# Collaborative Network Outage Troubleshooting with Secure Multiparty Computation

*Mentari Djatmiko, NICTA and UNSW*
*Dominik Schatzmann and Xenofontas Dimitropoulos, ETH Zurich*
*Arik Friedman, NICTA*
*Roksana Boreli, NICTA and UNSW*

## ABSTRACT

Troubleshooting network outages is a complex and time-consuming process. Network administrators are typically overwhelmed with large volumes of monitoring data, like SMTP and NetFlow measurements, from which it is very hard to separate between actionable and non-actionable events. In addition, they can only debug network problems using very basic tools, like *ping* and *traceroute*. In this context, intelligent correlation of measurements from different Internet locations is essential for analyzing the root cause of outages. However, correlating measurements across domains raises privacy concerns and hence is largely avoided. A possible solution to the privacy barrier is secure multi-party computation (MPC), that is, a set of cryptographic methods that enable a number of parties to aggregate private data without revealing sensitive information. In this article, we propose a distributed mechanism based on MPC for privacy-preserving correlation of NetFlow measurements from multiple ISPs, which helps in the diagnosis of network outages. We first outline an MPC protocol that can be used to analyze the scope (local, global, or semi-global) and severity of network outages across multiple ISPs. Then we use NetFlow data from a medium-sized ISP to evaluate the performance of our protocol. Our findings indicate that correlating data from several dozens of ISPs is feasible in near real time, with a delay of just a few seconds. This demonstrates the scalability and potential for real-world deployment of MPC-based schemes. Finally, as a case study we demonstrate how our scheme helped analyze, from multiple domains, the impact that Hurricane Sandy had on Internet connectivity in terms of scope and severity.

## INTRODUCTION

Internet outages caused by events like fiber cuts, power failures, routing problems, and prefix hijacking attacks may disrupt critical services; hence, it is essential to troubleshoot them in a timely manner. Past events have shown that network outages can be detrimental and extremely costly. For example, www.amazon.com was unreachable in January 2013 for approximately one hour, which, according to the latest earnings reports of the web site, corresponds to losses of US$4.9 million.[1] Besides, in October 2012, as a result of Hurricane Sandy, a large number of networks were disconnected from the Internet for several days.[2] Such large-scale outages that affect a very large number of clients are only the tip of the iceberg. In addition, a much larger number of lower-impact outages occur in the Internet on a daily basis. These smaller impact outages are even harder to troubleshoot because their root causes are often hidden.

Troubleshooting outages requires determining their root cause. An important part of root cause analysis is diagnosis of the scope and impact of an outage. In particular, identifying whether an outage is a local or global problem is typically the very first step in troubleshooting, while knowing the impact of an outage (i.e., the number of clients that were affected) is essential for prioritizing events. However, answering even these simple questions is difficult with existing network monitoring tools since they provide very little support for debugging network problems. Network operators often revert to mailing lists to request help in troubleshooting reachability problems from multiple vantage points. For example, the following message was posted in the NANOG mailing in March 2008: “Please try to reach my network 194.9.82.0/24 from your networks …

[1] *http://www.networkworld.com/news/2013/013113-amazoncom-suffers-outage-nearly-5m-266314.html?hpg1=bn*

[2] *http://www.zdnet.com/hurricane-sandy-doubled-failures-in-us-internet-infrastructure-7000009010/*

0163-6804/13/$25.00 © 2013 IEEE

Kindly anyone assist." Presently, such messages are still very common in network operators' mailing lists, which highlights the lack of automated schemes for troubleshooting outages.

## Outage Detection Approaches

The literature has exploited two main data sources for detecting and troubleshooting network outages [1–18]: *control plane measurements* and *active data plane measurements*.

Control plane measurements, particularly Border Gateway Protocol (BGP) update messages, have been used extensively to passively observe and detect anomalies in the announcement of BGP prefixes caused by events like prefix hijacking attacks. These approaches look for anomalous changes in the origin autonomous system (AS) or the AS path of a prefix. The control plane is typically observed from several vantage points using operational BGP data collected in real time by projects like RouteViews [19] and the RIPE Routing Information Service (RIS) [20]. The drawback of this approach is that many legitimate events may also result in changes in the path or origin of a prefix, and therefore can trigger a large number of false positive alarms.

A second frequently used data source is active data plane measurements, which are collected with tools like ping and traceroute. This approach periodically probes a number of destinations from multiple vantage points. Active probing helps reduce the number of false positives and localize outages; however, it involves a cumbersome trade-off between measurement overhead and detection granularity. Collecting fine-grained measurements over the global Internet requires frequently probing an enormous number of destinations, which is not scalable due to probing overhead. Reducing the overhead comes at the cost of introducing sampling in the spatial (i.e., IP address space) and temporal dimensions. For this reason, active data plane measurements are typically best suited for either monitoring a small set of destinations or detecting only large-scale events. These approaches typically exploit distributed active measurement platforms, like iPlane [21] and Archipelago [22].

In addition, perfSonar [24, 25] is a general-purpose collaborative network monitoring platform that provides several tools for network troubleshooting and is deployed by a number of independent research and educational networks. It is effectively used in this ecosystem for sharing and correlating non-sensitive network monitoring data, like certain Simple Network Management Protocol (SNMP) counters. However, it does not provide cryptographic mechanisms for privacy-preserving computations on highly sensitive data, like raw NetFlow records. Our work fills this gap in the context of collaborative network monitoring and specifically for the problem of troubleshooting network outages.

### A New Approach Based on One-way Traffic Measurements

Recently, a number of studies have exploited passive network performance measurements collected from ISPs [26, 27], end hosts [28], or network telescopes [29] for network outage detection. In this article, we focus specifically on passive data plane measurements of one-way traffic, which is a new approach to outage detection introduced in our previous work [26, 27]. Traffic measurements can be collected passively using flow monitoring technologies, like NetFlow. The key idea is the following: network outages result in:

- An increase in the number of unsuccessful (one-way) connections to a remote destination (IP address, prefix, or AS) and concurrently
- A decline in the number of successful (two-way) connections

These changes can easily be observed from passive network traffic measurements. In other words, we can passively use the regular traffic of a network as probe traffic for outage detection. This provides a rich source of data without imposing additional overhead for active measurements, and detects only the outages that actually affect the clients of a monitored network. In addition, one-way traffic monitoring enables prioritization of outages based on the number of clients that were affected by an event. The *Flow-Based Approach for Connectivity Tracking* (FACT) [26] is a system that implements these ideas. It monitors the traffic that enters and leaves a domain, and identifies successful and unsuccessful connections. A network outage is detected when a significant number of unsuccessful connections is associated with a specific destination.

## Down for Everyone or Is It Just Me?

While the benefits of FACT include the use of passive traffic measurements to detect outages, the detection outcomes for individual domains (Internet service providers, ISPs) may not be sufficient to determine the root cause of specific outages or troubleshoot them. A useful way to extend the results of local monitoring is to aggregate detection results from multiple domains. This approach has the potential to improve the results in determining both the scope of an outage and its impact across domains. First, the scope of an outage is essential for troubleshooting as it helps narrow down the potential cause(s) of the outage. A *local outage* is only detected by a single domain, a *global outage* is detected by all the participating domains, while a *semi-global outage* is detected by some domains but not by others. Second, to further facilitate troubleshooting, it is also beneficial to obtain information about the severity of the outage, that is, the number of distinct domains that cannot reach a specific destination and the total number of clients that were affected by the outage. We note that the scope and severity information needs to be determined in near real time by the ISPs to enable timely corrective actions that result in the improvement of service quality.

Unfortunately, aggregation of monitoring information across multiple ISPs is not a straightforward task. The commercial environment does not encourage sharing of commercial-

*A useful way to extend the results of local monitoring is to aggregate detection results from multiple domains. This approach has the potential to improve the results in determining both the scope of an outage and its impact across domains.*

*MPC is a set of cryptographic methods that, given input data from multiple parties, enable distributed private computation of mathematical functions. MPC provides formal guarantees on the confidentiality of the input and/or intermediate data, and on the correctness of the computation result.*

ly sensitive information related to the service quality of an ISP's network. In addition, distributed monitoring of network outages on the basis of passive network traffic measurements of one-way traffic requires exchanging information about contacted services, IP addresses, prefixes, and ASs. Such data is generally considered sensitive as it relates to personal information about visited web sites and services used by individuals. Therefore, the disclosure of such information is typically prohibited by privacy laws and regulation.[3] For these reasons, a data aggregation mechanism that preserves the confidentiality of individual records is crucial for enabling computations that utilize such data.

## MPC to the Rescue

Secure multi-party computation (MPC) [30] provides a viable solution to this problem. MPC is a set of cryptographic methods that, given input data from multiple parties, enable distributed private computation of mathematical functions. MPC provides formal guarantees on the confidentiality of the input and/or intermediate data, and on the correctness of the computation result. Until recently, the research field of MPC was almost exclusively of theoretical interest (e.g., improving the security of MPC and reducing the computational complexity), with very limited real-world use due to the prohibitive computational overhead. This, however, has started to change in the last few years, due to improved MPC protocols, with a number of demonstrations of feasible real-world applications.

In this work we build on a popular MPC framework based on Shamir's secret sharing (SSS) scheme [31], and it is implemented in the SEPIA MPC library [32]. In our framework, *input peers* provide data, and *privacy peers* perform the secure computation of a specific function of that data. The number of input peers directly affects the complexity of MPC, while the number of privacy peers correlates not only with the performance but also with the resilience of the MPC mechanism against various security attacks. MPC's attack resilience scales linearly with the number of privacy peers. Hence, the selection of the number of privacy peers adjusts a trade-off between performance and attack resiliency. We note that a party may function as both an input peer and a privacy peer. In a nutshell, the input peers use SSS to "encrypt" their input data and distribute the shares (i.e., the secret sharing results) to the privacy peers. The privacy peers then perform the computation on the shares and return the final computation results to the input peers.

The confidentiality of the input data is obtained through secret sharing. SSS is the most widely used secret sharing scheme and is based on polynomial interpolation. Given a secret $D$, the $(k, n)$ SSS scheme generates $n$ shares from $D$ and distributes them between $n$ privacy peers. $D$ is only computable given the knowledge of at least $k$ shares (where $k < n$), while the knowledge of $k - 1$ shares does not reveal any information about $D$. Hence, when SSS uses a polynomial of degree $k = \lfloor(n - 1)/2\rfloor$, MPC with SSS is resilient against up to $n/2$ colluding privacy peers who jointly would not be able to obtain any information about either the (private) data or the computation's intermediate results.

In the remainder of this article, we show how MPC can be efficiently integrated with network traffic monitoring to enable privacy preserving troubleshooting of network outages.

## Distributed Outage Monitoring with MPC

We now describe a novel application of MPC for troubleshooting unreachable destinations and how we can efficiently exploit MPC operations to provide privacy preserving near-real-time monitoring of outages. We first describe how we have integrated MPC within the FACT one-way traffic monitoring system to aggregate traffic measurement data across multiple domains/ISPs, while keeping individual ISPs' data hidden from other ISPs as well as from the entities performing data aggregation.

Figure 1 shows the architecture of MPC integration with multiple instances of FACT (Multi-FACT). Multi-FACT extends FACT with the capability to determine the scope and overall impact (severity) of outages. Each ISP operates a local instance of FACT, which collects one-way traffic measurements in the ISP's border routers to identify remote destinations that are unreachable from the local clients of the ISP. The unreachable destinations (e.g., IP addresses and IP prefixes) detected by FACT are aggregated with MPC while maintaining their confidentiality. To accomplish this, ISP data is split into MPC shares using SSS and forwarded to MPC privacy peers, which perform the data aggregation. Finally, the result of the aggregation operation (Multi-FACT output) is fed back to the ISPs for local estimation of the scope and severity of detected outages. Figure 1 presents an example with three MPC privacy peers performing the privacy preserving computations.

### Building Efficient MPC Operations Using Bloom Filters

The local monitoring result each instance of FACT produces is a list of unreachable destinations for different types of destinations: remote IP addresses, /24 prefixes, and BGP prefixes. For each destination, FACT also shows the number of clients that fail to reach it. Our goal is to extend this data with information about the total number of domains and clients that cannot reach a destination by aggregating information from other domains based on MPC. To realize this goal we exploit a multiset union operation. Recall that a multiset is a set in which elements can appear multiple times. To determine the scope of an outage, each FACT instance produces three sets that represent unreachable destinations at the three different levels of granularity above. The output of the union operation over the corresponding sets of all input providers is a multiset in which the multiplicity of each destination represents the number of domains that cannot reach it. In this way, the output multiset shows the number of domains

[3] *For example, the EU General Data Protection Regulation, http://www.ec.europa.eu/justice/data-protection/document/review2012/com_2012_11_en.pdf*

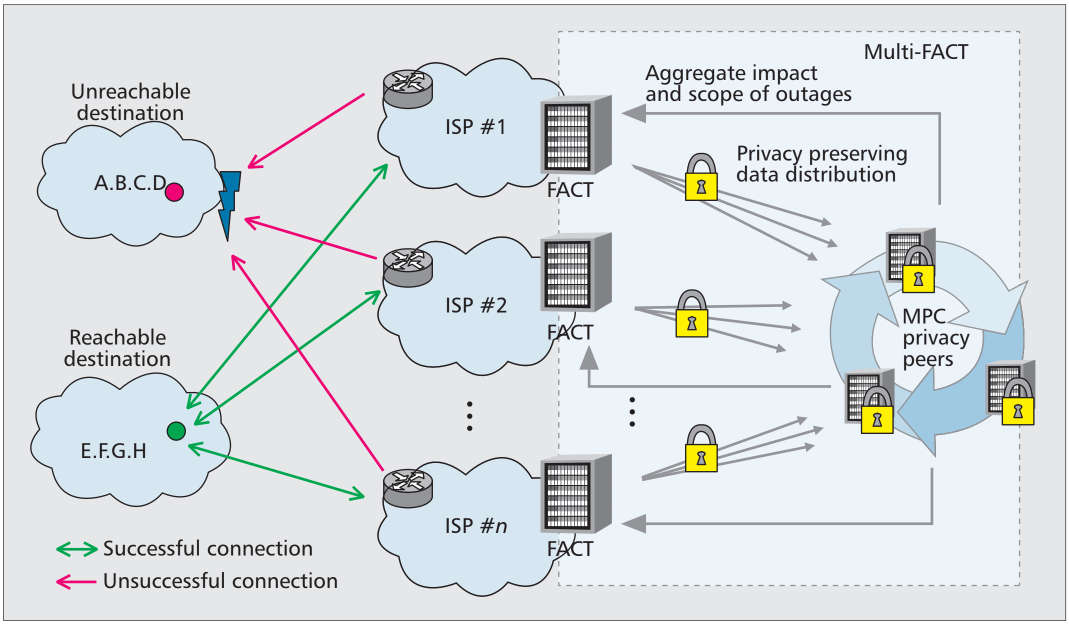

**Figure 1.** *Multi-FACT architecture: ISPs in the middle use FACT to track connectivity to remote destinations. Multi-FACT correlates local observations with MPC to find out the aggregate scope and severity of outages.*

*For each destination, FACT also shows the number of clients that fail to reach it. Our goal is to extend this data with information about the total number of domains and clients that cannot reach a destination by aggregating information from other domains based on MPC.*

that cannot reach a destination, without showing which domains fail. Accordingly, to determine the overall impact of an outage, each FACT instance produces three multisets, in which the multiplicity of each element represents the number of local clients that cannot reach a destination. After the multi-set union operation, the output encodes the total number of clients that fail to reach a destination, without revealing any other information.

The standard way to perform the multiset union operation requires comparisons between all possible combinations of elements. In MPC based on SSS, comparison operations can be up to three orders of magnitude more expensive than additions [32]. Performing all these comparisons would therefore require prohibitive computational overhead. To by-pass this problem, we exploit an efficient multiset union operation that is based on counting Bloom filters (CBFs) [33]. CBF is a space-efficient probabilistic data structure that uses multiple independent hash functions to encode a multiset into a fix-sized array of counters. Multiple CBFs constructed with the same hash functions can be efficiently aggregated simply by summing the corresponding counters. In this way, the output CBF represents the union of the input multi-sets. Therefore, we can perform the multi-set union operation in the CBF domain simply with an MPC addition of the input CBF arrays. Addition operations in the SSS scheme are generally very efficient, since they do not require any complex computation or intermediate sychronization rounds. CBFs can in general introduce an error, as different multiset elements may be mapped into the same array bucket. The error can be minimized by carefully selecting the size of the array. In particular, the false positive rate (FPR) introduced by a CBF in which one inserts *r* input elements can be controlled by setting the size *s* of the CBF based on the following equation:

$$s = 5 \cdot r \cdot (-\log_{10}(\mathrm{FPR})) \qquad (1)$$

## Evaluating MPC-Based Federated Outage Detection

We implemented a prototype of Multi-FACT using the SEPIA MPC library [32]. SEPIA is specifically designed for aggregating network events and statistics from multiple network domains and includes an efficient multiset union operation. In this section, we evaluate the feasibility of the proposed monitoring scheme by evaluating the runtime based on input data sizes derived from real-world incidents observed in a medium-sized ISP. The used data are described in more detail in the next section. Note that the measured computational runtime includes the time to distribute the shares of the input data, the time to compute the multiset union in MPC, and the time to receive the computation result.

The evaluation is performed in an OpenStack cloud, which consists of six workers. Each worker has 12 CPU cores of Intel Xeon 2.67 Gbytes and a gigabit network connection. We use nine virtual machines, where each machine has 2 Gbytes of memory and one virtual CPU based on KVM virtualization. We note that in the worst case, wide area network delays affect the presented experiments only by an additional delay that equals twice the maximum delay time between the ISPs and the privacy peers.

In all experiments, the MPC input peers and privacy peers are uniformly distributed across the virtual machines. We vary the input data size *r*, which corresponds to the number of unreachable destinations (on various levels of granularity), and the false positive rate (FPR) of the

CBF. We set the corresponding CBF size, *s*, according to Eq. 1. Moreover, we also vary the number of ISPs aggregating input data (i.e., the input peers in MPC terminology) and the number of MPC privacy peers. For each experimental configuration, we run Multi-FACT 10 times and compute the running time as the average of all runs.

Figure 2 shows the average computational running time for a varying number of input and privacy peers. We use a fixed input data size of 1729 unreachable destinations (which corresponds to the maximum input size we observed in our experiments with real ISP data) and an FPR of 0.1 percent, which corresponds to a CBF size of 32,768 counters. The number of input peers varies between 10 and 90 peers, matching the scenario in which this number of ISPs collaborates to troubleshoot network outages. The number of MPC privacy peers varies between three and nine peers, providing increasing levels of privacy protection (i.e., requiring between two to five privacy peers to collude to successfully access an ISP's private data). We observe in Fig. 2 that the average computational running time increases linearly with the number of input and privacy peers. This is due to the direct relation between the number of input peers and the number of CBFs to aggregate. Similarly, the running time increases linearly with the number of privacy peers. Furthermore, we see that in all the experiments the total running time is comfortably below 1 s, which is suitable for near-real-time monitoring requirements.

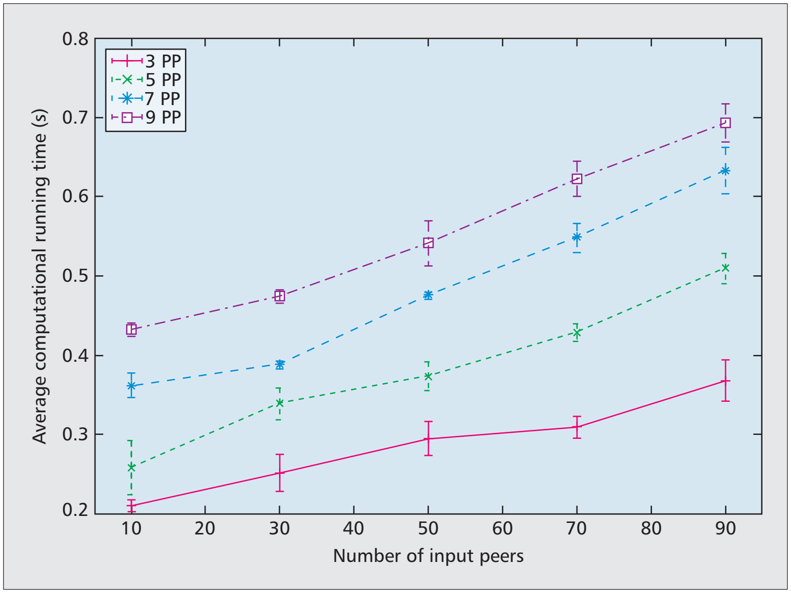

**Figure 2.** *The computational running time average over 10 runs for CBF size 32,768, and varying numbers of input and privacy peers.*

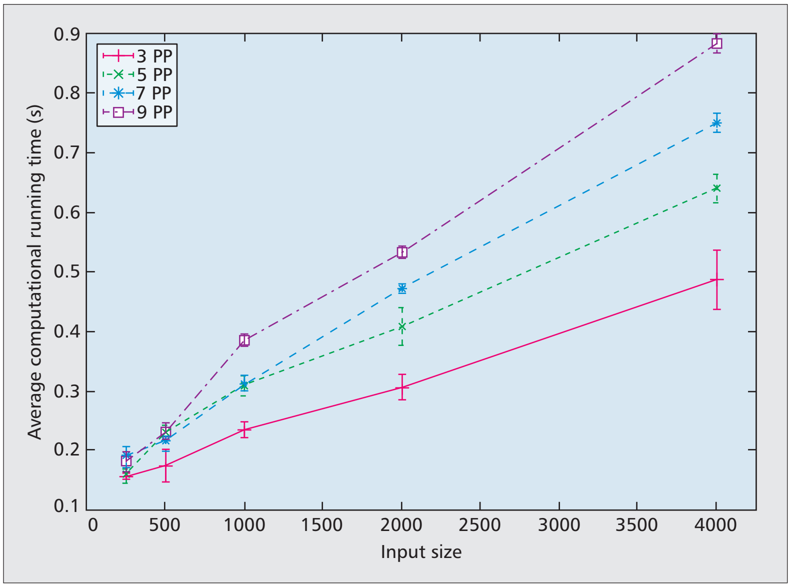

**Figure 3.** *The computational running time average over 10 runs for 50 input peers with varying numbers of privacy peers and varying input data sizes.*

Figure 3 shows the average computational running time as the input data size, and the number of privacy peers varies. In this case, the number of input peers is fixed to 50, while the number of privacy peers varies between three and nine. In Fig. 3 we observe that the running time scales linearly with the size of the input data. Recall that the multiset union operation using CBFs is essentially a fixed-size MPC array summation. Since the input data size correlates to the CBF size, the increase in the input data size results in the summation of larger size arrays. Furthermore, similar to Fig. 2, the running time scales linearly with the number of privacy peers and in all experiments requires less than 1 s.

Therefore, we see that aggregating input sets with several thousands of elements is feasible even with 90 input peers and 9 privacy peers in less than 1 s.

## Federated Outage Monitoring in the Wild: The Impact of Hurricane Sandy

In this section, we demonstrate the utility of our scheme in practice using actual traffic data from an ISP containing a well-known event. On the evening of October 29, 2012, Hurricane Sandy hit the east coast of North America. It was the largest Atlantic and second costliest hurricane on record, with estimates of damage at nearly US$75 billion. It specifically hit New Jersey and New York, where data centers are located, resulting in power loss, fiber cuts, and other damage that reportedly affected Internet connectivity.

We investigate the impact of Hurricane Sandy on the Internet connectivity of a remote location as measured with FACT and demonstrate the utility of correlating data from multiple domains. To address these points, we use unsampled NetFlow data collected between October 29 and November 1, 2012 from the border routers of SWITCH [34]. SWITCH is a medium-sized ISP in Switzerland that connects approximately 40 Swiss universities, government institutions, and research laboratories to the Internet. We analyze the event independently for four domains of SWITCH using FACT alone, and use Multi-FACT to discern the scope and severity of the detected outages.

Figures 4a and 4b show the impact of Hurricane Sandy measured in terms of the number of remote BGP prefixes that were unreachable by one of the four domains during the event. The figure on the top shows the local view of this domain without using Multi-FACT, while the figure on the bottom shows the local view of the domain when Multi-FACT is in use. In both figures we observe that almost 80 BGP prefixes were unreachable from the studied domain during the time of the event. This corresponds to a sharp increase in the number of unreachable prefixes compared to the dates before the event. In addition, we observe that the number of unreachable prefixes decreased over time, but even three days after the event a significant number of prefixes remained unreachable. Furthermore, observe the diurnal pattern, which results from using passive measurements of user traffic to detect outages. In Fig. 4a, we annotate with different colors the number of prefixes that had a higher impact, in terms of affected clients, on the studied domain. The number of affected clients represents the *severity of an outage*. We observe that most prefixes affected a fairly small number of clients, which reflects that Hurricane Sandy affected prefixes that were not particularly popular in Switzerland. The affected prefixes are registered to organizations located primarily in the vicinity of New York. In contrast, in Fig. 4b we use colors to highlight the impact of an outage in terms of the number of distinct domains that were affected. In this case, the colors allow clear discernment of whether the *scope of an event* is local or global. We observe that the vast majority of the affected prefixes were also unreachable for other domains, which helps to identify that the sharp increase in the unreachable destinations is the result of a *global* event that affected multiple organizations, rather than a local misconfiguration.

Recall that the scope of an outage is useful to determine the root cause, which in turn helps in troubleshooting the outage. Therefore, this information can steer the administrators of the affected network to take further troubleshooting steps. In this case, since the event is global and beyond the control of local administrators, the ISP can notify its peers in the remote ISP where the problem originates. In addition, it can provide informed answers to potential complaints about Internet connectivity from its clients. Alternatively, if the event was local, the administrators would have to further troubleshoot connectivity issues in their infrastructure.

In summary, we find that Hurricane Sandy affected several prefixes that were visibly unreachable even in Switzerland. In addition, our private aggregation scheme clearly helps to establish the global or local nature of events.

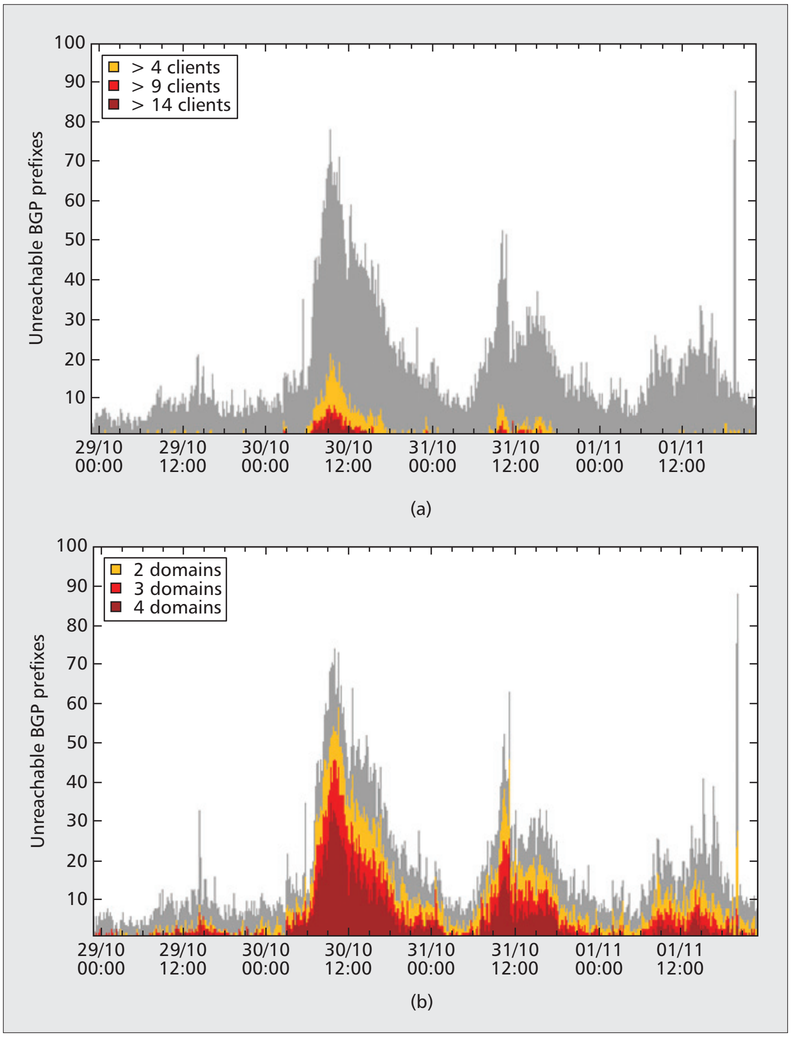

**Figure 4.** *a) Unreachable remote BGP prefixes detected by one domain during Hurricane Sandy: number of affected local clients; b) unreachable remote BGP prefixes detected by one domain during Hurricane Sandy: visibility of the outage across the other domains.*

## Conclusion

We have argued for the necessity of enabling privacy preserving computations in federated measurements for detection of network outages to improve the scope and accuracy of measurements. We have presented an overview of the outage monitoring approaches proposed in the research literature, followed by an extension of a recently proposed passive monitoring-based mechanism. We have shown how MPC can be integrated within a system that aggregates data monitored by multiple ISPs, enabling computations on data aggregates in a privacy preserving way. We have also provided details of a prototype implementation of the federated measurement system and addressed the efficiency requirements for a near-real-time monitoring system by showing that, with careful design and choice of MPC operations, fast and efficient operations are achievable. We hope that the material presented in this article will motivate more privacy preserving federated measurement systems.

## Acknowledgment

This research work has been supported by funding from National ICT Australia (NICTA). NICTA is a research organization funded by Australian Government research initiatives through the Australian Research Council (ARC). We are grateful to SWITCH for providing us data and would also like to thank Bernhard Ager for useful comments on earlier drafts of our article.

*Recall that the scope of an outage is useful to determine the root cause, which in turn helps in troubleshooting the outage. Therefore, this information can steer the administrators of the affected network to take further troubleshooting steps.*

## References

[1] L. Quan, J. Heidemann, and Y. Pradkin, "Trinocular: Understanding Internet Reliability through Adaptive Probing," *Proc. ACM SIGCOMM*, 2013, pp. 255–66.

[2] Y.-J. Chi, R. Oliveira, and L. Zhang, "Cyclops: The AS-Level Connectivity Observatory," *SIGCOMM Comp. Commun. Rev.*, vol. 38, no. 5, Oct. 2008, pp. 5–16.

[3] M. Lad *et al.*, "PHAS: A Prefix Hijack Alert System," *Proc. USENIX Sec. Symp.*, 2006.

[4] X. Hu and Z. Mao, "Accurate Real-Time Identification of IP Prefix Hijacking," *Proc. IEEE Sec. and Priv.*, 2007, pp. 3–17.

[5] Z. Zhang *et al.*, "iSPY: Detecting IP Prefix Hijacking on My Own," *IEEE/ACM Trans. Net.*, vol. 18, no. 6, Dec. 2010, pp. 327–38.

[6] C. Zheng *et al.*, "A Light-Weight Distributed Scheme for Detecting IP Prefix Hijacks in Real-Time," *Proc. ACM SIGCOMM*, 2007, pp. 277–88.

[7] X. Shi *et al.*, "Detecting Prefix Hijackings in the Internet with Argus," *Proc. ACM IMC*, 2012, pp. 15–28.

[8] A. Markopoulou *et al.*, "Characterization of Failures in an Operational IP Backbone Network," *IEEE/ACM Trans. Net.*, vol. 16, no. 4, Aug. 2008, pp. 749–62.

[9] R. Teixeira and J. Rexford, "A Measurement Framework for Pin-Pointing Routing Changes," *Proc. ACM SIGCOMM Wksp. Net.*, 2004, pp 313–18.

[10] Y. Huang *et al.*, "Diagnosing Network Disruptions with Network-Wide Analysis," *SIGMETRICS Perform. Eval. Rev.*, vol. 35, no. 1, June 2007, pp. 61–72.

[11] R. Bush *et al.*, "Internet Optometry: Assessing the Broken Glasses in Internet Reachability," *Proc. SIGCOMM IMC*, 2009, pp. 242–53.

[12] R. Bush *et al.*, "Testing the Reachability of (New) Address Space," *Proc. SIGCOMM INM*, 2007, pp. 236–41.

[13] E. Katz-Bassett *et al.*, "Studying Black Holes in the Internet with Hubble," *Proc. USENIX NSDI*, 2008, pp. 247–62.

[14] E. Katz-Bassett *et al.*, "LIFEGUARD: Practical Repair of Persistent Route Failures," *Proc. ACM SIGCOMM*, 2012, pp. 395–406.

[15] A. Dhamdhere *et al.*, "NetDiagnoser: Troubleshooting Network Unreachabilities Using End-to-End Probes and Routing Data," *Proc. ACM CoNEXT*, 2007.

[16] I. Cunha *et al.*, "Measurement Methods for Fast and Accurate Blackhole Identification with Binary Tomography," *Proc. ACM SIGCOMM IMC*, 2009, pp. 254–66.

[17] A. Schulman and N. Spring, "Pingin' in the Rain," *Proc. ACM SIGCOMM IMC*, 2011, pp. 19–28.

[18] V. Paxson, "End-to-End Routing Behavior in the Internet," *SIGCOMM Comput. Commun. Rev.*, vol. 36, no. 5, Oct. 2006, pp. 41–56.

[19] RouteViews, http://www.routeviews.org/.

[20] RIPE Routing Information Service, http://www.ripe.net/data-tools/stats/ ris/routing-information-service.

[21] H.V. Madhyastha *et al.*, "iPlane: an Information Plane for Distributed Services," *Proc. USENIX OSDI*, 2006, pp. 367–80.

[22] Archipelago Measurement Infrastructure; http://www.caida.org/projects/ ark/.

[23] B. Huffaker *et al.*, "Topology Discovery by Active Probing," *SAINT Wksp.*, 2002, pp. 90–96.

[24] B. Tierney *et al.*, "perfSONAR: Instantiating a Global Network Measurement Framework," *ACM SIGOPS SOSP Wksp. Real Overlays and Distrib. Sys.*, Oct. 2009.

[25] A. Hanemann *et al.*, "PerfSonar: A Service Oriented Architecture for Multidomain Network Monitoring," *Proc. ICSOC*, 2005, pp. 241–45.

[26] D. Schatzmann *et al.*, "FACT: Flow-based Approach for Connectivity Tracking," *Proc. PAM*, 2011, pp. 214–23.

[27] E. Glatz and X. Dimitropoulos, "Classifying Internet One-Way Traffic," *ACM IMC*, 2012, pp. 37–50.

[28] D. Choffnes, F. Bustamante, and Z. Ge, "Crowdsourcing Service-Level Network Event Monitoring," *Proc. ACM SIGCOMM*, 2010, pp. 387–98.

[29] A. Dainotti *et al.*, "Extracting Benefit from Harm: Using Malware Pollution to Analyze the Impact of Political and Geophysical Events on the Internet," *SIGCOMM Comp.. Commun. Rev.*, vol. 42, no. 1, 2012, pp. 31–39.

[30] R. Cramer and I. Damgård, "Multiparty Computation, an Introduction," *Contemporary Cryptology*, 2005, pp. 41–87.

[31] A. Shamir, "How to Share A Secret," *ACM Commun.*, vol. 22, no. 11, Nov. 1979, pp. 612–13.

[32] M. Burkhart *et al.*, "SEPIA: Privacy-Preserving Aggregation of Multidomain Network Events and Statistics," *Proc. USENIX Security*, 2010.

[33] D. Many, M. Burkhart, and X. Dimitropoulos, "Fast Private Set Operations with SEPIA," ETHZ, tech. rep. 345, Mar. 2012.

[34] The Swiss Education and Research Network (SWITCH); http://www.switch.ch.

## Biographies

MENTARI DJATMIKO (mentari.djatmiko@nicta.com.au) is a Ph.D. candidate at the University of New South Wales, Australia. She is also a member of NICTA's Networks research group. Previously she received her B.E. degree with first class honours in telecommunications engineering from the University of New South Wales in 2009. Her research interests include privacy and applied cryptography.

DOMINIK SCHATZMANN (schadomi@gmail.com) received his Ph.D. in 2012 and his M.S. in electrical engineering and information technology in 2007 from ETH Zürich, Switzerland. His research interests lie in the area of software defined networks, network monitoring, and network security. In 2007 he joined a startup company in Switzerland where he worked in the field of VoIP. Currently, he is working as a network engineer at an innovation laboratory of a national telecommunications operator.

ARIK FRIEDMAN (arik.friedman@nicta.com.au) is a researcher in NICTA's Networks Research Group. He received a Ph.D. from the Technion, Israel Institute of Technology, and an M.B.A. with specialization in technology and information systems from Tel-Aviv University. His research interests include privacy-preserving data mining and computer security. He previously held the position of Program Manager at Microsoft R&D, 2007–2011, where he worked on privacy and security products.

XENOFONTAS DIMITROPOULOS (fontas@tik.ee.ethz.ch) is a senior researcher and lecturer at ETH Zürich. He received his Ph.D. in 2006 from Georgia Tech. His research focuses on network monitoring and software defined networks. He has published more than 50 papers in peer-reviewed journals and conferences, and holds three patents. He has received prestigious grants from the European Research Council, Marie Curie, and Fulbright programs. In the past, he worked at the University of California at San Diego and IBM Research.

ROKSANA BORELI (roksana.boreli@nicta.com.au) is currently a research leader in privacy in NICTA's Networks research group. Prior to joining NICTA in 2004, she held a number of industry positions, including technology strategy manager at Xantic, a joint venture of telcos Telstra and KPN, and mobile satellite services manager at Telstra. She received a Ph.D. from the University of Technology, Sydney, in 1997. Her main research interests include trust and privacy enhancing technologies for Internet and mobile services, and she maintains an interest in transport protocols. She is currently the area editor for Elsevier *Computer Networks*.