Impact of Network Topology on Anonymity and Overhead in Low-Latency Anonymity Networks

Claudia Diaz[1], Steven J. Murdoch[2], and Carmela Troncoso[1]

[1] K.U. Leuven/IBBT, ESAT/SCD-COSIC
firstname.lastname@esat.kuleuven.be
[2] Computer Laboratory, University of Cambridge, UK
Steven.Murdoch@cl.cam.ac.uk

Abstract. Low-latency anonymous communication networks require padding to resist timing analysis attacks, and dependent link padding has been proven to prevent these attacks with minimal overhead. In this paper we consider low-latency anonymity networks that implement dependent link padding, and examine various network topologies. We find that the choice of the topology has an important influence on the padding overhead and the level of anonymity provided, and that Stratified networks offer the best trade-off between them. We show that fully connected network topologies (Free Routes) are impractical when dependent link padding is used, as they suffer from feedback effects that induce disproportionate amounts of padding; and that Cascade topologies have the lowest padding overhead at the cost of poor scalability with respect to anonymity. Furthermore, we propose an variant of dependent link padding that considerably reduces the overhead at no loss in anonymity with respect to external adversaries. Finally, we discuss how Tor, a deployed large-scale anonymity network, would need to be adapted to support dependent link padding.

1 Introduction

Anonymous communication systems protect the privacy of their users by hiding who is communicating with whom. These systems support applications with strong privacy requirements such as e-voting protocols, intelligence gathering (e.g., law enforcement agents infiltrated in criminal organizations) or high security military communications. Additionally, anonymous communication systems help individuals in difficult situations (e.g., journalists who must protect their sources) and provide privacy for ordinary people seeking to protect themselves from unwanted eavesdropping. The importance of such systems is increasing, and the largest deployed anonymity network, Tor [6], has attracted an estimated 250 000 users.

Many network services, such as web-browsing or online chat, require low-latency communication to remain usable. Low-latency anonymous communication networks are vulnerable to timing analysis, which can be performed by a

passive adversary to find correlations between streams and uncover communication partners [10, 12]. Furthermore, an active adversary can trace communications by embedding a 'watermark' on the packet flow by delaying, dropping or adding packets to influence these timings [9, 24].

A common solution to thwart timing analysis is the use of padding, i.e., dummy packets indistinguishable from (encrypted) real data. In this paper we consider Dependent Link Padding (DLP), a variant of padding in which the amount of dummy traffic generated at the output of a node depends on its input traffic.

We examine low-latency anonymous communication networks that implement DLP. We find that the topology of the network has a strong influence on both overhead and anonymity. Cascade networks introduce the lowest overhead, but at the cost of poor scalability in terms of anonymity. Fully connected networks (Free Routes) offer high anonymity, but suffer from feedback effects that cause huge overhead. Stratified networks are the best anonymity vs. overhead trade-off. Of all topologies, this provides the best level of anonymity, and its overhead is much lower than Free Routes. We introduce a restricted variant of the Stratified topology that further reduces the overhead at almost no cost in anonymity. Moreover, restricted topologies have better scalability.

In anonymity networks, connections between two routers are commonly encrypted and carry multiple data flows. We propose Reduced Overhead Dependent Link Padding (RO-DLP), a variant of dependent link padding that takes advantage of this property. RO-DLP provides the same level of protection as DLP towards external adversaries – who can observe communications but do not control any router – while substantially reducing the overhead. In the case of Stratified topologies the overhead factor is reduced from 27 using DLP to 8 using RO-DLP, and in its restricted version the reduction is from 23 to just 1.5.

Finally, we argue that, while the onion routing network protocol used by Tor supports padding, it is not compatible with DLP. We outline the modifications that are needed for supporting dependent link padding, and discuss their practical implications.

The remainder of the paper is organized as follows: we give an overview of anonymous communications, padding, and anonymity metrics in Section 2. Our system and adversary models are presented in Section 3. Section 4 introduces RO-DLP, our variant of dependent link padding. Section 5 describes our experimental setup, and we present the results in Section 6. We discuss the applicability of dependent link padding to Tor in Section 7. Finally, we conclude in Section 8.

2 Background and Related Work

Low-latency Anonymous Communications. Goldschlag, Reed, and Syverson introduced *onion routing* in 1996 [8], and a second generation protocol [6] has been implemented in the Tor network. Onion routing is designed to provide a bidirectional, low-latency anonymous communication channel that can be used for applications like web browsing. Onion routers perform cryptographic opera-

tions on the data they relay, so that the relationship between input and output data packets cannot be inferred from analyzing their content. The feasibility for delaying and reordering packets in onion routing is however limited by latency constraints, and therefore incoming and outgoing packets may be linked in these systems by means of timing analysis or end-to-end correlation attacks [10].

An important property of a multi-hop anonymity network is the *network topology*. One approach is Cascades, as adopted by AN.ON/JonDo [1], where clients select one out of several entry routers, but after that point the path through the remaining routers is fixed. An alternative is Free Routes, as adopted by Mixmaster [11], Mixminion [4], and Tor [6], where all routers in the network are connected to each other. Intermediate solutions have also been proposed, such as Restricted routing topologies based on expander graphs [3], or Stratified networks [7]. Dingledine *et al.* [7] showed that the topology of a high-latency anonymity network has a significant impact on traffic analysis resistance, reliability, scalability, and resistance to compromise. However, neither Cascades nor Free Routes have been shown to be conclusively superior, and the issue has long been a matter of debate [2].

Padding to Resist Timing Analysis. Let us consider a low-latency onion routing network that carries data flows of variable rate. To satisfy quality of service requirements, packets cannot be delayed too much, or dropped. Therefore, to conceal the relationship between incoming and outgoing flows, *dummy traffic* (padding) must be added to the data flows. Data packets leaving each node are augmented by *dummy packets* which the adversary cannot distinguish from (encrypted) real data packets. In addition, the start and end time of the flows must be obscured to prevent traffic analysis attacks based on correlating the timing of these events [12]. This can be achieved by synchronizing session start and end between all clients [13].

With respect to the rate of the padding, research in this field has centered on *Independent Link Padding* (ILP,) where all flows in the network are padded to a pre-arranged rate [13, 19, 21]. Because the timing and rate of packets in outgoing flows is not dependent on the timing and rate at the input, an adversary cannot correlate inputs and outputs. These padding strategies are however impractical if the traffic flows being routed by the network are bursty (e.g., web traffic), as any lulls would need to be filled with padding, at the same rate as the maximum throughput.

A more promising approach is *Dependent Link Padding* (DLP.) As with ILP, all traffic flows leaving a router are at the same rate, so as to provide timing analysis resistance. However, unlike ILP, this rate is different for each router, and it is a function of the traffic it is routing. This approach permits the amount of padding to be reduced, because when there is no input traffic, no output traffic needs to be generated. Similarly, bursts of traffic are permitted, and the burst is transmitted on all outputs.

An algorithm for performing DLP, whilst guaranteeing a maximum latency Δ at each node, and minimizing the amount of padding, was independently discovered by Venkitasubramaniam and Tong [22] and Wang *et al.* [23]. Their

algorithm is to, when a packet is received at time t, check whether a padding packet has been scheduled to be transmitted on the corresponding outgoing link. If it is, the padding packet is replaced with the real packet. If not, the real packet is scheduled at time $t + \Delta$ and padding packets are scheduled at the same time on all other outgoing links. In this way, no packet will be delayed for more than Δ and the scheme is optimal (as proven in [23]) in that it achieves mixing with the minimal amount of padding.

Besides packet timing, it is also important to consider other properties of padding schemes, such as the source and destination of padding, and which entities can distinguish dummy packets from real ones. Several variants have been proposed in the literature to address different trust and adversary models. For example, ISDN Mixes [13] use dummy traffic only in the link between the initiator and the local exchange, which discards the dummy packets, and assumes that at least one router in the path is honest. Partial-route padding and defensive dropping [10, 19] propose that dummy traffic be generated by the initiator and dropped by intermediate routers – and consider that some routers in the path may be malicious.

In both adaptive padding [18] and DLP schemes [22, 23], dummy packets are generated by intermediate routers, instead of the initiator. Subsequent routers cannot distinguish these dummy packets from (encrypted) real packets, and thus they are routed all the way to the end recipient, who discards them. The padding schemes in [18, 22, 23] consider trusted recipients and resist adversaries who compromise a subset of the routers.

Anonymity Metrics. By observing – or actively attacking – an anonymous communication system, the adversary typically obtains a probability distribution linking the initiator of a communication to all possible recipients, and vice versa. Then, one can use *Shannon entropy* [17] (or simply "entropy") as a measure of the adversary's uncertainty on who is the initiator (or recipient) of a communication [5, 16].

The analysis presented by Wang *et al.* [23] studied a single-node network, which offers high anonymity but no resistance to router compromise, low resilience to failures, and poor scalability. The anonymity provided by a single node is straightforward to compute with the metrics in [5, 16]: in a single-hop network routing C circuits, the probability of an initiator corresponding with each recipient is uniformly distributed, and anonymity is maximum (the entropy of the distribution is $\log_2(C)$.) Venkitasubramaniam and Tong [22] did examine multi-hop networks, but considered only the "information leaked by the timing of packets within a flow." "Anonymity" as defined in [22] is assumed to be maximum when the timing of packets does not leak any information – as is the case when dependent link padding is implemented.

Computing the anonymity of communications in complex networks while taking into account all information available to the adversary is infeasible to do analytically [15], as it requires enumerating all possible combinations of internal states in the routers, as well as initiator-recipient relationships. Previous comparisons of network topologies [7] avoided this problem by simplifying the

analyzed scenario – e.g., assuming that the load on each internal link within the network is exactly equal to the statistically expected load given a particular network topology. The Markov Chain Monte Carlo methodology recently proposed in [20] is based on sampling possible internal states and initiator-recipient correspondences that satisfy all constraints. This allows the efficient estimation (for a given confidence interval) of the adversary's probability distribution, taking into account all the available information. The model in [20] considers high-latency threshold mix networks. We note that the adversary's observation of a low-latency anonymity network that implements DLP, and has synchronous starts and ends of connections, is equivalent to that of a high-latency network made of threshold mixes. Therefore, the methodology can be used without major modifications to extend the analysis in [22] to consider routing constraints.

3 Model

System Model. We consider an anonymity system based on onion routing that implements dependent link padding and has synchronous starts and ends of connections. For simplicity, we assume that the path length of circuits is always three and require that all three routers in the path are distinct, but we note that our analysis is generalizable to other routing constraints [20].

When a client wishes to make a request, the system works as follows. First, it constructs a route selecting three routers (*nodes*) from the list of all available nodes, subject to topology constraints. It then connects to the first node (*entry*,) and exchanges keys to form an encrypted tunnel. Over this encrypted tunnel, the client connects to the second node (*middle*,) and then the third node (*exit*,) exchanging keys at each point such that each node knows the previous and next hops, but no more.

The connection through the three nodes, along with the corresponding keys, is known as a *circuit*. Once the circuit is established, the client requests that the last node creates a *stream* to carry the application data. Data is packaged into fixed length *cells* which are subsequently encrypted under the keys shared with the recipient, exit, middle, and entry nodes, and sent to the entry node. At each hop, one layer is removed until the recipient finally decrypts the payload. Note that DLP requires that the recipient be able to decrypt data cells, and to discard the dummy cells that have been added to the stream.

Multiple circuits may be carried on the *link* between any given pair of nodes. In addition to the circuit-level cryptography, which is end-to-end, there is also hop-by-hop link-level cryptography protecting the traffic between nodes. An external adversary will therefore not be able to tell, based on content, whether two cells correspond to the same circuit or to different ones.

Attacker Model. We assume that the adversary is *global*: it observes traffic on all communication links and knows the number of circuits routed over each of them. Furthermore, the adversary is *active*, and may introduce, delay, or drop cells. We note that DLP [22, 23] protects against active attacks, as all streams coming out of a node are identical – e.g., if the adversary deploys traffic

watermarking attacks [9, 24], then all outgoing streams will carry the watermark. Throughout the analysis, we also assume that all nodes are trustworthy, and thus the adversary is *external* and has no knowledge of node keys or other internal state information.

Note that denial of service attacks, long-term disclosure attacks, attacks involving corrupted nodes, and attacks on other protocol layers (e.g., dropping cells to force end-to-end retransmissions,) are not considered in this paper and left as a subject of future work.

4 RO-DLP: Reducing Padding Overhead in DLP

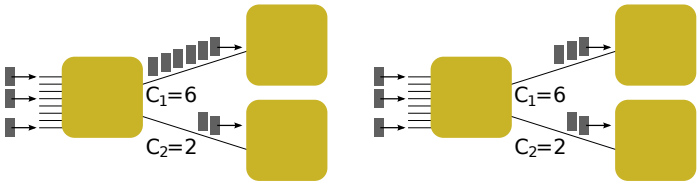

Fig. 1. Original DLP (left) and RO-DLP (right)

In the original DLP proposals [22, 23], nodes pad every outgoing circuit in the same way, independently of whether or not some circuits are being multiplexed over the same link. In anonymity networks however, nodes typically use link encryption, which hides the correspondence of cells to circuits within a link. In this section we present the *Reduced Overhead Dependent Link Padding* (RO-DLP) algorithm.[3] Compared to simple DLP, RO-DLP reduces the amount of dummy traffic sent over links that multiplex several circuits, while achieving the same level of security against global external adversaries that do not control nodes.

The goal of link padding is to prevent the adversary from learning the correspondence between incoming and outgoing circuits. Given that at time t the node forwards R_t cells, we show that it is enough to send R_t cells over links that contain a number c_i of circuits that is larger than R_t.

Let us consider a node n that routes C circuits over L links (note that $L \leq C$,) and let c_i denote the number of circuits multiplexed over the same link l_i ($1 \leq i \leq L$, and $\sum_{i=1}^{L} c_i = C$.) Initially, RO-DLP schedules a cell for each of the C outgoing circuits, as in DLP. Thus, at time t a set of C cells are scheduled, of which R_t correspond to cells that are being forwarded, and $C - R_t$

[3] We note that the term "Link Padding" has been used in the past [6] to mean padding that exists only on a single link and is not relayed to other nodes. In this paper, we use the terminology introduced by Wang et al. [23] where Dependent Link Padding refers to padding that, once generated, travels along the path until the end destination.

are dummy cells generated by node n. RO-DLP removes r_i dummy cells from link l_i as follows:

$$r_i = \begin{cases} 0 & \text{if} \quad c_i \leq R_t \\ c_i - R_t & \text{if} \quad c_i > R_t \end{cases}$$

The intuition behind this algorithm is the following. The adversary monitors the number of cells arriving at node n and can predict the number R_t of cells that will be forwarded at time t. When $c_i > R_t$ cells are sent over link l_i, the adversary knows that (at least) $c_i - R_t$ of these are dummy cells generated by n, and thus these do not provide any additional protection.

Consider a node that routes eight circuits over two outgoing links, such that $c_1 = 6$ and $c_2 = 2$, as shown in Figure 1. If only one cell is to be forwarded at time t (i.e., $R_t = 1$,) it is enough to send one cell on each of the outgoing links for the adversary to gain no information on the destination of the forwarded cell. One of the two cells sent will be the real cell, and the other will be a dummy cell going on one of the circuits of the other link. If, as in the example shown in the figure, three real cells are to be sent (i.e., $R_t = 3$,) then no padding can be removed from l_2, but we can still save three dummy cells in link l_1. From the perspective of the adversary, no additional information is leaked on the destination of the forwarded cells, compared to the case in which six cells are sent over l_1: in both cases, it could be that the three circuits for which there is a cell are routed over l_1, that one is routed over l_1 and two over l_2, or that two are routed over l_1 and one over l_2.

Note that if each link contains only one circuit, then no dummies can be removed and RO-DLP's overhead is the same as DLP's [22, 23]. If all circuits going through a node are routed over one single link (e.g., in a Cascade topology,) then no dummies would be sent by that node and RO-DLP would not generate any overhead. In Section 6.3 we present an evaluation of the reduction in overhead when RO-DLP is used with real traffic streams.

5 Experimental Setup

We have implemented a simulator to evaluate the anonymity and dummy traffic overhead in anonymity networks that implement dependent link padding. Our simulator generates networks of N nodes, where N is an input parameter. Users create circuits that traverse three nodes before reaching their destination. We call *entry node* the first node in the circuit path, *middle node* the second, and *exit node* the third and last node. We consider four possible topologies, shown in Figure 2:

- **Free Routes (FR):** Any combination of three distinct nodes is a valid circuit path. Given an entry node, we choose, uniformly at random, a middle node from the remaining $N - 1$ nodes, and an exit node from the remaining $N - 2$ nodes.
- **Stratified (S):** Nodes are divided into entries, middle nodes, and exits ($N/3$ nodes in each category,) such that any entry connects to any middle, and

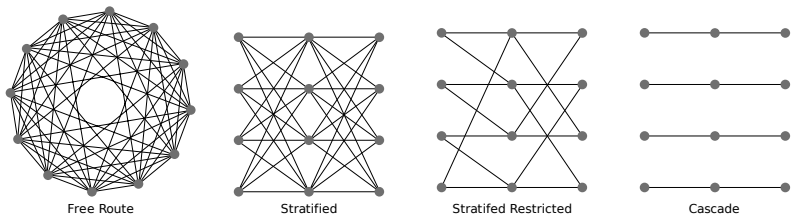

Fig. 2. Network topologies for $N = 12$

any middle to any exit. Given an entry node, we choose uniformly at random one of the $N/3$ middle nodes, and one of the $N/3$ exits.

- **Stratified Restricted (SR):** As in the previous case, nodes are divided into entry, middle and exit nodes. We have chosen values of N of the form $N = 3K^2$, where K is an integer. Each entry node is connected to $K = \sqrt{N/3}$ middle nodes, and each middle node to K exits. An entry node i $(0 \leq i \leq N/3 - 1)$ is connected to middle nodes $N/3 + [(i + j) \mod N/3]$, with $j = 0 \ldots K-1$; and a middle node i $(N/3 \leq i \leq 2N/3-1)$ is connected to exit nodes $2N/3 + [(i + j \cdot K) \mod N/3]$, with $j = 0 \ldots K - 1$. Given an entry node we construct the circuit paths choosing uniformly at random one of the $N/3$ exits, and then finding the middle node that connects the entry to the exit (in this topology each entry is connected to each exit by exactly one middle node.) The intuition behind this topology is to allow every entry node to connect to any exit node, while minimizing the number of links.
- **Cascades (C):** We consider $N/3$ parallel cascades of three nodes each. Given an entry node, the middle and exit nodes are fixed by the topology.

To make our evaluation as realistic as possible, we use as input real traffic data logged by a deployed Tor [6] node for a period of 24 hours. In particular, we have logged a timestamp and a circuit identifier[4] for each cell routed by the node. We consider *sessions* of 60 seconds – i.e., we divide the input into slices of 60-seconds duration, and assume that the traffic of sessions sufficiently separated in time is independent. We take into account both the forward and the backward traffic (i.e., requests and responses) in the bi-directional circuits that appear in that session.

We consider that the comparison of network topologies is fair when both individual nodes, as well as the network as a whole, carry the same amount of traffic, and we design our experiments in such a way that this condition is fulfilled. In the Stratified and Cascade topologies, we feed each of the $\frac{N}{3}$ entry nodes with the traffic of a session, with that node as first in the path, and the remainder of the path selected according to the network topology constraints. In Free Routes we follow a slightly different approach in order to keep the com-

[4] To anonymize the logs, the circuit ID and peer IP address were encrypted on collection, under a key which was discarded after logging was completed. This dataset will be made available by the authors upon request.

parison fair: we distribute the circuits of a session among three entries. In this way, both individual nodes and the overall network route the same amount of real traffic as in the other topologies. In Stratified and Cascade networks, nodes route (on average) C circuits either as entry, middle or exit node, and the total number of circuits in the network is $C_{\mathrm{T}} = C \cdot \frac{N}{3}$ ($\frac{N}{3}$ is the number of entry nodes.) In Free Routes, each node routes (on average) $\frac{C}{3}$ circuits as entry (plus $\frac{C}{3}$ as middle and $\frac{C}{3}$ as exit,) and the total number of circuits is $C_{\mathrm{T}} = \frac{C}{3} \cdot N$ (all N nodes are entry nodes.)

The nodes in our simulator implement the DLP and RO-DLP algorithms. We record the amount of traffic routed by the network per second, distinguishing between real and dummy traffic; and between intra-network traffic (sent between nodes) and traffic at the edge of the network (between nodes and end destinations.)

6 Results

We examine networks in terms of *anonymity loss* and dummy traffic (padding) *overhead factor*. The anonymity loss is the difference between the maximum achievable anonymity given the total number of circuits routed by the network and the actual anonymity that the network provides to its circuits. We note that when DLP is deployed, the timing of packets does not leak any information, and the anonymity provided by the system depends only on the routing constraints. Given a circuit c_x, we compute its anonymity loss as $H_{\mathrm{loss}} = H_{\mathrm{max}} - H(c_x)$. The maximum achievable anonymity is given by $H_{\mathrm{max}} = \log_2(C_{\mathrm{T}})$, where C_{T} is the total number of circuits routed by the network [5, 16]. For Stratified and Free Route networks, $H(c_x)$ is estimated by the method presented in [20], using the obtained lower bound as our estimation. In Cascades, we compute the anonymity $H(c_x)$ of a circuit c_x routed by cascade$_i$ as $H(c_x) = \log_2(C_i)$, where C_i is the number of circuits routed by cascade$_i$ (note that $\sum_i C_i = C_{\mathrm{T}}$.)

To present the results for the dummy traffic (padding) overhead, we use the *overhead factor*, which is computed as $\frac{Dum}{Real}$, where Dum is the number of dummy cells sent in the network every second, and $Real$ is the number of real data cells sent over the same time period. Thus, the overhead factor indicates the number of padding cells sent for each real data cell.

6.1 Feedback Effects in Free Route Networks

If dependent link padding is implemented in a Free Route network, feedback effects are likely to happen. The feedback effect occurs both with DLP and RO-DLP, and it provokes dummy traffic to be generated even in the absence of real traffic (this case leads to infinite padding overhead.) To illustrate this effect, consider two nodes routing two circuits in opposite directions, as shown in Figure 3. One real cell ▮ is sent into node A, on circuit Y. This cell is relayed ▮⟶ to node B. Node B will, after a delay of Δ, relay this cell onto the next hop of circuit Y, but also generate a padding cell ▯ on circuit X. When node A

receives this cell, it cannot tell that the cell is padding. Thus, A sends it onto the next hop of circuit X, and also generates a new dummy cell that is sent back to B, repeating the cycle. Note that feedback loops may form not only between pairs of nodes, but also in more complex structures involving several nodes.

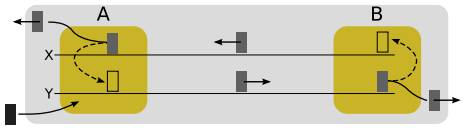

Fig. 3. Feedback loop with two nodes

Figure 4(a) compares the total traffic (number of cells per second) in a Stratified and a Free Route network that route the same input traffic (the networks have $N = 12$ nodes, route a total of $C_{\mathrm{T}} = 12$ circuits, and there are 10 cells per circuit within the first 3 seconds.) Although there is no more input traffic after $t = 4$, we can see that the Free Route network continues to generate dummy traffic that quickly becomes stationary. In the Stratified network, traffic stops once the last real cell has left the network (this happens at most $3 \cdot \Delta$ seconds after it has entered, and in our case Δ is 1 second and thus the last cell leaves before $t = 7$.)

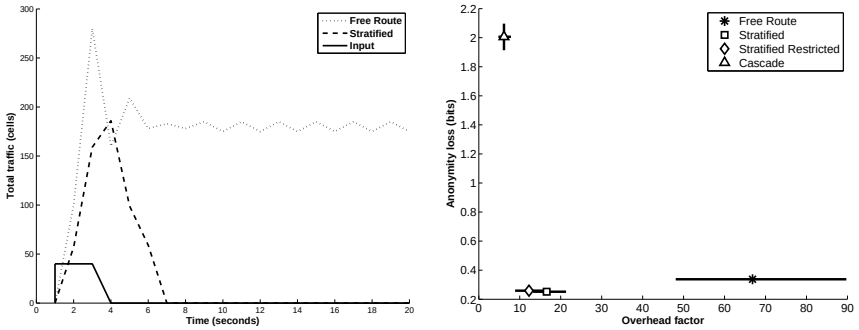

Fig. 4. Traffic in a Free Route and a Stratified network with the same input traffic (left.) Anonymity / Overhead tradeoffs for different topologies (right)

6.2 Comparison of Network Topologies

It is common for anonymity systems to offer a clear tradeoff between the level of anonymity and cost (with cost being in terms of delay and/or dummy traffic overhead,) such that more anonymity comes at a higher cost. In the scenarios considered in our analysis, the delay costs are fixed and identical for all the

network topologies, and thus we focus on the tradeoff between anonymity and dummy traffic overhead.

Figure 4(b) shows the tradeoff offered by the four considered topologies in a network with 12 nodes that implement RO-DLP. The x axis shows the overhead factor (i.e., number of padding cells sent in the network for each real cell, taking into account both the traffic between nodes and the traffic in the edges of the network.) The y axis shows the anonymity loss with respect to the maximum achievable level in each of the experiments (note that the maximum depends on the total number of circuits C_{T} routed by the network in each experiment.) Therefore, lower values in the y axis correspond to networks that come closer to providing maximum anonymity to the circuits they are routing. The symbol at the center of the plot for each topology represents the median values for anonymity and overhead, the lines indicate the first and third quartiles. Although it is not shown in the figure, the overhead of Free Routes tends to infinity when the real traffic is very low.

As we can see in the figure, the overhead is lowest in Cascades, and it increases as more routes are possible in the network (i.e., the next best is Stratified Restricted, then Stratified, and worst is Free Route.) This is rather intuitive, as restricting the routing implies that more circuits are routed (multiplexed) over fewer links, and thus less overhead is generated by the RO-DLP algorithm. The fact that Free Routes has a much higher overhead than the other topologies is due to the feedback effects explained in the previous section.

A more interesting effect appears when we look at anonymity. *A priori*, one could expect topologies with more overhead to provide better anonymity. However, this is not the case: the best anonymity is provided by Stratified topologies (closely followed by its Restricted variant,) instead of Free Routes. In Stratified networks, circuit routes going through the same node are always mixed, because the node is in the same path hop for both routes. In Free Routes however, circuits may pass by the same node and not be mixed if the node is at a different hop in the circuit paths. Consider for example a node n that is the entry node for circuit c_a and exit node for circuit c_b. Given that routes always have three hops, the adversary knows that the circuit c_a entering the network at n cannot go out of the network immediately, and thus the outgoing c_b cannot possibly be the exit of c_a – i.e., c_a and c_b are not mixed in n.

We note that all topologies except Cascades consistently provide very high anonymity levels: the anonymity loss is less than 0.4 bits, and its variance is very small. For Cascades, the median loss is 2 bits, which corresponds to partitioning the anonymity set in four. Indeed, a network consisting of four parallel cascades partitions the total anonymity set of circuits in four subsets, with each subset being routed by a separate cascade.

Overall, Stratified topologies provide the best anonymity / overhead tradeoffs, with restrictions in the routing reducing the overhead at the cost of slightly worse anonymity. Cascades are better than Stratified topologies in terms of overhead, but this comes at a high cost in anonymity (a problem that becomes worse

as the network grows, as shown in Section 6.4.) Free Routes are worse than Stratified topologies *both* in terms of anonymity as well as overhead.

6.3 Dummy Traffic Overhead with DLP and RO-DLP

In Section 4 we proposed a RO-DLP algorithm to reduce the overhead when several circuits are multiplexed over the same link. We note that multiplexing only happens in the links between network nodes, which typically carry many circuits. In our experiments, we assume that the links on the edges of the network – i.e., between nodes and external entities (initiators and responders) – carry only one circuit. Therefore, no multiplexing happens on the network edges and RO-DLP produces a similar overhead to DLP.

The boxplots[5] of Figure 5(a) show the intra-network overhead (i.e., only considering links between nodes) of RO-DLP compared to DLP. The results were obtained performing several dozens of simulation experiments on networks of 12 nodes, using real traffic as input, and having each node route the same amount of traffic as the Tor router from which the data was collected.

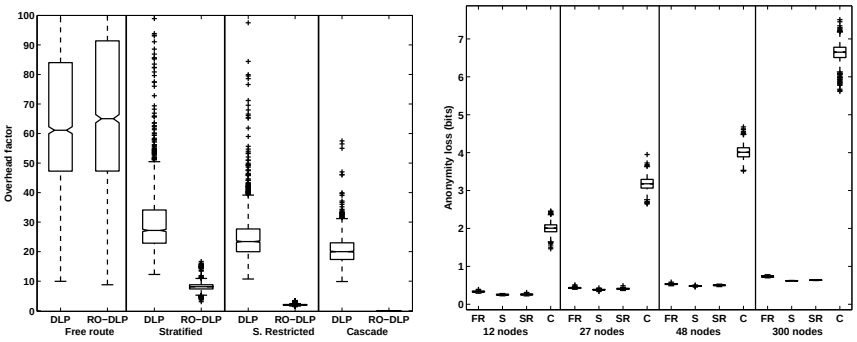

Fig. 5. Overhead (traffic on intra-network links only) of DLP and RO-DLP (left.) Network scalability vs. anonymity (right.)

In Cascades, all circuits going through a node are multiplexed over one link leading to the next node, which explains why RO-DLP reduces the intra-network overhead factor from 20 to zero (note that RO-DLP still generates padding cells on the edges, which is not shown, and thus the overall overhead is greater than zero.)

The overhead reduction of RO-DLP over DLP is rather significant in Stratified networks too. In Stratified Restricted topologies, the median overhead factor is reduced from 23 to 1.5 (i.e., from sending 23 padding cells for each real cell,

[5] The line in the middle of the box represents the median of the distribution of values over many experiments. The lower and upper limits of the box correspond, respectively, to the first and third quartiles of the distribution.

to just sending 1.5 padding cells per real cell;) and in Stratified from 27 to 8. As we can see, there is a very direct relationship between the number of possible routes (i.e., amount of circuit multiplexing) and the reduction in overhead: the fewer the possible routes, the lower the overhead.

RO-DLP does not have a beneficial effect in Free Route topologies though. This is due to two effects. First, because Free Routes allow many more possible circuit paths, circuits are more spread over links and thus links multiplex fewer circuits. This mitigates to a large extent the benefits of RO-DLP. Furthermore, RO-DLP fails to counter the feedback effects explained in Section 6.1, because it only affects the removal of padding cells at the node where they are generated. Once these cells have been sent to other nodes, they are treated as real traffic and bounced back and forth in the network (just as in simple DLP.)

6.4 Network Scalability: Anonymity and Overhead

We have performed most of our experiments on networks of only 12 nodes, given that the simulation time of experiments on bigger networks increases rapidly with the network size. In this section we show results on how anonymity and overhead varies with the size of the network that implements RO-DLP.

Figure 5(b) shows the anonymity loss for the four topologies and network sizes of 12, 27, 48 and 300 nodes. The y axis represents the anonymity loss in these networks with respect to the maximum achievable $H_{\max} = \log_2(C_{\mathrm{T}})$, where C_{T} is the total number of circuits routed by the network. Note that larger networks route more circuits and thus have a bigger $H_{\max}$. We can see in Figure 5(b) that Stratified, Stratified Restricted, and Free Route topologies scale very well in terms of anonymity – their anonymity remains very close to the maximum when the network size grows. In networks of 300 nodes, the anonymity loss for any of those topologies is less than one bit.[6]

Cascade topologies however, have poor scalability in terms of anonymity. In this topology a larger network implies more parallel cascades. Given that cascades are independent of each other, they provide a constant level of anonymity, and thus a bigger anonymity loss as $H_{\max}$ increases. Consider a network consisting of N nodes and $\frac{N}{3}$ cascades, and assume for simplicity that all cascades route the same number C of circuits; i.e., the total number of circuits routed by the network is $C_{\mathrm{T}} = \frac{N}{3}C$. The anonymity provided by the cascades is $H_{\mathrm{cascade}} = \log_2(C)$, and the maximum achievable anonymity is $H_{\max} = \log_2(C_{\mathrm{T}}) = H_{\mathrm{cascade}} + \log_2(\frac{N}{3})$. The anonymity loss is thus $\log_2(\frac{N}{3})$ on average; i.e., $\log_2(4) = 2$ bits for $N = 12$ nodes, $\log_2(9) = 3.17$ bits for $N = 27$ nodes, etc.

We show in Figure 6(a) and Figure 6(b) the overhead in intra-network links (that multiplex several circuits,) and at the edge of networks of 12, 27 and 48 nodes. As expected, overhead is unaffected by the growth of the network in Cascade topologies. The overhead factor remains at zero in the links between

[6] An anonymity loss of one bit is equivalent to the adversary partitioning the anonymity set of C_{T} circuits in two subsets.

cascade nodes, as all circuits are multiplexed over a single link; and it remains constant at the network edges, as circuits routed by parallel cascades do not mix with each other: bursts in the traffic of one circuit only produce padding in the circuits going through the same cascade.

In the other three topologies, we can observe that network size has a negative impact on the overhead factor of the network. This is because a traffic burst in a single circuit produces a burst of dummy traffic in all other circuits, and as more circuits are routed by the network, bursts occur at a higher frequency. The overhead factor is particularly large for the traffic on the edges of the network (Figure 6(a)) because links to clients and destinations contain a single circuit, and thus do not benefit from the optimization based on circuit multiplexing. In the case of intra-network traffic (Figure 6(b),) we can see that Stratified restricted topologies manage to keep the overhead factor just over 8 when the network grows to 48 nodes – while overhead reaches 30 in Stratified networks, and over 80 in Free Routes.

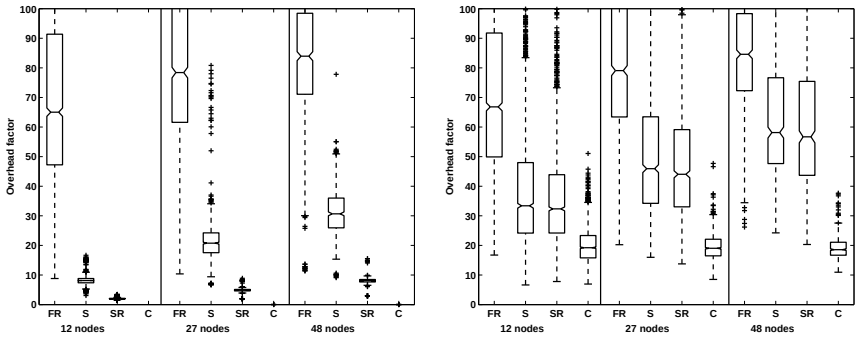

Fig. 6. Network scalability: overhead in intra-network links (left.) Overhead at the network edge right.)

7 Applying DLP to Tor

Although Tor is the most widely deployed anonymity network, it offers fairly weak protection against a global adversary, because traffic is not mixed. Tor aims to minimize latency and network load, so nodes neither add padding nor delay cells, hence it is trivial to perform timing analysis, either on an end-to-end or hop-by-hop basis. The Tor designers made these choices because latency would make interactive use intolerable, and existing ILP schemes had unacceptable overhead. However DLP, with the optimizations we have proposed in this paper, is a more promising approach.

In DLP, edge nodes need not generate padding, but they do need to consume it. Moreover, an adversary should not be able to distinguish padding from

normal traffic. If we consider only internal circuits – where both the initiator and destination run the Tor software, though they do not necessarily need to route anyone else's traffic – DLP is straightforward to implement. Tor already uses internal circuits for connecting to hidden services (where the destination server wishes to hide its identity), and when the destination server is known to be running a Tor node.

It may also be possible to implement DLP without the destination being aware of Tor, provided there is end-to-end encryption and the exit node can inject padding which will be ignored by the destination. This option is more complex, because it requires that the exit node be aware of the encryption protocol. In particular, if the end-to-end encryption scheme implemented at the destination silently drops malformed packets, then it can consume padding without any changes being necessary.

Network Topology. The original Tor design was a free-route network, however for efficiency reasons it has now moved to a more complex topology. Currently only a subset of nodes can act as the entry (because they must be fast and be highly reliable), and only a subset can act as the exit (because nodes must opt in to allowing exit traffic). To balance load over the network, nodes which can neither be entry nor exit are preferentially selected as middle nodes. Strictly speaking, entry and exit nodes can be selected for the middle position, provided that there is sufficient entry and exit bandwidth, respectively. However, most of the time this is not the case and in practice Tor has a network topology very close to Stratified. For this reason, it would not be a significant change to move to a fully Stratified topology.

Implementation of Padding Modes in Tor. In order to implement DLP, it is necessary that connections between nodes are encrypted, so that an external adversary cannot tell whether two cells belong to the same circuit or different ones. Tor uses TLS for protecting both confidentiality and integrity on links, so complies with this requirement.

With respect to the creation of padding, the Tor protocol does permit dummy cells to be inserted, although the current implementation does not generate padding nor are there any plans to do so. Two types of padding cells are offered: link padding and circuit padding. However neither meet our requirements; the former is detected as padding by nodes and dropped, and the latter can only be injected by the initiator. There is no way for a node to inject a padding cell such that subsequent nodes on the path cannot distinguish it from data cells sent by the initiator.

The fundamental problem for implementing DLP in Tor is that the variant of onion routing adopted uses a stream cipher. Each Tor relayed cell contains a circuit ID that identifies which circuit the message pertains to, and an encrypted payload. On receiving such a cell, the Tor node checks if a key has been negotiated for the given circuit ID. If so, the node uses AES CTR mode to decrypt the cell with the counter being the number of ciphertext blocks seen in that circuit. Then, the node verifies whether a 32 bit digest in the cell matches the SHA-1 hash of all valid cells in the circuit.

Therefore, if a cell is injected by an intermediate node, the counter will be desynchronized, the data corrupted, and the digest check will fail. To resolve this problem, a simple addition to the Tor protocol would be another type of link padding cell which triggers a new padding cell to be emitted for the same circuit on the output link. In this way, each hop could add their own padding, which would be maintained all the way to the last hop. As links are protected using TLS, an adversary cannot distinguish padding cells from real cells, and so the goal of the padding would be maintained. In the implementation, care would need to be taken that the processing time for a padding cell would be identical to that for a real cell, to resist side-channel attacks leaking information on cell type.

This approach would resist the external adversary considered in the anonymity analysis of RO-DLP. However, in a more realistic scenario the adversary may control some routers, and any corrupt node on the path would be able to trivially tell which cells are padding. Circuit padding cannot be used to resolve this weakness because intermediate nodes cannot inject new cells. However, if instead of CTR mode, a per-cell IV was used, this problem would not exist – i.e., padding would not affect the decryption of other cells. Intermediate nodes do not know the key shared by the sender and other nodes in the path, hence the padding will fail the integrity check at the final hop and be discarded. A node would therefore be able to add padding cells with a random IV, and intermediate nodes will be unable to distinguish them from real data cells. A downside of this approach is that there is the overhead of a IV per-cell. Nevertheless, it has the advantage that there is no longer any need for a reliable transport protocol between nodes, provided there is an end-to-end error recovery mechanism. Moving from TCP to UDP for node-to-node communication has been shown to offer significant performance benefits, especially under congestion [14].

8 Conclusions and Future Work

Dependent link padding prevents timing analysis in low-latency anonymity networks while minimizing the overhead. However, the impact of complex topologies [2, 7] on the performance of this technique had not yet been assessed. In this work we have analyzed anonymity / overhead trade-offs in low-latency anonymity systems that implement dependent link padding, and compared three topologies: Cascades, Free Routes and Stratified networks.

We have found that feedback effects appear in Free Route networks, leading to disproportionate padding overhead – a phenomenon not previously discussed in the literature. In contrast, Stratified networks and Cascades do not suffer from this problem, making them substantially more efficient. However, the level of anonymity provided by Cascades decreases severely when the network grows – while the other topologies maintain high anonymity, with Stratified networks being the best. We conclude that Stratified topologies offer the best trade-off between anonymity and overhead.

We have introduced a Restricted topology based on Stratified networks, which further reduces the overhead with almost no loss of anonymity. In addition, we have proposed RO-DLP, which takes advantage of circuit multiplexing in anonymity networks to reduce the amount of padding. Our experiments show that in Stratified Restricted topologies, RO-DLP reduces the overhead factor from 23 to just 1.5.

While dependent link padding is ideal for onion routing, as it offers good security without causing high latency, we have argued that the current Tor protocol cannot accommodate it. We have outlined modifications to the Tor protocol, such as moving from a per-stream IV to a per-cell IV, and discussed other applicability issues.

In this work we have assumed that all the nodes in the network are trustworthy. This is essential for RO-DLP to achieve the same level of protection against an external adversary, when compared to previous dependent link padding proposals. If the adversary has control over some of the nodes in the network [7], she would see partially padded circuits, and potentially correlate traffic based on timing analysis. Strategies for assigning padding to circuits in ways that minimize the effectiveness of this attack are left for future work.

Acknowledgements. The authors would like to thank George Danezis for early valuable discussions. C. Diaz and C. Troncoso are funded by the Fund for Scientific Research in Flanders (FWO). S. Murdoch is funded by The Tor Project. This work was supported in part by the IAP Programme P6/26 BCRYPT of the Belgian State.

References

1. Oliver Berthold, Hannes Federrath, and Stefan Köpsell. Web MIXes: A system for anonymous and unobservable Internet access. In *Design Issues in Anonymity and Unobservability (PET 2000)*, pages 115–129. Springer, LNCS 2009, 2000.
2. Rainer Bohme, George Danezis, Claudia Diaz, Stefan Kopsell, and Andreas Pfitzmann. Mix cascades vs. peer-to-peer: Is one concept superior? In *Proceedings of the Privacy Enhancing Technologies Workshop (PET 2004)*, pages 243–255. Springer LNCS 3424, 2004.
3. George Danezis. Mix-networks with restricted routes. In *Proceedings of the Privacy Enhancing Technologies Workshop (PET 2003)*, pages 1–17. Springer, LNCS 2760, 2003.
4. George Danezis, Roger Dingledine, and Nick Mathewson. Mixminion: Design of a Type III Anonymous Remailer Protocol. In *Proceedings of the IEEE Symposium on Security and Privacy*, pages 2–15. IEEE Computer Society, 2003.
5. Claudia Diaz, Stefaan Seys, Joris Claessens, and Bart Preneel. Towards measuring anonymity. In *Proceedings of the Privacy Enhancing Technologies Workshop (PET 2002)*, pages 54–68. Springer, LNCS 2482, 2002.
6. Roger Dingledine, Nick Mathewson, and Paul Syverson. Tor: The second-generation onion router. In *Proceedings of the USENIX Security Symposium*, pages 303–320, 2004.
7. Roger Dingledine, Vitaly Shmatikov, and Paul Syverson. Synchronous batching: From cascades to free routes. In *Proceedings of the Privacy Enhancing Technologies Workshop (PET 2004)*, pages 186–206. Springer LNCS 3424, 2004.

8. David Goldschlag, Michael Reed, and Paul Syverson. Hiding routing information. In *Proceedings of Information Hiding (IH 1996)*, pages 137–150. Springer, LNCS 1174, 1996.
9. Amir Houmansadr, Negar Kiyavash, and Nikita Borisov. RAINBOW: A robust and invisible non-blind watermark for network flows. In *Proceedings of the Network and Distributed System Security Symposium (NDSS 2009)*. The Internet Society, 2009.
10. Brian Neil Levine, Mike Reiter, Chenxi Wang, and Matthew Wright. Timing attacks in low-latency mix systems. In *Procceedings of Financial Cryptography (FC 2004)*, pages 251–265. Springer LNCS 3110, 2004.
11. Ulf Möller, Lance Cottrell, Peter Palfrader, and Len Sassaman. Mixmaster Protocol – Version 2. IETF Internet Draft, 2003.
12. Steven J. Murdoch and Piotr Zieliński. Sampled traffic analysis by Internet-exchange-level adversaries. In *Proceedings of the Privacy Enhancing Technologies Syposium (PETS 2007)*. Springer LNCS 4776, 2007.
13. Andreas Pfitzmann, Birgit Pfitzmann, and Michael Waidner. ISDN-MIXes: Untraceable communication with small bandwidth overhead. In *Kommunikation in Verteilten Systemen, Grundlagen, Anwendungen, Betrieb, GI/ITG-Fachtagung*, pages 451–463. Springer, 1991.
14. Joel Reardon. Improving Tor using a TCP-over-DTLS tunnel. Master's thesis, University of Waterloo, 2008.
15. Andrei Serjantov. *On the Anonymity of Anonymity Systems*. PhD thesis, University of Cambridge, 2004.
16. Andrei Serjantov and George Danezis. Towards an information theoretic metric for anonymity. In *Proceedings of the Privacy Enhancing Technologies Workshop (PET 2002)*, pages 41–53. Springer, LNCS 2482, 2002.
17. Claude Shannon. A mathematical theory of communication. *The Bell System Technical Journal*, 27:379–423, 1948.
18. Vitaly Shmatikov and Ming-Hsiu Wang. Timing analysis in low-latency mix networks: Attacks and defenses. In *Procceedings of the European Symposium on Research in Computer Security (ESORICS 2006)*, pages 18–33. Springer LNCS 4189, 2006.
19. Paul Syverson, Gene Tsudik, Michael Reed, and Carl Landwehr. Towards an analysis of onion routing security. In *Design Issues in Anonymity and Unobservability (PET 2000)*, pages 96–114. Springer LNCS 2009, 2000.
20. Carmela Troncoso and George Danezis. The Bayesian analysis of mix networks. In *Proceedings of the ACM Conference on Computer and Communications Security (CCS 2009)*, page 11. ACM, 2009.
21. Parvathinthan Venkitasubramaniam, Ting He, and Lang Tong. Relay secrecy in wireless networks with eavesdroppers. In *Proceedings of the Allerton Conference on Communication, Control and Computing*, 2006.
22. Parvathinthan Venkitasubramaniam and Lang Tong. Anonymous networking with minimum latency in multihop networks. In *Proceedings of the IEEE Symposium on Security and Privacy*, pages 18–32. IEEE Computer Society, 2008.
23. Wei Wang, Mehul Motani, and Vikram Srinivasan. Dependent link padding algorithms for low latency anonymity systems. In *Proceedings of the ACM Computer and Communications Security Conferece (CCS 2008)*, pages 323–332. ACM, 2008.
24. Xinyuan Wang, Shiping Chen, and Sushil Jajodia. Network flow watermarking attack on low-latency anonymous communication systems. In *Proceedings of the IEEE Symposium on Security and Privacy*, pages 116–130. IEEE Computer Society, 2007.