To appear in the Proceedings of the 6th IEEE European Symposium on Security and Privacy (Euro S&P'21). Virtual event. September 2021. This preprint is the author version.

SoK: A Framework for Asset Discovery: Systematizing Advances in Network Measurements for Protecting Organizations

Mathew Vermeer*, Jonathan West†, Alejandro Cuevas‡, Shuonan Niu†,
Nicolas Christin‡, Michel van Eeten*, Tobias Fiebig*, Carlos Gañán*, Tyler Moore†
**Delft University of Technology* {m.vermeer,m.j.g.vaneeten,t.fiebig,c.hernandezganan}@tudelft.nl
†*University of Tulsa* {codiwest,shn5898,tyler-moore}@utulsa.edu
‡*Carnegie Mellon University* {acuevasv,nicolasc}@cmu.edu

***Abstract*—Asset discovery is fundamental to any organization's cybersecurity efforts. Indeed, one must accurately know which assets belong to an IT infrastructure before the infrastructure can be secured. While practitioners typically rely on a relatively small set of well-known techniques, the academic literature on the subject is voluminous. In particular, the Internet measurement research community has devised a number of asset discovery techniques to support many measurement studies over the past five years. In this paper, we systematize asset discovery techniques by constructing a framework that comprehensively captures how network identifiers and services are found. We extract asset discovery techniques from recent academic literature in security and networking and place them into the systematized framework. We then demonstrate how to apply the framework to several case studies of asset discovery workflows, which could aid research reproducibility. These case studies further suggest opportunities for researchers and practitioners to uncover and identify more assets than might be possible with traditional techniques.**

1. Introduction

The bedrock of good security posture for any organization's IT security is accurate knowledge of which systems belong to its IT infrastructure. If a company does not know what systems and software it is using, it cannot ensure their security and, therefore, cannot secure the organization. All risk management strategies are predicated on having full visibility into the organization's assets. Hence, it is no coincidence that the first function in the NIST Cybersecurity Framework is to "identify" and the first category within this function is "asset management" [89]. For the same reasons, the ISO/IEC 27001 information security standard [4] requires the identification of assets that are associated with information and information processing facilities, as well as keeping this overview updated.

While the need to identify assets is obvious, how to accomplish this identification in practice is not. Organizations consistently struggle to keep a complete inventory of their assets—and consistently fail to do so. (See [9] for a couple of particularly telling examples.) Even a medium-sized organization can easily deploy tens of thousands of systems, software platforms and applications. This asset inventory is constantly changing, with many changes unplanned or unrecorded. This is further amplified by the problem of "shadow IT": IT systems that are "not known, accepted and supported" by an organization's official IT department [101]. All of this means that any centralized inventory of assets, such as those prescribed in the Information Technology Infrastructure Library (ITIL) and ISO procedures, necessarily contain errors and omissions. Automated techniques to identify these gaps are therefore essential for defenders to adopt.

Asset discovery is not only a defensive activity, but also a crucial component of both a red team's and attacker's process. Adversaries first need to know *what* they are targeting and what they *want* to target, before they can start attacking systems. Furthermore, the information deficit in organizations, as outlined above, means that attackers can get a serious advantage by having an equally good or even better overview of an organization's assets than the defenders on the inside of the network.

For both sides, attackers as well as defenders, numerous techniques have been researched and tools have been developed. Many of these techniques are integrated into various industry toolchains and handbooks. However, the field of Internet measurement also blossomed in recent years, and many new techniques have emerged that are not yet used or even known by practitioners involved in asset discovery. The techniques are often developed for other purposes, such as diagnosing network disruptions; they are often not explicitly associated with asset discovery; and they are fragmented across different academic venues and communities. Furthermore, in recent years, automation and commoditization of these tools has also become prevalent, with offensive and defensive uses alike. Automated tools and techniques to discover assets have been used to show just where asset management fails—think of exposed systems showing up in search engines like Shodan or in the scans of pen testers [3, 17]. While automated tools and services leveraging well-known asset discovery techniques are widely available, many defenders may still not be aware of the most recent techniques available to understand their network, and they may lack valuable information on what capabilities for asset discovery attackers potentially have.

In this paper, we fill this gap by surveying and systematizing new developments in asset discovery, *i.e.*, techniques that appeared in papers in 14 leading academic venues over the past five years. Our goal is twofold: First, we provide an overview of recent techniques. This exploration is time-consuming to acquire given that these techniques are spread out over different communities and are often not explicitly associated with the concept of asset discovery. In other words, we aim to lower the search costs for practitioners and researchers to find techniques that they might use or that might be used against them. Second, we provide a systematization of asset discovery techniques that illustrate how the different techniques can be chained to each other for the development of toolchains. One technique's outputs are another technique's inputs. We frequently also observe loops, in which an initial seed of assets is snowballed into an expanded set, which can then become the seed for the next cycle.

The objectives of the asset discovery process determine how to select and combine various techniques. These objectives will vary across the use cases of system administrators, security officers, red-teams, and actuaries. A pentester might only need to opportunistically discover a few vulnerable assets, preferably via passive techniques, to gain a foothold in the target organization. A system administrator, on the other hand, needs to find all Internet-reachable systems in order to secure them and has no reason to avoid active techniques. An actuary, determining the risk exposure of a potential cyberinsurance customer, needs a discovery toolchain that can measure assets' footprints reliably and consistently across large numbers of organizations and can therefore not rely on including the inside knowledge of system administrators in these organization. One key aim of our systematization is to support professionals in these use cases to improve their asset discovery processes.

We scope our work in two key ways. First, as mentioned above, we focus on the Internet measurement techniques that are reported on in the last five years (2015–2019) in 14 academic venues. We therefore focus on techniques utilized in recent work. These methods are often novel, but not always: if an older, more established technique is still valuable, it could be utilized by the authors. Often, while asset identification is required to complete the research, the paper's novelty lies elsewhere. Hence, while we do not completely catalog all asset identification techniques, we do comprehensively identify how asset identification is being carried out by researchers today. The scoping around Internet-based techniques also means that we focus on active measurement techniques, *i.e.*, we exclude passive techniques that require a privileged vantage point inside a network. This corresponds to our goal of providing a comprehensive document for red-team members during engagements, and a selection of techniques for organizations to understand the capabilities and visibility outsiders can gain on their network.

We provide the following contributions:

- We develop a framework that systematizes asset discovery techniques.
- We comprehensively survey asset discovery techniques extracted from academic literature published in top networking and security conferences, as well as related venues, and map these techniques onto the framework.
- We demonstrate how the framework can be used to construct workflows where techniques can be chained together for asset discovery use cases.

Structure: We introduce our framework, important terminology, and our literature search and systematization Methodology in Section 2.1. Subsequently we present our results in Section 3, and illustrate our findings with case studies in Section 4. Finally, we discuss our findings and conclude in Section 5.

2. Systematization Methodology

In this section, we first define the necessary terminology and processes to delineate the scope of our subsequent literature review. Our definitions include what we consider an asset, what it means to *discover* an asset, and what the formal meta-process of discovering an asset looks like. Next, we describe our literature review methodology and the steps we took to conduct the survey presented in this work. Finally, we conclude the section by introducing the notation we use for presenting the results of our survey.

2.1. Terms and Definitions

Assets The first item to define is the meaning of *asset*. When we turn to existing standardization frameworks for a clear definition, we find that ISO/IEC 27001 in its 2005 revision defines asset very broadly as *"anything that has value to the organization."* Similarly, the NIST framework regards as assets both hardware (*"ID.AM-1: physical devices and systems within the organization"*) and software (*"ID.AM-2: software platforms and applications"*) [89]. These two definitions take two very different perspectives on the term "asset": The NIST framework focuses on "hard" assets, *i.e.*, physical devices, systems, software platforms and applications. Conversely, the ISO/IEC definition also includes "soft" assets, such as information pertaining to the employees of an organization like job titles and email addresses, or business and financial information. As our survey focuses on external *network* measurements and new techniques to identify publicly-visible IT assets, we adopt and rephrase the definition of NIST:

Definition 1. We consider as assets: (i) all *network identifiers*, *e.g.*, addresses, FQDNs, and contents of DNS zones, and (ii) the *network services* reachable via these network identifiers, defined by the protocol they are implementing, and any information they provide upon initial connect in their banners, *e.g.*, implementation names and version numbers.

We specifically consider as out-of-scope inferred properties of these assets, *e.g.*, whether a certain network service is vulnerable to an exploit (either based on reasoning about the version number or attempting the exploit), inferring whether multiple network identifiers point to the same physical or logical host, or whether multiple discovered network services constitute a joint application service.

Asset Discovery Next, we define asset *discovery*.

Definition 2. The discovery of an asset means that the existence of an asset associated with a specific organization becomes, for the first time, known to an entity.

This entails that the entity which discovers an asset has not been aware of its existence, and that the discovery of an asset is independent of third parties knowing about the asset prior to the discovery process.

Bootstrapping Next, the issue of asset discovery leads to the question of *whose* assets are to be discovered. Going from our definition of discovery, we want to discover assets that belong to *specific* entities. Depending on the asset discovery technique used, it might be necessary to first manually discover and select assets connected to an entity, before the technique can be applied. We call this step “bootstrapping” and define it as follows:

Definition 3. Bootstrapping is the process of obtaining the initial seed of information that the asset discovery techniques require as input to discover assets.

The information selected in the bootstrapping stage differs depending on the objective. For penetration tests or network monitoring, one searches a specific organization’s assets. For benchmarking, risk profiling or risk prediction, the process might focus on the assets of a whole sector or group of organizations. In general and especially for adversarial asset discovery, the discoverer does not know a company’s address space in advance Even in the case of pen-testing, the organization might not have a complete understanding of all addresses where assets reside, because of shadow IT, outsourcing and cloud infrastructure.

Starting with network addresses is a common, but certainly not the only, technique for bootstrapping. It is relatively straightforward, as the organization name can be fed into common search engines and databases that associate it with the addresses and networks registered by the organization. Examples are querying databases that contain WHOIS, BGP [70, 78] or passive DNS [117] data for the relevant strings. Another example is using a general search engine to find web domains containing an organizations name [78]. Along the same lines, one can search for specific strings in databases of SSL/TLS certificates, as Bonkoski et al. do, to extract relevant domain names [18]. In case of a correct match, the certificate will contain domain name information on the organization of interest.

Bootstrapping techniques using string matching have to contend with incorrect identification. Organizations names might not be unique, or overlap with other names. A match of the name with a WHOIS record might thus incorrectly attribute the IP range of a similarly-named organization to an organization in the asset discovery scope. At the same time, an organization could own several network identifiers that cannot be matched with their name. This, for example, occurs during mergers and acquisitions, when IP ranges that used to belong to an acquired entity are still registered under the old name. Depending on the use case for asset discovery, different strategies are needed to deal with incorrect identification. We will discuss these in Section 4.

The asset discovery process Finally, from our definition of assets, discovery, the initial bootstrapping step, and the premise of asset discovery via external measurement, we arrive at a model for the formal asset discovery process, shown in Figure 1:

Definition 4. The general asset discovery process consists of an initial bootstrapping process, feeding network identifiers into a recursive process to discover more network identifiers (to be fed back into the process) and network services associated with these identifiers.

The initial discovery of network identifiers is restricted to the organization of which assets should be discovered (0), which may then yield further associated network identifiers for investigation (1). The discovery of network services associated with network identifiers (2) is straightforward, *i.e.*, one just checks for open ports on the network address (if the identifier is an address) or resolves the identifier to a network address and then checks for open ports. In addition to basic information on the network service, the banners of the detected open ports may then reveal further network identifiers (3) or further network services (4). As per our asset definition above, we explicitly leave out-of-scope the discovery of known software vulnerabilities in discovered network services.

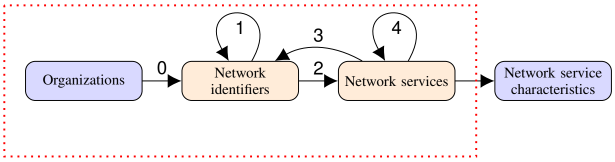

Figure 1. **Framework for asset identification techniques.** Each arrow represents a set of discovery techniques. We highlight the scope of our work with the red dotted line. The inference of software vulnerabilities, either by directly testing for them or by inferring them from version numbers, is the next step after asset discovery and explicitly out of scope for our survey. See Table 1 for examples for network identifiers and services.

TABLE 1. Examples for “Network Identifiers” and “Network Services” in Figure 1 encountered during our literature survey. While the list of network identifiers is exhaustive, the list of network services is not. The list only contains the network services that are explicitly mentioned in the literature.

Network identifiers	Network services
IPv4 address	Web server
IPv6 address	Name server
Domain and subdomain	Proxy server
Autonomous System	Mail server
BGP prefix	SSH server
IPv4 prefix	FTP server
IPv6 prefix	Cryptocurrency clients
	VPN
	Honeypot
	CMS services
	...

2.2. Literature Search Process

Based on our earlier definition of assets, and the asset discovery process, we can now search the scientific literature for methods and techniques informing or enabling the asset discovery process. We scope our literature search to the major publication venues of Computer Security, Network Measurement, and Network Operations from the past five years (*i.e.*, 2015–2019).

Initially, four of the co-authors independently investigated 940 papers from five years of a leading security conference, ACM CCS, and a leading networking conference, ACM IMC. The researchers were tasked to identify papers that performed asset discovery. They then sought consensus by discussing conflicts, i.e., papers not included or excluded by all researchers. Because our research setting focused on achieving consensus, we opted to not explicitly calculate intercoder reliability, consistent with the recommendations of McDonald et al. [83].

Using the 32 selected papers, the entire team identified criteria for what constitutes asset discovery, and hence the exclusion and inclusion of papers, and applied those criteria to the remaining venues indicated in Table 2. We

TABLE 2. OVERVIEW OF THE 93 PAPERS WE SELECTED FROM SIX MAJOR SECURITY AND SEVEN NETWORKING VENUES ANALYZED DURING THE LITERATURE SURVEY. IEEE/IFIP NOMS ONLY TAKES PLACE EVERY SECOND YEAR.

	2015		2016		2017		2018		2019		Total	
Security	*Papers*	*Selected*	*Papers*	*Selected*	*Papers*	*Selected*	*Papers*	*Selected*	*Papers*	*Selected*	*Papers*	*Selected*
ACM CCS	128	6	137	2	151	2	134	0	177	2	727	12
IEEE S&P	55	0	55	0	60	1	62	2	84	0	316	3
ISOC NDSS	51	2	60	3	68	0	71	0	89	1	339	6
USENIX Security	67	1	72	2	85	2	100	4	113	1	437	10
RAID	28	0	21	2	21	0	32	1	37	1	139	4
IEEE/IFIP DSN	75	0	65	2	80	1	62	4	54	1	336	8
ACSAC	48	0	48	3	48	1	60	1	60	0	264	5
Networking												
ACM SIGCOMM	40	0	39	0	37	0	40	1	32	1	188	2
ACM IMC	43	2	46	6	42	3	43	3	39	6	213	20
USENIX NSDI	42	0	45	0	46	1	40	0	49	1	222	2
USENIX ATC	47	0	47	1	60	0	76	0	71	0	301	1
IEEE/IFIP NOMS	–	–	222	2	–	–	221	0	–	–	443	2
PAM	27	4	30	1	20	2	20	7	20	0	117	14
TMA	16	0	16	2	29	0	34	1	35	1	130	4
Total	667	15	903	26	747	13	995	24	860	15	**4,172**	**93**

discarded papers clearly unrelated to the subject based on the papers' abstract. If the abstract and/or introduction of a paper indicated usage of an asset discovery process, we searched the rest of the sections. In cases where we did not find any asset discovery process, the paper was discarded.

In total, we found 93 papers that utilized asset discovery techniques. Notably, the venues from which we select most papers are PAM and ACM IMC. This aligns with our expectations, as both are venues focused on network measurements and novel measurement techniques. Other networking venues, as for example USENIX NSDI and ACM SIGCOMM saw generally fewer papers selected. This is related to those venues also featuring a major fraction of non-measurement networking papers, for example, high-performance networking related research. For security venues, the number of selected papers is again lower than for the purely network measurement focused venues. Again, this can be explained by the more diverse focus of these venues.

2.3. Systematization Syntax and Structure

The papers that we select for the systematization contain tools or techniques that can be used for asset discovery. While some papers have an asset discovery tool or technique as the main contribution, many simply use discovery as a means to an end. To perform a meaningful systematization and comparison of techniques, we have to transform this heterogeneous body of literature to a uniform, comparable terminology.

We formalize each paper's contribution to asset identification by identifying the corresponding edge it represents in Figure 1. We base this on our definition of the general asset discovery process: all tools and techniques described in the selected papers work from an asset as input, be it a network identifier, or network service. This input is then used to execute any number of tasks, after which the tool or technique produces a certain output, which again is one or multiple assets. In general, these processes can be represented in the form, where # is the edge the process corresponds to:

#: input asset $\Longrightarrow$ discovery method $\Longrightarrow$ output asset
network_identifier *network_identifier*
network_service *network_service*

Take as an example the usage of zone walking to enumerate hosts in an IPv6 reverse DNS zone [20]. This methodology relies on looking up an IPv6 prefix for an organization and returns the zone's entries as output. Both of these assets are network identifiers, hence it corresponds to Edge 1 from Figure 1. In this case, the discovery method can be represented in the following form:

1: IPv6 prefix ⇒ NSEC3 zone walking ⇒ addresses.

Each paper can include one or more such instances of techniques. Similarly, a paper can also include an asset discovery technique that can be decomposed into multiple different ones, each corresponding to an edge in Figure 1. Each sub-technique that results from this decomposition is individually systematized. Tables 3-6 apply this syntax to all of the literature discussed in Section 3. This framework allows us to create a clear overview of different techniques in the literature, and perform meaningful comparisons between them. Furthermore, it enables the easy chaining of techniques, as well as identifying opportunities to combine techniques that have not yet been pursued.

3. Asset discovery

In this section we present the systematization of techniques for asset discovery based on the model illustrated in Figure 1. While many papers necessarily must first complete the bootstrapping step mapping organizations of interest to network identifiers indicated in Edge 0, this step is invariably application-specific. Hence, we focus the remaining discussion on the more generally applicable steps 1–4 involving the discovery of network identifiers and services. Each edge 1–4 is discussed in the respective subsections 3.1–4. The literature is summarized using the systematization syntax in Tables 3, 4, 5, and 6, which is discussed in Section 3.5.

3.1. Discovering Network Identifiers from Network Identifiers – Edge (1)

Network identifiers signal the exposure of a company network. As such, a great part of the recent research has focused on developing new methods or extending well-known techniques to discover resources. We first discuss techniques that use known network identifiers to discover previously unknown identifiers. We group the techniques by the input asset, since they are each processed differently. We will first discuss methods that use domain names as an input, which are primarily concerned with DNS and passive DNS, as well as complications that arise due to the use of Cloud-based Security Providers. Next, we explain how onion URLs are processed differently than domains and IP addresses to discover network identifiers. We then examine efforts using IP addresses (v4 and v6) as input, which integrate closely with that of BGP prefixes, Autonomous System Numbers (ASNs), and the tools and

TABLE 3. Our systematization of papers containing techniques under Edge 1.

Citation	Edge	Input asset	Discovery technique	Output asset	Remarks
Domain Input					
[34, 117]	1	Domain	A/AAAA records from passive DNS	IPv4/IPv6 addresses	
[34]	1	Domain	CNAME record from passive DNS	Domain	
[30]	1	Domain	Extract A and AAAA from DNS ANY queries	IPv4 and IPv6 addresses	
[74]	1	Domain	Use TLD zonefiles to get IDNs	IP	Uses TLD zones to get IDNs & resolve names to IPs
[59]	1	Domain	DNS Zone walking	IPv6	
[24]	1	Domain	DNS logs	IP	
[14, 34, 47, 49, 69]	1	Domain	DNS A/AAAA query	IPv4/IPv6 addresses	
[62]	1	Domain	DNS A-record	IP	Leakage after terminating a DDoS protection service
[34]	1	Domain	DNS CNAME query	Domain	
[125]	1	Domain	Resolve domains in DNS MX, TXT records	IP address	Names in MX/TXT RRs resolve to IP addr. behind a CBSP
[125]	1	Domain	Passive DNS search	IP address	Data from passive DNS may reveal an IP hidden by a CBSP
[6, 33, 85]	1	Domain	Passive DNS search	IP address	
[76]	1	Domain	Passive DNS wildcard search	Subdomains	For example, *.domain.com
[104]	1	Domain	ZDNS	IP address	
[109]	1	Domain	Satellite tool	IP addresses	The tool discovers IP addresses of used CDN infrastructure
[25]	1	Domain	DomainScouter tool	Domains	Identifies IDNs; Add. checks needed to confirm ownership
[125]	1	Domain	Search domain in IP/SSL certificate pair collection; domain from certificate that resolves differently exposes real IP.	IP address	Discovers the real IP that is hidden by a CBSP
[111]	1	Domain	WHOIS	AS	
[53]	1	Domain	DNS A queries from geographically distributed vantage points	IP addresses	Different IP addresses can be returned for the same domain if websites are hosted by CDNs
[82]	1	Domain	CARONTE tool	IP address	Comb. of techniques to identify IPs of hidden services
IPv4 Input					
[86]	1	IPv6 addresses	6gen algorithm	IPv6 addresses	Generates IPv6 Addr. candidates from IPv6 hitlists
[45]	1	IPv6 addresses	Entropy/IP	IPv6 addresses	Generates IPv6 Addr. candidates from IPv6 hitlists
[10, 22, 28, 64, 93]	1	IP address	CAIDA prefix-to-AS data mapping (pfx2as)	AS	
[33, 49]	1	IPv4/IPv6 pr.	CAIDA prefix-to-AS	AS	
[96]	1	BGP prefix	CAIDA AS-to-organization mapping dataset	ASes	Identifies ASes that belong to the same organization
[98]	1	IP address	Censys data query	AS	
[98]	1	IP address	MaxMind AS data query	AS	
[50]	1	IP address	CAIDA topology dataset search	IP address	Discovers IP address belonging to the same router
[63]	1	IP address	OpenINTEL historical active DNS dataset	Domain	Uses historic data; records could be out of date
[22, 34]	1	IP address	Reverse DNS dataset search	Domain	
[97]	1	IP address	Zmap	Domain	
[49, 50, 95, 98]	1	IP address	Reverse DNS query	Domain	
[52]	1	IP address	TreeNET	IP prefix	
[51]	1	IP address	WISE	IP prefix	
[10, 56, 70]	1	IP address	RouteViews/RIPE RIS data	BGP prefix	
[27, 35, 44, 96, 129]	1	IP address	RouteViews/RIPE RIS data	AS	
[118]	1	IP address	MaxMind WHOIS dataset search	IP addresses	The output is the IP range belonging to the organization
[94]	1	IPv6 address	UAv6 technique	IPv6 address	UAv6 is an alias resolution technique for IPv6
[34]	1,2	Domain names	Multiple probes to determine hosting	Dom. names/IP addr./srv.	Identifies multiple domains hosted on the same IP
IPv6 Input					
[42]	1	IPv6 prefixes	Algorithm in paper	IPv6 addresses	Enumerates IPv6 reverse DNS entries
[41]	1	IPv6 prefixes	Algorithm in paper (using DNS NXDOMAIN)	IPv6 addresses	Enumerates IPv6 reverse DNS entries
[20, 59]	1	IPv6 prefix	Reverse zone scanning NSEC3	IPv6 addr./networks	The output is a list of IPs in the same IPv6 zone
[12]	1	IPv6 addresses	Algorithm in paper	IPv6 addresses	Generates IPv6 Addr. candidates from IPv6 hitlists

datasets built around them. Finally, we describe techniques specific to IPv6 address discovery. Because full Internet scans are not feasible on IPv6 due to the much larger address space, new methods are required to find active IPv6 addresses. Note that it is common to see techniques which mirror others in terms of inputs and outputs.

When searching for network identifiers for a given organization, one common starting point is the known domain names for the given organization. The purpose of DNS is to resolve a given domain name to the server's IP address. The use of DNS to derive an IP address does not require special tooling. Thus, it provides a simple example of using one network identifier to learn of another network identifier. Knowing the DNS resolution for previous points in time is sometimes beneficial, as an asset may still exist at the former IP address even if a domain does no longer points to it. Taking a domain as input, Tajalizadehkhoob et al. [117] and Vissers et al. [125] suggest the usage of passive DNS databases. These databases contain logs of DNS responses received by different resolvers. The entries in the passive DNS database can be filtered using the provided domain name, revealing the IP addresses that at one point in time were associated with that domain. Passive DNS is not only useful for resolving IP addresses. Liu et al. [76] utilize a passive DNS database to perform a wildcard search to find subdomains related to a domain. Their goal was finding potential shadowed domains: subdomains of a legitimate domain that are under the control of a malicious actor without the domain owner's knowledge. The technique is also relevant for the purposes of asset discovery. Furthermore, these additional subdomains may in turn uncover additional IP addresses.

Domain owners and organizations sometimes try to protect their websites from threats, notably denial-of-service attacks, by using a Cloud-based Security Provider (CBSP) that acts like a reverse proxy. In this scenario, the domain name is directed to an IP address under the control of the CBSP. The CBSP will then proxy the HTTP(S) traffic to the source IP address, where the web service is truly hosted, by using the domain in the HTTP Host header. Plain DNS is not helpful in these cases for discovering "true" source IP addresses. Vissers et al. [125] discuss several methods to discover source IP addresses of web servers which use a network identifier as an input, including querying a passive DNS database as discussed previously. Hiding the source IP address is problematic for protocols that do not contain any host information, such as FTP and SSH. Instead of connecting directly through an IP address, administrators can opt to create a subdomain that resolves directly to the source or "real" IP address. A method mentioned by Vissers et al. connects to a subdomain to find such a service, and subsequently discover the real IP address of a server. Another method proposed by the authors trawls an Internet-wide collection of SSL certificates. Domains using CBSPs will frequently have multiple certificates for the same domain hosted by different IP addresses, one of which ends up being the true source IP. Yet another method proposed by Vissers et al. involves DNS servers. Sometimes, DNS records such as MX or TXT records still reference the source IP address that the CBSP is hiding [120]. By requesting these MX and TXT records, the true source IP address for the web server can often be revealed even when the A record is pointed to the CBSP [125]. Finally, Scott et al. develop a methodology and tool to capture the IP addresses of Content Delivery Network (CDN) deployments used for input domains [109].

Onion URLs are network identifiers used to host Tor hidden services while obfuscating the server's true IP address. Matic et al. introduce CARONTE, an automated tool to discover location leaks that betray the source IP of Tor hidden services [82]. The automated tool uses several techniques to find potential Onion URL to IP mappings. The tool extracts URLs, email addresses and IP addresses from the page. The URLs and domains are resolved to collect more candidate IP addresses. Additionally, CARONTE extracts unique strings from the page and performs search engine queries containing that string. The domains of the pages that contain this string are also resolved and the IP addresses are added to the candidate IP collection. The tool also looks for potential identifiers in the leaf certificate of HTTPS hidden services. The final techniques utilize a certificate repository. Websites and hidden services hosted on the same server may share a certificate or public key. CARONTE searches the certificate repository to see if the hidden service's certificate or public key matches any other websites, thus creating a leak. It then validates the potential IP addresses by visiting each one directly without relaying through Tor nodes.

While IP addresses are often outputs of network identifier discovery methods, they can also be used as inputs. In a simple role-reversal, Gharaibeh et al. [50] and Cangialosi et al. [22] utilize reverse DNS to obtain domain names from IP addresses.

IP addresses may also be mapped to their BGP prefixes or used to discover which Autonomous System Number (ASN) it is routed through. Jonker et al. [64] seems to collect BGP data and use an unspecified process to map each IP address to the most specific BGP prefix containing the address. For those not interested in manually collecting BGP data, RouteViews is a project by the University of Oregon which uses probes placed in different places throughout the Internet to track BGP information and makes this information publicly available [103]. RIPE RIS is a similar tool for tracking how the Internet routes traffic [102].

Both RouteViews and RIPE RIS have a variety of use cases for finding network identifiers from another identifer, which are outlined below. Benson et al. [10], and Krenc et al. [70] utilize RouteViews as well as RIPE RIS to find the announced BGP prefixes that correspond to given queried IPv4 addresses. Chung et al. [27] and Foremski et al. [44] also use the RouteViews dataset to derive the ASN from an IP address. Yeganeh et al. [129] utilizes RouteViews in addition to RIPE RIS to accomplish the IP address to ASN mapping. CAIDA uses the RouteViews data to derive a dataset which maps BGP prefixes (for IPv4 and IPv6) to their respective ASNs, which saves on labor for some users; this dataset is commonly called pfx2as [7]. Chung et al. [28] utilizes CAIDA's pfx2as dataset to find BGP prefixes from IPv4 addresses. Then they use that same dataset to map the prefix to the ASN. Benson et al. [10] and Jonker et al. [64] also use CAIDA's pfx2as dataset with the BGP prefixes they previously obtained from IP addresses to determine to which autonomous system number (ASN) they each belong. Cangialosi et al. [22], Padmanabhan et al. [93], also use CAIDA's pfx2as dataset to map IP addresses directly to ASNs.

As a last note on Internet topology discovery, Gharaibeg et al. [50] and Czyz et al. [30] utilize CAIDA Ark data to find router interface IP addresses. CAIDA Ark is a measurement platform which collects traceroute data for random portions of the Internet among other measurements. Researchers may use this traceroute data to discover router IPv4 and IPv6 addresses. While this method of discovery is not as targeted to an organization as an active scan of an organization's known address space, one can still feasibly map a given router to an organization if the organization's address space is known.

Lastly, we discuss methods which are targeted directly at IPv6 addresses. Fiebig et al. [41, 42], Beverly et al. [12], Hu et al. [59], and Borgolte et al. [20] all look specifically at IPv6 hosts and use IPv6 addresses or (individual) IPv6 networks as inputs or outputs. The methods developed by Fiebig et al. [41, 42] and Borgolte et al. [20] take a more technical approach to the discovery process than previously discussed work. The authors develop a technique to walk an IPv6 network's reverse zone using either protocol level features of DNS [41, 42, 59] or by leveraging peculiarities of hashing in DNSSEC [20, 59]. The work by Fiebig et al. utilizes differences in responses between DNS labels that do and do not have children to prune the reverse zone search tree (NXDOMAIN). Similarly, Borgolte et al. [20] leverage DNSSEC for exploring zones. DNSSEC allows operators to sign DNS responses, in turn allowing clients to verify if a DNS response has been tampered with. The issue here is the ability to prove the non-existence of records. DNSSEC does this by returning a non-existence record listing the (alphabetically) previous and next entry in the zone (NSEC), or hashed versions of these records (NSEC3). Borgolte et al. leverage the structured nature of the IPv6 reverse zone to break NSEC3 hashes in reasonable time, thereby allowing exploration of reverse zones. The network identifiers discovered by these methods are a list of allocated IPv6 addresses. Foremski et al. [45] use entropy analysis and statistical modeling from a known set of IPv6 addresses to create a tool that can generate a list of additional possible IPv6 addresses. Related to the earlier efforts to directly discover IPv6 addresses, Murdock et al. [86] propose a technique that utilizes a so called "hitlist," *i.e.*, a list of allocated IPv6 addresses as, *e.g.*, discovered by one of the previous techniques to generate further IPv6 addresses that *might* be active based on the allocation pattern observed in the hitlist.

TABLE 4. Our systematization of papers containing techniques under Edge 2.

Citation	Edge	Input asset	Discovery technique	Output asset	Remarks
Database Queries					
[64]	2	Domain	DNS A, AAAA, NS queries	Web server, name server	
[118]	2	Domain	Custom crawler tool	CMS	IDs cPanel, Plesk, DirectAdmin, & Virtualmin instances
[34, 39, 46, 58, 69]	2	Domain	DNS MX record query	Mail server	
[34]	2	Domain	MX record from passive DNS	Mail server	
[34]	2	Domain	NS DNS queries	Name server	
[34]	2	Domain	NS record from passive DNS	Name server	
[78]	2	IP address	Open Resolver Project dataset search	Name server	The dataset contains IP addresses of open resolvers
[77]	2	Domain	WHOIS statistical parsing model	Name servers	WHOIS parsing using a statistical model
[66]	2	IP address	DNS query for hostnames under own domain	Name server	Identifies open resolvers through successful name resolution
Internet-Wide Scanning					
[13]	2	IP address	Censys/Shodan port 80 and 8080 scans	Proxy server	Identifies MikroTik routers with enabled proxy
[127]	2	Domain	Dmap	Web server	Uses DNS A/AAAA queries to discover the network service
[127]	2	Domain	Dmap	Mail server	Uses DNS MX queries to discover the network service
[118]	2	Domain	Port 22 banner scan	SSH server (version)	
[39]	2	IP address	Port 25 scan	Mail server	The scan on this port aims to find a running SMTP service
[80]	2	Domain	Port 443 scan for DNS-over-HTTPS paths	Name server	Discovers DNS server that offer DNS-over-HTTPS
[30]	2	IPv4/v6 pairs	Server sibling detection alg. from [11, 107]	Multiple services	Identifies IPv4/IPv6 multihoming
[128]	2	IP address	Angry IP scanner	Multiple services	Identifies protocols such as SSH and Telnet on a host
[38]	2	IP address	ZGrab	Multiple services	Application scanner with banner grab functionality
[123]	2	IP address	ZMap scan on port tcp/7	Echo server	Identifies servers running ECHO using TCP SYNs
[114]	2	IP address	Zmap scan on port tcp/21	FTP server	
[80]	2	IP address	ZMap scan on port tcp/853	Name server	Discovers name servers that offer DNS-over-TLS
[90]	2	Domain	Custom Java tool	Web server	Tests HTTPS connectivity
[79]	2	IP address	ZMap scan on port tcp/8080	Bytecoin client	
[79]	2	IP address	ZMap scan on port tcp/8333	Bitcoin client	
[2, 47, 106, 115]	2	Domain	ZMap scan on port tcp/443	Web server	
[71]	2	Domain	Censys cert scan	Web server	
[98]	2	IP address	ZMap scan on port 53	Name server	Send DNS A query to verify a port is offering a DNS service
[2]	2	IP address	ZMap UDP scan on ports for IKEv1 and IKEv2	VPN	Scan ports for IPsec IKE protocols to find VPNs
[47, 48]	2	IPv6 address	ZMapv6 scan on tcp/80,443, and udp/53,443	Web server, name server	ZMapv6 is ZMap with IPv6 scanning capabilities
[73]	2	IP address	ZMap for DNP3, Modbus, BACnet, Tridium Fox, and Siemens S7 protocols	Industrial control system	
[8]	2	IP address	ZMap on ports 25, 110, 443, 465, 587, 993, 995	Web server, Mail server	Ports correspond to email related services
[100]	2	IP address	Censys port 443 scan data	Web server	
[104]	2	IP address	ZMap with custom QUIC extension	QUIC services	
[5]	2	IP address	TCP ping for ports used by various protocols	Multiple services	
[84]	2	IP address	ZMap scan; analyze sigs. in fetched banners	Honeypot	Discovers/identifies honeypots; (From IEEE/IFIP INM)
[16]	2	IP address	Search IP in botnet population	*Bot services*	Creates a botnet churn simulator
[29]	2	IP/Dom./URLs	Scans of S3 buckets	S3 buckets	Identifies misconfigured/public Amazon S3 buckets
[88]	2	IP address	Active probes	Service (reverse proxy)	Identifies reverse proxies using TCP fingerprinting
[112]	2	IP/MAC addr.	Layer 2 and 3 timing probes	SDN service	Uses timing attacks to identify SDN services
[34]	1,2	Domain names	Multiple probes to determine hosting	Dom. names/IP addr./srv.	Identifies multiple domains hosted on the same IP
[60]	2,3,4	IP/MAC addr.	nmap/arpscan + HW side-channels	IP/MAC addr.	Identifies HW colocation on public cloud platforms

3.2. Discovering Network Services from Network Identifiers – Edge (2)

The techniques that correspond to this edge can be split up into two principal categories. The first category consists of techniques that typically leverage databases such as (passive) DNS and WHOIS by querying them, either actively or passively. It is one of the most straightforward methods of identifying network services accessible through a network identifier. Its range of network services is generally restricted to the types of records made available through those protocols, such as name servers and mail servers (DNS NS and MX records, respectively). These are well established protocols, and the data obtained from these sources is often standardized or predictable. In the specific case of passive databases, such as passive DNS, the data has been preprocessed, and researchers that use this data benefit from that. They also enable researchers to conduct longitudinal studies due to the large amounts of data that is collected over time. The second category is composed of Internet-wide scanning techniques, which are necessary to address the aforementioned shortcomings. The techniques described in this category utilize existing scanners, or create custom tools or algorithms that identify network services. These techniques are necessary because of the previous category's limited scope in terms of discoverable network services (essentially only web, mail, and name servers). As these techniques are not necessarily restrained by the specifications of existing protocols, they widen the search space and give researchers more freedom to find the network services necessary for their research. This is important for asset discovery, as there exists a plethora of additional network services that can be running on an organization's network, with many more that will be developed in the future. Techniques corresponding to this edge can be found in Table 4. We start by discussing the first category of techniques.

As noted in the previous section, DNS is used extensively in Internet measurement research. It also proves useful in identifying network services. Such techniques correspond to the first category. Many authors use DNS queries to discover network services. Jonker et al. [64] and Dell'Amico et al. [34] query name servers using a certain domain name and extract A, AAAA, and NS records to identify web servers and name servers. Similarly, Foster et al. [46], Durumeric et al. [39], Dell'Amico et al. [34] and Kountouras et al. [69] query for MX records in order to identify mail servers.

Some DNS servers provide recursive name resolution services to any user on the Internet, either on purpose or by accident. Such servers are called *open resolvers*. Liu et al. [78] describe a method that queries Open Resolver Project data [91] using an IP address to discover whether an open resolver is accessible through that address.

Querying WHOIS is also widely employed, but automatic parsing of such data remains an issue due to the lack of format standardization [77]. Liu et al. develop a statistical model for parsing WHOIS data that takes a domain as input, enabling users to automatically extract the addresses corresponding to a queried domain's name server [77]. The authors report an accuracy for parsed fields of over 99% for com domains. For TLDs outside of com, the accuracy decreases, although a specific number is not given. For instance, the model mislabels 16 out of 127 fields in emheartcu.coop's WHOIS record.

The second category will be discussed below, which includes Internet-wide scanners and other custom tools. Most of the papers discussed here involve the tool specifically involve Internet-wide port scanning and use the same scanner to discover network services, namely ZMap [40]. Since its introduction, ZMap has been ubiquitous in Internet measurement research. ZMap is a network scanner specially designed to enable fast Internet-wide scans [40]. The authors highlight three optimizations that allow it to perform such scans at a much faster rate. Firstly, ZMap does not throttle its transmission rate to avoid saturating the scanned or scanning network. Instead, ZMap sends messages as quickly as the network interface card permits. Furthermore, ZMap does not maintain state for each connection to track its scanning progress. Since the goal is to scan random portions of the address space, the scanner avoids storing previously scanned addresses by making use of randomly permuted IP addresses to select targets. It tracks connection timeouts by embedding state information in packet fields. Finally, ZMap opts not to retransmit lost packets. While this results in a 2% loss of network coverage, the authors view this to be an insignificant amount "for typical research applications." [40]

Researchers use this tool to identify network services accessible through domains or IP addresses. Adrian et al. [2], Springall et al. [115], and Scheitle et al. [106] scan port 443 on a host to identify web servers that support HTTPS. Durumeric et al. use ZMap scans on port 25 to find mail servers running SMTP [39]. Adrian et al. also use ZMap's UDP functionality to probe ports for IKEv1 [54] and IKEv2 [67] to find IPsec VPNs. Loe and Quaglia find hosts running Bitcoin and Bytecoin clients by using ZMap to scan ports 8333 and 8080, respectively [79]. Springall et al. also use ZMap to scan port 21 in order to to discover FTP servers [114]. Lu et al. scan port 853 to discover DNS servers that offer DNS-over-TLS [80]. Morishita et al. use ZMap to scan IP addresses on a large collection of ports. From the responses to the scans, they extract the banners and create signatures that allow them to accurately detect and discover different versions of honeypots [84]. Finally, Gasser et al. use ZMapv6 in their study, a variant for scanning IPv6 addresses. They perform TCP scans on ports 80 and 443, as well as UDP scans on ports 53 and 443 to discover web servers (potentially using the QUIC protocol [61]) and DNS servers [48].

While ZMap is used for network service discovery using port scans, it is not able to discover any specifics about the network service. Thus, if such custom functionality is desired, users need to develop a custom ZMap toolchain of their own. For instance, Adrian et al. implemented the SSH protocol in the ZMap toolchain to examine the version and Diffie-Hellman cipher usage in SSH servers, and also added DHE and DHE_EXPORT ciphers to discover HTTPS servers using TLS that support these ciphers [2]. Similarly, Scheitle et al. first use ZMap to discover HTTPS servers on port 443, after which they employ a custom TLS scanner to examine the corresponding TLS certificates. While Springall et al. discover FTP servers using standard ZMap functionality, they needed to implement a custom FTP enumerator to perform a connection and extract information from the server.

Durumeric et al. developed another scanning tool called ZGrab [38], which is an application scanner that scans ports on a host to discover what services are running on said host. At the time of publishing, ZGrab supported scanning of IP addresses in order to identify services such as HTTP, HTTP Proxy, HTTPS, SMTP(S), IMAP(S), POP3(S), FTP, CWMP, SSH, and Modbus by means of protocol handshake initiations. This tool is extensible, meaning that custom protocols can be added if necessary. ZGrab has since been deprecated and replaced by its successor, ZGrab2 [119]. As an improvement over ZGrab, ZGrab2 allows users to scan targets on multiple ports using multiple protocols.

Wullink et al. develop a scanning tool of their own that discovers network services [127]. Their tool, Dmap, takes domain names as input and discovers whether any DNS, HTTP, TLS, or SMTP services are accessible through that domain (or IP corresponding to that domain), using three crawlers. The HTTP crawler attempts to connect the website hosted on the domain, thereby discovering a web server. Apart from A, AAAA, and TXT records, the DNS crawler extracts MX records to discover the mail server corresponding for the input domain. The SMTP crawler attempts to connect to the discovered mail server address through both IPv4 and IPv6 addresses, if available.

Censys [23] and Shodan [110] are search engines that employ such Internet-wide network and application scanners to collect data about devices that are publicly accessible on the Internet. In fact, ZMap itself is the scanner behind Censys [38]. Bijmans et al. use Censys and Shodan scan data on ports 80 and 8080. By querying the data with IP addresses, they discover whether that IP address belongs to a router (specifically the MikroTik brand) with its proxy functionality enabled [13].

Other researchers opt to create custom tools tailored to their own specific needs. Tajalizadehkhoob et al. discover the presence of admin panels on a server by taking a domain and crawling the ports that are usually associated with cPanel, Plesk, DirectAdmin, and Virtualmin [118]. To improve measurement performance, they instructed the crawler to navigate to the URLs that are often used by these admin panels (e.g., /panel/).

The following works use techniques from the two different categories to achieve a goal. Czyz et al. utilize data from Internet-wide DNS ANY queries to extract A and AAAA records corresponding to the same domain name [30]. By probing the obtained IPv4 and IPv6 addresses, they identify the network services accessible through these addresses (e.g., SSH, Telnet, HTTP, among others). The authors found many IPv6 misconfigurations, where services were accessible through the IPv6 address but not the IPv4 address, presumably by accident. The implication of this finding is that you can determine what network services are offered on an IPv4 address by observing a multi-homed IPv6 address.

However, determining whether an IPv4 and IPv6 address pair actually point to the same server is not straightforward. Several authors have developed techniques for making this link. Beverly and Berger developed a TCP-layer fingerprinting approach that probes IPv4/IPv6 address pairs and utilizes TCP Options signatures and TCP timestamp skew to determine whether the address pair points to the same server [11]. Scheitle et al. developed a similar algorithm, but taking more features into account, such as network latency, TTL values, as well as other calculated features [107]. By combining either of these approaches with the findings from Czyz et al. [30], it is possible to discover the purpose of (and, therefore, the network services offered by) an IPv4 address, even though the service is not accessible through the IPv4 address.

3.3. Discovering Network Identifiers from Network Services – Edge (3)

Edge 3 encompasses techniques that leverage information from network services, such as DNS records, to discover network identifiers, such as IP addresses. Notably, this is the only edge that uses a higher level attribute to infer a lower level one, as illustrated in Figure 1. Some of the techniques discussed here are similar to those employed in Edge (1), except that the flow of information could be reversed, or a new flow may result from a misconfigured service. For example, more than simply resolving domain names to IP addresses (Edge (1)), DNS can be used to discover new network identifiers (Edge (3))– a difference we will illustrate below. Edge 3 techniques typically seek to discover *origin* IP addresses (i.e., the real network location of a service situated behind a content-delivery network or other cloud-based service provider), and do so by leveraging potentially misconfigured services and/or their byproducts (e.g., files these services create).

In their comprehensive study of techniques to bypass cloud-based service providers (CBSP), Vissers et al. describe several Edge (3) techniques, as they seek to expose the origin IP address of a CBSP-protected domain. An attacker with knowledge of the origin IP address can in turn completely bypass any CBSP protection, as traffic sent to that address would not be routed through the CBSP security infrastructure [125]. The first technique involves leveraging DNS records associated to services that may be running in the host, such as mail servers. For instance, if a CBSP only forwards HTTP traffic, SMTP will need to establish a direct connection with the mail server, thus leaking the origin IP address. Rather than just performing identifier-to-identifier resolution (domain to IP), a network service (SMTP) here provides us with a new identifier (origin IP). A second method described by both Vissers et al. [125] and Liu et al. [75] uses the zone transfer functionality of (misconfigured) DNS servers (network service) to obtain zone records (network identifiers). This is possible when an attacker pretends to be a secondary DNS server and asks a main DNS server for zone records. Unless the main DNS server has restricted zone transfers, it will oblige and send the records to the attacker.

A third set of techniques rely on potentially specific misconfigured services that may exist in the host that leak the IP addresses through a variety of ways. For instance, Vissers et al. note how "sensitive files" (e.g., verbose error pages or log files, such as `phpinfo()` files) revealed by misconfigured services could cause a leak [125]. Similarly, non-web protocols may also be a concern when improperly handled. Some cloud-based service providers act as a reverse proxy and rely on HTTP `Host` headers to separate requests for different clients. Hence, protocols that, unlike HTTP, do not contain `Host` information (such as SSH) may not be supported. Then, administrators may elect to create subdomains for non-web protocols which directly resolve to the origin IP address – effectively bypassing voluntarily any CBSP protection. A dictionary-based attack of common subdomains could then, easily be used to retrieve the origin IP address [125]. Services that trigger outbound connections can cause similar issues, as the connections may not be routed through the CBSP and leak the origin IP address.

3.4. Discovering Network Services from Network Services – Edge (4)

As described in previous sections, DNS is fundamental to asset discovery. Similar to Edge (3), we will describe several techniques based around DNS server functionality. In fact, most of the techniques in this section are DNS-based. While DNS techniques under Edges 1 and 2 focus on asset discovery through DNS queries (e.g., A/AAAA, MX, NS), the techniques described here involve the inner workings of the DNS protocol and the network service itself. It is often of interest to discover an organization's internal infrastructure of DNS resolvers. Due to its fundamental role, DNS security remains an important topic both in industry and research communities. The other techniques mentioned in this section cannot be grouped into a single category, as they use different methods and discover different assets. We start by describing the DNS-based techniques.

Al-Dalky and Schomp [32] devise a method to discover name servers by leveraging how the servers collaborate during resolution. The authors set up two instrumented hostnames and send two queries to the authoritative name server of their own domain. If the resolver is part of a pool of collaborating name servers, the two queries may arrive at the authoritative name server from two different IP addresses, revealing a previously unknown name server.

Al-Dalky et al. [31] scanned the IPv4 address space to discover DNS resolvers by sending crafted DNS requests for hostnames from their own domain and recording the queries arriving at their experimental authoritative server. When a DNS resolver receives a request with domain names specified, it needs to query name servers unless it is already cached from serving a prior request. By setting up researchers' own name servers, the researchers are able to get the message out of DNS resolvers, allowing them to understand how DNS resolvers work. In this case, they also configured the name servers to only respond to queries with the EDNS0-Client-Subset (ECS) option set to discover DNS resolvers using ECS.

Schomp et al. [108] developed a set of methods to discover client-side DNS infrastructure. Similar to Al-Dalky et al. [31], they also registered their own authoritative domain and deployed their own authoritative DNS to probe the IP address space by sending crafted DNS requests to potential DNS resolvers and receiving the data from DNS resolvers to their name servers. Again, the DNS requests they send out attempt to resolve various hostnames within their own domain, so the DNS recursive resolvers, which are out of their control, will query the

TABLE 5. OUR SYSTEMATIZATION OF PAPERS CONTAINING TECHNIQUES UNDER EDGE 3.

Citation	Edge	Input asset	Discovery technique	Output asset	Remarks
Reverse Information Flow					
[125]	3	Mult. services	Access net. service without explicit host info	IP address	Real IP leaked if a service on a subdomain behind CBSP lacks protocol information
[55]	3	Name server	Query open resolver for `v6only` zone	IPv6 address	Identifies IPv6 addresses of open resolvers
[60]	2,3,4	IP/MAC addr.	Probing (nmap/arpscan), HW side-channels	IP/MAC addr.	Identifies HW colocation on public cloud platforms
Misconfigurations					
[75, 125]	3	Name server	Dictionary attack	Domains	Only the domains in the used dictionary file are discovered
[75, 125]	3	Name server	DNS zone transfer	Domains	Only works if the DNS server has enabled zone transfers
[125]	3	Web server	Trigger outbound connection from web server	IP address	Return traffic not going through CBSP exposes the real host

TABLE 6. OUR SYSTEMATIZATION OF PAPERS CONTAINING TECHNIQUES UNDER EDGE 4.

Citation	Edge	Input asset	Discovery technique	Output asset	Remarks
DNS Based Discovery					
[87]	4	Name server	Authoritative name server discovery technique	Name server	Identifies auth NS from apex and NS records in the child
[31]	4	Name server	DNS query for hostnames under own domain	Name server	Identifies outbound addresses of (open) resolvers
[32]	4	Name server	DNS query for hostnames under own domain	Name server	Identifies open resolvers sharing a cache
[68]	4	Mail server	Send email to inexistent dst. in target domain	Name server	Uses email bounces to identify recursive resolvers of MTAs
[105]	4	Mail/DNS srv.	Send email from domain with auth. NS control	Name server	Identify recursor used by MTA due to triggered SPF check
Other					
[99]	4	Web server	Port scans using server-side requests	Multiple services	
[124]	4	Web server	`Prober` bash script tool	Multiple services	Discovers protocols on a server with ALPN/NPN
[60]	2,3,4	IP/MAC addr.	nmap/arpscan + HW side-channels	IP/MAC addr.	Identifies HW colocation on public cloud platforms

name servers as specified in the original DNS requests, arriving to the name servers the researchers have control of. The difference is that they studied the IP addresses arriving at their authoritative name space to find recursive DNS egress resolvers as well as open DNS ingress servers to gain a deeper understanding of different DNS resolvers.

Klein et al. [68] used a similar probing strategy to Schomp et al., but furthered the device discovery process to find hidden caches used in DNS infrastructure. Apart from directly sending DNS requests to open resolvers, they also used web browsers and mail servers to generate DNS requests. The former was achieved by embedding a script in an ad network page and attaching it to a static URL. The latter involves sending emails to non-existing email addresses in the target domains. To be compliant with the SMTP standard, email servers are required to generate a Delivery Status Notification (DSN, or bounce) message to the originator, and the server has to perform DNS resolution to generate the bounce message. The responses from DNS resolvers sent to researchers' name servers contain the data for discovering DNS services.

Scheffler et al.[105] leverage the functionality of both mail and name servers. Sender Policy Framework (SPF) was developed to combat spam by verifying the identity of senders. In SPF, valid mail tranfer agent (MTA) IP addresses are listed in a TXT record in the organization's domain. The authors set up their own domain and sent out emails as part of their measurement methodology. When the emails arrive at an MTA, they trigger an SPF check. The MTA's resolver then queries your authoritative name server, thereby revealing the IPs of those DNS resolvers.

Naab et al. [87] implement a custom DNS resolver that uses QNAME minimization to discover all of the available authoritative name servers within a queried zone. QNAME minimization [21] was developed to improve privacy by not sending the full original DNS query name (QNAME) in each query. Instead, name servers are only queried for the domain level they are responsible for. Their custom DNS resolver queries all available authoritative name servers for each zone in the domain using three different methods. It extracts name servers from the name in delegation (NS) records and from the IP addresses contained within glue (A) records. Lastly, name servers are extracted from all other NS records in the zone apex.

Pellegrino et al. [99] uses the server-side request functionality of a website running on a server. One can provide to a web server, for instance, a request containing the URL and port that one wants to probe. The web server then performs the server-side request to the provided URL and port. When the web server fails to parse the probed server's response as HTTP, it returns an error message. These error messages can in turn reveal which services are running on the probed server, such as the software running on the probed port or which service is running on the probed port.

Finally, Varvello et al. [124] create a bash script Prober to scan for multiple network services. Prober uses OpenSSL to perform ALPN and NPN negotiations on web servers. It does this to discover numerous protocols that might be announced by the server.

3.5. Systematization Summary

Tables 3 to 6 contain the systematization of the literature performed for this survey. Note, that one paper might occur in multiple lines if it uses multiple techniques, and multiple papers may appear on the same line, if they use the same asset discovery technique. The *Edge* column corresponds to the edge number in Figure 1. Furthermore, the techniques presented in the table follow the systematization syntax described in Section 2.3.

A simple, but interesting, observation is the significant underrepresentation of *network service-to-network identifier* and *network service-to-network service* (Edges (3) and (4)), when compared to the other two types of asset discovery techniques. Much of the surveyed literature deals with Internet measurement. Given its nature, the initiation of this type of research necessitates basic network identifiers, such as IP addresses and domains. This logically leads to an overrepresentation of asset discovery techniques that take a network identifier as input.

The techniques used in Edge (1) can be roughly grouped into three groups: discovery using 1) passive data sources, 2) functionality of existing technologies and infrastructure, and 3) novel algorithms making use of existing technologies. CAIDA prefix-to-AS and RouteViews data seem to be the standard choice for discovering BGP prefixes and ASs. Both passive and active DNS are widely used in academic research.

Edge (2), discovering network services using network identifiers, is the second overrepresented edge. A significant amount of the discovery techniques described in the literature revolve around the network scanner ZMap.

Edges (3) and (4) use network services to discover network identifiers and more network services, respectively. Most of the techniques associated with these two edges involve the usage of flaws in configuration or the technology itself.

4. Applying the Asset Identification Framework to Case Studies

Having mapped the literature onto the various edges of the asset identification framework, we now illustrate its use by applying that framework to several case studies. In the previous section, we extracted the particular edge that researchers used in their work. Here, we construct sequences of edges into paths indicating different asset-discovery processes from beginning to end.

External estimation of enterprise cyber risk A slew of firms, such as Security Scorecard, QuadMetrics, Bitsight, and CyberCube, now offer external assessments of enterprise cyber risks. Their tools provide quantitative scores based on externally observed network characteristics. For these scores to be valid, they first need an accurate asset inventory for the target organizations, which they have to obtain without the organization's cooperation.

While the methods employed by these firms are proprietary, we can glean some insights into their approach by studying the academic papers published by the founders of QuadMetrics [78, 130]. For the bootstrapping phase (Edge (0)), the authors used data from regional Internet registries to identify all the organizations asserting ownership over IP address space. From there, they associate the advertised IP addresses with each organization. They supplement this by searching for the organization's domain name on a search engine (Edge (0)), then resolving the domain to its IP address and identifying the subnet on which the IP address resides (Edge (1)). This in turn identifies more IP addresses belonging to the organization.

Because the researchers' goal is to identify network misconfigurations [19, 36] that could lead to a security breach, they identified network services where misconfigurations could readily be identified. This included identifying open DNS resolvers (Edge (2)) that could be used in DDoS amplification attacks. They also went beyond the identification of services to identifying weaknesses, including BGP misconfigurations and problematic HTTPS certificates, and open SMTP relays.

Take-away: We now have a standardized view into what these researchers did to identify assets. Others carrying on similar efforts could improve upon their findings by utilizing more Edge (1) and (3) actions from the Tables 3 and 5. The authors' choice of Edge (2) actions reflects their underlying goal of identifying security misconfigurations that may be predictive of future breaches, so it is not as clear whether the efforts would benefit from identifying additional network services.

Understanding IPv4/IPv6 security configuration inconsistencies A more research-driven problem is an investigation of configuration inconsistencies between IPv4 and IPv6 in dual-homed hosts. With the global IPv4 address exhaustion, more and more organizations start to deploy IPv6 on their networks. However, this opens the door to simple misconfigurations, where access policies differ between IPv4 and IPv6, as—depending on the platform—firewall configurations have to be configured in other places than for IPv4 [36].

To investigate this issue, we can either start with the exploration of IPv4 or IPv6 addresses of assets for all organizations connected to the Internet (Edge (0)). As soon as we identified IPv4 or IPv6 addresses, we can start "matching," *i.e.*, identifying corresponding IPv4 or IPv6 addresses (Edge (1)). This can be done by resolving the reverse DNS entries of the identified addresses, and subsequently resolving the corresponding A/AAAA RRs of the returned hosts (Edge (1)). As soon as we identified a set of dual homed hosts, we can explore the available services on these hosts (Edge (2)). This service exploration can then be used to further refine our network resources list, *e.g.*, by checking that services running on the identified IPv4 and IPv6 addresses present the same banners (Edge (3)).

Take-away: By using the proposed framework, we can formalize the discovery process of assets. Especially for the procedure of identifying multi-homed hosts, it provides clarity on *what* we want to discover using *which* means. More fundamentally, the asset discovery procedure as outlined in this paper provides a guideline from which researchers can start to design their discovery procedure. The seemingly obvious choice with what to start (IPv4 or IPv6 addresses) becomes more evident as an *and*-choice, not an *either-or*-choice. Furthermore, the option to utilize services to further refine the selection of multi-homed hosts seems obvious when using our framework. Nonetheless, earlier work did not directly utilize this option when conducting such a study, and only focused on address discovery using DNS (Edges (0) and (1)) [30].

Discovering IoT services Since `Nmap` creator Gordon "Fyodor" Lyon scanned "the entire Internet" in 2008 [81], replicating this feat has become commoditized. Nowadays, we have services and tools that literally save days of scanning networks, either by speeding up the process or simply providing the list of open ports as a service. These services have recently specialized in the identification of Internet-enabled devices that are openly accessible, the so-called, Internet of things (IoT). Based on similar principles as `Nmap`, multiple online search engines have emerged providing identification of services running on IoT devices such as Shodan.io, Zoomeye.org and Fofa.so.

The starting discovery technique of these engines is quite simple, *i.e.*, they take IPv4 addresses and identify any service running on that address (Edge (2)). Once all these services are identified they store any metadata related to the open service(s). After the initial exhaustive discovery data storage are completed, these tools allow to search for any Internet device, filtering by date, location, ports, protocols, operating system, and much more. All these different network resources and services are indexed in an online search engine which allows the researcher to discover resources and services over all edges of our framework. Taking Shodan [110] an example, we can map the network resources of any organization by entering its name (Edge (0)). This search will identify a set of network resources that can be used to identify more network resource (Edge (1)) and/or network services (Edge (2)) simply clicking on the identified resources. Shodan also allows searching for a particular network service (e.g., `UPnP`) which will discover additional network resources (Edge (3)) and network services (Edge (4)) that have that port open. Moreover, as Shodan also stores banner and protocol communication information for openly accessible network resources, it allows characterizing network services and thus identifying IoT devices.

Take-away: By scaling up a simple tool introduced in 2008, we can now identify services running in any openly accessible IoT device. Unfortunately the techniques behind these tools do not scale up for IPv6 connected devices. By leveraging IPv6 discovery techniques from Table 3 such as [42] and scan data as collected by Shodan [110], researchers could further identify more services running on IoT devices.

Illicit marketplace forensics Since the early 2010s, illicit online anonymous marketplaces [26, 113] have experienced rapid economic growth. Due to the very nature of the goods being sold (primarily narcotics [26], but with digital goods a sizeable component as well [126]), those markets are constantly under scrutiny and attempts at take-down by law enforcement.

The set of discovery tools described above might seem to be of limited use to this application case, since the markets are hosted on Tor hidden services [37], which conceal IP addresses and related metrics (e.g., BGP AS numbers); by the same token, no DNS records are available, if these servers are properly configured. However, the hidden service protocol involves the registration of a `.onion` pseudo-DNS name with Tor's directory servers. Similar to what happens with DNS, some operators elect to have vanity addresses—the now defunct Silk Road market was notoriously hosted at silkroad6ownowfk.onion. As such, the `.onion` address, like a DNS name, becomes a network identifier that might be enumerated. Some online aggregators such as dark.fail provide a list of verified links (primarily to prevent phishing scams) as a form of directory for some `.onion` services (Edge (0)). Along the same lines, even Wikipedia contains sometimes links to these services (*e.g.*, the former Silk Road address is mentioned in the relevant article). In addition, researchers have shown that it was possible to exhaustively list all onion services by passively listening to announcements [15] (Edge (1)). While this specific problem has long been fixed, it is worth noting that `.onion` names, much like domain names are discoverable network identifiers.

In addition, misconfiguring a hidden service frequently leaks information that can be used to identify traditional network identifiers and services (Edges (3) and (4)). For instance, the notorious AlphaBay marketplace briefly leaked an email address belonging to its operator—ultimately facilitating the marketplace take-down and the operator's arrest [65]. While AlphaBay is an extreme example of a data leak, it is not particularly unique—Silk Road [26] also fell victim to a similar mishap. Setting aside such egregious mistakes, tools like CARONTE [82] automate the exploitation of information leaks to attempt to deanonymize hidden services by revealing their IP addresses (Edge (1)).

More generally, much like their legitimate counterparts, modern online anonymous marketplaces rely on several servers, for load balancing traffic, denial-of-service resilience, and decoupling of basic functions (e.g., backend database vs. "hot wallet" server hosting cryptocurrency holdings). Using the type of asset discovery techniques earlier discussed, one of these servers leaking information (*e.g.*, an IP address) could point investigators to other servers of interest. This is what seems to have happened with Silk Road, though details in the criminal complaint [122] are scarce. For instance, even though no public details about the investigative techniques used in Operation Onymous [121] were shared, the mere fact that a police operation succeeded in taking down multiple marketplaces at the same time suggests that successful asset discovery took place. Because the take-down was incomplete (leading markets such as Evolution survived this police operation), we can only speculate that the markets that were identified might have shared part of their infrastructure, which turned out to be vulnerable. For instance, they might have had some portions of their services hosted on the same vulnerable platforms.

More generally, a hidden service could leak information by its mere existence. Indeed, assume that you run a virtual private server, and notice that one of your hosted machines only connects to a single IP address, which happens to be a Tor node, and that all traffic patterns resemble typical Tor patterns: this is pretty clear evidence that a Tor hidden service is running on the hosted machine. Subsequently, an adversary might use this information to discover *which* hidden service is running on the machine. Historically, this could be done by injecting a large number of nodes in the Tor network and hope the hidden service eventually picks one of the adversarial nodes as a "guard" (the first connection in the Tor circuit used by the hidden service to talk to the rest of the world), thereby revealing its IP address to the adversary [92] (Edge (4)). While this issue has mostly been resolved by picking long-term guards, more recently, novel passive attacks against hidden services [72] have been proposed. The short story is that maintaining anonymity is extremely difficult, and subject to similar techniques.

Take-away: Even in cases where assets purposely attempt to avoid detection, the techniques described above make it possible for an investigator to obtain considerable additional information from limited information leaks.

5. Discussion and Concluding Remarks

Asset discovery is typically not the main focus of network measurement research. As a result, it often does get the attention it deserves, since enumerating a population of assets involves many trade-offs. It would be far better to explicitly consider the choices and leverage novel techniques proposed by others. However, in practice, an understandable focus on primary measurement tasks, and the lack of a consistent framework to "plug together" asset discovery techniques, often result in an incomplete or ad hoc asset discovery process.

To address these issues, we present a framework for asset discovery. Our framework proposes a syntax to make explicit the steps in the asset-discovery process. This in turn provides a natural way to identify gaps in study design, thereby creating opportunities to build on earlier efforts by broadening the set of assets discovered. Furthermore, our systematization of recent advances in active asset discovery can help researchers select relevant techniques. Finally, we apply our framework to various use cases, which illustrates how techniques can be combined and how to identify where gaps remain.

Looking at the past five years of newly developed asset discovery techniques, the introduction of ZMap in 2013 had a long-lasting impact on the asset discovery process by enabling exhaustive scans of the IPv4 address space. However, the continued migration toward IPv6 has prompted research on IPv6 space scanning, which cannot be done exhaustively. Despite preliminary progress, this remains an area of future research.

DNS emerges as a central tool for discovering assets, especially network services. Interestingly, this appears to be a fairly recent trend, as the academic research community has seemingly overlooked DNS as an asset discovery tool for many years. Similarly, we find DNS helpful in locating IPv6 network resources. We therefore conclude that DNS-related asset discovery techniques could become more prevalent, as more corner cases and niche features of DNS are explored.

For topology-related asset discovery, the CAIDA prefix-to-AS and the RouteViews data have become the *de facto* standard across several recent publications. Nonetheless, these datasets also highlight that asset discovery is usually not the main goal. Instead, we find that researchers regularly create datasets and techniques to model how the Internet works, and not to discover assets in the cases we outline, such as red team use. Similarly, we find a large body of work focusing on different forms of misconfigurations [36, 43] and how to discover them on the Internet, without having the operational aspect of computer security in mind. This, again, highlights the value of our classification and formalization we provide in this work, as it enables researchers to assess those techniques' value for being used in asset discovery.

Our framework can also foster new approaches to asset discovery, by re-assessing and re-positioning established techniques in a new context. For instance, the discussion of illicit marketplace exploration in Section 4 shows how our framework can help identify functional analogies and similarities between *a priori* different protocols. Service discovery techniques for a given protocol may then be successfully repurposed for a different protocol. In turn, our framework can help implement asset discovery for concealed network services, *e.g.*, via CBSP or even Tor.

Additionally, this paper sheds light on some causes behind the replication challenges present in cybersecurity and network measurement research. A natural explanation for why it is hard to replicate network measurement results is that we are studying a dynamic network whose configurations regularly change (*e.g.*, IP address churn through DHCP, changing BGP routes). While these are real impediments, inconsistent use of different combinations of techniques can yield different asset compositions for investigation, which hampers replication. While not eliminating these problems, our framework does make such differences explicit, which may point to a solution.

Besides its immediate significance, we believe our work will remain relevant and valuable for researchers in the future. While protocols evolve, *e.g.*, DNS over HTTPS (DoH), the underlying concepts—connections, state, and identifiers—remain the same as they were 40 years ago, when TCP/IP was first introduced. Consequently, the techniques we describe will continue to be usable in research for some time. Furthermore, many of the techniques are not explicitly advertised as asset discovery techniques. Having them described and referenced in our paper might help keep future researchers from having to reinvent the wheel. The framework itself can be continuously extended as new techniques emerge in scientific literature. When such new techniques arise, researchers can map them onto the framework using the proposed syntax and combine them into tool-chains needed for conducting their own research. For example, we have found a lack of IoT discovery techniques in the scientific literature, even though there are more IoT connections to the Internet than non-IoT connections [116]. Additionally, we noted a dearth of research focusing on discovery of network services running on non-standard ports, an important but overlooked subject.

The proposed syntax and framework is very general and abstract enough to fit changes in networking. Even networks that do not operate on TCP/IP communicate through network identifiers and network services. Nonetheless, major changes in networking and protocols are entirely possible (e.g. New IP [1], LoRaWAN [57]). If a significant change occurs in the future such that networks will behave differently than those of today, the techniques mentioned this systematization will become less relevant. In that case, however, our framework can still incorporate the changes, as the concept of communication does not change: network services will always have to make identifiers accessible to users.

Acknowledgments. This research was partially supported by the Air Force Research Laboratory (AFRL) under agreement number FA8750-19-1-0152, and by the European Commission through the H2020 program in projects CyberSecurity4Europe (Grant No. #830929), and Safe-DEED (Grant No. #825225). The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright notation thereon. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of AFRL or the U.S. Government.

References

[1] *A Brief Introduction about New IP Research Initiative*. URL: https://www.huawei.com/en/industry-insights/innovation/new-ip (visited on 02/18/2021).

[2] David Adrian, Karthikeyan Bhargavan, Zakir Durumeric, Pierrick Gaudry, Matthew Green, J. Alex Halderman, Nadia Heninger, Drew Springall, Emmanuel Thomé, Luke Valenta, Benjamin VanderSloot, Eric Wustrow, Santiago Zanella-Béguelin, and Paul Zimmermann. "Imperfect Forward Secrecy: How Diffie-Hellman Fails in Practice". In: *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security (CCS)*. 2015.

[3] Areej Albataineh and Izzat Alsmadi. "IoT and the Risk of Internet Exposure: Risk Assessment Using Shodan Queries". In: *Proceedings of the 20th IEEE International Symposium on "A World of Wireless, Mobile and Multimedia Networks" (WoWMoM)*. 2019.

[4] Alliantist Ltd. *ISO 27001 Annex A.8 - Asset Management*. en-GB. June 2018. URL: https://www.isms.online/iso-27001/annex-a-8-asset-management/ (visited on 06/10/2020).

[5] Omar Alrawi, Chaoshun Zuo, Ruian Duan, Ranjita Pai Kasturi, Zhiqiang Lin, and Brendan Saltaformaggio. "The Betrayal At Cloud City: An Empirical Analysis Of Cloud-Based Mobile Backends". In: *Proceedings of the 28th USENIX Security Symposium (USENIX Security)*. 2019. URL: https://www.usenix.org/conference/usenixsecurity19/presentation/alrawi.

[6] S. Alrwais, X. Liao, X. Mi, P. Wang, X. Wang, F. Qian, R. Beyah, and D. McCoy. "Under the Shadow of Sunshine: Understanding and Detecting Bulletproof Hosting on Legitimate Service Provider Networks". In: *Proceedings of the 38th IEEE Symposium on Security & Privacy (S&P)*. 2017.

[7] CAIDA: Center for Applied Internet Data Analysis. *Routeviews Prefix to AS Mappings Dataset (Pfx2as) for IPv4 and IPv6*. URL: https://www.caida.org/data/routing/routeviews-prefix2as.xml (visited on 06/23/2020).

[8] Nimrod Aviram, Sebastian Schinzel, Juraj Somorovsky, Nadia Heninger, Maik Dankel, Jens Steube, Luke Valenta, David Adrian, J. Alex Halderman, Viktor Dukhovni, Emilia Käsper, Shaanan Cohney, Susanne Engels, Christof Paar, and Yuval Shavitt. "{DROWN}: Breaking {TLS} Using SSLv2". In: *Proceedings of the 25th USENIX Security Symposium (USENIX Security)*. 2016. URL: https://www.usenix.org/conference/usenixsecurity16/technical-sessions/presentation/aviram.

[9] Mohammed Bashir and Nicolas Christin. "Three case studies in quantitative information risk analysis". In: *Proceedings of the 2008 CERT/SEI Making the Business Case for Software Assurance Workshop*. Sept. 2008. URL: https://www.andrew.cmu.edu/user/nicolasc/publications/ash.pdf.

[10] Karyn Benson, Alberto Dainotti, kc claffy, Alex C. Snoeren, and Michael Kallitsis. "Leveraging Internet Background Radiation for Opportunistic Network Analysis". In: *Proceedings of the 2015 Internet Measurement Conference (IMC)*. 2015.

[11] Robert Beverly and Arthur Berger. "Server Siblings: Identifying Shared IPv4/IPv6 Infrastructure Via Active Fingerprinting". In: *Proceedings of the 10th Passive and Active Measurement (PAM)*. 2015.

[12] Robert Beverly, Ramakrishnan Durairajan, David Plonka, and Justin P Rohrer. "In the IP of the Beholder: Strategies for Active IPv6 Topology Discovery". In: *Proceedings of the 2018 Internet Measurement Conference (IMC)*. 2018.

[13] Hugo L. J. Bijmans, Tim M. Booij, and Christian Doerr. "Just the Tip of the Iceberg: Internet-Scale Exploitation of Routers for Cryptojacking". In: *Proceedings of the 26th ACM SIGSAC Conference on Computer and Communications Security (CCS)*. 2019.

[14] Henry Birge-Lee, Yixin Sun, Anne Edmundson, Jennifer Rexford, and Prateek Mittal. "Bamboozling Certificate Authorities with {BGP}". In: *Proceedings of the 27th USENIX Security Symposium (USENIX Security)*. 2018. URL: https://www.usenix.org/conference/usenixsecurity18/presentation/birge-lee.

[15] Alex Biryukov, Ivan Pustogarov, and Ralf-Philipp Weinmann. "Trawling for Tor hidden services: Detection, measurement, deanonymization". In: *Proceedings of the 34th IEEE Symposium on Security & Privacy (S&P)*. 2013.

[16] Leon Böck, Emmanouil Vasilomanolakis, Max Mühlhäuser, and Shankar Karuppayah. "Next Generation P2P Botnets: Monitoring Under Adverse Conditions". In: *Proceedings of the 21st International Symposium on Recent Advances in Intrusion Detection (RAID)*. 2019.

[17] Roland Bodenheim, Jonathan Butts, Stephen Dunlap, and Barry Mullins. en. In: *International Journal of Critical Infrastructure Protection* 7.2 (June 2014). DOI: 10.1016/j.ijcip.2014.03.001. URL: http://www.sciencedirect.com/science/article/pii/S1874548214000213 (visited on 04/06/2020).

[18] Anthony J. Bonkoski, Russ Bielawski, and J. Alex Halderman. "Illuminating the Security Issues Surrounding Lights-Out Server Management". In: *7th USENIX Workshop on Offensive Technologies*. 2013. URL: https://www.usenix.org/conference/woot13.

[19] Kevin Borgolte, Tobias Fiebig, Shuang Hao, Christopher Kruegel, and Giovanni Vigna. "Cloud Strife: Mitigating the Security Risks of Domain-Validated Certificates". In: *Proceedings of the 25th Network and Distributed System Security Symposium (NDSS)*. 2018.

[20] Kevin Borgolte, Shuang Hao, Tobias Fiebig, and Giovanni Vigna. "Enumerating Active IPv6 Hosts for Large-Scale Security Scans via DNSSEC-Signed Reverse Zones". In: *Proceedings of the 39th IEEE Symposium on Security & Privacy (S&P)*. 2018.

[21] S. Bortzmeyer. *DNS Query Name Minimisation to Improve Privacy*. en. RFC 7816. RFC Editor, Mar. 2016. DOI: 10.17487/RFC7816. URL: https://www.rfc-editor.org/info/rfc7816 (visited on 06/25/2020).

[22] Frank Cangialosi, Taejoong Chung, David Choffnes, Dave Levin, Bruce M. Maggs, Alan Mislove, and Christo Wilson. "Measurement and Analysis of Private Key Sharing in the HTTPS Ecosystem". In: *Proceedings of the 23rd ACM SIGSAC Conference on Computer and Communications Security (CCS)*. 2016.

[23] *Censys*. URL: http://censys.io/ (visited on 06/02/2020).

[24] D. Chiba, T. Yagi, M. Akiyama, T. Shibahara, T. Yada, T. Mori, and S. Goto. "DomainProfiler: Discovering Domain Names Abused in Future". In: *Proceedings of the 46th IEEE/IFIP International Conference on Dependable Systems and Networks (DSN)*. 2016.

[25] Daiki Chiba, Ayako Akiyama Hasegawa, Takashi Koide, Yuta Sawabe, Shigeki Goto, and Mitsuaki Akiyama. "DomainScouter: Understanding the Risks of Deceptive IDNs". In: *Proceedings of the 22nd International Symposium on Recent Advances in Intrusion Detection (RAID)*. 2019. URL: https://www.usenix.org/conference/raid2019/presentation/chiba.

[26] Nicolas Christin. "Traveling the Silk Road: A measurement analysis of a large anonymous online marketplace". In: *Proceedings of the 22nd World Wide Web Conference (WWW)*. 2013.

[27] Taejoong Chung, David Choffnes, and Alan Mislove. "Tunneling for Transparency: A Large-Scale Analysis of End-to-End Violations in the Internet". In: *Proceedings of the 2016 Internet Measurement Conference (IMC)*. 2016.

[28] Taejoong Chung, Yabing Liu, David Choffnes, Dave Levin, Bruce MacDowell Maggs, Alan Mislove, and Christo Wilson. "Measuring and Applying Invalid SSL Certificates: The Silent Majority". In: *Proceedings of the 2016 Internet Measurement Conference*. Nov. 2016. URL: https://doi.org/10.1145/2987443.2987454 (visited on 06/01/2020).

[29] A. Continella, M. Polino, M. Pogliani, and S. Zanero. "There's a Hole in that Bucket! A Large-scale Analysis of Misconfigured S3 Buckets". In: *Proceedings of the 34th Annual Computer Security Applications Conference (ACSAC)*. 2018.

[30] Jakub Czyz, Matthew Luckie, Mark Allman, and Michael Bailey. "Don't Forget to Lock the Back Door! A Characterization of IPv6 Network Security Policy". In: *Proceedings of the 23rd Network and Distributed System Security Symposium (NDSS)*. 2016.

[31] Rami Al-Dalky, Michael Rabinovich, and Kyle Schomp. "A Look at the ECS Behavior of DNS Resolvers". In: *Proceedings of the 2019 Internet Measurement Conference (IMC)*. 2019.

[32] Rami Al-Dalky and Kyle Schomp. "Characterization of Collaborative Resolution in Recursive DNS Resolvers". In: *Proceedings of the 13th Passive and Active Measurement (PAM)*. 2018.

[33] Wouter B de Vries, Roland van Rijswijk-Deij, Pieter-Tjerk de Boer, and Aiko Pras. "Passive Observations of a Large DNS Service: 2.5 Years in the Life of Google". In: *Proceedings of the 2018 International Workshop on Traffic Monitoring and Analysis (TMA)*. 2018.

[34] Matteo Dell'Amico, Leyla Bilge, Ashwin Kayyoor, Petros Efstathopoulos, and Pierre-Antoine Vervier. "Lean On Me: Mining Internet Service Dependencies From Large-Scale DNS Data". In: *Proceedings of the 33rd Annual Computer Security Applications Conference (ACSAC)*. 2017.

[35] Amogh Dhamdhere, David D. Clark, Alexander Gamero-Garrido, Matthew Luckie, Ricky K. P. Mok, Gautam Akiwate, Kabir Gogia, Vaibhav Bajpai, Alex C. Snoeren, and Kc Claffy. "Inferring Persistent Interdomain Congestion". In: *Proceedings of the 2018 ACM SIGCOMM Conference (SIGCOMM)*. 2018.

[36] Constanze Dietrich, Katharina Krombholz, Kevin Borgolte, and Tobias Fiebig. "Investigating System Operators' Perspective on Security Misconfigurations". In: *Proceedings of the 25th ACM SIGSAC Conference on Computer and Communications Security (CCS)*. 2018.

[37] Roger Dingledine, Nick Mathewson, and Paul Syverson. "Tor: The Second-Generation Onion Router". In: *Proceedings of the 13th USENIX Security Symposium (USENIX Security)*. 2014.

[38] Zakir Durumeric, David Adrian, Ariana Mirian, Michael Bailey, and J. Alex Halderman. "A Search Engine Backed by Internet-Wide Scanning". In: *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security (CCS)*. 2015.

[39] Zakir Durumeric, J. Alex Halderman, David Adrian, Ariana Mirian, James Kasten, Elie Bursztein, Nicolas Lidzborski, Kurt Thomas, Vijay Eranti, and Michael Bailey. "Neither Snow Nor Rain Nor MITM...: An Empirical Analysis of Email Delivery Security". In: *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security (CCS)*. 2015.

[40] Zakir Durumeric, Eric Wustrow, and J Alex Halderman. "ZMap: Fast Internet-Wide Scanning and Its Security Applications ZMap: Fast Internet-Wide Scanning and Its Security Applications". In: *Proceedings of the 22nd USENIX Security Symposium (USENIX Security)*. 2013.

[41] Tobias Fiebig, Kevin Borgolte, Shuang Hao, Christopher Kruegel, and Giovanni Vigna. "Something from Nothing (There): Collecting Global IPv6 Datasets from DNS". In: *Proceedings of the 12th Passive and Active Measurement (PAM)*. 2017.

[42] Tobias Fiebig, Kevin Borgolte, Shuang Hao, Christopher Kruegel, Giovanni Vigna, and Anja Feldmann. "In rDNS We Trust: Revisiting a Common Data-Source's Reliability". In: *Proceedings of the 13th Passive and Active Measurement (PAM)*. 2018.

[43] Tobias Fiebig, Anja Feldmann, and Matthias Petschick. "A one-year perspective on exposed in-memory key-value stores". In: *Proceedings of the 2016 ACM Workshop on Automated Decision Making for Active Cyber Defense*. 2016.

[44] Pawel Foremski, Oliver Gasser, and Giovane C. M. Moura. "DNS Observatory: The Big Picture of the DNS". In: *Proceedings of the 2019 Internet Measurement Conference (IMC)*. 2019.

[45] Pawel Foremski, David Plonka, and Arthur Berger. "Entropy/IP: Uncovering Structure in IPv6 Addresses". In: *Proceedings of the 2016 Internet Measurement Conference (IMC)*. 2016.

[46] Ian D. Foster, Jon Larson, Max Masich, Alex C. Snoeren, Stefan Savage, and Kirill Levchenko. "Security by Any Other Name: On the Effectiveness of Provider Based Email Security". In: *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security (CCS)*. 2015.

[47] Oliver Gasser, Benjamin Hof, Max Helm, Maciej Korczynski, Ralph Holz, and Georg Carle. "In Log We Trust: Revealing Poor Security Practices with Certificate Transparency Logs and Internet Measurements". In: *Proceedings of the 13th Passive and Active Measurement (PAM)*. 2018.

[48] Oliver Gasser, Quirin Scheitle, Pawel Foremski, Qasim Lone, Maciej Korczyński, Stephen D. Strowes, Luuk Hendriks, and Georg Carle. "Clusters in the Expanse: Understanding and Unbiasing IPv6 Hitlists". In: *Proceedings of the 2018 Internet Measurement Conference (IMC)*. 2018.

[49] Oliver Gasser, Quirin Scheitle, Sebastian Gebhard, and Georg Carle. "Scanning the IPv6 Internet: Towards a Comprehensive Hitlist". In: *Proceedings of the 2016 International Workshop on Traffic Monitoring and Analysis (TMA)*. 2016. URL: http://dl.ifip.org/db/conf/tma/tma2016/tma2016-final51.pdf.

[50] Manaf Gharaibeh, Anant Shah, Bradley Huffaker, Han Zhang, Roya Ensafi, and Christos Papadopoulos. "A Look at Router Geolocation in Public and Commercial Databases". In: *Proceedings of the 2017 Internet Measurement Conference (IMC)*. 2017.

[51] Jean-François Grailet and Benoit Donnet. "Revisiting Subnet Inference WISE-Ly". In: 2019. URL: https://github.com/JefGrailet/WISE (visited on 06/28/2019).

[52] Jean-Franois Grailet, Fabien Tarissan, and Benoit Donnet. "Passive Observations of a Large DNS Service: 2.5 Years in the Life of Google". In: *Proceedings of the 2016 International Workshop on Traffic Monitoring and Analysis (TMA)*. 2016. URL: http://dl.ifip.org/db/conf/tma/tma2016/tma2016-final12.pdf.

[53] Shuai Hao, Yubao Zhang, Haining Wang, and Angelos Stavrou. "End-Users Get Maneuvered: Empirical Analysis of Redirection Hijacking in Content Delivery Networks". In: *Proceedings of the 27th USENIX Security Symposium (USENIX Security)*. 2018. URL: https://www.usenix.org/conference/usenixsecurity18/presentation/hao.

[54] D. Harkins and D. Carrel. *The Internet Key Exchange (IKE)*. en. RFC 2409. RFC Editor, Nov. 1998. DOI: 10.17487/rfc2409. URL: https://www.rfc-editor.org/info/rfc2409 (visited on 06/10/2020).

[55] Luuk Hendriks, Ricardo de Oliveira Schmidt, Roland van Rijswijk-Deij, and Aiko Pras. "On the Potential of IPv6 Open Resolvers for DDoS Attacks". In: *Proceedings of the 12th Passive and Active Measurement (PAM)*. 2017.

[56] T. Hlavacek, A. Herzberg, H. Shulman, and M. Waidner. "Practical Experience: Methodologies for Measuring Route Origin Validation". In: *Proceedings of the 48th IEEE/IFIP International Conference on Dependable Systems and Networks (DSN)*. 2018.

[57] *Homepage - LoRa Allicance*. URL: https://lora-alliance.org/ (visited on 02/18/2021).

[58] Hang Hu and Gang Wang. "End-to-End Measurements of Email Spoofing Attacks". In: *Proceedings of the 27th USENIX Security Symposium (USENIX Security)*. 2018. URL: https://www.usenix.org/conference/usenixsecurity18/presentation/hu.

[59] Qinwen Hu, Muhammad Rizwan Asghar, and Nevil Brownlee. "Measuring IPv6 DNS Reconnaissance Attacks and Preventing Them Using DNS Guard". In: *Proceedings of the 48th IEEE/IFIP International Conference on Dependable Systems and Networks (DSN)*. 2018.

[60] M. Inci, G. Irazoqui, T. Eisenbarth, and B. Sunar. "Efficient, Adversarial Neighbor Discovery using Logical Channels on Microsoft Azure". In: *Proceedings of the 32nd Annual Computer Security Applications Conference (ACSAC)*. 2016.

[61] Jim Roskind. *QUIC: Design Document and Specification Rationale*. en. Apr. 2012. URL: https://docs.google.com/document/d/1RNHkx_VvKWyWg6Lr8SZ-saqsQx7rFV-ev2jRFUoVD34/ (visited on 06/02/2020).

[62] L. Jin, S. Hao, H. Wang, and C. Cotton. "Your Remnant Tells Secret: Residual Resolution in DDoS Protection Services". In: *Proceedings of the 48th IEEE/IFIP International Conference on Dependable Systems and Networks (DSN)*. 2018.

[63] Mattijs Jonker, Alistair King, Johannes Krupp, Christian Rossow, Anna Sperotto, and Alberto Dainotti. "Millions of Targets under Attack: A Macroscopic Characterization of the DoS Ecosystem". In: *Proceedings of the 2017 Internet Measurement Conference (IMC)*. 2017.

[64] Mattijs Jonker, Anna Sperotto, Roland van Rijswijk-Deij, Ramin Sadre, and Aiko Pras. "Measuring the Adoption of DDoS Protection Services". In: *Proceedings of the 2016 Internet Measurement Conference (IMC)*. 2016.

[65] U.S. Department of Justice. *US vs. Cazes, Verified Complaint for Forfeiture in Rem*. https://www.justice.gov/opa/press-release/file/982821/download. July 2017.

[66] Andrew J. Kaizer and Minaxi Gupta. "∼Open Resolvers: Understanding the Origins of Anomalous Open DNS Resolvers". In: *Proceedings of the 10th Passive and Active Measurement (PAM)*. 2015.

[67] Kaufman, C. *Internet Key Exchange (IKEv2) Protocol*. RFC 4306. RFC Editor, Dec. 2005. DOI: 10.17487/RFC4306. URL: https://www.rfc-editor.org/info/rfc4306 (visited on 06/10/2020).

[68] Amit Klein, Haya Shulman, and Michael Waidner. "Counting in the Dark: DNS Caches Discovery and Enumeration in the Internet". In: *Proceedings of the 47th IEEE/IFIP International Conference on Dependable Systems and Networks (DSN)*. 2017.

[69] Athanasios Kountouras, Panagiotis Kintis, Chaz Lever, Yizheng Chen, Yacin Nadji, David Dagon, Manos Antonakakis, and Rodney Joffe. "Enabling Network Security Through Active DNS Datasets". In: *Proceedings of the 19th International Symposium on Recent Advances in Intrusion Detection (RAID)*. 2016.

[70] Thomas Krenc and Anja Feldmann. “BGP Prefix Delegations: A Deep Dive”. In: *Proceedings of the 2016 Internet Measurement Conference (IMC)*. 2016.

[71] D. Kumar, Z. Wang, M. Hyder, J. Dickinson, G. Beck, D. Adrian, J. Mason, Z. Durumeric, J. A. Halderman, and M. Bailey. “Tracking Certificate Misissuance in the Wild”. In: *Proceedings of the 39th IEEE Symposium on Security & Privacy (S&P)*. 2018.

[72] Albert Kwon, Mashael AlSabah, David Lazar, Marc Dacier, and Srinivas Devadas. “Circuit Fingerprinting Attacks: Passive Deanonymization of Tor Hidden Services”. In: *Proceedings of the 24th USENIX Security Symposium (USENIX Security)*. 2015.

[73] Frank Li, Zakir Durumeric, Jakub Czyz, Mohammad Karami, Michael Bailey, Damon McCoy, Stefan Savage, and Vern Paxson. “You’ve Got Vulnerability: Exploring Effective Vulnerability Notifications”. In: *Proceedings of the 25th USENIX Security Symposium (USENIX Security)*. 2016. URL: https://www.usenix.org/conference/usenixsecurity16/technical-sessions/presentation/li.

[74] B. Liu, C. Lu, Z. Li, Y. Liu, H. Duan, S. Hao, and Z. Zhang. “A Reexamination of Internationalized Domain Names: The Good, the Bad and the Ugly”. In: *Proceedings of the 48th IEEE/IFIP International Conference on Dependable Systems and Networks (DSN)*. 2018.

[75] Daiping Liu, Shuai Hao, and Haining Wang. “All Your DNS Records Point to Us: Understanding the Security Threats of Dangling DNS Records”. In: *Proceedings of the 23rd ACM SIGSAC Conference on Computer and Communications Security (CCS)*. 2016.

[76] Daiping Liu, Zhou Li, Kun Du, Haining Wang, Baojun Liu, and Haixin Duan. “Don’t Let One Rotten Apple Spoil the Whole Barrel: Towards Automated Detection of Shadowed Domains”. In: *Proceedings of the 24th ACM SIGSAC Conference on Computer and Communications Security (CCS)*. 2017.

[77] Suqi Liu, Ian Foster, Stefan Savage, Geoffrey M. Voelker, and Lawrence K. Saul. “Who Is .Com? Learning to Parse WHOIS Records”. In: *Proceedings of the 2015 Internet Measurement Conference (IMC)*. 2015.

[78] Yang Liu, Armin Sarabi, Jing Zhang, Parinaz Naghizadeh, Manish Karir, Michael Bailey, and Mingyan Liu. “Cloudy with a Chance of Breach: Forecasting Cyber Security Incidents”. In: *Proceedings of the 24th USENIX Security Symposium (USENIX Security)*. 2015. URL: https://www.usenix.org/conference/usenixsecurity15/technical-sessions/presentation/liu.

[79] Angelique Faye Loe and Elizabeth Anne Quaglia. “You Shall Not Join: A Measurement Study of Cryptocurrency Peer-to-Peer Bootstrapping Techniques”. In: *Proceedings of the 26th ACM SIGSAC Conference on Computer and Communications Security (CCS)*. 2019.

[80] Chaoyi Lu, Baojun Liu, Zhou Li, Shuang Hao, Haixin Duan, Mingming Zhang, Chunying Leng, Ying Liu, Zaifeng Zhang, and Jianping Wu. “An End-to-End, Large-Scale Measurement of DNS-over-Encryption: How Far Have We Come?” In: *Proceedings of the 2019 Internet Measurement Conference (IMC)*. 2019.

[81] Gordon Fyodor Lyon. *Nmap network scanning: The official Nmap project guide to network discovery and security scanning*. Insecure, 2009.

[82] Srdjan Matic, Platon Kotzias, and Juan Caballero. “CARONTE: Detecting Location Leaks for Deanonymizing Tor Hidden Services”. In: *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security (CCS)*. 2015.

[83] Nora McDonald, Sarita Schoenebeck, and Andrea Forte. “Reliability and Inter-rater Reliability in Qualitative Research: Norms and Guidelines for CSCW and HCI Practice”. In: *Proceedings of the 2019 ACM CHI Conference on Human Factors in Computing Systems (CHI)*. 2019.

[84] Shun Morishita, Takuya Hoizumi, Wataru Ueno, Rui Tanabe, Carlos Gañán, Michel J.G. van Eeten, Katsunari Yoshioka, and Tsutomu Matsumoto. “Detect Me If You... Oh Wait. An Internet-Wide View of Self-Revealing Honeypots”. In: *Proceedings of the 2019 IFIP/IEEE Symposium on Integrated Network Management (IM)*. 2019. URL: https://ieeexplore.ieee.org/document/8717918.

[85] G. C. M. Moura, M. Müller, M. Wullink, and C. Hesselman. “nDEWS: A New Domains Early Warning System for TLDs”. In: *Proceedings of the2016IEEE Network Operations and Management Symposium (NOMS)*. 2016.

[86] Austin Murdock, Frank Li, Paul Bramsen, Zakir Durumeric, and Vern Paxson. “Target Generation for Internet-Wide IPv6 Scanning”. In: *Proceedings of the 2017 Internet Measurement Conference (IMC)*. 2017.

[87] Johannes Naab, Patrick Sattler, Jonas Jelten, Oliver Gasser, and Georg Carle. “Prefix Top Lists: Gaining Insights with Prefixes from Domain-Based Top Lists on DNS Deployment”. In: *Proceedings of the 2019 Internet Measurement Conference (IMC)*. 2019.

[88] A. Nappa, R. Munir, I. Tanoli, C. Kreibich, and J. Caballero. “RevProbe: Detecting Silent Reverse Proxies in Malicious Server Infrastructures”. In: *Proceedings of the 32nd Annual Computer Security Applications Conference (ACSAC)*. 2016.

[89] National Institute of Standards and Technology. *Framework for Improving Critical Infrastructure Cybersecurity, Version 1.1*. en. Tech. rep. NIST CSWP 04162018. National Institute of Standards and Technology, Apr. 2018. DOI: 10.6028/NIST.CSWP.04162018. URL: http://nvlpubs.nist.gov/nistpubs/CSWP/NIST.CSWP.04162018.pdf (visited on 04/06/2020).

[90] Carl Nykvist, Linus Sjöström, Josef Gustafsson, and Niklas Carlsson. “Server-Side Adoption of Certificate Transparency”. In: *Proceedings of the 13th Passive and Active Measurement (PAM)*. 2018.

[91] *Open Resolver Project*. URL: http://openresolverproject.org/.

[92] Lars Øverlier and Paul Syverson. “Locating Hidden Servers”. In: *Proceedings of the 27th IEEE Symposium on Security & Privacy (S&P)*. 2006.

[93] Ramakrishna Padmanabhan, Amogh Dhamdhere, Emile Aben, kc claffy, and Neil Spring. “Reasons Dynamic Addresses Change”. In: *Proceedings of the 2016 Internet Measurement Conference (IMC)*. 2016.

[94] Ramakrishna Padmanabhan, Zhihao Li, Dave Levin, and Neil Spring. “UAv6: Alias Resolution in IPv6 Using Unused Addresses”. In: *Proceedings of the 10th Passive and Active Measurement (PAM)*. 2015.

[95] Ramakrishna Padmanabhan, Aaron Schulman, Dave Levin, and Neil Spring. “Residential Links under the Weather”. In: *Proceedings of the 2019 ACM SIGCOMM Conference (SIGCOMM)*. 2019.

[96] J. Park, A. Khormali, M. Mohaisen, and A. Mohaisen. “Where Are You Taking Me? Behavioral Analysis of Open DNS Resolvers”. In: *Proceedings of the 2019 USENIX Symposium on Networked Systems Design and Implementation (NSDI)*. 2019. URL: https://www.usenix.org/conference/nsdi19/presentation/jin.

[97] J. Park, A. Khormali, M. Mohaisen, and A. Mohaisen. “Where Are You Taking Me? Behavioral Analysis of Open DNS Resolvers”. In: *Proceedings of the 49th IEEE/IFIP International Conference on Dependable Systems and Networks (DSN)*. 2019.

[98] Paul Pearce, Ben Jones, Frank Li, Roya Ensafi, Nick Feamster, Nick Weaver, and Vern Paxson. “Global Measurement of {DNS} Manipulation”. In: *Proceedings of the 26th USENIX Security Symposium (USENIX Security)*. 2017. URL: https://www.usenix.org/conference/usenixsecurity17/technical-sessions/presentation/pearce.

[99] Giancarlo Pellegrino, Onur Catakoglu, Davide Balzarotti, and Christian Rossow. “Uses and Abuses of Server-Side Requests”. In: *Proceedings of the 19th International Symposium on Recent Advances in Intrusion Detection (RAID)*. 2016.

[100] Adrienne Porter Felt, Richard Barnes, April King, Chris Palmer, Chris Bentzel, and Parisa Tabriz. “Measuring {HTTPS} Adoption on the Web”. In: *Proceedings of the 26th USENIX Security Symposium (USENIX Security)*. 2017. URL: https://www.usenix.org/conference/usenixsecurity17/technical-sessions/presentation/felt.

[101] Christopher Rentrop and Stephan Zimmermann. “Shadow IT: Management and Control of Unofficial IT”. In: *Proceedings of the The 6th International Conference on Digital Society (ICDS)*. 2012.

[102] RIPE. *Routing Information Service (RIS)*. Nov. 2019. URL: https://www.ripe.net/analyse/internet-measurements/routing-information-service-ris/routing-information-service-ris (visited on 06/22/2020).

[103] *Routeviews – University of Oregon Route Views Project*. en-US. URL: http://www.routeviews.org/routeviews/ (visited on 06/22/2020).

[104] Jan Rüth, Ingmar Poese, Christoph Dietzel, and Oliver Hohlfeld. “A First Look at QUIC in the Wild”. In: *Proceedings of the 13th Passive and Active Measurement (PAM)*. 2018.

[105] Sarah Scheffler, Sean Smith, Yossi Gilad, and Sharon Goldberg. “The Unintended Consequences of Email Spam Prevention”. In: *Proceedings of the 13th Passive and Active Measurement (PAM)*. 2018.

[106] Quirin Scheitle, Oliver Gasser, Theodor Nolte, Johanna Amann, Lexi Brent, Georg Carle, Ralph Holz, Thomas C. Schmidt, and Matthias Wählisch. “The Rise of Certificate Transparency and Its Implications on the Internet Ecosystem”. In: *Proceedings of the 2018 Internet Measurement Conference (IMC)*. 2018.

[107] Quirin Scheitle, Oliver Gasser, Minoo Rouhi, and Georg Carle. “Large-Scale Classification of IPv6-IPv4 Siblings with Variable Clock Skew”. In: *Proceedings of the 2017 International Workshop on Traffic Monitoring and Analysis (TMA)*. 2017.

[108] Kyle Schomp, Tom Callahan, Michael Rabinovich, and Mark Allman. “On Measuring the Client-Side DNS Infrastructure”. In: *Proceedings of the 2013 Internet Measurement Conference (IMC)*. 2013.

[109] Will Scott, Thomas Anderson, Tadayoshi Kohno, and Arvind Krishnamurthy. “Satellite: Joint Analysis of CDNs and Network-Level Interference”. In: *Proceedings of the 2016 USENIX Annual Technical Conference (USENIX ATC)*. 2016. URL: https://www.usenix.org/conference/atc16/technical-sessions/presentation/scott.

[110] *Shodan*. URL: https://www.shodan.io/ (visited on 06/02/2020).

[111] Haya Shulman and Michael Waidner. “One Key to Sign Them All Considered Vulnerable: Evaluation of DNSSEC in the Internet”. In: *Proceedings of the 2017 USENIX Symposium on Networked Systems Design and Implementation (NSDI)*. 2017. URL: https://www.usenix.org/conference/nsdi17/technical-sessions/presentation/shulman.

[112] J. Sonchack, A. Dubey, A. Aviv, J. Smith, and E. Keller. “Timing-based Reconnaissance and Defense in Software-defined Networks”. In: *Proceedings of the 32nd Annual Computer Security Applications Conference (ACSAC)*. 2016.

[113] Kyle Soska and Nicolas Christin. “Measuring the Longitudinal Evolution of the Online Anonymous Marketplace Ecosystem”. In: *Proceedings of the 24th USENIX Security Symposium (USENIX Security)*. 2015. URL: https://www.usenix.org/conference/usenixsecurity15/technical-sessions/presentation/soska.

[114] Drew Springall, Zakir Durumeric, and J. Alex Halderman. “FTP: The Forgotten Cloud”. In: *Proceedings of the 46th IEEE/IFIP International Conference on Dependable Systems and Networks (DSN)*. 2016.

[115] Drew Springall, Zakir Durumeric, and J. Alex Halderman. “Measuring the Security Harm of TLS Crypto Shortcuts”. In: *Proceedings of the 2016 Internet Measurement Conference (IMC)*. 2016.

[116] *State of the IoT 2020: 12 billion IoT connections, surpassing non-IoT for the first time*. URL: https://iot-analytics.com/state-of-the-iot-2020-12-billion-iot-connections-surpassing-non-iot-for-the-first-time/ (visited on 02/18/2021).

[117] Samaneh Tajalizadehkhoob, Maciej Korczynski, Arman Noroozian, Carlos Ganan, and Michel van Eeten. “Apples, Oranges and Hosting Providers: Heterogeneity and Security in the Hosting Market”. In: *Proceedings of the2016IEEE Network Operations and Management Symposium (NOMS)*. 2016.

[118] Samaneh Tajalizadehkhoob, Tom Van Goethem, Maciej Korczyński, Arman Noroozian, Rainer Böhme, Tyler Moore, Wouter Joosen, and Michel van Eeten. “Herding Vulnerable Cats: A Statistical Approach to Disentangle Joint Responsibility for Web Security in Shared Hosting”. In: *Proceedings of the 24th ACM SIGSAC Conference on Computer and Communications Security (CCS)*. 2017.

[119] The ZMap Project. *Zmap/Zgrab2: Fast Go Application Scanner*. The ZMap Project. Aug. 2016. URL: https://github.com/zmap/zgrab2 (visited on 06/02/2020).

[120] Olivier van der Toorn, Roland van Rijswijk-Deij, Tobias Fiebig, Martina Lindorfer, and Anna Sperotto. “TXTing 101: Finding Security Issues in the Long Tail of DNS TXT Records”. In: *2020 IEEE European Symposium on Security and Privacy Workshops (EuroS&PW)*. IEEE. 2020.

[121] U.S. Attorney’s Office, Southern District of New York. *Dozens Of Online “Dark Markets” Seized Pursuant To Forfeiture Complaint Filed In Manhattan Federal Court In Conjunction With The Arrest Of The Operator Of Silk Road 2.0*. http://www.justice.gov/usao/nys/pressreleases/November14/DarkMarketTakedown.php. Nov. 2014.

[122] *United States of America vs. Ross William Ulbricht*. United States District Court, Southern District of New York. Indictment 14CRIM068. Feb. 2014.

[123] Benjamin VanderSloot, Allison McDonald, Will Scott, J. Alex Halderman, and Roya Ensafi. “Quack: Scalable Remote Measurement of Application-Layer Censorship”. In: *Proceedings of the 27th USENIX Security Symposium (USENIX Security)*. 2018. URL: https://www.usenix.org/conference/usenixsecurity18/presentation/vandersloot.

[124] Matteo Varvello, Kyle Schomp, David Naylor, Jeremy Blackburn, Alessandro Finamore, and Konstantina Papagiannaki. “Is the Web HTTP/2 Yet?” In: *Proceedings of the 10th Passive and Active Measurement (PAM)*. 2016.

[125] Thomas Vissers, Tom Van Goethem, Wouter Joosen, and Nick Nikiforakis. “Maneuvering Around Clouds: Bypassing Cloud-Based Security Providers”. In: *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security (CCS)*. 2015.

[126] Rolf van Wegberg, Samaneh Tajalizadehkhoob, Kyle Soska, Ugur Akyazi, Carlos Hernandez Gáñan, Bram Klievink, Nicolas Christin, and Michel van Eeten. “Plug and Prey? Measuring the Commoditization of Cybercrime via Online Anonymous Markets”. In: *Proceedings of the 27th USENIX Security Symposium (USENIX Security)*. 2018. URL: https://www.usenix.org/conference/usenixsecurity18/presentation/van-wegberg.

[127] Maarten Wullink, Giovane C. M. Moura, and Cristian Hesselman. “Dmap: Automating Domain Name Ecosystem Measurements and Applications”. In: *2018 Network Traffic Measurement and Analysis Conference (TMA)*. June 2018.

[128] Haitao Xu, Fengyuan Xu, and Bo Chen. “Internet Protocol Cameras with No Password Protection: An Empirical Investigation”. In: *Proceedings of the 13th Passive and Active Measurement (PAM)*. 2018.

[129] Bahador Yeganeh, Ramakrishnan Durairajan, Reza Rejaie, and Walter Willinger. “How Cloud Traffic Goes Hiding: A Study of Amazon’s Peering Fabric”. In: *Proceedings of the 2019 Internet Measurement Conference (IMC)*. 2019.

[130] Jing Zhang, Zakir Durumeric, Michael Bailey, Mingyan Liu, and Manish Karir. “On the Mismanagement and Maliciousness of Networks”. en. In: *Proceedings of the 2014 Network and Distributed System Security Symposium*. 2014. URL: https://www.ndss-symposium.org/ndss2014/programme/mismanagement-and-maliciousness-networks/ (visited on 12/04/2019).