"IPLess" Stochastic Anonymous Routing Algorithm Using Multi-Agent Systems

Magdy Saeb[1], Meer Hamza[2], and Ahmed EL-Shikh[3]
(Corresponding author: Magdy Saeb)

Arab Academy for Science, Technology & Maritime Transport[1]
School of Engineering, Computer Engineering Department, Alexandria
Arab Academy for Science, Technology & Maritime Transport[2]
College of Computing and Information Technology, Alexandria
Software Engineering competence Center (SECC), ITIDA, MCIT, EGYPT[3]
(E-mail: mail@magdysaeb.net)

(Received Nov. 11, 2005; revised and accepted Jan. 7, 2006)

Abstract

The ability of communicating with a selected party anonymously is an important aspect of network security. Communicating anonymously means that no adversary can discern who is communicating with whom at which time and how much information is being sent. A great deal of work has been performed to provide such anonymity with different degrees of success. We propose an algorithm for anonymous communication that is based on the notion of the traditional onion routing algorithm. However, the proposed algorithm relies on a stochastic approach and is carried out by a swarm of cooperating mobile agents. We introduce the concepts of anonymous game, linear heterogeneous onion and non-linear heterogeneous onion to provide data manipulation. We provide mathematical models that are based on the algorithm, to estimate the degree of anonymity. Applying this algorithm, and the accompanying models, one is able to measure quantitatively the success probability of message delivery. We show that this message delivery is successful with no receiver address being part of the message. In addition, a detailed security and impact analyses are provided.

Keywords: Anonymous communication, mobile agent, network security, onion routing, stochastic process

1 Introduction

Security is one of the most important aspects in a communications network. There are many security aspects, some are concerning with hiding the content of the transmitted messages, others focus on making certain services available to authorized personnel only. The ability of communicating with a selected party anonymously is an important aspect of network security. Communicating anonymously means that no adversary can discern who is communicating with whom at which time and how much information is being sent. The need of feeling in a private environment and being anonymous is increasing in many day-to-day activities. Anonymity can be essential in some applications related to e-commerce and mcommerce, peer-to-peer secure connectivity for linking distributed business parties, electronic banking solutions, entertainment, e-government, and location-sensitive communications such as GSM and GPRS. Moreover, mobile wireless computing, electronic voting and certainly secure military applications are all fields that require a sufficient level of anonymity for both of the service provider and the end-user. The anonymity attribute of each of these services can affect number of other service's attributes such as confidentiality, return-oninvestment, accuracy, human safety and completeness of critical missions. As an example, in electronic voting systems; feeling anonymous support the feeling of being private and safe, which will facilitate the ability to choose your preferred candidate. The final gain is an accurate voting result. Another imperative example appears in the military application. It is evident that communications subnet that reveals the identities of communicated parties can lead the enemy to be able to destroy one or both of them. In the last example the need of anonymity is not limited by the human feelings or economic aspects, it will completely affect the completeness of the mission. The rest of this paper is organized as follows: Section 2 introduces a study of the existing solutions, adversary types, attack types, design issues, implementation issues and methodologies for calculating the degree of anonymity. Section 3 introduces our proposed algorithm. Section 4 provides a stochastic process analysis. Section 5 presents the security analysis and results. Finally, we provide a summary and our conclusions.

2 Study of Anonymous Communication Systems (ACS)

Research on communication anonymity has provided many algorithms for various fields of applications that require anonymity with varying degrees. These examples can be found in [7, 15, 22, 26, 30, 34]. Surveys on the existing solutions for providing anonymity can be found in references [5, 15, 16, 20, 21, 31].

2.1 A Brief ACS Review

2.1.1 Overview of the Existing Solutions

In order to provide anonymity, the existing ACS systems are based on fundamental ideas that incline towards one of three choices. These are: (1) Exhausting the attacker through a cooperative stream of information traffic as in DC-Net [4], (2) Performing a rerouting behavior through a sequence of intermediate nodes such as Chaum Mixes [3], (3) Providing coverage for the real traffic through bulk of dummy traffic [6]. The targeted service type varies from simply sending an anonymous e-mail [1], to browsing web anonymously [34] or even real time peer-to-peer communications [22]. Some algorithms send data in plain format [1], other encrypts data once [4] and other using nested encryption [11]. ACS shows different trends in casting messages through the systems. Broadcasting the message appears in [4, 9, 28]. The approach adopted by reference [29] uses multi-cast while [12] relies on a unicast mechanism.

2.1.2 Onion Routing as a Base of the Proposed Algorithm

Onion Routing [11, 12] is one of the most prominent algorithms in the "Re-routing" family of ACS. It provides a general anonymous solution for the internet. Onion Routing has two versions "Onion Routing I" and "Onion Routing II". Tor [8] solves some problems in original Onion Routing. Freedom [10] provides a similar Onion Routing from the user point of view, but he enables the selection of certain trusted nodes in the start and the end of rerouting path.

A lot of algorithms were based on Onion Routing either to solve some type of attacks such as Online/Offline Onions [19], UREOnions [13], DUO-Onions [17] and Hydra- Onions [17], or to provide a similar anonymous service for different types of applications such as Ad-hoc Onion Routing [20] that provides the required anonymous communications for mobile agents. Onion Routing has certain enviable features when compared to other solutions. These are; it provides both initiator and receiver anonymity and of course communication anonymity. It is a practical solution, which suffers little from the scalability problem. It does not have a performance degradation problem such as in DC-Net- based family. It approximately supports real-time communications. Finally, it supports most types of communications over the internet. All of the previous features direct us to focus on the onion routing and to try to build an algorithm based on its fundamental nature.

2.1.3 Elimination of Hidden Algorithms

In order to provide the anonymous communication service in the rerouting family; there are always three concealed algorithms that are involved in the operation. They are used to determine the degree of anonymity. These algorithms are: (1) Anonymous Channel/Path Formation, (2) Anonymous Channel/Path Maintenance, and (3) Anonymous Channel/Path Destruction. If we take the onion routing as an example, the channel formation phase is carried out through an onion formation algorithm that constructs the forward and backward path and at the same time distributes the required encryption keys [11, 12]. Path maintenance, to the desired lifetime, occurs through enabling the intermediate routing nodes to store direct previous node address, direct next node address and decryption/ encryption symmetrical key in their own local routing tables. The last algorithm is responsible to destroy the rerouting path in order to limit the attacker's ability to track and identify the true initiator or receiver. Certainly, the three algorithms determine the effective degree of anonymity provided by each ACS. However, when the first algorithm is performed by a single Item-of-Interest (IOI) recognizing all this information, it will develop into a single point of failure. In the second algorithm, such critical data is stored and remains static for the lifetime of the path. They can be collected and analyzed from all of the compromised nodes. The third algorithm provides a tradeoff between the security requirement and the communication overhead [20] as a cost of path formation and destruction activities. The ideal case should not depend on any one of these algorithms. Our proposed approach has been designed to abolish these three hidden algorithms and replace them with a pure stochastic mechanism in a suitable and acceptable overhead level.

2.2 The Provided Anonymity by ACS

2.2.1 Studied Attacker Model

The type of attack depends on the attacker's ability and stipulation. The attacker or the adversary may be: (1) Internal or External, (2) Passive or Active and (3) Static-Adaptive [25]. "Global Passive Adversary" [23] is similar to the "Very Strong Attacker" introduced in [2]. Both are from the external type, they can observe all communication links adaptively and have no message correlation ability. This is the result of the difficulty of recognizing the message form because of the nested message encryption. The global adversary has a time limitation; he cannot observe all communication links for the invited time. On the other hand, [15, 16] stated a more practical adversary model called "Passive Internal Adversary". Little

importance was allocated to its stability nature, i.e., if it is static or adaptive. They also introduced a generalized model of the adversary algorithm that was used to collect data, analyze it probabilistically and identify the initiator. The internal passive type of attacks was shown to be harder to be detected. In this paper, we study the provided degree of anonymity against this type of attacker.

2.2.2 Quantitative Anonymity Study

Different trials were carried out to quantify the degree of the provided anonymity. Mathematical formulae were introduced for the first time in [29] to describe the different levels of anonymity mentioned subjectively in [25]. These references have pointed out that using mathematical model will be the most promising approach to evaluate the anonymity service provided by MIX networks. A useful mathematical model was built for the attacker behavior as in [15, 16]. We are using this model as a basis for our anonymity study, as mentioned before.

2.2.3 Implementation Issues

The first intuitive impression against the real life implementation of the ACSs goes towards the "Time Delay" and "Bandwidth Wasting" as indicators for poor performance. It is a tradeoff between providing suitable degree of anonymity and service cost; in the form of time delay and bandwidth. On the other hand, reference [12] stated that the communication overhead in the onion routing is very small and acceptable. Performance, Reliability and Scalability were mentioned as the basic problems facing the implementation of the Ad-Hoc Onion Routing algorithm [20]. In this paper, mathematical modeling is used to analyze these parameters.

2.2.4 Information Elements

Information elements To limit the ability of the adversary, the solutions in the rerouting family were intended to do one of the following three behaviors: (1) Hide some elements of information from the attacker, (2) Distribute the information elements between several parties in order to make it difficult to correlate them, (3) Formulate them to be varying with time. (4) Finally, eliminate the information constituents from the communication network.

There are three categories of information inside any rerouting ACS: (1) Communication Items Information, (2) Transferred Data Information, and (3) Rerouting Path Information. The first category includes all the information that holds the identity of items involved in the communication such as; Initiator identity, Receiver identity, Initiator proxy identity, Receiver proxy identity, Session identity and Connection identity. The second category contains all information that is related to the transferred message itself. It includes, the "Plain Data" of the communication message, and the "Encryption Keys" for encrypt/decrypt the messages. The third category holds information concerning the path through which the message is rerouted. This includes, Path Selection Information, Path Formation Information, Intermediate Nodes Information, Path Length, Path Type (fixed or variable), Path Topology (Simple or Complex), Path Lifetime, Path Destruction Algorithm and Trigger and Reply Path. The ideal case is to let no one in the ACS know much information as possible. Our approach is based on this idea and it does not permit any one to know in advance this information during the communication session.

Path Lifetime, Path Destruction Algorithm and Path Destruction Triggers affect the degree of the provided anonymity. Path lifetime is a very critical issue. If the rerouting path is fixed for a relatively long time, it will become more vulnerable to the adversary. If it was destructed after a short time, the system will suffer from the large context overhead associated with path formation and deformation activities. System administrator can be forced to destruct all paths or channels at certain time instance or on regular basis to give the systems the required level of immunity against the denial of service attack [35]. Theoretically, if all the transmission activities are completed on an ad-hoc basis, there will be no path construction or destruction phases and hence no path lifetime or triggers. The more information elements that you can abolish from the ACS, the much anonymity degree you can achieve. In our approach, we propose the elimination of all of these activities from the ACS.

2.3 Mobile Agent in ACS

The early start for the concept of employing agents in the ACS was in [18], the main purpose of this work was not to provide a model to the complete ACS as a set of functions done by set of agents. The second valuable appearance was in [20], each node was completely replaced with a mobile agent under a control of some monitoring agents that arrange the whole operation. The main target was the replacement of the old physical routers with mobile agent ones and providing the anonymous service for mobile agent applications. Further investigations employing mobile agents in ACS can be found in [31, 32, 33]. In this paper, mobile agents are employed as routers. Mobile agents collaboratively execute a stochastic anonymous game to provide general anonymous service.

3 The Proposed Approach

In order to demonstrate the contents of our approach, we will go through a detailed explanation of its basic schema. We start by investigating the objectives of this approach. Furthermore, we explain some of the basic definitions and system configuration. Finally, the detailed steps of the algorithm and some other aspects such as the onion formation and path discovery algorithm are described.

3.1 System Configuration

The structure of the proposed anonymous communication systems contains the Physical Network Nodes and the Mobile Agents Swarm. The physical network nodes represent the communication infrastructure, while the swarm of mobile agents will take the responsibility for providing the anonymous communication service for both the initiator and the receiver. System configuration contains "Setup Configuration", which describes the system condition before starting running the system and occurs only once, and "Running Configuration", which describes how agents swarm will act in the running time.

Setup Configuration: A predefined and equal number of agents are generated in each node at this phase. Each agent has a unique identification number. The agent is permitted to move on a hope-by-hope basis. The swarm distribution in the setup phase is assumed a "uniform distribution".

Running Configuration: The System is built from a set of layers collaborating with one another to provide the required anonymity. There will be four layers: (1) Network Physical Nodes layer, (2) Mobile Agents Swarm layer, (3) Anonymous Communication Team layer and (4) Anonymous Communication Game layer.

The "Network Physical Nodes" layer appears as system states in the Carrier Tunnelling-Routing Agents (CTRA) random walk process. CTRAs form the second layer called Mobile Agents Swarm. A Random Walk Stochastic Process that fulfills the Markov criteria is represented by a Markov Transition Matrix. Random walk process maintains the uniform distribution of agents that the system starts with. "Mobile Agent Teams" or "Anonymous Communication Teams" (ACT) layer is a virtual layer that contains number of teams of mobiles agents. The minimum size of any team is two agents. Each agent in the team is permitted to hold and store the identities of each member of its team. Those identities are in the form of a pseudonym. "Anonymous Communication Game" (ACG) layer represents the effective layer, which contains the true messages and padding ones. The padding messages are resulted from the Message Diffusion Tree (MDT).

3.2 The Proposed Algorithm

In this section, we will show the steps that the anonymous communication system should follow in order to deliver the message anonymously from its initiator to its receiver. These steps are as follows:

1) The anonymous communication system is configured and running in its steady state as previously described in the system configuration section.

2) The initiator AA, that desires to send a message anonymously through the ACS, prepares the message as will described later in the data manipulation section- before sending it to a number of its proxy agents (PAs).

3) The initiator AA chooses a number of PAs from its registered PAs and Valid Proxy Agents (VPAs) to send the message to them. The data of the VPAs are derived from the anonymous communication clusters (ACC) data.

4) Each PA/CTRA will resend only if its current lifetime is less than the maximum lifetime of the message, without counting the time for encryption or decryption, to a random member (CTRA) in his ACT. This member was chosen from the entire team, according to a uniform probability distribution. The PA increments the message lifetime counters and stores a copy of the message in its memory to simulate the behavior of the Warehouse Proxy Agent WPA/ CTRA.

5) Each CTRA that receives the message, repeats the activities that had been performed in step 4. If any one of these CTRAs is a WPA of the required receiver, this means that the message has been successfully gone through its entire journey to its receiver except only onestep. This does not mean that the WPA will terminate the retransmission game. Since, if it does terminate, the attacker can easily observe this action and unfold the procedure.

Formally, the algorithm is summarized as follows:
[Given a communication network with a number of nodes equal to "N"; all nodes can be seen as a clique. Each node has an equally generated number of mobile agents (CTRAs) with total number equal to "n" Each application agent "AA" has a registered set of proxy agents "PA". Teams with fixed sizes are generated, which contain number of agents that represent an ammoniums communication cluster of AAs "ACC", each agent store only the identification of his team(s) members].These concepts are explained as follows: The preparation of the "Heterogeneous Onion" will be illustrated in the message format and data manipulation section. Valid proxy Agent "VPA" is a member in a team that represents an ACC containing the desired respondent. According to the polymorphism property; and mobile agent can be CTRA, PA, or WPA.

Inputs: The encrypted message(s) or as we call it the "Heterogeneous Onion", set of anonymous communication clusters "ACCs", set of registered proxy agents "PAs", Game Type as a Boolean, Maximum Message Lifetime (μ) as an integer.

Algorithm Body:

Begin Algorithm
CTRAs start random walk process;

Send Message ()
Begin
Select PAs and VPAs ()
Begin
Take the set of registered proxy agents "PAs" and the set of anonymous communication clusters "ACCs" as an input;

If (Game Type == Single Team)
Select one "VPA";
Else
Select random number of "PAs";
Select random number of "VPAs";
Return Selection;
End;

Game ()
Begin
Send message to selected "PAs" and "VPAs";
For Each "PAs/CTRAs" and "VPAs/CTRAs"
Begin
If ("Current Message Lifetime"$> \mu$)
Destroy Message
Else
Select random team member (CTRA) from its corresponding team using uniform probability distribution;
Increment "Current Message Lifetime";
Store copy of the message;
Game ();
End;
End Game;
End Send Message;

Collect Messages ()
Begin
Each AA collect the stored messages from its registered PAs/WPAs;
Wait for fixed time interval;
Collect Messages ();
End Collect Messages;
End Algorithm

Success Criteria: The encrypted message(s) or "Heterogeneous Onion" reached - at least one time- to WPA(s) for the desired respondent AA(s).

3.3 The Mathematical Model

The random walk process for the CTRAs is performed when each CTRA chooses one network node from the adjacent nodes of the current occupied one. The selection is completed according to a uniform probability distribution with value equal to $1/(a_b)$, where a_b is the number of adjacent nodes to the node, $(a_b >= 1)(b)$. If the CTRA is carrying a message during his walk, we call this action as "Tunnelling". We assume that tunnelling process, performed by the CTRAs before retransmitting the message, is an inverse Bernoulli stochastic process. At each node, CTRA can take the decision to continue tunnelling with a probability equal to (r), or take the decision to retransmit the message with a probability equal to $(1-r)$. We call (r) as the "tunnelling parameter", which will affect the length of tunnel at each game step. Tunnelling behavior can be modelled by the following equation:

$$P(TunnellingLength = \beta) = r(1-r)^{\beta-1}, \qquad (1)$$

where r denotes the probability of retransmission $(0 < r < 1)$ and β denotes the length of tunnel.

If CTRA assumes a random decision to route (retransmit) the message, it will choose randomly, employing a uniform probability distribution, one mobile agent from its team and send the message to it. This decision is completely independent from the true respondent because it does not know it. This situation can be modelled by the following stochastic transition matrix.

$$\begin{bmatrix} 0 & \cdots & \cdots & \frac{1}{r-1} & \frac{1}{r-1} \\ \vdots & 0 & \cdots & \ddots & \frac{1}{r-1} \\ \vdots & \vdots & 0 & \vdots & \vdots \\ \frac{1}{r-1} & \ddots & \cdots & 0 & \vdots \\ \frac{1}{r-1} & \frac{1}{r-1} & \cdots & \cdots & 0 \end{bmatrix}$$

where $T = ACTsize(T \geq 2)$.

Each CTRA will repeat either the routing or the tunnelling behavior until the game ends actually when the message reaches the warehouse proxy agent, (WPA) of the respondent and ends physically when the "Message Lifetime, MLT" reaches its maximum. Similar to the tunnelling process, anonymous communication game is an inverse Bernoulli stochastic process and can be modelled according to the following equation that calculates the probability of the game length presuming the value (α):

$$P(GameLength = \alpha) = p(1-p)^{\alpha-1}, \qquad (2)$$

where p denotes probability of message retransmitted to required CTRA, i.e., WPA of the respondence t and is equal to $1/(T-1)$.

The total number of transactions preformed by the team contains both tunnelling and rerouting activities. We call it the "Effective Game Length" since it represents the total delay time between the initiator and respondent. Since no CTRA is permitted to retransmit the message from the same node that it received it in, as an assumption, effective game length's minimum value, accrues when all game actions are only rerouting without any tunnelling- is equal to (2α). Tunnelling activities can happen on all game plays except the last one. That is when the message reaches the respondent WPA and is equal to$(\alpha - 1)$. Total tunnelling activities is equal to the sum of all tunnels' lengths. Replacing tunnels' lengths with expected tunnel length will provide the expected value of the effective game length, which can be calculated from the following equation:

$$\alpha_{Eff} = 2\alpha + (\alpha - 1)\beta_{Exp}, \qquad (3)$$

where β_{Exp} denotes expected tunnelling length equal to $(1-r)/r$.

For a predetermined message lifetime (μ), the message can or cannot be delivered to the destination WPA. If the message did not reach respondent WPA in all game plays, the game had failed. The probability of game delivery failure in a game with total number of retransmissions equal to (μ) is:

$$P(DeliveryFailure) = (1-p)^{\mu}, \quad (4)$$

where p denotes probability of message retransmitted to required CTRA -i.e. WPA of the respondent-and is equal to $1/(T-1)$; μ denotes message lifetime.

It is evident that the probability of game delivery success in a game with total number of retransmissions equal to (μ) is as follows:

$$P(DeliverySuccess) = 1-(1-p)^{\mu}. \quad (5)$$

To increase the probability of delivery success, a multi-game or a multi-team session is conducted. The probability of game delivery success in a multi-game, utilizing τ teams, session with a total number of retransmissions equal to (μ) is:

$$P(DeliverySuccess) = 1-((1-p)^{\mu})^{\tau}, \quad (6)$$

where τ denotes number of teams in the multi- game session. All of the previous equations provide us with a comprehensible mathematical model for the anonymous communication system and game. In the next section, we will provide the assessment results of those equations.

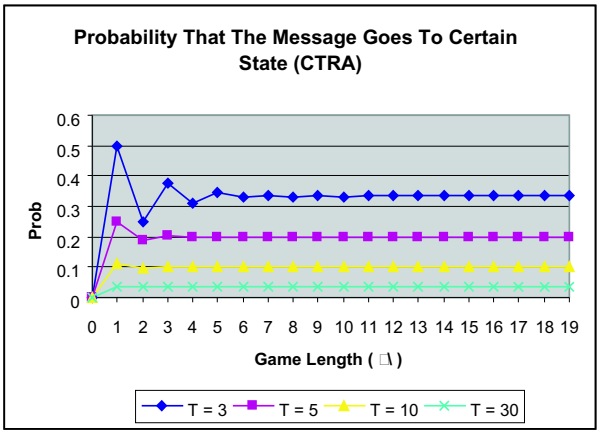

Figure 1: Probability of the message being sent to certain team member vs. game length

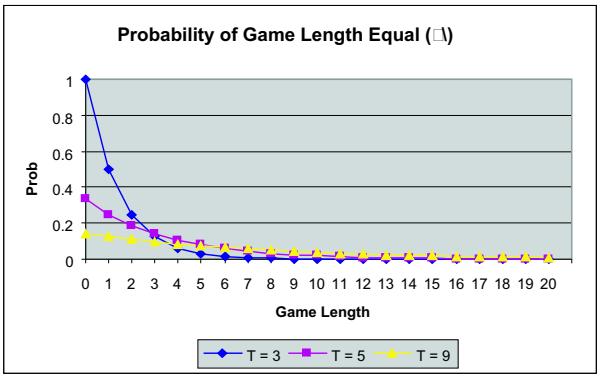

Figure 2: Probabilities of different game lengths for team sizes equal to 3, 5 and 9

4 Stochastic Processes Assessment

The assessment for previously mentioned processes and accompanying equations is discussed in this section.

4.1 CTRA Motion Analysis

The random walk process according to regular Markov transition matrix; grantees that any CTRA will cover all network nodes in a certain number of hopes and that the swarm distribution will be uniform at any time instance. This process also provides the maximum amount of noise to the system.

4.2 Game Analysis

4.2.1 Game Coverage

In the anonymous communication game, each CTRA can be modelled as a state and the message rerouting can be modelled as a stochastic transition matrix. The probability that the message be in certain state -rerouted to this CTRA from another one- varies at first, then goes to the steady state equal to the limit at game length equal to infinity. The following figure shows this phenomenon for different team sizes ($T = 3, 5, 10, 30$).

It is clear that the steady state is reached faster as the team size increases. The steady state grantees that game coverage to all CTRAs will be approximately equal if the message lifetime (μ) was multiple times of the team size. It is clear that game coverage can be interpreted in different mean to be the probability to find the message in certain state, CTRA at any time. This prevents the attacker from having any prior knowledge about the anonymous communication game, which guarantees maximum immunity against an attack.

4.2.2 Game Length

The game length (or number of message reroutes between CTRAs) is determined by an inverse Bernoulli process according to Equation (2). Figure 2 shows the probabilities that the game length assumes certain value regarding three team sizes: 3, 5 and 9 CTRAs. As the team size increases, the probability that the game length take higher values decreases. For any team size; the probability of

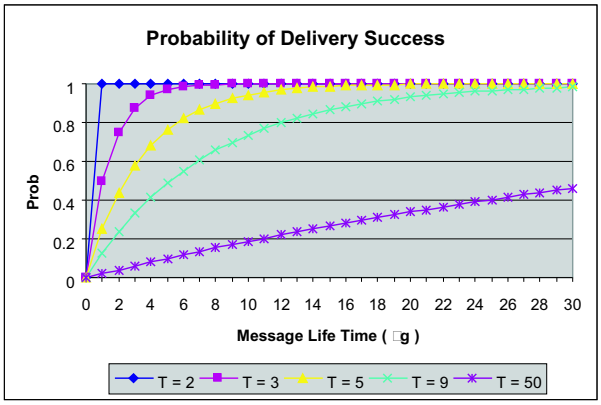

Figure 3: Probabilities of delivery success vs. game length for team sizes equal to 2, 3, 5, 9 and 50

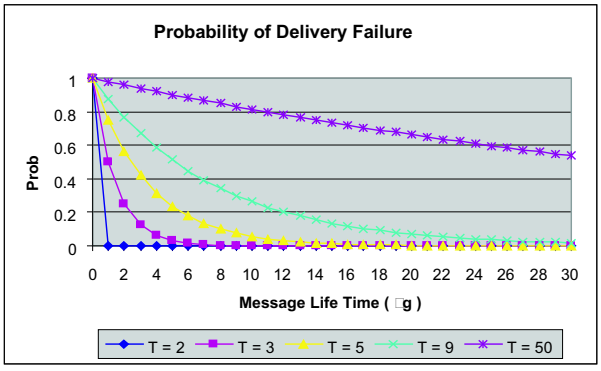

Figure 4: Probabilities of delivery failure vs. game length for team sizes equal to 2, 3, 5, 9 and 50

game length equal to zero is usually equal to $p/(1-p)$ where $(p = 1/(T-1)$, and T is the team size. This probability decreases dramatically as the game length increases.

4.2.3 Delivery Success and Failure

Delivery success means that the specified message lifetime (μ) is long enough for the message to be rerouted from the PA to WPA. Figure 3 shows the relation between game length and probability of success for five team sizes (T): $2, 3, 5, 9$ and 50.

The probability of success increases with the increase of the message lifetime (μ). For smaller team sizes, the probability of success reaches the unity faster than for the larger team sizes. Figure 4 shows the complement, i.e., the probability of failure against game length. It is clear that the probability of failure increases as the team size increases. This leads us to the next section; the reliability analysis.

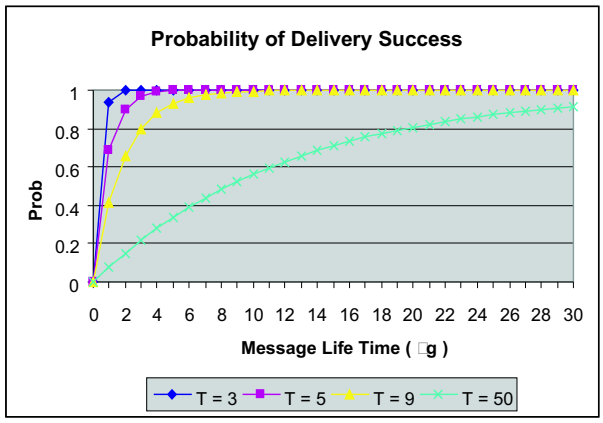

Figure 5: Probabilities of delivery success vs. game length for team sizes equal to 3, 5, 9, and 50 & 4

4.2.4 Reliability Analysis

To increase the probability of message delivery success and give immunity against natural and intended game failure, the message is sent initially to a selected number of PAs that was called VPAS. This means that there will be a number of simultaneous games running at the same time, which will increase the chance of delivery success. This can be seen from another point of view as increasing the level of reliability of the ACS against intended denial of service attacked or natural failures. Figure 5 shows the probability of success vs. game length for team sizes (T) equal 3, 5 and 9 and number of simultaneous games equal $(\tau = 4)$.

4.2.5 Performance Analysis

Time delay is one of the most important aspects in any ACS. Some ACSs suffer from timing attack because of its real time behavior. The traditional onion routing is a clear example. The immunity of the traditional onion routing algorithm increases as the traffic volume increases [20]. The larger the traffic volume, the more unexpected timing delay can appear. In the proposed algorithm, the expected game length increases -linearly as a feature of the inverse Bernoulli process- as the team size increases. There are two important parameters, the game length that represents the number of retransmissions in the ACG, and the effective game length that represents the total number of transactions in the game including tunnelling and retransmission activities. Figure 6 shows the relation between expected game (α) length and the team size (T).

As mentioned before, the effective game length represents the total number of transactions taken in-between the message initiation and receiving. The effective path length depends mainly on the tunnelling parameter $(0 < r < 1)$. If the tunnelling parameter (r) takes the value of unity, no tunnelling will accrue at all. Figure 7 shows the relation between effective game length and the tunnelling

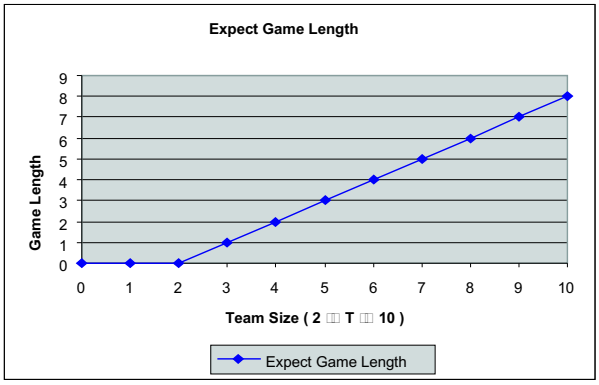

Figure 6: Expected game length vs. team size.

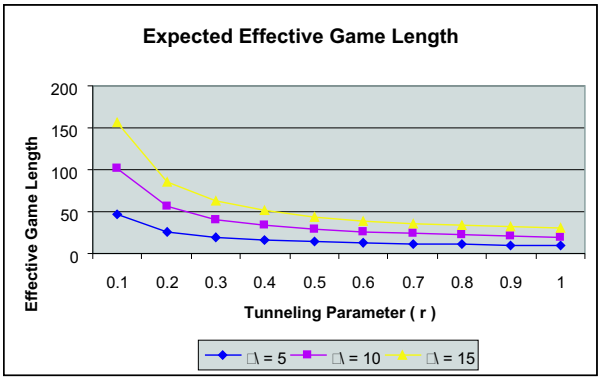

Figure 7: Effective Game Length vs. tunnelling parameter for game length (α) equal to 5, 10 and 15

parameter for different game lengths.

4.2.6 Scalability Analysis

Scalability means that the system is able to deliver the required services with the increase of the number of service providers, service receivers and system size. In our system, the overall space had been divided into clusters or ACTs. To provide scalability, number of teams and teams' sizes can be adjusted beside the number of simultaneous games to maintain the required performance and reliability requirements. As an example, team size can be increased while maintaining the same level of delivery success by adding more simultaneous games. Larger teams can serve larger Anonymous Communication Cluster (ACC). Figure 8 shows the ability to adjust higher delivery success probability for larger team size ($T = 50$) by employing 9 teams in a multi-team game.

5 Security Analysis

In this section, we provide a detailed security analysis.

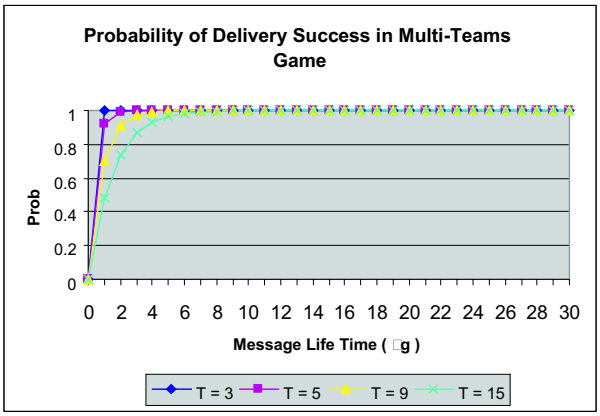

Figure 8: Probabilities of delivery success vs. game length for team sizes equal to 3, 5, 9, and 15.

5.1 Probability of True Sender Being Identified

5.1.1 Basic Assumptions

Statistical methods will be used through this analysis to provide a quantitative analysis for the provided anonymity degree. Our analysis will be built on a lot of basis mentioned in [14, 16]. The concepts of path length selection, path topologies, threat model, adversary type are used without modifications. In addition, elimination rules that were mentioned in [14], which are used to construct the non-sender set (NS) will be used here as well. However, in the proposed algorithm, the "Path Length Selection" is performed stochastically. Therefore, we have only variable path length and effective path length as illustrated in performance analysis section. "Path Topology" is always a complicated path, i.e., arbitrary path without disjoint constraints. This topology is initially generated from the ACG.

We interpret the ability to reconstruct the rerouting path from the same perspective as in [35]. If first node in the rerouting path is compromised, all information of the rerouting path is clear. In our case, compromising the first node does not explore the complete session, but guarantee the observability of the initiator without guarantee the linkability due to the existence of the WPAs. The difference between observability and linkability will be explained in Section 5.1.3.

5.1.2 Attacked Resources

In traditional rerouting algorithm, the physical network nodes play the role of routers. In our proposed algorithm, the CTRAs play this role. The attacker has two choices; either to attack the physical network nodes or to attack the CTRAs. It is evident that attacking a mobile agent is simpler than attacking a physical network node. The impact analysis section will shows that attacking CTRAs usually increases the probability to discover the identity of

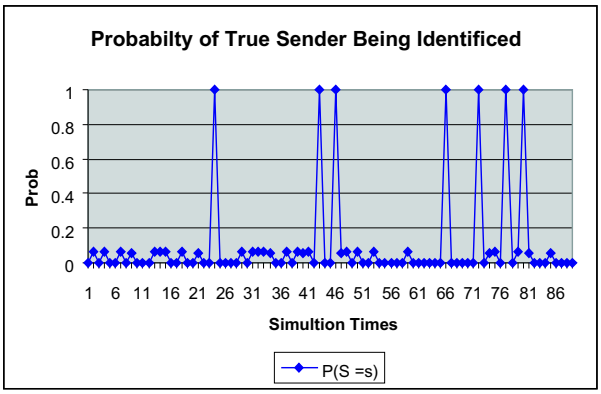

Figure 9: Probability to detect the true initiator in 90 sessions using traditional onion routing

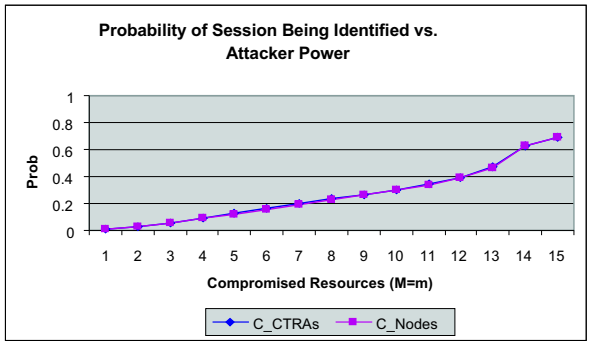

Figure 10: Probability of true communication session being discovered vs. attacker power ($N = 15, n = 15, T = 5 and \mu = 8$)

the communication session than attacking physical nodes for swarm size to network size ratio equal to unity. For higher values of this ratio, attacking physical network nodes usually provides higher probability than attacking CTRAs with highly significant difference.

5.1.3 Observability vs. Linkability

As illustrated in the introduction section. The anonymous communication researches differentiate between the communication anonymity, observabililty and linkability [24]. Unobservability guarantees unlinkability and hence anonymity. In traditional rerouting systems for anonymous communication, the receiver is always compromised. If the attacker observes the initiator and is able to correlate the message observed at the initiator site and the receiver site, he will be able to link initiator to receiver in this communication session. In this case, no anonymous service had been provided at all.

A simulation of 90 communication sessions preformed by a traditional rerouting algorithm on ACS with network size equal to 100 ($N = 100$) and very limited attacker power equal to 6 ($M = 6$) and relatively small rerouting path length equal to 9 ($L = 9$) had been conducted. Figure 9 shows the result of the ability of the attacker to detect the identity of the true sender.

The majority of these communication sessions gave very small probabilities. Little number of sessions gives probability equal to unity. These sessions were the sessions that select a compromised node randomly from the network to be the first node in the rerouting path. The ratio of sessions that give unity to the total number is close to the compromised node (M) to the network size (N). The result will exactly reach the (M/N) ratio as the number of simulation increases.

If we can make the attacker unable to link the initiator to the receiver even if he was able to observe the initiator, the ability of the attacker to detect the identity of the true sender will decrease dramatically. In our algorithm, the message continues to reroute even after it reaches its

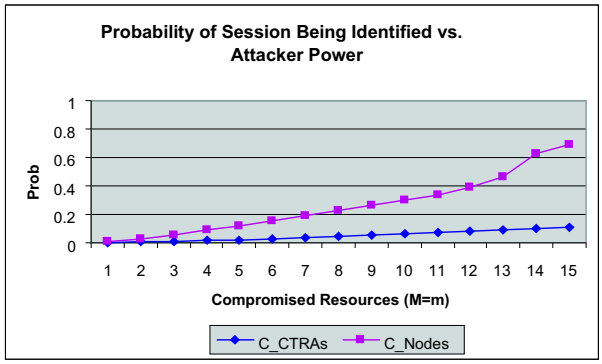

Figure 11: Probability of true communication session being discovered vs. attacker power ($N = 15, n = 45, T = 5 and \mu = 8$) teams

destination. In this case, all intermediate nodes can be a valid destination from the attacker point of view. The result will be an increase in the provided anonymity degree, as we will illustrate in the security analysis section.

5.1.4 Anonymous Equations

This section introduces the equations required to calculate the provided degree of anonymity for case of compromised CTRAs only Equation (7) and compromised nodes only Equation (8). These equations will be used to calculate the probability that the identity of the true communication session being identified.

We will use the following parameters:
μ = message lifetime, as mentioned before.
L = actual game length.
$L*$= assumed game length by the attacker.
α = stochastic game length, as mentioned before.
N = number of network nodes.
M = number of compromised network nodes.
n = number of CTRAs (swarm size).
m = number of compromised CTRAs.

T = team size.
i = compromised resources in the rerouting path.
j = certain instance that represents the location of compromised resources (i) in the rerouting path.
The equations are:

$$P(S=s) = \sum_{L=1}^{\mu}(\sum_{k=w}^{u}(\sum_{z=0}^{v}(\sum_{L^*=1}^{\mu} G(L,L^*,i,k)P(L^*=\alpha))P(z=j))P(k=i))P(L=\alpha) \quad (7)$$

$$P(S=s) = \sum_{L=1}^{\mu}(\sum_{k=w}^{u}(\sum_{z=0}^{v}(\sum_{L^*=1}^{\mu} H(L,L^*,i,k)P(L^*=\alpha))P(z=j))P(k=i))P(L=\alpha) \quad (8)$$

where $G(L,L^*,i,k)$ denotes probability to identify the identity of the true sender in a certain communication session when only CTRAs are compromised; $H(L,L^*,i,k)$ denotes probability to identify the identity of the true sender in a certain communication session when only physical nodes are compromised.

$$\begin{aligned} w &= Max[0, T+1-(n-m)] \\ u &= Min[m, T+1] \\ v &= \binom{L+1}{Floor[(\frac{L}{T+1})i]} \\ P &(L=\alpha) = P(L^*=\alpha) = pq^{\alpha-1} \\ p &= 1/(T-1) \\ q &= 1-p \\ P &(z=j) = 1/\binom{L+1}{Floor[(\frac{L}{T+1})i]} \\ P &(k=i) = \frac{\binom{m}{i}\binom{n-m}{T+1-i}}{\binom{n}{T+1}} = \frac{\binom{M}{i}\binom{N-M}{T+1-i}}{\binom{N}{T+1}} \end{aligned} \quad (9)$$

The following section presents the impact of the changes in some of the parameters of these two equations and compares the behavior of the two ACS in various cases.

5.2 Impact Analysis

In this section, we provide a detailed impact analysis for the proposed system.

5.2.1 Impact of Attacker Power

The power of an internal-passive attacker can be measured in terms of the number of compromised network nodes (M) or CTRAs (m). Figure 14 shows the impact of attacker power on the probability of true communication session being discovered in both cases (compromised nodes and compromised CTRAs). This probability increases as attacker power increases, which is matching with the preliminary intuitive thinking. Figure 10 shows small difference between the two compromised cases at (Swarm size / Network size) ratio equal to unity. The gap increases as this ratio increases. Figure 11 shows this case at (Swarm size / Network size) ratio equal to 3.

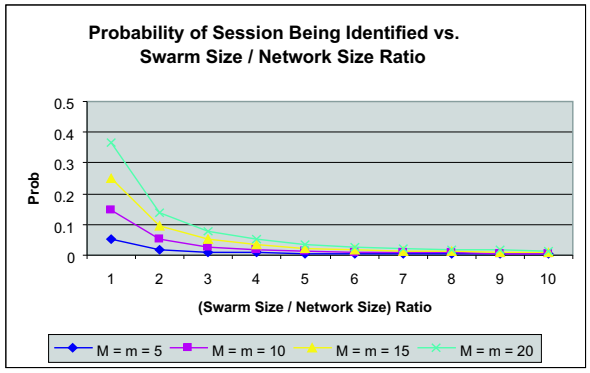

Figure 12: Probability of true initiator identity being discovered vs. (Swarm size/Network size) ratio for attacker power (m) equal to $5, 10, 15, 20$

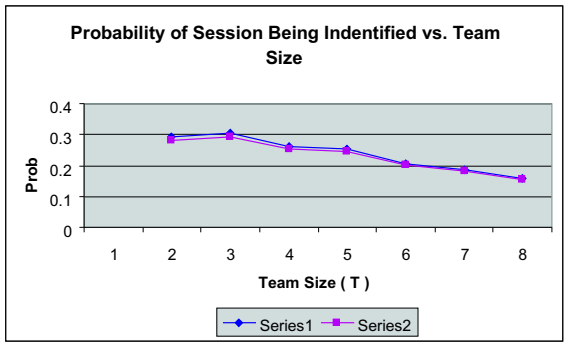

Figure 13: Probability of communication session being identified versus team size ($N = n = 25, m = 15, \mu = 8$) teams

5.2.2 Impact of Swarm Size

Naturally, swarm size does not affect the probability of true communication session being discovered if the attacker compromises network nodes only. The effect of swarm size appears if the attacker attacks CTRAs. For limited power attacker, which cannot attack more than certain number of resources; the swarm size can have significant effect on the provided degree of anonymity. Figure 12 shows the case of network with ($N = 25$) and swarm size to network size ratio varies from Equation (1) to (9) against limited attacker power equal to 5, 10, 15 and 20 CTRAs respectively. The swarm size decreases the probability that the true communication session being identified dramatically as it increases. The effect of swarm size is one of the most positive points in the proposed algorithm over the traditional one. In traditional rerouting solution, ACS owner or administrator cannot mitigate the increasing in the attacker power. It is im-

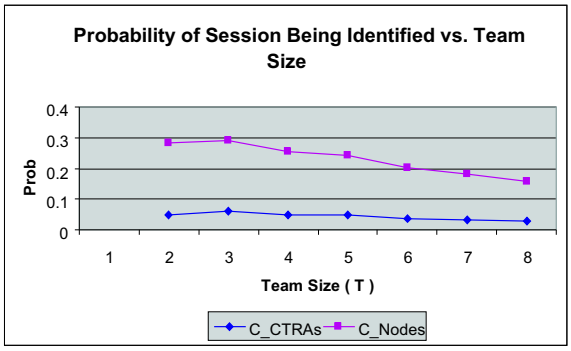

Figure 14: Probability of communication session being identified vs. team size ($N = 25, n = 75, m = 15, \mu = 8$)

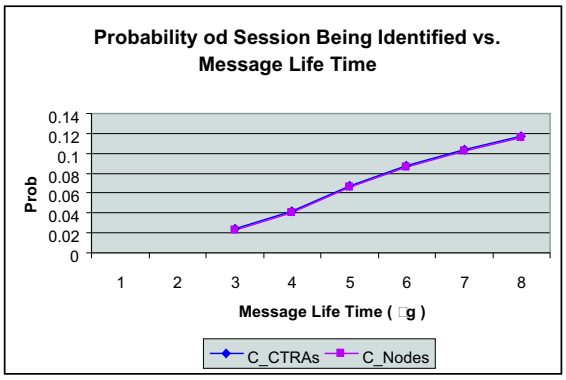

Figure 15: Probability of communication session being identified vs. μ ($N = n = 15$, $m = 5$ and $T = 4$)

possible to add more physical network nodes to minimize the effect of attacker power. In our case, the swarm size is a control factor. The administrator can increase the swarm size to minimize the effect of attacker power.

From the obtained results, it is clear that at certain thresholds -i.e. certain (Swarm Size/Network Size) ratio- the change in attacker power does not have significant effect on the probability of true session being identified. As an example, for (Swarm Size/Network Size) ratio equal to 2 and 6; the duplication in attacker power gives rising in the probability less than (0.05) and (0.01) respectively.

5.2.3 Impact of Team Size

It is very evident that, as the team-size increases, while the attacker power still constant; the probability of true initiator identity being discovered will change. Figure 13 and Figure 14 show the impact of team size for two systems with different (swarm size/network size) ratios. This decreasing is very logical.

5.2.4 Impact of Message Lifetime

Intuitively, one expects that the probability of the initiator being identified decreases as the path length - the

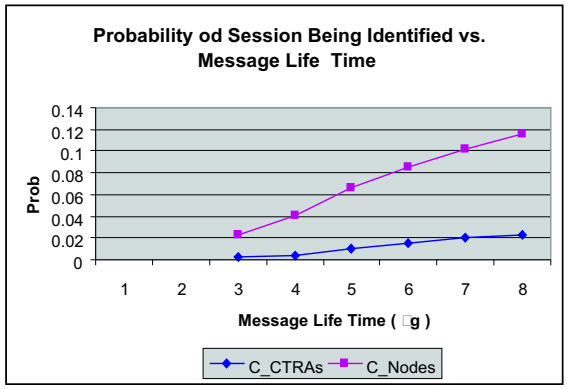

Figure 16: Probability of communication session being identified vs. μ ($N = 15$, $n = 45$, $m = 5$ and $T = 4$)

maximum game length in our case (μ) - increases. For the first time, reference [31] stated that the contrary is true. The results for our proposed algorithm also assure this perception. Figure 15 and Figure 16 show that the calculated probability increases as the message lifetime increases.

5.2.5 Comparison with Traditional Rerouting Algorithms

As mentioned in Section 5.1.1, our analysis is based on some of the concepts mentioned in [16, 31]. Therefore, it is helpful to provide a comparison between the levels of provided anonymity in the proposed and traditional rerouting algorithms.

The most important comparison is the comparison of the provided anonymity against the attacker power at a rerouting path length is equal to the message lifetime (μ). Figures 17 and 18 show this comparison, and clarifies the large difference in the provided anonymity degree.

The two figures show comparison results of the proposed algorithm to the traditional rerouting algorithm for (swarm size/network size) ratio equal one. Figure 19 shows the result for ratio equal 3. There is a difference in response time even if the (μ) is set to be equal to the path length (L) due to the tunnelling actions as illustrated in Section 4.2.5. If the message has been delivered in a response time approximately equal to the effective path length, it can be seen as an acceptable cost for the increasing in the degree of the provided anonymity. The comparison had been done against simple fixed path length in the traditional onion routing; this is due to the minimum impact of path topology at any game length and the minimum impact of path selection strategy at sufficient long path length [31].

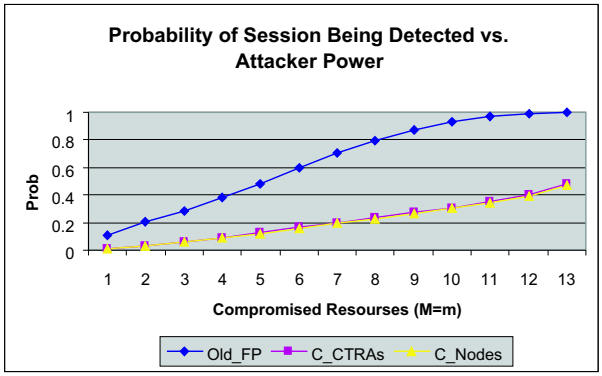

Figure 17: Probability of communication session being identified vs. attacker power ($N = 15, n = 15, T = 5, L = \mu = 8$)

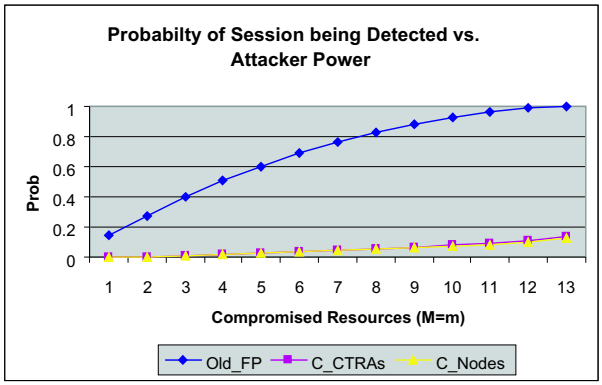

Figure 18: Probability of communication session being identified vs. attacker power ($N = 15, n = 15, T = 5, L = \mu = 3$)

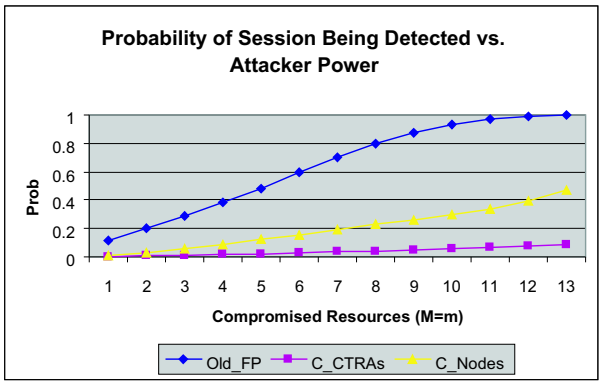

Figure 19: Probability of communication session being identified vs. attacker power ($N = 15, n = 45, T = 5, L = \mu = 8$)

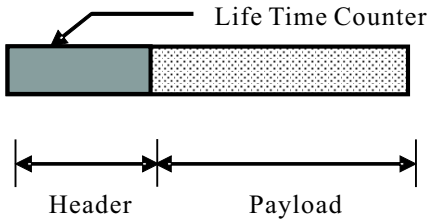

Figure 20: The message format

6 Message Format and Data manipulation

As we have thought before, our message does not contain any information about its initiator or its receiver. Particularly concerning the receiver, it does not contain its IP address or any linkable pseudonym to its identity. This is the reason we call our algorithm "*IPLess*". The message in our algorithm contains only two parts the message header, and the original payload that contains the information, which the initiator want to send to the receiver.

The message header contains, as shown in Figure 20, The *Message Lifetime Counter*. The *Message Lifetime Counter* is a counter that is incremented by each CTRA receives the message. Its initial value -when the message goes out from the initiator AA- is set to zero. When it exceeds the limit of the maximum message lifetime (μ) the message is destroyed by the first CTRA receives it. The purpose of this counter is to prevent the *message life look* and hiding the true game length.

In the traditional onion routing algorithm, the message had been encrypted as a one part with a nested encryption process. This process uses the reverse sequence of the keys that had been distributed on the intermediate nodes in the anonymous channel formation phase prior to the transmission itself [11, 12]. In our proposed algorithm, the sequence of the intermediate nodes (agents or formally *CTRA*) is not known prior to the transmission phase. We do not have a path (Channel) construction phase. In addition, there is no any IOI in our algorithm that knows the path before the start; neither of the communication session nor during the session itself. Even after the session ends, no IOI can know exact path.

In order to enable the feature of stochastically determine the rerouting path during the transmission phase itself; the onion must be formed with a different manner and is not affected by the rerouting activities. We introduce the concept of heterogeneous onion to solve this problem. The heterogeneous onion does not designed to prevent the attacker from being able to correlate the message form while it passes through different compromised resources. The main purpose is to hide the content strongly enough to secure the logical data of the communication session. This feature cannot be seen as a weakness in security.

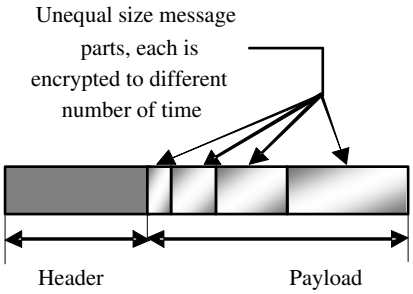

Figure 21: The heterogeneous message format

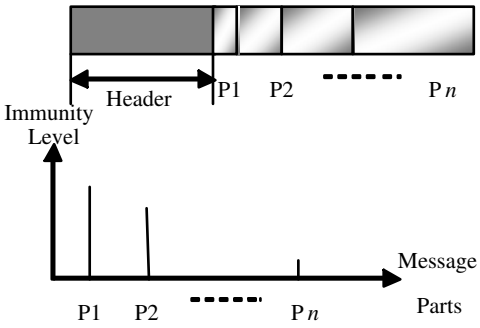

Figure 22: The immunity degree in the linear heterogeneous onion

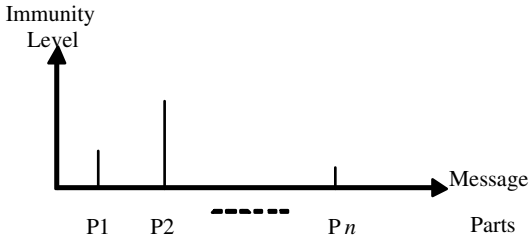

Figure 23: The immunity degree in non-linear heterogeneous onion

The message correlation, to some degree, cannot be assumed to be prevented 100% even after nested encryption in traditional onion routing [**?**]. Repetitive attack and its countermeasure URE-Onion algorithm is a shiny example of the active attacker's correlation ability. Our algorithm provides natural immunity against the repetitive attack, because it is very rare to find two messages transferred through two identical games. The heterogeneous onion is formed by the initiator AA when it follows this sequence of steps and as demonstrated in Figure 21.

1) The initiator AA divides the message to unequal size parts. The number of these parts and their sizes are predefined and well known to both the initiator AA and the receiver AA.

2) The initiator AA starts to encrypt each part with a different times of encryptions. The first part is encrypted only one time. The second part is encrypted two times, the third one three times and so on. This is called "Linear Heterogeneous Onion", as shown in Figure 22. The different parts of the message can be encrypted to different levels in unordered fashion, "Non-linear Heterogeneous Onion", as shown in Figure 23. All encryptions are done by the same key, which is the private key of that AA.

The second purpose of that heterogeneous onion is to solve the weakness point in the traditional onion. The traditional onion starts, i.e., when it goes out from the initiator - with its highest immunity degree. Each rerouting step, a layer is decrypted, which means that the immunity degree of that onion had been decreased by one degree.

7 Summary and Conclusion

The need to communicate anonymously with a chosen party has become an essential service in today's communication systems. Anonymity service can be provided for the initiator, respondent or both in this work:

- We have proposed an algorithm for a stochastic anonymous communication system that utilizes mobile agents to provide anonymity for both initiator and respondent.
- We have discussed the concepts of anonymous communication game.
- We have provided a mathematical model for this type of games to calculate game parameters such as mobile agents' motion, game length, delivery success and failure probabilities, reliability, scalability, performance analysis in terms of response time and multi-team game.
- Using this mathematical model, one is able to study the impact of attacker power, team size, swarm size, message lifetime.
- We have introduced a comparison with one type of traditional rerouting algorithms.
- In addition, we have suggested an "IPLess" message format that is associated with the concepts linear and non-linear heterogeneous onion.
- Finally, a simple data manipulation algorithm had been illustrated to facilitate the operation of stochastic game.

Our algorithm is constructed on the fundamental nature of traditional onion routing algorithm, which is one of the most illustrious algorithms of the rerouting family of the

ACSs. However, we were able to provide a number of novel concepts in the discipline of ACS such as stochastic rerouting, anonymous game, heterogonous onion and finally the "IPLess" message. This message format can be interpreted, in more generalized perspective, as a message with no pseudonym to the initiator, the respondent or the communication session. The message can be incorporated in the optional field of the IP options field. The proposed approach was shown, through mathematical approach, to provide more anonymity degree than the traditional rerouting algorithm at similar ACS conditions.

References

[1] *Anonymous Remailer [Online]*, Available: http://www.lcs.mit.edu/research/anonymous.html

[2] O. Berthold, H. Federrath, and M. Kohntopp, "Project anonymity and unobservability in the Internet," in *Workshop on Freedom and Privacy by Design, Conference on Freedom and Privacy*, pp. 57-65, 2000.

[3] D. Chaum, "Untraceable electronic mail, return addresses, and digital pseudonyms," *Communications of the ACM*, vol. 24, no. 2, pp. 84-88, 1981.

[4] D. Chaum, "The dining cryptographers problem: Unconditional sender and recipient untraceability," *Journal of Cryptology*, vol. 1, no. 1, pp. 65-75, 1988.

[5] C. Diaz, and B. Preneel, "Anonymous communication," appeared in *WHOLES, A Multiple View of International Privacy in a Networked World*, 2004.

[6] C. Diaz, and B. Preneel, "Reasoning about the Anonymity Provided by Pool Mixes that Generate Dummy Traffic," in *Proceedings of 6th Information Hiding Workshop*, pp. 309-325, May 2004.

[7] R. Dingledine, *The Free Haven Project: Design and Deployment of an Anonymous Secure Data Haven*, Master's thesis, MIT, Jun. 2000.

[8] R. Dingledine, N. Mathewson, and P. Syverson, "Tor: The second-generation onion router," in *Proceedings of the 13th USENIX Security Symposium*, pp. 303-320, Aug. 2004.

[9] S. Goel, M. Robson, M. Polte, and E. G. Sirer, *Herbivore: A Scalable and Efficient Protocol for Anonymous Communication*, in Cornell University Computing and Information Science Technical Report, TR2003-1890, Feb. 2003.

[10] I. Goldberg and A. Shostack, *Freedom Network 1.0 Architecture and Protocols*, http://www.freedom.net/info/freedompapers/index.html, 1999.

[11] D. Goldschlag, M. Reed, and P. Syverson, "Hiding routing information," in *Information Hiding*, LNCS 1174, pp. 137-150, Springer-Verlag, June 1996.

[12] D. Goldschlag, M. Reed, and P. Syverson, "Onion routing for anonymous and private internet connection," *Communications of the ACM*, vol. 42 no. 2, pp. 39-41, Feb. 1999.

[13] M. Gomulkiewicz, M. Klonowski, and M. Kutylowski, "Onions based on universal re-encryption - anonymous communication immune against repetitive attack," in *Workshop on Information Security Applications, WISA'2004*, pp. 400-410, 2004.

[14] Y. Guan, X. Fu, R. Bettati, and W. Zhao, *A Quantitative Analysis of Anonymous Communications*, Technical Report TR01-016, Department of Computer Science, Texas A & M University, July 2001.

[15] Y. Guan, X. Fu, R. Bettati, and W. Zhao, "An optimal strategy for anonymous communication protocols," in *Proceedings of 22nd IEEE International Conference on Distributed Computing Systems (ICDCS 2002)*, pp. 256-266, 2002.

[16] Y. Guan, X. Fu, R. Bettati, and W. Zhao, "A quantitative analysis of anonymous communications," *IEEE Transaction on Reliability*, vol. 53, no. 1, pp. 103-115, Mar. 2004.

[17] J. Iwanik, M. Klonowski, and M. Kutylowski, "DUO-onions and hydra-onions failure and adversary resistant onion protocols," in *Proceedings of IFIP International Federation for Information Processing*, vol. 175, pp. 1-15, Jan. 2005.

[18] S. Kitazawa, M. Soshi, and A. Miyaji, "An agent-based model of anonymous communication protocols," in *10th IEEE International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprise*, pp. 177-182, 2001.

[19] M. Klonowski, M. Kuty_owski, and F. Zag'orski, "Anonymous communication with on-line and off-line onion encoding," in *Proceedings of SOFSEM*, pp. 229-238, 2005.

[20] L. Korba, R. Song, and G. Yee, *Anonymous Communications for Mobile Agents*, Institute for Information Technology, National Research Council of Canada, NRC Paper number: NRC-44948, 2000.

[21] L. Korba, and R. Song, *Investigation of Network-based Approaches for Privacy*, Institute for Information Technology, National Research Council of Canada, NRC 44900, Nov. 2001.

[22] A. Mislove, G. Oberoi, A. Post, C. Reis, P. Druschel, and D. S. Wallach, "AP3: Cooperative, decentralized anonymous communication," in *Proceedings of SIGOPS-EW, Leuven*, Belgium, Sep. 2004.

[23] R. E. Newman, I. S. Moskowitz, P. Syverson, and A. Serjantov, "Metrics for traffic analysis prevention," in *Proceedings of Privacy Enhancing Technologies Workshop (PET 2003)*, pp. 25-28, Mar, 2003.

[24] A. Pfitzmann and M. Kohntopp, "Anonymity, unobservability, and pseudonymity, a Proposal for terminology," Hannes Federath (Ed.), Designing Privacy Enhancing Technologies, Lecture Notes in Computer Science, LNCS 2009, pp. 1-9, Springer-Verlag, 2001.

[25] J. Raymond, "Traffic analysis: Protocols, attacks, design issues, and open problems," in *Hannes Federrath, editor, Designing Privacy Enhancing Technologies: Design Issues in Anonymity and Observability*, LNCS 2009, pp. 10-29, Springer-Verlag, 2000.

[26] M. G. Reed, P. F. Syverson and D. M. Goldschlag, "Protocol using anonymous connections: Mobile application", in *Proceedings of the 5th International Workshop on Security Protocols*, pp. 13-23, 1997.

[27] M. K. Reiter, and A. D. Rubin, "Crowds: Anonymity for web transactions," *Communications of the ACM*, vol. 42, no. 2, pp. 32-48, 1999.

[28] R. Sherwood, B. Bhattacharjee, and A. Srinivasan, "P5: A protocol for scalable anonymous communication," in *Proceedings of the IEEE Symposium on Security and Privacy*, pp. 58-70, May 2002.

[29] C. Shields and B. N. Levine, "A Protocol for anonymous communication over the internet," in *Proceedings of the ACM Conference on Computer and Communications Security*, pp. 33-42, 2000.

[30] E. G. Sirer, M. Polte, and M. Robson, *CliqueNet: A Self-Organizing, Scalable, Peer-to-Peer Anonymous Communication Substrate*, Unpublished manuscript, http://www.cs.cornell.edu/People/egs/papers/cliquenet-iptp.pdf, Dec. 2001.

[31] R. Song and L. Korba, "Review of network-based approaches for privacy," in *Proceedings of the 14th Annual Canadian Information Technology Security Symposium*, pp. 13-17, Ottawa, NRC 44905, May, 2002.

[32] R. Song and L. Korba, "Simulation on agent-based onion routing network," in *NRC/ERB 1105*, NRC 46529, 12, Nov. 2003.

[33] R. Song and L. Korba, "Scalability of agent-based onion routing network," in *Proceedings of the 19th International Conference on Computers and Their Applications (CATA-2004)*, Seattle, USA, NRC-46541, Mar. 2004.

[34] P. Syverson, M. Reed, and D. Goldschlag, "Private web browsing," *Journal of Computer Security*, vol. 5, no. 3, pp. 237-248, 1997.

[35] M. Wright, M. Adler, B. N. Levine, and C. Shields, 'An analysis of the degradation of anonymous protocols," in *Proceedings of the Network and Distributed Security Symposium - NDSS'02*, Feb. 2002.

Magdy Saeb received the BSEE. In Electrical Engineering, School of Engineering, Cairo University, in 1974; the MSEE. and Ph.D. in Electrical & Computer Engineering, School of Engineering from the University of California, Irvine, in 1981, 1985 respectively. He was with Kaiser Aerospace and Electronics, Irvine California, and the Atomic Energy Establishment, Anshas Egypt. He is currently a professor and head of the Computer Engineering Department, Arab Academy for Science, Technology & Maritime Transport, School of Engineering, Alexandria, Egypt. His current research interests include Computer Network Security, FPGA Implementation of Encryption and Steganography security techniques, Encryption processors, and mobile agents-based security techniques.

Ahmed S. El-Shikh, Process Improvement Engineer in Software Engineering competence Center (SECC), Egypt. He has six years of experience in software industry. He worked as a research assistant for three years in the Faculty of Engineering, Mansoura University. He is editor for the Egypt SPIN newsletter from two years ago. He holds a BS. in "Control and Computer engineering", from Mansoura University, Egypt, 1998 and a post-gradate diploma in the "Total Quality Management" from the American University in Cairo, 2004. He is a certified "Personal Software Process Engineer" from The Software Engineering Institute (SEI), USA, 2005. His interests include; network security, software engineering, software quality management, statistical quality control and process Improvement models and approaches specially "CMMI" and "Six Sigma".