View metadata, citation and similar papers at core.ac.uk

brought to you by CORE

provided by Edinburgh Research Explorer

THE UNIVERSITY of EDINBURGH

Edinburgh Research Explorer

Towards Securing Peer-to-peer SIP in the MANET Context: Existing Work and Perspectives

Citation for published version:
Davoust, A, Gagnon, F, Esfandiari, B, Kunz, T & Cormier, A 2018, Towards Securing Peer-to-peer SIP in the MANET Context: Existing Work and Perspectives. in 2017 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData). pp. 223-229. DOI: 10.1109/iThings-GreenCom-CPSCom-SmartData.2017.38

Digital Object Identifier (DOI):
10.1109/iThings-GreenCom-CPSCom-SmartData.2017.38

Link:
Link to publication record in Edinburgh Research Explorer

Document Version:
Peer reviewed version

Published In:
2017 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData)

General rights
Copyright for the publications made accessible via the Edinburgh Research Explorer is retained by the author(s) and / or other copyright owners and it is a condition of accessing these publications that users recognise and abide by the legal requirements associated with these rights.

Take down policy
The University of Edinburgh has made every reasonable effort to ensure that Edinburgh Research Explorer content complies with UK legislation. If you believe that the public display of this file breaches copyright please contact openaccess@ed.ac.uk providing details, and we will remove access to the work immediately and investigate your claim.

Download date: 05. Jun. 2018

Towards Securing Peer-to-peer SIP in the MANET Context: Existing Work and Perspectives

Alan Davoust
School of Informatics
University of Edinburgh
Edinburgh, Scotland
Email: adavoust@inf.ed.ac.uk

François Gagnon
Cybersecurity Research Lab
Cégep Sainte-Foy
Québec, Canada

Babak Esfandiari, Thomas Kunz, and Alexandre Cormier
Department of Systems and Computer Engineering
Carleton University
Ottawa, Canada

***Abstract*—The Session Initiation Protocol (SIP) is a key building block of many social applications, including VoIP communication and instant messaging. In its original architecture, SIP heavily relies on servers such as proxies and registrars.**

Mobile Ad hoc NETworks (MANETs) are networks comprised of mobile devices that communicate over wireless links, such as tactical radio networks or vehicular networks. In such networks, no fixed infrastructure exists and server-based solutions need to be redesigned to work in a peer-to-peer fashion. We survey existing proposals for the implementation of SIP over such MANETs and analyze their security issues. We then discuss potential solutions and their suitability in the MANET context.

I. Introduction

SIP is a protocol best known for Voice over IP (VoIP) applications. It is a key protocol in the IP Multimedia Subsystem (IMS), used to initiate (and close) communication sessions. It does not handle the voice (or more general media) communication itself. This makes SIP applicable to a variety of social applications, including video calls and instant messaging.

In many scenarios, such as military or emergency response operations, the network infrastructure could be made up of mobile devices carried by people or vehicles, communicating directly over wireless connections, without Internet services such as DNS. Such networks are known as Mobile Ad hoc Networks (MANET). Implementing SIP in the MANET context requires a peer-to-peer (P2P) alternative to the standard client-server solution, and this P2P solution must be further adapted to the constraints of MANETs. Existing solutions to deploy SIP over MANET are surveyed in section III.

A key functionality of SIP is to manage the registration of users, i.e. the storage of a directory linking SIP URIs with their present status (online or not, and if so at which network location). The main P2P solution for this is to use a Distributed Hash Table (DHT), as proposed in early P2PSIP literature [28] and formalized as the RELOAD protocol of RFC 6940 [18].

DHTs can be straightforwardly deployed over MANET; however, their use in this context raises new efficiency and security problems. The basic efficiency problem is that point-to-point routing in a MANET is much more expensive than over the wired Internet, and therefore the classic multi-hop DHT routing algorithms may end up criss-crossing the network, at a very high cost.

On the security side, the main weakness of a DHT is usually seen to be the key-based routing functionality, which is used to locate the peer responsible for a given piece of stored data: malicious nodes in the network can disrupt the routing procedure and lead to data storage or retrieval requests being sent to other malicious nodes, who may not fulfill the requests properly. We discuss these security threats in section IV.

Solutions to this problem usually rely on redundant routing and/or redundant storage [4], [19], [32], which is problematic in the MANET context. For one thing, it drastically increases the (already high) cost of these operations, and for another it may rely on underlay properties that do not hold for MANET: for example, a remote peer cannot necessarily be reached by two physically disjoint paths, or this will not happen when the underlay routing is oblivious to the security issue.

The main contribution of this paper is an analysis of these security solutions (section V) and of the security and performance trade-offs specific to the MANET context. We discuss these in section VI, before drawing some conclusions and pointing to future work opportunities.

II. Background: The Session Initiation Protocol

As an IETF standard, SIP is defined by RFC 3261, and is normally based on a client-server architecture, i.e. a network of SIP User Agent devices (e.g. IP phones) and SIP servers (typically managed by larger organizations or Internet providers). SIP devices may act as clients (User Agent Client, UAC), when they initiate communication, and as servers (UAS) when they receive messages (typically when another user is attempting to reach them).

SIP servers may have any or all of three roles:

- SIP Registrar: SIP clients, identified by an ID similar to an email address (sip:alice@company.co) register with an Address-Of-Record (AOR) entry indicating whether they are online and their current IP address. The registrar manages the storage of these AOR.
- SIP Proxy: A SIP proxy forwards requests to other SIP servers, with the goal of routing the requests closer to the target. For example, an INVITE request must be routed to the registrar of the callee, then to the callee UAS.
- SIP Redirect: A redirect server simply redirects requests to other valid servers.

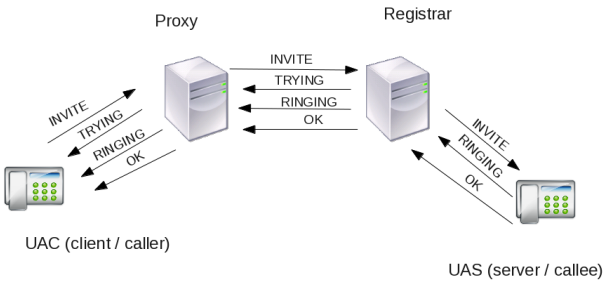

Fig. 1. A simple SIP call flow. The message chronology goes from top to bottom.

The standard scenario of a SIP communication session (Figure 1) begins when Alice wants to communicate with Boris. Alice's device (acting as a UAC) sends an INVITE request to a SIP proxy, which responds to Alice's UAC with a '100 TRYING' response, and forwards the request to Boris's registrar. To locate this registrar in the Internet context, Alice's proxy would use a DNS lookup for the domain of Boris's SIP address.

Boris's registrar receives the request, responds again with 'TRYING', then looks up Boris's current location and forwards the INVITE request there. Boris's UAS responds with 'RINGING' (this message is forwarded back to Alice), then sends an 'OK' message if Boris accepts to initiate the session (picks up the phone). The actual applicative session (e.g. voice communication or messaging) then happens over a separate protocol.

Finally, either Alice or Boris hangs up, and their user agent sends a 'BYE' message which is propagated to the other user agent (not shown).

III. P2PSIP over MANET: State of The Art

There are currently only a few proposed solutions that specifically address the implementation of SIP over MANETs: TacMAN [22], the unnamed proposal of T. Wongsaardsakul's 2010 PhD thesis [34], two separate approaches by Banerjee et al. [5], [6], respectively identified as "Loosely Coupled" (LCA) and "Tightly Coupled" Approach (TCA), SIPHoc [30], [29], MANETSip [16], AdSIP [35], and finally two unnamed proposals, by O'Driscoll et al. [24] and Aburumman et al. [1], respectively.

A. Registration Solutions

In the absence of conventional SIP servers to act as registrars, the main problem is how to handle (or replace) this registration service.

As appropriately analyzed by Stuedi [29] the different solutions to store information in a distributed system with N nodes range from local information storage (with insertion cost $O(1)$) and broadcast-based discovery (with cost $O(N)$), to full replication, with insertion cost $O(N)$ and lookup cost $O(1)$. A DHT with efficient key-based routing provides both insertion and lookup at a cost of $O(log(N))$. However, this logarithmic cost is that of key-based routing, and is counted in *overlay hops*: calculated in MANET hops, it may not be logarithmic at all (e.g. if the underlay uses AODV routing [25], then a single overlay unicast requires – in the worst case – a full broadcast in the underlay).

These registration solutions can be used to classify existing implementations of SIP over MANET, as follows.

1) Local Storage: The "pull" solution of TacMAN [22] and the LCA approach by Banerjee et al. [5] are proposals where each participant in the network simply stores their own information locally, and remote peers wanting to contact them must broadcast a search to locate them, using the underlying MANET routing protocol (HOLSR in TacMAN, AODV in LCA).

2) Dominating Set: In the TCA approach by Banerjee et al.[6] and in AdSIP [35], a subset S of the peers are chosen to act as registrars, where this subset constitutes a dominating set of the network graph (i.e. any node in the network has at least one neighbour in S). In the solution proposed by Aburumman et al. [1], the SIP registration and proxy services are also implemented by a subset of the nodes, although not necessarily a dominating set. A similar mechanism is used in MANETSip [16], although how the nodes forming this subset are chosen is unclear.

In all these cases, the nodes register only with the nearest node(s) in S: in [6] and [1] the nodes of S are the heads of clusters, and each node registers with its clusterhead, whereas in AdSIP each node registers with all of its neighbours in S via a one-hop broadcast. [1] also includes backup servers (one per cluster) that replicate data from the cluster's primary SIP registrar and proxy to provide load balancing and easier hand-off when the clusterhead leaves.

The lookup operation then requires a (limited) broadcast within S. AdSIP claims to improve over TCA by producing a cheaper broadcast, mainly due to S forming a connected graph in the physical network. In MANETSip, the broadcast in S uses an efficient multicast protocol called ODMRP [21].

3) DHT storage: The solution by Wongsaardsakul [34] and SIPHoc [30] use unstructured DHT-like solutions without key-based routing, O'Driscoll et al. [24] propose a hierarchical DHT solution with a Chord backbone.

Wongsaardsakul's solution is a cross-layer designed DHT called SMON (Structured Mesh Overlay Network), based on an earlier design called CrossROAD [12]. As in other DHTs, the peers generate an ID through a hash of their IP address (or some other method), and are responsible for storing objects with keys numerically closest to their ID. SMON does not have any particular structured topology (as opposed to a Chord ring for instance), instead the peers have full knowledge of all the other participating peers and their IDs, which are obtained from the (slightly modified) underlying protocol OLSR [11], which handles the MANET routing.

The underlying "distributed directory" of SIPHoc, called MAND, is comparable to SMON, except that it supports local storage, global storage, and hash-based storage[1] (by

[1]We note that the details of the hash-based solution are unclear.

configuration). A similar system to MAND and SMON called OneHopOverlay4MANET[23] was also proposed outside of the P2PSIP literature, also implementing a cross-layered designed DHT using OLSR as an underlay.

The solution proposed by O'Driscoll et al. requires all of the nodes to participate in a hierarchical DHT. The nodes are clustered and only the clusterheads form a Chord ring, with the nodes within each cluster forming a structured or unstructured network. This proposal lacks an implementation or an evaluation, and it is unclear how the routing across clusters might work: presumably there needs to be a relationship between the IDs of each cluster.

B. DHT over MANET Solutions

It is intriguing that despite the normative focus on DHTs found in the P2PSIP proposals, few existing solutions in the MANET context make use of the full DHT model[2]: SMON and MAND are only a DHT in the sense that they allocate storage to the nodes using keys and node IDs in the same address space, but the network is not structured. Routing is done with OLSR, which does not scale very well: the nodes must have knowledge of the entire network topology.

However, it is worth noting that outside the specific SIP over MANET literature, a few efficient solutions have been proposed to implement DHTs over MANETs, which would most likely be directly usable to implement P2PSIP. We present them briefly here.

1) MADPastry: MADPastry[37] is an integration of the Pastry DHT [27] with the AODV MANET routing protocol [25]. The main idea is to create a topology-aware Pastry overlay, adapted to a mobility scenario.

The approach in MADPastry uses *Random Landmarking* [33], where a small number of well-known reference keys are chosen in the identifier space, and "landmark" measurements are computed by the peers responsible for these keys. The overlay then reconfigures itself periodically to maintain a logical topology adapted to the physical locations of the nodes.

2) Virtual Ring Routing: Virtual Ring Routing (VRR)[9] is a proposal to use a virtual ring topology based on random node identifiers (e.g. using hashes) directly at the MANET level, to accomplish MANET routing in addition to supporting DHT storage using keys in the same identifier space.

The nodes organize in a virtual ring similar to Chord, without the finger tables to distant nodes. Each node only maintains connections to r neighbours of the virtual ring, plus connections to its physical neighbours. For example with $r = 4$ the node maintains multi-path routes to its two immediate predecessors and two followers in the virtual ring.

In addition, paths between virtual neighbours are also recorded at every physical hop to facilitate routing.

3) Cell Hash Routing: Cell Hash Routing (CHR) [3] is a proposal where the geographical space where the MANET exists is partitioned into equal-sized cells. The DHT algorithm assigns keys to cells (rather than to individual nodes) and nodes inside the cell manage these keys. If a cell is empty then the nearest non-empty cell takes over.

Routing is done using an adapted version of GPSR[20], a geographic routing algorithm.

[2]The only true DHT-based solution appears to be that of O'Driscoll et al., although its specification is insufficient for us to comment on the details.

IV. Threat Model

The key security properties for general P2P networks are resilience and integrity. Resilience deals with service availability, while integrity covers the content delivered by the network.

In some P2P contexts, it might be important to provide other security properties, such as anonymity and confidentiality: anonymity aims to prevent malicious nodes from learning about the origin of requests (e.g., Tarzan [15], MorphMix [26], and Octopus [32]), whereas confidentiality means malicious nodes should not learn about the content that is being queried from the DHT (e.g., [14]).

As we will see later, some of these properties are necessary to enable certain defenses against other attacks.

In the context of the P2PSIP application, our main focus is on the Sybil attack and on attacks on the *Storage and Retrieval* functionality of the DHT. These include both denial of service (DoS) attacks and attacks on the integrity of stored data.

It is worth noting that DoS attacks may also happen at different network levels, and we focus here on the risks at the DHT level.

A. Sybil Attack

The Sybil attack [13] comes from the inherent openness of P2P systems and consists of one malicious peer being able to act as multiple different logical nodes in the system (i.e., controlling multiple IDs in the DHT). By itself, this kind of attack does not compromise the resilience nor the integrity of the network. However, the ability for a malicious entity to easily control a large number of nodes greatly enhance its ability to perform attacks. For instance, if an entity can setup several malicious nodes in a network, his chances of disrupting the network through a DoS attack are vastly superior than if he controls a single one.

An extreme case of the Sybil attack is the Eclipse attack [17] where a malicious entity controls all the nodes "surrounding" his target. Hence, every query from the target peer passes through a malicious node which can then manipulate the data as he sees fit since the target has a compromised view of the logical network.

B. Storage & Retrieval Attacks

Storage & retrieval attacks affect the primary objective of P2P systems which is to store information and provide access to that information. Several different attacks can fall into this category.

1) Routing-Based Attacks: The simplest routing attack consists of dropping requests seeking the next hop to reach the node (denoted as P(k)) responsible for storing the information a querier is interested in. This attack can take place whenever a malicious node happens to be on the request path and poses resilience issues.

A more sophisticated version of such a routing attack is to provide a wrong answer to the querier sending him on a false trail. Two objectives can be sought through this attack: prevent the querier from getting an answer by looping through a set of malicious nodes (which achieve the same DoS goal as dropping the query) or fooling the querier by providing a false (and malicious) P(k). This then sets up the table for a Query-Based attack (see below).

2) Query-Based Attacks: The easiest query attack is for a malicious node to refuse to serve the data by not answering the query; a DoS attack.

A more interesting scenario is for a malicious node to answer a query with bogus data. If the querier is unable to detect the attack, the data integrity will be compromised leading to potentially serious problems. If the querier detects the attack, a DoS is likely to occur unless strong mechanisms are implemented to recover from such a situation.

3) Resource Exhaustion Attack: RELOAD [18] mentions a type of resource exhaustion attack where a node is asked to store an abnormally large amount of data by malicious nodes. As a consequence, the attacked node might be unable to store legitimate data for which it should be responsible.

4) Injection Attack: In a P2P SIP application where the DHT acts as a naming service (mapping each SIP ID to an Address-Of-Record), *injection attacks* become a real threat. In this attack, an attacker inserts/overwrites data in the DHT instead of focusing on attacking the query mechanism.

5) Replay Attack: If the address (location identifier) of a peer may change over time, then a replay attack becomes possible: an attacker currently located at address A could reuse an old DHT entry saying SIP ID of user U is located at A (U was located at A in the past) to fool a querier into thinking U is at A while a malicious node is there.

This may happen if IP addresses are dynamically allocated, or if a peer's location is defined by a changing DHT identifier, as in MADPastry, DART or CHR.

V. Existing Security Solutions

For each main type of attack, we present existing defenses in general terms, and discuss their applicability for the different DHT solutions listed in section III.

We first address Sybil attack resistance, then storage and retrieval attacks. For the latter, we distinguish the problem of securing the key-based routing (KBR) process from the complementary problem of ensuring the integrity of the data being returned by a queried peer.

A. Sybil Attack Resistance

As the Sybil attack consists in an attacker controlling several distinct (virtual) identities, the main defenses for this attack are (i) increasing the cost of new identities, or (ii) detecting multiple virtual identities linked to the same physical entity, or (iii) designing a protocol to limit the impact of Sybil identities.

1) Public Key Infrastructure (PKI): The main solution to the Sybil attack is to have a trusted, centralized certificate authority (CA) to certify that a given public key is associated with a given real-world entity. This assumes that an attacker cannot control multiple such real-world identities, and relies on a centralized authority that allows new nodes to join the network. The central authority must be able to validate the real-world identity of each node.

The applicability of this method mainly depends on the context in which the P2P system will be used. For a private system, e.g. a military battlefield network, the use of a PKI will make sense. But for open systems this will be a problem.

One difficulty is to decide which identities to bind the keys to. In [10], Castro et al. propose to assign certificates based on nodes' IP addresses, to compensate for the lack of other identities. This approach requires that nodes have fixed IPs.

They also propose a revocation mechanism to activate when nodes are misbehaving, but this requires the presence of an online CA.

In addition, some MANET-specific DHT protocols (e.g. CHR, MADPastry) use location-dependent identifiers. In such a protocol, nodes must change identifier as they move, and the CA would have to re-certify each new identifier, which even with an online CA seems highly impractical.

Persea [2] takes a different approach to the PKI by decentralizing it. When a node joins a Persea DHT, an existing node delegates part of the ID space it is responsible for to the new node and certifies the ID it assigns to the joining node. The certificate is then stored in the DHT itself. This prevents an attacker that has successfully joined the DHT from gaining more power by inviting Sybil peers as it can only delegate part of its own ID space. For this to be effective, it must be hard for an attacker to join through an honest node, which could be achieved using another Sybil attack protection mechanism.

2) Crypto-Puzzles: Another way of mitigating a Sybil attack is to raise the cost of identities. In S/Kademlia[7], identities are defined by key pairs where the public key begins with one or more zeros (more zeros makes the crypto-puzzle more difficult to solve). This means generating an identity is costly because the nodes must generate many key pairs before finding one that matches the zeros requirement.

For a MANET, the computation cost of generating identities should not be too high since nodes (embedded resources) are likely to have limited computing power, and one weakness of this solution is that malicious actors could use offline resources [3] to massively solve crypto-puzzles.

3) Physical Triangulation: In a MANET, Sybil nodes will correspond to the same physical device, and it has been proposed [8] that Sybils could be detected by identifying nodes located in the same place by triangulating their location with respect to several "landmark" nodes. Although with high mobility this approach will be challenging, the general idea is useful: with location-dependent identifiers, Sybils will either all be clustered together in the same part of the identifier space

[3]Resources outside the P2P network, e.g., cloud.

(a fact which can be used to design Sybil-resistant protocols), or are more likely to be detected (as a physically close node may detect the fact that their identifier does not match their location).

B. Securing Key-Based Routing

Securing KBR aims to ensure that an honest peer can obtain the address of the peer responsible for key k, which we note $P(k)$.

The first approach to this problem is to secure the intermediate steps of the KBR protocol (i.e. ensure that peers comply with the routing algorithm, or detect peers who do not), and the second is to make $P(k)$ verifiable, i.e. once the KBR returns an address, the querying peer can verify that the addressed peer is indeed responsible for k according to the algorithm.

1) Octopus: Octopus [32] aims to detect when peers attempt to manipulate their routing tables and redirect queries to malicious attackers. This is achieved by sending anonymous lookup requests to these predecessor nodes, and checking their result. Octopus uses Onion routing to create anonymous queries to intermediate nodes, and relies on public key certificates. The certificates of peers found to be malicious are revoked by the CA, who must also investigate each case to avoid legitimate nodes being slandered by malicious nodes.

One problem with this approach is that it requires the routing protocol to be verifiable without local information. For standard KBR, this is possible, but MANET-specific protocols such as VRR or MADPastry allow peers to redirect requests when they know of shorter physical routes than the KBR route. Such protocols would be incompatible with Octopus.

2) Redundant Paths: HALO and Cyclone: HALO [19] and Cyclone[4] are two proposals to support redundant KBR, i.e. routing messages through multiple paths. As a result, a malicious node attempting to disrupt the KBR would need to control many different nodes to break all the redundant routing paths, and the querying node has a higher probability of reaching $P(k)$. If different paths return different answers, it should be apparent which node is actually responsible for k (the one with the ID closest to k).

The approach of Cyclone is to construct separate overlay networks (e.g. several Chord rings) involving the same nodes but with different topologies. This means that independent paths will exist in the different networks.

In HALO, which was designed for Chord, there is a single Chord ring and the peers send lookup requests for the same key but to different peers. The key idea is to compute the likely locations in the address space of nodes that have $P(k)$ in their finger table, then ask each of those nodes to perform $locate(k)$.

3) Quorums: An alternative approach to independent paths (but still essentially redundancy) is the use of *quorums* along the routing path [36]: a quorum is a group of peers responsible for the same step of a distributed algorithm (e.g. a step along a DHT route), and provided that a majority (2/3 or 3/4 depending on the algorithm) of the quorum members are legitimate, the quorum members can vote (or use a Byzantine agreement protocol) to figure out the correct outcome of their shared processing step.

Redundant paths give a node a higher probability of reaching $P(k)$ without going through a malicious node, unless $P(k)$ itself is malicious. However, the routing mechanisms in DHT make it difficult to obtain *disjoint* paths from one node to another. Instead, multiple paths are likely to converge and pass through the same nodes close to $P(k)$, which makes it possible for one node to corrupt many paths. Different solutions have been proposed to allow nodes to obtain disjoint paths, including HALO and Cyclone (see above).

4) Myrmic: Myrmic [31] is a cryptographic solution to make sure that the responding node is indeed responsible for the key. The idea is that a neighbourhood authority (NA) signs certificates stating that peer P is responsible for a given range of the ID space. The main difficulty is to handle churn, since when peers join or leave the DHT, data that they are responsible for changes. The NA revokes past certificates by issuing new ones and distributes them to each node's neighbours. In order to check that some node P is actually responsible for key k, P must show a certificate saying so, and P's neighbours must not have any more recent certificate (stating a different range). The querying node must therefore be able to contact P's neighbours. Presumably, the list of neighbours to be contacted ("witnesses") would be specified in the certificate itself.

The NA acts as local CA to circumvent the requirement of an always online central authority. However, if NA are peers in the network, they must not be malicious otherwise they could issue false certificates.

C. Data Integrity

The above solutions to secure KBR aim to ensure that $locate(k)$ requests reach their correct target. However, if the peer $P(k)$ itself is malicious, then those techniques do not help. Additional approaches have been proposed to ensure the integrity of stored data, in order to address this problem.

1) Redundant Storage: Redundant storage implies that the information is stored at (and retrieved from) multiple nodes. In the presence of inconsistencies during retrieval, the querier will know an attack attempt is probably underway. However, if as little as one answer from the retrieval process differs from the others, the querier has no idea which is the real value. Whenever an attacker can manage to manipulate all the answers (only one in the absence of duplication) for the retrieval process to the same erroneous value, the querier has no way of knowing an attack is underway.

2) Anonymous Data Verification: Following the idea of Octopus (above) regarding routing, peers can also ensure that the peers storing their data (e.g. their OAR entry) are honest by periodically requesting their own data, which they can then check against the data that they stored. Of course, this relies on anonymity, which ensures that a malicious node cannot distinguish "test" queries from "real" queries.

3) Verifiable Data: By focusing on the P2P SIP application, it becomes easier to make data cryptographically verifiable: if

peers store the AOR for their own SIP ID, they can sign their entries using their private key, and another peer wishing to lookup up this peer's address can verify the signature using the public key associated with this peer ID. This prevents malicious peers from creating fake entries for other SIP IDs.

However, if the peer identifiers may change, replay attacks become possible. In this case, a signed DHT entry should be valid only for a specific time window. This will force peers to refresh their DHT entry, but will limit the opportunity for replay attacks.

This idea is used in RELOAD [18], where a PKI must be in place, although without an always-online central CA. The CA is necessary for pre-configuration of nodes before them joining the network but does not participate in the actual network operations.

VI. Discussion and Conclusion

As a result of our previous analysis, we have identified two main trade-offs between performance and security.

The first trade-off is related to Key-Based Routing, and the second is related to optimizations of KBR in the MANET context.

A. Redundancy and performance with Key-Based Routing

Key-based routing is an essential feature of DHT, that provides scalability and efficiency, since peers can route messages in a large network while only keeping track of a small subset of the other peers. However, KBR also creates a major security vulnerability, for which most defenses rely on redundancy (in storage or routing). Redundant routing increases the cost of routing (it is multiplied by a constant factor), and depending on network parameters it is possible that the additional cost offsets the efficiency gains of KBR.

Most existing cross-layer designs for SIP in MANET contexts use key-based storage and MANET routing, and are therefore not vulnerable to attacks on KBR. Although the lower-level routing may still be vulnerable, this leaves one level of vulnerability rather than two. On the other hand, this requires the peers to have full knowledge of the DHT membership, and greatly reduces scalability.

B. KBR in mobile contexts

KBR is known to be much less efficient when deployed naively over a MANET, and a number of MANET-specific designs have been proposed to mitigate this problem. However, these performance optimizations also incur a cost in terms of security, for two reasons: changing identifiers and ad-hoc routing.

a) Changing identifiers: In several DHT-based MANET routing protocols (e.g. MADPastry, CHR and DART), the peers' identifiers are location-dependent, and must therefore be changed when peers move far enough. This improves the performance of KBR in the MANET, but creates an important security vulnerability. The fact that even legitimate peers may adopt new identifiers makes it more difficult to detect and defend against Sybil attacks, and creates a risk of replay attacks.

b) Local routing decisions: In MANET-specific DHT routing protocols, including those mentioned above and VRR, routing is not strictly key-based: peers can make local routing decisions based on their knowledge of their physical neighbourhood. Therefore, routing decisions cannot be verified by another peer, as in the idea of Octopus. Quorum solutions may mitigate this problem, but in protocols with changing identifiers attackers could easily create local groups of Sybils that would act as a quorum.

C. Conclusion

In this paper we surveyed various techniques in the literature for the deployment of SIP, a useful protocol to support social applications, in mobile ad-hoc networks.

Our security analysis of these solutions shows two important trade-offs between scalability and security, due to key-based routing and to the optimizations of key-based routing in a MANET context.

In addition, the MANET context also reduces the applicability of cryptography-based solutions (certifications), including those provisioned in the RELOAD protocol, as devices could fall into malicious hands and start to misbehave.

On the bright side, the P2PSIP application has some interesting properties such as verifiable data, which reduces security risks related to data integrity.

In future work, we intend to explore more specifically the attacks on P2PSIP and solutions using cross-layer designed DHT, i.e. secure extensions to DHT-based MANET routing, or "natively" redundant MANET routing protocols (e.g. ODMRP).

Acknowledgement

This research was sponsored by the Army Research Laboratory/US Army RDECOM-Americas and was accomplished under Cooperative Agreement Number W911NF-16-1-0345. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Army Research Laboratory/US Army RDECOM-Americas or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation herein

References

[1] A. Aburumman, W. J. Seo, C. Esposito, A. Castiglione, R. Islam, et al. A secure and resilient cross-domain sip solution for manets using dynamic clustering and joint spatial and temporal redundancy. *Concurrency and Computation: Practice and Experience*, 2016.

[2] M. N. Al-Ameen and M. Wright. Persea: a sybil-resistant social dht. In *Proceedings of the third ACM conference on Data and application security and privacy*, pages 169–172. ACM, 2013.

[3] F. Araujo, L. Rodrigues, J. Kaiser, C. Liu, and C. Mitidieri. CHR: a distributed hash table for wireless ad hoc networks. In *Distributed Computing Systems Workshops, 2005. 25th IEEE International Conference on*, pages 407–413. IEEE, 2005.

[4] M. S. Artigas, P. G. Lopez, and A. F. G. Skarmeta. A novel methodology for constructing secure multipath overlays. *IEEE Internet Computing*, 9(6):50–57, Nov 2005.

[5] N. Banerjee, A. Acharya, and S. K. Das. Peer-to-peer SIP-based services over wireless ad hoc networks. In *BROADWIM: Broadband Wireless Multimedia Workshop*, 2004.

[6] N. Banerjee, A. Acharya, and S. K. Das. Enabling SIP-based session setup in ad hoc networks. In *Proceedings of INFOCOM*. Citeseer, 2005.

[7] I. Baumgart and S. Mies. S/Kademlia: A practicable approach towards secure key-based routing. In *Parallel and Distributed Systems, 2007 International Conference on*, volume 2, pages 1–8. IEEE, 2007.

[8] R. A. Bazzi and G. Konjevod. On the establishment of distinct identities in overlay networks. *Distributed Computing*, 19(4):267–287, 2007.

[9] M. Caesar, M. Castro, E. B. Nightingale, G. O'Shea, and A. Rowstron. Virtual ring routing: Network routing inspired by dhts. In *Proceedings of the 2006 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications*, SIGCOMM '06, pages 351–362, New York, NY, USA, 2006. ACM.

[10] M. Castro, P. Druschel, A. J. Ganesh, A. I. T. Rowstron, and D. S. Wallach. Secure routing for structured peer-to-peer overlay networks. In *5th Symposium on Operating System Design and Implementation (OSDI 2002), Boston, Massachusetts, USA, December 9-11, 2002*, 2002.

[11] T. Clausen, P. Jacquet, C. Adjih, A. Laouiti, P. Minet, P. Muhlethaler, A. Qayyum, and L. Viennot. Optimized Link State Routing Protocol (OLSR). Technical report, 2003. Network Working Group.

[12] F. Delmastro. From pastry to crossroad: Cross-layer ring overlay for ad hoc networks. In *Pervasive Computing and Communications Workshops, 2005. PerCom 2005 Workshops. Third IEEE International Conference on*, pages 60–64, March 2005.

[13] J. R. Douceur. The sybil attack. In *Peer-to-Peer Systems, First International Workshop, IPTPS 2002, Cambridge, MA, USA, March 7-8, 2002, Revised Papers*, pages 251–260, 2002.

[14] M. Fonville. Confidential peer-to-peer file-sharing using social-network sites. In *13th Twente Student Conference on IT, Jun*, volume 21, page 10, 2010.

[15] M. J. Freedman and R. Morris. Tarzan: a peer-to-peer anonymizing network layer. In *Proceedings of the 9th ACM Conference on Computer and Communications Security, CCS 2002, Washington, DC, USA, November 18-22, 2002*, pages 193–206, 2002.

[16] S. Fudickar, K. Rebensburg, and B. Schnor. MANETSip - a dependable SIP overlay network for MANET including presentity service. In *Networking and Services, 2009. ICNS '09. Fifth International Conference on*, pages 314–319, April 2009.

[17] H. Ismail, D. Germanus, and N. Suri. Detecting and mitigating P2P eclipse attacks. In *21st IEEE International Conference on Parallel and Distributed Systems, ICPADS 2015, Melbourne, Australia, December 14-17, 2015*, pages 224–231, 2015.

[18] C. Jennings, B. Lowekamp, E. Rescorla, S.Baset, and H. Schulzrinne. REsource LOcation And Discovery (RELOAD) Base Protocol. RFC 6940, RFC Editor, January 2014.

[19] A. Kapadia and N. Triandopoulos. Halo: High-assurance locate for distributed hash tables. In *NDSS*, volume 8, page 142, 2008.

[20] B. Karp and H.-T. Kung. GPSR: Greedy perimeter stateless routing for wireless networks. In *Proceedings of the 6th annual international conference on Mobile computing and networking*, pages 243–254. ACM, 2000.

[21] S.-J. Lee, W. Su, and M. Gerla. On-demand multicast routing protocol in multihop wireless mobile networks. *Mobile Networks and Applications*, 7(6):441–453, 2002.

[22] L. Li and L. Lamont. Support real-time interactive session applications over a tactical mobile ad hoc network. In *Military Communications Conference, 2005. MILCOM 2005. IEEE*, pages 2910–2916. IEEE, 2005.

[23] M. A. Mojamed. *A one hop Overlay System for mobile Ad Hoc Networks*. PhD thesis, University of Stirling, 2016.

[24] A. O'Driscoll, S. Rea, and D. Pesch. Hierarchical clustering as an approach for supporting P2P SIP sessions in ubiquitous environments. In *9th IFIP International Conference on Mobile Wireless Communications Networks, MWCN 2007, Cork, Ireland, 19-21 September, 2007*, pages 76–80. IEEE, 2007.

[25] C. E. Perkins and E. M. Royer. Ad-hoc on-demand distance vector routing. In *IN PROCEEDINGS OF THE 2ND IEEE WORKSHOP ON MOBILE COMPUTING SYSTEMS AND APPLICATIONS*, pages 90–100, 1997.

[26] M. Rennhard and B. Plattner. Introducing morphmix: peer-to-peer based anonymous internet usage with collusion detection. In *Proceedings of the 2002 ACM Workshop on Privacy in the Electronic Society, WPES 2002, Washington, DC, USA, November 21, 2002*, pages 91–102, 2002.

[27] A. Rowstron and P. Druschel. Pastry: Scalable, decentralized object location, and routing for large-scale peer-to-peer systems. In R. Guerraoui, editor, *Middleware 2001: IFIP/ACM International Conference on Distributed Systems Platforms Heidelberg, Germany, November 12–16, 2001 Proceedings*, pages 329–350, Berlin, Heidelberg, 2001. Springer Berlin Heidelberg.

[28] K. Singh and H. Schulzrinne. Peer-to-peer internet telephony using SIP. In *Proceedings of the international workshop on Network and operating systems support for digital audio and video*, pages 63–68. ACM, 2005.

[29] P. Stuedi. *From Theory to Practice: Fundamental Properties and Services of Mobile Ad Hoc Networks*. PhD thesis, ETH Zurich, 2008.

[30] P. Stuedi, M. Bihr, A. Remund, and G. Alonso. SIPHoc: Efficient SIP middleware for ad hoc networks. In R. Cerqueira and R. H. Campbell, editors, *Middleware 2007, ACM/IFIP/USENIX 8th International Middleware Conference, Newport Beach, CA, USA, November 26-30, 2007, Proceedings*, volume 4834 of *Lecture Notes in Computer Science*, pages 60–79. Springer, 2007.

[31] P. Wang, I. Osipkov, N. Hopper, and Y. Kim. Myrmic: Secure and robust DHT routing. Technical report, U. of Minnesota, 2006.

[32] Q. Wang and N. Borisov. Octopus: A secure and anonymous dht lookup. In *Distributed Computing Systems (ICDCS), 2012 IEEE 32nd International Conference on*, pages 325–334. IEEE, 2012.

[33] R. Winter, T. Zahn, and J. Schiller. Random landmarking in mobile, topology-aware peer-to-peer networks. In *Distributed Computing Systems, 2004. FTDCS 2004. Proceedings. 10th IEEE International Workshop on Future Trends of*, pages 319–324. IEEE, 2004.

[34] T. Wongsaardsakul. *P2P SIP over mobile ad hoc networks*. PhD thesis, Evry, Institut national des télécommunications, 2010.

[35] S. Yahiaoui, Y. Belhoul, N. Nouali-Taboudjemat, and H. Kheddouci. AdSIP: Decentralized SIP for mobile ad hoc networks. In *Advanced Information Networking and Applications Workshops (WAINA), 2012 26th International Conference on*, pages 490–495, March 2012.

[36] M. Young, A. Kate, I. Goldberg, and M. Karsten. Practical robust communication in DHTs tolerating a byzantine adversary. *2013 IEEE 33rd International Conference on Distributed Computing Systems*, 0:263–272, 2010.

[37] T. Zahn and J. Schiller. MADPastry: A DHT substrate for practicably sized MANETs. In *Proc. of ASWN*, 2005.