Privacy in the Pervasive Era:  
A Distributed Firewall Approach  

(Extended Version)  

Leonardo Maccari  
DISI – University of Trento  
Trento, Italy  
Email: leonardo.maccari@disi.unitn.it  

Renato Lo Cigno  
DISI – University of Trento  
Trento, Italy  
Email: locigno@disi.unitn.it  

Abstract—Pervasive computing and communications are (slowly) enabling local ad-hoc services, including social networking based on wireless mesh networks. Preserving privacy in a pervasive environment is one of the key challenges ahead: How can users define their “communication boundaries”? how can the network avoid wasting resources and eventually collapse under the burden of undesired traffic that will be discarded at the receiver machine? In this paper we propose the adoption of distributed filtering techniques implementing a network-wide firewall whose goal is defining precisely, and under the user control, the boundaries in space, time, information content, and logical addressing of a user communication scope. The techniques adopted include rules definition for limiting the communications scope, their distribution within an ad-hoc mesh network and their selected usage (if any) by network nodes trying to prevent that undesired traffic, which will be in any case discarded at the destination, gobbles network resources. Initial results based on an implementation integrated with OLSR are presented, and requirements to make the distributed firewall independent from the routing protocol adopted are also discussed.

Index Terms—Privacy; Distributed Filtering; Firewalling; Ad Hoc Networks; Mesh Networks; Local Social Networks.

I. INTRODUCTION

With the success of the Internet and its convergence with mobile communications, people lives take place (also) in the cyber-αγορά1. Life in public spaces is governed by behavioral rules, respect of other people space and privacy, and it is traditionally limited in time and space: retreating home or changing place changes the situation and the rules. The cyber-αγορά instead is anywhere at anytime, collapsing a multiplicity of situations and scenarios into a single social space that imposes new rules of behavior: rules that are often difficult to understand, more difficult to describe, and even more difficult to enforce.

Fig. 1 depicts a high level view of the cyber-αγορά: Users have direct connections within the mesh network, interact one another and with ubiquitous sensors and other machines (including cars) with communication capabilities. Mesh nodes may act as gateways to the global Internet and users nodes can be connected on the mesh and, at the same time to the global mobile communication system. What messages should a user receive, who and what are the legitimate destinations of, e.g., a user location, how many hops in the ad-hoc network should a message travel for the the benefit of its sender. These are all legitimate questions which require the definition of filtering rules, their diffusion in the ad-hoc network and their enforcement in some (all?) nodes. Besides, there is the issue of performance: Where is the best place to enforce a rule? What is the cost, in terms of resource waste as well as other more “user centric” metrics like loss of privacy, of wrong decisions or limited filtering capabilities?

Summarizing all these issues in two words there is a need to correctly define the ‘communication scope’ for a node/user in a pervasive environment. We decide to focus our research on a specific kind of network, that is a social network built on top of a distributed wireless network. In this context the adoption of filtering is important for protecting both the user’s privacy and to avoid the misuse of the network resources. To the best of our knowledge, this is a novel problem, which has so far never been tackled in the context we analyze.

The rest of the paper is organized as follows: Sect. II states the problem in a networking scenario and defines it as as a problem of distributed firewalling; Sect. III gives a more formal definition of the problem, enabling to split it into more manageable subproblems; Sect. IV introduces existent literature to support our view and the assumptions made; Sect. V presents initial results from a sample implementation integrated, within an Omnet++ simulation environment, with the OLSR routing protocol; Sect. VI ends the paper with a discussion of the contribution so far.

II. INFORMATION FILTERING IN PERVERSIVE NETWORKS

Consider again the scenario defined in Fig. 1. A user is immersed in a wireless mesh, where different services are offered on-demand, from basic connectivity, to localization services, to local social networks and so on. Let’s assume, without loss of generality that nodes and users are on a one-to-

1αγορά is the original Greek spelling of ‘agora’, the public open space of political, social, and commercial life in ancient Greek city-states.
one relationship\(^2\). A given node, say \(n_i\), wants and must share information with other nodes in the same area, however, not all information is to be shared with everyone: the node might want to restrict to a subset of the other nodes his location, it is not interested in receiving information on social games, and so forth.

The scenario sketched above raises three fundamental issues for its effective implementation: \(i\) each node should define a set of rules \(S_i\) for traffic involving himself; \(ii\) an efficient way of distributing \(S_i\) must be found; and \(iii\) a proper strategy for the implementation of rule-sets must be devised, which achieves some given goals for the benefit of both the network and the users.

This paper focus on the second issue: we assume that \(S\) is defined and known to every node, so that the description of preferences and restrictions by any node \(n_i\) reduces to a list of rules identifiers defining a proper subset \(S_i \subseteq S\).

The rules in \(S_i\) can be enforced in \(n_i\) only, as in a traditional local firewall. This, however, is extremely inefficient, unpractical, and possibly not even feasible. It is inefficient when packets are carried all the way from the source to \(n_i\), which finally drops them. It is unpractical when \(n_i\) wants to send some information to a subset only of other nodes, and he has to know them in advance to send unicasting packets to all of them. Finally, it can be simply impossible for packets which are inherently broadcast, like for instance position beacons in vehicular networks, so that filtering at the source is not possible; a practical case is a car stopped due to some emergency on the carriageway: its beacons should be delivered to other cars to avoid accidents, but delivery to pedestrian nodes in the same area is meaningless, while in other cases (e.g., a car approaching too fast a pedestrian crossing) delivery to pedestrian nodes is not only useful, but of the utmost importance.

\(^2\)This is not true, since a (human) user can be the owner controller of many nodes e.g., when he is driving a car where multiple devices are present, but this additional complexity does not add much insight into the problem and does not help to correctly formalize it in first instance

This simple description makes it clear that the second issue stated above can be mapped on a problem of distributed firewall, i.e., a system where the rules sets \(S_i\) are distributed among the nodes of the network, which in turn cooperate by enforcing them to the best of their capabilities. The trivial solution of distributing all \(S_i\) to all nodes \(n_j\) is normally unfeasible, since in the general case the cost of analyzing the rules is liner in the set dimension [1] and meaningful scenarios include hundreds of nodes each one with tens to hundreds of desired rules, so that there are thousands of rules to check for handling each packet.

III. FORMALIZING THE PROBLEM

In a nutshell the problem we tackle is how the rule-sets \(S_i\) should be distributed; which are the nodes in the ad-hoc network that can best implement the filtering; and how a generic node \(n_j\) can prune the list of rules it received \(\bigcup_{n_i \in N} S_i\), so that the local filtering function remains efficient. \(N\) is the set of nodes \(n_i\) participating in the wireless ad-hoc network, including mesh nodes if they are present. The final goal is the minimization of some goal cost function, or at least the fulfillment of some goal on that cost function.

Users’ information flow, and is to be filtered, on the topology build for on the ad-hoc network by the selected routing protocol. A natural choice for the distribution of \(S_i\)s is thus to append them as adds-on to routing messages. Initial results discussed in Sect. V are obtained implementing the distribution as extension of OLSR as described in Sect. V-A.

Early works on this subject [2] identified as metric the fraction of packets that were correctly filtered before reaching a destination not wanting them. This is however only a partial metric, since it does not account for the network effort and it does not take into account the cost of filtering at nodes. Without any of these additional constraints and considerations, it is clear that the optimal solution is simply implementing every rule in every node, which we already discussed as unfeasible.

Moreover the metric above is not able to properly differentiate between filtering rules that define undesired traffic, from filtering rules that limit the distribution of outgoing information.

A. A usable definition of privacy

There is an ongoing debate among social scientists on a usable definition of “privacy”. In the past many distinct definitions have been proposed, all of which have shown to be too vague or too narrow if applied to a context different from the one for which they were initially crafted. Lately, one approach that has been raising attention is the one proposed by Helen Nissenbaum and described in [3] called the Contextual Integrity Framework (CIF). Basically, instead of concentrating on the definition of privacy the attention is moved to the definition of a framework that can be used to analyze privacy implications case by case to check if the social norms that regulate a certain exchange of information have been broken generating a privacy breach.
The CIF relies on two basic building blocks, contexts and norms, and tries to build context-relative information norms, that are norms that can regulate the behavior of a person with respect to privacy in a certain situation. Context-relative information norms are defined in terms of context, actors, types of information and principles of transmission:

- The context is the place where the exchange happens, being it a geographical space, a building, a social place (school, hospital...). More than the geographical reference itself, the context is important to define the aim of the exchange of information;
- The actors are described by the roles they play and the features they have. The relationships that intercur between the ends of the exchange are defined as a consequence;
- The attributes of the information are the tags applied to the information that is exchanged and express the kind of privacy attention it requires;
- The transmission principles are rules applied to the current or to previous information exchange regarding the information that is being exchanged (for instance, the obligation to not further propagate the information).

Applying these categories to the analysis of a certain information exchange it is easier to construct the context-relative information norms that regulate the expectations of privacy that the actors have over the exchange and thus verify if a certain action has hindered any of those expectations. CIF has been formalized and used as a base to develop formal methods to verify the respect of privacy policies [4]. In our case, CIF is a useful resource to define generic privacy policies applicable to our context.

B. Formalizing privacy rule-sets

CIF can be applied to the case of privacy management in an ad-hoc wireless network, building on the model of [4]. We discuss how it can be used to manage the problem of the distributed firewalling application. Defining a formal grammar is not our goal at this stage of the work. We only want to illustrate how much a problem can be framed using CIF. We will focus on a specific application: social networking. The nodes form an ad-hoc network that they use to transport the typical applications of social networking (such as chat, as in [5]) together with requests to other possible services such as queries to sensors present in the devices.

In an ad-hoc network the exchange of information can take place with a push of a pull model. With the former case a node spreads information about itself without any request, it is the case of hello messages and routing information sent with proactive routing protocols. In the latter, the information is requested and an answer is produced by the owner of the information or by any intermediary. The request (or a pushed message) may be unicast to a single node or targeted at a group of nodes, responses are generally unicast. A node may act as one of five logical entities:

1) the sender of the message: \( N_s \),
2) the forwarder of the message: \( N_f \),
3) the final target of the message: \( N_t \),
4) the owner of the information: \( N_o \),
5) the generator of the information: \( N_g \).

An information may be generated by a certain node but owned by another. For instance, the request \textit{what is the state of node \( n_i \)?} may be received by node \( n_i \) and answered. Node \( n_i \) acts as \( N_g \). But if \( n_i \) periodically broadcasts its state to the one-hop neighbors the same request may be answered by some other node \( n_j \), which act as \( N_o \) since it does have the requested information. Thus 'own' here is does not mean to have as legitimate property, but only to hold as legitimate retention, which does not necessarily imply further diffusion. Each node will decide to answer, forward, drop or mangle (normally encrypt) a certain message (request or response) according the rules it is aware of.

Let \( \mathcal{R} \) be the set of roles defining the actors, and let a separate tag correspond to each role \( r \in \mathcal{R} \). Each node can then be defined by a set of tags, and these tags may be added to the node information. Roles may be auto-assigned (e.g., student) or they may be negotiated with other nodes (i.e., Part of group \( G \)). In these cases roles and tags are explicitly announced ('sponsored') by nodes for verification. Roles may also be assigned by other nodes (i.e., no more than three hops away from me) and in this case they are not sponsored but may be implicitly verified. For simplicity we refer to tags with a meaningful name, but they might be numerically coded for a specific domain or single ad-hoc network.

Tags are also used to express contexts relative to the nodes. In the scenario of an ad-hoc network they may be based on the position of the nodes (e.g., GPS based), or they may be automatically derived from the environment (i.e., the tag university may be adopted by a node automatically if the presence of the university’s wireless access points is sensed). Geographical position is not a one to one match with how CIF defines contexts, a context is a semantically meaningful value that is used to determine the norms that apply to a certain situation. So, the position may be enriched with user-supplied information (even if they both are at the University, a professor may be in the context working and a student in the context hanging around ... or vice versa!). Contexts are the most difficult value to capture from CIF theory since they assume the same understanding of the implications of a certain situation. Luckily, a context definition is strongly biased by the service it refers to, even if it is on-demand and realized on an ad-hoc network. A social network application on an ad-hoc network in an urban park can implicitly subsume the sharing and leisure contexts, and exclude other critical ones such as hospital or emergency.

The attributes of the communications may be expressed with a code defining the kind of application that is requested and the related subfields. For instance, \( N_o \) may perform a generic request for the state of \( N_i \) (busy, off-line...), or it may request to share a picture specifying the attributes that have been automatically added to the picture by the camera (position and time of the shot).

Principles of information propagation can be simplified to
three kinds. i) Openly accessible, meaning that the information is not subject to restriction and can be given away.  
ii) Localized information, which is limited to a topological neighborhood from the generator: it should not go beyond a certain number of hops or meters.  
iii) Group-based (or attribute-based) restricted. Groups may be intended as interest groups (case iii.a, like a set of nodes interested in receiving a certain information) or groups in a classical access control meaning (case iii.b: the set of node to which the information is restricted).

The group-based information propagation is the most complex (and interesting). In case iii.a, tags may be auto-assigned and help a more efficient distribution of the information. In case iii.b instead, unless we assume that groups constitute connected sets in the network, which is a strong assumption, some technical measure must be added to simple filtering to restrict the access to the information. Imagine a situation in which a group is split in two non-contiguous regions, and an information must be transmitted from one region to another passing across nodes that do not belong to the group. A node that must forward the information from one region to the other can decide to encrypt it using credentials shared among the group. So, the propagation principle restricted to group $G$ corresponds to the application of encryption when the information is to be sent outside of a set of nodes that all belong to the group.

A filtering rule is defined by a context, one or more possible roles, the attributes of the communication this rule applies to, and a principle of propagation. Any rule drives a specific action for the information, in general answer, forward, drop or mangle (as in the case of application of cryptography).

Let’s consider an example. Node $n_i$ has defined the following set of rules:

- **R1** If the context is not *university* for both ends, the role of the requester node is *student*, the information attribute describes a generic chat protocol and the principle of propagation is *any*, then drop the request;
- **R2** If the information attributes define a request for a picture taken inside the Smith's home area (specified by GPS coordinates) and the requester is not in group SmithFamily then drop the request;
- **R3** If the information attributes define a response to a picture taken inside the home area (specified by GPS coordinates), the requester group is SmithFamily, and the principle of propagation is *encrypted*, then encrypt ID-based public key identified by SmithFamily if possible or drop;
- **R4** If the role of the receiver is not less three hops away from $n_i$ and traffic type is routing messages, then drop the message.

Let’s assume that at steady state the rules and the tags of each node are known in advance to the other nodes. Rule R1 means that a node $n_j$ that act as forwarder $N_f$ and is in the condition to forward a chat request to $n_i$ will check that the tags *university* are applicable to both nodes, that the requester is a student and thus will let the request pass. The same traffic will not be allowed if one of the endpoints do not share the *university* tag. For instance, a professor may use this rule in the time intervals he allows students to contact him for tutoring. When the professor is home, the context *university* does not apply to him and the request should be dropped.

Rules R2 and R3 express the fact that pictures taken inside the home of the Smith’s family should not be exported outside the family, so that no response will be forwarded to nodes outside of the group and information must be encrypted with the ID-based public key identified by SmithFamily.

R4 specifies that routing messages generated by node $n_i$ should not be forwarded more than three hops away from $n_i$. Consider the case of a network based on OLSR routing. Each node produces 1-hop HELLO messages, MPR (Multi Point Relay) nodes send TC (Topology Control) messages that include the presence of all the MPR selector nodes (nodes that have selected the specific node as MPR). Rule R4 imply that $n_i$ cannot be an MPR, since MPR must send TC messages to all the network. With this rule, if a node $n_j$ which act as $N_f$ is an MPR and receives a TC containing the IP of $n_i$, which act as $N_g$ for this information, regardless of the actual sender of the message, it will check if the hopcount is higher or equal than two (two hops from the $n_i$ MPR, which is acting here as $N_g$, to $n_j$ MPR, plus one from $n_i$ to its MPR) and drop the message if it contains only references to $n_i$ or mangle it (erasing the reference to $n_i$). This way, $n_i$ will be invisible to the part of the network farther than 3-hops from it.

### C. Introducing constraints

Let’s assume three simplifying hypotheses: in the wireless ad-hoc network there is a way to efficiently:

- **H1** Exchange tags;
- **H2** Exchange rule-sets $S_i$;
- **H3** Create groups and distribute the corresponding cryptographic material.

For instance a network where nodes may periodically use an Internet connection to synchronize their information on a web-based platform. It is the case that is most similar to the current geolocalized social networks, in which nodes interact with short-range communications but use the centralized web server for the exchange of large data-sets.

What are the problems to be resolved in this context? Let’s consider two sub-cases:

- **Fully ad-hoc network:** All the nodes apply the filter at their input chain and at their forward chain (using IPtables terminology). In this case, traffic is filtered by the intermediaries and even if a request arrives at destination, the privacy policies implemented at destination are always applied.
- **Mixed mesh/ad-hoc network:** Not all the nodes implement filters, but some nodes have a higher management role in the network. This happens when the network is not only ad-hoc but it is supported by mesh nodes that ensure a better coverage and provide further computing and communication capabilities. The manager of the mesh
nodes can implement filtering rules in order to limit the impact of certain traffic on the network, or forbid the disclosure of certain environmental information, or simply to offer a richer and safer environment to its customers.

In the first case the actions of forward/drop on a certain pull request applied by an intermediary do not have privacy consequences. Filtering on these packets is applied only for performance reasons, that is, avoiding to route traffic that will be dropped at destination. For push traffic instead, even the actions forward/drop impact the privacy of \(N_g\) and, of course the efficiency of the network. The actions answer/mangle performed by an intermediary node always have impact on privacy since they spread or modify information on behalf of \(N_g\).

One problem arising regardless of the scenario is the growth \(S\). The more tags are used by the nodes, the more the rules may be elaborate, the more complex are the services offered, the more rules they might support. Some rules may be automatically generated by the platforms and updated using user's feedback. Consequently even with few tags and services the rule-set of a node may grow to hundreds of rules. In a network made of hundreds of nodes, \(S\) will contain tens of thousands of rules, since the rules are per-node. The computational burden for such an elaboration is very high, so that the filtering at node \(n_j\) must be done on a reduced rule-set \(F_j \subset S\). Reducing rule-sets may mean aggregating atomic rules in more generic ones or pruning the rule-sets from the less used ones.

Aggregation of rule-sets is performed using ad-hoc data structures that can be searched with better performances. For instance, firewall rules as well as routing tables can be aggregated (see [1] for a survey of packet matching techniques), but the computational effort of maintaining such data structures is not negligible in very dynamic networks.

Instead, how to prune a rule-set strictly depends on the properties of the network: topology, routing protocol, traffic patterns, mobility models. There are in the literature attempts to use local reduced caches of the possible rule-set [2] or to use specific fast hash functions to represent rule-sets even with stateful firewalls [6]. Still, even under under hypothesis H1 and H2 there are interesting research issues:

- in a fully ad-hoc network with a running routing protocol, what is the local strategy a node should implement to reduce its rule-set in order to achieve the best efficiency or/and the best privacy? As said, applying forward/drop/answer/mangle decision for push/pull traffic has distinct consequences on both efficiency and privacy.
- in a mixed ad-hoc/mesh network what are the consequences on efficiency and privacy of the filtering applied only by the mesh-nodes?

If we eliminate some hypothesis the challenges are increased. If the rules-sets are not known, there must be a way to share them in the network:

- is the cost of sharing complete rule-sets compatible with realistic usage patterns of the network?

- if the rule-sets are only partially shared or shared with only a local part of the network, what are the impacts on the efficiency and privacy of the network?

- is it possible to securely distribute the rule-sets lowering the chances of a denial of service attack by injecting willingly damaging rules in the network?

If hypothesis H1 is dropped the distribution of the profiles of the nodes must be analyzed. While some tags may be distributed together with routing information, what is the impact of the additional traffic and what is the number of tags per node that can be supported?

Lastly if hypothesis H3 is dropped, then the mangle action must be revised and some applications may not be available on the network, unless a distributed PKI algorithm is applied, which is out of the scope of this research. In the rest of the paper, where meaningful, we will consider this hypothesis valid.

IV. STATE OF THE ART AND JUSTIFICATIONS

In this section we are going to justify some of the stating points and assumptions we have done. First of all, we start with the observation that a mesh networking scenario as the one described so far is not yet another potential application for wireless ad-hoc networks, but it is a likely development of current technologies. As a starting point, social networking is a widely used application but its market model has so far been web-centric and centralized. The big players involved in social networking have no interest in distributing their centralized platform but latest advances are going in this direction to exploit local interactions among users. The interactions start with simple recognition of near-by users to more elaborate forms of sharing of data like the one proposed by the startups lokast.com or color.com which focus on the direct sharing of objects between users. Portable game devices from Sony and Nintendo have included p2p networking possibility [7] and in the latest Sony PS Vita the interaction between nearby users has been enhanced compared to previous models. The trend is confirmed by the fact that new standards for ad-hoc networking are being developed (such as IEEE 802.11s) and that other platforms are already present for Android-based portable devices (such as Lokast library from Qualcomm). This, together with the success of leisure-based portable devices (e-book readers, tablets, gaming devices) make us think that there is a concrete possibility to have in the future p2p ad-hoc social networks and applications.

Some of the background assumptions we made need to be technically justified. Given that for this application we consider proactive routing a suitable routing approach we will focus on solutions that better suit the OLSR protocol. Group formation is at the base of the group communications we described and of the sharing of common tags. In [8] a generic group formation strategy based on a proactive routing protocol is described which can be used to create groups based on various parameters (topology, physical distance or specific attributes of the node).
Many papers address the problem of hardening the OLSR routing protocol in order to make it less prone to insider attackers that send fake routing information. In our case this is an even more sensitive point since each node sends filtering rules. The approach that may better fit our scenario is the one explained in [9] which mixes ID-based cryptography with distributed algorithms to perform distributed periodic key refresh. ID-based cryptography is one of the possibilities we introduce in this paper to solve the problem of non contiguous groups. This approach, mixed with the advanced signature scheme proposed in [10] could be used to ensure the integrity and correctness of routing messages.

If a signature scheme is present, then more complex security features can be introduced to tackle another important problem, that is verify the enforcement of the rule-sets. If a node does not enforce a certain rule, it may be because it is not able to do so, or because it refuses to do so in order to leak information. This case can be seen as a variation of the well-known attacks on routing protocols (packet dropping, wormholes...) against which many solutions have been proposed, based on active or passive feedback and reputation schemes [11]. Some of the techniques proposed for routing attacks could be adapted to verify the enforcement of rule-sets, starting from simple passive monitoring of the messages that MPR nodes forward to others.

We have assumed that the quantity of rules is big enough to need some algorithms to prune the rule-sets or optimize their parsing, but how big are rule-sets to be expected? There are studies on user behaviour dealing with the size and quality of rules that a user chooses in order to preserve his privacy in a social network when dealing with location sharing. In [12] and their following works the authors tested the behavior of users that have to define their own rule-sets observing that for location sharing only users have rule-sets ranging from 2 to 10 rules, and that the satisfaction of each user measured with an a posteriori feedback is also a function of the complexity (an thus, of the number of rules) of their rule-set. More rules are better performing but also more cumbersome to manage. To handle complex rule-sets in [13] they introduce the possibility of nudging users, that is, suggest to the user a rule that would fit his behaviour using learning algorithms. This approach would help the user create his own rule-set that is an incentive to use complex rule-sets even for users not practical with their privacy settings. Extending its application to more services than just position sharing (chat, applications, sharing ...) would make the resulting rule-sets rapidly grow. Given this, it is not unlikely to imagine rule-sets made of several tens, up to hundred of rules.

Is the authors’ opinion that what exposed so far confirms that some of the features we described can be achieved with the right mix of present literature. To justify some of the simplifying assumptions we made, imagine that the presented social network may not be an isolated network but an extension of an existent web-based social network, such as the Diaspora distributed social network. In this case, following the federated approach of Diaspora, each user could first join the social network with a web-based procedure, receive an ID and an ID-based key pair (released by one of the well-known servers of the federation), then use the same identity to set-up a distributed extension of the social network to be used without need of direct Internet connection. This approach justifies hypothesis H2 since it is imaginable that the rule-sets may not be auto-generated but they may be chosen from a set of pre-configured rule-sets received when joining the network for the first time. In this case not the whole rule-set but only a numeric identifier would be sent to the rest of the network, which greatly reduces the impact on network resources. Also hypothesis H3 is helped by the presence of off-line interaction, since we have seen that the group formation and hardening of routing protocol can be obtained on top of a key management system.

The idea of a distributed firewall itself is not unexplored even if while firewalls have been used for decades, distributed firewalls have not received the same attention. An initial model of distributed firewall has been proposed by Bellovin et al. in [14] where the firewall is delocalized from a bastion host to the endpoints of a still traditional centralized network. Recently, the subject has received more attention from some authors that approached the possibility of performing a true distributed firewalling. Works like [2], [15], [16] introduce the concepts of distributed filtering and study the performance and initial integration with routing strategies. Other authors, see [17], [18], reverse the problem and propose networks in which communication is possible only if a previous security handshake has been performed by the end-nodes of the communication (deny-by-default networks).

Porting some of the known techniques used in firewalls to a distributed network may be not straightforward at all. For instance, rules can be grouped by certain keys in order to use a better performing data structure than a linear list. In a very dynamic environment the routing tables are constantly changing, and together with the routes also rule-sets are changed, so that keeping rule-set in an efficient data-structure can be a costly operation. A very simple indexing based on destination or source IP addresses can help in the most simple cases but not when the rules are referred to multicast addresses (group-based communications) or in a more generic case where the rules do no target an IP address but may target some higher-level features (like resource IDs).

The problem addressed by research on federated wireless ad-hoc networks is similar to the one we are addressing. For instance, the system depicted in [19] has some features in common with the one we consider but the solutions imagined are targeted at a very different environment, (military network with different constraints) and consequently the solutions proposed diverge enough to be not applicable to our scenario.

A. Filtering in distributed environments

The approach described so far can be specialized to implement a privacy-aware distributed firewall for an ad-hoc network.
While a node that implements the rules described so far is itself a firewall, in this subsection we refer to firewalls with a traditional meaning, that is nodes that perform filtering at network/transport level and simple pattern matching/mangling at higher levels. We recall that a firewall is a network host that implements certain actions on packets described by a set of rules. A rule is defined as a pattern matching part and the corresponding action. Firewalls can be stateless (the decision on each packet is independent) or stateful, in this paper we concentrate on stateless firewalling. We will use Netfilter/iptables terminology in the rest of the paper to describe firewalls. Netfilter/iptables is the software that implements the firewall in kernel and user space of Linux, which is the base of the majority of commercial firewalls and of many mobile applications. It implements a linear list of rules matching parts of a packets and a target that can be an action such as drop, accept, mangle. Functionally, it divides the point where the filter is applied in chains, the input chain is used for packets that are destined to the firewall, the forward chain is applied to packets that are going to be routed through the firewall and the output chain is for packets generated from the firewall itself (see [20] for details).

In this context the communication attributes are the ports used. Services are bound to a specific port, and they are standard and well known (chat, presence, geo location, . . . ). Firewalling can be moved to upper layers in the stack to verify other properties of packets, for instance, identify into routing messages the references to a certain IP address. The roles are IP addresses or groups of IPs. Note however that if no cryptography is applied groups must be in contiguous topological regions. Groups expressed with topological features such as less than k hops away from n_i are easily implementable matching the hop-count in an ad-hoc network if n_i act as N_o, while are instead more difficult if n_i act as N_o \not= N_o.

Since a request to a UDP or TCP service is directed to the node that runs the service itself and the semantic value of the communications attributes is not very rich, we can expect that in general N_o = N_g, unless neighbors use some form of cache. For pull protocols this implies that N_o will not delegate the response to any other node. If an invalid request reaches N_g it will be filtered out and no privacy breach takes place. Firewalls are then a valuable resource to avoid the waste of bandwidth and energy due to the propagation in the network of a request that will be dropped at destination. For push protocols instead a firewall plays an active role in protecting the privacy of the information, since the information is forwarded by nodes others than N_g, but we have to assume that N_o,s maintain additional information associated to rules to correctly enforce them.

V. INITIAL DESIGN AND RESULTS

In this section we report the initial results of the application of distributed firewalling in wireless ad-hoc networks, assuming OLSR is the routing protocol adopted.

A. OLSR-based Network

OLSR has been chosen since its features fit well with the considered scenario. First of all, it is a proactive protocol, which reflects the stateful nature of social networks. When a node enters an OLSR-based network it will quickly build its routing table containing all the nodes in the network. If the OLSR packets are enriched with more informations (the user-id, the current state . . . ) the user will be informed about the other users present in the network without the need to run additional discovery protocols. The effort needed for this feature is the increased size of HELLO and TC messages of a few bytes, which is generally sustainable. OLSR also naturally supports broadcast traffic and, with some modifications, multicast traffic [21] which permits typical social network applications, such as group chat.

OLSR requires more resources compared to reactive protocols, but its scalability can be improved using fisheye strategies. If the mobility of the nodes is low and some latency from initial connection to full connectivity is tolerable, richer information can be added to state messages (such as rule-set parts). Most social applications are used on standard computers but mobile terminals are gaining popularity as means to access the web-based applications. This gives us the freedom to chose a mobility model that represents human behaviors and consider our nodes to be slowly moving (with pedestrian speed, see [22]), thus adding the benefit of a realistic scenario from the very beginning. Moreover, the typical usage pattern of social networks is to keep the social application as a background task of some other more important task for sessions that last from tens of minutes to hours, which allows to trade-off start-up time with other requirements. Under these assumptions OLSR can be tuned to increase the size of signaling messages reducing at the same time their frequency of propagation and radius by adjusting the TTL. OLSR signaling enriched with other kinds of information is also a good example of push traffic on which we can test some metrics to validate the assumptions done in the previous section.

Other features of OLSR are useful to filtering purposes. The knowledge of the full topology can be used to approximate potential fluxes on the network. With this approach the nodes that are more involved with routing may perform more accurate filtering in order to increase the total number of filtered packets. For instance, using a very simple heuristic, the nodes that are MPR can be treated differently from the others.

B. Evaluation Metrics

For the evaluation of the performance of a distributed firewall and in particular of the rule-set reduction and distribution policies, the following metrics can be introduced:

M1) Total number of false negatives on forward/drop targets.
M2) Total number of unfiltered packets measured at destination.
M3) Total number of false positives in answer targets.
M4) Traffic overhead introduced for rule-set distribution.
M5) Node resources overhead introduced and impact on traffic performance.
Metric M1 is meaningful to evaluate the efficiency of packet filters when applied for traffic reduction in case of pull traffic and the efficiency of privacy protection for push traffic. Assuming the validity of hypothesis H1 and H2 the comparison is performed between the number of packets that would be filtered with the full rule-sets always enforced and with the rule-set reduction techniques applied (the metric is calculated hop-by-hop). The same metric can be calculated if hypothesis H2 is dropped. For push traffic the metric indicates the number of packets that will reach nodes that should have not received them given the originator’s will, in this case it is a privacy metric.

Metric M2 is meaningful to evaluate the degree of traffic shaping and consequently of privacy preserving in mixed ad-hoc/mesh networks, where only a subset of the nodes implement firewalling. It is calculated as the number of packets that reach their destination against the global policy, conversely to M1 it is calculated only at destination, not hop-by-hop. It is useful to evaluate the ratio of active/inactive firewall nodes together with the rule-set reduction strategies applied and may be used for push and pull traffic. Metric M3 counts the number of answers that an intermediary gave on behalf of the final destination of the request. It is a privacy metric that can be applied in cases in which firewalling is applied not only at network layer but also at higher layers.

Metric M4 is meaningful when hypothesis H2 does not hold and the rule-sets must be distributed using ad-hoc messages or piggybacked to other signaling traffic. It compares the increase of signaling traffic with and without hypothesis H2. Its value is meaningful when related to the variation of the other metrics.

Finally metric M5 evaluates the cost of enforcing the policies on the nodes resources and on the traffic performance. For instance, the CPU time spent matching a packet against a rule-set produces energy consumption on the nodes and introduces additional routing delay, thus influences the overall round-trip-time.

All the metrics can be calculated as absolute values in consistent scenarios or as relative to the ideal case of all valid hypothesis and full rule-sets enforced.

The validity of hypothesis H3 influences the usage patterns of the network, from a best-effort approach (even in the matters of privacy) to a more solid access-control based content sharing. In case of negotiated groups, more evaluation is needed to verify the correct application of cryptography at each step of the path from source to destination.

C. Preliminary results

Using a modified version of Inet module for Omnet++ simulator the integration between a network firewall and OLSR routing has been developed. At this stage of the work firewalling is limited to IP addresses and UDP/TCP ports and group traffic has not been implemented. A pre-shared list of rule-sets is defined off-line and each rule-set is associated to 8-bit numeric IDs. Each node implements its own rule-set in the input chain adds the corresponding ID to HELLO messages. MPR nodes in turn forward the ID together with the advertised address of the MPR selector.\(^3\)

Two scenarios have been simulated so far, the first one is a mesh/ad-hoc scenario with 15 regularly spaced mesh nodes, and a variable number of ad hoc nodes. All the 15 mesh nodes are actively filtering and the number of ad-hoc nodes is increased from 10 to 80. This scenario has been simulated in order to test metric M2 when there is an infrastructure network that tries to implement a specific rule-set and some ad-hoc nodes that generate and receive traffic. The simulated area is kept constant and the generated traffic is uniformly distributed among the possible ad-hoc nodes. The mesh nodes have a higher communication radius so that they are preferred for long paths, but the increasing the density of ad hoc nodes makes it easier for a node to have routes that do not involve mesh nodes. Figure V-C shows that increasing the density, the number of packets that are filtered at destination (M2 here is measured as a the ratio between the packets arrived at destination with and without firewall on mesh nodes) decreases. We expect these results to be quite dependent on the scenario; however, even with only 10 different scenarios as in Figure V-C the standard deviation (reported in the plot as error bars) is not so high. Even filtering only in the mesh nodes always reaches at least a 50% efficiency on the packets that reach the final destinations, which is an encouraging result.

The second scenario illustrates the results of metric M1 applied to a possible strategy for the reduction of rule-set sizes and of the number of filtering nodes. This is accomplished with a tighter integration between OLSR routing and filtering. Only the nodes that have a number of MPR selector above a given

---

\(^3\)The simulation code is still not public due to its not-yet-complete development, but additional information and the code itself can be obtained by the authors.
threshold actively filter and use the entire rule-set \( S \). The other nodes use a reduced filter where all the rules that correspond to routes longer than two hops are removed. The intuition behind this strategy is that, under the assumption of uniformly distributed traffic, the more MPR selectors a node has the more traffic it will route. If a node has no MPR selectors it will route only traffic for their immediate neighbors. Metric M2 is not shown simply because every node implements at least rules for 1 and 2 hops routes, so that no packet is going to be filtered in input chains. Metric M1 shows instead the efficiency of the global filtering function. Figure V-C shows that in a network composed of 55 ad-hoc nodes this strategy is able to filter on average more than 60% of the packets. Note that this metric is calculated per node per hop, not at destination like M2.

When the MPR selector threshold is set to one, all MPR apply the full rule-set, so that unfiltered packets are concentrated on the first two hops of the traffic. With a higher threshold less nodes apply the full rule-set, but since they are the ones who have been chosen as MPR by more neighbors (so they route more traffic), the decay of performance is no worse than linear.

Obviously there is an uneven distribution of filtering load, which consequently makes CPU load uneven between nodes. Still, having less nodes with full rule-set has the consequences of reducing the time needed for the routing decision and thus limit the average latency on a path.

VI. CONCLUSION

The use of ad-hoc networks to provide services on-demand, i.e., when and where they are needed is slowly rolling out and changing the Internet scenario. This changed scenario expose nodes and users to novel threats and problems; for instance all the “security and privacy” guaranteed by traditional NAT-plus-firewall architectures may fail entirely.

In this work, we have first defined and then formalized problems related to distributed firewalling in ad-hoc networks, taking as reference sample application a social network.

A prototype implementation over OLSR in Omnet++ has been developed and initial results for some metrics of interest show that this architecture is indeed viable and sustainable, though many issues remains open.

So far we have focused on the where and how to apply the filtering rules in a cooperative scenario; however, problems related to rules generation, their distribution and finally their enforcement in non-cooperative environments remain open and draw a clear roadmap for future research.

ACKNOWLEDGEMENTS

This work is funded by The Trentino programme of research, training and mobility of post-doctoral researchers, incoming Post-docs 2010, CALL 1, PCOFUND-GA-2008-226070. For more information see http://pervacy.eu.

A short version of this technical report has been presented as a poster in [23].

REFERENCES


