AFRL-RI-RS-TR-2018-165

LOS ANGELES/COLORADO RESEARCH EXCHANGE FOR NETWORK DATA

UNIVERSITY OF SOUTHERN CALIFORNIA

JUNE 2018

FINAL TECHNICAL REPORT

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED

STINFO COPY

AIR FORCE RESEARCH LABORATORY
INFORMATION DIRECTORATE

■ **AIR FORCE MATERIEL COMMAND** ■ **UNITED STATES AIR FORCE** ■ **ROME, NY 13441**

NOTICE AND SIGNATURE PAGE

Using Government drawings, specifications, or other data included in this document for any purpose other than Government procurement does not in any way obligate the U.S. Government. The fact that the Government formulated or supplied the drawings, specifications, or other data does not license the holder or any other person or corporation; or convey any rights or permission to manufacture, use, or sell any patented invention that may relate to them.

This report is the result of contracted fundamental research deemed exempt from public affairs security and policy review in accordance with SAF/AQR memorandum dated 10 Dec 08 and AFRL/CA policy clarification memorandum dated 16 Jan 09. This report is available to the general public, including foreign nations. Copies may be obtained from the Defense Technical Information Center (DTIC) (http://www.dtic.mil).

AFRL-RI-RS-TR-2018-165 HAS BEEN REVIEWED AND IS APPROVED FOR PUBLICATION IN ACCORDANCE WITH ASSIGNED DISTRIBUTION STATEMENT.

FOR THE CHIEF ENGINEER:

/ S /
FRANCES A. ROSE
Work Unit Manager

/ S /
JOHN D. MATYJAS
Technical Advisor, Computing
And Communications Division
Information Directorate

This report is published in the interest of scientific and technical information exchange, and its publication does not constitute the Government's approval or disapproval of its ideas or findings.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. REPORT DATE *(DD-MM-YYYY)*	2. REPORT TYPE	3. DATES COVERED *(From - To)*
JUNE 2018	FINAL TECHNICAL REPORT	SEP 2012 – DEC 2017

4. TITLE AND SUBTITLE

LOS ANGELES/COLORADO RESEARCH EXCHANGE FOR NETWORK DATA

5a. CONTRACT NUMBER
N/A

5b. GRANT NUMBER
FA8750-12-2-0344

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)

John Heidemann

5d. PROJECT NUMBER
DHSP

5e. TASK NUMBER
US

5f. WORK UNIT NUMBER
CI

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
University of Southern California / Information Sciences Institute
4676 Admiralty Way, Ste. 1001
Marina del Rey, CA 90292

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

Air Force Research Laboratory/RITE
525 Brooks Road
Rome NY 13441-4505

10. SPONSOR/MONITOR'S ACRONYM(S)
AFRL/RI

11. SPONSOR/MONITOR'S REPORT NUMBER
AFRL-RI-RS-TR-2018-165

12. DISTRIBUTION AVAILABILITY STATEMENT
Approved for Public Release; Distribution Unlimited. This report is the result of contracted fundamental research deemed exempt from public affairs security and policy review in accordance with SAF/AQR memorandum dated 10 Dec 08 and AFRL/CA policy clarification memorandum dated 16 Jan 09

13. SUPPLEMENTARY NOTES

14. ABSTRACT
This final report summarizes the objectives for the LACREND project and the technical progress made against those objectives. The research objective of LACREND project is to (1) develop new validated collection and anonymization methods, (2) curate in-demand datasets, and (3) collect, manage and distribute unique data to enable novel research in Internet security, topology, and reliability. LACREND accomplished those objectives, providing 1497 datasets making up 76.9TB compressed or 324.6TB uncompressed to 223 unique researchers to date. In addition, we published 70 publications and established several new measurement methods, including Trinocular outage detection, BotDigger botnet detection, and Verfploeter anycast mapping**.** Finally, we have distributed 23 software packages and numerous updates.

15. SUBJECT TERMS

datasets, network security research

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			FRANCES A. ROSE
U	U	U	UU	32	19b. TELEPHONE NUMBER (*Include area code*)

TABLE OF CONTENTS

1.0 SUMMARY **1**

2.0 INTRODUCTION **1**

3.0 METHODS, ASSUMPTIONS AND PROCEDURES **1**

3.1 Detailed Description of Public Technical Approach 1

3.2 Comparison with Current Technology 1

3.3 Technology Transition and Technology Transfer Targets and Plans 1

3.4 Data Rights 2

4.0 RESULTS AND DISCUSSION **2**

4.1 Technical Progress Over Contract 2

4.2 Technology Transition and Transfer Over Course of Project 2

4.3 Datasets Created Over the Course of the Project 3

4.7 Software Packages Released Over Contract 8

5.0 KEY RESULTS **9**

5.1 Technical Contributions **9**

5.1.1 Detailed Summaries of Trinocular Outage Detection **9**

5.2.2 Detailed Summaries of BotDigger Botnet Detection **12**

5.2.3 Brief Summaries of Other Technical Contributions **15**

6.0 CONCLUSIONS AND RECOMMENDATIONS **25**

6.1 Conclusions **25**

6.2 Recommendations **25**

LIST OF ABBREVIATIONS AND ACRONYMS **26**

LISTOF FIGURES

Figure 1 Dataset distribution count by quarter 5
Figure 2 Dataset distribution size (compressed) by quarter 6
Figure 3 Dataset distribution size (uncompressed) by quarter 6
Figure 4 Dataset distribution unique number of users per quarter, by quarter 7
Figure 5 Observed outage duration 10
Figure 6 Blocks with largest degree of outages 11
Figure 7 Number of /24 blocks down in each round 12
Figure 8 Botdigger system overview 15

LISTOF TABLES

Table 1 Dataset request and distribution by month…….. 3

1.0 SUMMARY

The LACREND project (1) developed new validated collection and anonymization methods, (2) curated in-demand datasets, and (3) collected, managed and distributed unique data to enable novel research in Internet security, topology, and reliability. This report summarizes the completion of those objectives.

2.0 INTRODUCTION

The LACREND project builds on eight years of LANDER-project experience with PREDICT where we have pioneered new methodologies and provided 7.9 TB of data in 294 datasets to 55 researchers, breaking new ground in each area. The result of these activities was to provide data to improve Internet security and robustness by providing and generating new datasets that are in-demand by the research and operations communities.

The LACREND project (https://ant.isi.edu/lacrend/) participated as a data provider and a data-hosting site under the IMPACT program (https://impactcybertrust.org/). LACREND provided the research community with a rich set of high-quality network data, including traffic traces and network topology information. The ultimate goal of this work was to enable new research and improvements in network security.

3.0 METHODS, ASSUMPTIONS AND PROCEDURES

3.1 Detailed Description of Public Technical Approach

The scope of this effort was to collect Internet datasets related to network traffic (such as anonymized flows and packet headers, DNS [Domain Name System] queries, spam logs) and network topology (such as IPv4 [Internet Protocol, Version 4] address censuses, IP reputation data, network outage data), and develop new methods of measurement and handling sensitive datasets. Our ultimate goal was that these datasets will enable novel research that improves Internet security and reliability.

Our effort consisted of an operational component and a development component. Our objective in the operational component was to successfully and efficiently provide datasets to researchers. In the development component, our milestones were the definition, prototyping, and deployment into production of new data collection methodologies.

3.2 Comparison with Current Technology

Network data is essential to inform research in network security and resilience. Early in research, iteration between data and ideas can help inspire novel solutions. As solutions are developed, data is needed to develop and test alternatives. And when multiple solutions are available, data is essential to separate the best from the good or even poor solutions.

Our project provided novel datasets that enable this work, by developing new data collection methods, collecting new datasets, and developing new methods of handling sensitive data.

3.3 Technology Transition and Technology Transfer Targets and Plans

As a research infrastructure project, our primary technology transition plan was to **deliver data to IMPACT researchers.** We quantify the amount of data provided below.

In addition, as we developed new measurement methods, we sought to **transition those methods** to other researchers through publication of research results and distribution of open source software via the ANT website (https://ant.isi.edu/).

3.4 Data Rights

USC asserts for itself, that the Government's rights to access, use, modify, reproduce, release, perform, display, or disclose only the following technical data or computer software should be restricted as described below.

Software Licensing: The software developed as part of this project was be made available through an open source license, including GPLv2 and BSD-style licenses.

Technical data: Distribution of data generated through this project will be limited as described below. The basis for these restrictions are for protection of the privacy of those about whom the data concerns, compliance with University use of Institutional Review Boards concerning human subjects research (for USC and CSU, where applicable). Restrictions on data are asserted by the organization generating the data (USC, CSU, or otherwise).

All datasets will be released through the IMPACT program and subject to those terms.

We expect packet-related data to require a MOA similar to that at https://ant.isi.edu/datasets/usc_researcher_dua_160816.pdf , and is provided as IMPACT quasi-restricted, allow commercial use, and prohibit redistribution, and forbid deanonymization. These policy methods to protect privacy are necessary to avoid risks of deanonymization possible by combining arbitrary datasets.

Some sensitive data (such as data without IP anonymization) may be restricted, requiring a third party legal agreement with USC or CSU.

These policies are consistent with best current practices in IT data sharing, as developed in the PREDICT and IMPACT programs.

4.0 Results and Discussion

4.1 Technical Progress Over Contract

Collected and provided datasets of anonymized packet, flows, and schemes for DNS query traffic. Adapted stream-segmentation techniques for DNS traffic and developed new methods to support the use of sensitive network datasets. Developed stream-segmentation methods network, and 2253code-to-data auditing methods network datasets.

Collect and provide datasets of IPv4 Censuses, IPv4 Surveys, and IPv4 Hitlists and History. Develop new methods to track IPv4 network outages, visualize IPv4 network outages, identify network scanners and summarize and anonymize NTP attacks.

Provided the Government a capability to understand telephone and Internet infrastructure in the context of disaster scenarios (e.g. hurricanes, earthquakes). Developed new methods to collect data evaluating experimental protocols and implementations of DNS resolution using parallel resolution of DNS traffic. Provided a mechanism for automated analysis of outages, collected by a third party, with granularity as fine as blocks of 256 IP addresses and 11 minute changes in status.

4.2 Technology Transition and Transfer Over Course of Project

As a research infrastructure project, our primary technology transition plan was to **deliver data to IMPACT researchers.** We quantify the amount of data provided below.

In addition, as we developed new measurement methods, we sought to **transition those methods** to other researchers through publication of research results and distribution of open source software via the ANT website (https://ant.isi.edu/). Publications are listed in the next section, and software distributions in the

following section.

4.3 Datasets Created Over the Course of the Project

We added **385 datasets** making up more than **54TB** of compressed data between Sept. 2012 to Dec. 2017. Table 1 shows the breakout of the dataset requests and distribution by month, while Figures 1-4 show dataset distribution count by quarter, compressed and uncompressed and dataset distribution size by quarter, compressed and uncompressed.

Table 2 Dataset request and distribution by month

Month	Datasets			Unique Monthly Users		
	IMPACT	Non-IMPACT	Total	IMPACT	Non-IMPACT	total
2012-01	2	-	2	1	-	1
2012-02	4	-	4	2	-	2
2012-03	9	-	9	1	-	1
2012-04	3	1	4	2	1	3
2012-06	1	-	1	1	-	1
2012-08	-	1	1	-	1	1
2012-09	2	3	5	1	1	2
2012-11	-	2	2	-	1	1
2012-12	4	1	5	3	1	4
2013-01	-	5	5	-	3	3
2013-03	2	1	3	2	1	3
2013-04	3	-	3	1	-	1
2013-05	-	1	1	-	1	1
2013-06	43	35	78	2	5	7
2013-07	2	3	5	1	2	3
2013-09	1	1	2	1	1	2
2013-12	33	-	33	1	-	1
2014-01	-	32	32	-	3	3
2014-02	4	205	209	1	2	3
2014-03	6	23	29	3	6	9
2014-04	65	19	84	3	3	6
2014-05	3	2	5	3	1	4
2014-06	4	11	15	1	2	3
2014-07	2	1	3	2	-	2
2014-08	1	1	2	1	1	2
2014-09	14	-	14	4	-	4
2014-10	70	2	72	4	2	6
2014-11	27	5	32	3	2	5
2014-12	1	13	14	1	4	5
2015-01	6	-	6	1	-	1
2015-02	9	5	14	2	1	3
2015-03	84	-	84	4	-	4
2015-04	142	-	142	6	-	6
2015-05	2	13	15	1	4	5
2015-06	-	1	1	-	1	1
2015-07	18	2	20	4	1	5

Month	Datasets			Unique Monthly Users		
	IMPACT	Non-IMPACT	Total	IMPACT	Non-IMPACT	total
2015-08	2	-	2	1	-	1
2015-09	8	-	8	3	-	3
2015-10	22	-	22	8	-	8
2015-12	2	7	9	1	2	3
2016-01	1	-	1	1	-	1
2016-02	6	2	8	2	1	3
2016-03	4	-	4	4	-	4
2016-04	10	6	16	10	2	12
2016-05	-	16	16	-	3	3
2016-06	-	5	5	-	1	1
2016-08	-	8	8	-	2	2
2016-09	-	1	1	-	1	1
2016-10	2	2	4	2	1	3
2016-12	3	7	10	2	4	6
2017-01	4	1	5	3	1	4
2017-02	3	5	8	2	4	6
2017-03	7	-	7	3	-	3
2017-04	20	4	24	5	2	7
2017-05	21	2	23	6	1	7
2017-06	14	7	21	7	4	11
2017-07	9	-	9	7	-	7
2017-08	3	5	8	2	4	6
2017-09	3	-	3	3	-	3
2017-10	12	4	16	9	3	12
2017-11	13	-	13	7	-	7
2017-12	15	-	15	4	-	4
Total	751	471	1222	155	87	242

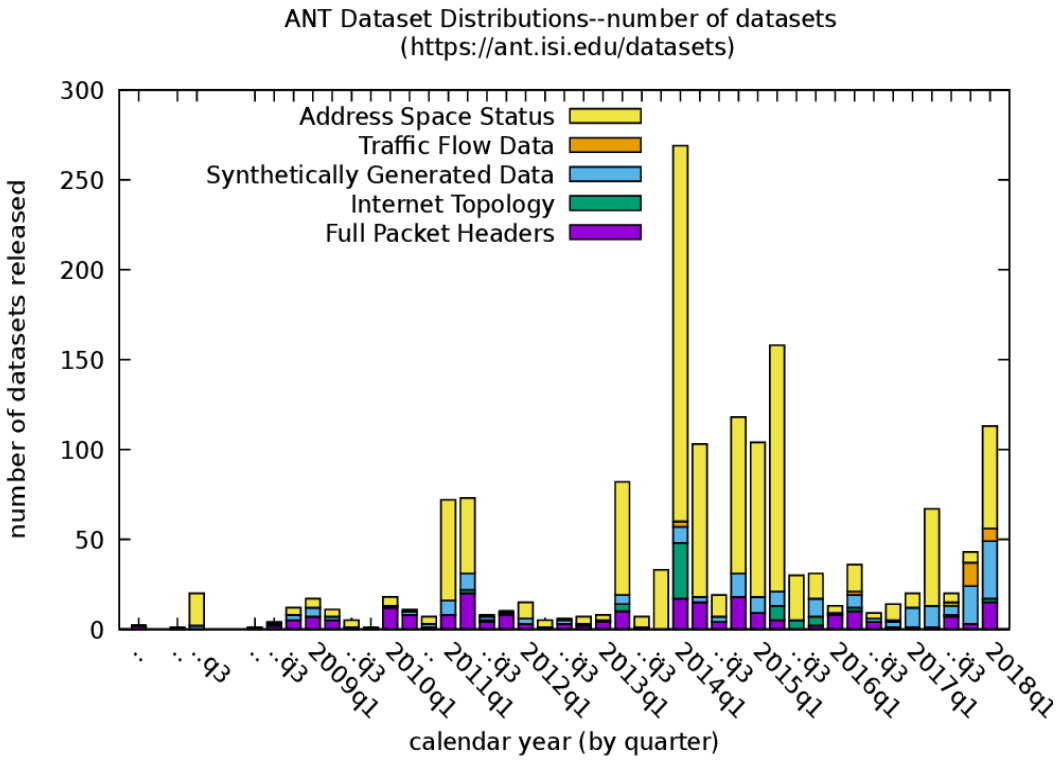

Figure 1 Dataset distribution count by quarter

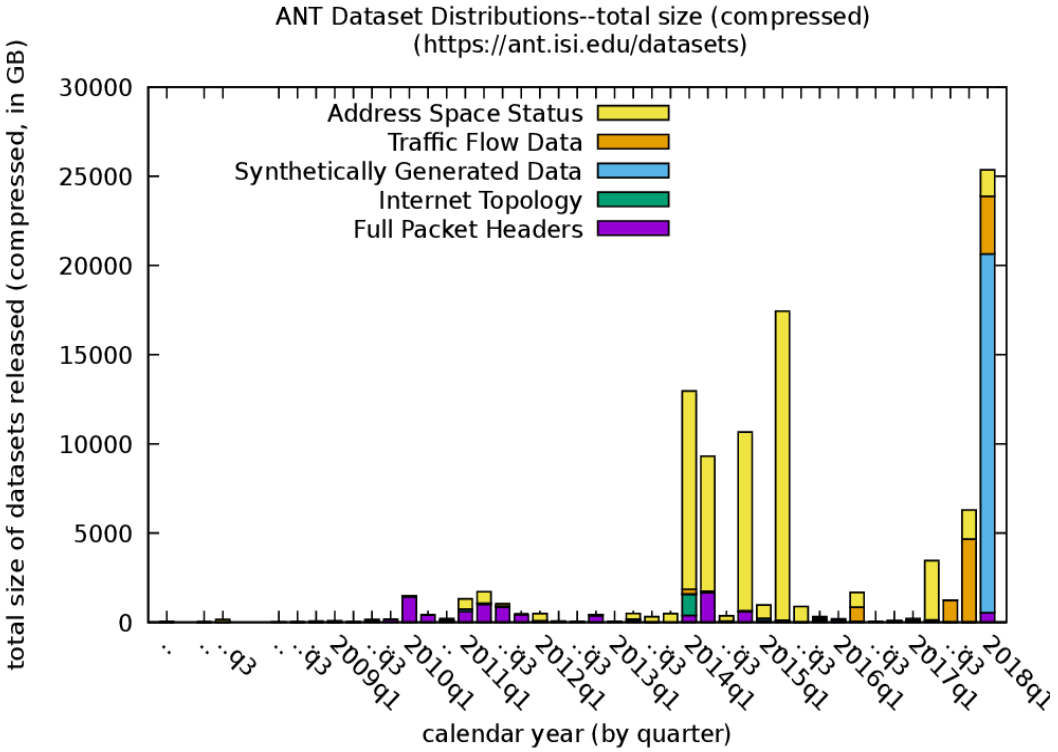

Figure 2 Dataset distribution size (compressed) by quarter

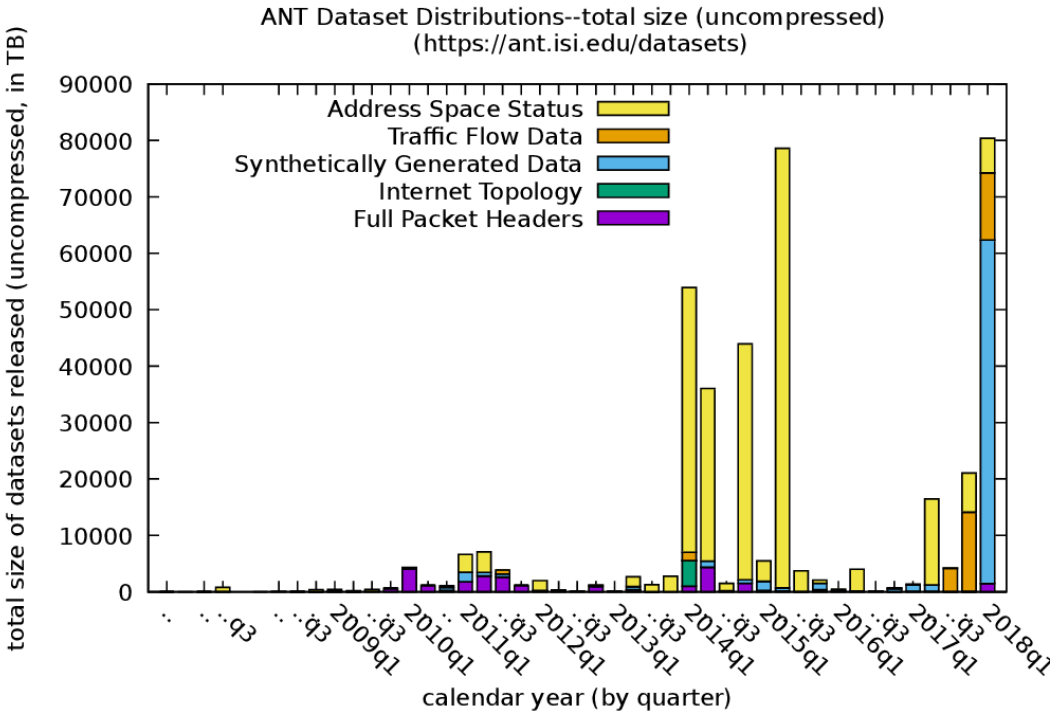

Figure 3 Dataset distribution size (uncompressed) by quarter

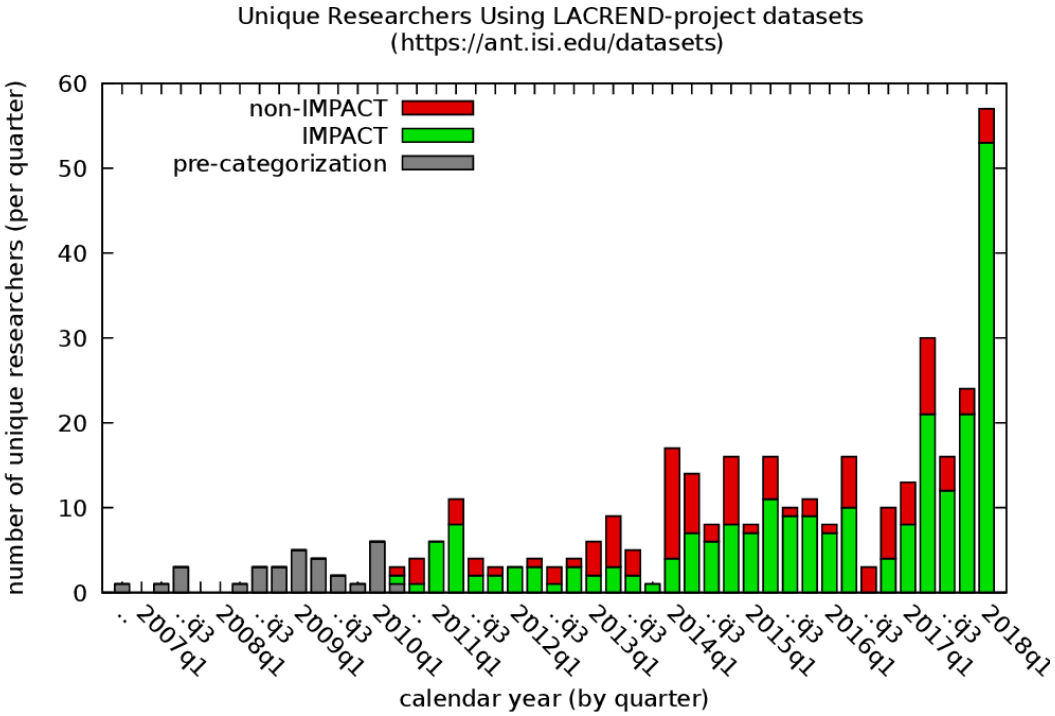

Figure 4 Dataset distribution unique number of users per quarter, by quarter

4.7 Software Packages Released Over Contract

The following unique software packages were released over the duration of the contract. All have open source licenses, usually GPLv2. All are available from https://ant.isi.edu/software/ .

1. **antlink** Manage a tree of git or other VC repositories with funky symlinks
2. **ant_rdns_crawler** The ANT RNDS crawler discovers reverse DNS names for the entire IPv4 space, quickly, politely, and correctly.
3. **AuntieTuna** Chrome browser extension to detect phishing websites
4. **babarchive** Manage babarchives, checksumed directory trees that can be validated
5. **babarchive** Manage babarchives, checksumed directory trees that can be validated
6. **dag scrubber** Dag Scrubber is our tool for scrubbing packets of user data and optionally doing IP address anonymization. It supports both pcap and ERF format ("dag", giving the legacy name).
7. **dag trace generator** The DAG Trace generator is a collection of tools for parsing a DAG formatted packet header trace. (Please see the enclosed README for instructions.)
8. **digit** Digit is a client query tool for T-DNS (DNS with TCP and TLS), designed to measure performance.
9. **dnsanon** extract DNS traffic from pcap to text with optionally anonymization
10. **dnsanon_rssac** Dnsanon_rssac is an implementation of RSSAC-002v2 processing for DNS statistics
11. **dns-replay-client** dns-replay-client reads DNS query stream, replays them against a real DNS server with correct timing and outputs the latency for each query (optional). Multiple dns-replay-client instances can work coordinately to generate aggregated DNS query replay stream, with a separated program: dns-replay-controller.
12. **dns-replay-controller** dns-replay-controller reads DNS query stream and distributes queries to replay clients
13. **icmptrain** Rapid probing of IPv4.
14. **icmptrain-hadoop-reader** A plugin for Hadoop that parses icmptrain output from our ipv4 censuses and surveys.
15. **IP Hitlist Generation** We have developed a set of map/reduce processing scripts that run in Hadoop to consume our Internet address censuses and output hitlists. (This scripts depend on our internal Hadoop configuration and so will require some modification to work elsewhere, but we make them available and encourage feedback about their use.)
16. **LANDER Trace Software** LANDER Trace Capture software handles for packet capture, scrubbing, and triggering user-provided scripts
17. **lonlat2color** For geolocation of IP address maps we needed to convert (lon, lat) to color in HSL and RGB color schemes. We provide Perl and Python implementations.
18. **print_datafile** A command-line tool that prints icmptrain output from our ipv4 censuses and surveys.
19. **stream_merger** Stream merger is a tool to merge multiple traffic streams by feeding them through a FIFO/Drop tail queue and adjusting packet timing due to queueing. Its input is several packet trace files. The output is a single merged packet trace.
20. **mtracecap** A utility for capturing packets concurrently on several network devices and saving output in a single file while making an effort to minimize packet reordering in the output. This tool allows breaking output into multiple files based on size and time and compressing it on the fly by piping to a separate compression process.
21. **tdns-client-proxy** Tdns-client-proxy is a client-side proxy for DNS, designed to run on a

computer taking UDP in and sending it privately with T-DNS to a remote recursive resolver

22. **tdns-server-proxy** Tdns-server-proxy is a server-side proxy for DNS. It listens to incoming private T-DNS (with TCP and TLS) and turns it back into UDP queries to a local DNS resolver
23. **T-DNS support for unbound patch** Unbound patches add STARTTLS handling to incoming unbound queries (but not outgoing T-DNS)

5.0 KEY RESULTS

We next summarize key results from the above work.

5.1 Technical Contributions

The project made significant technical contributions to the development of new data collection systems. Those detailed reports are provided in the publications listed above. We summarize Trinocular, one technical contribution, here, and then summarize specific contributions from a number of publications briefly.

5.1.1 Detailed Summaries of Trinocular Outage Detection

As an example detailed technical contribution we next describe the Trinocular outage detection system.

Introduction and Problem Statement: Although rare, network outages are a serious concern since users depend on connectivity, and operators strive for multiple “nines” of reliability. Replicated services and content delivery networks may conceal outages, but not eliminate them, and the size of the Internet means outages are always occurring somewhere. Outages are triggered by natural disasters, political upheavals, and human error.

Prior work has generally focused on outages from the perspective of routing. Groups today directly monitor routing, track routable prefixes with control- and data-plane methods, and study traffic to unoccupied addresses.

While these approaches are useful to detect and sometimes mitigate large outages related to routing, most of the Internet uses default routing, and we show that most outages are smaller than routable prefixes. While some systems target probing to detect specific kinds of smaller outages, to our knowledge, no service today actively tracks outages in all Internet edge networks.

The contribution of our work on Trinocular was to address this gap, providing unbiased, accurate measurements of Internet reliability to all analyzable edge networks. First, we described Trinocular, an adaptive probing system to detect outages in edge networks. Our system was principled, deriving a simple model of the Internet that captured the information pertinent to outages, parameterizing the model with long-term observations, and learning current network state with probing driven by Bayesian inference.

Second, using experiments, analysis, and simulation, we validated that these principles result in a system that was predictable and precise: we detected 100% of outages longer than our periodic probing interval, with known precision in timing and duration. It was also parsimonious, requiring minimal probing traffic. On average, each Trinocular instance increased traffic to covered networks by no more than 0.7% of the Internet’s “background radiation”. This low rate allowed a single computer to monitor the entire analyzable Internet, and multiple concurrent instances to identify outage scope.

Finally, we used Trinocular to observe two days of Internet outages from three sites. We also

adapted our model to re-analyze existing data, developing three years of trends from measurements of samples of the Internet. This reanalysis included observations of outages due to Hurricane Sandy in 2012, the Japanese Earthquake in March 2012, and the Egyptian Revolution in January 2012.

The full technical description of Trinocular is in: Lin Quan, John Heidemann and Yuri Pradkin 2013. Trinocular: Understanding Internet Reliability Through Adaptive Probing. Proceedings of the ACM SIGCOMM Conference (Hong Kong, China, Aug. 2013), 255–266. We next summarize three key results.

Key Result: Trinocular is Correct. We first explored the correctness of our approach: if an outage occurs, do we always see it? For a controlled evaluation of this question, we ran Trinocular and probed 4 /24 blocks at our university from 3 sites: our site in Los Angeles, and universities 1600 km and 8800 km distant in Colorado and Japan. We controlled these blocks and configured them in two-hour cycle where the network is up for 30 minutes, goes down at some random time in the next 20 minutes, stays down for a random duration between 0 and 40 minutes, then comes back up. This cycle guaranteed Trinocular would reset between controlled outages. We studied these blocks for 122 cycles, yielding 488 observations as dataset A controlled, combining data for 4 controlled blocks from datasets A1w (2013-01-19, 4 days), A3w (2013- 01-24, 1 day), A4w (2013-01-25, 2 days), and A7w (2013-02-12, 2 days).

Figure 5 shows these experiments, with colored areas showing observed outage duration rounded to integer numbers of rounds. We grouped true outage duration on the x into rounds with dotted black lines. Since periodic probing guarantees we tested each network every round, we expected to find all outages that lasted at least one round or longer. We also saw that we missed outages shorter than a round roughly in proportion to outage duration (the white region of durations less than 11 minutes). While these experiments were specific to blocks where addresses always responded (A(E(b)) = 1), they generalized to blocks with $A \geq 0.3$ since we later showed that we took enough probes to reach a definitive conclusion for these blocks. These results confirm what we expected based on our sampling schedule: if we probed a block with $A \geq 0.3$, we always detected outages longer than one round.

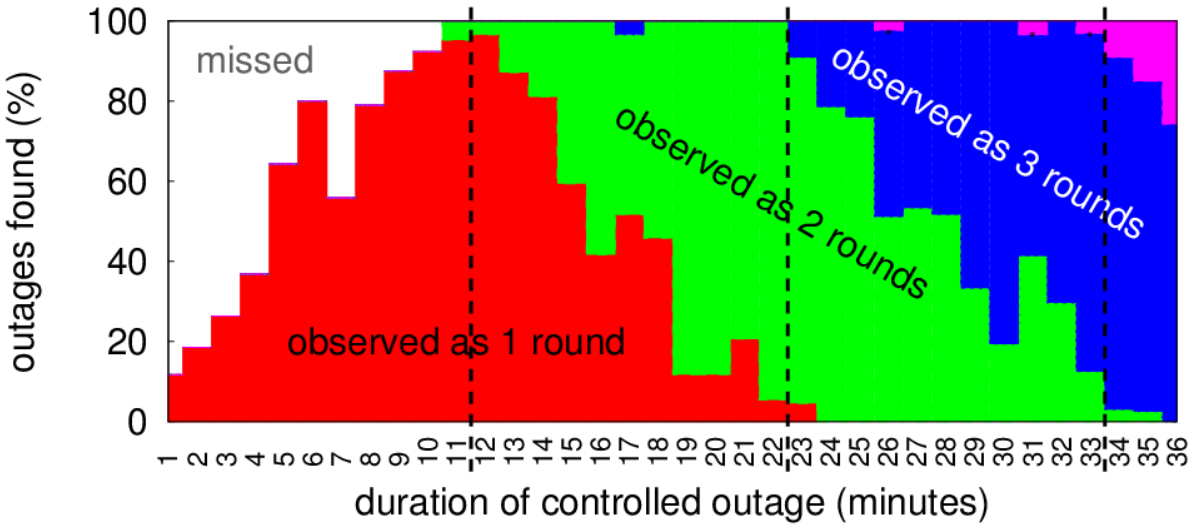

Figure 5 Observed outage duration

Key Result: Re-analysis of data with Trinocular to Evaluate Hurricane Sandy. We observed a noticeable increase in network outages following Hurricane Sandy. The Hurricane made landfall in the U.S. at about 2012-10-30 T00:00 UTC. When we focused on known U.S. networks, we saw about triple the number of network outages for the day following landfall, and above-baseline outages for the four days following landfall.

Visualizing outages: The Figure 6 visualizes the 400 blocks in the U.S. with the largest degree of outages, and label (a) shows a strong cluster of outages at 2013-10-30 (UTC) corresponding with hurricane landfall. Hurricane-related outages tend to be long, lasting one or more days. We believe these outages correspond to residential power outages.

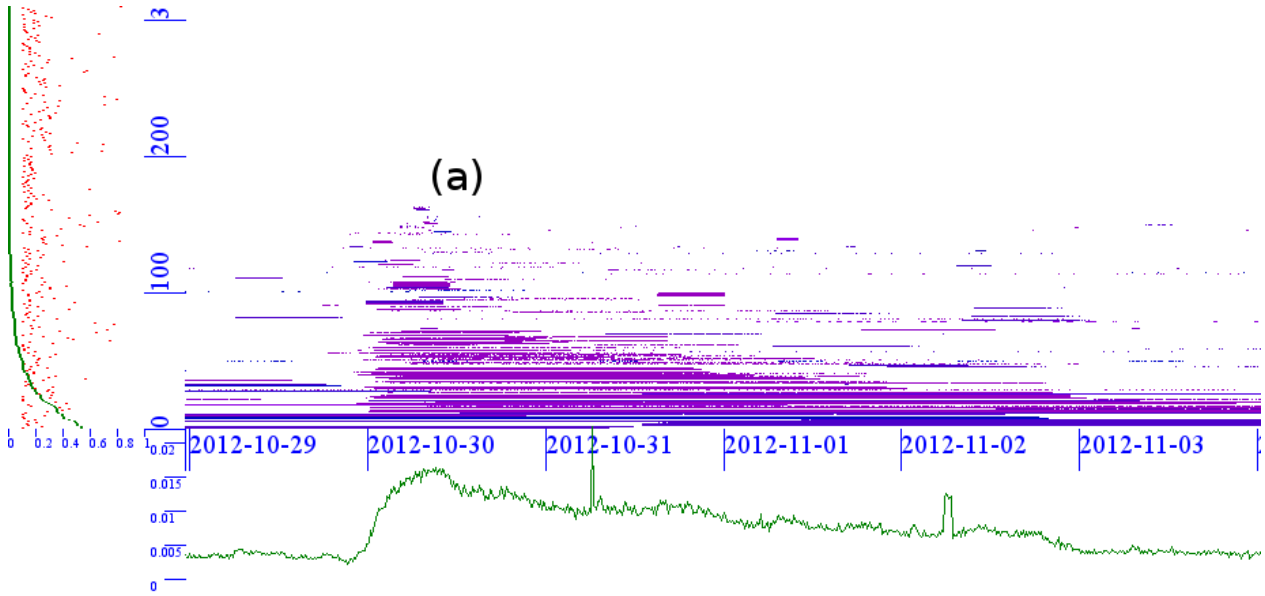

Figure 6 Blocks with largest degree of outages

Quantifying outages: We know that some part of the Internet is always down, so to place these outages in perspective, the Figure 7 plots the exact number of /24 blocks that are down in each round (this value is the marginal distribution of the above figure). We plot each round as small red points (with small jitter to make consecutive more distinct), and we show 24-hour median values with the dark line.

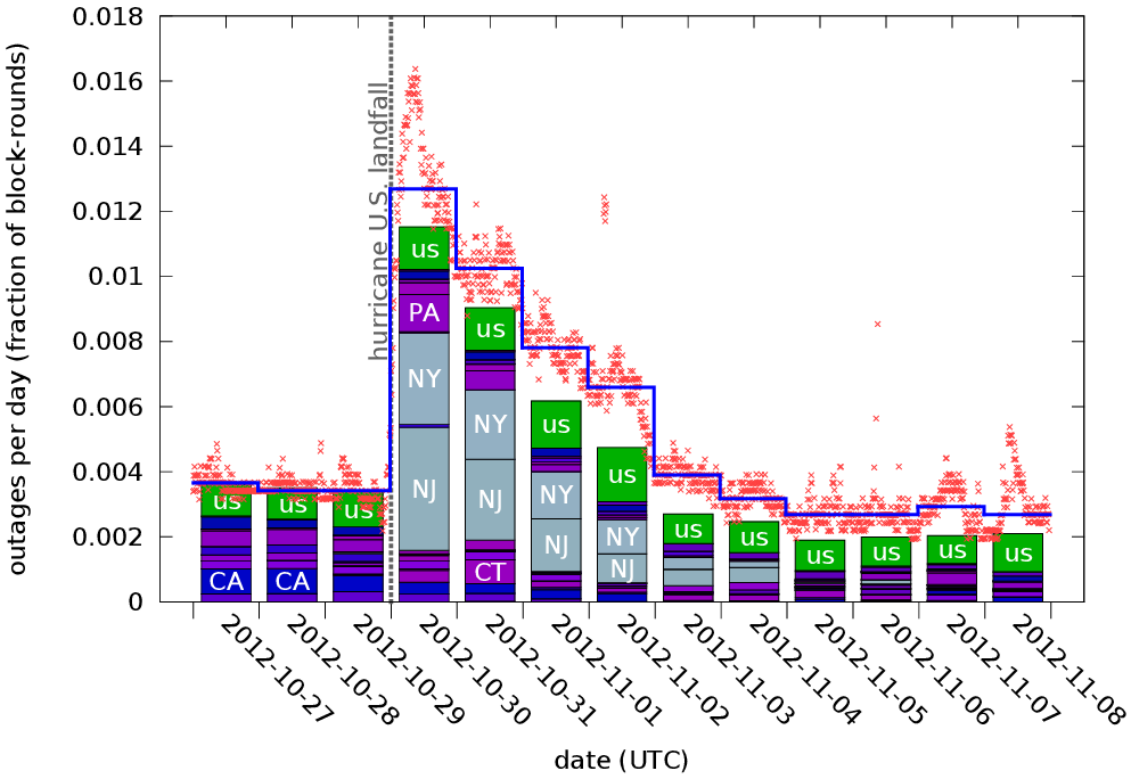

Figure 7 Number of /24 blocks down in each round

This figure shows U.S. networks had an outage rate of about 0.36% before landfall. (This rate seems somewhat less than the global average.) This rate jumps to 1.27%, about triple the prior U.S. baseline, for the 24-hours following landfall. The outage level drops over the next four days, finally returning to the baseline on 2012-11-03.

Locating outages: To confirm the correlation between the hurricane and these outages, we looked at the weighted blocks by state. The bars in this figure identify outages by state. The top "US" portion represents outages that are geolocated in the U.S., but not to a specific state. This figure shows that there are large increases in the amount of outages in New York and New Jersey (the lighter colored bars in the middle of the graph) after hurricane landfall on 2012-10-30, about three times the prior baseline. These problems are generally resolved over the following four days. (Because of our more sensitive methodology, we see more outages here than in our prior analysis [14], but our qualitative results are similar.)

Current Status: Trinocular has been deployed and operational since Oct. 2014, and we have engaged in a pilot project working with the FCC to examine Trinocular use in FCC reporting of telephony outages. Trinocular data has also been of interest to third parties. As of Dec. 2017, we have provided 59 **Trinocular datasets** to **13 different researchers**. We expect to continue developing Trinocular beyond the LACREND project.

5.2.2 Detailed Summaries of BotDigger Botnet Detection

As a second example detailed technical contribution we next describe the BotDigger botnet detection system.

Introduction and Problem Statement: Cyber security constitutes one of the most serious threats to the current society, costing billions of dollars each year. Botnets is a very important way to perform

many attacks. In botnets, the botmaster and bots exchange information through C&C channels, which can be implemented using many protocols, such as IRC, HTTP, Overnet. Although using P2P protocols as C&C channels is getting popular, HTTP-based botnets are still very common as they are easy to implement and maintain. In a HTTP-based botnet, the botmaster publishes the commands under a domain, then the bots query the domain to fetch the contents periodically. In the early years, the domain was hard-coded in binary, introducing a single point of failure. To become more robust, botnets began to generate C&C domains dynamically on the fly using DGA. In particular, hundreds or thousands of domains can be algorithmically generated every day, but the botmaster only registers one or a few of them as C&C domains and publishes the commands there. DGA technique evades static blacklists, avoids single point of failure, and also prevents security specialists from registering the C&C domain before the botmaster.

There are four reasons to detect DGA botnets using DNS traffic. First, the DGA bots have to send DNS queries to look up the IP addresses of C&C domains. Second, the amount of DNS traffic is much less than the overall traffic. Focusing on a relatively small amount of traffic helps to improve performance, making it possible to detect bots in real time. Third, the DNS traffic of DGA bots has different patterns compared to legitimate hosts. For example, DGA bots send more DNS queries than legitimate hosts. Last, if we can detect bots only using DNS traffic when they look for C&C domains, we can stop the attacks even before they happen.

Many previous works have been introduced to detect DGA-based botnets and malicious domains (e.g., C&C domains, phishing domains) using DNS traffic. They share some common assumptions, such as DGA domains generated by the same algorithm have similar linguistic attributes, DGA domains' attributes are different from legitimate ones, and so forth. Based on these assumptions, classification and/or clustering algorithms are applied for detection. However, many of these past works require multiple hosts infected by the same type of botnet existing in the collected traces. Consequently, they have to collect DNS traffic at an upper level (e.g., TLD servers, authoritative servers, etc.), or from multiple RDNS servers among networks. The advantage of these works is that evidences from multiple bots can be collected and analyzed. However, they also introduce several challenges. First, the DNS traffic at an upper level is hard to access for most of enterprise and/or university network operators. Second, sharing DNS traffic among networks may introduce privacy issues. Third, it is computationally expensive to run clustering/classification algorithms on large traces collected from multiple networks. Finally, the most significant challenge is that an enterprise network may not have multiple bots, especially bots infected by the same botnet. For example, Pleiades detects less than 100 bots of the same botnet in a large North American ISP that includes over 2 million clients hosts per day. As a comparison, our university network has around 20,000 clients, which is only 1% of the ISP, meaning that on average only 1 host infected by the same botnet exists in the network.

As part of LACREND, we developed BotDigger, a system that detects an individual bot by only using DNS traffic collected from a single network. This single network can be a company network, a university network, or a local area network (LAN). Notice that "detecting individual bot in a network" does not mean BotDigger cannot detect all the bots in a network. If there are multiple bots in the same network, BotDigger can still detect them, but individually. BotDigger uses a chain of evidences, including quantity evidence, linguistic evidence, and temporal evidence to detect bots. In particular, quantity evidence means that the number of suspicious second level domains (2LDs) queried by bots are much more than the legitimate hosts. Two temporal evidences are used: 1) the number of suspicious 2LDs queried by a bot suddenly increases when it starts to look for the registered C&C domain, 2) once the bot hits the registered C&C domain, the

number of queried suspicious 2LDs will decrease. The basis of linguistic evidence is that the DGA NXDomains and C&C domains queried by a bot are generated by the same algorithm, thus they share similar linguistic attributes. We apply the above evidences sequentially to detect bots and the queried DGA NXDomains. After that, we extract signatures from the DGA NXDomains and apply them on the successfully resolved domains to extract the corresponding C&C domain candidates.

The contributions of BotDigger as follows:

1. It defines a chain of evidences, including quantity evidence, temporal evidence and linguistic evidence, to detect DGA-based botnets.
2. We introduced and implemented BotDigger, a system that detects an individual DGA-based bot using DNS traffic collected in a single network.
3. We evaluate BotDigger with two datasets from Colorado State University and the NetSec lab, as well as two DGA-based botnets.
4. The results show that BotDigger detects more than 99.8% of the bots with less than 0.5% false positives.

An overview of the methodology is shown in Figure 8. First, several filters are applied to remove unsuspicious NXDomains (e.g., the domains with invalid TLDs). The remaining suspicious NXDomains are then grouped by host who sends the queries. Notice that in the following steps, we focus on the queried domains from each individual host, and that is the reason why our method can detect individual bot. Now the quantity evidence is applied to extract outliers in terms of the number of queried suspicious 2LD NXDomainss. For each outlier, we use the temporal evidence to extract the period of time when a bot begins to query DGA domains until it hits the registered C&C domain, denoted as (t begin, t end). If such period cannot be extracted, then the host is considered as legitimate, otherwise the host is considered suspicious and the following analysis is performed.

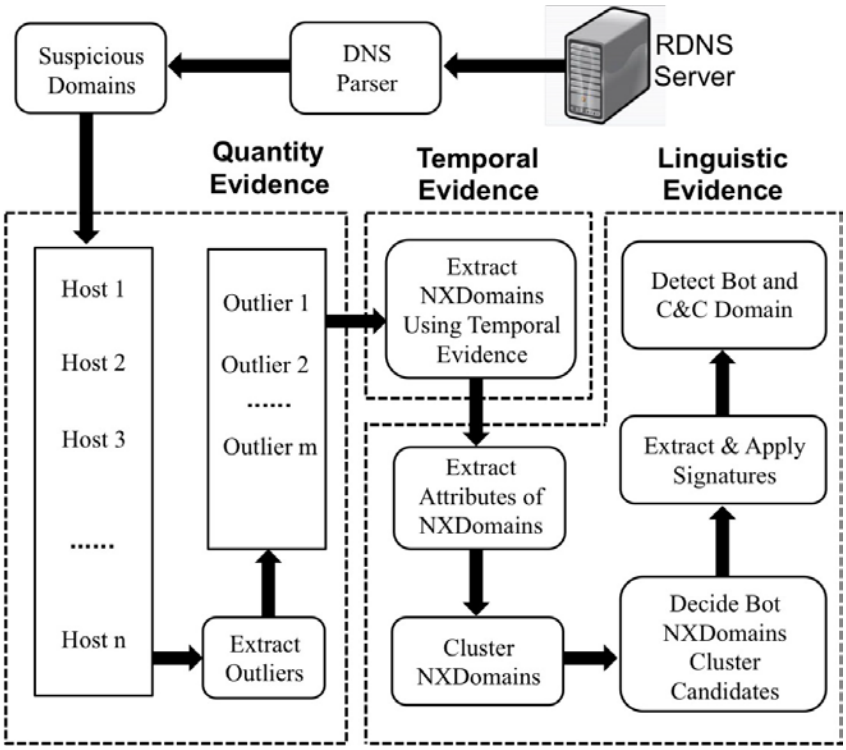

Figure 8 Botdigger system overview

The next step focuses on the suspicious NXDomains being queried between t begin and t end. The linguistic attributes of these NXDomains are extracted and then the linguistic evidence is applied on the extracted attributes. The assumption of linguistic evidence is that a bot queries many suspicious NXDomains that have similar linguistic attributes thus they will likely be clustered together. In particular, a hierarchical clustering algorithm is applied on the attributes. The output is one or more clusters of linguistic attributes and the corresponding suspicious NXDomains. We consider the clusters whose sizes are greater than a threshold (named BotClusterThreshold as defined our paper) as bot NXDomain cluster candidates. If all the clusters of a host are smaller than the threshold, then the host is considered legitimate.

Finally, to identify the C&C domains, the DGA domain signatures are extracted from the bot NXDomain cluster candidates and matched against all the successfully resolved domains queried between t begin and t end. The domains that match the signatures are considered as C&C domain candidates, and the host is labeled as a bot. The host is not labeled if no C&C domain candidate can be extracted. Note that we do not precisely label C&C domain. Instead, we label multiple C&C domain candidates. It is possible, however, unlikely, that a successful request is done by the infected host right after a series of failed requests, and the legitimate domain in the request is close in lexicographic distance. For this reason, we can never be absolutely certain that a successful DNS request is a C&C server. Precisely labeling single C&C domain is future work.

BotDigger System Additional Information: Full details about botdigger are in the publication: Han Zhang, Manaf Gharaibeh, Spiros Thanasoulas and Christos Papadopoulos 2016. BotDigger: Detecting DGA Bots in a Single Network. Proceedings of the IEEE International Workshop on Traffic Monitoring and Analyses (Louvain La Neuve, Belgium, Apr. 2016), 16–21.

5.2.3 Brief Summaries of Other Technical Contributions

We cannot reproduce all that material here and refer to those publications for details, but we

summarize key results from selected papers.

1. Wouter B. de Vries, Ricardo de O. Schmidt, Wes Hardaker, John Heidemann, Pieter-Tjerk de Boer and Aiko Pras 2017. Verfploeter: Broad and Load-Aware Anycast Mapping. Proceedings of the ACM Internet Measurement Conference (London, UK, 2017), 477–488.

IP anycast provides DNS operators and CDNs with automatic fail-over and reduced latency by breaking the Internet into *catchments*, each served by a different anycast site. Unfortunately, *understanding* and *predicting* changes to catchments as anycast sites are added or removed has been challenging. Current tools such as RIPE Atlas or commercial equivalents map from thousands of vantage points (VPs), but their coverage can be inconsistent around the globe. This paper proposes *Verfploeter*, a new method that maps anycast catchments using active probing. Verfploeter provides around 3.8M passive VPs, 430x the 9k physical VPs in RIPE Atlas, providing coverage of the vast majority of networks around the globe. We then add load information from prior service logs to provide calibrated predictions of anycast changes. Verfploeter has been used to evaluate the new anycast deployment for B-Root, and we also report its use of a nine-site anycast testbed. We show that the greater coverage made possible by Verfploeter's active probing is necessary to see routing differences in regions that have sparse coverage from RIPE Atlas, like South America and China.

2. Moritz Müller, Giovane C. M. Moura, Ricardo de O. Schmidt and John Heidemann 2017. Recursives in the Wild: Engineering Authoritative DNS Servers. Proceedings of the ACM Internet Measurement Conference (London, UK, 2017), 489–495.

In Internet Domain Name System (DNS), services operate authoritative name servers that individuals query through recursive resolvers. Operators strive to provide reliability by operating multiple name servers (NS), each on a separate IP address, and by using IP anycast to allow NSes to provide service from many physical locations. To meet their goals of minimizing latency and balancing load across NSes and anycast, operators need to know how recursive resolvers select an NS, and how that interacts with their NS deployments. Prior work has shown some recursives search for low latency, while others pick an NS at random or round robin, but did not examine how prevalent each choice was. This paper provides the first analysis of how recursives select between name servers in the wild, and from that we provide guidance to operators how to engineer their name servers to reach their goals. We conclude that all NSes need to be equally strong and therefore we recommend to deploy IP anycast at every single authoritative.

3. Kensuke Fukuda, John Heidemann and Abdul Qadeer 2017. Detecting Malicious Activity with DNS Backscatter Over Time. ACM/IEEE Transactions on Networking. 25, 5 (Aug. 2017), 3203– 3218.

Network-wide activity is when one computer (the originator) touches many others (the targets). Motives for activity may be benign (mailing lists, CDNs, and research scanning), malicious (spammers and scanners for security vulnerabilities), or perhaps indeterminate (ad trackers). Knowledge of malicious activity may help anticipate attacks, and understanding benign activity may set a baseline or characterize growth. This paper identifies DNS backscatter as a new source of information about network-wide activity. Backscatter is the reverse DNS queries caused when targets or middleboxes automatically look up the domain name of the originator. Queries are visible to the authoritative DNS servers that handle reverse DNS. While the fraction of backscatter they see depends on the server's location in the DNS hierarchy, we show that activity that touches many targets appear even in sampled observations. We use information about the queriers to classify originator activity using machine-learning. Our algorithm has reasonable

accuracy and precision (70–80%) as shown by data from three different organizations operating DNS servers at the root or country-level. Using this technique, we examine nine months of activity from one authority to identify trends in scanning, identifying bursts corresponding to Heartbleed and broad and continuous scanning of ssh.

4. Lan Wei and John Heidemann 2017. Does Anycast Hang up on You? IEEE International Workshop on Traffic Monitoring and Analysis (Dublin, Ireland, Jul. 2017).

Anycast-based services today are widely used commercially, with several major providers serving thousands of important websites. However, to our knowledge, there has been only limited study of how often anycast fails because routing changes interrupt connections between users and their current anycast site. While the commercial success of anycast CDNs means anycast usually work well, do some users end up shut out of anycast? In this paper we examine data from more than 9000 geographically distributed vantage points (VPs) to 11 anycast services to evaluate this question. Our contribution is the analysis of this data to provide the first quantification of this problem, and to explore where and why it occurs. We see that about 1% of VPs are anycast unstable, reaching a different anycast site frequently (sometimes every query). Flips back and forth between two sites in 10 seconds are observed in selected experiments for given service and VPs. Moreover, we show that anycast instability is persistent for some VPs— a few VPs never see a stable connection to certain anycast services during a week or even longer. The vast majority of VPs only saw unstable routing towards one or two services instead of instability with all services, suggesting the cause of the instability lies somewhere in the path to the anycast sites. Finally, we point out that for highly-unstable VPs, their probability to hit a given site is constant, which means the flipping are happening at a fine granularity—per packet level, suggesting load balancing might be the cause to anycast routing flipping. Our findings confirm the common wisdom that anycast almost always works well, but provide evidence that a small number of locations in the Internet where specific anycast services are never stable.

5. Jelena Mirkovic, Genevieve Bartlett, John Heidemann, Hao Shi and Xiyue Deng 2017. Do You See Me Now? Sparsity in Passive Observations of Address Liveness. IEEE International Workshop on Traffic Monitoring and Analysis (Dublin, Ireland, Jul. 2017), 1–9.

Accurate information about address and block usage in the Internet has many applications in planning address allocation, topology studies, and simulations. Prior studies used active probing, sometimes augmented with passive observation, to study macroscopic phenomena, such as the overall usage of the IPv4 address space. This paper instead studies the completeness of passive sources: how well they can observe microscopic phenomena such as address usage within a given network. We define textitsparsity as the limitation of a given monitor to see a target, and we quantify the effects of interest, temporal, and coverage sparsity. To study sparsity, we introduce inverted analysis, a novel approach that uses complete passive observations of a few end networks (three campus networks in our case) to infer what of these networks would be seen by millions of virtual monitors near their traffic's destinations. Unsurprisingly, we find that monitors near popular content see many more targets and that visibility is strongly influenced by bipartite traffic between clients and servers. We are the first to quantify these effects and show their implications for the study of Internet liveness from passive observations. We find that visibility is heavy-tailed, with only 0.5% monitors seeing more than 10% of our targets' addresses, and is most affected by interest sparsity over temporal and coverage sparsity. Visibility is also stronglybipartite. Monitors of a different class than a target (e.g., a server monitor observing a client target) outperform monitors of the same class as a target in 82-99% of cases in our datasets. Finally, we find that

adding active probing to passive observations greatly improves visibility of both server and client target addresses, but is not critical for visibility of target blocks. Our findings are valuable to understand limitations of existing measurement studies, and to develop methods to maximize microscopic completeness in future studies.

6. Ricardo de O. Schmidt, John Heidemann and Jan Harm Kuipers 2017. Anycast Latency: How Many Sites Are Enough? Proceedings of the Passive and Active Measurement Workshop (Sydney, Australia, Mar. 2017), to appear.

Anycast is widely used today to provide important services such as DNS and Content Delivery Networks (CDNs). An anycast service uses multiple sites to provide high availability, capacity and redundancy. BGP routing associates users to sites, defining the catchment that each site serves. Although prior work has studied how users associate with anycast services informally, in this paper we examine the key question how many anycast sites are needed to provide good latency, and the worst case latencies that specific deployments see. To answer this question, we first define the optimal performance that is possible, then explore how routing, specific anycast policies, and site location affect performance. We develop a new method capable of determining optimal performance and use it to study four real-world anycast services operated by different organizations: C-, F-, K-, and L-Root, each part of the Root DNS service. We measure their performance from more than 7,900 vantage points (VPs) worldwide using RIPE Atlas. (Given the VPs uneven geographic distribution, we evaluate and control for potential bias.) Our key results show that a few sites can provide performance nearly as good as many, and that geographic location and good connectivity have a far stronger effect on latency than having many sites. We show how often users see the closest anycast site, and how strongly routing policy affects site selection.

7. John Heidemann 2017. DNS Privacy, Service Management, and Research: Friends or Foes. Talk at ISOC NDSS Workshop on DNS Privacy.

(no formal abstract) This invited talk is part of a panel on the tension between DNS privacy and service management. In the talk I expand on that topic and discuss the tension between DNS privacy, service management, and research. I give suggestions about how service management and research can adapt to proceed while still providing basic privacy.

8. Anant Shah, Romain Fontugne and Christos Papadopoulos 2016. Towards Characterizing International Routing Detours. Proceedings of the 12th Asian Internet Engineering Conference (AINTEC) (Bangkok, Thailand, Nov. 2016), to appear.

There are currently no requirements (technical or otherwise) that routing paths must be contained within national boundaries. Indeed, some paths experience international detours, i.e., originate in one country, cross international boundaries and return to the same country. In most cases these are sensible traffic engineering or peering decisions at ISPs that serve multiple countries. In some cases, such detours may be suspicious. Characterizing international detours is useful to a number of players: (a) network engineers trying to diagnose persistent problems, (b) policy makers aiming at adhering to certain national communication policies, (c) entrepreneurs looking for opportunities to deploy new networks, or (d) privacy-conscious states trying to minimize the amount of internal communication traversing different jurisdictions. In this paper we characterize international detours in the Internet during the month of January 2016. To detect detours, we sample BGP RIBs every 8 hours from 461 RouteViews and RIPE RIS peers spanning 30 countries. We use geolocation of ASes which geolocates each BGP prefix announced by each AS, mapping its presence at IXPs and geolocation infrastructure IPs. Finally, we analyze each global BGP RIB

entry looking for detours. Our analysis shows more than 5K unique BGP prefixes experienced a detour. 132 prefixes experienced more than 50% of the detours. We observe about 544K detours. Detours either last for a few days or persist the entire month. Out of all the detours, more than 90% were transient detours that lasted for 72 hours or less. We also show different countries experience different characteristics of detours.

9. Giovane C. M. Moura, Ricardo de O. Schmidt, John Heidemann, Wouter B. de Vries, Moritz Müller, Lan Wei and Christian Hesselman 2016. Anycast vs. DDoS: Evaluating the November 2015 Root DNS Event. Proceedings of the ACM Internet Measurement Conference (Nov. 2016).

Distributed Denial-of-Service (DDoS) attacks continue to be a major threat on the Internet today. DDoS attacks overwhelm target services with requests or other traffic, causing requests from legitimate users to be shut out. A common defense against DDoS is to replicate a service in multiple physical locations/sites. If all sites announce a common prefix, BGP will associate users around the Internet with a nearby site, defining the catchment of that site. Anycast defends against DDoS both by increasing aggregate capacity across many sites, and allowing each site's catchment to contain attack traffic, leaving other sites unaffected. IP anycast is widely used by commercial CDNs and for essential infrastructure such as DNS, but there is little evaluation of anycast under stress. This paper provides the first evaluation of several IP anycast services under stress with public data. Our subject is the Internet's Root Domain Name Service, made up of 13 independently designed services ("letters", 11 with IP anycast) running at more than 500 sites. Many of these services were stressed by sustained traffic at 100\times normal load on Nov. 30 and Dec. 1, 2015. We use public data for most of our analysis to examine how different services respond to stress, and identify two policies: sites may absorb attack traffic, containing the damage but reducing service to some users, or they may withdraw routes to shift both good and bad traffic to other sites. We study how these deployment policies resulted in different levels of service to different users during the events. We also show evidence of collateral damage on other services located near the attacks.

10. Z. Hu, L. Zhu, J. Heidemann, A. Mankin, D. Wessels and P. Hoffman 2016. Specification for DNS over Transport Layer Security (TLS). Technical Report 7858. Internet Request For Comments.

This document describes the use of Transport Layer Security (TLS) to provide privacy for DNS. Encryption provided by TLS eliminates opportunities for eavesdropping and on-path tampering with DNS queries in the network, such as discussed in RFC 7626. In addition, this document specifies two usage profiles for DNS over TLS and provides advice on performance considerations to minimize overhead from using TCP and TLS with DNS. This document focuses on securing stub-to- recursive traffic, as per the charter of the DPRIVE Working Group. It does not prevent future applications of the protocol to recursive-to- authoritative traffic. This document is a product of the DNS PRIVate Exchange Working Group of the IETF. This is now a Proposed Standard.

11. Manaf Gharaibeh, Han Zhang, Christos Papadopoulos and John Heidemann 2016. Assessing Co- Locality of IP Blocks. Proceedings of the 19th IEEE Global Internet Symposium (San Francisco, CA, USA, Apr. 2016).

Many IP Geolocation services and applications assume that all IP addresses with the same /24 IPv4 prefix (a /24 block) are in the same location. For blocks that contain addresses in very different locations (such blocks identifying network backbones), this assumption can result in

large geolocation error. This paper evaluates this assumption using a large dataset of 1.41M /24 blocks extracted from a delay measurements dataset for the entire responsive IPv4 address space. We use hierarchal clustering to find clusters of IP addresses with similar observed delay measurements within /24 blocks. Blocks with multiple clusters often span different geographic locations. We evaluate this claim against two ground-truth datasets, confirming that 93% of identified multi-cluster blocks are true positives with multiple locations, while only 13% of blocks identified as single-cluster appear to be multi-location in ground truth. Applying the clustering process to the whole dataset suggests that about 17% (247K) of blocks are likely multi-location.

12. Han Zhang, Manaf Gharaibeh, Spiros Thanasoulas and Christos Papadopoulos 2016. BotDigger: Detecting DGA Bots in a Single Network. Proceedings of the IEEE International Workshop on Traffic Monitoring and Analysis (Louvain La Neuve, Belgium, Apr. 2016), 16–21

To improve the resiliency of communication between bots and C&C servers, bot masters began utilizing Domain Generation Algorithms (DGA) in recent years. Many systems have been introduced to detect DGA-based botnets. How- ever, they su↵er from several limitations, such as requiring DNS traffic collected across many networks, the presence of multiple bots from the same botnet, and so forth. These limitations make it very hard to detect individual bots when using traffic collected from a single network. In this paper, we introduce BotDig- ger, a system that detects DGA-based bots using DNS traffic without a priori knowledge of the domain generation algorithm. BotDigger utilizes a chain of evidence, including quantity, temporal and linguistic evidence to detect an individual bot by only monitoring traffic at the DNS servers of a single network. We evaluate BotDigger's performance using traces from two DGA-based botnets: Kraken and Conflicker. Our results show that BotDigger detects all the Kraken bots and 99.8% of Conficker bots. A one-week DNS trace captured from our university and three traces collected from our research lab are used to evaluate false positives. The results show that the false positive rates are 0.05% and 0.39% for these two groups of background traces, respectively.

13. Kensuke Fukuda and John Heidemann 2015. Detecting Malicious Activity with DNS Backscatter. Proceedings of the ACM Internet Measurement Conference (Tokyo, Japan, Oct. 2015), 197–210.

Network-wide activity is when one computer (the originator) touches many others (the targets). Motives for activity may be benign (mailing lists, CDNs, and research scanning), malicious (spammers and scanners for security vulnerabilities), or perhaps indeterminate (ad trackers). Knowledge of malicious activity may help anticipate attacks, and understanding benign activity may set a baseline or characterize growth. This paper identifies DNS backscatter as a new source of information about network-wide activity. Backscatter is the reverse DNS queries caused when targets or middleboxes automatically look up the domain name of the originator. Queries are visible to the authoritative DNS servers that handle reverse DNS. While the fraction of backscatter they see depends on the server's location in the DNS hierarchy, we show that activity that touches many targets appear even in sampled observations. We use information about the quarriers to classify originator activity using machine-learning. Our algorithm has reasonable precision (70–80%) as shown by data from three different organizations operating DNS servers at the root or country-level. Using this technique, we examine nine months of activity from one authority to identify trends in scanning, identifying bursts corresponding to Heartbleed and broad and continuous scanning of ssh.

14. Liang Zhu, Zi Hu, John Heidemann, Duane Wessels, Allison Mankin and Nikita Somaiya

2015. Connection-Oriented DNS to Improve Privacy and Security. Proceedings of the 36th IEEE Symposium on Security and Privacy (San Jose, California, USA, May 2015), 171–186.

The Domain Name System (DNS) seems ideal for connectionless UDP, yet this choice results in challenges of eavesdropping that compromises privacy, source-address spoofing that simplifies denial-of-service (DoS) attacks on the server and third parties, injection attacks that exploit fragmentation, and reply-size limits that constrain key sizes and policy choices. We propose T-DNS to address these problems. It uses TCP to smoothly support large payloads and to mitigate spoofing and amplification for DoS\@. T-DNS uses transport-layer security (TLS) to provide privacy from users to their DNS resolvers and optionally to authoritative servers. TCP and TLS are hardly novel, and expectations about DNS suggest connections will balloon client latency and overwhelm server with state. Our contribution is to show that T-DNS significantly improves security and privacy: TCP prevents denial-of- service (DoS) amplification against others, reduces the effects of DoS on the server, and simplifies policy choices about key size. TLS protects against eavesdroppers to the recursive resolver. Our second contribution is to show that with careful implementation choices, these benefits come at only modest cost: end-to-end latency from TLS to the recursive resolver is only about 9% slower when UDP is used to the authoritative server, and 22% slower with TCP to the authoritative. With diverse traces we show that connection reuse can be frequent (60– 95% for stub and recursive resolvers, although half that for authoritative servers), and after connection establishment, experiments show that TCP and TLS latency is equivalent to UDP\@. With conservative timeouts (20 s at authoritative servers and 60 s elsewhere) andestimated per-connection memory, we show that server memory requirements match current hardware: a large recursive resolver may have 24k active connections requiring about 3.6 GB additional RAM\@. Good performance requires key design and implementation decisions we identify: query pipelining, out-of-order responses, TCP fast-open and TLS connection resumption, possible. and plausible timeouts.

15. Xun Fan, Ethan Katz-Bassett and John Heidemann 2015. Assessing Affinity Between Users and CDN Sites. Proceedings of the 7th IEEE International Workshop on Traffic Monitoring and Analysis (Barcelona, Spain, Apr. 2015).

Large web services employ CDNs to improve user performance. CDNs improve performance by serving users from nearby Front-End (FE) Clusters. They also spread users across Front-End Clusters when one is overloaded or unavailable and others have unused capacity. Our paper is the first to study the dynamics of the user-to-Front-End Cluster mapping for Google and Akamai from a large range of client prefixes. We measure how 32,000 prefixes associate with Front-End Clusters in their CDNs every 15 minutes for more than a month. We study geographic and latency effects of mapping changes, showing that 50– 70% of prefixes switch between Front-End Clusters that are very distant from each other (more than 1,000 km), and that these shifts sometimes (28–40% of the time) result in large latency shifts (100 ms or more). Most prefixes see large latencies only briefly, but a few (2–5%) see high latency much of the time. We also find that many prefixes are directed to several countries over the course of a month, complicating questions of jurisdiction.

16. Lin Quan, John Heidemann and Yuri Pradkin 2014. When the Internet Sleeps: Correlating Diurnal Networks With External Factors. Proceedings of the ACM Internet Measurement Conference (Vancouver, BC, Canada, Nov. 2014), 87–100.

As the Internet matures, policy questions loom larger in its operation. When should an ISP, city, or government invest in infrastructure? How do their policies affect use? In this work, we develop

a new approach to evaluate how policies, economic conditions and technology correlates with Internet use around the world. First, we develop an adaptive and accurate approach to estimate block availability, the fraction of active IP addresses in each /24 block over short timescales (every 11 minutes). Our estimator provides a new lens to interpret data taken from existing long-term outage measurements, thus requiring no additional traffic. (If new collection was required, it would be lightweight, since on average, outage detection requires less than 20 probes per hour per /24 block; less than 1% of background radiation.) Second, we show that spectral analysis of this measure can identify diurnal usage: blocks where addresses are regularly used during part of the day and idle in other times. Finally, we analyze data for the entire responsive Internet (3.7M/24 blocks) over 35 days. These global observations show when and where the Internet sleeps—networks are mostly always-on in the US and Western Europe, and diurnal in much of Asia, South America, and Eastern Europe. ANOVA (Analysis of Variance) testing shows that diurnal networks correlate negatively with country GDP and electrical consumption, quantifying that national policies and economics relate to networks.

17. Zi Hu, Liang Zhu, Calvin Ardi, Ethan Katz-Bassett, Harsha V. Madhyastha, John Heidemann and Minlan Yu 2014. The Need for End-to-End Evaluation of Cloud Availability. Proceedings of the Passive and Active Measurement Workshop (Marina del Rey, California, USA, Mar. 2014), 119– 130.

People's computing lives are moving into the cloud, making understanding cloud availability increasingly critical. Prior studies of Internet outages have used ICMP-based pings and traceroutes. While these studies can detect network availability, we show that they can be inaccurate at estimating cloud availability. Without care, ICMP probes can underestimate availability because ICMP is not as robust as application-level measurements such as HTTP. They can overestimate availability if they measure reachability of the cloud's edge, missing failures in the cloud's back-end. We develop methodologies sensitive to five "nines" of reliability, and then we compare ICMP and end-to-end measurements for both cloud VM and storage services. We show case studies where one fails and the other succeeds, and our results highlight the importance of application-level retries to reach high precision. When possible, we recommend end-to-end measurement with application- level protocols to evaluate the availability of cloud services.

18. Lin Quan, John Heidemann and Yuri Pradkin 2014. Visualizing Sparse Internet Events: Network Outages and Route Changes. Computing. 96, 1 (Jan. 2014), 39–51.

To understand network behavior, researchers and enterprise network operators must interpret large amounts of network data. To understand and manage network events such as outages, route instability, and spam campaigns, they must interpret data that covers a range of networks and evolves over time. We propose a simple clustering algorithm that helps identify spatial clusters of network events based on correlations in event timing, producing 2-D visualizations. We show that these visualizations where they reveal the extent, timing, and dynamics of network outages such as January 2011 Egyptian change of government, and the March 2011 Japanese earthquake. We also show they reveal correlations in routing changes that are hidden from AS- path analysis.

19. Alefiya Hussain, Yuri Pradkin and John Heidemann 2013. Replay of Malicious Traffic in Network Testbeds. Proceedings of the 13th IEEE Conference on Technologies for Homeland Security (HST) (Waltham, Massachusetts, USA, Nov. 2013), (to appear).

In this paper we present tools and methods to integrate attack measurements from the Internet with controlled experimentation on a network testbed. We show that this approach provides greater fidelity than synthetic models. We compare the statistical properties of real- world attacks

with synthetically generated constant bit rate attacks on the testbed. Our results indicate that trace replay provides fine time- scale details that may be absent in constant bit rate attacks. Additionally, we demonstrate the effectiveness of our approach to study new and emerging attacks. We replay an Internet attack captured by the LANDER system on the DETERLab testbed within two hours.

20. Matt Calder, Xun Fan, Zi Hu, Ethan Katz-Bassett, John Heidemann and Ramesh Govindan 2013. Mapping the Expansion of Google's Serving Infrastructure. Proceedings of the ACM Internet Measurement Conference (Barcelona, Spain, Oct. 2013), 313–326.

Modern content-distribution networks both provide bulk content and act as "serving infrastructure" for web services in order to reduce user- perceived latency. Serving infrastructures such as Google's are now critical to the online economy, making it imperative to understand their size, geographic distribution, and growth strategies. To this end, we develop techniques that enumerate IP addresses of servers in these infrastructures, find their geographic location, and identify the association between clients and clusters of servers. While general techniques for server enumeration and geolocation can exhibit large error, our techniques exploit the design and mechanisms of serving infrastructure to improve accuracy. We use the EDNS-client-subnet DNS extension to measure which clients a service maps to which of its serving sites. We devise a novel technique that uses this mapping to geolocate servers by combining noisy information about client locations with speed-of-light constraints. We demonstrate that this technique substantially improves geolocation accuracy relative to existing approaches. We also cluster server IP addresses into physical sites by measuring RTTs and adapting the cluster thresholds dynamically. Google's serving infrastructure has grown dramatically in the ten months, and we use our methods to chart its growth and understand its content serving strategy. We find that the number of Google serving sites has increased more than sevenfold, and most of the growth has occurred by placing servers in large and small ISPs across the world, not by expanding Google's backbone.

21. Lin Quan, John Heidemann and Yuri Pradkin 2013. Trinocular: Understanding Internet Reliability Through Adaptive Probing. Proceedings of the ACM SIGCOMM Conference (Hong Kong, China, Aug. 2013), 255–266

Natural and human factors cause Internet outages—from big events like Hurricane Sandy in 2012 and the Egyptian Internet shutdown in Jan.%7e2011 to small outages every day that go unpublicized. We describe Trinocular, an outage detection system that uses active probing to understand reliability of edge networks. Trinocular is principled: deriving a simple model of the Internet that captures the information pertinent to outages, and populating that model through long-term data, and learning current network state through ICMP probes. It is parsimonious, using Bayesian inference to determine how many probes are needed. On average, each Trinocular instance sends fewer than 20 probes per hour to each /24 network block under study, increasing Internet "background radiation" by less than 0.7%. Trinocular is also predictable and precise: we provide known precision in outage timing and duration. Probing in rounds of 11 minutes, we detect 100% of outages one round or longer, and estimate outage duration within one- half round. Since we require little traffic, a single machine can track 3.4M /24 IPv4 blocks, all of the Internet currently suitable for analysis. We show that our approach is significantly more accurate than the best current methods, with about one-third fewer false conclusions, and about 30% greater coverage at constant accuracy. We validate our approach using controlled experiments, use Trinocular to analyze two days of Internet outages observed from three sites, and re-analyze three years of existing data to develop trends for the Internet.

22. Xun Fan, John Heidemann and Ramesh Govindan 2013. Evaluating Anycast in the Domain Name System. Proceedings of the IEEE Infocom (Turin, Italy, Apr. 2013), 1681–1689.

IP anycast is a central part of production DNS. While prior work has explored proximity, affinity and load balancing for some anycast services, there has been little attention to third-party discovery and enumeration of components of an anycast service. Enumeration can reveal abnormal service configurations, benign masquerading or hostile hijacking of anycast services, and help characterize anycast deployment. In this paper, we discuss two methods to identify and characterize anycast nodes. The first uses an existing anycast diagnosis method based on CHAOS-class DNS records but augments it with traceroute to resolve ambiguities. The second proposes Internet-class DNS records which permit accurate discovery through the use of existing recursive DNS infrastructure. We validate these two methods against three widely-used anycast DNS services, using a very large number (60k and 300k) of vantage points, and show that they can provide excellent precision and recall. Finally, we use these methods to evaluate anycast deployments in top-level domains (TLDs), and find one case where a third-party operates a server masquerading as a root DNS anycast node as well as a noticeable proportion of unusual DNS proxies. We also show that, across all TLDs, up to 72% use anycast.

23. Lin Quan, John Heidemann and Yuri Pradkin 2013. Visualizing Sparse Internet Events: Network Outages and Route Changes. Computing. (Jan. 2013), to appear.

To understand network behavior, researchers and enterprise network operators must interpret large amounts of network data. To understand and manage network events such as outages, route instability, and spam campaigns, they must interpret data that covers a range of networks and evolves over time. We propose a simple clustering algorithm that helps identify spatial clusters of network events based on correlations in event timing, producing 2-D visualizations. We show that these visualizations where they reveal the extent, timing, and dynamics of network outages such as January 2011 Egyptian change of government, and the March 2011 Japanese earthquake. We also show they reveal correlations in routing changes that are hidden from AS- path analysis.

24. Lin Quan, John Heidemann and Yuri Pradkin 2012. Visualizing Sparse Internet Events: Network Outages and Route Changes. Proceedings of the First ACM Workshop on Internet Visualization (Boston, Mass., USA, Nov. 2012).

To understand network behavior, researchers and enterprise network operators must interpret large amounts of network data. To understand and manage network events such as outages, route instability, and spam campaigns, they must interpret data that covers a range of networks and evolves over time. We propose a simple clustering algorithm that helps identify spatial clusters of network events based on correlations in event timing, producing 2-D visualizations. We show that these visualizations where they reveal the extent, timing, and dynamics of network outages such as January 2011 Egyptian change of government, and the March 2011 Japanese earthquake. We also show they reveal correlations in routing changes that are hidden from AS- path analysis.

6 Conclusions and Recommendations

6.1 Conclusions

We have completed all project deliverables and provided **1497 datasets** making up **76.9TB compressed** or **324.6TB uncompressed** to **223 unique researchers to date.** In addition, we published **70 publications** and established **several new measurement methods**, Trinocular outage detection, BotDigger botnet detection, and Verfploeter anycast mapping**.** Finally, we have distributed **23 software packages** and numerous updates.

6.2 Recommendations

Although we completed all objectives of the original proposal, as an example of research infrastructure, continuing to update the datasets we provide with fresh versions, and continuing to provide this data to fresh researchers are both important, ongoing needs. In addition, important new directions in measurement include expanding census and outage detection to IPv6 and adapting packet header collection to 10, 40, and 100Gb/s media. We hope to continue this work in the LACANIC effort, a new project with DHS support.

LIST OF ABBREVIATIONS AND ACRONYMS

2LD	Second Level Domain (name)
ACM	Association for Computing Machinery
ACSAC	Annual Computer Security applications Conference
AFRL	Air Force Research Laboratory
AIMS	Active Internet Measurement Systems
AINTEC	Asian Internet Engineering Conference
ANSI	American National Standards Institute
ANT	the Analysis of Network Traffic group (our research project, https://ant.isi.edu)
AS	Autonomous System
BAA	Broad Area Announcement
BIND	Berkeley Internet Name Daemon (DNS server software)
C&C	Command and Control (of a botnet)
CAIDA	Center for Applied Internet Data Analysis
Co-PI	Co-Principal Investigator
CSU	Colorado State University
DAG	Data Acquisition and Generation
DANE	DNS-based Authentication of Named Entities
DGA	Domain-name Generation Algorithm
DETER	
DHS	Department of Homeland Security
DNS	Domain Name System
DNSSEC	Domain Name System Security
DNS-OAR	DNS Operations, Analysis and Research Center
DDoS	Distributed DoS
DoS	Denial of Service
Dx.x	Deliverable x.x
ECMP	Equal Cost Multi-path
FCC	Federal Communications Commission
HPCC	High Performance Computing Center
HTTP	HyperText Transport Protocol
HSARPA	Homeland Security Advanced Projects Research Agency
HSL	Hue/Saturation/Lightness
Gb/s	Gigabits per second
ICERM	Institute for Computational and Experimental Research in Mathematics
ICMP	Internet Control Message Protocol
IEEE	Institute of Electrical and Electronics Engineers
IMC	Internet Measurement Conference
IMPACT	Information Marketplace for Policy and Analysis of Cyber-risk & Threats
IP	Internet Protocol
IPv4	IP version 4
IPv6	IP version 6
IRB	Institutional Research Board
IRC	Internet Relay Chat
ISC	Internet Systems Consortium
ISI	Information Sciences Institute

ISOC	Internet Society
ISP	Internet Service Provider
IT	Information Technology
LACANIC	Los Angeles/Colorado Application and Network Information Community
LACREND	Los Angeles/Colorado Research Exchange for Network Data
LANDER	LosAngeles Network Data Exchange and Repository
Mb/s	Megabits per second
MOA	Memorandum of Agreement
NANOG	North American Network Operators Group
NDSS	Network and Distributed System Security Symposium
NTP	Network Time Protocol
NXDomain	Non-existant (DNS) Domain name
OMB	Office of Management and Budget
PI	Principle Investigator
PREDICT	Protected REpository for the Defense of Infrastructure against Cyber Threats
P2P	Peer-to-Peer
R&D	Research and Development
RGB	Red/Green/Blue
RDNS	Reverse DNS
SIGCOMM	Special Interest Group in Communications
SOW	Statement of Work
STARTTLS	Start the TLS protocol
TB	Terabytes
TCP	Transmission Control Protocol
T-DNS	TCP and TLS-based DNS
TLD	Top Level (Internet) Domain
TLS	Transport Layer Security
TLSA	TLS record in DNS for use with DNAE
TTx	Technology Transfer
TR	Technical Report
UDP	User Datagram Protocol
UK	United Kingdom
US	United States
USA	United States of America
USC/ISI	University of Southern California/Information Sciences Institute
UTC	Coordinated Universal Time
VA	Virginia
VC	Version Control