Mitigating Traffic Analysis Attack in Smartphones with Edge Network Assistance

Yaodan Hu, Xuanheng Li, Jianqing Liu, Haichuan Ding, Yanmin Gong, Yuguang Fang

***Abstract*—With the growth of smartphone sales and app usage, fingerprinting and identification of smartphone apps have become a considerable threat to user security and privacy. Traffic analysis is one of the most common methods for identifying apps. Traditional countermeasures towards traffic analysis includes traffic morphing and multipath routing. The basic idea of multipath routing is to increase the difficulty for adversary to eavesdrop all traffic by splitting traffic into several subflows and transmitting them through different routes. Previous works in multipath routing mainly focus on Wireless Sensor Networks (WSNs) or Mobile Ad Hoc Networks (MANETs). In this paper, we propose a multipath routing scheme for smartphones with edge network assistance to mitigate traffic analysis attack. We consider an adversary with limited capability, that is, he can only intercept the traffic of one node following certain attack probability, and try to minimize the traffic an adversary can intercept. We formulate our design as a flow routing optimization problem. Then a heuristic algorithm is proposed to solve the problem. Finally, we present the simulation results for our scheme and justify that our scheme can effectively protect smartphones from traffic analysis attack.**

***Index Terms*—Multipath routing, network security, traffic analysis**

I. Introduction

Nowadays, with the development of technology in both hardware and software, smartphones become more and more powerful while the prices get more and more affordable. This leads to a striking growth of smartphone usage. As reported by Gartner in 2015 [1], the sales of smartphones to end users in 2014 reached 1.2 billion in total, which is increased by 28.4% compared to that in 2013. One of the reasons for this increasing sale is the various applications, i.e., apps, smartphones bring to us, which dramatically enrich the features and functionalities of smartphones. There is nearly 3 million apps in Google Play [2] and the number of app downloads is expected to reach 197 billion by the end of 2017 [3]. We can watch videos whenever we want; we can check transportation information before we go outside; or we can share our life with our friends through social networks. Though enjoying the convenience smartphones bring to us, we are also undertaking risks by using those fascinating apps. Since users always download apps according to their interests, a user's private information would be compromised if an attacker could identify what apps the user is using. Imagine if an attacker finds a man using homosexual dating app, this person might be blackmailed by the attacker. Besides, if an attacker has already exploited some vulnerable apps, he can narrow down the victims to users who are using those apps instead of randomly selected users.

Traffic analysis and fingerprinting are examples of the most common methods to infer which app is being used the app a user is using. Earlier works identify apps by inspecting HTTP payloads [4], [5]. In [4], Dai et al. propose NetworkProfiler to automatically generate network profiles and identify Android apps. They generate network profiles by User-Interface (UI) fuzzing, which monitors the traffic traces while trying different execution paths in an app. Then they identify apps by inspecting the HTTP payloads and extracting app fingerprints. Since their work needs to inspect the payloads, their work fails to work if HTTPS or TLS is adopted. Nevertheless, even with encryption, app traffic can still be identified by performing traffic analysis with machine learning [6]–[8]. Taylor et al. [6] propose AppScanner to identify apps via both unencrypted and encrypted traffic. They distinguish different apps' traffic by dividing the traffic into bursts and flows to minimize the interference between different apps. Then features are extracted from those flows and trained with supervised learning algorithms. They evaluate their work with 110 most popular apps in Google Play and prove their work to be accurate and efficient. Our work is motivated by their work.

Current work against traffic analysis includes traffic morphing and multipath routing, etc. In this paper, we take multipath routing approach. Multipath routing transmits traffic by splitting traffic into several subflows and sending them through different routes. This increases attackers' difficulty to identify the app because he needs to compromise more paths to collect more parts, which costs more resources. Multipath routing has been thoroughly discussed in Wireless Sensor Networks (WSNs) and Mobile Ad Hoc Network (MANETs) [9]–[12]. Lou et al. [9] propose a scheme called SPREAD for multipath routing in MANETs. They apply secret sharing into their scheme to enable the receiver to recover the message while enhancing the security. They discuss the appropriate choice of the number of shares to guarantee sufficient security level. They modify Dijkstrra algorithm by using "link cache" [13] to decompose routes into disjoint links and use this modified algorithm to find the most secure paths. Their scheme does

This work was partially supported by National Science Foundation under CNS-1409797, IIS-1722791 and CNS-1423165.

Y. Hu, J. Liu, H. Ding, and Y. Fang are with the School of Electrical and Computer Engineering, University of Florida, Gainesville, FL 32611 USA (email: cindy.hu@ufl.edu; jianqingliu@ufl.edu; dhcbit@gmail.com; fang@ece.ufl.edu).

X. Li is with the School of Information and Communication Engineering, Dalian University of Technology, Dalian 116024, China (e-mail: lixuanheng@mail.dlut.edu.cn).

Y. Gong is with the School of Electrical and Computer Engineering, Oklahoma State University, Stillwater, OK 74078 USA (email: yanmin.gong@okstate.edu)

978-1-5386-3180-5/18/$31.00 ©2018 IEEE

achieve recovery capability but does not take link capacity into consideration. Different from WSNs and MANETs, multipath routing in smartphone has its own features. With the development of edge computing, more edge nodes such as Road Side Units (RSU) are available for us to transmit data. Besides the quantity of these nodes, they are also more powerful and the networks are more stable, which provides us with more reliable communication connections to deliver users' traffic compared with the opportunistic connections in MANETs. Though the transmission environment is better, the edge network is much more heterogeneous than WSNs. Smartphones are equipped with various interfaces such as cellular networks, WiFi Access and Bluetooth, which enable us to connect to multiple nodes simultaneously. Nevertheless, different interfaces can provide different data rates with different communication range. Furthermore, different intermediate edge nodes also provide different services. Therefore, when considering multipath routing for smartphones with edge network assistance, we should also take the heterogeneity into consideration. For example, we should split the traffic and map out routing paths according to different channel capacities.

In this paper, we focus on how to split and transmit the traffic while dealing with compromised nodes and eavesdropping threats. We address this security problem by formulating it as a flow routing optimization problem with several network constraints. Our goal is to minimize the traffic an adversary can get while satisfying certain network constraints. We solve this optimization problem with a heuristic algorithm. Then we simulate our scheme and justify its effectiveness against traffic analysis.

The rest of the paper is organized as follows. We present our system model and formulate our problem in Section II. Then in Section III, we solve the flow routing problem with a heuristic algorithm. In Section IV, we conduct simulation study to evaluate our scheme and discuss the results. In Section V, we conclude this paper.

II. System Model and Problem Formulation

A. System Model

We consider the network illustrated in Fig. 1. Let s denote a mobile user, who is using an app and send/receive data to/from the network service provider d through a set of relay nodes $\mathcal{N} - \{s, d\} = \{2, \ldots, n, \ldots, N-1\}$. The relay nodes are nodes which can assist mobile users to transmit data traffic. A relay node can be an RSU, or another mobile user who is willing to provide help. The mobile user can connect to these relay nodes simultaneously through various interfaces, such as 5G, WiFi or Bluetooth. If a node i can directly transmit data to another node j with signal received by j above a given threshold P_{Tx}, we denote such tuple of nodes (i, j) as a link. The set of directed links is denoted as E, then the network can be represented as a graph $G(\mathcal{N}, E)$. The control center, denoted by c, is a powerful node who has the knowledge of the whole network, such as the graph G and the spectrum availability. The control center serves as an agent for the service provider and can be implemented in a cloud. It is responsible for performing flow routing optimization for mobile users and the service provider. When a mobile user wants to send traffic to a service provider, he needs to inform the data center of required flow rate at each time instant, then the control center would perform flow routing to find the optimal routes and flow rate for each route and inform him the results. Note that the traffic splitting scheme should be applied not only when the user sends traffic to the service provider but also when he receives traffic from the service provider, and it would be the same process for the service provider in the latter situation. The traffic is split into L data flows. The l-th data flow is called the *commodity* l. Each commodity corresponds to a path from s to d. Let $\mathcal{L} = \{1, 2, \ldots, l, \ldots, L\}$ denote the set of commodities and $r_l \geq 0$ denote the flow rate on route l. Let $f_{ij,l}$ represent the flow rate flowing from i to j for the lth commodity. Each link can either carry the full traffic for a given commodity or none of it. Table I gives a summary of notations.

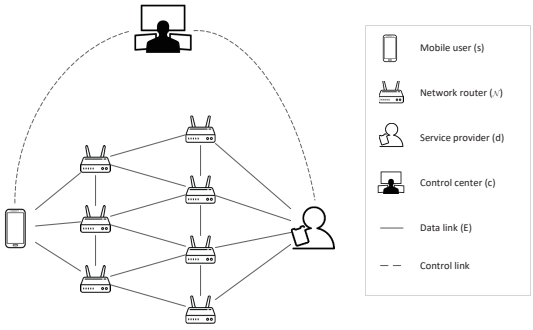


Fig. 1. System model

TABLE I
A LIST OF NOTATIONS

s	Mobile user
d	Service provider
c	Control center
$\mathcal{N}$	The set of nodes
N	The number of nodes
E	The set of direct links
G	Network graph
$\mathcal{L}$	The set of commodities
L	The number of commodities
r_l	Flow rate of commodity l
$f_{ij,l}$	Flow rate for commodity l on link (i, j)
$y_{ij,l}$	Indicator of link (i, j) for commodity l
p_i	The probability that node i is compromised
α_l	The probability that route for commodity l is compromised
F	The flow rate of the originally traffic before splitting
c_{ij}	The capacity of link (i, j)
T_{ij}	The set of neighbours of node i

B. Related Model in Multi-hop Networks

1) Transmission Range and Interference Range: Assume all nodes use the same static power P for transmissions. Ac-

cording to the model proposed in [14], the power propagation gain can be denoted as

$$g_{ij} = \gamma \cdot d_{ij}^{-\beta} \tag{1}$$

in which γ is an antenna related constant, β is the path loss exponent and d_{ij} is the distance between node i and j. Therefore, an link (i, j) exists if and only if $d_{ij} \leq R_{Tx}$, where $R_{Tx} = \gamma P / {P_{Tx}}^{1/\beta}$. We denote the set of neighbours of node i as $T_i = \{j : d_{ij} \leq R_{Tx}\}$.

2) Link Capacity: According to the Shannon-Hartley theorem, if a node i transmits data to node j on link (i, j) with bandwidth W, the achievable data rate, i.e., the capacity of link (i, j) with band m can be calculated as follows:

$$c_{ij} = W \log_2(1 + \frac{g_{ij} P}{\eta}) \tag{2}$$

in which η is the ambient Gaussian noise power at node j.

C. Adversarial Model

The payloads of traffic are encrypted using secure protocols such as HTTPS/TLS. An adversary is a malicious party who intends to identify what app the mobile user is using by analysing traffic shape based on the traffic he receives. Assume the probability that node i is compromised is p_i and the probability of a node being compromised is assumed to be independent. Note that this information is also available at the control center. If a relay node is compromised, all traffic flowing through this node would be captured by the adversary. Besides, we assume that the adversary will only conduct passive attack such as traffic analysis because he wants to hide himself. The route for commodity l is compromised as long as at least one relay node on that route is compromised. The adversary has the capability to distinguish different packets and the same packet will not be considered multiple times.

We assume the source and destination nodes are mutually authenticated so both ends are secure and trusted in our scenario.

D. Problem Formulation

In this part, we will introduce a few system level constraints in network and illustrate our objective function. In this paper, since we are focusing on a security problem, we are not going to discuss the communication issues such as power control or channel allocation.

We use a boolean variable $y_{ij,l} \in \{0, 1\}$ to indicate whether the commodity l will employ link (i, j) or not; if $y_{ij,l} = 1$, the link carries the traffic, and vice versa. Therefore, the flow rate of link (i, j) for commodity l can be represented as:

$$f_{ij,l} = r_l \cdot y_{ij,l} \tag{3}$$

For the mobile user s, as a source node, it should only send out traffic but not receive traffic, therefore

$$\sum_{j \in T_s} f_{js} = \sum_{j \in T_s} \sum_{l \in \mathcal{L}} f_{js,l} = 0 \tag{4}$$

$$\sum_{j \in T_s} f_{sj} = \sum_{j \in T_s} \sum_{l \in \mathcal{L}} f_{sj,l} = F \tag{5}$$

F is the flow rate of the original traffic before splitting. Obviously,

$$\sum_{l \in \mathcal{L}} r_l = F \tag{6}$$

For a relay node i, on the one hand, a relay node will transmit all traffic it receives. On the other hand, since the adversary only conducts passive attack, he would not drop packets maliciously. Therefore, we have

$$\sum_{j \in T_i}^{j \neq s} f_{ij,l} = \sum_{k \in T_i}^{k \neq d} f_{ki,l} \quad \forall i \in \mathcal{N} - s, d \quad \forall l \in \mathcal{L} \tag{7}$$

For the service provider, as a destination node, it should receive all traffic if no packet drops and would not send out traffic

$$\sum_{j \in T_d} f_{jd} = \sum_{j \in T_d} \sum_{l \in \mathcal{L}} f_{jd,l} = F \tag{8}$$

$$\sum_{j \in T_d} f_{dj} = \sum_{j \in T_d} \sum_{l \in \mathcal{L}} f_{dj,l} = 0 \tag{9}$$

Besides above constraints, it is also required that the total traffic flowing through link (i, j) should not exceed its capacity

$$\sum_{l \in \mathcal{L}} f_{ij,l} \leq c_{ij} \tag{10}$$

c_{ij} denotes the capacity of link (i, j).

Meanwhile, each node should not transmit same commodities multiple times, which can be represented as follows

$$\sum_{j \in T_i} y_{ij,l} \leq 1 \quad \forall i \in \mathcal{N} \quad \forall l \in \mathcal{L} \tag{11}$$

Note that if (4),(5) and (7)sufficient conditions for (8) and (9). Therefore it would be sufficient to list (4),(5) and (7) only.

In this paper, we attempt to minimize the traffic an adversary captures to extract identifying features. We consider two situations: single-node attack and multi-node attack. In single-node attacks, the adversary can only compromise a relay node at a time. This is practical because compromising multiple relay nodes costs more and an adversary might not have sufficient resources. In multi-node attacks, the adversary can collude with others and share traffic information with each other; in the worst case, every node except the source node and the destination node might be compromised simultaneously.

Under a single-node attack, for each relay node, the expectation of flows an adversary can get is given as

$$p_i \sum_{l \in \mathcal{L}} \sum_{j \in T_i} y_{ij,l} r_l \tag{12}$$

To minimize the traffic an adversary is expected to receive, we minimize the maximum of flows an adversary is expected to receive. Therefore, the objective function is

$$\min_{y_{ij,l}, r_l} \max p_i \sum_{l \in \mathcal{L}} \sum_{j \in T_i} y_{ij,l} r_l \tag{13}$$

Under a multi-node attack, we consider the worst case in which every node might be compromised. The path for

commodity l is compromised as long as at least a relay node on that path is compromised. Based on this, the probability that the path for commodity l is compromised is:

$$\alpha_l = 1 - \prod_i (1 - p_i \sum_{j \in T_i} y_{ij,l}) \tag{14}$$

Therefore, in the worst case, the traffic amount the adversary can get is $\sum_l \alpha_l r_l$ and the objective function can be formulated as

$$\min_{y_{ij,l}, r_l} \sum_l \alpha_l r_l \tag{15}$$

We only consider the single-node attack in this paper. The multi-node attack scenario can intuitively be modelled as a mixed integer non-convex non-linear problem, which is much more complex and we will not discuss it in this paper due to space limit.

In summary, the optimal traffic splitting routing problem in this paper can be formulated s follows.

$$\min_{y_{ij,l}, r_l} \max \quad p_i \sum_{l \in \mathcal{L}} \sum_{j \in T_i} y_{ij,l} r_l \tag{16}$$

$$\text{s.t.} \quad y_{ij_l} \in \{0, 1\} \tag{17}$$

$$\sum_{j \in T_s} \sum_{l \in \mathcal{L}} r_l \cdot y_{js,l} = 0 \tag{18}$$

$$\sum_{j \in T_s} \sum_{l \in \mathcal{L}} r_l \cdot y_{sj,l} = F \tag{19}$$

$$\sum_{l \in \mathcal{L}} r_l = F \tag{20}$$

$$\sum_{j \in T_i}^{j \neq s} r_l \cdot y_{ij,l} = \sum_{k \in T_i}^{k \neq d} r_l \cdot y_{ki,l} \quad \forall i \in \mathcal{N}, l \in \mathcal{L} \tag{21}$$

$$\sum_{l \in \mathcal{L}} r_l \cdot y_{ij,l} \leq c_{ij} \quad \forall (i,j) \in E \tag{22}$$

$$\sum_{j \in T_i} y_{ij,l} \leq 1 \quad \forall i \in \mathcal{N} \forall l \in \mathcal{L} \tag{23}$$

III. A Heuristic Algorithm for Traffic Splitting

The complexity of the optimization above comes from two parts: (i) allocating the data volume for each commodity and (ii) determining the binary variables $y_{ij,l}$. To find the globally optimal solution is NP-hard, therefore we develop a heuristic algorithm to solve this problem.

First we denote $\max_{y_{ij,l}, r_l} p_i \sum_{l \in \mathcal{L}} \sum_{j \in T_i} y_{ij,l} r_l$ as t. Therefore, the original optimization problem can be represented as:

$$\min_{y_{ij,l}, r_l, t} \quad t \tag{24}$$

$$\text{s.t.} \quad t \geq p_i \sum_{l \in \mathcal{L}} \sum_{j \in T_i} y_{ij,l} r_l \tag{25}$$

$$(17), (18), (19), (20), (21), (22), (23)$$

This is still a mixed-integer non-linear problem. We solve this problem with performance bound for the heuristic algorithm. First, we fix the flow rate of each commodity r_l. By doing this, the problem becomes a mixed-integer linear problem and we can use branch-and-bound [15] to solve it. Note that when the scale of the network is large, branch-and-bound method may become slow. Thus, to improve the efficiency, we can relax $y_{ij,l}$ to $[0, 1]$ and then transform the result back to $y_{ij,l} \in \{0, 1\}$ according to a specific criterion. Second, we fix $y_{ij,l}$ and the problem becomes a linear problem. Then we use dual-simplex method to solve it. We iteratively repeat these two steps until the objective function (24) converges.

Algorithm 1 gives a summary of the proposed solution. Here, δ is a small value used to judge whether the objective function converges. We will compare our solution with the method in which we directly relax $y_{ij,l} \in \{0, 1\}$ to $y_{ij,l} \in [0, 1]$. This method gives a lower bound for the globally optimal solution of the original problem.

Algorithm 1 Summary of the Proposed Scheme

1: **Initialize** variables $r_l^{(0)}$; set $t^{(0)} = 1$, $t^{(1)} = 0$, $a = 1$
2: **while** $|t^{(a)} - t^{(a-1)}| \geq \delta$ **do**
3: Solve the following problem with $r_l = r_l^{(a-1)}$and obtain $y_{ij,l}^{(a)}$:

$$\min_{y_{ij,l}, t} \quad t$$
$$\text{s.t.} \quad (25), (17), (18), (19), (20), (21), (22), (23)$$

4: Solve the following problem with $y_{ij,l} = y_{ij,l}^{(a)}$ and obtain $r_l^{(a)}$ and $r^{(a)}$:

$$\min_{r_l, t} \quad t$$
$$\text{s.t.} \quad (25), (17), (18), (19), (20), (21), (22), (23)$$

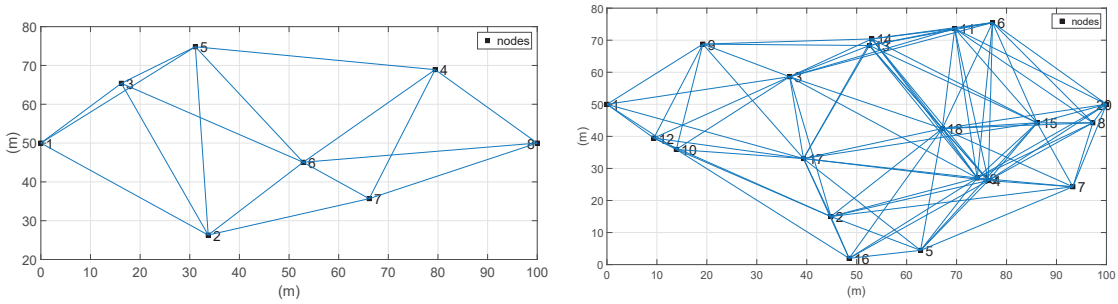
5: $a = a + 1$
6: **end while**
7: Output $y_{ij,l}^{(a)}$,$r_l^{(a)}$,$t^{(a)}$

IV. Simulation Results

A. Simulation Setup

We conduct simulations with two networks: network (a) consisting of 8 randomly generated nodes in a $100 * 60m^2$ area as shown in Fig. 2(a) and network (b) consisting of 20 randomly generated nodes in a $100 * 80m^2$ area as shown in Fig. 2(b). The source node is denoted as node 1, the destination node is denoted as node 20, and the rest are relay nodes. We assume the transmission range of each node in (a) to be 50m and in (b) 40m. Besides, the link capacity is set as 1 M/s for each link. The total flow rate is set as 1 M/s. The probability that node i is compromised in (a) and (b) is set as shown in Table II.

δ is set to 10^{-5}. Note that there may exist no feasible solutions if the required flow rate is too large, that is, even if all links are used, there are not enough capacity to transmit the traffic. In this paper, we assume there always enough capacity to transmit data.



(a) (b)

Fig. 2. (a) The network consisting 8 nodes and (b) the network consisting 20 nodes with the locations and connectivity between these nodes

TABLE II
PROBABILITY THAT A NODE IS COMPROMISED IN NETWORK (A) AND NETWORK (B)

Network (a)		Network (b)			
node i	p_i	node i	p_i	node i	p_i
1	0	1	0	11	0.4684
2	0.2	2	0.1368	12	0.5053
3	0.3	3	0.1737	13	0.5421
4	0.4	4	0.2105	14	0.5789
5	0.5	5	0.2474	15	0.6158
6	0.6	6	0.2842	16	0.6526
7	0.7	7	0.3211	17	0.6895
8	0	8	0.3579	18	0.7263
		9	0.3947	19	0.7632
		10	0.4316	20	0

B. Results and Analysis

Table III and Table IV show the routing results and the corresponding flow rate for each route in network (a) and (b), respectively. The simulation results match our intuition well: based on the form of the objective function, if a node with the largest probability of compromise in route l_1 is smaller than that in l_2, l_1 will have higher priority when the algorithm arranges the traffic, that is, most traffic will be distributed over l_1. We also notice that sometimes redundant node might be introduced, such as node 9 in second route when there are 3 routes in network (b). This is reasonable because the redundant nodes indicate the efforts our algorithm makes to split the traffic to minimize the traffic an adversary can receive at a node.

Fig. 3(a) and Fig. 3(b) show the impact of the number of routes on the expectation of traffic an adversary compromises in both network (a) and network (b). As we can see, by splitting traffic, we significantly limit the quantity of traffic an adversary can receive. Let

$$t_{sec} = \frac{\text{optimal result}}{\text{total amount of traffic}} \tag{26}$$

denote the security level and threshold t_1 the required security level. If $t_sec \leq t_1$, our scheme is considered to be successful when preventing the adversary from identifying smartphone app.

Note that the more routes we have, the more secure we achieve, but the more communication overhead is required. To balance the trade-off between security and communication cost, we compare the heuristic algorithm with the algorithm in which $y_{ij,l}$ is relaxed to $[0, 1]$ and gives a lower bound for the globally optimal solution. The result shows that the heuristic method approaches to the lower bound when the number of commodities becomes larger. Let

$$gap = \frac{1}{\text{optimal result} - \text{lower bound}} \tag{27}$$

denote the effectiveness of the heuristic algorithm and threshold t_2 the required performance level. If $gap \geq t_2$, our scheme is considered to achieve satisfactory performance. Typically, if there are more relay nodes, we need more routes to make our scheme more effective.

V. CONCLUSION AND FUTURE WORK

In this paper, we propose a multipath routing scheme against traffic analysis, and present numerical results for the proposed scheme. We intend to limit the traffic that a capability-limited adversary can intercept. We formulate our problem as a flow routing optimization problem with network constraints and solve it with a heuristic algorithm. Through simulation study, we demonstrate that our scheme can work well against traffic analysis with satisfactory results. In our future work, we plan to investigate the scenario in which the adversaries collude with each other and share their information, i.e., the multi-node attack scenario.

REFERENCES

[1] G Laurence and R Janessa. Gartner says smartphone sales surpassed one billion units in 2014. https://www.gartner.com/newsroom/id/2996817, 2015.

[2] Number of apps available in leading app stores as of march 2017. https://www.statista.com/statistics/276623/number-of-apps-available-in-leading-app-stores/.

[3] The total number of mobile app downloads in 2017. http://www.businessofapps.com/data/app-statistics/.

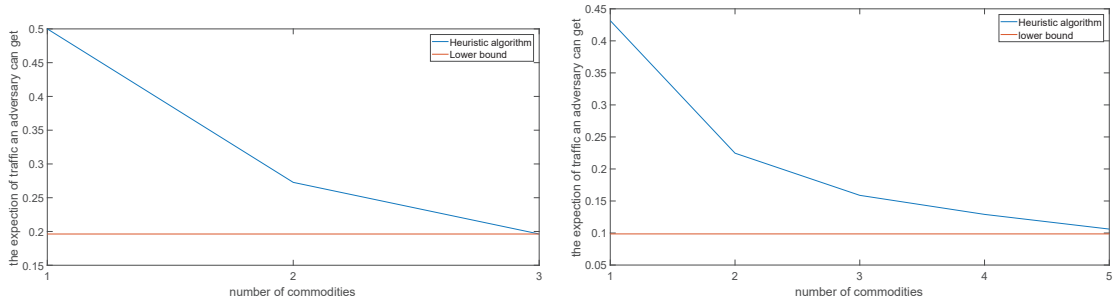
[4] Shuaifu Dai, Alok Tongaonkar, Xiaoyin Wang, Antonio Nucci, and Dawn Song. Networkprofiler: Towards automatic fingerprinting of android apps. In *INFOCOM, 2013 Proceedings IEEE*, pages 809–817. IEEE, 2013.

TABLE III
ROUTING RESULTS AND CORRESPONDING FLOW RATE FOR EACH PATH FOR 1, 2, AND 3 ROUTES FOR NETWORK (A)

# of routes	optimal result	flow rate of each route	routes			
1	0.5	1	1	5	4	8
2	0.2727	0.5455	1	5	4	8
		0.4545	1	3	6	8
3	0.1963	0.3925	1	5	4	8
		0.3271	1	3	6	8
		0.2804	1	2	7	8

TABLE IV
ROUTING RESULTS AND CORRESPONDING FLOW RATE FOR EACH PATH FOR 1, 2, 3, 4, AND 5 ROUTES FOR NETWORK(B)

# of routes	optimal result	flow rate of each route	routes								
1	0.4316	1	1	10	2	4	20				
2	0.2246	0.5205	1	10	2	4	20				
		0.4795	1	3	11	20					
3	0.1588	0.3680	1	10	2	4	20				
		0.3390	1	9	3	11	20				
		0.2930	1	3	13	6	20				
4	0.1291	0.2991	1	10	2	4	20				
		0.2756	1	3	11	20					
		0.2381	1	3	13	6	20				
		0.1872	1	9	3	12	17	14	6	8	20
5	0.1061	0.2460	1	10	2	4	20				
		0.2266	1	3	11	20					
		0.1958	1	9	13	6	20				
		0.1777	1	3	14	6	20				
		0.1540	1	3	17	4	20				



(a) (b)

Fig. 3. (a) The network consisting 8 nodes and (b) the network consisting 20 nodes with the locations and connectivity between these nodes

[5] Hossein Falaki, Dimitrios Lymberopoulos, Ratul Mahajan, Srikanth Kandula, and Deborah Estrin. A first look at traffic on smartphones. In *Proceedings of the 10th ACM SIGCOMM conference on Internet measurement*, pages 281–287. ACM, 2010.

[6] Vincent F Taylor, Riccardo Spolaor, Mauro Conti, and Ivan Martinovic. Appscanner: Automatic fingerprinting of smartphone apps from encrypted network traffic. In *Security and Privacy (EuroS&P), 2016 IEEE European Symposium on*, pages 439–454. IEEE, 2016.

[7] Sebastian Zander, Thuy Nguyen, and Grenville Armitage. Automated traffic classification and application identification using machine learning. In *Local Computer Networks, 2005. 30th Anniversary. The IEEE Conference on*, pages 250–257. IEEE, 2005.

[8] Tim Stöber, Mario Frank, Jens Schmitt, and Ivan Martinovic. Who do you sync you are?: smartphone fingerprinting via application behaviour. In *Proceedings of the sixth ACM conference on Security and privacy in wireless and mobile networks*, pages 7–12. ACM, 2013.

[9] Wenjing Lou, Wei Liu, and Yuguang Fang. Spread: Enhancing data confidentiality in mobile ad hoc networks. In *INFOCOM 2004. Twenty-third AnnualJoint Conference of the IEEE Computer and Communications Societies*, volume 4, pages 2404–2413. IEEE, 2004.

[10] Patrick PC Lee, Vishal Misra, and Dan Rubenstein. Distributed algorithms for secure multipath routing in attack-resistant networks. *IEEE/ACM Transactions on Networking*, 15(6):1490–1501, 2007.

[11] Tao Shu, Marwan Krunz, and Sisi Liu. Secure data collection in wireless sensor networks using randomized dispersive routes. *IEEE transactions on mobile computing*, 9(7):941–954, 2010.

[12] Jinho Choi. Secure multipath routing in wireless multihop networks based on erasure channel modeling. In *Wireless Advanced (WiAd), 2012*, pages 6–10. IEEE, 2012.

[13] Wenjing Lou and Yuguang Fang. Predictive caching strategy for on-demand routing protocols in wireless ad hoc networks. *Wireless networks*, 8(6):671–679, 2002.

[14] Y Thomas Hou, Yi Shi, and Hanif D Sherali. Spectrum sharing for multi-hop networking with cognitive radios. *IEEE Journal on selected areas in communications*, 26(1), 2008.

[15] Branch and bound. https://en.wikipedia.org/wiki/Branch_and_bound.