attribute-Based Encryption (CPABE) is a public key encryption method deduced to encrypt the information in the form of cipher text associated with attributes. The implementation of proposed work is executed in the JADE platform and results are evaluated and compared with existing protocols such as Fragmentation and AES.

[20] describes the device fingerprint extraction, registration and authentication algorithm. As the traffic on the internet is growing at a rapid rate, there is also the need of a technique to reduce the traffic on the internet. Comparison analysis of the proposed algorithm and comparison with the traditional client server based mechanism is done in terms of network traffic. The proposed algorithm reduces the traffic around the authenticator to a great extent as compared to the client server based mechanism.

[21] presented a Genetic based approach for solving the path problems in the mobile agent's itinerary. The research proposed a novel crossover operator that avoids premature convergence and offers feasible paths with better fitness value than its parents. The performance is validated in terms of integrity and securing agent's itinerary.

[22] 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. The performance of the system is accessed in terms of integrity and confidentiality.

[23] proposed a novel framework for protecting the mobile agent from cross layer attacks in wireless sensor networks, thus maintaining the integrity and security of mobile agent code. Table 2. shows the comparison of different security services provided by different literature survey.

Table 2. Comparison of literature review in providing different security services (Y-Yes, N-No)

References	Category	Security Objective			Security Services		
		Agent's code	Agent's Result	Agent's Itinerary	Confidentiality	Integrity	Authentication
[9]	Prevention	Y	N	N	Y(Code confidentiality	N	N
[10]	Prevention	Y	N	N	Y(Code confidentiality	N	N
[11]	Detection	Y	Y	Y	N	Y	N
[12]	Detection	N	Y	N	Y(Result)	Y (Result)	Y (Result)
[13]	Prevention	Y	Y	Y	Y(all)	Implicit integrity	Y
[14]	Prevention	N	Y	Y	Y(Result)	N	Y
[15]	Prevention	Y	Y	Y	Y	Y	Y
[16]	Prevention	N	N	Y	Y(Code confidentiality)	N	Y
[17]	Prevention	N	Y	N	N	Y	Y

[18]	Prevention	Y	N	Y	Y	Y	N
[19]	Prevention	N	Y	N	Y	N	Y
[20]	Prevention	Y	Y	Y	N	Y	Y
[21]	Detection	N	Y	N	N	Y	N
[22]	Prevention	N	Y	N	N	Y	Y
[23]	Prevention	Y	N	Y	Y	Y	Y

V. CONCLUSION AND FUTURE SCOPE

This paper discusses many types of mobile agent attacks. The methods that can be used to counteract each form of attack are also discussed. Not all techniques are successful against all types of attacks, it has been discovered. When dealing with specific attacks, certain techniques are more effective. As a result, the technique to use can be selected by the type of application to be developed. As a result, it's feasible to deduce that the security method used is determined by the type and design of the programs.

References

[1] P. Ahuja and V. Sharma, “A Review on Mobile Agent Security,” *Int. J. Recent Technol. Eng.*, no. 2, pp. 2277–3878, 2012.

[2] P. Mittal and M. K. Mishra, *Towards Extensible and Adaptable Methods in Computing*. Springer Singapore, 2018.

[3] G. Vigna, “Protecting mobile agents through tracing,” *Proc. 3rd ECOOP Work. Mob. …*, 1997, [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.41.2081&rep=rep1&type=pdf.

[4] D. Deenadayalan, A. Kangaiammal, and B. K. Poornima, *Integrated Intelligent Computing, Communication and Security*, vol. 771. Springer Singapore, 2019.

[5] J. Mir and J. Borrell, “Protecting mobile agent itineraries,” *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 2881, pp. 275–285, 2003, doi: 10.1007/978-3-540-39646-8_26.

[6] M. Alfalayleh and L. Brankovic, “An overview of security issues and techniques in mobile agents,” *IFIP Adv. Inf. Commun. Technol.*, vol. 175, pp. 59–78, 2005, doi: 10.1007/0-387-24486-7_5.

[7] P. K. Jolly and S. Batra, “Security against Attacks and Malicious Code Execution in Mobile Agent Using IBF-CPABE Protocol,” *Wirel. Pers. Commun.*, vol. 107, no. 2, pp. 1155–1169, 2019, doi: 10.1007/s11277-019-06329-7.

[8] F. Hohl, “Time Limited Blackbox Security: Protecting Mobile Agents From Malicious Hosts,” pp. 92–113, 1998, doi: 10.1007/3-540-68671-1_6.

[9] S. Srivastava and G. C. Nandi, “Protection of mobile agent and its itinerary from malicious host,” *2011 2nd Int. Conf. Comput. Commun. Technol. ICCCT-2011*, pp. 405–411, 2011, doi: 10.1109/ICCCT.2011.6075189.

[10] M. Uddin, J. Memon, R. Alsaqour, A. Shah, and M. Z. A. Rozan, “Mobile Agent based Multi-layer Security Framework for Cloud Data Centers,” *Indian J. Sci. Technol.*, vol. 8, no. 12, 2015, doi: 10.17485/ijst/2015/v8i12/52923.

[11] M. Rath and B. K. Pattanayak, “Security protocol with IDS framework using mobile agent in robotic MANET,” *Int. J. Inf. Secur. Priv.*, vol. 13, no. 1, pp. 46–58, 2019, doi: 10.4018/IJISP.2019010104.

[12] S. Srivastava and G. C. Nandi, “Self-reliant mobile code: A new direction of agent security,” *J. Netw. Comput. Appl.*, vol. 37, no. 1, pp. 62–75, 2014, doi: 10.1016/j.jnca.2013.01.004.

[13] H. Idrissi, E. M. Souidi, and A. Revel, “Security of mobile agent platforms using access control and cryptography,” *Smart Innov. Syst. Technol.*, vol. 38, pp. 27–39, 2015, doi: 10.1007/978-3-319-19728-9_3.

[14] S. S. Ahila, “Overview of Mobile Agent Security,” no. 978, 2014.

[15] M. Rezaul Karim, “Security for Mobile Agents and Platforms: Securing the Code and Protecting its Integrity,” *J. Inf. Technol. Softw. Eng.*, vol. 08, no. 01, 2017, doi: 10.4172/2165-7866.1000220.

[16] H. Q. Qadori, Z. A. Zukarnain, Z. M. Hanapi, and S. Subramaniam, “FuMAM: Fuzzy-Based Mobile Agent Migration Approach for Data Gathering in Wireless Sensor Networks,” *IEEE Access*, vol. 6, no. c, pp. 15643–15652, 2018, doi: 10.1109/ACCESS.2018.2814064.

[17] N. Singhal, A. Dixit, R. P. Agarwal, and A. K. Sharma,

"A reliability based approach for securing migrating crawlers," *Int. J. Inf. Technol.*, vol. 10, no. 1, pp. 91–98, 2018, doi: 10.1007/s41870-017-0065-0.

[18] N. Allali, Z. Chaouch, and M. Tamali, "Dashboard of intelligent transportation system (ITS) using mobile agents strategy on notification authentication process," *Int. J. Electr. Comput. Eng.*, vol. 9, no. 1, p. 621, 2019, doi: 10.11591/ijece.v9i1.pp621-628.

[19] S. Alami-Kamouri, N. Moukafih, G. Orhanou, and S. Elhajji, "Mobile agent security based on cryptographic trace and sos agent mechanisms," *J. Commun.*, vol. 15, no. 3, pp. 221–230, 2020, doi: 10.12720/jcm.15.3.221-230.

[20] U. Kumar and S. Gambhir, "Device Fingerprint and Mobile Agent based Authentication Technique in Wireless Networks," *Int. J. Futur. Gener. Commun. Networking*, vol. 11, no. 3, pp. 33–48, 2018, doi: 10.14257/ijfgcn.2018.11.3.04.

[21] C. Lamini, S. Benhlima, and A. Elbekri, "Genetic algorithm based approach for autonomous mobile robot path planning," *Procedia Comput. Sci.*, vol. 127, pp. 180–189, 2018, doi: 10.1016/j.procs.2018.01.113.

[22] M. Nasr, "Self-Protected Mobile Agent Paradigm for DDoS Threats Using Block Chain Technology," *Int. J. Adv. Netw. Appl.*, vol. 10, no. 06, pp. 4070–4075, 2019, doi: 10.35444/ijana.2019.10064.

[23] S. Nithya and C. Gomathy, "Smaclad: Secure Mobile Agent Based Cross Layer Attack Detection and Mitigation in Wireless Network," *Mob. Networks Appl.*, vol. 24, no. 1, pp. 259–270, 2019, doi: 10.1007/s11036-018-1201-1.