Towards Anonymity in Mobile Ad Hoc Networks: The Chameleon Protocol and its Anonymity Analysis

Leonardo A. Martucci, Christer Andersson, and Simone Fischer-Hübner

Karlstads University, Department of Computer Science
Universitetsgatan 2, 651-88 Karlstad, Sweden
{leonardo.martucci, christer.andersson, simone.fischer-huebner}@kau.se

Abstract. This paper presents Chameleon, a novel anonymous overlay network for mobile ad hoc environments. As far we know, Chameleon is the first low-latency anonymous overlay network applied in a mobile ad hoc setting. It was designed with the special characteristics of mobile ad hoc networks in mind, such as limited battery lifetime, user mobility and vanishing nodes. In this paper, we also evaluate Chameleon against a number of requirements that an anonymous overlay network should adhere to in order to be suitable for mobile ad hoc networks. In particular, the anonymity properties of Chameleon are thoroughly analyzed.

1 Introduction

Mobile ad hoc networks are constituted of mobile platforms that establish on-the-fly wireless connections among themselves, and ephemeral networks without central entities to control it. Mobile ad hoc networking is an important building block for ubiquitous computing, as it allows instantaneous networking between mobile devices without the interference or aid of central devices for network establishment. Mobile ad hoc networks present many interesting research challenges due to their mobile and decentralized nature as well as their self-configuration and self-maintenance requirements. Among the most challenging aspects of mobile ad hoc networks is the users' privacy. The quest for privacy in mobile ad hoc networks is currently focused on introducing anonymity in the network layer, with several anonymous routing protocols being recently proposed [14, 26, 6]. However, such solutions prevent the usage of standardized ad hoc routing protocols, meaning, in practice, that all network nodes must run a non-standard routing protocol.

Our proposal, Chameleon, is an anonymous overlay network tailored for mobile ad hoc environments, aiming, with reasonable performance costs, to provide sender anonymity against recipients and relationship anonymity against local observers. In addition, Chameleon provides conditional anonymity against malicious Chameleon users, as well as protection against single attackers trying to compromise large portions of a network by assuming multiple identities. Chameleon builds on a flexible design that provides isolation and independence from both the application and transport layers, allowing the usage of standardized mobile ad hoc routing protocols. To the best of our knowledge, Chameleon is the first low-latency anonymous overlay network being applied in a mobile ad hoc setting.

Chameleon was specially designed with the characteristics of mobile ad hoc environments in mind. Therefore, when designing Chameleon, key characteristics of those environments, such as limited battery lifetime, user mobility and vanishing nodes, for instance, were taken into account. The core functionalities of Chameleon are inspired by the traditional Crowds system [18] for anonymizing HTTP traffic. This decision was made according to a previous evaluation of Peer-to-Peer (P2P) based anonymous overlay networks in the context of ad hoc networks [4]. Although none of the studied techniques were fully compliant with the characteristics of mobile ad hoc networks, Crowds [18] was deemed as an appropriate choice for a foundation upon which Chameleon could be developed. A number of adaptations to Crowds were made. For example, Chameleon enables end-to-end encryption between a sender and a recipient, employs certificates to hinder attackers from assuming multiple identifies, and acts as a general overlay network accepting all messages from the application layer.

The rest of this paper is organized as follows. Section 2 reviews related work aiming to provide anonymity in mobile ad hoc environments. Section 3 presents a discussion regarding identification and anonymity in mobile ad hoc networks, which we called the identity-anonymity paradox. In Section 4, we introduce Chameleon by describing its basic foundations, including the protocol overview and its assumptions. Section 5 presents the assumed attacker model in Chameleon and, further, analyzes the offered degree of anonymity against this attacker model. Finally, Section 6 presents concluding remarks and future research plans.

2 Definitions & Related Work

Anonymity is often seen as the best strategy for enabling privacy. Pfitzmann and Hansen [17] define anonymity as: “the state of being not identifiable within a set of subjects, the *anonymity set*”. The anonymity set includes all possible subjects in a given scenario (e. g., senders of a message). Related to anonymity is unlinkability, which is defined in [17] as: “unlinkability of two or more items means that within this system, these items are no more and no less related than they are concerning the *a priori* knowledge”. Anonymity can be defined in terms of unlinkability: *relationship anonymity* means that an observer is not able to link a specific sender to a corresponding receiver; *sender anonymity* entails that a message cannot be linked to the origin sender; and *receiver anonymity* implies that a message cannot be linked to the receiver of that message. When applying these definitions on Chameleon, it can be noted that Chameleon aims mainly at providing sender anonymity against recipients and relationship anonymity against local observers. Regarding general schemes for enabling anonymity in mobile ad hoc networks, there are currently two main strategies:

1. *Replacing the standard ad hoc routing protocol with a routing protocol that enables anonymous communication (see Figure 1).*
   In recent years, a number of such proposals have been published, including: ANODR [14], MASK [26], SDAR [6], and ARM [21]. Most of these solutions aim to anonymize Route Request (RREQ) and Route Reply (RREP) messages during route discovery. The main advantage of this approach is that messages can be directly transmitted to the destination using in average shorter paths in comparison with anonymous overlay networks (see below). The main disadvantage is the mere fact that the standard routing protocol is being replaced. This forces users to run another routing protocol when they want to be anonymous. Therefore, the risk is that such solutions will end up with a small user base, and, thus, a degraded degree of anonymity. Another disadvantage is that the anonymity offered by this type of solutions could be exposed in cases when a connection-oriented transport layer, such as TCP, is being used above the anonymous routing protocol (see Figure 1);
2. *Introducing an anonymous overlay network above the ad hoc routing protocol or the transport protocol (see Figure 2).*
   This type of solution, which Chameleon adheres to, introduces an anonymous overlay network on top of either the network layer or the transport layer. One advantage with introducing anonymity by the means of an overlay network is flexibility; such a solution is independent of the routing protocol and, further, is compatible with applications expecting services from a reliable transport layer. One disadvantage is that the performance can be expected to be slightly worse compared to anonymous routing protocols, as messages are routed through a set of intermediary overlay nodes instead of being transmitted via the shortest route between the sender and the recipient. A recent proposal belonging to this category is [12], where Jiang *et al.* propose a number of adaptations to make Chaum's classical mix concept [8] suitable for ad hoc networks. In contrast to Chameleon, this proposal claims to provide anonymity against a global observer. Still, it is recognized in [12] that to meet this goal, bandwidth-consuming dummy traffic is likely to be needed. Further, this proposal requires more nodes to perform special (costly) functions than Chameleon, as a subset of the nodes have to act as intermediary mixes during message transfer, whereas, in Chameleon, the directory servers (see Section 4) theoretically do not need to be more than a single node. Finally, we foresee that to fully protect against global observers, a far greater random delay than 0–100 ms, as was employed in [12], have to be incurred at each mix in the path.

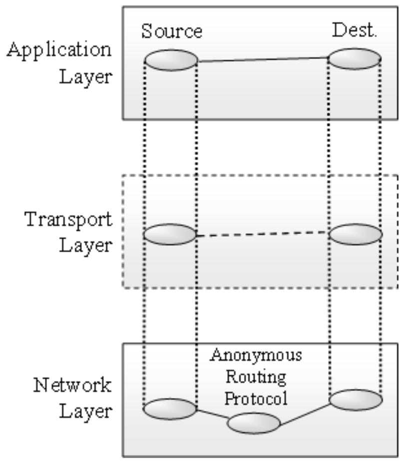

Fig. 1. Communication between a sender and recipient using an anonymous routing protocol.

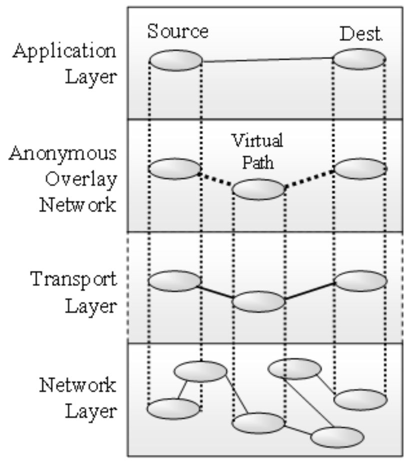

Fig. 2. Communication between a sender and recipient using an anonymous overlay network.

3 Identities in Chameleon – the *Identity-Anonymity Paradox*

In order to implement identities in Chameleon, each Chameleon node owns a set of certificates used to authenticate against other Chameleon nodes. We assume that certificates are obtained either by a side-channel, or when the nodes are in contact with the certificate authority, possibly located in a fixed network. This section discusses why digital certificates were selected as identifiers in Chameleon, and also why we consider that the most reasonable option for all anonymous communication mechanisms and also security models for mobile ad hoc networks to be proposed from now on.

By definition [9], mobile ad hoc networks *may* operate in isolation – that is, in the absence of any fixed infrastructure. Therefore, the concept of autonomous systems is not applicable in mobile ad hoc environments, as there is no entity controlling the network and providing services such as routing, security or addressing[1]. The lack of standardized addressing schemes allows network nodes to change their IP addresses (and MAC addresses as well), or even to have multiple network interfaces (either real or virtual) with multiple identifiers. Thus, obtaining unique, persistent and trustworthy identifiers from layers below application (regarding the TCP/IP model) is not realistic.

[1] There are currently no standards for IP assignment in mobile ad hoc networks. Recently, the Autoconf Internet Engineering Task Force (IETF) Working Group [2] was assigned to study, among other questions, the problem of addressing in mobile ad hoc networks.

The consequence of such fact is that traditional identification systems that rely on the usage of network or data link information are basically useless in such environments.

The lack of reliable network and data link identification might give the impression that nodes in mobile ad hoc networks are naturally anonymous, especially if we consider using the Sybil attack[2] [11] as an enabler for achieving anonymity. The Sybil attack would allow the usage of multiple identifiers simultaneously with a lifetime equivalent to the lifetime of one session or TCP connection, for instance. Therefore, both IP and MAC addresses would constantly change and, in principle, it would not be possible to associate or track those identifiers.

Although the concepts of anonymity and identities can be understood as opposites, without identities, reliable anonymity is not achievable in mobile ad hoc environments. First, because such scheme would be vulnerable to traffic analysis and positioning techniques. Furthermore senders and recipients could be easily pinpointed and their relationships exposed since both senders and receivers establish direct connections, thereby, having their anonymity properties compromised. In addition, the lack of persistent identities is harmful for the network sanity, since all security mechanism for mobile ad hoc networks would hold without some form of trustworthy identifiers. We named this need of identifiers to achieve anonymity as the *identity-anonymity paradox*.

The consequences of this paradox and its relation with the Sybil attack lead to a clear interpretation of the definition of mobile ad hoc networks in the RFC 2501 regarding the operation in isolation and a better understanding of the foundations behind the issue of identifiers in proposed security mechanisms for mobile ad hoc environments. A taxonomy of such mechanisms is presented below, where security models are classified into three families regarding the way that identifiers are generated and obtained:

i. *Intermittently connected to an established infrastructure* – security models belonging to this group assume that mobile ad hoc networks connect periodically (or at least occasionally) to an established infrastructure, such as the Internet. Therefore, it is possible to rely on the established security infrastructure that already exists in the Internet, such as a PKI (Public Key Infrastructure), and therefore, distribute digital certificates among the participants of an ad hoc network. Security schemes in this group include proposals that rely on Internet access [13] and proposals combining crypto-based techniques [5] with digital certificates;

ii. *Setting a Certificate Authority in the mobile ad hoc network* – the assumption is that one or more devices have a special role in the network, such as personal Certificates Authorities (CA) and repositories. These CA are responsible for issuing

[2] In a Sybil attack, malicious users assume multiple identities, preventing the usage of security mechanisms based on filters or trust assumptions.

certificates or credentials to devices in the mobile ad hoc networks. There are two basic approaches to set one or more CA in a mobile ad hoc network:

(a) One or more devices have a special role in the network, such as issuing certificates and publishing revocation lists, for instance. Solutions such as the Resurrecting Duckling model [22] are based on a central device that controls the network. In Martucci *et al.* [15], a security architecture is presented using multiple CA-like devices that control and secure a service-oriented ad hoc network. These solutions can operate isolated from an established infrastructure, although one or more nodes play a special role regarding security;

(b) A set of ad hoc network devices has parts of a private key that is used to issue certificates usually based on threshold cryptography. As long as a sufficient part of these nodes is the network range, digital certificates can be issued. Threshold cryptography was first proposed in the context of ad hoc networks in Zhou and Haas [27]. How many nodes and which nodes are needed to issue a certificate is usually implementation dependent;

iii. *PGP-like (Pretty Good Privacy) security models* – the assumption is that every device has one or more public/private key pairs and that every device can issue its own certificates and distribute them as well. Security often relies on the concept of web of trust. Such solutions are distributed enough to operate in complete isolation from any deployed infrastructure, however there are absolute no guarantees regarding protection against Sybil attacks, what is a major drawback of security models belonging to this family, such as the proposal of Capkun *et al.* [7] for instance.

Several conclusions can be drawn when putting the aforementioned taxonomy, the RFC 2501 definition and identity-anonymity paradox into the same picture. First, security schemes for ad hoc networks need to guarantee the uniqueness of the network identifiers, usually by the means of digital certificates. Second, the provisioning of reliable anonymous communication for nodes in a mobile ad hoc network, persistent identifiers are also needed. Third, to achieve reliable certificate distribution in ad hoc networks to prevent Sybil attacks, some sort of trusted third party (either centralized or distributed) is needed, which includes solutions from families *i* and *ii*, but not from family *iii*. Finally, regarding the RFC 2501 definition, to our understanding, a mobile ad hoc network may either depend intermittently on some deployed infrastructure (and therefore may operate in isolation for a given time frame) or it could operate in complete isolation from the deployed infrastructure, given that some support systems (a third trusted party) is deployed in the mobile ad hoc network.

Given all the aforementioned reasons, identities in Chameleon are implemented as digital certificates. The strategy for issuing and distributing identifiers depends on the security model chosen. From the point of view of the security model, Chameleon is an add-on for providing anonymous communication.

4 Chameleon: an Anonymous Overlay Network

This section introduces Chameleon. It is structured as follows. Section 4.1 outlines the Chameleon protocol, including its assumptions and basic functionalities. Section 3 discusses the need for persistent identifiers in mobile networks for the purposes of protecting against attackers assuming multiple identities. Finally, Section 4.2 further specifies message transfer, path establishment, and path repairing in Chameleon.

4.1 Protocol Basics and Assumptions

The idea of Chameleon is that one user's action is hidden within the actions of many other users. By sending messages through virtual paths, a user can participate in a communication session while at the same time hiding his identity among the identities of the other users in the mobile ad hoc network.

A virtual path functions by routing encrypted messages through chains of nodes. To protect against traffic analysis, the appearance of the messages is changed at each node in the path through encryption. Generally, there are two main strategies for constructing virtual paths for anonymous overlay networks. One approach, applied in e. g., Tor [10] and other layered encryption approaches, is to let the first node decide the whole path by wrapping a message in several layers of encryption – one for each intermediary node along the path. These layers are thereafter peeled off (by decryption), one by one, at each subsequent node on the path. In an alternative strategy, applied in e. g., Crowds, the first node decides its successor, and then the intermediate nodes decide their respective successors, until some node decides to end the path, based on some criteria, and then forwards the message to the destination.

To deal with high mobility and to enable efficient path repairing in case of disappearing nodes, Chameleon employs the same strategy for establishing virtual paths as Crowds. Therefore, during path establishment, the decision of extending the path or not depends on the result of the toss of a biased coin, which bias is determined by the "probability of forwarding" p_f, where p_f is bounded by the interval $[0.5, 1)$. With the probability $(1 - p_f)$, the path is ended and a connection is established with the destination; otherwise the path is extended to another randomly chosen node, at which the same process is repeated. The path length L is thus probabilistic and denotes the sum of the appearances for each node on the path (excluding the destination node), and $\min(L) = 2$. The expected length of L, L_{exp}, is given in equation (1) [18], where the greater the p_f, the longer the L_{exp} [3].

$$L_{exp} = (p_f)/(1 - p_f) + 2 \tag{1}$$

[3] The relationship between p_f and the resulting degree of anonymity is further elaborated in [3].

Virtual paths are bidirectional, meaning that messages can travel forward (towards the destination) or backward (towards the source). As in Crowds, the destination's IP address is known only to the nodes belonging to the path, and path rebuilding is performed in the forward direction only (to enable path rebuilding also in the backward direction, intermediary nodes would require greater knowledge about the path and, eventually, the identity of the sender). To provide better protection against local observers, link encryption is employed between the nodes in the virtual path. Unlike Crowds, conditionally on the destination type, end-to-end encryption may also be applied between the sender and destination (see Section 4.2).

Finally, Chameleon relies on the following assumptions:

i. It is expected that certificates are obtained a priori from a third trusted party, which is, most likely, located in a fixed network. Whether this assumption collides or not with the definition of mobile ad hoc networks in [9] is polemic among authors in the field. In our opinion, it is expected for a node in a mobile ad hoc network to have occasional contact with a fixed network and, therefore, to a set of trusted devices. This assumption is also present in other papers dealing with the problem of anonymity in ad hoc networks, such as [14, 26, 6];
ii. Chameleon assumes that it is possible to establish secure sessions in the transport layer, with mutual authentication using digital certificates and symmetric key establishment. Secure sessions can be achieved using standard protocols, such as TLS.
iii. Since the IP and hardware addresses are not necessarily unique identifiers that can be linked, with a long-term one-to-one relationship, to a corresponding user, we assume that the mobile ad hoc environment is a service-based network, such as Jini [16], Salutation [20], SLP (Service Location Protocol) [24] or UPnP [23] networks. Therefore, all network services, including potential anonymity services, are announced through a localization service, such as Jini's Lookup Server or UPnP's Simple Service Discovery Protocol.

4.2 Detailed Protocol Description

In the remainder of this paper, we use the following notation for describing the networks nodes in a Chameleon scenario:

i. Ψ denotes the set of nodes $\{\psi_1, \psi_2, ..., \psi_n\}$ situated in the mobile ad hoc network;
ii. Γ denotes the set of Chameleon users $\{\gamma_1, \gamma_2, ..., \gamma_n\}$, where $\Gamma \subset \Psi$. A virtual path is defined as a path connecting the sender, γ_s, with the last node before the destination, γ_{last}, where γ_s and γ_{last} are interconnected by zero or more nodes from Γ. When we describe the protocol, γ_i denotes the current node. The cardinality of Γ is denoted $|\Gamma|$, and $min(|\Gamma| = 3)$, since this is the minimum amount of members

needed to provide some level of anonymity against the attacker model presented in Section ;

iii. D denotes the destination, which can be classified in three disjoint sets: $D_{\overline{sec}}$ accepts only unencrypted requests; D_{sec} accepts secure requests using a standard secure transport protocol between γ_{last} and D, and; D_{Γ} understands Chameleon protocol messages, enabling end-to-end encryption between γ_s and D;

iv. $\Phi \subset \Gamma$ denotes a set of decentralized directory servers $\{\phi_1, \phi_2, ..., \phi_n\}$ announcing the set of network addresses of the nodes in Γ, IP_{Γ}, along with their digital certificates, to other nodes in Γ. To reveal as little as possible information to Φ, each node in Γ requests IP_{Γ} at regular time intervals. The restriction $\Phi \subset \Gamma$ decreases the likelihood of corrupted directory servers announcing false information, since they can be detected as malicious nodes and filtered out by other Chameleon users. The announcement of IP_{Γ} follows one of the main principles of zero configuration networking [25], which assumes the existence of a service discovery system in network environments such as mobile ad hoc networks. The nodes in Φ act as a distributed version of the blender in Crowds.

The following notation is used for the messages types in Chameleon:

i. θ denote application data passed to Chameleon from the application layer;

ii. m_{γ_i,γ_j} denote messages passed between Chameleon nodes γ_i and γ_j via the lower layers. The messages m_{γ_i,γ_j} are link encrypted between γ_i and γ_j using the symmetric key $E_{k_{\gamma_i,\gamma_j}}$ (established using a secure transport layer protocol). For the cases where $D \in D_{sec}$ or $D \in D_{\overline{sec}}$, the payload of m_{γ_i,γ_j} includes: IP_D – the IP address of D; $p\#_{\gamma_i,\gamma_j}$ – a path identifier (a randomly generated integer for identifying packet streams between nodes γ_i and γ_j); and the data payload θ – see equation (2), where $\cdot$ denotes concatenation. For the case where $D \in D_{\Gamma}$, m_{γ_i,γ_j} has two optional fields to achieve end-to-end encryption and data integrity – see equation (3). The first field contains a symmetric key $k_{\gamma_s,D}$, which is encrypted with the D's public key, Pu_D. The symmetric key $k_{\gamma_s,D}$ is used to set an end-to-end secure channel between γ_s and D. The second field is used to send the output of a keyed-hash function for message integrity, with input data θ and key $k_{\gamma_s,D}$;

$$m_{\gamma_i,\gamma_j} = E_{k_{\gamma_i,\gamma_j}}[p_{\#\gamma_i,\gamma_j} \cdot IP_D \cdot \theta] \tag{2}$$

$$m_{\gamma_i,\gamma_j} = E_{k_{\gamma_i,\gamma_j}}[p_{\#\gamma_i,\gamma_j} \cdot IP_D \cdot E_{k_{\gamma_s D}}[\theta] \cdot E_{Pu_D}[k_{\gamma_s D}] \cdot hash_{k_{\gamma_s,D}}(\theta)] \tag{3}$$

iii. An acknowledgment message is generated in γ_{last} and sent towards γ_s to inform that a message has reached its destination. Equation (4) describes the $ack\gamma_{i+1}, \gamma_i$ acknowledgement message sent from γ_{i+1} to γ_i.

$$ack_{\gamma_{i+1},\gamma_i} = E_{k_{\gamma_{i+1},\gamma_i}}[p_{\#\gamma_{i+1},\gamma_i}] \tag{4}$$

IP_D	$IP\gamma_{i-1}$	$p_{\#\gamma i-1,\gamma i}$	$IP\gamma_{i+1}$	$p_{\#\gamma i,\gamma i+1}$	TTL

Fig. 3. An entry in the Chameleon routing table.

Each node in Chameleon maintains a routing table with the following entries (see Figure 3): the destination's IP address (IP_D); the backward and forward path identifiers ($p_{\#\gamma_{i-1},\gamma_i}$ and $p_{\#\gamma_i,\gamma_{i+1}}$); the address of the preceding and succeeding nodes in the virtual path ($IP_{\gamma_{i-1}}$ and $IP_{\gamma_{i+1}}$) and; the time-to-live (TTL) counter, a decremental counter indicating the remaining lifetime of a given entry in the table. The path identifiers are managed in the same way as the *path_id* in Crowds [18]. In Chameleon, the tuple $[IP_{\gamma_i}, IP_{\gamma_{i+1}}, p_{\#\gamma_i,\gamma_{i+1}}]$ identifies a path connection between two nodes γ_i and γ_{i+1}.

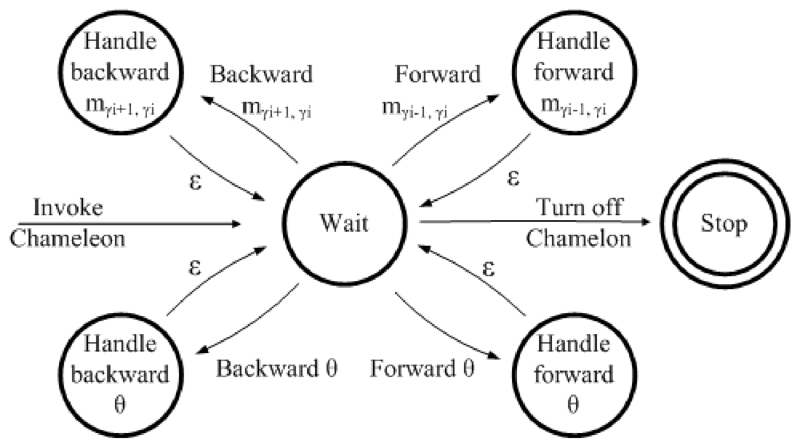

Fig. 4. The Chameleon main state transition diagram for each node in Chameleon. A node can play the roles of γ_s, γ_i, or γ_{last}, depending on the type the incoming message.

A Chameleon node can be described as a local proxy server following the state transition diagram in Figure 4[4]. Its role is threefold; first, it may serve as the user's local proxy to which the user's applications forward their data, θ. In this case the

[4] In a coming implementation, we plan to implement parallelism to enable Chameleon to serve multiple messages at the same time. For clarity reasons, we omit this feature in the current state transition diagrams

node constitute the first node on the virtual path, γ_s. This situation is represented by the "Handle forward θ" state in Figure 4, which in turn can be expanded to the diagram in Figure 5. In the second case, a node can be an intermediary peer in one or more virtual paths. This situation is represented by the "Handle forward $m_{\gamma_{i-1},\gamma_i}$" and "Handle backward $m_{\gamma_{i+1},\gamma_i}$" state in Figure 4, which in turn can be expanded to either of the diagrams in Figure 6 or 8, depending on the message direction. Finally, a node can act as the last peer in a virtual path, γ_{last}. In this case, it acts as a proxy server towards *D*. The diagram in Figure 7 (representing the expansion of the "Handle backward θ" state in Figure 4) depicts this case.

In the remainder of this section, we key out the protocol details by (1) describing virtual path establishment, (2) describing how data is sent from γ_s to *D*, and, (3) describing how virtual paths are repaired in the event of a path break.

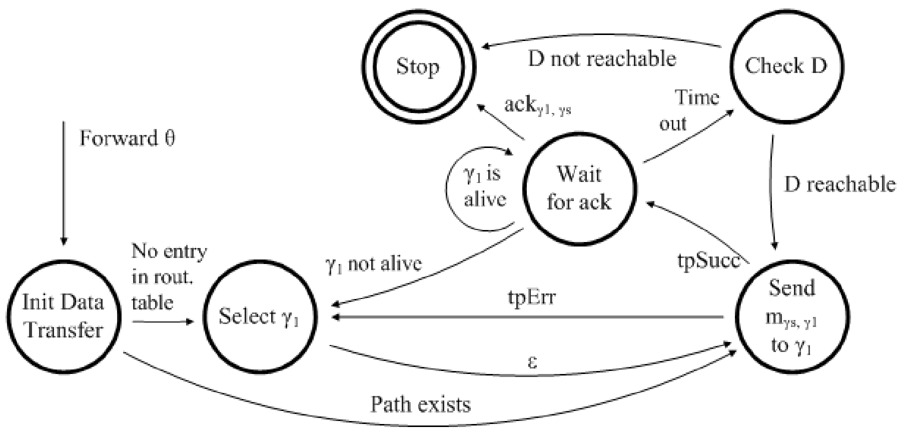

Fig. 5. State transition diagram for a node γ_s receiving data from the application layer. The acronyms *tpSucc* and *tpErr*, used in this section, denote transitions indicating whether the sending of a message was accomplished successfully (*tpSucc*) or not (*tpErr*) in the transport layer.

A. *Building virtual paths.* In Chameleon, the virtual paths are constructed as follows, assuming that there is no entry in the routing table for the designated destination address, IP_D:

(i) Path establishment is initiated when a node γ_s receives θ from the application layer. Then, γ_s randomly selects[5] a node γ_1 from Γ, as visualized in the

[5] If γ_s possesses no recent information about Γ, it contacts a directory server ϕ_i and requests this information. The nodes γ_s and ϕ_i mutually authenticate using their certificates.

"Select γ_1" state in Figure 5. Then, γ_s and γ_1 establish a secure session in the transport layer, exchanging a symmetric key k_{γ_s,γ_1} for link encryption. The sender γ_s then assembles and encrypts m_{γ_s,γ_1} (in which θ is piggy-backed) and forwards m_{γ_s,γ_1} to γ_1 ("Send m_{γ_s,γ_1} to γ_1" state in Figure 5). In cases when γ_s cannot send m_{γ_s,γ_1} to γ_1, it selects another new random node γ_1 from Γ and repeats the process;

(ii) Now, γ_i (i. e., $i = 1$), triggers the state transition diagram in Figure 6, and starts by decrypting $m_{\gamma_{i-1},\gamma_i}$. Assuming there is no corresponding entry for $m_{\gamma_{i-1},\gamma_i}$ in the Chameleon routing table of γ_i, a biased coin is tossed ("Toss biased coin" state in Figure 6). If the decision of the coin toss is to end the path, θ (encapsulated in $m_{\gamma_{i-1},\gamma_i}$) is forwarded to D. In this case, γ_i becomes the last node in the virtual path, γ_{last}. Otherwise, the path is extended one hop and a new node γ_{i+1} is selected randomly from Γ. The message $m_{\gamma_i,\gamma_{i+1}}$ is then encrypted and forwarded to γ_{i+1}, where this process is repeated. Eventually, a path will be established between γ_s and γ_{last}, where γ_s and γ_{last} are interconnected by zero or more intermediary Chameleon nodes.

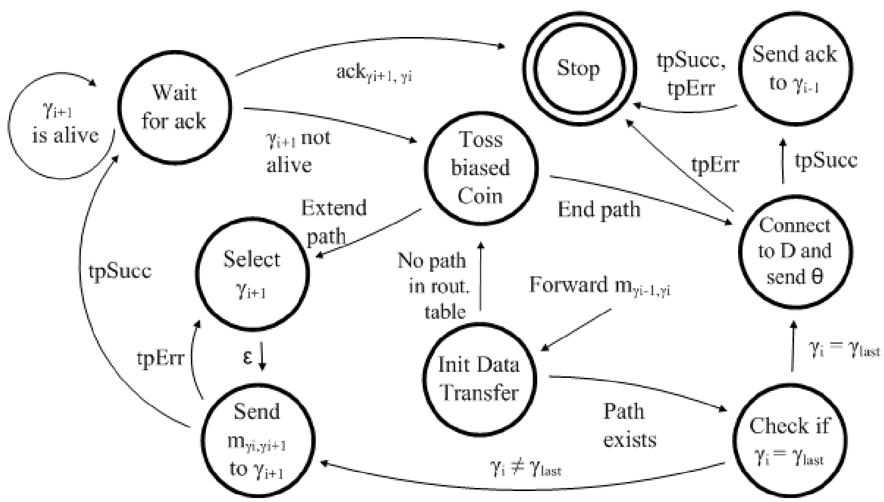

Fig. 6. State transition diagram for a node γ_i receiving a message $m_{\gamma_{i-1},\gamma_i}$, including path repairing.

B. *Sending and forwarding data*. In Chameleon, data is passed from γ_s to D in the following way, assuming that a virtual path is already established:

(i) When γ_s receives θ from an application, γ_s assembles and encrypts m_{γ_s,γ_1}, and sends it to γ_1, as depicted in the "Send Message m_{γ_s,γ_1} to γ_1" state in Figure 5;

(ii) Regarding the intermediary nodes, an incoming $m_{\gamma_{i-1},\gamma_i}$ is treated according to the state transition diagram depicted in Figure 6. At each node, $m_{\gamma_{i-1},\gamma_i}$ is decrypted, and $m_{\gamma_1,\gamma_{i+1}}$ is generated and encrypted before being forwarded. Eventually, the last node on the path, γ_{last}, will receive $m_{\gamma_{last-1},\gamma_{last}}$. Then, γ_{last} sends θ to D (either encrypted or unencrypted, depending on the destination type, see Section 4.2). Provided that the connection with D was successful, $ack_{\gamma_{last},\gamma_{last-1}}$ is sent backwards along the path to acknowledge γ_s that D did receive θ;

(iii) The sending of data in the backward direction is initiated when γ_{last} receives θ from D (see Figure 7). Then, γ_{last} encapsulates θ in $m_{\gamma_{last},\gamma_{last-1}}$ and sends it to γ_{last-1} on the virtual path. Since messages traveling in the backward direction are not acknowledged, the state transition diagram in Figure 7 always goes to the "Stop" state, independent of whether or not it was possible to send the message to γ_{last-1}. This process is repeated at each intermediary node until the message eventually reaches γ_s (see Figure 8). If a timeout threshold is exceeded, the "Check D" state is invoked (see Figure 5), where γ_s checks the status of D (this is possible since the ad hoc network is a service-based network). The timeout should be large enough to allow intermediary nodes to conduct path repairing, but, on the other hand, not too large, since this would risk to compromise the protocol performance.

C. *Repairing virtual paths.* Path repairing is initiated in two situations: first, when γ_i fails to send $m_{\gamma_i,\gamma_{i+1}}$ to γ_{i+1}, and, second, when γ_i waits for $ack_{\gamma_{i+1},\gamma_i}$ and notices that γ_{i+1} is not alive (γ_i polls γ_{i+1} at regular intervals during the "Wait for $ack_{\gamma_{i+1},\gamma_i}$" state to assert that γ_{i+1} is still alive, as illustrated in Figures 5 and 6). The node γ_i tosses a biased coin and either forwards θ directly to D or selects a new node γ_{i+1} as its successor in the path. In this way, the path is restored from the point where it was broken, and not from the beginning. No explicit path destruction is conducted after the communication session via the virtual paths has ended. Instead, the TTL field in the routing table (see Figure 3) ensures that inactive path entries are deleted.

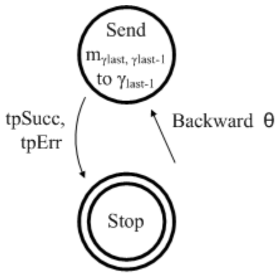

Fig. 7. Chameleon backward data θ state transition diagram for γ_{last}.

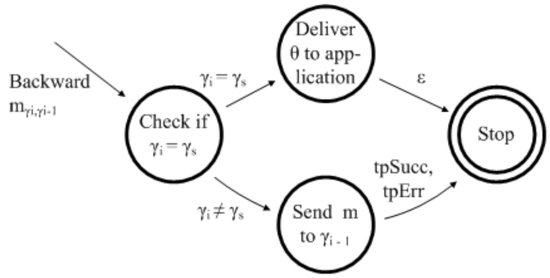

Fig. 8. Chameleon backward $m_{\gamma_i,\gamma_{i-1}}$ state transition diagram for γ_i.

5 Theoretical Analysis

Six different requirements were defined in [4] which an anonymous overlay network should adhere to (at least to an acceptable degree[6]) in order to be suitable in mobile ad hoc network environments. Below, we list these requirements, and briefly discuss to what extent Chameleon meets these requirements:

1. *Scalability*: the workload on each participant in Chameleon remains virtually constant as the number of participants grows, as in Crowds [18]. It is proved in [18] that for each node in the network, the expected number of virtual paths a node will be appearing on at a particular time is given by: $\frac{1}{(1-p_f)^2} * (1 + \frac{1}{n})$, where n is the number of Crowds users. This equation holds for Chameleon as well, when substituting n for $|\Gamma|$;
2. *Strong anonymity properties*: an anonymous overlay network should provide adequate protection against, for instance, malicious users and different types of observers. Chameleon offers sender and relationship anonymity against local observers. Unlike Crowds, Chameleon enables both link-to-link and end-to-end encryption for certain destination types on the overlay layer. However, due to performance reasons Chameleon does not protect against a global observer. The anonymity properties of Chameleon are further analyzed in Section 5.2;
3. *Fair distribution of work*: an anonymous overlay network should be fair regarding the distribution of workload among the participants. A possible source for unfairness in Chameleon is the workload implied for the operators of the directory

[6] The requirements are not orthogonal. We foresee trade-offs, e. g., between anonymity and performance, when designing new anonymous overlay networks for mobile ad hoc networks.

servers Φ. We plan to try to remedy this unfairness by making the allocation of the directory servers dynamic. An alternate option, that would obsolete the directory servers, is to force the nodes in Γ to announce their presence by controlled flooding. However, this would increase the rate of control messages in the protocol;

4. *Performance-wise lightweight solution*: in order to reduce computational overhead and increase battery lifetime, an anonymous overlay network should generate few messages and perform few public key operations. Chameleon uses public key encryption sparsely and avoids layered encryption. The protocol overhead is low; assuming knowledge about Γ, $2L$ public key operations and $2L-1$ Chameleon messages are needed to establish a path, where L denotes the path length. In comparison, MorphMix [19] generates $6L+(L-2)(L+1)$ messages and needs at least $13L$ public key operations when establishing a path. Additionally, in contrast to Chameleon, the earlier mentioned mix-based proposal by Jiang *et al.* [12] uses nested public key encryption for both path establishment and message transfer. Lastly, no performance consuming dummy traffic is used, as Chameleon does not protect against global observers[7];
5. *Adherence to the P2P-model*: mobile ad hoc networks are most often assumed to function without the aid of central hardware and services [9]. Unlike e. g., Crowds, Chameleon is a fully P2P-based protocol, although all nodes in Γ need to agree on the value of p_f;
6. *Manage a dynamic topology*: in most proposed mobile ad hoc network scenarios, it is assumed that nodes frequently enter and leave the network. Chameleon addresses dynamic topologies by, among other things, an optimized path repairing process in the forward direction. A virtual path is repaired only from the point of breach (see Figure 6), in contrast to other approaches, such as MorphMix [19], that rebuild a broken path entirely from scratch.

5.1 Attacker Model of Chameleon

The attacker model of Chameleon assumes all nodes, including the attackers, to have the same radio range. The following types of attackers are included in the attacker model:

1. *Local observer* ($\psi_{obs} \in \Psi$): this is a passive observer whose radio range covers γ_s;
2. *Malicious insiders* ($\Gamma' \subset \Gamma$): this attacker is represented by $|\Gamma'|$ (collaborating) malicious members of Γ, aiming to occupy all positions on the virtual path (except, obviously, the position of γ_s);

[7] It is commonly believed that omnipresent protection against a global observer (i. e., during periods of both high and low traffic) can only be achieved if all nodes transmit a constant flow of traffic, requiring the usage of dummy traffic.

3. *Malicious outsider* ($\psi' \in \Psi$): this attacker is represented by a malicious node aiming to control an intermediary node linking a pair of Chameleon nodes in a given virtual path;
4. *Destination* (D): this attacker attempts to disclose the identity of γ_s;
5. *Malicious directory servers* ($\phi' \subset \Phi$): these constitute attackers hosting the directory service for the purposes of collecting and misusing information about the members of Γ, or helping other attackers, such as malicious insiders, by for example only submitting the addresses of compromised nodes.

5.2 Anonymity Analysis of Chameleon

The metric applied in this section is based on the metric applied for evaluating the anonymity properties of Crowds [18]. In this metric, each user is considered separately, and the resulting value spectra is a function of (among other parameters) the size of the anonymity set and the amount of malicious insiders. The degree of anonymity for a subject γ_i can be expressed as $A_{\gamma_i} = 1 - P_{\gamma_i}$, where P_{γ_i} is the probability that γ_i is the originator of a particular message. A_{γ_i} is measured on a continuous scale ranging from `absolute privacy` to `provably exposed` (see Figure 9), including the following intermediary points of interest:

– `Absolute privacy:` the probability that a given subject γ_i is linked to a particular message is zero, and, hence, $A_{\gamma_i} = 1$;
– `Beyond suspicion`: a subject γ_i in the anonymity set $\{\gamma_1, \gamma_2, ..., \gamma_i, ..., \gamma_n\}$ is `beyond suspicion` if it appears no more likely than any other subject in the anonymity set of being linked to a particular message, that is, $A_{\gamma_i} = \min\{A_{\gamma_1}, A_{\gamma_2}, ..., A_{\gamma_i}, ..., A_{\gamma_n}\}$;
– `Probable innocence`: the probability that a given subject γ_i is linked to a particular message is less than $\frac{1}{2}$, and, thus, $A_{\gamma_i} \geq \frac{1}{2}$;
– `Possible innocence`: there is a non-trivial chance that a particular subject γ_i is not the originator of a given message ($A_{\gamma_i} > \nabla_{limit}$, where $0 < \nabla_{limit} < \frac{1}{2}$);
– `Exposed:` a given subject γ_i can be unambiguously linked to a given message, and, hence, $A_{\gamma_1} = 0$;
– `Provably exposed:` $A_{\gamma_1} = 0$ as above and, furthermore, it could be proved to a third party that the subject γ_i is linked to the given message.

Below follows an analysis of the offered degree of anonymity for Chameleon users against the attacker model defined in Section 5.1:

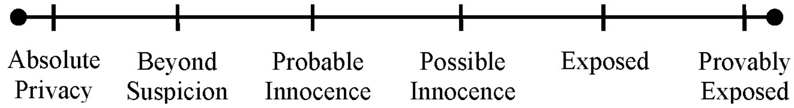

Fig. 9. Degrees of anonymity in the Crowds-based anonymity metric [18].

A. Anonymity against a *local observer* (ψ_{obs}):

(i) *Sender anonymity*: since ψ_{obs} is within γ_s's radio range, ψ_{obs} can observe all messages emanating from γ_s. However, except during periods of low traffic, ψ_{obs} cannot tell whether γ_s was the originator of these messages or not, as γ_s could instead be forwarding another node's messages. ψ_{obs} will further be incapable of recognizing earlier observed traffic flows reappearing inside its radio range, since every message is link encrypted between each pair of Chameleon nodes. In periods of low traffic, however, there is a nontrivial risk that ψ_{obs} may suspect that γ_s is indeed the originator of the observed messages, e. g., by using traffic analysis. Still, ψ_{obs} cannot know for certain whether γ_s constitutes the origin sender, as this node might be communicating with a "hidden terminal". The hidden terminal problem is a notorious problem in wireless networks, see Figure 10. Thus, the degree of sender anonymity amounts to `possible innocence`;

(ii) *Receiver anonymity*: to break receiver anonymity, ψ_{obs} must be within the radio range of D and γ_{last}. In this case, ψ_{obs} may conclude that a given message is intended for a given D. However, the larger the network, the less the likelihood of D and γ_{last} being subsumed by the radio range of ψ_{obs}. Thus, the degree of receiver anonymity approaches `beyond suspicion` for networks where the physical size of the network is larger than the radio range of the attacker, which is a reasonable assumption given our attacker model;

(iii) *Relationship anonymity*: except for the special case when the radio range of ψ_{obs} contains the full virtual path, ψ_{obs} cannot link γ_s to D, since ψ_{obs}'s network view is incomplete and the messages' appearances change between the nodes. For large networks, the degree of relationship anonymity amounts to `beyond suspicion`.

B. Anonymity against $|\Gamma'|$ *malicious insiders*:

(i) *Sender anonymity*: due to the probabilistic nature of the path construction, a malicious insider $\gamma_i \in \Gamma'$ on a given virtual path cannot tell for sure whether the previous node γ_{i-1} is γ_s, or not. The situation for the malicious insiders in Chameleon is similar to that of "collaborative jondos" in Crowds (see [18]). Thus, the degree of sender anonymity is `probable innocence`, provided that equation (5) [18] holds. Here, it can be noted that the greater the p_f and the larger the $|\Gamma|$, the more malicious insiders can be tolerated. It can further be noted that although not affecting the degrees of anonymity *per se*, the certificate-based protection against Sybil attacks (see Section 3) makes it more costly for malicious insiders to take control of a sufficiently large portion of the network to break equation (5).

$$|\Gamma| \geq \frac{p_f}{(p_f - \frac{1}{2})} * (|\Gamma'| + 1) \tag{5}$$

(ii) *Receiver anonymity*: a malicious insider on the virtual path with a given γ_s will always learn IP_D, since it is encapsulated in $m_{\gamma_i,\gamma_{i+1}}$. In these cases, the degree of anonymity is `exposed`. On the other hand, if none of the $|\Gamma'|$ malicious insiders are part of the virtual path, the degree is `absolute privacy`. The probability that none of the $|\Gamma'|$ malicious insiders are part of a particular path (and, thus, that the degree of receiver anonymity is `absolute privacy`) is given by:

$$P(\texttt{absolute privacy}) = \left(\frac{|\Gamma| - |\Gamma'|}{|\Gamma|}\right)^{L_{exp}-1} = 1 - P(\texttt{exposed}) \tag{6}$$

(iii) *Relationship anonymity*: a malicious insider can only break the properties of relationship anonymity by breaking the properties of sender anonymity (since this attacker knows D). Thus, the degree of relationship anonymity is `probable innocence` provided that equation (5) holds.

C. Anonymity against a *malicious outsider* ($\psi' \in \Psi$):

(i) *Sender anonymity*: we start by defining the following events:

– E_{route} denotes the event that a malicious outsider $\psi' \in \Psi$ is selected, on the lower layers, to route a message between γ_i and γ_j. The probability of

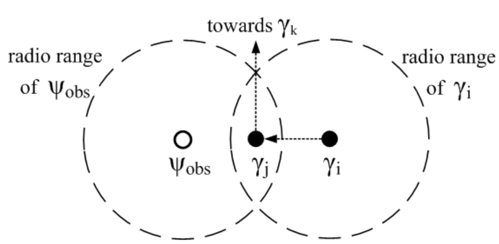

Fig. 10. The hidden terminal problem. Here, ψ_{obs} cannot determine for sure whether γ_j is the origin sender or is forwarding a message from another node γ_i outside his radio range.

E_{route} occurring is likely to be low, since ψ' needs to possess information about the physical locations of γ_i and γ_j, as well as their radio ranges, to be used as an intermediary routing link between γ_i and γ_j. Alternatively, ψ' could misuse the underlying routing protocol to deceive γ_i and γ_j so that it appears that ψ' constitute an intermediary path between γ_i and γ_j;

– E_{dir} denotes the event that ψ' can conclude that γ_i precedes γ_j in the path. The attacker ψ' may suspect that the first routed m_{γ_i,γ_j} determines which node is preceding the other. However, due to the expected mobile behavior of the nodes in a mobile ad hoc network, ψ' cannot exclude the possibility that the first observed m_{γ_i,γ_j} was preceded by a number of other messages, routed either directly between γ_i and γ_j, or via another node;
– Finally, $E_{\gamma_i=\gamma_s}$ denotes the event that $\gamma_i = \gamma_s$.

Although the probability of $(E_{route} \wedge E_{dir})$ occurring is likely to be low, we nonetheless assume these events to find a lower bound for the degree of sender anonymity. In this case, we can express the sought probability of $E_{\gamma_i=\gamma_s}$ occurring given the event $(E_{route} \wedge E_{dir})$ as the inverse of the expected number of hops, since the attacker could be situated in either of the hops between two Chameleon nodes.

$$P(E_{\gamma_i=\gamma_s} \mid E_{route} \wedge E_{dir}) = \frac{1}{H_{exp}} = \frac{1}{(L_{exp} - 1) - R_L} \qquad (7)$$

In Equation (7), H_{exp} denotes the expected number of *hops* (i. e., the number of virtual links between the nodes). Using *a priori* knowledge, an attacker ψ' can only guess that he is situated on the right hop with the probability given by $\frac{1}{H_{exp}}$, since ψ' could be situated on any of the expected number of hops (see Figure 11 for an illustration of an attacker ψ' routing messages between γ_s and γ_j). R_L denotes the expected reduction in the actual number of hops due to *local loops*: a local loop occurs if a node selects itself as its successor, see Figure 11.

Since $A_{\gamma_i} = 1 - P_{\gamma_i}$ according to the Crowds metric, $1 - P(E_{\gamma_i=\gamma_s} \mid E_{route} \wedge E_{dir})$ denotes the amount of sender anonymity against a malicious outsider. In Appendix A, we prove that for $L_{exp} \geq 4$ and $|\Gamma| \geq 3$, the expected number of hops is always greater than two ($H_{exp} > 2$), meaning that the attacker always must expect that there is at least two different hops he could be situated on. Thus, according to Equation (7), the degree of anonymity is `probable innocence`. According to Equation (1), $L_{exp} \geq 4$ can be achieved if $p_f \geq \frac{2}{3}$ is chosen. For large values of $|\Gamma|$, the actual degree is more likely to approach `beyond suspicion`. Furthermore, it is not for certain that the event $(E_{route} \wedge E_{dir})$ will occur in the first place;

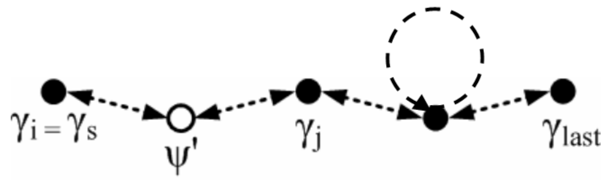

Fig. 11. An illustration of ($E_{route} \wedge E_{dir} \wedge E_{\gamma_i=\gamma_s}$) including a local loop.

(ii) *Receiver anonymity*: ψ' cannot learn IP_D directly, since m_{γ_i,γ_j} is link encrypted between γ_i and γ_j. Using an analogous reasoning as above, the degree of receiver anonymity can be shown to be `probable innocence` in the worst case for all $L_{exp} > 4$;

(iii) *Relationship anonymity*: since the protocol assures that γ_s will never communicate directly with D, the degree of relationship anonymity is `beyond suspicion`.

D. Anonymity against a *destination* (D): from the perspective of D, γ_s could be any node $\gamma_i \in \Gamma$, since $L \geq 2$. For this reason, both the degrees of sender and relationship anonymity are `beyond suspicion`.

E. *Anonymity against malicious directory servers* ($\phi' \subset \Phi$): although ϕ' possesses information about all IP addresses $\in \Gamma$, it cannot use this information, as such, to break any anonymity property. Therefore, the degrees of anonymity against malicious directory servers are `absolute privacy`. However, the malicious directory servers could still help other attackers (especially malicious insiders), to succeed with their attacks by announcing false information to the users of Chameleon. For example, a malicious directory server could announce a set Γ' only containing compromised nodes. The specification and evaluation of a secure and efficient mechanism that hinders malicious directory servers from performing such *partitioning attacks* is left as future research, but such a mechanisms will probably be comprised of one or more of the following strategies:
   - *Redundancy:* the more the directory servers in Φ, the stronger the protection against malicious directory servers, since the probability that a user chooses a non-malicious directory server increases with a growing $|\Phi|$;
   - *Distributed reputation metrics:* this relates to mechanisms that assign trust values to the nodes in Φ, so that misbehaving directory servers could be found a filtered out. A trust-based service discovery protocols that suits Chameleon is described in [15]. In this proposal, certificates tailored to include trust information are employed for device authentication;

– *Cycling through the directory servers:* always using the same directory server for obtaining Γ should be avoided. Instead, the Chameleon users should use different directory servers so that, for instance, users could be alarmed when the receive two instances of Γ that differ significantly.

In Table 1, the offered degrees of anonymity in Chameleon are summarized.

Table 1. Degrees of anonymity in Chameleon.

	Sender Anonymity	*Receiver Anonymity*	*Relationship Anonymity*
Local observer (ψ_{obs})	possible innocence	beyond suspicion (for large networks)	beyond suspicion (for large networks)
Malicious insiders (Γ')	probable innocence if $\lvert\Gamma\rvert \geq \frac{p_f}{(p_f - \frac{1}{2})} * (\lvert\Gamma'\rvert + 1)$	P(absolute privacy) = $\left(\frac{\lvert\Gamma\rvert - \lvert\Gamma'\rvert}{\lvert\Gamma\rvert}\right)^{L_{exp}-1}$	probable innocence
Malicious out-sider (ψ')	probable innocence if $L_{exp} \geq 4$ and $\lvert\Gamma\rvert \geq 3$	probable innocence if $L_{exp} \geq 4$ and $\lvert\Gamma\rvert \geq 3$	beyond suspicion
Destination	beyond suspicion for $\lvert\Gamma\rvert \geq 3$	–	beyond suspicion

6 Conclusions

This paper introduced Chameleon, a low-latency anonymous overlay network tailored for mobile ad hoc networks, providing, for instance, efficient path repairing, and a reduced amount of control messages in comparison to other anonymous overlay networks. In the paper, we emphasized that in order to provide anonymity and security in mobile ad hoc networks in the first place, there is a need for persistent identifiers. Based on this, we advocated for the use of certificates to protect against Sybil attacks. Moreover, the protocol was specified with the help of state transitions diagrams. Chameleon was specially designed to minimize the effects caused by user mobility and vanishing nodes, and consequently, to minimize the power demanded. To achieve that, Chameleon does not rely on dummy traffic or layered encryption. The usage of layered (i. e, nested) encryption, for instance, demands a total reconstruction of the anonymous path, since it not allows path rebuilding from the point of rupture only.

Chameleon is inspired by the Crowds system, although it differs from Crowds in a number of ways, including: end-to-end encryption between the sender and recipient, certificate-based protection against Sybil attacks, and a distributed service discovery mechanism replacing the role of the blender. In this paper, we also defined an attacker model and analyzed the anonymity properties of Chameleon, which differs from the one of Crowds in many aspects. Furthermore, the attacker model considered for Chameleon is also more complete and suitable for ad hoc network environments than the one used in Crowds. In particular, Chameleon offers sender anonymity against destinations as well as receiver and relationship anonymity against local observers for large networks. Current research plans include analyzing protocol performance by the means of simulation.

References

1. *Proceedings of the 4th ACM International Symposium on Mobile Ad Hoc Networking and Computing (MOBIHOC'03)*, New York, NY, USA, 1–3 Jun 2003. ACM Press.
2. IETF Ad Hoc Network Autoconfiguration Working Group. Ad Hoc Network Autoconfiguration (autoconf), 2006. See http://www3.ietf.org/html.charters/autoconf-charter.html.
3. Christer Andersson, Reine Lundin, and Simone Fischer-Hübner. Privacy Enhanced WAP Browsing with mCrowds: Anonymity Properties and Performance Evaluation of the mCrowds System. In *Proceedings of the ISSA 2004 Enabling Tomorrow Conference*, Gallagher Estate, Midrand, South Africa, 30 Jun – 2 Jul 2004.
4. Anonymized. Requirements for Privacy-Enhancements for Mobile Ad Hoc Networks. In 3rd *German Workshop on Ad Hoc Networks (WMAN 2005), Proceedings of INFORMATIK 2005 - Informatik LIVE! Band 2*, volume 68 of *LNI*, pages 344–348. GI, 19–22 Sep 2005.
5. Tuomas Aura. Cryptographically Generated Addresses (cga). RFC-3972, Mar 2005. See http://www.ietf.org/rfc/rfc3972.txt.
6. Azzedine Boukerche, Khalil El-Khatib, Li Xu, and Larry Korba. SDAR: A Secure Distributed Anonymous Routing Protocol for Wireless and Mobile Ad Hoc Networks. In *Proceedings of the 29th Annual IEEE International Conference on Local Computer Networks (LCN'04)*, pages 618–624, 2004.
7. Srdjan Capkun, Jean-Pierre Hubaux, and Levente Buttyàn. Mobility Helps Security in Ad Hoc Networks. In *Proceedings of the 4th ACM International Symposium on Mobile Ad Hoc Networking and Computing (MOBIHOC'03)* [1], pages 46–56.
8. David Chaum. Untraceable Electronic Mail, Return Addresses and Digital Pseudonyms. *Communication of the ACM*, 24(2):84–88, Feb 1981.
9. M. Scott Corson and Joseph Macker. Mobile Ad Hoc Networking (MANET): Routing Protocol Performance Issues and Evaluation Considerations. RFC-2501, Jan 1999. See http://www.ietf.org/rfc/rfc2501.txt.
10. Roger Dingledine, Nick Mathewson, and Paul Syverson. Tor: The Second-Generation Onion Router. In *Proceedings of the 13th USENIX Security Symposium*, Aug 2004.
11. John R. Douceur. The Sybil Attack. In P. Druschel, F. Kaashoek, and A. Rowstron, editors, *Peer-to-Peer Systems: Proceedings of the 1st International Peer-to-Peer Systems Workshop (IPTPS)*, volume 2429, pages 251–260. Springer-Verlag, 7–8 Mar 2002.

12. Shu Jiang, Nitin H. Vaidya, and Wei Zhao. A Mix Route Algorithm for Mix-net in Wireless Mobile Ad Hoc Networks. In *Proceedings of the 1^{st} IEEE International Conference on Mobile Ad Hoc and Sensor Systems (MASS2004)*, 24–27 Oct 2004.
13. Frank Kargl, Stefan Schlott, and Michael Weber. Identification in Ad Hoc Networks. In *Proceedings of the $39^{t}h$ Hawaiian International Conference on System Sciences (HICSS-39)*, Kauai, HI, USA, 4–7 Jan 2006. IEEE Computer Society.
14. Jeijun Kong and Xiaoyan Hong. ANODR: ANonymous On Demand Routing with Untraceable Routes for Mobile Ad-hoc Networks. In *Proceedings of the 4^{th} ACM International Symposium on Mobile Ad Hoc Networking and Computing (MOBIHOC'03)* [1], pages 291–302.
15. Leonardo A. Martucci, Christiane M. Schweitzer, Yeda R. Venturini, Tereza C. M. B. Carvalho, and Wilson V. Ruggiero. A Trust-Based Security Architecture for Small and Medium-Sized Mobile Ad Hoc Networks. In *Proceedings of the 3^{rd} Annual Mediterranean Ad Hoc Networking Workshop, Med-Hoc-Net*, pages 278–290, Jun 2004.
16. SUN Microsystems. The Jini Architecture Specification – Version 1.2, 2001. See http://www.sun.com/software/jini/specs/.
17. Andreas Pfitzmann and Marit Hansen. Anonymity, Unlinkability, Unobservability, Pseudonymity, and Identity Management - A Consolidated Proposal for Terminology v0.27, 20 Feb 2006. Also available as http://dud.inf.tu-dresden.de/literatur/.
18. Michael Reiter and Avi Rubin. Crowds: Anonymity for Web Transactions. In *DIMACS Technical report*, pages 97–115, 1997.
19. Marc Rennhard and Bernhard Plattner. Introducing Morphmix: Peer-to-Peer based Anonymous Internet Usage with Collusion Detection. In *Proceedings of the Workshop on Privacy in Electronic Society (WPES02)*, 21 Nov 2002.
20. The Salutation Consortium. Salutation Architecture Specification (Part 1), Version 2.1a, 1999. See http://www.salutation.org.
21. Stefaan Seys and Bart Preneel. ARM: Anonymous Routing Protocol for Mobile Ad Hoc Networks. In *International Workshop on Pervasive Computing and Ad Hoc Communications (PCAC06), Proceedings of the 20^{th} IEEE International Conference on Advanced Information Networking and Applications (AINA 2006)*, 18–19 Apr 2006.
22. Frank Stajano and Ross Anderson. The Resurrecting Duckling: Security Issues for Ad Hoc Wireless Networks. In *Proceedings of the 3^{rd} AT&T Software Symposium*, Oct 1999.
23. UPnP Forum. UPnP Device Architecture, Version 1.0, Jun 2000. See http://www.upnp.org/download/UPnPDA10_20000613.htm.
24. John Veizades, Erik Guttman, Charles E. Perkins, and Scott Kaplan. Service Location Protocol. RFC-2165, Jun 1997. See http://www.ietf.org/rfc/rfc2165.txt.
25. IETF Zero Configuration Networking Working Group. Zero Configuration Networking (zeroconf), 2003. See http://www.zeroconf.org/zeroconf-charter.html.
26. Yanchao Zhang, Wei Liu, and Wenjing Lou. Anonymous Communication in Mobile Ad Hoc Networks. In *Proceedings of the* 24^{th} *Annual Joint Conference of the IEEE Communication Society (INFOCOM 2005)*, Miami, FL, USA, 13–17 Mar 2005.
27. Lidong Zhou and Zygmunt J. Haas. Securing Ad Hoc Networks. *IEEE Network*, 13(6):24–30, 1999.

Appendix A

Proof Outline: Sender Anonymity Against a Malicious Outsider is `probable innocence` *for* $L_{exp} \geq 4$.

The following events were defined in Section 5.2: E_{route} denotes that a malicious outsider $\psi' \in \Psi$ is selected, on the lower layers, to route a message between γ_i and γ_j; E_{dir} denotes that ψ' can conclude that γ_i precedes γ_j in the path; and $E_{\gamma_i=\gamma_s}$ denotes that $\gamma_i = \gamma_s$. To calculate the best case for the attacker, we assume $(E_{route} \wedge E_{dir})$. The objective of this proof is to define $A_{\gamma_i} = 1 - P(E_{\gamma_i=\gamma_s}|E_{route} \wedge E_{dir})$, which denotes the amount of sender anonymity against a malicious outsider.

1. We start by defining the expected number of hops between γ_s and γ_{last} as follows:

$$H_{exp} = (L_{exp} - 1) - R_L \tag{8}$$

   Without local loops (see Section 5.2), the expected number of hops would simply be $L_{exp} - 1$. However, as with the Crowds protocol, local loops are permitted in Chameleon because a node can randomly choose itself as its successor. By definition, local loops do not affect the virtual path length that denotes the number of appearances of *nodes* between γ_s and γ_{last} (thus including reoccurring nodes) [18]. Still, each local loop decreases the actual number of hops with one, since local messages are not transmitted through the common air interface (i. e., no "hop" is created between the nodes). Therefore, in Equation (8) above, R_L denotes the expected reduction of the number of hops due to local loops. The formula for R_L will be derived below.
2. The next step is to express $P(E_{\gamma_i=\gamma_s}|E_{route} \wedge E_{dir})$. Since the attacker is situated on either of the H_{exp} hops along the virtual path (since $E_{route} \wedge E_{dir}$ is given), the attacker can, using *a priori* knowledge, calculate the possibility that he is routing messages from γ_s in the following way (since the attacker could be situated on either of the hops between γ_s and γ_{last}, as illustrated in Figure 11)[8]:

$$P(E_{\gamma_i=\gamma_s} \mid E_{route} \wedge E_{dir}) = \frac{1}{H_{exp}} = \frac{1}{(L_{exp} - 1) - R_L} \tag{9}$$

3. Since $A_{\gamma_i} = 1 - P_{\gamma_i}$ according to the Crowds metric, the amount of sender anonymity against a malicious outsider can be expressed in the following way:

$$A_{\gamma_i} = 1 - P(E_{\gamma_i=\gamma_s} \mid E_{route} \wedge E_{dir}) \tag{10}$$

[8] Equations (9) and (13) hold when $D \notin D_\Gamma$, which represents the best case for the attacker.

4. To complete the proof, we need to derive an expression for R_L. In order to do this, we first need to model the probabilities of local loops happening during path construction. Given a decision to extend the path, the probability for a node of choosing another random node as the successor (i. e., not causing a local loop) is given by Equation (11), while Equation (12) denotes to probability for a node of choosing itself as its successor (i. e., creating a local loop):

$$P_L = \left(\frac{|\Gamma| - 1}{|\Gamma|}\right) \tag{11}$$

$$P_{\bar{L}} = \left(\frac{1}{|\Gamma|}\right) \tag{12}$$

5. Since the respective random selections of the successor nodes at each node γ_k constitute independent events, the probability for having a certain number of local loops in the virtual path can be modeled by the binomial distribution. More specifically, the probability of having $\#_L$ local loops during path construction can be expressed as follows (where $0 \leq \#_L \leq L_{exp} - 1$):

$$\binom{L_{exp} - 1}{\#_L}\left(\frac{|\Gamma| - 1}{|\Gamma|}\right)^{(L_{exp}-1)-\#_L}\left(\frac{1}{|\Gamma|}\right)^{\#_L} \tag{13}$$

6. Naturally, the sum of the probabilities of having 0, 1, ..., $(L_{exp} - 1)$ local loops adds up to one:

$$\sum_{\#_L=0}^{L_{exp}-1}\left[\binom{L_{exp} - 1}{\#_L}\left(\frac{|\Gamma| - 1}{|\Gamma|}\right)^{(L_{exp}-1)-\#_L}\left(\frac{1}{|\Gamma|}\right)^{\#_L}\right] = 1 \tag{14}$$

7. Further, we can note that there are two cases we can disregard when modeling R_L:
   - *No local loops:* this case, which naturally does not affect R_L, is omitted for clarity;
   - *Only local loops:* since we assume E_{route}, this case cannot happen, since if it would happen, there would be no hops, and, thus, no attacker.
8. A final observation is that if there is $\#_L$ local loops on the path, the *actual* number of hops for a given instance of a virtual path is reduced by $\#_L$. Thus, $\#_L$ constitutes a “scaling factor” when modeling R_L. For example, one local loop decreases the actual number of hops with one, two local loops decrease the actual number of hops with two, etc. For this reason, and when disregarding the two special cases described above, we can express R_L as follows:

$$R_L = \sum_{\#_L=1}^{L_{exp}-2}\left[(\#_L)\binom{L_{exp} - 1}{\#_L}\left(\frac{|\Gamma| - 1}{|\Gamma|}\right)^{(L_{exp}-1)-\#_L}\left(\frac{1}{|\Gamma|}\right)^{\#_L}\right] \tag{15}$$

and, after simplifying the equation above, we have:

$$R_L = \frac{L_{exp} - 1}{|\Gamma|} - (L_{exp} - 1)\left(\frac{1}{|\Gamma|}\right)^{L_{exp}-1} = (L_{exp} - 1)\left(\frac{1}{|\Gamma|} - \left(\frac{1}{|\Gamma|}\right)^{L_{exp}-1}\right) \quad (16)$$

where the first value of the this equation denotes the expected number of loops and the second value represents the expected reduction factor caused by *only local loops*. With Equation (16), it can be shown that R_L decreases with an increasing size of $|\Gamma|$.

9. The expected number of hops can be further derived only in terms of the L_{exp} and Γ.

$$H_{exp} = (L_{exp} - 1) - R_L = (L_{exp} - 1) - (L_{exp} - 1)\left(\left(\frac{1}{|\Gamma|}\right) - \left(\frac{1}{|\Gamma|}\right)^{L_{exp}-1}\right)$$

$$= (L_{exp} - 1)\left(1 - \left(\frac{1}{|\Gamma|}\right) + \left(\frac{1}{|\Gamma|}\right)^{(L_{exp}-1)}\right) = (L_{exp} - 1)\left(\frac{|\Gamma| - 1}{|\Gamma|} + \left(\frac{1}{|\Gamma|}\right)^{(L_{exp}-1)}\right) \quad (17)$$

It can be easily shown by induction that H_{exp} increases with an increasing L_{exp} or with an increasing cardinality of Γ.

10. If $L_{exp} \geq 4$ and $|\Gamma| = 3$ (worst case scenario) then:

$$H_{exp} = 3 * \left[\frac{2}{3} + \frac{1}{3^3}\right] = 2 + \frac{1}{9} = \frac{19}{9} > 2$$

Since H_{exp} increases with an increasing L_{exp} or with an increasing $|\Gamma|$, it follows that $H_{exp} > 2$ for $L_{exp} \geq 4$. Hence, $P(E_{\gamma_i=\gamma_s} \mid E_{route} \wedge E_{dir}) < \frac{1}{2}$ and $A_{\gamma_i} \geq \frac{1}{2}$, meaning that in this case, the provided degree of sender anonymity against malicious outsiders is at least `probable innocence`.

□