Waypoint Routing: A Network Layer Privacy Framework

Alfredo Matos, Susana Sargento, Rui L. Aguiar
Instituto de Telecomunicações, Universidade de Aveiro
{alfredo.matos|susana|ruilaa}@ua.pt

Abstract—This paper presents a routing framework that embeds location and communication privacy into the routing mechanisms. It conceals endpoint identification by introducing waypoints, through encrypted routing hints, where each waypoint has knowledge of the next hop, assuring network privacy over several waypoints. Based on IPv6 extension headers and Onion Routing techniques, the network waypoints comply with normal routing procedures, avoiding explicit tunneling or full packet encryption. By focusing on the network as a cooperative entity for privacy preservation, we propose a lightweight approach that can be easily deployed, establishing a good compromise between privacy and optimal routing.

Index Terms—routing, privacy, waypoint, hints

I. INTRODUCTION

The growing pervasiveness of computer networks in daily life turned privacy into a major concern. On the network layer it is important to protect the communication contents, since it transports all upper layer information. It is also important to protect the endpoint and its location, because the IP address uniquely identifies both the peer behind the communication and its topological location. Location privacy becomes particularly important because the topological position expressed in addresses can be converted into geographical location: the hierarchical assignment of IP addresses can be surveyed and mapped, tracing addresses to specific cells with a fixed geographical position. The increase of private and commercial services surveying network locations has led to surprising accuracy for such IP to location services [1].

In this hostile environment, we aim to provide privacy, especially location privacy, to the end-points, thus safeguarding the user behind the communication. To achieve this goal, it is necessary to prevent the nodes in the network (both peer nodes and normal routers) from learning the actual identity (identifiers) of the user and its location.

Most solutions that address network privacy threats usually rely on the concepts of Chaum Mixes [2] to provide anonymity, mitigating the side effects of using IP addresses: if packets are anonymous, the observed address cannot be traced back to the original sender, voiding location information, since it does not concern the message sender. However, such protocols usually require trade offs between privacy and performance, or sacrifice compliance with standard routing protocols. The most prominent example of such solutions is Tor [3], which delivers privacy at a cost. Tor employs multiple encryption layers on every packets to conceal the packet origins and routing hops, incurring in a hefty performance penalty. It is also used on top of IP, due to its non-standard requirements, such as circuit establishment and fixed data cell size. The performance overhead and the lack of integration with routing schemes (IPv4 and IPv6) undermine the adoption of such privacy solutions on the network. Therefore, privacy has a price that only some end-users pay, becoming a cooperative peer-to-peer effort. This is what we call pushing privacy support into the edges of the network, resulting not only in crippling performance, but also on compromised privacy: packets travel to the network edges where forwarding is performed by untrusted endpoints in a peer-to-peer system, that can inspect the ongoing traffic [4]. Distributed environments also pose severe difficulties concerning lawful interception, which is a requirement for any commercial network.

To create a truly adoptable solution, we must reduce performance costs, allowing a solution that can be easily deployed inside the network, compatible with current routing schemes. In this scope, we propose Waypoint Routing, a lightweight framework that through a novel cryptographic routing scheme, enables treating privacy as a value added service (VAS) that can be provided by the network. We provide end-user privacy by hiding the original sender, through several encryption points, mandatory Waypoints (WP) inside the network, making it impossible to identify the end user or his location. This is achieved by using encrypted IPv6 Routing and Extension headers, based on the concepts of Onion Routing [5] and Tor [3] mechanisms. The encrypted extension headers define lightweight overlay privacy routes, where each router is aware only of the next hop in the route, at a reduced cost, thus minimizing performance impact because it requires less encryption (only extension headers are encrypted). Also, by keeping the packets inside the network, and routed through trusted entities, we avoid traffic inspection by untrusted peers [4], along with smaller delay.

To measure the impact on communication delay and the advantages of the proposed solution, we introduce the notion of routing optimality in a privacy context, and compare it to peer-to-peer privacy systems, demonstrating how the use of core network entities can benefit the current privacy landscape. The remainder of the paper is organized as follows: In Sec. II we present and discuss existing proposals. Sec. III presents the cryptographic and routing mechanisms, whereas in Sec. IV we describe the complete architecture for WP Routing privacy support. After an evaluation in Sec. VI, we conclude the paper in Sec. VII by summarizing the results and discussing future work.

II. RELATED WORK

Location privacy has been mostly addressed in absolute geographic location environments, particularly with GPS technologies [6], [7], [8]. These solutions show that introducing confusion [7] or imprecise location [6] benefits location privacy. While not directly applicable, such solutions provide a better understanding of location privacy issues, especially considering that IP addresses can yield surprisingly accurate geographical position, available from public services [1].

To address location privacy issues stemming from topological information, several approaches have been proposed. Based on Locator/Identifier split, Matos et al. [9] proposed HIP Privacy Extensions where a hierarchical framework hides the real network location of nodes within a topological protection area covering a large address space. Location privacy is provided by reducing or removing the topological meaning from addresses. However, this requires widespread network support, relying on forwarding agents. Also, its hierarchical nature and mobility support can be approximated
to hierarchical mobility schemes such as Hierarchical MIPv6 (HMIPv6) [10], or localized schemes such as Proxy MIPv6 (PMIPv6) [11], which also reduce the topological information carried by the IP address while increasing location privacy.

Another approach is to conceal node addresses altogether, thus removing location information and protecting privacy, as proposed by Chaum Mixes [2]. In this case, message receivers are not able to determine the original sender because the message is anonymized by the mix, through cryptographic mechanisms. This concept has been applied in Onion Routing [5] and Tor [3] using the notion of encrypted virtual circuits. The circuit progressively decrypts the routed packets, according to a layered construct: the packet is encrypted many times over by the sender, and is successively decrypted towards the destination, providing anonymity and privacy. However, Tor favors user enrollment as routers. Consequently, traffic reaches the network edges, leading to reduced privacy and suboptimal routing. It also redefines transport mechanisms that were already solved in the network stack (e.g. fragmentation). While Tor mechanisms are flexible (e.g. route telescoping provides reusable circuits) and secure, they require that users become routers that perform full packet encryption, leading to performance issues.

However, by looking at the anonymous communication field, there are several solutions which can provide similar results. As highlighted in the recent survey on anonymous communications by Ren and Wu [12], solutions such as Crowds [13] and Tarzan [14], randomly forward messages to different network nodes, thus introducing confusion in the communication, and consequently privacy, given that a node is not able to determine the real origin of the packet. Interestingly, Tarzan follows a combination of Tor and Crowds concepts, but defines a peer-to-peer overlay network.

Nevertheless, among the mix net approach derivatives, Tor is still considered as providing very good mechanisms for privacy and location protection because it conceals any user information concerning addresses. We believe that by relaxing the requirements introduced by Chaum Mixes, where the inputs must differ from the outputs, along with Tor requirements that mandate full packet encryption, we can provide a solution that solves both user and location privacy using IPv6 mechanisms, remaining attractive to network providers, and coming a step closer to a deployable solution.

III. WAYPOINT ROUTING

Waypoint (WP) Routing is the mechanism that enables introducing waypoints in the communication path, that are only aware of the next hop in a route and replace source and destination addresses, diluting any information beyond next and previous hop. This process conceals the packet origin and consequently the sender’s identification and location, which is tied to the original address. In fact, no assumptions can be made about the observed addresses regarding the original source and destination, inline with the Chaum Mix [2] approach. Replacing addresses is only possible through the use of encrypted routing hints included in IPv6 extension header options, as shown in Fig. 2 and described in the following sections.

A. Overview

As seen in Figure 1, an overlay route is defined by several Waypoint Routers (WPR) that anonymize the traffic flowing between the end-user, the Waypoint Client (WPC), and a selected destination. To establish the route, the WPC contacts each selected WPR, previously retrieved from a directory or discovery service as proposed in Sec. IV-A, establishing authentication and shared cryptographic material. The selected WPRs compose a virtual circuit defining how packets flow (i.e. which waypoints are visited).

To use the circuit, a client encrypts the next hop as well as the destination address, placing them in an IPv6 extension header and forwarding the packet to the first WPR in a route. Upon receiving a packet, each WPR decrypts the addresses conveyed in the extension header, the Routing Hint, thus determining the next hop which becomes the new destination. To finish the forwarding process, the WPR replaces the routing hint (in the extension header) with the one corresponding to the next hop, as well as the packet’s source and destination. The WPR also decrypts the Circuit Identifier, as discussed in Sec. III-B, to ensure that the packet reaches the proper destination. Both these fields have the same size of an IPv6 address, fitting in the IPv6 extension header, shown in Fig. 2, which is transparent to standard routers.

The Waypoint Routing process can be summarized as a cryptographic source-based routing mechanism for IPv6, where the packet’s source and destination change at each WPR. The routes, defined in Sec. III-C, can be established during the authentication process using the shared keys between WPC and WPR (obtained through a Diffie-Hellman exchange) and are composed by the encrypted hints and reusable IP-level circuits, as shown in Sec. III-B.
The two main WP Routing concepts are hints and circuits, which must be integrated into the routing infrastructure. These mechanisms, compatible with standard IPv6 routing rules, are necessary to properly route packets, as presented below.

\[ H = E_k(IP_T). \]  

(1)

The Routing Hint, defined in Eq. 1 and present in each packet, encrypts the address of the next hop in the route (T). According to IPv6 destination option rules, it is processed by the packet’s destination, which decrypts the hint to obtain the next WPR in route. Therefore, in general terms, a node WPRn will receive a hint Hn, corresponding to the encrypted address of the following WPR, IPWPRn+1, with key kn. The hint Hn will be stored at the previous route, WPRn-1, and sent to WPRn, yielding the address of WPRn+1. This process transforms WPRn into a shield that anonymizes communication between adjacent routers. The routing process is further clarified in the next section.

However, if only the next hop was included in the packet, a different circuit would be necessary for each target, as the only available mechanisms to differentiate targets would be the associated keys (which define a circuit). To reuse circuits along the path, we introduce the Circuit Identifier (CID), inspired by Onion Routing [5]: the CID encrypts the final destination, eliminating the need to keep state on every WPR for a specific target and allowing the reuse of circuits between routes (which only requiring hop-by-hop hints). It uses several layers of encryption, as described by Eq. 2, one for each WPR, ensuring that packets traverse the defined route, and that the CID changes (through different encryption keys) between hops (preventing tracking based on the CID), forming a hint index 2, used where circuit multiplexing is required. Wherever multiplexing is not required, the WPR decrypts the CID and forwards the message to the hint stored for the (idxn, CIDn) as described before, resulting in a default hint behavior (each required multiplexing operation should be explicitly registered at each WPR). In fact, with the introduced CID, extending or splitting a route (similar to route “telescoping” [3]) only implies registering a new hint and a corresponding CID in the required WPR, eliminating the need of an entirely new route. This process also guarantees that the global target is only obtainable trough the collaboration of all selected WPR. This raises the number of encrypted addresses in the packet to 2, where one is a hop-by-hop encrypted address, the routing hint, and the other is the encrypted final destination, the CID.

### C. Routes

In order to understand how routes are formed, and the exchanged information (Hints and CIDs), we present an example of a three WP route setup and the associated forwarding process. The example is shown in Fig. 3, representing the information present at each involved element. Reaching this state is an iterative process: first, it is necessary to establish a WPR route including WPR1, to reach T1, then it is necessary to include WPR2, forming a two WPR route, and later WPR3, forming the final route, with three elements.

To start the first phase, WPC contacts WPR1, establishing a symmetric key (k1) for hint encryption, through an authenticated Diffie-Hellman exchange. This process is repeated for every WPR. To facilitate the key retrieval mechanisms at the WPR, a key index, idx1, for WPR1, is also agreed upon for future inclusion in the messages.

To reach T1 via WPR1, the WPC encrypts the address of T1 with key k1, forming the routing according to Eq. 1, where routing hint H1 is the result of encrypting the IP address of T1 with key k1. The hint is then inserted into an IPv6 extension header, included in the packet. At this point, since the Hint is similar to the CID, the CID can be omitted, indicating that WPR1 is in fact the exit router. The packets are sent to WPR1, with the required index, idx1, consistent with Fig. 2.

To increase privacy, the client should include more WPR elements in the route. Therefore, the route can be extended to WPR2, where the client registers, establishing (idx2, k2). Consequently, a new hint must be added at WPR1, in the form of E_k2(IP_T), to be included in the extension header when WPR1 forwards packets to WPR2. Also, the new CID is the target address, encrypted by k2 and k1, respectively. When WPR1 receives a packet, it decrypts the hint present in the extension header, E_k1(IP_T), with the corresponding key, indexed by idx1, thus determining the next hop. Before forwarding the packet, WPR1 decrypts the CID, concluding that it is not the last WPR in the route. As a result, before forwarding the packet, the encrypted CID, E_k2(IP_T), is copied into the hint field.

To introduce WPR3 in the route, the process is similar: the client registers at WPR3, obtaining (idx3, k3) and updates the hint present at WPR1, which becomes E_k2(IP_WPR3). Also, the initial CID becomes subjected to three layers of encryption, shown in Fig. 3 as E_k1(E_k2(E_k3(T1))). At this point, when WPR2 receives a packet, it decrypts the contained hint, E_k2(IP_WPR3), determining WPR3 as the next hop. As it has no further hints for this CID, it decrypts the CID and inserts it into the Hint field, corresponding to E_k3(IP_T). The behavior at WPR3 requires decrypting the received hint,
resulting in $T_1$. At this point, since no CID was provided, $WPR_3$ becomes the exit node, performing Network Address Translation (NAT) for the WPC, the last shield in the process. The final state of this process is represented by Fig. 3, which details the defined route using three waypoints. It shows the shared keys, the information present at each point, and the content of the headers on each hop of the route.

IV. PRIVACY AS A SERVICE

The proposed Waypoint Routing mechanism enables a lightweight approach to privacy at the network level. By requiring less encryption and being seamlessly integrated into IPv6, it becomes simpler to deploy. The reduced performance cost, analyzed in Sec. V, allows privacy to be deployed inside the network, on core routers, as opposed to the edges, by end-users. This can contribute overall adoption and also enables a new paradigm of perceiving privacy as a Value Added Service (VAS). Because it can be delivered by existing network providers, Waypoint Routing permits the creation of a Privacy Service (PS). The PS requires the introduction of a logical entity, the Privacy Controller, that together with a set of WPR, forms the PS architecture illustrated in Figure 4, thus completing the framework.

A. Privacy Controller

As part of the PS, the Privacy Controller (PC) is the entity responsible for managing the privacy infrastructure composed of WPRs, and for providing end-user authentication and access control. As the primary privacy control entity, the PC handles WPR discovery, allowing route establishment. Moreover, it can contribute to hint and address management in different scenarios.

As a service, the PS shares a contract with its users, enabling authentication and access control to the WP routing service. The trust association created between the provider and user leads to several outcomes: as a service, the PS must meet often ignored legal requirements, such as lawful interception and data records preservation; also, the user enjoys a new privacy environment, where the PS is contractually bound to provide privacy, where in other scenarios this is a user provided best-effort feature. The second core PC competency is to enable route setup by providing the user with WPR information. By managing a set of WPRs, the PC can easily interact with the user to provide an adequate set of WPRs for the desired destination. This takes a centralized view of WPR discovery, putting PS in the driver seat of route establishment. It is important to note that, when requesting routes, the WPC is responsible for managing the privacy infrastructure composed of WPRs, forms the PS architecture illustrated in Figure 4, thus completing the framework.

B. Route Selection

Route (and WPR) selection can be one of the biggest contributions made by the PS, through the PC. We separate routes into explicit, User Generated Routes, and implicit, Service Generated Routes. In User Generate Routes, the endpoint undertakes the bulk of the effort for WPR selection and route setup, using as many PS as desired. Here, the PC functions as a WPR discovery service. In Service Generated Routes, the PS can take advantage of the Hint routing mechanism to provide on-demand and implicit user-independent routes.

Using explicit routes, the WPC has the responsibility of establishing the route. This implies authenticating at the PC and obtaining a list of WPR, highlighted in Fig. 4. The node can include an IP address in the Route Request as an indication of destination, obtaining routers closer to the optimal solution provided by IP mechanisms (see Sec. V-B). Afterwards, the WPC contacts each selected WPR, authenticating and establishing the required keys. In this scenario, the PS facilitates WPR discovery. However, the node can use several PS for a single connection, increasing his privacy. Service generated routes are implicit routes that can be used as a means of increasing user privacy, without requiring the user to actively participate in the process. This comes as additional privacy provided by the PS, without involving the user. This is only possible due to the characteristics of the proposed cryptographic solution presented in Sec. III. Given that hints are defined on a hop by hop basis, at any point the WPR or PC can extend the existing route by pushing forward its hint, and generating a new hint towards the WPR that will be receiving the original hint. This process is deemed subpathing. It should be noted that, because the original sender is not aware about added WPRs, it will not share a key with them and will not be part of the CID layered encryption. To solve this issue, the WPR in the extended subpath must not attempt to decrypt the CID and only forward it.

V. EVALUATION

To understand and discuss the benefits of the Waypoint Routing, we must evaluate its performance. We provide a two-fold analysis based on overhead and on path optimality, which determines how “close” the privacy aware routes are to the optimal ones obtained with current routing mechanisms. We later show how this proves to be one of the Waypoint Routing main advantages.

A. Performance

To assess the overhead, we perform an analytical evaluation of the control overhead in data packets, and a ratio of encrypted bytes over transmitted information, allowing a better understanding of the privacy costs. Assuming a packet with the Maximum Transfer Unit (MTU) size of 1500 bytes, we calculate the percentage of control overhead. In our scheme, a packet is composed by the IPv6 header, 40 bytes long, and two IPv6 options, of 24 bytes each: a modified routing
header to include the hint, and a destination option to include the Circuit Identifier (conveyed in the reserved space). This represents a 48 byte increase on maximum size data packets, which is a 3.2% percent overhead increase. In other solutions, like Tor, this overhead is larger: Tor uses a 500 byte fixed cell size for data, which is then appended with headers, compounding a 960 byte overhead (not factoring in IPv4/IPv6 headers). This represents 65%, 8% overhead increase. As far as encryption is concerned, the proposed solution mandates that the two hints are encrypted at every point. This requires 32 bytes at most, while other solutions require much more, like Tor which encrypts/decrypt the 500 byte cell at every hop (not counting management data structures).

B. Path Optimality

We propose a path optimality measure, which is the difference in path cost between the optimal route defined by the routing mechanisms (such as RIP, OSPF, or BGP), and the “privacy-aware” route. The objective is to provide a broad idea of how privacy impacts the route, and then to draw conclusions on the overhead imposed on the users to achieve privacy. We compare, in a simulated environment, three different approaches against the optimal route: 1) an edge Path, which is a random route using the networks edges (end-to-end routes, as they exist in the current Tor deployment); 2) a core path, using random routes based on the network core (to support privacy in the network); 3) and the composed path, which is a simple route selection mechanism that uses network knowledge to build privacy-aware routes minimizing path cost, resulting in less overhead.

In the simulation, we consider the network to be a square node matrix, shown in Fig. 5, where edges (dashed) represent endpoints and the inner matrix nodes represent routers, with fixed coordinates. We define the path cost as the cartesian distance, calculated hop-by-hop. In a three-node route, the path cost would be the sum of the cartesian distances between each of the three nodes. The optimal path cost is the cartesian distance between source and destination, a simplification of the shortest path. The edge path uses random edge-nodes to act as WPRs (only dashed nodes in Fig. 5, whereas the core path resorts to random core nodes (inner matrix nodes). The composed path, in the simulation, uses a divide and conquer strategy: it selects a mix of nodes in the source’s quadrant, a center node, and nodes in the destination’s quadrant, forming a composed route. Every result presented is the average of 10⁴ executions in a custom simulation environment.

Fig. 5 presents the findings concerning path cost, with a 3 hop route and varying network size. This shows that the core path presents an overhead decrease of 28.4% on average, when compared to the edge path. Even more, the composed path reduces the average overhead by 69.6% when compared to the edge path approach, and 57% when compared to the core path. The composed path approach is the closest to the optimal path, with roughly double the path cost.

We apply the same scenario on a fixed network size of 10⁴ (100x100 matrix) nodes and varying route size, as shown in Fig. 7. This graph shows the same type of gains: the core path strategy decreases overhead in more than 27.4%, while the composed path brings the overhead down by more than 65%, when compared to the edge path.

With a generalized performance increase, the exciting conclusion is that using the network core (e.g. both core path and composed path approaches) can lead to a significant performance boost and overhead reduction, which in turn can result in better adoption.

VI. DISCUSSION

In the previous sections we presented the Waypoint routing concepts and architecture to define a new routing infrastructure that enables privacy support, along with an evaluation of the proposed solutions. Derived from Chaum Mix concepts, our proposal resembles other mix network approaches such as Tor, by anonymity standards. However, by preventing packets
from reaching the network edges (end-hosts), we provide a more effective privacy solution that is not exposed to the perils of malicious exit nodes or traffic inspection [4]. Instead, we rely on the PS, which is legally bound to secure user data. This brings the network operator back into the privacy game, enabling it as a privacy provider, capable of delivering optimized traffic and privacy data paths stemming from his deep knowledge of the network. This not only provides better performance, but removes the peers from providing security. While in theory anonymity provides a better solution, in practice malicious exit nodes can subvert the system, whereas here the exit nodes (and all peers in the path) are trusted. Thus, the user is trading part of his anonymity towards the different privacy services for trusted privacy. This also proves to be an advantage over peer-to-peer systems like Tarzan, because here the WPR entities can be certified and trusted, whereas in systems like Tarzan, and even Tor, attackers can try to operate nodes to gain knowledge about the ongoing communication, and even inspect traffic as the last hop.

Beyond the privacy gains, the other key advantage is performance. When compared to approaches that favor user involvement, like Tor, the evaluation showed that, just by moving from the edges to network routers can yield a 30% overhead decrease. And when applying a simple strategy of quadrant allocation, the overhead can be reduced by as much as 70%. This evaluation showed that the approach has a significant relative advantage by allowing WPR selection, further highlighting the benefits of converting privacy into a service, provided either by the network operator or by a VASP, which only benefits from network knowledge and smart route construction. Also, by diversifying WPR and PS, user privacy can increase due to the added confusion.

When compared to Tor, layered encryption in WPR is restricted to the global hint, while hop hints use normal encryption. This shows that the encryption in our scheme is only a small fraction of what is required in Tor, which encrypts the entire cell. Also, by working with IP-level mechanisms, as opposed to Tor that works on the transport layer, using only IPv6 headers results in massive overhead reduction, from the estimated 65% of Tor to an acceptable 3%. We conclude that the proposal incurs in a privacy overhead penalty, but with significantly better results than other proposals due to the low amount of extra information required. However, one of the most fundamental differences is that Tor does not make assumptions on how the Onion Routers are selected, whereas Waypoint Routing makes strong assumptions on the architecture definition, to keep traffic out of the network’s rings, and to maintain communication on an optimal and secure level (avoiding traffic from being routed by endpoints). If we approximate Tor to the random model analyzed in Sec. V, and Waypoint Routing to the other analyzed variants, core path and composed path, we can see that a core-oriented scheme can boost the performance of the overall privacy scheme, greatly reducing the performance impacts on expected delay, in some cases as much as 70%.

VII. CONCLUSIONS

In this paper we proposed a source based routing mechanism for IPv6 using encrypted addresses within extension headers, concealing unique addresses and location through the use of Waypoint Routers. Beyond the end-user benefits, and due to its lightweight properties, the framework fosters privacy as a value added service. This promotes using the network infrastructure as the driver for privacy, getting the network operator off the sidelines in the process and leading to considerable privacy and performance gains.

The proposed Waypoint Routing mechanism allows scalable performance by bringing privacy mechanisms back into the network layer. In doing so, we allow the construction of “privacy-aware” routes, based on waypoints. WPR uses less encryption, creating less overhead, and routes closer to the optimal and native network mechanisms. Even though the simulated mechanisms model simple route scenarios (starting with an iterated triangle inequality), they already indicate that transferring privacy towards the inner network nodes, followed by smart route selection, can lead to a smaller performance impact resulting from privacy enhancing technologies.

In the future, we aim at exploring route construction and maintenance, conducting further experiments and simulations based on network graphs that better model the networking reality, consolidating the presented results. Furthermore, we expect to support the proposed architecture with packet level simulation, in order to corroborate the analysis carried out in this paper. Nonetheless, the encouraging results and privacy benefits already allow us to conclude that the proposed solution can contribute to better adoption of privacy technologies.

ACKNOWLEDGMENTS

We would like thank Emanuel Fonseca, Rodolphe Marques and Rui Ferreira for the insightful discussions and fresh ideas.

REFERENCES