Enlightening the Darknets: Augmenting Darknet Visibility with Active Probes

This article has been accepted for publication in IEEE Transactions on Network and Service Management. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TNSM.2023.3267671

Enlightening the Darknets: Augmenting Darknet Visibility with Active Probes

Francesca Soro,[1] Thomas Favale,[1] Danilo Giordano, [1] Idilio Drago,[2]
Tommaso Rescio,[1] Marco Mellia,[1] Zied Ben Houidi,[3] Dario Rossi[3]
[1]Politecnico di Torino, first.last@polito.it
[2]Università di Torino, idilio.drago@unito.it
[3]Huawei Technologies Co. Ltd, zied.ben.houidi@huawei.com,dario.rossi@huawei.com

***Abstract*—Darknets collect unsolicited traffic reaching unused address spaces. They provide insights into malicious activities, such as the rise of botnets and DDoS attacks. However, darknets provide a shallow view, as traffic is never responded. Here we quantify how their visibility increases by responding to traffic with interactive responders with increasing levels of interaction.**

We consider four deployments: Darknets, simple L4-Responders, vertical L7-Responders bound to specific ports, and DPIpot, a honeypot that responds to all protocols on any port. We contrast these alternatives by analyzing the traffic attracted by each deployment and characterizing how traffic changes throughout the responder lifecycle on the darknet. We show that the deployment of responders increases the value of darknet data by revealing patterns that would otherwise be unobservable. We measure *Side-Scan* phenomena where once a host starts responding, it attracts traffic to other ports and neighboring addresses. DPIpot uncovers attacks that darknets and L7-Responders would not observe, e.g., large-scale activity on non-standard ports. And we observe how quickly senders can identify and attack new responders.

The "enlightened" part of a darknet brings several benefits and offers opportunities to increase the visibility of sender patterns. This information gain is worth taking advantage of, and we, therefore, recommend that organizations consider this option.

***Index Terms*—Darknets, Measurements, Network security.**

I. Introduction

Darknets or network telescopes are IP addresses advertised by routing protocols without hosting any services. They have been used for years as *passive sensors* in a variety of network monitoring activities and research projects [1], [2], [3], [4]. Traffic reaching a darknet is inevitably unsolicited. Therefore, it is helpful to highlight network scans (both from malicious and legitimate scanners), backscattering (i.e., traffic received from victims of attacks carried out with spoofed IP addresses), and traffic due to bugs and misconfigurations [2].

To increase the visibility of attackers' activities, *honeypots* allow researchers to obtain more information about events on darknets [5], [6], [7], [8]. Honeypots are *active sensors* that collect information by responding to unsolicited traffic. The goal is to engage with potential attackers using simulators that replicate the basic functions of real systems (low-interaction honeypots) or actual live systems deployed in controlled environments (high-interaction honeypots).

Darknets and honeypots are complementary: the former provides a broad but shallow view of scanning activity; the latter provides deeper insights into specific attack patterns. A combination of the two cases could enrich the type of information currently being gleaned from darknets while providing broad coverage and deep insights. Darknet traffic is known to change significantly, not only in the IP address space but also due to production services hosted "near" the darknet's address space [9], [10].

We present our efforts to systematically and quantitatively compare different levels of interactive responders that we deploy *within* different portions of our darknet address space. We consider the following four types of sensors: (i) Darknet, silent listeners that capture received traffic; (ii) L4-Responders, which completes the TCP handshake and stores all possible application layer requests sent by clients; (iii) L7-Responders, low-interaction honeypots that mimic specific application protocols on their usual and known ports; (iv) DPIpot, a novel responder that identifies the application protocol used by the sender regardless of the destination port.

We design two experiments to observe how senders[1] interact with the responders. In the first setup, we activate the responders in a /24 darknet, while keeping a second /24 network as a pure darknet. We run this setup for months and observe the traffic that each sensor attracts over an extended period of time. In the second experiment, we first turn off all responders to measure the effects of darkening a network with active services. After 15 days, we turn on responders in our second /24 darknet, measuring the transient effects of deploying active responders in a darknet.

Our goal is to revisit and update some well-known facts about darknet deployments and add new and fresh insights that highlight the advantages and disadvantages of alternative response strategies. Summarizing our key findings:

- We quantify *Side-Scan* phenomena where a node hosting a service receives more traffic for other services and ports. Unlike [9], which observes similar patterns in CDN nodes, we quantify how the type of service hosted in the darknet critically affects *Side-Scan*.
- Activating responders leads to an increase in traffic on darknet neighboring IP addresses too. The joint use of active responders and passive darknets increases the visibility of sender patterns and improves the understanding of phenomena and attacks.

[1]We call *sender* hosts contacting our darknet, e.g., attackers, scanners, etc.

This article has been accepted for publication in IEEE Transactions on Network and Service Management. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TNSM.2023.3267671

- L4-Responders and L7-Responders increase the visibility of darknets, as reported in [11], [5]. However, the lack of a wide range of application-level responders limits interaction and traffic visibility compared to our multiple L7-Responders and DPIpot.
- DPIpot decouples services from ports, shedding light on activities directed to non-standard ports, offering a rich picture that is unseen in other deployments. As a side-effect, it may "trap" senders in some particular activities, slowing them. This trade-off shows how passive and interactive deployments are complementary.
- We observe several scanning patterns that senders employ to discover hosts and services in a network. We document how fast hosts become the target of in-depth activity from multiple senders once found online by some initial scanners. Conversely, senders keep coming back (for weeks) to IP addresses that once hosted active responders.

In addition to these analyzes, we provide the complete set of responders used in our analysis as open-source software. We release DPIpot to foster its development and use. We also make the data analyzed in the following sections available online in anonymized form to allow for reproducibility.

We provide an overview of related work (Sec. II), explain our methodology (Sec. III), and describe macroscopic traffic characteristics (Sec. IV). We examine the changes in different deployments (Sec. V) and the benefits of DPIpot (Sec. VI). Finally, we observe what happens when we darken and lighten a network (Sec. VII), before concluding the paper (Sec. VIII).

II. Related work

A. Darknet research and infrastructure

Darknets have been employed for years in various network monitoring and research activities [1]. Examples include the study of (i) DDoS attacks [8], [12], [13], (ii) IPv4 address space usage [14], (iii) Internet censorship [15], (iv) large-scale internet scanning [16], [4], [17], and (v) botnets and malware proliferation [6], [18].

In terms of infrastructure, previous efforts have characterized the differences between centralized and sparse implementations, size, and location of darknets [1], [3]. Several actors maintain darknet infrastructures, including the decades-old CAIDA/UCSD [19] project, darknets operated by major network operators [2], [3], and other projects run by universities and security companies worldwide [7], [10], [20], [21], [22], [23].

Recent work [9] used servers from Akamai's Content Delivery Network (CDN) to study unsolicited traffic. Unlike a traditional darknet, CDN nodes provide public services and thus receive and process production traffic. Nevertheless, all `TCP/UDP` ports that do not host production services can be reached by unsolicited traffic. The authors show that production servers attract unsolicited traffic that is quite different from the traffic observed in an ordinary darknet. We extend these findings by uncovering and investigating different deployment combinations and services. Although our deployment is based on honeypots and thus lacks components of a production environment, we show that the combination of services exposed in a host is important and shapes the mix of attacks and noise that targets it.

Similar to our methodology, authors [11] propose Spoki, which completes TCP handshakes in the darknet to record the first payload sent by scanners. The authors of [5] present eX-IoT, IoT honeypots deployed in darknets. Spoki is similar to our L4-Responders, while eX-IoT is a new category of L7-Responders. Our methodology includes multiple functions beyond Spoki and eX-IoT, including multiple categories of L7-Responders and a new DPIpot that performs deep packet inspection (DPI) on-the-fly to decide how to respond to scanners. We show that advanced responders shed light on a new wave of scanners and attackers that are not visible in a pure darknet or when using a single responder type such as in [11], [5].

B. Honeypot systems and analysis

We study the impact of deploying active services on the darknet using honeypots as responders. Honeypots have been used in security activities for years, with well-established projects such as the Honeynet Project [24] and TPot [25] providing several alternatives. Previous works on honeypots have covered many aspects, such as (i) introducing new honeypots that target specific protocols or services [26], [27], (ii) evaluating the effectiveness of different types of honeypots [28], and (iii) presenting techniques to detect honeypots[29], [30]. Readers are invited to review the survey at [31], which provides a broad overview of honeypot research.

Some authors present a general characterization of honeypot traffic, focusing on the origin of attacks, the targeted services, and the frequency of attacks (e.g., [32], [33], [27], [34], [35]). A recent work [36] compared the use of honeypots in different geographic locations. Another work [37] allocated unused addresses in a cloud for honeypot deployment. We revisit these efforts here evaluating the deployment of honeypots compared to what is observed in dark spaces. In addition, we review how measurements from different active responders differ from (and influence) measurements collected in darknets.

Our DPIpot leverages DPI to decide how to respond to incoming traffic. This setup allows attacks on non-standard ports to be detected. Some *meta-honeypots* allow flexible configuration of backends to handle traffic on non-standard ports [38], [39], [25]. Most of these systems act as proxies (at various levels) logging the traffic forwarded to the backend. However, they lack DPIpot mechanisms to identify traffic on-the-fly for a variety of protocols.

Honeytrap [40] is the closest honeypot to DPIpot. Honeytrap is a meta-honeypot that performs protocol identification. However, it implements only a limited number of protocol fingerprints. Honeytrap supports about 26 services and provides the ability to extend the set of protocol identification rules. DPIpot instead relies on a state-of-the-art DPI library (nDPI [41]), which is widely used in other network applications and provides hundreds of protocol fingerprints. In fact, we compared different DPI solutions in [42] and concluded that nDPI provides the best coverage and precision for DPIpot.

This article has been accepted for publication in IEEE Transactions on Network and Service Management. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TNSM.2023.3267671

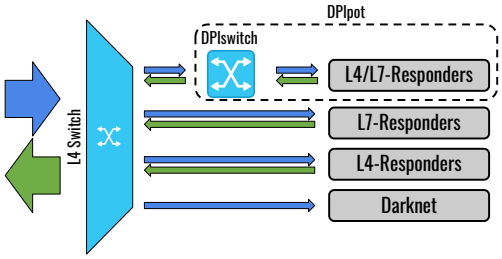


Fig. 1. Infrastructure overview. The infrastructure is divided from no interaction (Darknet) to the highest level of interaction (DPIpot).

III. Methodology and Datasets

A. Infrastructure

Fig. 1 schematically represents our measurement infrastructure. Unsolicited traffic that reaches our dedicated address space is routed – totally unfiltered – to one of our four *deployments* that correspond to different levels of interactivity:

1) Darknet: IP addresses that just receive traffic without responding to any packet;
2) L4-Responders: responders that complete the TCP three-way handshake, capture eventual application requests from clients, but never respond to an application message;
3) L7-Responders: honeypots that mimic popular application layer services. We use state-of-the-art honeypots to simulate well-known services; L7-Responders act as vertical responders that only interact with a limited number of ports and services;
4) DPIpot: our novel responder that performs L7 switching of requests using DPI. It decides on-the-fly which protocol to use, and responds to TCP connections on all TCP ports. Unlike L7-Responders, DPIpot decouples the default TCP ports from the application protocols.[2]

Note that none of our deployments respond to UDP or malformed TCP packets to prevent abuse (see ethical concerns in Sec. III-E).

For the L7-Responders deployments, we rely on the honeypots organized and distributed by the TPot project [25]. We activate honeypots to handle a range of popular application protocols. TPot offers a collection of third-party *low-interaction* honeypots, i.e., programs crafted to simulate a vulnerable service communicating over a given L7 protocol. Most of our L7-Responders offer login interfaces only [43], registering the brute-force attempts against services (e.g., RDP, POP3, and IMAP). Some L7-Responders rely on more sophisticated honeypots, e.g., simulating a vulnerable server accessible via SSH/Telnet [38], or serving pages that mimic actual services accessible over the web [44]. We defer the reader to the documentation of TPot for details.

Our L7-Responders offer *vertical services* only: They are deployed behind the standard TCP ports of the given service, e.g., the HTTP honeypot is deployed on port TCP/80 whereas the Remote Desktop Protocol (RDP) honeypot responds on port TCP/3389. To investigate the impact of responding to traffic arriving on other ports, we implement and deploy DPIpot. DPIpot listens on *all* TCP ports. On receiving a new TCP connection request, it completes the three-way handshake and waits for the first message from the client. Then it analyzes the payload looking for the application-layer protocol. DPIpot relies on `nDPI` [41]. This choice gives us a flexible system that supports hundreds of protocols, which is far more than in previous projects [40]. If a known protocol is found and one of the L7-Responders can handle it, DPIpot directs traffic to such backend; otherwise it acts like L4-Responders. Note that DPIpot can identify and direct traffic only in cases that are *client initiated*, i.e., where the client sends the first application-layer message. Otherwise, it behaves like L4-Responders– e.g., in telnet or SMTP, where the client waits for the server banner before attempting to log in.

Both our L4-Responders and DPIpot are implemented in Python using the Twisted framework [45]. Our architecture (see Fig. 1) is intrinsically distributed, and Twisted is scalable. However, as we will show later, the deployment of responders increases the traffic reaching the darknet by orders of magnitude. To prevent abuses, we thus intentionally limit the capacity of our infrastructure (see Sec. III-E).

B. Data capture and processing

We isolate one /23 network to perform experiments with our multiple *deployments* – i.e., darknet, L4-Responders, L7-Responders or DPIpot. Our setup is deployed in /16 campus network (at Politecnico di Torino, in Italy) that hosts servers and clients. Our infrastructure captures all packets hitting the /23 darknet. We use `tcpdump` and store all traces on a high-end server to generate separate logs for each deployment.

We here characterize the traffic focusing on TCP flows, defined by the usual 5-tuple (client/server IP addresses, client/server ports, and transport-layer protocol). A new flow starts when a `SYN` segment is received, and it terminates after the connection is closed (in case of the active responders) or an idle time. We annotate each flow with useful metadata and statistics, including the application protocol identified by `nDPI`, if any L7 payload is present.

According to the capabilities of each responder, we identify different *flow stages*:

- `SYN`: Flows for which we observe only the `SYN` message(s), eventually retransmitted by the client multiple times; This is the most common case on darknets, but it happens also on the blocked ports of other deployments or when a responder is unable to cope with the workload;
- `2WH`: Incomplete three-way handshake, where the client ignores (or resets) the `SYN/ACK` message, as in the case of stealth-`SYN` port scans;
- `3WH`: Client and server complete the TCP three-way handshake, but exchange no payload – this is expected in L4-Responders and DPIpot when clients wait for servers to initiate the conversation;
- `L7 payload`: Client and server open the TCP connection and exchange some application-layer messages.

In addition, we record malformed TCP messages, e.g., `SYN/ACK` likely arriving due to backscattering or other packets with bogus TCP flags, as well as any other protocol (UDP,

[2]To avoid resource starvation, L4-Responders and DPIpot implement active and inactive timeouts dropping active (idle) connections after 60 s (10 s).

This article has been accepted for publication in IEEE Transactions on Network and Service Management. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TNSM.2023.3267671

TABLE I
DEPLOYMENTS AND PROTOCOLS. EACH ROW REFERS TO THE TRAFFIC OF 8 IP ADDRESSES DURING OUR FIRST CAPTURE PERIOD (FROM 15/04/2021 TO 16/06/2021). FOR DIRECT COMPARISON, WE REPORT NUMBERS ONLY FOR THE 8 FIRST ADDRESSES IN DARKNET EXT.

Deployment	Category	Addr.	Category: Port	Category: Application	Flows	Flows with L7 Payload	Sender Addr.
DPIpot	*All*	136:143	0:65535	*All below*	1927 M	1207 M	133 k
L7-Responders	*Mail*	128:135	25, 110, 143, 465, 993, 995	pop(s), imap(s), smtp(s)	4 M	71 k	120 k
	Terminal	120:127	22, 2222*, 23, 2323*	ssh, telnet	10 M	5 M	139 k
	Fileserver	112:119	135:139, 445	netbios, CIFS	11 M	6 M	119 k
	Remote Desktop	104:111	3389, 5900, 5901, 5800*, 5801*, 5938*, 6568*	ms rd, vnc, teamviewer, anydesk	13 M	7 M	122 k
	Database	96:103	3306, 33060*, 1433, 4022*, 1434*, 5432*, 27017	mysql, mssql, postgres, mongodb	4 M	212 k	121 k
	Proxy	88:95	8080, 8000*, 3128	generic, squid	5 M	43 k	121 k
	Web	80:87	80, 443	http(s)	4 M	93 k	127 k
	All	72:79	*All above*	*All above*	32 M	23 M	157 k
L4-Responders	*Mail*	64:71	25, 110, 143, 465, 993, 995	-	4 M	36 k	123 k
	Terminal	56:63	22, 2222, 23, 2323	-	6 M	546 k	123 k
	Fileserver	48:55	135:139, 445	-	5 M	1 M	120 k
	Remote Desktop	40:47	3389, 5900, 5901, 5800, 5801, 5938, 6568	-	6 M	546 k	147 k
	Database	32:39	3306, 33060, 1433, 4022, 1434, 5432, 27017	-	4 M	356 k	123 k
	Proxy	24:31	8080, 8000, 3128	-	5 M	38 k	123 k
	Web	16:23	80, 443	-	4 M	59 k	131 k
	All	8:15	0:65535	-	13 M	6 M	146 k
Darknet Int	–	2:5;176:179	0:65535	-	4 M	0	125 k
Darknet Ext	–	2:5;176:179	0:65535	-	4 M	0	111 k

(∗) Ports that are forwarded to the L7-Responders, even if the backend (i.e., TPot) does not host any honeypot. The L7-Responders reset the connection in these cases, as opposed to the darknet (which never responds to traffic) and the L4-Responders (which always try to open a connection request).

ICMP, etc). These cases are however not discussed in the paper but included in the public traces we release to the community.

C. First experiment: Deployments and categories

We perform two experiments. In the first round, we record traffic for two months, from the 15th of April to the 16th of June 2021. This capture starts several months after the deployment of the active responders, thus representing a picture of a stable deployment of active responders in a darknet.

In this first experiment, we split the /23 network into two /24 networks. One /24 network hosts no service and operates as a classic darknet, hereafter called *Darknet Ext* – see Tab. I. Unless explicitly mentioned, all results referring to our first experiment and using a darknet as baseline rely upon this *Darknet Ext*.

Then, we split the other /24 addresses into groups of 8 IP addresses. We deploy several categories of responders inside this single /24 as reported in Tab. I (see the third column). Some host L4-Responders and L7-Responders and respond to 8 specific service *categories*, 8 for L4-Responders, and 8 for L7-Responders. Each category defines which services the responder supports. We configure the responders to receive and handle only traffic that arrives at ports typically hosting services belonging to such category, silently dropping packets arriving on other ports. We create categories for database, file, mail, proxy, remote desktop, terminal, and web services. We report all ports opened for each category on Tab. I, together with some typical applications relying on such ports. We also create an *extra* category denoted as *All*, for which we accept all traffic going to any port. In the case of the *all* category in L4-Responders, we perform a TCP handshake for flows arriving in any TCP port. For the L7-Responders category denoted as *All*, we pass all traffic to the TPot backend, regardless of whether there is a honeypot active on that port or not. If no honeypot is present, the backend explicitly resets the connection.

We devote 8 IP addresses to host DPIpot, which responds in all ports. It performs DPI on the arriving packets to identify the most appropriate responder based on the payload, and eventually forwards traffic to a honeypot offered by TPot.

The remaining IP addresses in the /24 hosting the active responders act as darknet. We select 8 of these IP addresses and call them *Darknet Int*.

D. Second experiment: Deploying and removing responders

We perform a second experiment to assess the transient impact of activating and shutting down responders in the darknet. We start by shutting down all active responders on the 25th of January 2022, thus letting the complete /23 behave like a darknet. This allows us to observe whether (and how) senders continue to search for the responders after they are removed from the network.

On the 9th of February 2022, we light up fresh responders in the /24 that previously served as darknet (*Darknet Ext*). This /24 had been used as a darknet for many years before the start of our experiments. As such, it allows us to observe the speed senders discover new services as well as all transition steps from a darknet IP address into active responders.

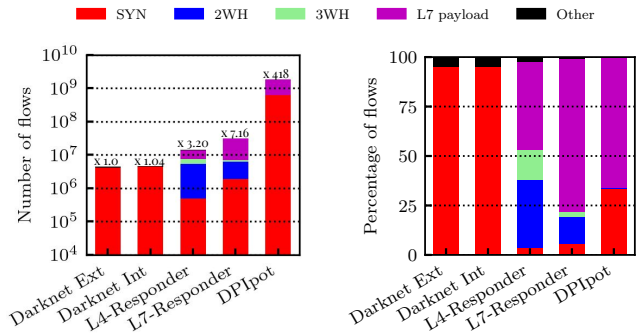
We deploy L4-Responders, L7-Responders, and DPIpot using 8 IP addresses for each deployment, which are distributed in the /24 network as reported in Tab. II. In this experiment, we use only the *All* category for L4-Responders and L7-Responders. The deployment is instrumental to maintain an equally spaced set of dark IP addresses between each group of active responders. This would let us measure whether the placement of active responders impacts the neighboring addresses.

This article has been accepted for publication in IEEE Transactions on Network and Service Management. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TNSM.2023.3267671

TABLE II
IP ALLOCATION IN THE FRESH DEPLOYMENT OF ACTIVE RESPONDERS.

Deployment	Category	Addr.
Darknet_4	–	192:255
DPIPot	*All*	184:191
Darknet_3	–	128:183
L7-Responder	*All*	120:127
Darknet_2	–	64:119
L4-Responder	*All*	56:63
Darknet_1	–	0:55
Darknet^*_{ext}	–	–

$(*)$ **Darknet**$_{ext}$ in this experiment is equivalent to **Darknet Int** in Tab. I.



(a) Flows (notice the y-log scale). (b) Traffic share in the deployment

Fig. 2. Flows reaching different deployments. Numbers inside the left plot mark the increase with respect to *Darknet Ext*.

E. Ethics

We take several countermeasures to restrict the impact of our measurements on third-party networks. First, and most importantly, we never send packets if our packets may worsen the position of attack victims. In particular, we never send UDP traffic, as it could make our infrastructure part of DDoS attacks relying on spoofed addresses and amplification techniques. For the same reason, we silently drop all TCP packets with `SYN/ACK` flags and other malformed flows, as they may arrive from victims of DDoS attacks with spoofed addresses. Responding to such packets may help the attackers to overload the victims' networks.

The traffic we observe may come from infected machines that are taking part in botnets. As previously said, we explicitly limit the capacity of our infrastructure to avoid creating too much traffic for the networks hosting such infected machines. The setup discussed in this paper can comprehensively sustain at most a few Mbps of traffic upstream, which is far insufficient to overload remote networks.

Finally, IP addresses sending traffic to our infrastructure may uncover vulnerable computers exploited by attackers [46]. We take all measures to protect such IP addresses. We anonymize addresses in the datasets we release publicly. We also collaborate with our security team and our upstream providers, actively notifying them about novel attacks and senders.

IV. MACROSCOPIC CHANGES IN TRAFFIC

We report a high-level characterization of the different deployments aiming to answer the following question: How much extra information one would get when some IP addresses inside a darknet actively respond to incoming traffic?

For easy comparisons, we restrict our analysis to 8 addresses per deployment. We focus on those addresses in the *all* category in the case of L4-Responders and L7-Responders, and get 8 addresses from *Darknet Ext* and *Darknet Int* (see Tab. I). Here we focus on our first experiment setup, and whenever not explicitly mentioned we report statistics for the first month of our dataset for easy visualization.

A. Breakdown per flow stage

Fig. 2 reports the number of flows received in each deployment, breaking it down per flow stage. The left plot details the number of flows (notice the y-log scale) while the right plot details the share in each deployment.

Darknets observe a large majority of TCP `SYN` messages, with a few UDP and bogus TCP segments (about 8% of the total). As soon as we start replying with the L4-Responders, the number of flows grows by a factor of 4 compared to the darknets[3] (cfr. Tab. I). Although our deployment shall perform the full 3-way handshake, a small portion of flows remain in the `SYN` stage, i.e., connection requests to which our L4-Responders deployment cannot reply due to short-term congestion. Interestingly, 35% of the flows terminate at the `2WH` stage, most likely corresponding to "TCP-SYN scans" (also reported in [11]). About one fourth of the open TCP flows carries no payload, i.e., likely host discovery actions performed with a "TCP-connect scan".

Consider now the L7-Responders. the number of flows doubles again. The `SYN` stage flows are now about 7%. Part of this traffic is again caused by the limits we impose on our infrastructure. However, as we will see later, once we respond to traffic in some ports, more scans are observed in other ports too. This effect increases the number of `SYN`-stage flows. Naturally, we observe a strong increase of `L7 payload` flows, which are now about 72% of the total.

Moving to DPIpot, it attracts 3 and 2 orders of magnitude more flows than the darknet and the L7-Responders, respectively. The number of flows grows to billions – about 70 times more than in the L7-Responders, and 600 times more than in the darknets. Here we see around 40% of cases finishing on `SYN` stage, which correspond to periods in which our deployment hits its capacity.

It is worth commenting that the share of `Other` traffic remains similar in all deployments. This suggests that responding to TCP traffic as we do in our deployments does not stimulate senders to generate packets using UDP/ICMP.

Fig. 3 reports the number of flows observed for each IP address selected for this analysis. Here, we see that the number of flows reaching each deployment is well-divided among the IP addresses belonging to the given deployment. Little variations are seen for the L4-Responders, where a couple

[3]Other subsets of darknet addresses yield a stable number of flows

This article has been accepted for publication in IEEE Transactions on Network and Service Management. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TNSM.2023.3267671

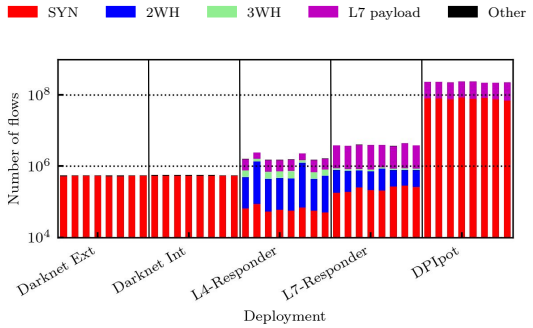


Fig. 3. Number of flows per deployment. Each bar in the figure reports numbers for a single IP address of several deployments.

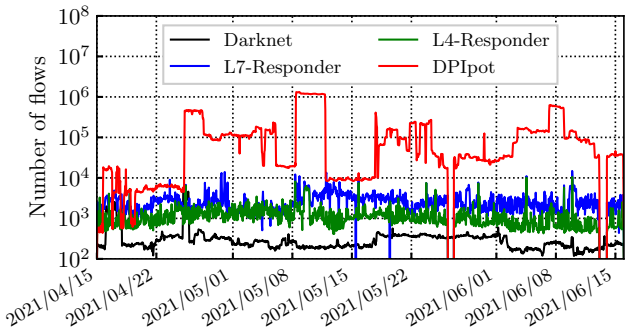


Fig. 4. Temporal evolution of the number of flows.

of sources contribute with large-scale attempts against two L4-Responders IP addresses.

More interestingly, the number of unique sources contacting each deployment changes considerably (see numbers in the last column of Tab. I). Differences are visible between *Darknet Ext* and *Darknet Int*. In fact, IP addresses belonging to *Darknet Int* attract thousands of sources more than those in *Darknet Ext*. We conjecture that this behavior is a consequence of the presence of active responders in the same /24 subnet. Once sources find services in a subnet, they search for services in the neighbor addresses. We will investigate this in more detail in Sec. VII.

Finally, L4-Responders, L7-Responders and DPIpot attract more senders than the darknets. In sum, deploying active responders sheds light on new scanners and attackers that would not be uncovered with simple darknets.

B. Temporal evolution

Darknets and honeypots are known to receive variable traffic over time. Fig. 4 reports the average per-hour number of flows received by each deployment (*All* category). Here we report time series covering the full 2-month dataset of our first experimental setup. Notice the y-log scale. As expected, the darknet is steadily the least contacted deployment with a few hundred flows per hour on average, except during sporadic scans hitting the address space [2], [8], [10]. Both L4-Responders and L7-Responders show a noisier pattern over time, again with small episodes of increases. The DPIpot registers much more variable figures. For instance, flows per hour top to more than 1 million on May 7th to suddenly vanish on May 12th. We will detail this case in Sec. VI-C. As said above, these episodes bring DPIpot to the limits we impose on the infrastructure.

Takeaway: The number of flows grows by orders of magnitude with increasingly interactive responders. Vertical honeypots attract many times more flows than darknets. DPIpot pushes this increase further thanks to its ability to respond to application traffic on non-standard ports. This growth creates temporal bursts of traffic that challenge the deployment itself and calls for protection mechanisms to avoid collapsing the infrastructure and biasing the collected data.

V. Ports and Senders

We have observed the effects of answering darknet traffic in terms of traffic volume. We now assess changes in traffic patterns along two axes: the targeted services and sender IP addresses. This section answers the following question: How the presence of active services changes the behavior of the groups of senders and of contacted services?

For this analysis, we again rely on our first experimental setup, i.e., long-run assignment of addresses to active responders.

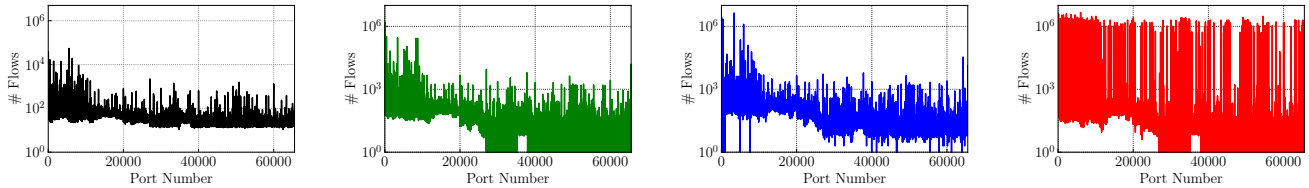
A. Changes on probed ports

The traffic volume is expected to vary over the exposed ports. Already in a darknet, well-known ports are expected to be more frequently contacted than others. Fig. 5 reports the number of flows per port for the different deployments. Here we see interesting effects of deploying the active responders. Common for darknet, L4-Responders, and L7-Responders, senders concentrate their interest on well-known ports below 1024. On the contrary, DPIpot attracts much more flows on very uncommon ports (notice the y-log scale). Investigating the `L7 payload`, these flows are related to Remote Desktop Protocol (RDP), hinting at a specific attack (we investigate this in Sec. VI).

In detail, focus on the darknet (black plot on the left). While some ports do receive a much larger share of traffic as expected, scanners cover the whole port range. This confirms how darknets are effective to observe senders performing horizontal scans, i.e., doing host discovery.

When we start responding to requests, the picture drastically changes. For instance, the top ports in the L4-Responders account for more than 60% of the flows. This percentage grows to more than 70% in L7-Responders. See how the number of flows increases for low, well-known ports in Fig. 5-b and Fig. 5-c. That is, once a target is discovered, senders activate the next stages of scans or attacks. Interestingly, senders contacting L4-Responders skip some ports starting from port 27 000 (the number of flows goes below 1 in the y-log scale). Curiously, notice the continuous group of ports $[35000 : 38000]$ where senders again check all ports. For the sake of completeness, note that a few ports go unchecked also in L7-Responders.

This article has been accepted for publication in IEEE Transactions on Network and Service Management. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TNSM.2023.3267671



(a) Darknet (b) L4-Responders (c) L7-Responders (d) DPIpot

Fig. 5. Number of flows per destination port for the four deployments. The presence of different active responders changes the observed traffic per port.

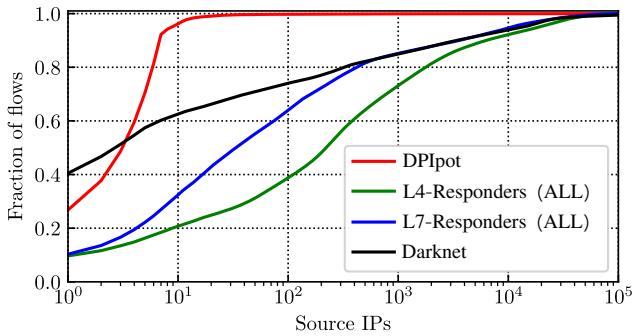
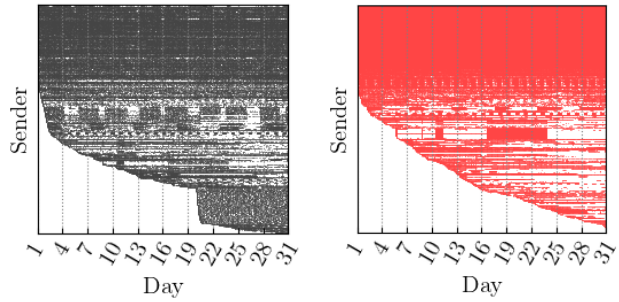


Fig. 6. Fraction of flows per sender IP address.



(a) Darknet (b) DPIpot

Fig. 7. Activity pattern of top-1000 sender IP addresses. Each row corresponds to a sender's IP address.

We observe a completely different picture for DPIpot. Some hundreds of ports get millions of flows, and the remaining ports get some uneven distribution of traffic. Unlike the darknets and L7-Responders, more than 15 000 ports never received any flows, similarly to the L4-Responders case. Recall that, for both L4-Responders and DPIpot cases, all ports result *open* during port scans. We conjecture that either the senders get trapped performing activities on the found open ports, or they are more cautious and abort (or time out) scans after finding a high number of open ports.

Takeaway: Active responders in some ports engage senders, which activate the next stage in their scans or attacks. This increases the traffic and sometimes challenges the monitoring infrastructure. Enabling all ports traps senders in some activities, possibly limiting/biasing their activity.

B. Changes on traffic senders

We now investigate changes seen in the set of senders contacting the deployments. We start by highlighting the last column of Tab. I. The number of unique senders varies substantially across the deployments. In fact, senders increase by around 40% in the L7-Responders when compared to the *Darknet Ext*. Even more interesting *Darknet Int*, i.e., those IP addresses hosted in the same subnet with the responders, observe around 12% more senders than the pure darknet subnet.

We dig into the behavior of these senders in Fig. 6. It reports the cumulative fraction of flows from each sender. The x-axis reports (in log scale) the rank of sender IP addresses according to the volume for each deployment.

In the darknet, the three most active senders generate 40% of flows. These are well-known scanners reported multiple times in blocklists. The top 10 most active senders are responsible for 63% of the flows. Some of them are always active. Some senders probe at a high rate (hitting 20 000 flows per hour) and disappear. We also observe a tail of more than 63 000 IP addresses. This tail is in line with previous work [2] that shows darknet traffic is dominated by bugs and misconfiguration, with only a minority of senders actually performing scans and attacks. Flows are more distributed across senders in L4-Responders and L7-Responders, with the top-10 accounting for 20% and 32% of traffic, respectively.

The figure is completely different in DPIpot where the top-10 most active senders account for 95% of the flows. These senders are involved in RDP abuses observed on non-standard ports. They generate millions of flows per hour, triggering our rate-limiting countermeasures.

Fig. 7 offers a visual representation of the activity of the top 1000 most active senders over time. Each row corresponds to an IP address. A dot is present if that IP address is active at that hour. We register the presence of senders that are active most of the time and the continuous arrival of new senders. For instance, a group of 200 new senders appears on day 20 in the darknet. These senders are likely bots that perform a coordinated scan reaching our address space. In general, we can distinguish different patterns: some senders are persistent, while others keep coming back periodically. A few senders interact only for some time before disappearing and never coming back. The latter is more visible in Fig. 7(b) for DPIpot.

We complement the analysis in Fig. 8, where we investigate scan patterns performed by the top 100 most active senders. We show only the darknet for brevity. The x-axis reports the

This article has been accepted for publication in IEEE Transactions on Network and Service Management. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TNSM.2023.3267671

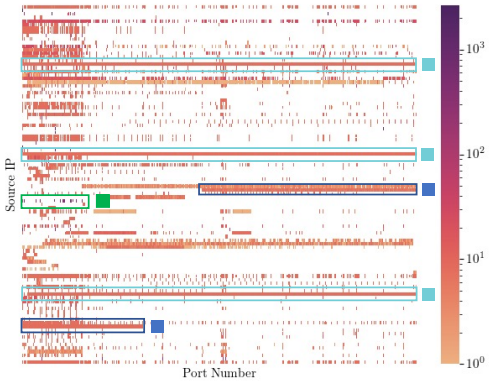


Fig. 8. Top-100 senders vs. destination port. Addresses are sorted numerically.

destination port of flows, sorted by value. Each row refers to a single sender IP address. A dot is present if that IP address sends a flow to such a port. The darker the color, the larger the number of flows.

For readability, we highlight some patterns with colors. First, a few horizontal scanners are visible (cyan). These senders check all ports, even in the darknet. Second, we observe some vertical scanners (green) - i.e., senders that send lots of packets for a few ports. Third, some senders cover a large set of continuous ports, covering a subset of all ports (dark blue). All these patterns are seen for other deployments too, which however show yet other behaviors. For example, besides the horizontal scanners, DPIpot allows us to observe very targeted scans on a few ports, i.e, vertical attacks, and a large number of coordinated scanners, i.e., groups of senders that target the same few ports simultaneously. This has been confirmed by our recent works we leverage embeddings to discover common sender patters [47].

Takeaway: The presence of active responders attracts a new wave of senders, which target also the addresses remaining dark in the subnet. Most of the traffic comes from a few thousand senders that are involved either with vertical or horizontal scans and attacks. Unlike vertical honeypots, DPIpot lets us observe scan patterns where also non-standard ports get the attention of attackers.

VI. Amplification of Service-Specific Deployments

We now focus on the extra information different responders offer compared to darknets We consider our first experiment.

Here, we answer the question: Does the presence of specific services attract traffic to other services? What happens when one deploys services on non-standard ports?

A. Service amplification

To quantify the extra traffic per deployment, we define the *amplification factor* as the ratio between the number of flows seen on a given port(s) for the 8 IP addresses of a specific deployment, and the number of flows directed to the same port(s) on the 8 IP addresses belonging to the *Darknet Ext*.

First, we run a preliminary test to verify whether the amplification factor changes when comparing IP addresses

TABLE III
Amplification for L4-Responders and L7-Responders. Cases in which no amplification is observed are marked with a hyphen.

L4-Responders								
	DB	*File*	*Mail*	*Proxy*	*RD*	*Terminal*	*Web*	*Others*
DB	**15.4**	4.3	–	–	–	–	–	–
File	1.6	**42.0**	–	–	–	–	–	–
Mail	1.5	4.1	**6.5**	–	–	–	–	–
Proxy	1.5	4.2	–	**2.7**	–	1.2	–	1.2
RD	1.5	4.2	–	–	**21.2**	1.6	–	–
Terminal	1.5	4.1	–	–	–	**9.3**	–	1.3
Web	1.5	4.2	1.4	1.2	–	1.3	**8.1**	–

L7-Responders								
	DB	*File*	*Mail*	*Proxy*	*RD*	*Terminal*	*Web*	*Others*
DB	**9.3**	3.9	–	–	–	–	–	–
File	1.6	**116.3**	–	–	–	–	–	–
Mail	1.5	3.6	**9.6**	–	–	–	–	–
Proxy	1.5	3.8	–	**2.8**	–	–	–	1.2
RD	1.5	3.8	–	–	**254.9**	–	–	–
Terminal	1.5	3.6	–	–	–	**46.6**	–	1.2
Web	1.5	3.8	1.2	1.2	–	1.2	**5.3**	–

belonging to the same deployment. For this, we take all groups of 8 sequential IP addresses (/29 subnets) in the *Darknet Ext* and compute – for each destination port – the amplification factor for each group pair. Not reported here for brevity – the distribution of the amplification factors is centered between 0.9 and 1.1. We therefore consider significant any amplification factor outside this range.

Fig. 9 shows the amplification factor for some selected ports. We identify five major behaviors, which we label with capital letters and for which we provide two examples per category:

A) Invariant (around 50 000 ports): the traffic reaching these ports does not change significantly from the darknet to the other deployments. Ports like 2 000 and 6 379 receive only *port scan* attempts, whose volume does not change when responders are present;
B) Homogeneous (around 13 000 ports): senders find possible services on some open ports. These open ports trigger senders to contact several other ports on the host – e.g., ports 2 375 and 2 323 in the figure. However, for these ports, senders do not send any L7 payload - e.g., because waiting for servers to initiate the exchange. Here, L7-Responders and DPIpot behave just like L4-Responders;
C) L7 client-initiated (around 500 ports): these are clear cases of open services on default ports with client-initiated protocols, e.g., SSH and RDP on ports 22

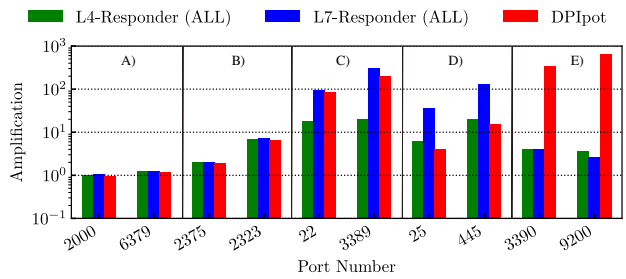
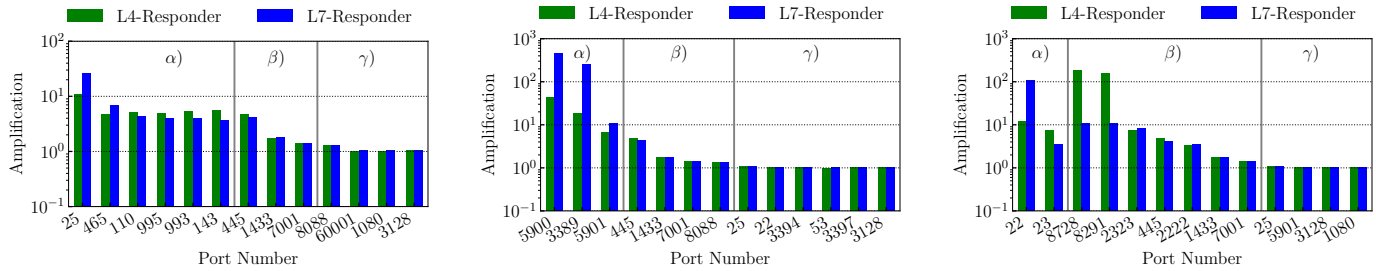


Fig. 9. Amplification factor for the most targeted ports.

This article has been accepted for publication in IEEE Transactions on Network and Service Management. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TNSM.2023.3267671



(a) Mail (b) Remote Desktop (c) Terminal

Fig. 10. Amplification factor for selected deployments. β marks cases of *Side-Scans*.

and 3 389. Both L7-Responders and DPIpot are effective to engage with the senders. Since frequent attacks are present, we observe very large amplification factors. L4-Responders are less interesting for the senders, with reduced amplification factor;

D) L7 server-initiated (around 10 ports): open services on default ports for which the senders expect the server to initiate the L7 exchange. In this case, the L7-Responders vertical honeypots are more effective, while DPIpot behaves as the L4-Responders, e.g., on SMTP and SMB on ports 25 and 445, respectively;

E) Large-scale attacks on non-standard ports (around 1 500 ports): Senders discover services on non-standard ports and perform attacks. Only DPIpot, the only one able to identify the L7-protocol, let us quantify such behavior. In particular, we have witnessed an extensive RDP attack on multiple non-standard ports, resulting in around 1 500 ports for which DPIpot amplification grows to almost $1,000$.

Takeaway: Different deployments amplify different behaviors. Overall, the obtained information clearly grows from case A) to case E). The latter is particularly interesting, showing the effects of performing traffic analysis on non-standard ports. DPIpot leads to the discovery of active attacks on non-standard ports, otherwise unseen with darknets or L4-Responders. A simple darknet would offer a more limited view overall.

B. Targeted services and Side-Scans

So far we evaluated responders that support *All* services at the same time. We now check what happens if we partition responders so that they behave as vertical services. We consider L4-Responders and L7-Responders, with categories/ports/applications defined in Tab. I. For each category, we compute the amplification factor with respect to the corresponding categories/ports/applications in darknet addresses.

Tab. III summarizes results. For each vertical deployment, we report the amplification factor only when significant. Rows report the category of the deployment while columns report the corresponding categories in the *Darknet Ext* as reference. As expected, activating specific services attracts the attention of senders on them (see main diagonal, in bold). L4-Responders suffice to observe more traffic, but L7-Responders clearly generates much more interaction. Exceptionally, L4-Responders see a higher amplification factor than L7-Responders in some cases (e.g., *DB*). This is likely a consequence of our lack of honeypots in the L7 backends (see Tab. I). In this case, while L7-Responders reply with an uninteresting response (reset the connections), the limitation to the TCP handshake offered by L4-Responders further engages the senders.

We observe also significant amplification factors on services for which the deployment does *not* answer, i.e., where we drop the `SYN` packets. Regardless of the deployment, once senders find an IP address that is alive (i.e., hosting a popular service), they target other ports in the *DB* and *File* categories. The case of the *Web* category is particularly interesting: when a service is found active on ports typically hosting HTTP services, senders apparently start targeting multiple other services/ports on the same host. We refer to this phenomenon as *Side-Scan* activity.

Fig. 10 reports some of the most relevant *Side-Scans*.

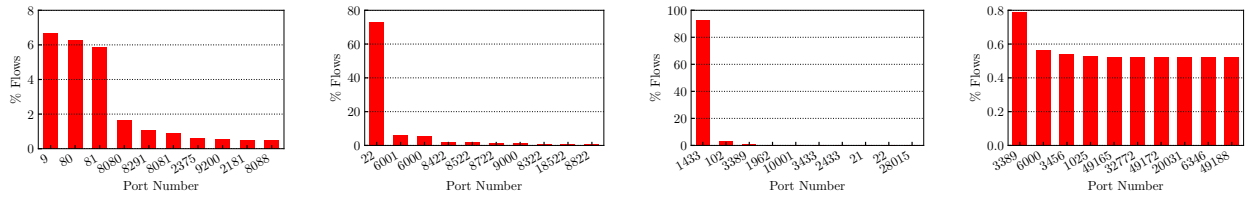
α) marks the well-known (and open) ports for the category. Here as expected we get significant amplification factors, with L7-Responders getting significantly more traffic than L4-Responders for some honeypots.

β) marks those *Side-Scan* ports that suddenly get targeted - despite being blocked for the particular deployment. These are the ports senders target in vertical attacks/scans triggered by a different category. For instance, when opening ports of the *Mail* category (plot in the left-hand side), we observe significant amplification factors on ports (445, 1 433), which are usually used in *File* and *DB* services. We see curious *Side-Scans* also on ports (7 001,8 088). Similar effects can be seen for the *Remote Desktop* category.

More expected, ports (2 222, 2 323) are often used as alternative ports for terminal services – and senders *Side-Scan* these ports when finding standard terminal ports open (rightmost plot). Ports (8 728, 8 291) are known to be vulnerable services in old versions of software routers. We observe frequent "door-knocking" attempts: the sender checks port 22 first; if open but no banner is offered, they check ports (8 728, 8 291). L7-Responders do offer a banner on port 22. Thus, flows on ports (8 728, 8 291) are smaller than in L4-Responders that offers no banner on port 22 [48].

Finally, γ) exemplifies some ports that remain invariant, i.e., they are neither the initial target nor reached in *Side-Scans*. Most ports fall in this class.

This article has been accepted for publication in IEEE Transactions on Network and Service Management. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TNSM.2023.3267671



(a) HTTP (b) SSH (c) MsSQL-TDS (d) RDP

Fig. 11. Flows percentages on top-10 ports for DPIpot and different L7 protocols.

TABLE IV
TOP-5 PROTOCOLS RECOGNIZED IN DPIPOT.

Protocol	Flows	Sender Addr.	Dest. Ports	% of Flows on Standard Ports
RDP	329 652 678	1 415	28 333	0.8
HTTP	444 715	13 705	9 381	6.2
TLS	221 565	2 806	11 999	4.6
SSH	119 698	1 097	187	72.9
MsSQL-TDS	31 596	3 193	448	92.6

Takeaway: We observe high amplification in both L4-Responders and L7-Responders. Deployments targeting a particular service uncover *Side-Scans*, which vary according to the service exposed and the behavior of the responder.

C. DPIpot additional visibility

We now dig into DPIpot data to check the *Side-Scan* phenomena in this case. Tab. IV shows that DPIpot observed a vast majority of RDP flows - with 1 415 senders generating more than 330 M flows in one month. These senders target more than 28 thousand ports, with the standard port 3 389 accounting for only 0.8% of flows. This behavior is also seen in Fig. 5 and Fig. 6 where the IP addresses involved in this attack dominate the traffic DPIpot collects.

DPIpot lets us observe also other popular protocols like HTTP, TLS, and SSH, with multiple senders targeting thousands of ports. Some of these attacks focus mostly on the default port - like SSH or MsSQL-TDS where 72.9% and 92.6% of the flows are to the default ports.

To check how senders choose the port to probe for a given protocol, Fig. 11 details the most popular target ports for some L7 protocols. Start from the HTTP case. Port 9 results as the most popular port. This is a *Side-Scan* performed by an Internet mapping project of the University of Michigan, which targets port 9 (about 30 000 flows) and 7 (about 50 flows only), sending bogus HTTP requests [49]. This scan activity would likely go unnoticed on traditional honeypots. Besides this curious scan, DPIpot recognizes HTTP requests on non-standard ports that it correctly handles. Given the popularity of solutions based on HTTP protocol, it is not surprising to see senders probe open ports with HTTP requests.

Move to SSH now. Here, most flows target port 22. Yet, the senders check other ports where system administrators may move the SSH service, e.g., 8 422, 8 522, 18 522. This behavior suggests a targeted *Side-Scan* where senders generate the port to target with some domain-driven algorithm. The *Side-Scan* using the MsSQL-TDS protocol is even more vertical. Most of the attacks are directed to the default port 1 443, but some few requests go to port 102, likely trying to abuse some Microsoft Exchange service.

At last, the RDP case is worth more details. RDP has become a viable solution for malicious hosts for installing ransomware [50] via attacks that start with password brute-force [51] as well as a common backdoor [52].

Thanks to DPIpot, we observe 1 415 senders performing password brute-force attacks. The attackers however execute the brute force in almost any port announcing RDP support. Fig. 12 shows the targeted ports, ranked per number of received flows. Notice the log-log scale. The step-wise behavior of the figure suggests the presence of a group of 1 000 ports that receive the most requests, followed by a second group of ports that are contacted less frequently. This second group may be due to an initial discovery horizontal scan, after which senders come back to perform the brute-force attack. The inner plot shows that there is also a clear pattern for the top-300 ports. Checking which ports each sender targets, we recognize three macro-categories:

- Senders (around 700) that vertically probe only standard RDP port 3 389 and the immediately adjacent ones;
- Senders that focus on a small group of selected ports (e.g., ports 1 289, 23 390, 1 025, 3 418, 50 000, 554, 3 336) - likely chosen via domain knowledge. The four IP addresses involved in this attack belong to the same network and have never been reported at the time of writing. They generate 3.5 million flows;
- Senders that scan thousand of ports (16 IP addresses). These addresses have been reported as heavy scanners [53] and perform a similar activity. This suggests they are part of the same botnet.

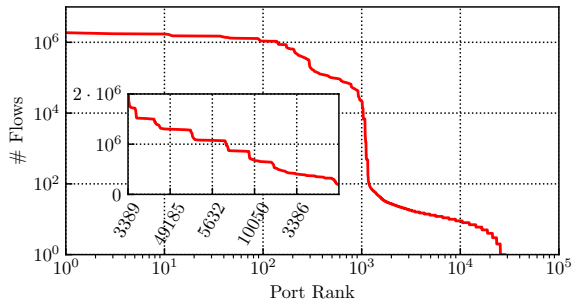
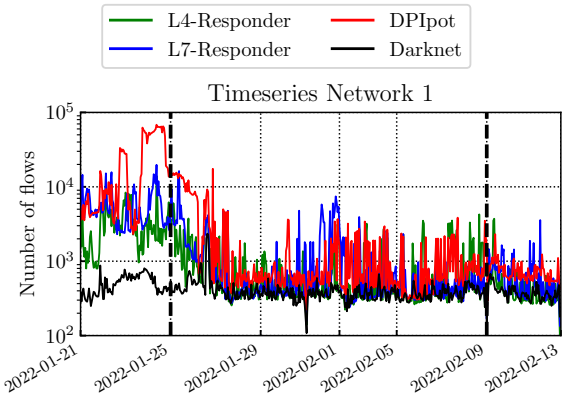
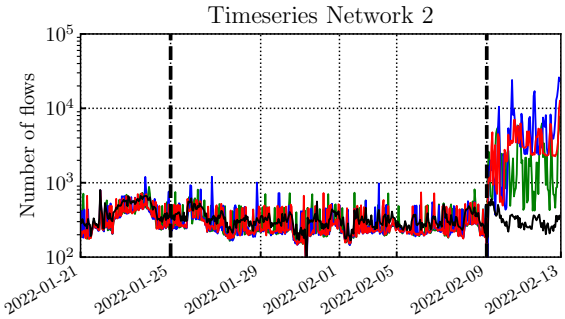


Fig. 12. Flows per port (RDP). Zoom on the first 300 ports in the inner axis.

This article has been accepted for publication in IEEE Transactions on Network and Service Management. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TNSM.2023.3267671



(a) Powering off responders (IP ranges in Tab. I)



(b) Powering up responders (IP ranges in Tab. II)

Fig. 13. Effects of removing and adding active responders in darknets.

Takeaway: DPIpot unveils unexpected *Side-Scan* attacks and scans where senders target non-standard ports. It also triggers activity that L7-Responders in the standard ports do not observe. Senders may become very aggressive, calling for precautions to avoid overloading the monitoring infrastructure.

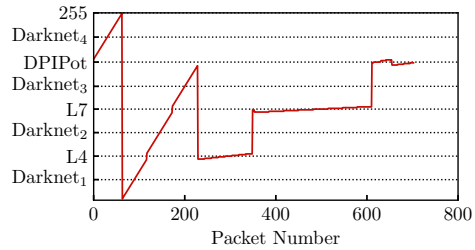
VII. Darkening and Enlightening Networks

We now shift our attention to our second experimental setup, in which we shut all active responders down, before enlightening new active responders in the other darknet.

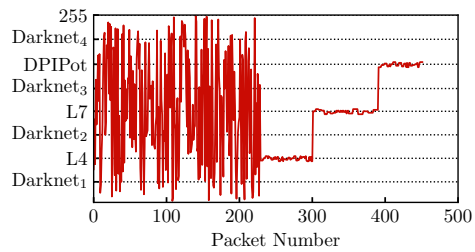
We answer the following questions: Do senders continue to reach IP addresses that once hosted active responders? How fast does a newly-active IP address become a target of the senders unseen in the darknet? How does the deployment of active responders impact neighboring IP addresses?

A. From light to darkness, and back

Fig. 13 describes the traffic evolution for some deployments in our infrastructure. Let us focus first on the deployments that have been shut down. Fig. 13(a) depicts the time series of the number of flows per hour for groups of 8 IP addresses hosting the *Darknet Ext*, L4-Responders (All), L7-Responders (All) and DPIpot in our first experimental setup (see Tab. I). Before the shutdown (first black dashed vertical line) the active responders observe more than 10^3 flows per hour, whereas *Darknet Ext* between 10^2 and 10^3 flows per hour, respectively. The number of flows per hour remains orders of magnitude



(a) Sequential host scan - targeted port scan on responders



(b) Random host scan - targeted port scan on responders

Fig. 14. Example of host discovery patterns observed in the infrastructure.

higher in the active responders when compared to *Darknet Ext* even two days after the responders are down. In fact, the traffic remains noisier for the IP addresses that were hosting the responders for weeks. In sum, the senders that target the active responders insist on reaching these responders, and the traffic does not return to the darknet levels even two weeks after the shutdown.

Focus now on Fig. 13(b) which depicts the deployment of fresh responders in the network that originally hosted *Darknet Ext*. Before the activation of any service, all groups of IP addresses observe the same amount of traffic (10^2–10^3 flows per hour). As soon as we deploy active responders on Feb 9th, 2022, we spot an immediate increase in traffic for all cases. We will show later that this increase is partly caused by a new wave of senders that immediately and suddenly reach each responder to perform an in-depth port scan. This result hints at coordination with those senders that perform initial host discovery. Again, it confirms the advantage of having a deployment that mixes both types of responders.

To shed more light on senders' strategies for port and service scanning, Fig. 14 shows two examples of common patterns observed when the responders are deployed. The figure depicts the sequence of IP addresses a given sender targets over time. On the y-axis, we report the type of darknet/responders on such IP addresses.

The first example of Fig. 14(a) illustrates the behavior of sequential scanners. These scanners sequentially visit every IP address in the /24 subnet to find open services. After this host discovery, they get back to those IP addresses hosting responders to perform in-depth port scans and application attacks. Some scanners start from a random initial IP address (as in Fig. 14(a)), while others start from the first address in the /24 subnet.

This article has been accepted for publication in IEEE Transactions on Network and Service Management. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TNSM.2023.3267671

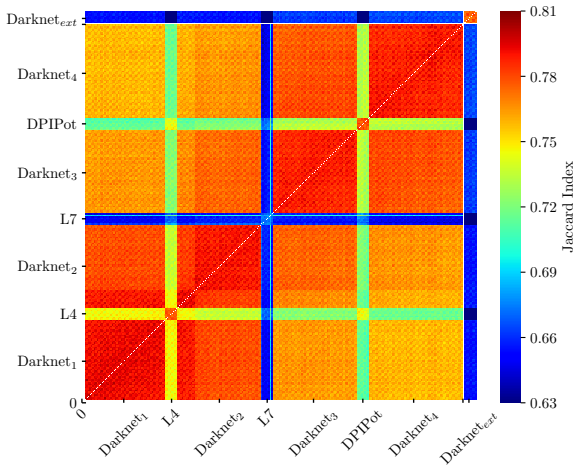


Fig. 15. Jaccard Index among aggressive senders targeting each IP address.

Fig. 14(b) instead shows an example of a scanner that performs a random host scan: these scanners keep contacting random IP addresses in the /24 subnet to perform host discovery. Once this stage is completed, they come back to those responders for in-depth activity, similar to the sequential scanners.

Takeaway: It takes days or even weeks for senders to stop targeting responders that went offline. Conversely, as soon as an IP address is found responding to traffic, it becomes a target of (new) senders almost immediately to perform an in-depth host and service discovery. Senders employ diverse strategies to perform the discover activities.

B. Disturbing the neighbours

We further investigate if the presence of active responders causes disturbance to IP addresses remaining dark in the same /24 subnet. This question is important for those running darknets to understand to what extent active responders pollute the darknet traffic. Recall from previous sections that while active responders do attract more senders, they sometimes trap senders in specific activities, thus biasing senders' behaviors. Here we verify how neighbor addresses are impacted.

Recall from the last column of Tab. I that the number of senders varies substantially across the deployments. We confirm such behavior in our second experiment. For those darknet IP addresses close to active responders, the increase in the number of senders starts immediately after the responders become active.

We now investigate if the additional senders contacting dark addresses are similar to the ones reaching the responders. For this, we compute the Jaccard Index for all pairs of addresses in the /24 subnet where we have deployed fresh responders. To filter out occasional senders, we restrict the analysis to *aggressive senders* – those sources that send at least 100 packets over 1 month.

Fig. 15 shows the Jaccard Index in the form of a heatmap. Overall, the figure shows two main effects: 1) active responders attract a different set of senders, and 2) there is a pollution effect, but not directly nearby the responders.

For 1), notice the low Jaccard Index when comparing active responders with darknet addresses (e.g., the rows/columns corresponding to L4, L7, and DPIpot). This decrease is due to an increase in the number of senders that target *only* the responders (causing an increase in the denominator of the Jaccard Index). This behavior confirms that some senders perform only the "host scan" phase, while other senders become active to perform subsequent phases of attacks, e.g., "port scans", "application scans" and "vulnerability exploitation" on addresses that are found alive.

For 2), darknet addresses at the beginning (end) of the /24 address space tend to observe a higher fraction of senders in common with neighboring addresses (causing an increase in the numerator of the Jaccard Index). This fact is reflected in the darker red pattern seen along the diagonal of the Jaccard Index. This is an effect of the sequential scanners that stop their activity before completing the scan of the entire /24 subnet.

Finally, focusing on the Jaccard Index computed among addresses in the external /24 darknet (top and rightmost groups), we observe a different set of senders. This behavior is due to the set of senders scanning one /24 being different from the set of senders scanning the second /24.

Takeaway: Senders involved in darknet scans are typically different from those seen in subsequent attack stages. These new senders are seen only when active responders are present. Interestingly, the presence of responders attracts new senders also for addresses remaining dark.

VIII. Conclusion

We systematically analyzed the impact of deploying interactive responders on the darknet address space. Our results show the clear benefit of engaging with senders, with more and more interactive responders that allow one to collect richer data on the senders' behaviors. We also showed that a careful design in the deployment, with the ability to turn on and off responders at need, offers even more opportunities, uncovering a new wave of senders that otherwise would remain unobserved.

We show that each deployment has its own benefits, unveiling different activities and bringing new perspectives. Combining the several interaction levels augments visibility. However, deployments may impact each other (e.g., polluting neighboring addresses) and may foster traffic increase to the point of saturating the monitoring infrastructure.

Beyond our findings, several challenges are waiting ahead of such hybrid infrastructures. For example, a large amount of collected information calls for automatic methods for analyzing the data, uncovering correlations between deployments, fingerprinting senders and, ultimately, identifying the rise of novel scans and cyber threats. Distributing our active responders to other IPv4 ranges, IPv6 networks, and different geographical locations is also a challenge that we will face in future work.

Acknowledgement

This work was partially supported by Huawei R&D Center (France), the SmartData@PoliTO center for Big Data technologies and the project SERICS (PE00000014) under the MUR National Recovery and Resilience Plan funded by the European Union - NextGenerationEU.

References

[1] C. Fachkha and M. Debbabi, "Darknet as a Source of Cyber Intelligence: Survey, Taxonomy, and Characterization," *Commun. Surveys Tuts.*, vol. 18, no. 2, pp. 1197–1227, 2016.

[2] K. Benson, A. Dainotti, K. Claffy, A. Snoeren, and M. Kallitsis, "Leveraging Internet Background Radiation for Opportunistic Network Analysis," in *Proceedings of the ACM Internet Measurement Conference*, ser. IMC'15, 2015, pp. 423–436. [Online]. Available: http://dl.acm.org/citation.cfm?doid=2815675.2815702

[3] E. Wustrow, M. Karir, M. Bailey, F. Jahanian, and G. Huston, "Internet Background Radiation Revisited," in *Proceedings of the 10th ACM SIGCOMM Conference on Internet Measurement*, ser. IMC'10, 2010, pp. 62–74. [Online]. Available: http://portal.acm.org/citation.cfm?doid=1879141.1879149

[4] A. Dainotti, A. King, K. Claffy, F. Papale, and A. Pescape, "Analysis of a "/0" Stealth Scan From a Botnet," *IEEE/ACM Trans. Netw.*, vol. 23, no. 2, pp. 341–354, 2015.

[5] M. S. Pour, D. Watson, and E. Bou-Harb, "Sanitizing the iot cyber security posture: An operational cti feed backed up by internet measurements," in *2021 51st Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN)*. IEEE, 2021, pp. 497–506.

[6] M. Antonakakis, T. April, M. Bailey, M. Bernhard, E. Bursztein, J. Cochran, Z. Durumeric, J. Halderman, L. Invernizzi, M. Kallitsis, D. Kumar, C. Lever, Z. Ma, J. Mason, D. Menscher, C. Seaman, N. Sullivan, K. Thomas, and Y. Zhou, "Understanding the Mirai Botnet," in *Proceedings of the 26th USENIX Security Symposium*, ser. USENIX Security'17, 2017, pp. 1093–1110. [Online]. Available: https://www.usenix.org/conference/usenixsecurity17/technical-sessions/presentation/antonakakis

[7] L. Metongnon and R. Sadre, "Beyond Telnet: Prevalence of IoT Protocols in Telescope and Honeypot Measurements," in *Proceedings of the 2018 Workshop on Traffic Measurements for Cybersecurity*, ser. WTMC'18, 2018, pp. 21–26. [Online]. Available: http://dl.acm.org/citation.cfm?doid=3229598.3229604

[8] M. Jonker, A. King, J. Krupp, C. Rossow, A. Sperotto, and A. Dainotti, "Millions of Targets Under Attack: A Macroscopic Characterization of the DoS Ecosystem," in *Proceedings of the ACM SIGCOMM Internet Measurement Conference*, ser. IMC'17, 2017, pp. 100–113. [Online]. Available: http://dl.acm.org/citation.cfm?doid=3131365.3131383

[9] P. Richter and A. Berger, "Scanning the Scanners: Sensing the Internet from a Massively Distributed Network Telescope," in *Proceedings of the Internet Measurement Conference*, ser. IMC'19, 2019, pp. 144–157. [Online]. Available: http://dl.acm.org/doi/10.1145/3355369.3355595

[10] F. Soro, I. Drago, M. Trevisan, M. Mellia, J. Ceron, and J. J. Santanna, "Are Darknets All The Same? On Darknet Visibility for Security Monitoring," in *Proceedings of the IEEE International Symposium on Local and Metropolitan Area Networks*, ser. LANMAN, 2019, pp. 1–6. [Online]. Available: https://ieeexplore.ieee.org/document/8847113/

[11] R. Hiesgen, M. Nawrocki, A. King, A. Dainotti, T. C. Schmidt, and M. Wählisch, "Spoki: Unveiling a new wave of scanners through a reactive network telescope," in *Usenix Security Symposium 2022*. USENIX Association, 2022.

[12] D. Moore, C. Shannon, D. Brown, G. Voelker, and S. Savage, "Inferring Internet Denial-of-Service Activity," *ACM Trans. Comput. Syst.*, vol. 24, no. 2, pp. 115–139, 2006.

[13] C. Fachkha, E. Bou-Harb, and M. Debbabi, "Inferring Distributed Reflection Denial of Service Attacks from Darknet," *Comput. Commun.*, vol. 62, no. C, pp. 59–71, 2015.

[14] A. Dainotti, K. Benson, A. King, B. Huffaker, E. Glatz, X. Dimitropoulos, P. Richter, A. Finamore, and A. Snoeren, "Lost in Space: Improving Inference of IPv4 Address Space Utilization," *IEEE Journal on Selected Areas in Communications*, vol. 34, no. 6, pp. 1862–1876, 2016.

[15] A. Dainotti, C. Squarcella, E. Aben, K. Claffy, M. Chiesa, M. Russo, and A. Pescape, "Analysis of Country-Wide Internet Outages Caused by Censorship," *IEEE/ACM Trans. Netw.*, vol. 22, no. 6, pp. 1964–1977, 2014.

[16] Z. Durumeric, M. Bailey, and J. Halderman, "An Internet-Wide View of Internet-Wide Scanning," in *Proceedings of the 23rd USENIX Conference on Security Symposiu*, ser. SEC'14, 2014, pp. 65–78. [Online]. Available: http://dl.acm.org/citation.cfm?id=2671225.2671230

[17] E. Raftopoulos, E. Glatz, X. Dimitropoulos, and A. Dainotti, "How Dangerous Is Internet Scanning? A Measurement Study of the Aftermath of an Internet-Wide Scan," in *Proceedings of the 7th Workshop on Traffic Monitoring and Analysis*, ser. TMA'15, 2015, pp. 158–172. [Online]. Available: http://link.springer.com/10.1007/978-3-319-17172-2_11

[18] S. Staniford, D. Moore, V. Paxson, and N. Weaver, "The Top Speed of Flash Worms," in *Proceedings of the ACM Workshop on Rapid Malcode*, ser. WORM'04, 2004. [Online]. Available: http://portal.acm.org/citation.cfm?doid=1029618.1029624

[19] CAIDA/UCSD, "The ucsd network telescope," 2021. [Online]. Available: https://www.caida.org/projects/network_telescope/

[20] F. Soro, M. Allegretta, M. Mellia, I. Drago, and L. Bertholdo, "Sensing the Noise: Uncovering Communities in Darknet Traffic," in *Proceedings of the Mediterranean Communication and Computer Networking Conference*, ser. MedComNet, 2020, pp. 1–8. [Online]. Available: https://ieeexplore.ieee.org/document/9191555/

[21] GreyNoise, 2021. [Online]. Available: https://greynoise.io/

[22] M. Bailey, E. Cooke, F. Jahanian, A. Myrick, and S. Sinha, "Practical Darknet Measurement," in *Proceedings of the 40th Annual Conference on Information Sciences and Systems*, ser. CISS'06, 2006, pp. 1496–1501. [Online]. Available: http://ieeexplore.ieee.org/document/4068042/

[23] J. Czyz, K. Lady, S. Miller, M. Bailey, M. Kallitsis, and M. Karir, "Understanding IPv6 Internet Background Radiation," in *Proceedings of the 13th ACM Internet Measurement Conference*, ser. IMC'13, 2013, pp. 105–118. [Online]. Available: http://dl.acm.org/citation.cfm?doid=2504730.2504732

[24] Honeynet, "The honeynet project," 2021. [Online]. Available: https://www.honeynet.org/

[25] TPot, "The all in one honeypot platform," 2021. [Online]. Available: https://github.com/telekom-security/tpotce

[26] E. De Cristofaro, A. Friedman, G. Jourjon, M. A. Kaafar, and M. Z. Shafiq, "Paying for likes? understanding facebook like fraud using honeypots," in *Proceedings of the 2014 Conference on Internet Measurement Conference*, ser. IMC '14, New York, NY, USA, 2014, pp. 129–136.

[27] S. Liebergeld, M. Lange, and R. Borgaonkar, "Cellpot: A concept for next generation cellular network honeypots," *Internet Society*, pp. 1–6, 2014.

[28] W. Han, Z. Zhao, A. Doupé, and G.-J. Ahn, "Honeymix: Toward sdn-based intelligent honeynet," in *Proceedings of the 2016 ACM International Workshop on Security in Software Defined Networks & Network Function Virtualization*, ser. SDN-NFV Security'16, New York, NY, USA, 2016, p. 1–6.

[29] A. Vetterl and R. Clayton, "Bitter harvest: Systematically fingerprinting low- and medium-interaction honeypots at internet scale," in *Proceedings of the 12th USENIX Workshop on Offensive Technologies (WOOT 18)*. Baltimore, MD, USA: USENIX Association, 2018. [Online]. Available: https://www.usenix.org/conference/woot18/presentation/vetterl

[30] S. Morishita, T. Hoizumi, W. Ueno, R. Tanabe, C. Gañán, M. J. G. van Eeten, K. Yoshioka, and T. Matsumoto, "Detect me if you... oh wait. an internet-wide view of self-revealing honeypots," in *Proceedings of the IFIP/IEEE Symposium on Integrated Network and Service Management (IM)*, 2019, pp. 134–143.

[31] M. Nawrocki, M. Wählisch, T. Schmidt, C. Keil, and J. Schönfelder, "A Survey on Honeypot Software and Data Analysis," *arXiv:1608.06249*, 2016.

[32] V. Yegneswaran, P. Barford, and D. Plonka, "On the design and use of internet sinks for network abuse monitoring," in *International Workshop on Recent Advances in Intrusion Detection*. Springer, 2004, pp. 146–165.

[33] P. Barford, Y. Chen, A. Goyal, Z. Li, V. Paxson, and V. Yegneswaran, "Employing honeynets for network situational awareness," in *Cyber situational awareness*. Springer, 2010, pp. 71–102.

[34] E. Alata, V. Nicomette, M. Kaâniche, M. Dacier, and M. Herrb, "Lessons learned from the deployment of a high-interaction honeypot," in *2006 Sixth European Dependable Computing Conference*. IEEE, 2006, pp. 39–46.

[35] C. F. Torres, M. Steichen, and R. State, "The art of the scam: Demystifying honeypots in ethereum smart contracts," in *28th USENIX Security Symposium (USENIX Security 19)*, Santa Clara, CA, 2019, pp. 1591–1607. [Online]. Available: https://www.usenix.org/conference/usenixsecurity19/presentation/ferreira

[36] J. Thom, Y. Shah, and S. Sengupta, "Correlation of cyber threat intelligence data across global honeypots," in *Proceedings of the IEEE 11th*

Annual Computing and Communication Workshop and Conference, ser. CCWC, pp. 0766–0772.

[37] A. Brzeczko, A. S. Uluagac, R. Beyah, and J. Copeland, "Active deception model for securing cloud infrastructure," in *Proceedings of the IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, 2014, pp. 535–540.

[38] Cowrie, "Ssh/telnet honeypot," 2021. [Online]. Available: https://github.com/cowrie/cowrie

[39] Glutton, "Generic low interaction honeypot," 2021. [Online]. Available: https://github.com/mushorg/glutton

[40] Honeytrap, "Advanced honeypot framework," 2021. [Online]. Available: https://github.com/honeytrap/honeytrap

[41] L. Deri, M. Martinelli, T. Bujlow, and A. Cardigliano, "nDPI: Open-source High-speed Deep Packet Inspection," in *Proceedings of the International Wireless Communications and Mobile Computing Conference*, ser. IWCMC, 2014, pp. 617–622.

[42] T. Rescio, T. Favale, F. Soro, M. Mellia, and I. Drago, "Dpi solutions in practice: Benchmark and comparison," in *Proceedings of the IEEE Security and Privacy Workshops (SPW)*, 2021, pp. 37–42.

[43] Heralding, "Credentials catching honeypot," 2021. [Online]. Available: https://github.com/johnnykv/heralding

[44] SNARE/TANNER, "Web application honeypot sensor," 2021. [Online]. Available: http://mushmush.org/

[45] Twisted, "Event-driven networking engine written in python," 2021. [Online]. Available: https://twistedmatrix.com/trac/

[46] P. Sokol, J. Míšek, and M. Husák, "Honeypots and honeynets: Issues of privacy," vol. 2017, no. 1, p. 4.

[47] L. Gioacchini, L. Vassio, M. Mellia, I. Drago, Z. B. Houidi, and D. Rossi, "Darkvec: Automatic analysis of darknet traffic with word embeddings," in *Proceedings of the 17th International Conference on Emerging Networking EXperiments and Technologies*, ser. CoNEXT '21. New York, NY, USA: Association for Computing Machinery, 2021, p. 76–89. [Online]. Available: https://doi.org/10.1145/3485983.3494863

[48] V. Riyadi, "Securing mikrotik router," 2018. [Online]. Available: https://mum.mikrotik.com/presentations/ID18/presentation_5554_1540255240.pdf

[49] U. of Michigan, "Why am i receiving connection attempts from the university of michigan?" 2013. [Online]. Available: https://cse.engin.umich.edu/about/resources/connection-attempts/

[50] Z. Wang, C. Liu, J. Qiu, Z. Tian, X. Cui, and S. Su, "Automatically traceback rdp-based targeted ransomware attacks," *Wireless Communications and Mobile Computing*, vol. 2018, 2018.

[51] M. Boddy, B. Jones, and M. Stockley, "Rdp exposed-the threat that's already at your door," *Sophos, Inc, Sophos White Paper*, 2019.

[52] T. Bai, H. Bian, M. A. Salahuddin, A. Abou Daya, N. Limam, and R. Boutaba, "Rdp-based lateral movement detection using machine learning," *Computer Communications*, vol. 165, pp. 9–19, 2021.

[53] VirusTotal, 2021. [Online]. Available: https://www.virustotal.com/

BIOGRAPHIES

Francesca Soro is a Scientist at the Austrian Institute of Technology in Vienna. Her research interests focus on the application of Machine Learning to the cybersecurity and network traffic monitoring fields, in particular, on Anomaly detection applications for cybersecurity in critical infrastructures and cyber-physical systems. She is also an Instructor for the IAEA training courses in Computer Security for Nuclear Security. Soro received her Ph.D. Degree at Politecnico di Torino, Italy, in the Electrical, Electronics, and Communications Engineering.

Thomas Favale is a technical team leader in cybersecurity at WSENSE. He got his Ph.D. in Electronic and Telecommunication at Politecnico di Torino. His research interests focus on traffic anonymization and cybersecurity.

Danilo Giordano (S'22), Ph.D., is an Assistant Professor at Politecnico di Torino and member at the SmartData@Polito lab. His research interests focus on data analytics in Small Data and Big Data environments using statistical and Machine Learning (ML) techniques. In particular, he is interested in the development and application of ML in the context of network measurements and predictive maintenance and study future developments in shared mobility in smart cities. He has co-authored more than 40 conference and journal papers and is a member of the editorial board of the Computer Network journal. He was awarded the best student paper award at the ITC conference and the IETF Applied Networking Research Prize in 2016.

Idilio Drago is an Associate Professor at the University of Turin, Italy, in the Computer Science Department. His research interests include network security, machine learning, and Internet measurements. He is particularly interested on how big data and machine learning can help to extract knowledge from network data and help secure the network and automate network management tasks. Drago has a Ph.D. in computer science from the University of Twente, the Netherlands. He was awarded an Applied Networking Research Prize in 2013 by the IETF/IRTF for his work on cloud storage traffic analysis.

Tommaso Rescio is a Network Engineer at Google, Warsaw, Poland. He collaborated on this work while receiving a scholarship from the Consortium GARR, Italy, as well as a Master's Student at the Computer and Control Engineering Department of Politecnico di Torino, Italy. His work approaches the application of Deep Packet Inspection (DPI) techniques to the improvement of network honeypot systems.

Marco Mellia (F'21) is a full professor at the Computer and Control Engineering Department of Politecnico di Torino, Italy, where he coordinates the SmartData@PoliTO centre, an interdisciplinary lab involving more than 50 researchers with a focus on Machine Learning and Data Science and applications to network management, cybersecurity, smart cities and predictive maintenance. He has co-authored over 250 papers published in international journals and presented at leading conferences. He won the IRTF ANR Prize at IETF-88, and the best paper awards at IEEE P2P'12, ACM CoNEXT'13, IEEE ICDCS'15, ACM CCR'16, ITC'18. He is the Editor in Chief of the Proceedings of the Proceedings of ACM on Networking.

This article has been accepted for publication in IEEE Transactions on Network and Service Management. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TNSM.2023.3267671

Zied Ben Houidi is a Principal AI Researcher in the Huawei Paris Research Center working on the intersection of NLP and networks with applications to network control, data analysis and security among others. He received his PhD from Université Pierre et Marie Curie in France in 2010 while he was working at Orange Labs on data-driven performance analysis of core networks' routing protocols. He then joined Bell Labs, the research arm of Nokia, where he proposed and led various research projects on network data valorization (e.g. human-level behavior analytics) as well as automated reasoning for standards specification. The projects led to several deployments, patents, and demos as well as publications in top-tier venues.

Dario Rossi is the Director of Huawei AI4NET Lab and Director of the DataCom Lab at the Paris Research Center, France. Before joining Huawei in 2018, he held Full Professor positions at Telecom Paris and Ecole Polytechnique and was the holder of Cisco's Chair NewNet@Paris. He has coauthored 20+ patents and 200+ papers with 7500+ citations in leading conferences and journals. He is a Senior Member of IEEE and ACM and received 9 best paper awards, the Google Faculty Research Award (2015) and the IRTF Applied Network Research Prize (2016).