Towards Securing Internet eXchange Points Against Curious onlooKers

Marco Chiesa
Université catholique de Louvain

Daniel Demmler
Technische Universität Darmstadt

Marco Canini
Université catholique de Louvain

Michael Schapira
Hebrew University of Jerusalem

Thomas Schneider
Technische Universität Darmstadt

ABSTRACT

The growing relevance of Internet eXchange Points (IXPs), where an increasing number of networks exchange routing information, poses fundamental questions regarding the privacy guarantees of confidential business information. To facilitate the exchange of routes among their members, IXPs provide Route Server (RS) services to dispatch the routes according to each member's export policies. Nowadays, to make use of RSes, these policies must be disclosed to the IXP. This state of affairs raises privacy concerns among network administrators and even deters some networks from subscribing to RS services. We design SIXPACK (which stands for "Securing Internet eXchange Points Against Curious onlooKers"), a RS service that leverages Secure Multi-Party Computation (SMPC) techniques to keep export policies confidential, while maintaining the same functionalities as today's RSes. We assess the effectiveness and scalability of our system by evaluating our prototype implementation and using traces of data from one of the largest IXPs in the world.

CCS Concepts

•Security and privacy → Privacy-preserving protocols; •Networks → Network privacy and anonymity;

Keywords

internet exchange points; routing; privacy; secure multiparty

1. BACKGROUND

Protecting the privacy of sensitive business data on the Internet is a topic that is subject to ever-growing attention in a highly-connected, insecure world. We focus on the goal of preventing the leakage of business policies in Internet routing.

With the advent of Internet eXchange Points as the new physical convergence points for Internet traffic, new privacy concerns arise. *Internet eXchange Points* (IXPs) are shared network infrastructures where heterogeneous economic entities meet in order to exchange Internet traffic with each other [2]. To do so, each IXP *member* first establishes physical connectivity with the IXP network and then it both announces the set of IP prefix destinations for which it is willing to receive traffic and starts receiving route announcements from the other members of the IXP.

The Border Gateway Protocol is used to spread and select the routes used to reach prefixes among each pair of members. At medium to large IXPs, a *Route Server* (RS) service is introduced to ease the exchange of BGP announcements among multiple members [10]. The RS establishes a BGP session with each of the IXP members, collects and distributes their BGP announcements according to each member's *export policy*, i.e., the set of other IXP members that are allowed to receive the route announcement originated by a member.

Despite the fact that such a RS service eases the management of BGP sessions and facilitates peering, its usage is not widespread. One of the main barriers is that the export policy of each member must be revealed to the IXP in order to correctly forward the BGP announcements. This information is considered confidential, primarily due to commercial reasons. Indeed, our interaction with network administrators reveals such privacy concerns and, moreover, some networks avoid connecting to RSes for precisely this reason. We point out that beyond privacy concerns, revealing sensitive information also entails the potential risk of triggering attacks [8].

Related work. The work most related to ours is [6], which was the first to apply an SMPC approach to interdomain routing. The performance of [6] is far from practical as it tries to solve the global Internet routing problem. We focus on IXPs and study the technical challenges that arise in this specific context by providing practical SMPC implementations that are beyond the scope of this short paper.

2. THE SIXPACK DISPATCHER SYSTEM

We propose a practical solution for protecting confidential peering policies of the IXP members, i.e., the specification of what BGP routes a member is willing to announce to other members, by executing today's RS services on critical data via SMPC [4, 11]. SMPC allows multiple parties to jointly compute the outcome of a function f while keeping their inputs to it and the outputs of f private. We leverage state-of-the-art accomplishments in SMPC to design SIXPACK, the first IXP route server service for ranking, selecting, and dispatching BGP routes without leaking any confidential

ANRW '16, July 16, 2016, Berlin, Germany

ISBN 978-1-4503-4443-2/16/07...$15.00

DOI: http://dx.doi.org/10.1145/2959424.2959427

business peering information. In our design, we outsource the SMPC part to two non-colluding computing parties, called Route Server 1 and Route Server 2, which carry out the dispatching of BGP routes (see Fig. 1). Each IXP member announces its BGP routes for each IP prefix destination to the RSes in plain text, and creates two "shares" of its (private) peering policy that are sent to the two route servers. Each route server, in turn, sends to each IXP member, upon completion of the SMPC, a share of its output, that the member can use to recover its selected routes. We consider two different route dispatch approaches, which differ in the number of routes that are exported to the IXP members: *SINGLE* and *ALL*.

SINGLE. In this dispatching approach, SIXPACK collects BGP announcements from all the IXP members, computes the best exportable route for each member, and dispatches to each member its selected route. This approach executes the exact same functionality as today's RS, while in addition preserving confidentiality. Performing the best route selection process at the IXP has two main advantages. First, an IXP may have additional information relevant to the route computation than an IXP member (e.g., knowledge of congestion level and other performance metrics). Second, some BGP routers have limited capacity and are incapable of coping with hundreds of thousands of BGP routes.

ALL. In this dispatching approach, SIXPACK relays all exportable BGP announcements between its members. That is, SIXPACK performs route filtering so as to enforce members' export policies, but does not select best routes for its members. This approach *enhances* the functionality of today's RS and preserves confidentiality. The main advantage here is that each IXP member now has the ability to select its route according to its own preference of routes (reflecting, e.g., its business and operational interests).

Threat model. To clarify our security assumptions, our threat model focuses on the RSes as parties in the SMPC computation that have a perfect view of all BGP routes (but not the export policies) announced through the IXP but do not monitor the actual flow of traffic. We assume that both parties adhere to the protocol but attempt to infer as much information as possible about the private inputs (i.e., export policies) of the IXP members. Our goal is to prevent both parties from learning anything about these private inputs.

Route dispatching. In Fig. 1, we show an overview of the SIXPACK mechanism for the ALL approach. Member A wants to announce a route R to member B. It first sends R to RS 1 stripping off the export policy, which is secret-shared with both RSes. The first RS generates a key K, encrypts R with K, and redistributes the encrypted route to all the members. The key K is given in input to the SMPC, which guarantees that only the members who are, based on the export policy, allowed to receive R will be able to reconstruct the key K from the SMPC output, while the non-legitimate receivers will receive a dummy key. Observe that the export policy is never visible at either of the two RSes.

3. IMPLEMENTATION AND EVALUATION

We implemented a prototype for SIXPACK. The SMPC part of the system, i.e., the two RSes, is implemented using the ABY framework [3]. ABY provides low-level primitives, written in C++, for building SMPC functions that are evaluated with the GMW protocol [4]. The computation of an SMPC function consists of a setup phase, which can be precomputed, and an online phase, which depends on the real inputs. For implementing the rest of the system, i.e., the distribution and processing of all the BGP updates among RSes and IXP members, we used Python. We assess our system using a two-hour trace of BGP updates from one of the largest IXPs, with more than 600 members. The measurements were performed on two computers with a 3.5 GHz CPU and 16 GiB RAM connected via a local Gbps network.

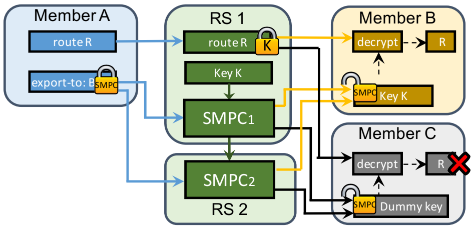

Figure 1: Conceptual overview of SIXPACK

The worst setup and online runtime we measured in our evaluation were 72ms and 19 ms, respectively, for 32 inputs in the SINGLE case. Our unoptimized SIXPACK prototype, written in Python, processed 99% of the BGP updates in real-time with a latency of 120 ms and negligible communication overhead, even without precomputing the setup phase of the SMPC. It is worth to recall that the convergence time of BGP routing in the Internet is in the order of minutes [7, 9].

4. DISCUSSION AND FUTURE WORK

We believe that our scheme can increase trust in IXPs and motivate further adoption of RS services.

Choosing the non-colluding parties. We assume that the RS service at the IXP consists of two route servers that are operated by non-colluding parties. We believe that neutral international entities (e.g., RIPE), who are trusted in the Internet networks for services such as DNS, can be assigned the task of each running an instance of a RS. These instances are executed on machines that connect to the IXP network at the same colocation data center where the IXP network is hosted, hence minimizing the risk of being compromised by the IXP while keeping latency to a minimum [1].

Optimization. The running time of SIXPACK can be further reduced by means of the following three improvements: (i) precomuting the setup phase, (ii) exploiting the fact that the BGP route distribution is prefix-independent (even per-route independent in the ALL approach) to parallelize the computation of the BGP prefixes, and (iii) rewriting the Python code in a more performance-oriented programming language, thus gaining an additional decrease in runtimes.

Future work. We believe that future research should concentrate on extending the functionality of privacy-preserving RS services. One interesting direction is extending SIXPACK to receive as input (beyond members' export policies) also members' (private) local preferences over BGP routes and then running SMPC to select the best (exportable) path per member. We envision a system that combines the IXP members' routing policies with the IXP's global information (e.g., members port congestion) to provide better route dispatching in a privacy preserving manner. Another interesting direction is to apply the privacy-preserving techniques presented in this paper to the next-generation IXP fabrics based on the Software-Defined-Networking paradigm [5].

5. ACKNOWLEDGEMENTS

This research is (in part) supported by European Union's Horizon 2020 research and innovation programme under the ENDEAVOUR project (grant agreement 644960). The work of the 4'th author was supported by the Israeli Center for Research Excellence in Algorithms and an ERC Starting Grant. This work was supported by the European Union's 7^{th} Framework Program (FP7/2007-2013) under grant agreement n. 609611 (PRACTICE), by the DFG as part of project E3 within the CRC 1119 CROSSING, and by the German Federal Ministry of Education and Research (BMBF) within CRISP (www.crisp-da.de).

6. REFERENCES

[1] Real-Time-Statistics AMS-IX. https://ams-ix.net/technical/statistics/real-time-stats.

[2] B. Ager, N. Chatzis, A. Feldmann, N. Sarrar, S. Uhlig, and W. Willinger. Anatomy of a Large European IXP. In *SIGCOMM'12*, 2012.

[3] D. Demmler, T. Schneider, and M. Zohner. ABY – A Framework for Efficient Mixed-Protocol Secure Two-Party Computation. In *NDSS'15*.

[4] O. Goldreich, S. Micali, and A. Wigderson. How to Play any Mental Game or A Completeness Theorem for Protocols with Honest Majority. In *STOC'87*.

[5] A. Gupta, R. MacDavid, R. Birkner, M. Canini, N. Feamster, J. Rexford, and L. Vanbever. An Industrial-Scale Software Defined Internet Exchange Point. In *NSDI'16*.

[6] D. Gupta, A. Segal, A. Panda, G. Segev, M. Schapira, J. Feigenbaum, J. Rexford, and S. Shenker. A new approach to interdomain routing based on secure multi-party computation. In *HotNets'12*.

[7] C. Labovitz, A. Ahuja, A. Bose, and F. Jahanian. Delayed Internet Routing Convergence. In *SIGCOMM'00*.

[8] S. Machiraju and R. H. Katz. Reconciling Cooperation with Confidentiality in Multi-Provider Distributed Systems. Technical report, EECS Department, University of California, Berkeley, Aug 2004.

[9] Z. M. Mao, R. Bush, T. Griffin, and M. Roughan. BGP Beacons. In *IMC'03*.

[10] P. Richter, G. Smaragdakis, A. Feldmann, N. Chatzis, J. Boettger, and W. Willinger. Peering at Peerings: On the Role of IXP Route Servers. In *IMC'14*.

[11] A. C. Yao. How to Generate and Exchange Secrets. In *FOCS'86*.