Schweizerische Eidgenossenschaft
Confédération suisse
Confederazione Svizzera
Confederaziun svizra

Federal IT Steering Unit FITSU
Federal Intelligence Service FIS

Reporting and Analysis Centre for Information Assurance MELANI
https://www.melani.admin.ch/

TECHNICAL ANNEX TO SEMI-ANNUAL REPORT 2019/2

SELECTED PUBLICATIONS PUBLISHED IN 2019

30 APRIL 2020
REPORTING AND ANALYSIS CENTRE FOR INFORMATION ASSURANCE MELANI
https://www.melani.admin.ch/

1 Content

2 Preface

In the past some readers of the MELANI report asked us about more in depth, technical articles. We therefore decided to publish for the first time a technical annex with this semi-annual MELANI report 2019/2.

In this first issue the GovCERT, which is part of MELANI, publishes an article on the analysis of the Trickbot botnet. In their article, they look at dropper and payload timestamps and the infrastructure.

We could also gain the CyberDefence Campus (CYD) to publish some articles for this first technical annex. The CYD Campus was founded in January 2019 by the Federal Office for Defence Procurement armasuisse and is an important part in the governmental Cybersecurity ecosystem. In the article "To share or not to share: a behavioral perspective on human participation in security information sharing" the authors propose a behavioral framework to theorize on how and why human behavior and security information sharing are associated. The OpenSky Report 2019 focuses on the analysis of Traffic Alert and Collision Avoidance System (TCAS) in real world scenarios. The authors analyzed 250 billion aircraft transponder messages received from 126'700 aircrafts through the OpenSky network. The last article discusses machine learning based detection of command and control channels with a focus on the Locked Shields Cyber Defence Exercise. For the readers that still urge for more information we also listed links to other publications from the CYD campus in chapter 5.

In the future, we plan to include other organizations to publish their technical articles. If you or your organization are interested, please contact us for more information. We hope that this first technical annex meets our tech savvy audiences expectations and we are happy to receive your feedback.

Trickbot – An analysis of data collected from the botnet

GovCERT.ch

September 20, 2019

1 Introduction

We are monitoring various threats and in that context we have collected quite some data about the Trickbot botnet in the past few years. This paper is based on an analysis of selected aspects of our Trickbot data collection. Some of our analysis is rather straightforward, yet, we also take the freedom to make some speculative statements, which might turn out to be debatable or plain wrong. In that spirit we are open for discussions and are happy to receive comments by the readers of this article.

Our analysis consists of two main parts. In the first part we consider the PE timestamps of Trickbot droppers (i.e., the binaries being distributed by the Trickbot operators) and of the respective payloads (i.e., the PE binaries which are unpacked and then executed once a dropper is executed). The analysis is based on a collection of approximately 2100 droppers and corresponding payloads which were collected between July 2016 and February 2019. The main insights from this analysis are:

- The PE timestamp of many trickbot droppers is backdated, while the PE timestamp of the payloads is unmodified and thus reflects the actual production time of samples.
- The same payload is re-packed over and over again into different droppers. We have observed up to 69-fold repacking.
- The working times of the operators is consistent with working hours in the Moscow time zone.
- The production of Trickbot binaries is likely operated by humans, and thus not fully automated.

In the second second part we analyse a collection of Trickbot config files which we have collected by emulating the protocol over a period of 4-5 months end of 2018 beginning of 2019. The config files contain information on the Trickbot infrastructure such as exfiltration sites used by different stealer modules, the first level C2 infrastructure, etc., as well as lists of targeted financial institutions.

The main insights from this analysis are:

- There is a sequence visible in two configuration types (static injects and mailconf) that shows that the attackers are regularly exchanging these infrastructure elements.
- The sequence is less clear in the main configuration file where we can observe some temporal overlapping of the C2 servers.

- The lifetime of how long a C2 server remains in service varies. The C2 servers in the main config are used only for a short time (with some exceptions) and the C2 servers from the static inject and mailconf file are used for a longer period.
- This leads to the conclusion that the attackers are actively managing their infrastructure by exchanging the C2 servers on a regular base.
- We also extracted the targets from the configuration files and observed that the main targets are banks in the US, Great Britain, Ireland and Germany. Interestingly, German targets were added during our analysis period in the month of November.

2 An Analysis of Dropper and Payload Timestamps

As many malware families, Trickbot is delivering its samples (“droppers”) in packed form. The effective “payload” is contained within the dropper and unpacked upon execution. The payloads are also in PE format and can be easily recovered using simple memory dumping and PE restoration techniques.

Our subsequent analysis is based on a collection of approximately 2100 droppers and corresponding payloads which were collected between July 2016 and February 2019. For each dropper we consider three different timestamps: The *dropper's PE timestamp*, the corresponding *payload's PE timestamp*, and the *first-seen timestamp* of the dropper as reported by VirusTotal [1] and / or Abuse.CH [2] (if both services report a first-seen timestamp for a sample, we choose the earlier of the two).

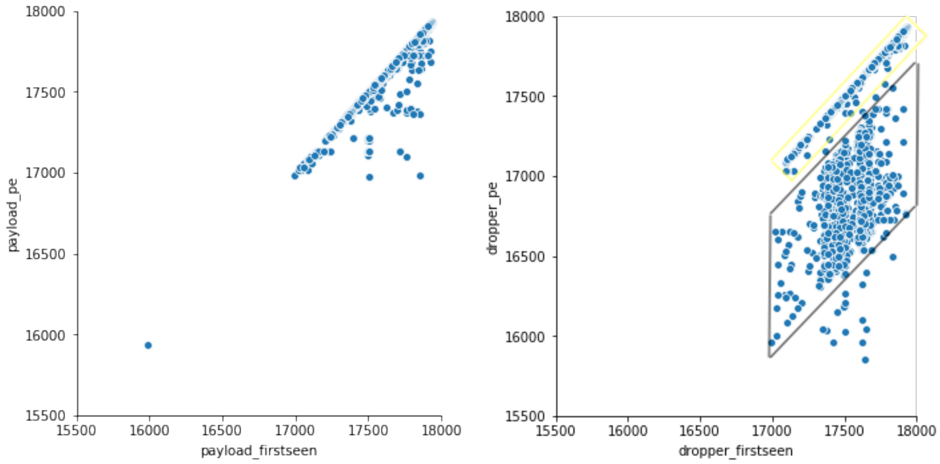
2.1 Backdated Droppers and Unmodified Payload Timestamps

For non-targeted malware that is distributed through spam waves (such as the Trickbot family), we would expect that the first-seen time of a sample is a reasonable estimate for the time when a sample was released into the wild. Further assuming that samples are produced shortly before being released into the wild, we can expect the first-seen times to approximate the production times (i.e., the PE timestamps) of samples.

In the following we use the first seen timestamps to analyze whether, and if so, to what extent Trickbot payload and dropper PE timestamps are forged. Figure 1 compares the first seen times with the PE timestamps of droppers and payloads.

First we look at the relation between *payload PE timestamps* vs. first-seen timestamps in Figure 1(a). Our interpretation of this figure is that the *payload PE timestamps are not backdated*, i.e., that the PE payload timestamps correspond to the actual compilation times of the payloads. The reason is that the distribution in the plot corresponds to what one would expect from a random process such as the collection of malware samples using honeypots. In fact, we see that most samples are caught relatively soon (first seen timestamp is roughly equal to the PE timestamp) and the number of samples that survive longer in the wild is falling off quickly.

Next we consider the *dropper PE timestamps* in Figure 1(b). The figure suggests that there are two type of droppers: those that are not backdated (the “yellow samples” in the figure) and those that are backdated by roughly 300 - 1000 days (the “black samples” in the figure). One could argue that the “black samples” are not backdated samples but rather just samples that go undetected in the wild for a longer time. We do not think so because there



(a) Payload PE timestamp vs. first-seen times from threat feeds.

(b) Dropper PE timestamp vs. first-seen times from threat feeds.

Figure 1: PE timestamps vs. first-seen dates for droppers and payloads (measured in days since 1970).

is a time gap between the yellow and black samples. As mentioned earlier, catching samples in the wild is a random process probably following a Poisson distribution. The existence of the gap is not consistent with such a random process. A much more plausible explanation is that gap is caused by backdating the black droppers.

Further evidence for dropper backdating. There is another observation that strengthens the backdating hypothesis for droppers and the "non-modification hypothesis" for payloads. The earliest published research (we have found) mentioning the Trickbot family dates back to fall 2016 [3, 4]. This research suggests that the inception of the Trickbot family likely dates back to summer or fall 2016.

We have looked at the timestamps of the samples mentioned in the research reports, and they support our observations: In fact, the Malwarebytes [3, 4] article contains hashes of a dropper[1] and payload [2] pair. The respective timestamps of dropper and payload are `09.03.15 00:49` and `11.10.16 19:04`. The payload timestamp is consistent with the conjectured inception date of the Trickbot family and thus seems not to be backdated. On the other hand, the dropper timestamp dates back to spring 2015 way before the family's conjectured inception date and is therefore likely backdated.

In a nutshell, we believe that payload PE timestamps reflect the actual production time of the payloads. Concerning droppers, it seems that there are roughly two categories of droppers. Namely those that are backdated by several hundred of days and those that are not backdated.

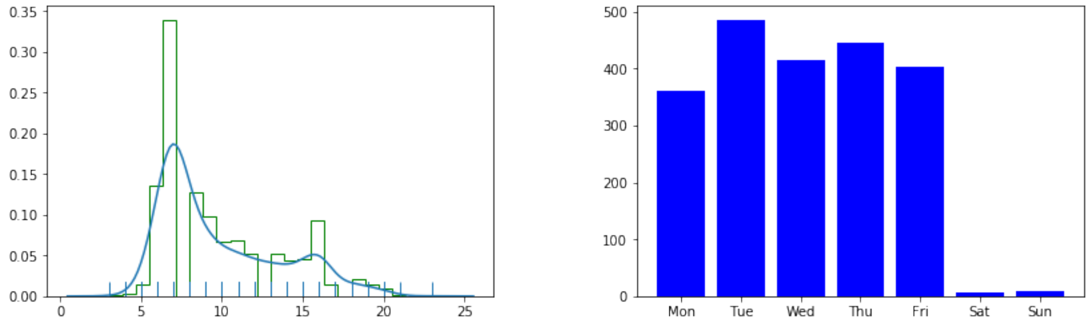
[1] `f26649fc31ede7594b18f8cd7cdbbc15`

[2] `f24384228fb49f9271762253b0733123`

2.2 Working Days and Hours of Trickbot Operators

Under the assumption that the payload PE timestamps reflect the actual production dates we can try to establish the hours of activity of the operators producing Trickbot samples. To this end we have plotted the distribution of the hours found in the payload PE headers in Figure 3(a). The plot clearly shows periods of activity and periods of rest. The lifetime of these periods matches rather well with a human's activity and rest periods. We thus conclude that the production of new samples is not entirely automated but rather performed by humans. Figure 3(b) shows the number of samples produced on different days of the week. This is again highly consistent with human working habits: most weekends are off, slight under-productivity on Mondays etc.

Timestamps have been used to "determine" the timezone of malware operators in the past [7]. There is inherent uncertainty of a couple of hours in such attributions, due to the fact that the malware operators can be early birds, late risers etc. (assuming that cybercrime operations allow for flexible working hours). Moreover, PE timestamps can be modified at will. Yet, Trickbot has been attributed to Eastern actors in several publications [5, 6]. We believe that the working hours in our plot seem to be compatible with this attributions. For instance, the period of rest which is 22h - 3h in UTC time, translates into a period of rest from 1h - 6h in UTC+3 which e.g., corresponds to Moscow's timezone.



(a) PE timestamps of payloads (in hours UTC) on x-axis, relative frequency on y-axis.

(b) PE timestamps of payloads grouped by weekdays on x-axis, absolut frequencies on y-axis.

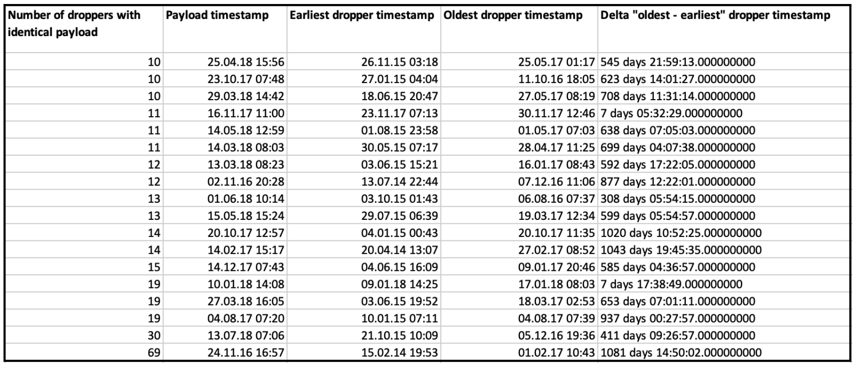
Figure 2: PE timestamps vs. first-seen dates for droppers and payloads.

2.3 Repackaging of Payloads

A widely known technique to avoid AV detection is to pack the same malware sample using different variants of packing algorithms resulting in different binaries which are deployed in the wild. We were wondering whether we could find signs of payload packing in our Trickbot data set. To this end we have clustered droppers that contain the same[3] payload. An excerpt

[3] We consider payloads to be equal when they have same PE timestamp. Since we unpack payloads from memory the resulting payloads are not identical, i.e., they do not have the same hash value. For a selected few samples we have verified manually using binary diffing techniques that payloads with the same PE timestamps contain essentially equivalent code, and that the code of payloads with different PE timestamps has lower similarity.

from the results is shown in Figure 5. The results clearly confirm that the Trickbot operators are practicing repacking.



Number of droppers with identical payload	Payload timestamp	Earliest dropper timestamp	Oldest dropper timestamp	Delta "oldest - earliest" dropper timestamp
10	25.04.18 15:56	26.11.15 03:18	25.05.17 01:17	545 days 21:59:13.000000000
10	23.10.17 07:48	27.01.15 04:04	11.10.16 18:05	623 days 14:01:27.000000000
10	29.03.18 14:42	18.06.15 20:47	27.05.17 08:19	708 days 11:31:14.000000000
11	16.11.17 11:00	23.11.17 07:13	30.11.17 12:46	7 days 05:32:29.000000000
11	14.05.18 12:59	01.08.15 23:58	01.05.17 07:03	638 days 07:05:03.000000000
11	14.03.18 08:03	30.05.15 07:17	28.04.17 11:25	699 days 04:07:38.000000000
12	13.03.18 08:23	03.06.15 15:21	16.01.17 08:43	592 days 17:22:05.000000000
12	02.11.16 20:28	13.07.14 22:44	07.12.16 11:06	877 days 12:22:01.000000000
13	01.06.18 10:14	03.10.15 01:43	06.08.16 07:37	308 days 05:54:15.000000000
13	15.05.18 15:24	29.07.15 06:39	19.03.17 12:34	599 days 05:54:57.000000000
14	20.10.17 12:57	04.01.15 00:43	20.10.17 11:35	1020 days 10:52:25.000000000
14	14.02.17 15:17	20.04.14 13:07	27.02.17 08:52	1043 days 19:45:35.000000000
15	14.12.17 07:43	04.06.15 16:09	09.01.17 20:46	585 days 04:36:57.000000000
19	10.01.18 14:08	09.01.18 14:25	17.01.18 08:03	7 days 17:38:49.000000000
19	27.03.18 16:05	03.06.15 19:52	18.03.17 02:53	653 days 07:01:11.000000000
19	04.08.17 07:20	10.01.15 07:11	04.08.17 07:39	937 days 00:27:57.000000000
30	13.07.18 07:06	21.10.15 10:09	05.12.16 19:36	411 days 09:26:57.000000000
69	24.11.16 16:57	15.02.14 19:53	01.02.17 10:43	1081 days 14:50:02.000000000

Figure 3: Repacking of payloads. Table shows clusters of droppers (which are different) but which contain the same payload once unpacked.

We have also included the timestamps of the payload as well as of the earliest and oldest dropper containing the payload. The table further confirms our previous analysis: it clearly shows that in many but not all cases the same payload is packed into droppers whose timestamps vary considerably due to backdating.

2.4 Trickbot Production Cycles?

In this last and possibly most speculative part of our PE analysis we are comparing dropper and payload PE timestamps. Naively, we would expect that payloads are produced / compiled first, and then packed, resulting in the dropper containing the payload. As a consequence we would expect that dropper PE timestamps are somewhat older than the payload PE timestamps, and that the difference in timestamps reflects the *production time* of a Trickbot sample.

Figure 4 compares the PE timestamps of droppers and payloads. The plot reveals roughly two groups of samples. Those that fall into the "green region" and those that fall into the "red region". The red region consists of the samples whose droppers are backdated (see our discussion above). This region is useless for our analysis of production times. The samples in the "green region" are those whose payload and dropper are roughly produced around the same time. These are thus the samples that are fit for a production time analysis.

The table in Figure 5 shows the distribution of production times of the "green samples". For a total of 838 samples (which corresponds to $\sim 39\%$ of our sample set) we found a production time in the range of $0h - 24h$.

We did not come up with a conclusive analysis of the numbers in Figure 5. The samples in the $0h - 2h$ production range seem to be somewhat plausible and can be explained by an automated tool chain that first compiles the payload, let's say on one machine, and then passes on the payload to a packer machine. Yet we would expect this production times to be somewhat constant and we have no good explanation why the production process of some samples apparently takes many hours. Maybe a deeper analysis of the samples and the packers used in Trickbot production could shed some light on this issue.

Last but not least we would like to point out that it is uncertain whether the numbers in Figure 5 indeed reflect the production times: (i) Unlike for normal compilers, we do not

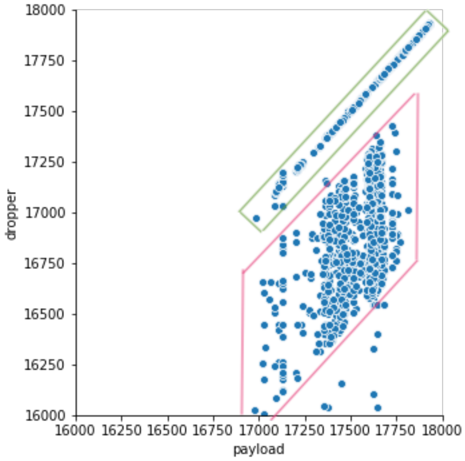


Figure 4: PE timestamps for payloads on x-axis, for droppers on y-axis (measured in days since 1970).

know how packers set the PE timestamps of the dropper files they produce. (ii) We have found that for 8% of the samples the dropper PE timestamp is $0h$ to $24h$ *older* than the payload timestamp. The existence of such samples can be hypothetically explained by clock synchronization issues between multiple machines or services used for compilation and subsequent sample packing. This however implies that we cannot necessarily trust PE timestamps, even for the samples whose timestamp is not intentionally forged (iii) As we said earlier, PE timestamps can be forged at will.

Production time	Number of samples
0h-1h	262
1h-2h	160
2h-3h	97
3h-4h	81
4h-5h	54
5h-6h	43
6h-7h	37
7h-8h	42
8h-9h	22
9h-10h	5
10h-11h	3
11h-12h	1
12h-13h	1
13h-14h	3
14h-15h	8
15h-16h	3
16h-17h	0
17h-18h	2
18h-19h	1
19h-20h	4
20h-21h	2
21h-22h	3
22h-23h	3
23h-24h	1

Figure 5: Number of Trickbot samples with production times of $0-24$ hours in 1 hour intervals.

3 Infrastructure Analysis

In this section we are going to have a deeper look at the networking infrastructure of Trickbot based on the information we collected during approximately 5 months. We do not go into details about Trickbot networking protocol as the focus lies on the temporal analysis. However a brief introduction of the way Trickbot communicates might be helpful for the further understanding, Figure 6 shows a high-level schema of how Trickbot communicates.

The most common infection vector are weaponized Office documents that trigger the download of the Trickbot binary or a dropping of Trickbot after an Emotet infection has happened. The first method is using Powershell code that is embedded in the Office document. The Powershell scripts download the binary directly from a webserver and executes it. The second is commonly seen during targeted ransomware attacks such as reported by Trend Micro [10] and us [11].

After the successful infection, Trickbot begins to communicate with the first stage C2 servers that are in the configuration delivered within the binary. These first stage C2 servers are mostly compromised systems. The communication is encrypted and uses either TCP port 443 or (often) TCP port 447 or 449. Interestingly, the certificates used for these communications are self-signed and use the default parameters of OpenSSL (”organizationName=Internet Widgits Pty Ltd”). The malware then downloads the next actual configuration file (we name it main.cfg) with a list of C2 servers to connect to. Communication however remains identical using SSL with the aforementioned ports. Depending on the module, additional C2 servers come into play that are contained in additional configuration files. In the following we focus on the configuration file of the injectDll module (or more precisely injectDll32 or injectDll64 depending on the platform), which is used for credential theft and injects within the browser.

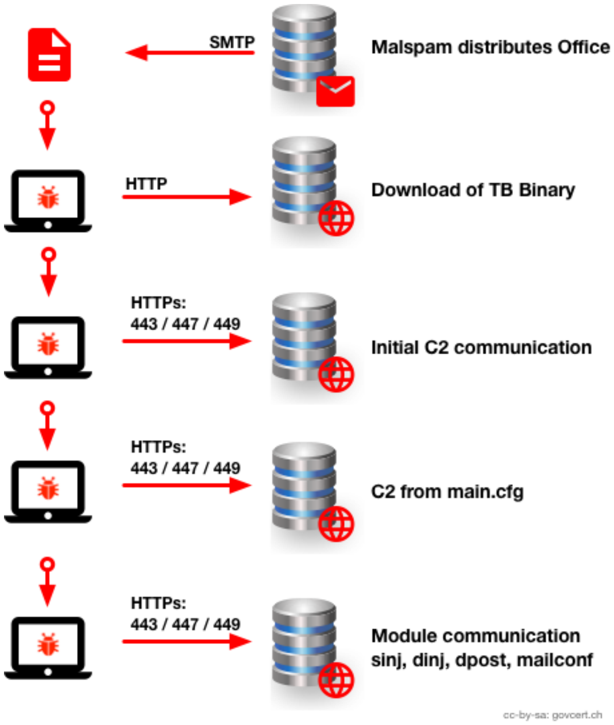


Figure 6: TrickbotNetwork

There are 3 types of configuration files shown in Table 1 that are going to be discussed later in this document.

File	Description	No of files
sinj	Static injects, contains targets, C2 servers, used by injectdll	1566
dinj	Dynamic injects, contains targets, C2 servers, used by injectdll	1559
dpost	Password Grabber, contains exfiltration IP addresses, used by injectdll	1697
mailconf	Email stealer, contains exfiltration IP addresses, used by mailsearcher	1648
main	Main configuration of Trickbot	7156

Table 1: Overview of collected configuration files

We have analyzed the configuration files and extracted IP addresses, domain names and targets for their temporal and spatial traits and are going to present them in the following sections.

3.1 Analysis of C2 Servers

Information about C2 servers is stored in the configuration files mentioned above. We have extracted the IP addresses, Autonomous System (AS), geolocation and their temporal behavior. The term temporal behaviour explains how the infrastructure elements are changing over time. In the following chapters we are going to analyze the C2 servers for basic configuration, static injects, dynamic injects, mail exfiltration and credential theft.

3.1.1 Analysis of Main Configuration

We collected a total of 316 IP addresses in the main configuration files. These show interesting patterns as there are some hosting providers that are often used. In Listing 7 an excerpt of a typical main configuration file of Trickbot is shown. In the context of the network analysis, the `<srv>` tags are important, as they consist of the IP address and the port number. We extracted and analyzed the IP addresses and are introducing the results in the subsequent sections. The `<gtag>` displays the campaign ID. After the `<servs>` section the module configuration follows. In our example, the System Reconnaissance and the Browser Inject modules are configured.

```
<mcconf>
<ver>1000292</ver>
<gtag>tt0002</gtag>
<servs>
<srv>51.68.170[.]58:443</srv>
<srv>68.3.14[.]71:443</srv>
<srv>174.105.235[.]178:449</srv>
<srv>195.54.162[.]247:443</srv>
<srv>181.113.17[.]230:449</srv>
...
</servs>
<autorun>
<module name=”systeminfo” ctl=”GetSystemInfo”/>
<module name=”injectDll”/>
</autorun>
</mcconf>
```

Figure 7: Excerpt from main configuration file

This example shows C2 servers hosted on TCP port 443 and TCP port 449, but no usage of TCP port 447 which is also known to be used by Trickbot. We extracted and analyzed the IP addresses for their AS as shown in Figure 8.

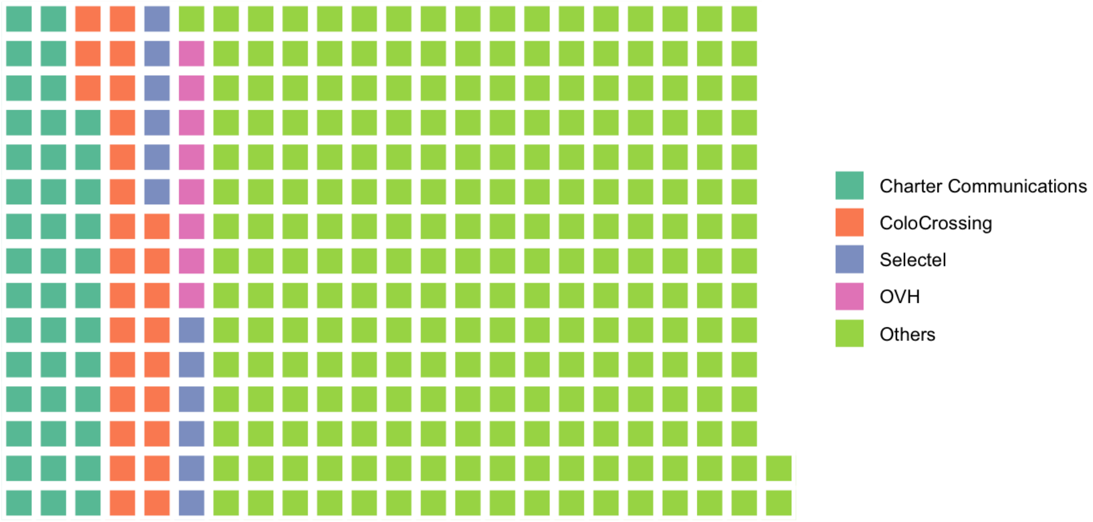


Figure 8: C2 servers in Main Configuration

To get the temporal context, we took the timestamp of the first and last appearance of a given IP address within a configuration file. Figure 9 shows that the lifetimes vary. The majority of the IPs are very short lived, while others have a lifetime of several days, or even weeks. We do not know the exact reason for this pattern, but we assume that most IP addresses are only short-lived because they are blacklisted or used for detection of an infection after a very short time thus forcing the attackers to change them quickly. Why other IPs have a longer lifetime cannot be answered, perhaps these are just testing systems that only appeared in pre-production configuration files. It would also be interesting to correlate the disappearance of IP addresses with their appearance in blacklists, but this was out of the scope of this article.

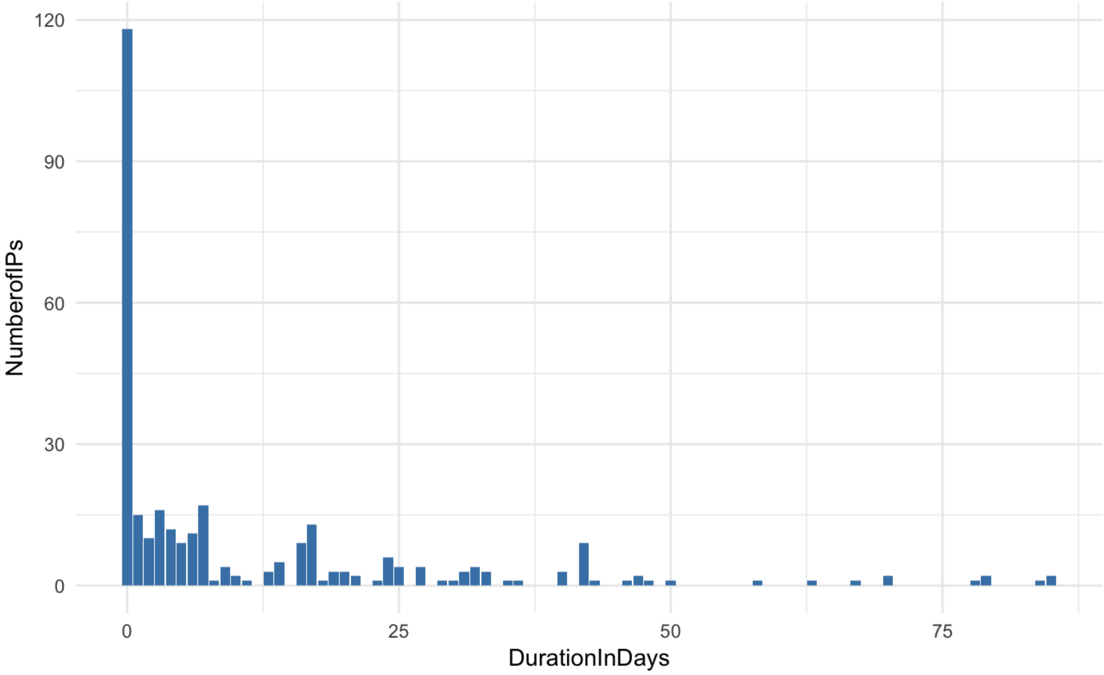


Figure 9: C2 servers in Main Configuration

A small extract from October and November shows this pattern in more detail in Figure 10.

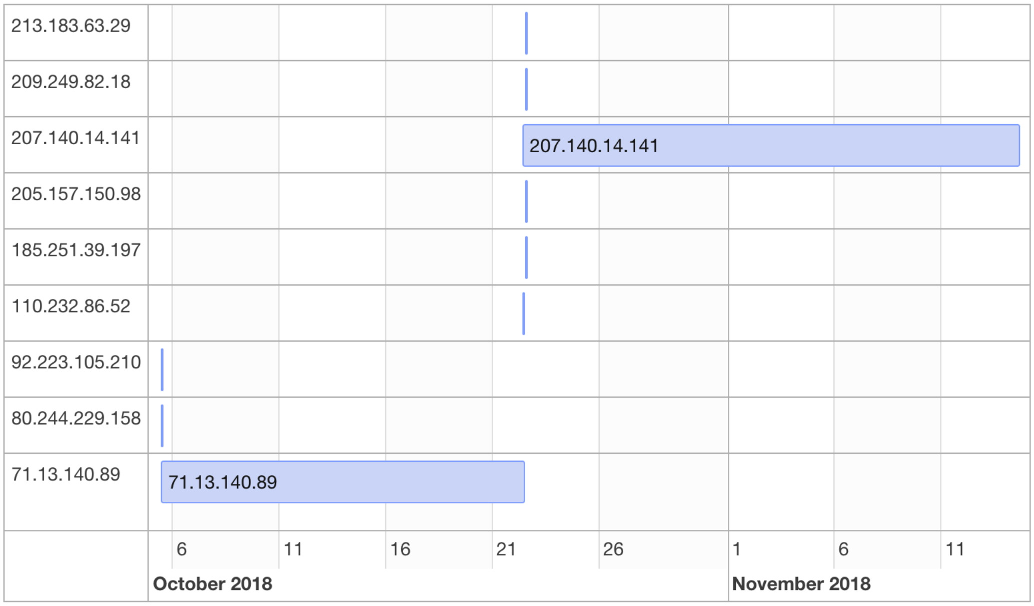


Figure 10: C2 servers in Main Configuration

3.1.2 Analysis of Static Configuration

Sinj files describe the Static Configuration of Trickbot with an example shown in Figure 11

```
<slist>
<sinj>
<mm>hXXps://www.rbsidigital[.]com*</mm>
<sm>hXXps://www.rbsidigital[.]com/default.aspx*</sm>
<nh>krsajxnbficgmrhtwsoezpklqvyd[.]net</nh>
<url404></url404>
<srv >162.248.225[.]103:443</srv>
</sinj>
```

Figure 11: Sinj Configuration Example

The parameter `<mm>` describes the target host, the `<sm>` the target URL and the `<srv>` the IP address of the server that is contacted for the injects.

We have analyzed the destination IPs of the sinj configuration files. We do not know for sure whether these are hacked systems or owned by the attackers. However, there are a few traces that may indicate the latter. If these were hacked systems, one would expect a more random distribution of registrar information which is clearly not the case as can be seen in Figure 13. Many of these IP addresses seem to have been running Nginx and are showing its default webpage. However we do not have enough evidence to either verify or falsify this.

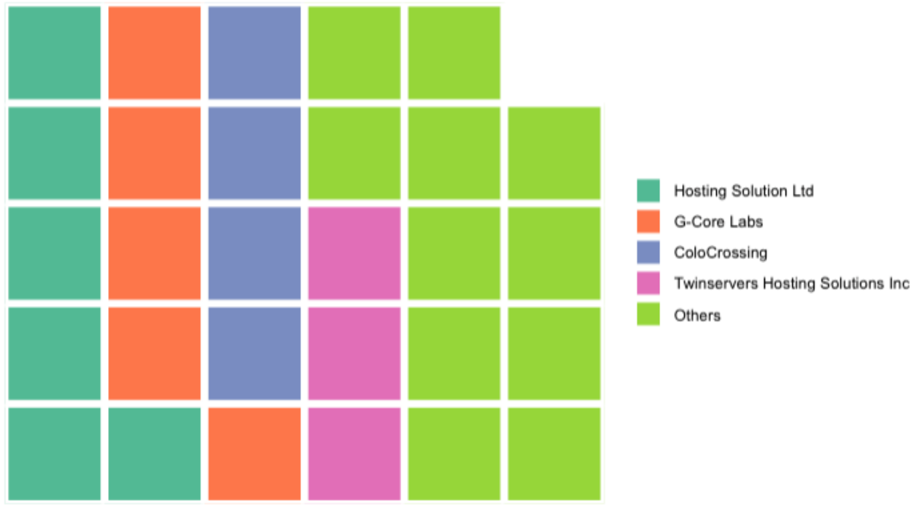


Figure 12: C2 servers in Static Configuration

It is interesting that many of the servers in our dataset are located either at Hosting Solution or G-Core Labs. Table 2 shows the IPs and their respective ASNs.

IP	Country	AS	AS Description
104.149.50[.]68	US	40676	Psychz Networks
107.174.15[.]76	US	36352	ColoCrossing
108.174.60[.]156	US	36352	ColoCrossing
131.153.19[.]122	NL	60558	Phoenix Nap, LLC.
131.153.19[.]58	NL	60558	Phoenix Nap, LLC.
154.16.195[.]34	NL	49981	WorldStream B.V.
162.247.155[.]116	US	30235	Twinservers Hosting Solutions Inc.
162.247.155[.]128	US	30235	Twinservers Hosting Solutions Inc.
162.247.155[.]155	US	30235	Twinservers Hosting Solutions Inc.
162.248.225[.]103	US	14576	Hosting Solution Ltd.
162.248.4[.]55	US	62838	Reprise Hosting
165.231.102[.]50	NL	41564	Packet Exchange Limited
185.180.197[.]117	US	14576	Hosting Solution Ltd.
185.180.197[.]35	US	14576	Hosting Solution Ltd.
185.180.197[.]36	US	14576	Hosting Solution Ltd.
185.180.198[.]147	US	14576	Hosting Solution Ltd.
185.20.184[.]74	NL	50673	Serverius Holding B.V.
192.252.210[.]19	US	46562	Total Server Solutions L.L.C.
192.99.178[.]144	CA	16276	OVH SAS
198.46.160[.]190	US	36352	ColoCrossing
198.8.91[.]37	US	46562	Total Server Solutions L.L.C.
204.155.31[.]137	US	14576	Hosting Solution Ltd.
23.94.160[.]49	US	36352	ColoCrossing
31.131.27[.]213	US	56851	PE Skurykhin Mukola Volodumurovuch
92.38.149[.]175	US	202422	G-Core Labs S.A.
92.38.149[.]45	US	202422	G-Core Labs S.A.
92.38.149[.]50	US	202422	G-Core Labs S.A.
92.38.149[.]52	US	202422	G-Core Labs S.A.
92.38.149[.]53	US	202422	G-Core Labs S.A.

Table 2: Static Configuration Country and ASN Distribution

Most of the IPs are short lived and can only be observed during one day as can be seen in Figure 13. However there are a few that last longer, but not more than 6 days which is in sharp contrast to the IPs in the main config which have some very long-living elements. Whether the 6days are merely coincidental or if these are really longer lived elements is difficult to tell. Nevertheless, the difference to the temporal pattern of the main config is noteworthy even if we cannot provide a good explanation.

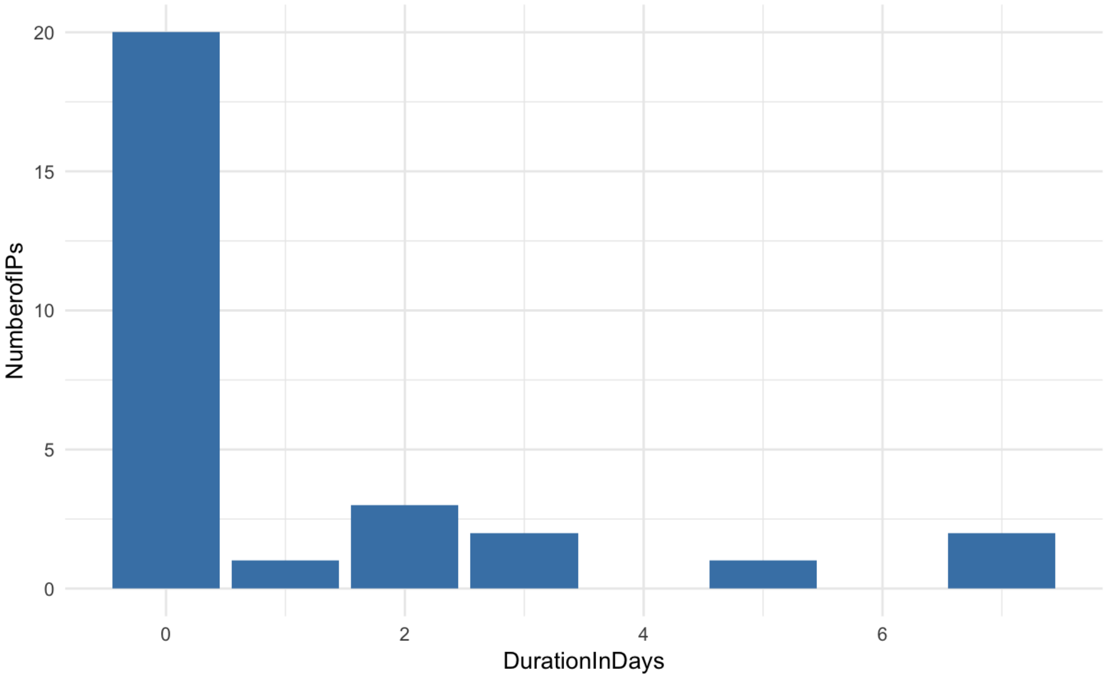


Figure 13: C2 servers in Static Configuration Lifetime

Plotted on a timeline (see Figure 14) there are three remarkable elements:

- The lifetime of IP addresses essentially does not overlap. IP addresses are rather used sequentially.
- The diagram shows a sequence of IPs used for static injects.
- Some of them were seen for several days, others were just used in one occasion, the longest period was 7 days.

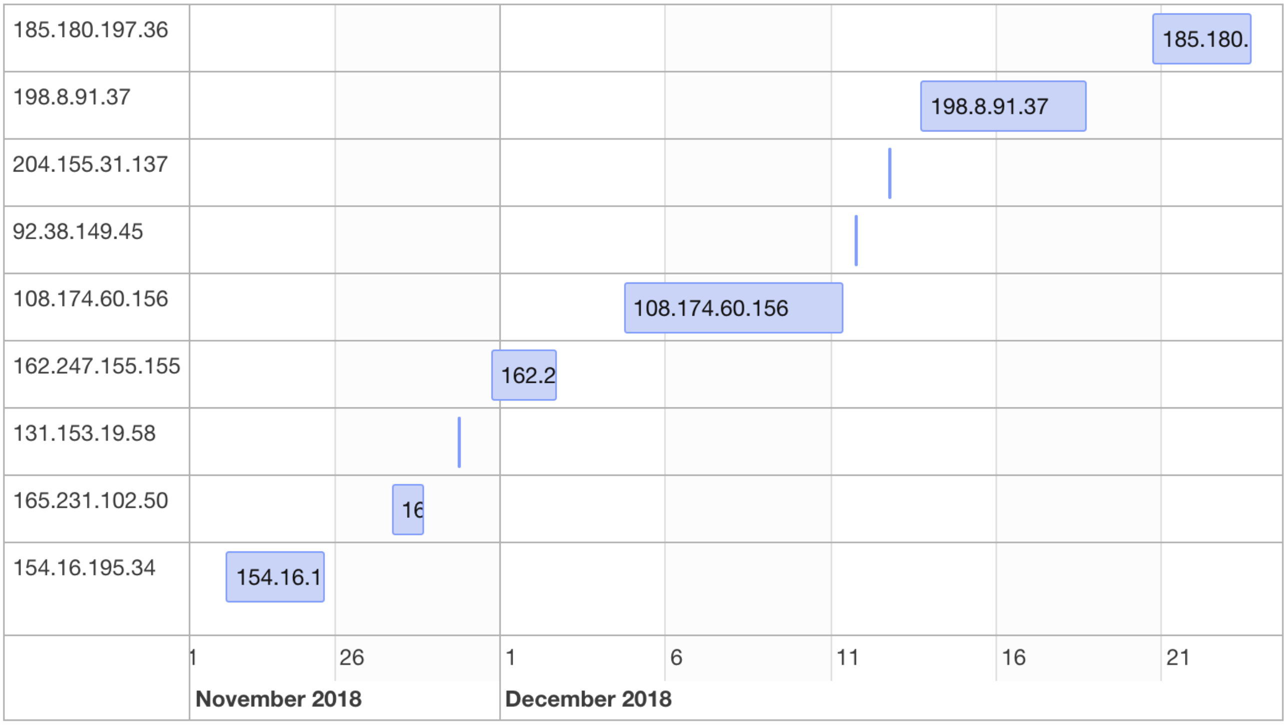


Figure 14: C2 servers in Static Configuration over Time (extract)

3.1.3 Analysis of dpost

As already mentioned, dpost configuration files contain exfiltration points for stolen credentials. The configuration files have the following format as shown in Listing 18. The format is pretty self-explanatory as it just has the handlers (C2 servers) where the stolen credentials are sent to. Interestingly this is done using plain http with the stolen data sent out in cleartext. See also the blog post by Fortinet about the pwgrab module [12].

```
<dpost>
<handler>hXXp://24.247.181[.]125:8082</handler>
<handler>hXXp://96.36.253[.]146:8082</handler>
<handler>hXXp://46.146.252[.]178:8082</handler>
...
</post>
```

Figure 15: Listing of dpost configuration file (extract)

If we plot the IP addresses over time as in shown in Figure 16, the pattern is different from the other configuration files:

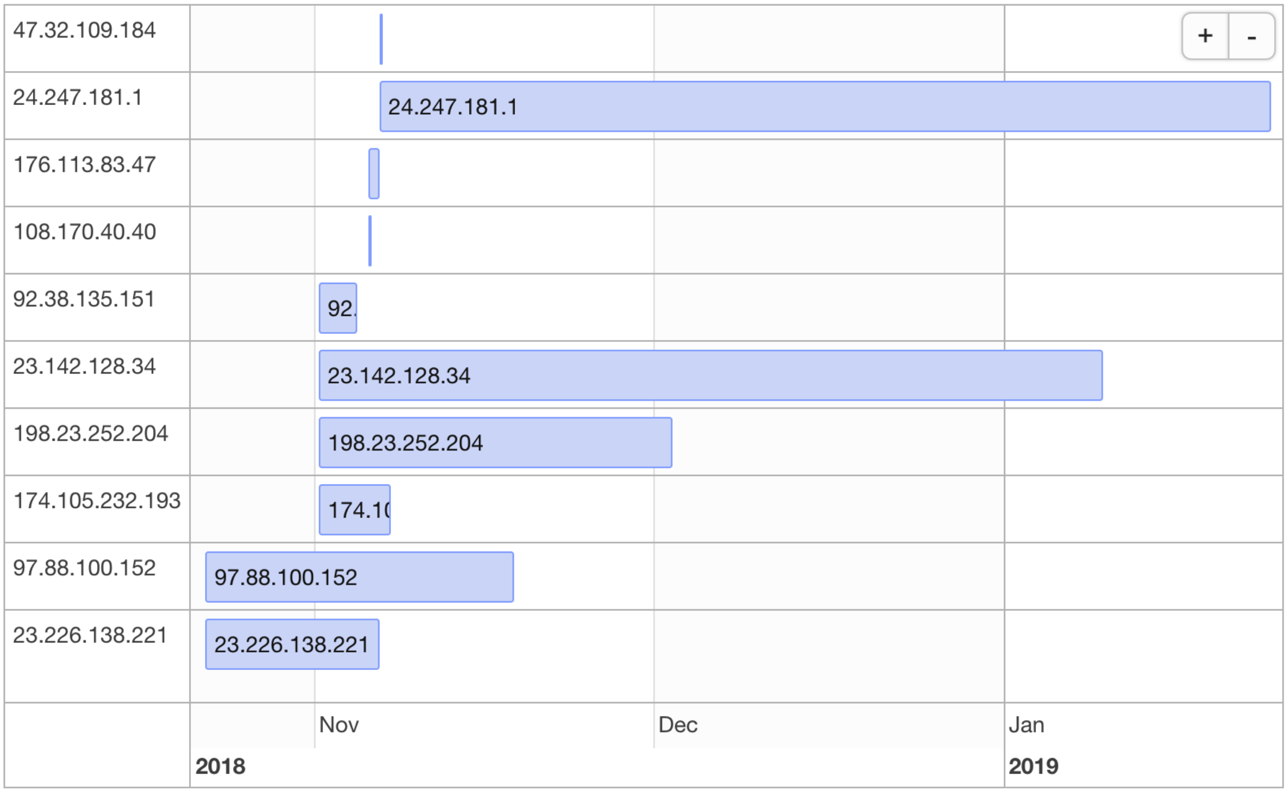


Figure 16: C2 servers in dpost config over time

We see that some IPs have slong lifetimes whereas others are only very short lived. This gets much more evident when we plot it in a histogram (see Figure 17) showing the count of IPs with a certain lifetime. Most IP addresses are short lived, meaning one day or less while some are active for a longer time, the longest being 127 days (46.146.252.178/ASN12768/ER-TELECOM-AS, RU).

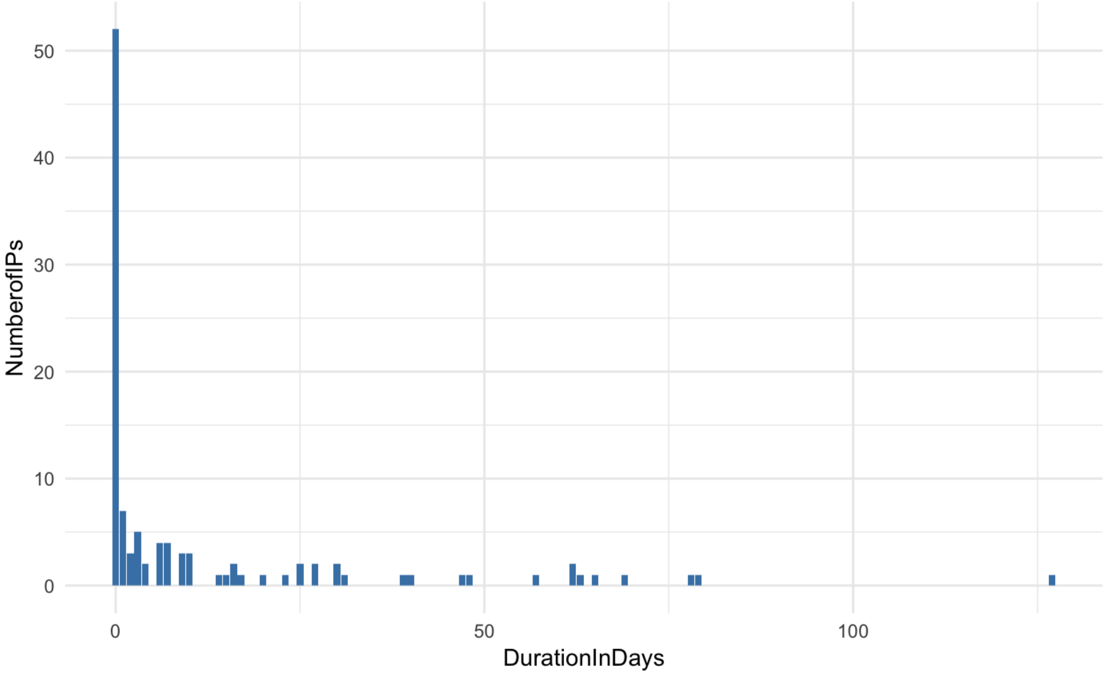


Figure 17: C2 servers in dpost config lifetime

3.1.4 Analysis of Mailconf

We harvested various mailconf files of Trickbot which configure one of the possible data exfiltration points. These are simple configuration files similar to the dpost configs. The `<handler>` denotes the C2 server where the harvested email is sent to. An example is shown in Listing 18.

```
<mail>
<handler >195.123.245[.]131:443</handler>
</mail>
```

Figure 18: Listing of dpost configuration file (extract)

Figure 19 shows the IP addresses plotted onto a timeline. One can clearly see that the IPs are seldom used at the same time but are replacing one another after lifetime of a few days to a few weeks. It seems that they have only one IP address active at a given time. As we have only monitored the actors over 4 months we do not have enough data to make a histogram showing the lifetime distribution.

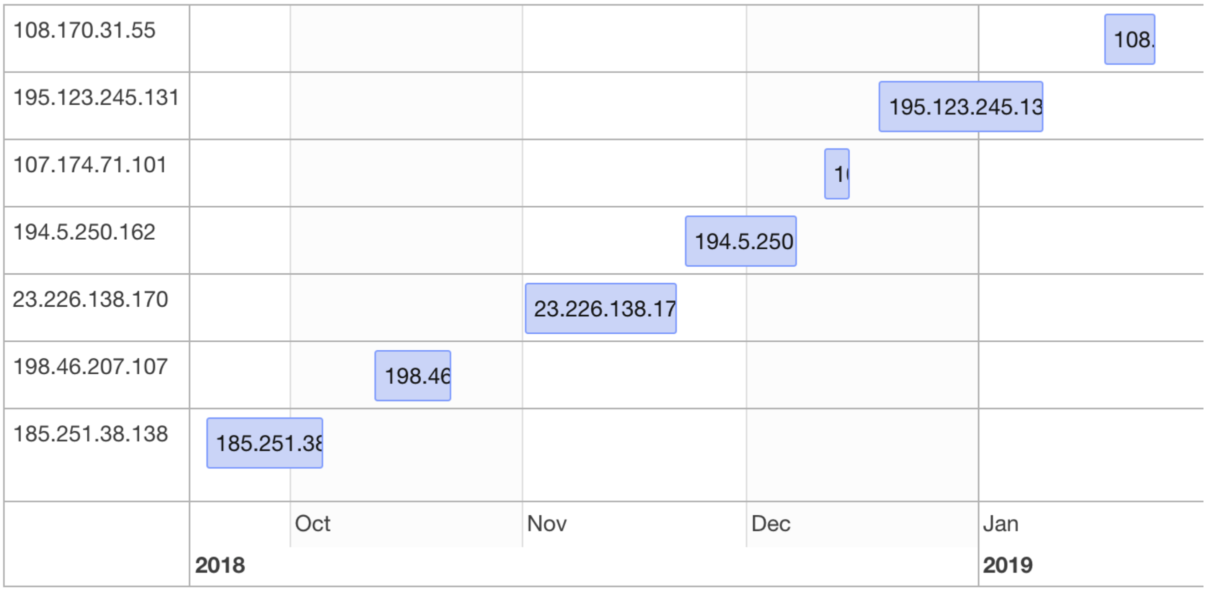


Figure 19: C2 servers in Mailconf

In Table 3 the networks and countries of these servers are listed. We can see that there is some tendency to use hosters in the US and in Eastern Europe but apart from that we have not enough data to draw any conclusions.

IP	Country	AS	AS Description	Country of AS
107.174.71[.]101	US	36352	AS-COLOCROSSING - ColoCrossing	US
108.170.31[.]55	US	20454	SSASN2 - SECURED SERVERS LLC	US
185.251.38[.]138	NL	48282	MCHOST-AS	RU
195.123.245[.]131	CZ	204957	LAYER6	UA
198.46.207[.]107	US	36352	AS-COLOCROSSING - ColoCrossing	US
23.226.138[.]170	US	8100	Quadranet	US

Table 3: Mailconf Country and ASN Distribution

3.2 Targets

For the determination of targets, we focused on the Static Configuration. In order to determine the country of the target, we looked where the web server was located assuming that most banks position their ebanking servers in the country of their most relevant customer base. We checked the result manually and made adjustments where necessary. We took the top 5 values as there is a gap between the 5th and 6th country. We have observed the following illustrated in Figure 20:

- Trickbot has a lot of targets in the US region.
- Switzerland is currently not a target (apart from big international financial institutes).
- The campaigns are spread widely and are neither targeted to a region nor done in a way that tries to adapt to the victims.

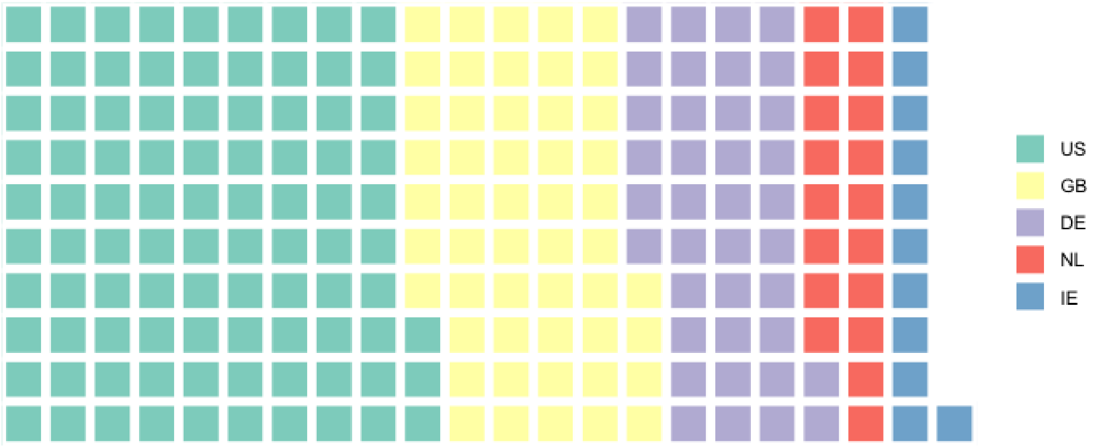


Figure 20: Targets per Country

Our results are matching with the results published by Fortinet at Botconf 2018 [8] even though there might be some minor discrepancies, probably based on differences in the method of determining the target's country.

Having a look at the temporal distribution of the target countries to time in Figure 21, we can observe a few noteworthy points:

- The number of targets remains stable over an extended time period.
- In November, we observe a steep rise in the number of targets.
- The scattered points at the beginning and the end is probably due to lack of data from our side and has no special meaning.
- Switzerland is currently not a target (apart from big international financial institutes).

By looking deeper into the data, we observed that Germany became a target on November 7th 2018: when we started the tracker in autumn 2018, we had a stable rate of 250 - 260 targets in the list. This went on until November the 7th when we noticed a large increase to 318 targets. When we compare the list of the attacked organizations we can see that nearly all of them are located in Germany. We believe that the attacker began targeting German financial institutes at this point. After that, the target list remained stable again. The decrease at the end of the measurement is most probably due to our reduced visibility because the criminals made significant changes in Trickbot. In contrast to other malware families such as Dridex, Gozi or Retefe, there seems to exist only one configuration file used for all countries.

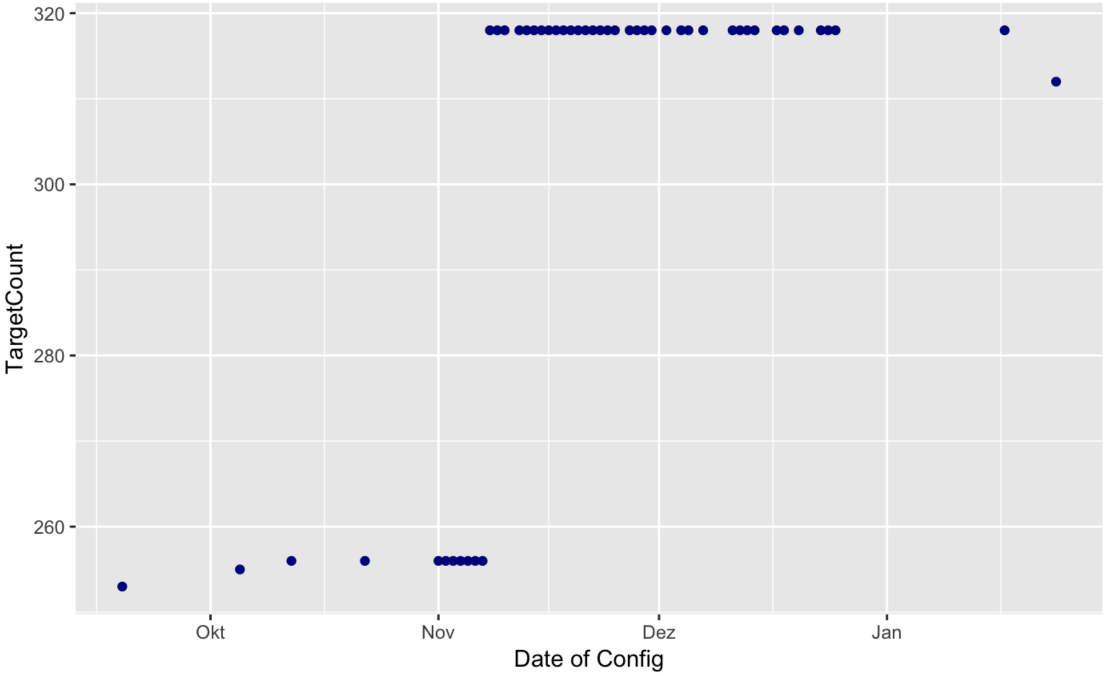


Figure 21: Number of targets over time

3.3 Conclusions Network Analysis

As we have shown, there seems to be some kind of coordination about the networking infrastructure. Even though there is a lot of uncertainty yet, we believe that the analysis proofs that the actors are actively managing their infrastructure and exchanging it on a regular base. We observe a clear sequence in the data of the static inject servers as well as in the mailconf servers. For the larger amount of C2 servers used in the main config, the sequence is less clear, there is more overlapping (as one would expect). Although we do not have that much data when it comes to dpost configuration, the pattern seems to be similar to the one seen with C2 servers from the main config. The lifetime of how long a server is being used greatly varies. However, most IP addresses are used for a very short time period but there are several IPs with a much longer malicious lifespan. We can also see that there is a preference for certain AS (Colocrossing, Charter and G-Core Labs), but as these are huge providers, it cannot be told if this is on purpose as the attackers prefer these networks or if it is merely a coincidence. One of the most important part of the work of a CERT is to determine which organisations of its constituency are at risk. This is why we try to extract configuration files that contain the target lists on a regular base. When analyzing the Trickbot target list we can see that the attackers have a strong focus on the US, Great Britain and Ireland, Germany and the Netherlands. In the analyzed configuration files we saw a sharp rise in the number of targets on November 7th when a lot of German targets were added to the target list.

References

[1] https://www.virustotal.com

[2] https://abuse.ch/

[3] Trickbot: We Missed you, Dyre. https://www.fidelissecurity.com/threatgeek/threat-intelligence/Trickbot-we-missed-you-dyre

[4] Introducing Trickbot, Dyreza's successor. https://blog.malwarebytes.com/threat-analysis/2016/10/trick-bot-dyrezas-successor/

[5] A Nasty Trick: From Credential Theft Malware to Business Disruption. https://www.fireeye.com/blog/threat-research/2019/01/a-nasty-trick-from-credential-theft-malware-to-business-disruption.html

[6] The Business of Organized Cybercrime: Rising Intergang Collaboration in 2018. https://securityintelligence.com/the-business-of-organized-cybercrime-rising-intergang-collaboration-in-2018/

[7] Spring Dragon – Updated Activity. https://securelist.com/spring-dragon-updated-activity/79067/

[8] Fortinet - Trickbot: The Trick is on you. https://www.botconf.eu/wp-content/uploads/2018/12/2018-F-Bacurio-Junior-J-Salvio-Trickbot-The-Trick-is-On-You-presented.pdf

[9] Cyberreaon - A one-two punch of Emotet, Trickbot, & Ryuk Stealing & ransoming data. https://www.cybereason.com/blog/one-two-punch-emotet-Trickbot-and-ryuk-steal-then-ransom-data

[10] Trendmicro - Emotet-Distributed Ransomware Loader for Nozelesn Found via Managed Detection and Response. https://blog.trendmicro.com/trendlabs-security-intelligence/emotet-distributed-ransomware-loader-for-nozelesn-found-via-managed-detection-and-response/

[11] GovCERT.ch - Severe Ransomware Attacks Against Swiss SMEs. https://www.govcert.admin.ch/blog/36/severe-ransomware-attacks-against-swiss-smes

[12] Fortinet - Deep Analysis of Trickbot New Module pwgrab. https://www.fortinet.com/blog/threat-research/deep-analysis-of-Trickbot-new-module-pwgrab.html

JOURNAL OF CYBERSECURITY

Journal of Cybersecurity, 2019, 1–13
doi: 10.1093/cybsec/tyz006
Research paper

Research paper

To share or not to share: a behavioral perspective on human participation in security information sharing

Alain Mermoud [1,2,*], Marcus Matthias Keupp [2,3], Kévin Huguenin [1], Maximilian Palmié [4] and Dimitri Percia David [1,2]

[1]Department of Information Systems, Faculty of Business and Economics (HEC Lausanne), University of Lausanne (UNIL), 1015 Lausanne, Switzerland; [2]Department of Defence Management, Military Academy (MILAC) at ETH Zurich, 8903 Birmensdorf, Switzerland; [3]University of St. Gallen, Dufourstrasse 50, 9000 St. Gallen, Switzerland; [4]Institute of Technology Management, University of St. Gallen, Dufourstrasse 40a, 9000 St. Gallen, Switzerland

*Corresponding address: Tel: +41 58 484 82 99; E-mail: alain.mermoud@gmail.com

Received 21 July 2018; revised 17 May 2019; accepted 14 June 2019

Abstract

Security information sharing (SIS) is an activity whereby individuals exchange information that is relevant to analyze or prevent cybersecurity incidents. However, despite technological advances and increased regulatory pressure, individuals still seem reluctant to share security information. Few contributions have addressed this conundrum to date. Adopting an interdisciplinary approach, our study proposes a behavioral framework that theorizes how and why human behavior and SIS may be associated. We use psychometric methods to test these associations, analyzing a unique sample of human Information Sharing and Analysis Center members who share real security information. We also provide a dual empirical operationalization of SIS by introducing the measures of SIS frequency and intensity. We find significant associations between human behavior and SIS. Thus, the study contributes to clarifying why SIS, while beneficial, is underutilized by pointing to the pivotal role of human behavior for economic outcomes. It therefore extends the growing field of the economics of information security. By the same token, it informs managers and regulators about the significance of human behavior as they propagate goal alignment and shape institutions. Finally, the study defines a broad agenda for future research on SIS.

Key words: security information sharing; psychometrics; economics of information security; behavioral economics, behavioral psychology

Introduction

Security information sharing (SIS) is an activity whereby individuals exchange information that is relevant to analyze or prevent cybersecurity incidents. Such information includes, but is not limited to, the identification of information system vulnerabilities, phishing attempts, malware, and data breaches, as well as results of intelligence analysis, best practices, early warnings, expert advice, and general insights [67].

Prior research has proposed that SIS makes every unit of security investment more effective, such that individuals can reduce investments dedicated to generate cybersecurity in their organization. As a result of these individual improvements, total welfare is also likely to increase [41, 47]. Hence, SIS likely contributes to strengthening the cybersecurity of firms, critical infrastructures, government, and society [19, 45, 46, 48, 54].

However, these theoretical expectations hardly seem to materialize. Recent contributions have noted that SIS is at suboptimal levels, implying negative consequences for the cybersecurity of organizations and society [19]. Game-theoretic simulation suggests that individuals may free-ride on the information provided by others while not sharing any information themselves [47, 55]. Researchers and international organizations have been warning for years that individuals seem reluctant to share security information, although the technical infrastructure for information exchange does exist [32, 33, 47, 73]. Legislators have attempted to resolve this problem by creating regulation that makes SIS mandatory.[1] However, reviews suggest that despite these attempts, individuals still seem reluctant to share security information [16, 44, 72, 103]. They may even 'game' the system in an attempt to circumvent regulation [5, 71, 72].

All these findings imply that human behavior may be significantly associated with the extent to which SIS occurs (if at all). It is therefore not surprising to see recent work emphasizing that the study of human behavior is key to the understanding of SIS [19]. More specifically, this work predicts that SIS can only be imperfectly understood unless the human motivation to (not) participate in SIS is studied [53, 65, 98].

However, few contributions have addressed this research gap to date. Since an excellent account of the SIS literature exists [64], we refrain from replicating this account here. We rather point to the fact that this account shows that very few empirical studies on non-public SIS exist. These few studies concentrate on analyzing incident counts and aggregate data, but they do not study human behavior at the individual level of analysis (see Ref. [64] for a tabulated overview).

Our study intends to address this gap by proposing how and why human behavior and SIS may be associated, and by providing an empirical test of this association. Following prior recommendations [6], we adopt an interdisciplinary approach. Recently, interdisciplinary studies were productive in showing the extent to which human behavior is associated with knowledge sharing [87, 106].

We build a theoretical framework anchored in behavioral theory, arguing that SIS is associated with human behavior. We use psychometric methods to test these associations, analyzing a unique sample of 262 members of an Information Sharing and Analysis Center (ISAC) who share real security information. The remainder of this article is structured as follows. Section 2 develops the behavioral framework and deducts testable hypotheses from this framework. Section 3 details the sampling context, measures, and empirical methods. The results are explained in Section 4. Section 5 discusses both the theoretical, empirical, and practical contributions our study makes and points to some limitations of our approach that open up paths for future research.

Theoretical Framework and Hypotheses

Behavioral research relativized some of the strong formal assumptions that neoclassical economics had ascribed to human behavior, particularly those of rationality, perfect information, and selfish utility maximization ("homo oeconomicus"). In contrast, it showed that human beings have bounded instead of perfect rationality. They often violate social expectations, have limited information-processing capacity, use heuristics when making decisions, are affected by emotion while doing so, and retaliate even if the cost of retaliation exceeds its benefits [13, 27, 37, 58, 59, 89].

Moreover, humans do not necessarily maximize higher level (i.e. organizational, societal) goals, even if it would be economically rational for them to do so. Theoretical work on SIS has suggested early that individual and organizational interests may not always be aligned and that the individual is not necessarily an indifferent agent [42]. Goal-framing theory suggests that individual goals may not necessarily be congruent with higher level goal frames, implying that the individual can defect from organizational maximization goals [66]. Particularly in the case of collective action, the individual may behave in ways that are not conducive to the overall group goal [78, 79]. For the context of SIS, this research implies that individually, humans might not necessarily participate in SIS although it would be optimal to do so for society as a whole.

Particularly, human exchange relationships are not necessarily characterized by rational economic optimization, but instead by human expectations about fairness, reciprocity, and trust [36, 37, 39, 68]. Therefore, the argument can be made that SIS may be associated with human behavior. Indeed, prior research argues that the understanding of SIS requires an analysis of what behavior may motivate humans to participate in SIS and what may deter them from doing so [8, 10].

Human behavior is the result of human motivation, intention, and volition. It manifests itself in goal-directed (i.e. nonrandom) and observable actions [90, 93, 102]. Sharing information implies human action from at least the side of the individual who shares. Moreover, SIS constitutes an economic transaction by which knowledge resources are shared, rather than acquired [17]. Hence, SIS differs from discrete arm's length transactions, whereby a single individual simply trades financial means for access to information. Instead, SIS is characterized by continued social interaction among many individuals who mutually exchange information assets [106].

Therefore, humans are unlikely to randomly participate in SIS, such that SIS does not occur "naturally." Hence, theorizing is required regarding how and why human behavior may be associated with SIS. Applying prior behavioral research to our research context, we develop testable hypotheses about five salient constructs which may be associated with SIS. In all of these hypotheses, our focal individual is an indifferent individual who, independently of the motives of other individuals, ponders whether or not to participate in SIS. We believe this perspective is conservative and conducive to empirical analysis since it neither requires assumptions about the behavior of other individuals nor a dyadic research setting.

Attitude

Behavioral theory suggests that attitudes have a directive influence on human behavior [1]. Attitude is a psychological tendency that is expressed by evaluating a particular entity with some degree of favor or disfavor [30]. Hence, an individual's favorable or unfavorable attitude towards a particular behavior predicts the extent to which this behavior actually occurs [2, 3].

Much empirical work has confirmed and detailed this attitude–behavior link, particularly in the context of information systems adoption and intention to use (see Refs [14] and [62] for extensive literature reviews). More specifically, this attitude-behavior link influences individuals' intention to share knowledge [17]. Moreover, an affirmative attitude towards knowledge sharing positively influences participation rates [87]. Descriptive work has conjectured (though not tested or confirmed) that individual attitudes about the meaningfulness of SIS might be associated with actual participation in SIS [32]. Therefore, if the focal individual has a positive attitude towards SIS, s/he should be more likely to participate in SIS. Therefore,

> H1: SIS is positively associated with the extent to which the focal individual has a positive attitude towards SIS.

[1] For example, the USA created the 2002 Sarbanes-Oxley Act and the 2015 Cybersecurity Information Sharing Act (CISA). The Health Insurance Portability and Accountability Act (HIPAA) requires organizations to report breaches of protected health information (PHI) to the U.S. Department of Health and Human Services (HHS). In December 2015, the European Parliament and Council agreed on the first EU-wide legislation on cybersecurity by proposing the EU Network and Information Security (NIS) Directive.

Reciprocity

Behavioral theory suggests that human behavior is characterized by inequity aversion [39]. As they socially interact with others, humans expect to receive equitable compensation whenever they voluntarily give something to others, and they punish those unwilling to give something in return [22, 95]. Hence, when humans are treated in a particular way, they reciprocate, that is, they respond likewise [36]. As a result, reciprocity is a shared behavioral norm among human beings that governs their social cooperation [38, 50].

Economic exchange relationships are therefore shaped by the reciprocity expectations of the participants involved in this exchange [61]. In such relationships, reciprocity is a dominant strategy that is conducive to a socially efficient distribution of resources [7, 20]. Therefore, the extent to which the focal individual participates in information exchange is likely associated with that individual's expectation that his/her efforts are reciprocated.

For example, reciprocal fairness is an important variable in the design of peer selection algorithms in peer-to-peer networks. By integrating reciprocal response patterns such as "tit-for-tat," operators can optimize peer-to-peer traffic [101]. The value of a unit of security information is proportional to the incremental security enhancement that this unit is supposed to provide to the recipient [18, 49]. Hence, whenever the focal individual shares such information units, it creates value for the counterparty. By the above arguments, the focal individual likely refuses to participate in future exchanges unless such value creation is reciprocated by the counterparty.

On the one hand, the focal individual may expect that information sharing is reciprocated by "hard rewards," that is, in monetary terms, by a higher status inside the ISAC or his or her own organization, or in terms of career prospects (transactional reciprocity). On the other hand, the focal individual may also expect that whenever s/he shares a unit of information, s/he receives useful information in return, such that a continuous social interaction that is beneficial to both parties emerges (social reciprocity). Prior research suggests that both these types of reciprocity are associated with information exchange patterns between individuals [63, 80, 88]. Therefore,

> H2a: SIS is positively associated with the extent to which the focal individual expects his or her information sharing to be transactionally reciprocated.
>
> H2b: SIS is positively associated with the extent to which the focal individual expects his or her information sharing to be socially reciprocated.

Executional Cost

Behavioral theory suggests that humans are loss-averse, that is, they attempt to avoid economic losses more than they attempt to realize economic benefits. Much experimental research has confirmed this tendency [58, 59, 92, 96, 97].

An economic exchange relationship can be fraught with significant transaction cost, i.e. the time, material, and financial resources that the focal individual must commit before an exchange is made [104]. Hence, if SIS is associated with high transaction costs for participation, the focal individual is likely to avoid the necessary resource commitments to finance this cost. For example, Ref. [106] argue that when knowledge contribution requires significant time, sharing tends to be inhibited. Consistent with their conceptualization, we term such transaction costs "executional cost."

As a result, in the presence of high executional cost, the focal individual likely adapts his or her behavior in an attempt to avoid these costs. For instance, if the focal individual learns that in a given ISAC environment, SIS is taking too much time, is too laborious, or requires too much effort, the individual likely reduces or terminates participation in SIS [67]. For example, an abundance of procedural rules that govern the processing and labelling of shared information and the secure storage and access to shared data likely stalls information sharing activity [33]. Thus, high executional cost likely dissuades the focal individual from participating in SIS. Therefore,

> H3: SIS is negatively associated with the extent to which the focal individual expects information sharing to be fraught with executional cost.

Reputation

Behavioral theory suggests that humans deeply care about being recognized and accepted by others [11, 15]. Many philosophers have argued that the desire for social esteem fundamentally influences human behavior and, as a result, economic action [21].

Depending on the outcomes of particular social interactions with other individuals, the focal individual earns or loses social esteem. Hence, over time each individual builds a reputation, that is, a socially transmitted assessment by which other individuals judge the focal individual's social esteem [31, 69]. For example, academic researchers strive to increase the reputation of their department by publishing scholarly work [60]. The desire to earn a reputation as a competent developer is a strong motivator for individuals to participate in open source software development although they receive no monetary compensation for the working hours they dedicate to this development [99].

When this reasoning is transferred to the context of SIS, the focal individual may be inclined to share information because s/he hopes to build or improve his or her reputation among the other participants of SIS. Prior research suggests that this desire constitutes an extrinsic motivation that may be associated with an individual's intention to share information [25, 81], and intention is a precursor of behavior. Therefore,

> H4: SIS is positively associated with the extent to which the focal individual expects information sharing to promote his or her reputation in the sharing community.

Trust

Behavioral theory suggests that humans simplify complex decision-making by applying heuristics [82, 97], particularly when they attempt to reduce the cost of information acquisition and valuation [40].

Whenever a focal individual is unable or unwilling to objectively evaluate information conveyed by other individuals, s/he likely resorts to heuristics to simplify the evaluation process [24]. In the context of SIS, this implies that whenever the focal individual receives security information from another individual, s/he cannot necessarily be sure about the extent to which (if any) this information is valuable or useful. This assessment is associated with significant transaction cost, for example, for due diligence procedures that attempt to value the information received. The individual may also lack technological competence and expertise, such that time-consuming discussions with experts are required for proper valuation. All in all, upon the receipt of a particular unit of information, the focal individual is faced with a complex valuation problem which s/he may seek to simplify by applying heuristics.

Trust is an implicit set of beliefs that the other party will behave in a reliable manner [43]. This set of beliefs is a particularly effective heuristic because it can reduce the transaction cost associated with this valuation. If the focal individual trusts the information received is useful and valuable, s/he can simplify evaluation procedures, and particularly so if the involved individuals interact in dense networks with agreed standards of behavior. Therefore, trust is a facilitator of economic organization and interaction [51, 70]. For example, mutual trust among the participants of peer-to-peer networks can reduce transactional uncertainty [105]. Moreover, trust can mitigate information asymmetry by reducing transaction-specific risks [9]. It is also a significant predictor of participation in virtual knowledge sharing communities [86].

Such trust, in turn, is positively associated with knowledge sharing in both direct and indirect ways [56], whereas distrust is an obstacle to knowledge sharing [4]. More specifically, trust is a facilitator in information security knowledge sharing behavior [87]. Thus, the extent to which the focal individual trusts the information s/he receives is valuable should be positively associated with his or her propensity to participate in SIS. Therefore,

> H5: SIS is positively associated with the extent to which the focal individual trusts that the counterparty provides valuable information.

Interaction Effects

By consequence, we suggest that trust negatively moderates the associations between attitude and reciprocity on the one hand and SIS on the other hand. We argued that trust is a facilitator of economic exchange. In other words, trust likely reduces the focal individual's perceived cost of engaging in SIS, in that s/he requires fewer or lesser alternative stimuli [66]. A neutral focal individual who has not participated in SIS before is unlikely to participate unless s/he has a positive attitude towards SIS. That individual must hence construct the meaningfulness of SIS "internally," that is, convince him- or herself that SIS is useful. By contrast, if the focal individual trusts that the information s/he receives will be useful, s/he uses the counterparty to "externally" confirm such meaningfulness of SIS. The process of the internal construction of the meaningfulness of SIS is therefore at least partially substituted by the external, trust-based affirmation of such meaningfulness. We would hence expect that the significance of the association between attitude and SIS decreases with the extent to which the focal individual trusts the information s/he receives will be useful.

By the same token, since trust is a facilitator of economic exchange, it likely reduces the association between reciprocity and SIS. An indifferent focal individual cannot be completely sure about the behavior of the exchange counterparty, such that s/he requires continuous transactional or social reciprocity for SIS to perpetuate the exchange. In the absence of any trust that the information received is useful, SIS likely ends as soon as this reciprocity requirement is no longer met. In contrast, whenever the focal individual trusts that the information s/he receives will be useful, s/he has a motive to participate in SIS that is independent of such reciprocity concerns. Hence, trust is likely to act at least partially as a substitute for reciprocity, such that the focal individual should emphasize to a lesser extent that reciprocity will be required if s/he is expected to begin or perpetuate SIS. Therefore,

> H6a–c: The extent to which the focal individual trusts that information received from the counterparty is effective negatively moderates the respective positive associations between attitude, transactional, and social reciprocity on the one hand and SIS on the other hand.

Methods

Sampling Context and Population

Our study focused on the 424 members of the closed user group of the Swiss national ISAC, the "Reporting and Analysis Centre for Information Assurance" (MELANI-net). An ISAC is an organization that brings together cybersecurity managers in person to facilitate SIS between operators of critical infrastructures. For a general introduction to the concept of an ISAC, see Ref. [107]. For some illustrative examples of ISACs across different countries, see Ref. [34]. For a detailed description of MELANI-net, its organization, and history, see Ref. [29]. The ISAC we study is organized as a public-private partnership between the government and private industry; it operates on a not-for-profit basis. Membership in MELANI-net is voluntary. In Switzerland, there is no regulation that makes SIS mandatory; hence, individuals are free to share or not share information, and they can also control the group of individuals with whom they want to share the information. This implies our study design can capture the full range of human behavior from perfect cooperation to total refusal.

The members of the closed user group are all senior managers in charge of providing cybersecurity for their respective organizations. They come from both private critical infrastructure operators and from the public sector. They have to undergo government identification and clearance procedures as well as background checks before being admitted for ISAC membership. They share classified, highly sensitive information the leaking or abuse of which may cause significant economic damage. There is no interaction of these members with the public whatsoever, and no external communication to the public or any publication of SIS results is made. For all of these members, the exchange of SIS can be assumed to be relevant, as they manage critical infrastructures that are ultimately all connected and operate with similar IT systems, such that cybersecurity problems that relate to any particular individual are likely of interest to other participants too.

Within this closed user group, individuals can contact each other by an internal message board whenever a particular individual has shared information about a threat that is of interest to other members. They do so by commenting on the initial information shared in order to establish a first contact, which then leads to further social exchange between the two individuals. Once contact is made by a short reply to the threat information, the individuals involved in the conversation meet on their own initiative to share detailed security information between them (e.g. informally over lunch, in group meetings, or small industry-specific conferences, but always face-to-face). Each individual decides for him- or herself if s/he wants to meet, with whom, and in what form. They also freely decide about the extent of the information shared (if any). MELANI-net officials neither force nor encourage individuals to interact; both in terms of social interaction in general and regarding the sharing of any particular unit of information.

Measures

Our study analyzes human behavior on the individual level of analysis. We therefore chose a psychometric approach to operationalize our constructs [77]. We adopted psychometric scales from the extant literature wherever possible and kept specific adaptions to our population context to a minimum. Table 1 explains and details all variables, their item composition and wording (if applicable), dropped items (if any), factor loadings, and Cronbach alphas and cites the sources they were taken from.

SIS is operationalized dually by the two constructs "frequency" and "intensity." Intensity measures the extent to which the focal individual reacts to any threat information shared by another individual and thus begins social interaction with that other individual. Intensity is thus a reactive measure of how intensely the focal individual engages in knowledge sharing with others upon being informed of a threat.[2] Since information sharing is not mandatory, this measure captures the individual's free choice to (not) engage in exchange relationships with other individuals. In contrast, frequency is a proactive measure; it captures how often an individual shares security information that s/he possesses him- or herself.

To capture respondent heterogeneity, we controlled for gender, age, and education level. Further, we controlled for the individual's ISAC membership duration in years, because a respondent's sharing activity may co-evolve with the length of ISAC membership. "Gender" was coded dichotomously (male, female). "Age" was captured by four mutually exclusive categories (21–30, 31–40, 41–50, 50+ years). "Education" was captured by six mutually exclusive categories (none, bachelor, diploma, master, PhD, other). We also controlled for the industry affiliation of the organization that the individual represents and combined these into five categories (government, banking and finance, energy, health, telecom and IT, all others).

Implementation

Data for all variables were collected from individual respondents by a questionnaire instrument. We followed the procedures and recommendations of Ref. [28] for questionnaire design, pre-test, and implementation. Likert-scaled items were anchored at "strongly disagree" (1) and "strongly agree" (5) with "neutral" as the midpoint. Categories for the measure "intensity" were ordered hierarchically.

The questionnaire was developed as a paper instrument first. It was pre-tested with seven different focus groups from academia and the cybersecurity industry.[3] Feedback obtained was used to improve the visual presentation of the questionnaire and to add additional explanations. This feedback also indicated that respondents could make valid and reliable assessments.

Within the closed user group, both MELANI-net officials and members communicate with each other in English. Switzerland has four official languages, none of which is English, and all constructs we used for measurement were originally published in English. We therefore chose to implement the questionnaire in English to rule out any back-translation problems. Before implementation, we conducted pre-tests to make sure respondents had the necessary language skills. The cover page of the survey informed respondents about the research project and our goals and also made clear that we had no financial or business-related interest.

The paper instrument was then implemented as a web-based survey using "SelectSurvey" software provided by the Swiss Federal Institute of Technology Zurich. For reasons of data security, the survey was hosted on the proprietary servers of this university. The management of MELANI-net invited all closed user group members to respond to the survey by sending an anonymized access link, such that the anonymity of respondents was guaranteed at all times. Respondents could freely choose whether or not to reply. As a reward for participation, respondents were offered a research report free of charge that summarized the responses. Respondents could freely choose to save intermediate questionnaire completions and return to the survey and complete it at a later point in time.

The online questionnaire and the reminders were sent to the population by the Deputy Head of MELANI-net together with a letter of endorsement. The survey link was sent in an e-mail describing the authors, the data, contact details for IT support, the offer of a free report, and the scope of our study. Data collection began on 12 October 2017 and ended on 1 December 2017. Two reminders were sent on 26 October and 9 November 2017. Of all 424 members, 262 had responded when the survey was closed for a total response rate of 62%.

Analysis

Upon completion of the survey, sample data were exported from the survey server, manually inspected for consistency and then converted into a STATA dataset (Vol. 15) on which all further statistical analysis was performed. Post-hoc tests suggested no significant influence of response time on any measure. There was no significant overrepresentation of individuals affiliated with any particular organization, suggesting no need for a nested analytical design.

We performed principal component factor analysis with oblique rotation on all items. Validity was tested by calculating item-test, item-rest, and average inter-item correlations. Reliability was measured by Cronbach alpha. High direct factor-loadings and low cross-loadings indicate a high degree of convergent validity [52]. The final matrix suggested seven factors with an eigenvalue above unity. The first factor explained 14.56% of the total variance, suggesting the absence of significant common method variance in the sample [84]. The detailed factor-loadings and their diagnostic measures are given in Table 2. Upon this analysis, three items were dropped (viz. Table 1) because they had low direct and high cross factor loadings. Finally, for any scale, individual item scores were added, and this sum was divided by the number of items in the scale [85, 94].

The construct intensity is ordered and categorical, therefore we estimated ordered probit models. A comparison with an alternative ordered logit estimation confirmed the original estimations and indicated the ordered probit model fit the data slightly better. The construct frequency is conditioned on values between 1 and 5, therefore we estimated Tobit models. Both models were estimated with robust standard errors to neutralize any potential heteroscedasticity. Consistent with the recommendation of Ref. [26], we incrementally built all models by entering only the controls in a baseline model first, then added the main effects, and finally entered the interaction effects. In both estimations, we mean centered the measures before entering them into the analysis. Model fit was assessed by repeated comparisons of Akaike and Bayesian information criteria between different specifications. Since all the categorical controls age, education and industry are exhaustive and hence perfectly collinear, Stata automatically chose a benchmark category for each of these (cf. footnotes b to Tables 5 and 6).

[2] The measure *intensity* is ordered and categorical in that it asks respondents to provide an estimate rather than an exact percentage figure. We preferred this approach in order to give respondents an opportunity to provide an estimate, such that they would not be deterred by the need to provide an exact figure. We also captured an alternative measure of intensity by a Likert scale, but found that models with the ordered categorical measure fit the data better. We also contrasted the Tobit model that used the scale-based measure for *frequency* with an alternative ordered probit model that used a categorical specification of that variable, but found that the former model fit the data much better.

[3] Further detailed information about these pre-tests is available from the corresponding author.

Table 1: Constructs, items, and scales used in the survey

Measures (source)	Type	Item	Text	Factor loading	Cronbach alpha
SIS constructs					
Intensity of SIS (novel)	Ordered categorical measure	n/a	How often do you comment on shared information? • Never • Rarely, in less than 10% of the chances when I could have • Occasionally, in about 30% of the chances when I could have • Sometimes, in about 50% of the chances when I could have • Frequently, in about 70% of the chances when I could have • Usually, in about 90% of the chances I could have • Every time	n/a	n/a
Frequency of SIS [87]	Likert scale	ISKS1	I frequently share my experience about information security with MELANI	0.8075	0.8945
		ISKS2	I frequently share my information security knowledge with MELANI	0.8903	
		ISKS3	I frequently share my information security documents with MELANI	0.8850	
		ISKS4	I frequently share my expertise from my information security training with MELANI	0.8600	
		ISKS5	I frequently talk with others about information security incidents and their solutions in MELANI workshops	0.6898	
Behavioral constructs					
Attitude [87]	Likert scale	AT1	I think SIS behavior is a valuable asset in the organization	Dropped	0.6761
		AT2	I believe SIS is a useful behavioral tool to safeguard the organization's information assets	0.7751	
		AT3	My SIS has a positive effect on mitigating the risk of information security breaches	0.6376	
		AT4	SIS is a wise behavior that decreases the risk of information security incidents	0.7849	
Transactional reciprocity [100]	Likert scale	HR1	I expect to be rewarded with a higher salary in return for sharing knowledge with other participants	0.8822	0.7956
		HR2	I expect to receive monetary rewards (i.e. additional bonus) in return for sharing knowledge with other participants	0.8743	
		HR3	I expect to receive opportunities to learn from others in return for sharing knowledge with other participants	Dropped	
		HR4	I expect to be rewarded with an increased job security in return for sharing knowledge with other participants	0.7499	
Social reciprocity [63]	Likert scale	NOR1	I believe that it is fair and obligatory to help others because I know that other people will help me some day	Dropped	0.8003
		NOR2	I believe that other people will help me when I need help if I share knowledge with others through MELANI	0.8464	
		NOR3	I believe that other people will answer my questions regarding specific information and knowledge in the future if I share knowledge with others through MELANI	0.8714	
		NOR4	I think that people who are involved with MELANI develop reciprocal beliefs on give and take based on other people's intentions and behavior	0.6946	
Executional cost [106]	Likert scale	EC1	I cannot seem to find the time to share knowledge in the community	0.6964	0.7882
		EC2	It is laborious to share knowledge in the community	0.6950	
		EC3	It takes me too much time to share knowledge in the community	0.8626	
		EC4	The effort is high for me to share knowledge in the community	0.7913	
Reputation [106]	Likert scale	R1	Sharing knowledge can enhance my reputation in the community	0.6312	0.6996
		R2	I get praises from others by sharing knowledge in the community	0.6890	
		R3	I feel that knowledge sharing improves my status in the community	0.7922	
		R4	I can earn some feedback or rewards through knowledge sharing that represent my reputation and status in the community	0.7039	
Trust [87]	Likert scale	TR1	I believe that my colleague's information security knowledge is reliable	0.7510	0.8598
		TR2	I believe that my colleague's information security knowledge is effective	0.8688	
		TR3	I believe that my colleague's information security knowledge mitigates the risk of information security breaches	0.8460	
		TR4	I believe that my colleague's information security knowledge is useful	0.8039	
		TR5	I believe that my colleagues would not take advantage of my information security knowledge that we share	Dropped	

Table 2: Final set of factor loadings after oblique rotation[a]

Item	Loading on oblimin-rotated factor							Uniqueness
	Factor 1	Factor 2	Factor 3	Factor 4	Factor 5	Factor 6	Factor 7	
ISKS1	0.8075							0.27
ISKS2	0.8903							0.19
ISKS3	0.885							0.20
ISKS4	0.86							0.21
ISKS5	0.6898							0.44
AT2							0.7751	0.32
AT3	0.3412						0.6376	0.38
AT4							0.7849	0.31
NOR2					0.8464			0.23
NOR3					0.8714			0.18
NOR4					0.6946			0.36
HR1				0.8822				0.16
HR2				0.8743				0.19
HR4				0.7499				0.41
EC1			0.6964					0.49
EC2			0.695					0.45
EC3			0.8626					0.21
EC4			0.7913					0.32
R1						0.6312		0.49
R2						0.689		0.51
R3						0.7922		0.29
R4						0.7039		0.44
TR1		0.751						0.36
TR2		0.8688						0.21
TR3		0.846						0.26
TR4		0.8039						0.29
Eigenvalue	3.786	2.951	2.502	2.329	2.24	2.142	1.851	
Proportion of variance explained (%)	14.56	11.35	9.62	8.96	8.62	8.24	7.12	
Cumulative variance explained (%)	14.56	25.91	35.53	44.49	53.11	61.34	68.46	

[a]Blank cells represent factor loadings smaller than 0.30.

Results

Table 3 provides descriptive statistics for all variables. Table 4 specifies Spearman correlations; for the sake of brevity, correlates for controls are omitted. Tables 5 and 6 document all models and their respective diagnostic measures. Since we handled missing data conservatively by list-wise deletion, the sample size of the respective models is smaller than that of the full sample.

H1 is partially supported. A positive attitude towards SIS is positively associated with the intensity ($P < 0.05$), but not with the frequency of SIS. This may suggest that whenever the focal individual believes SIS is an effective activity, his or her behavior is responsive to information shared by other individuals.

H2a is fully supported. Social reciprocity is associated with both the intensity ($P < 0.01$) and the frequency of SIS ($P < 0.05$). This finding is in line with our theoretical expectation that individuals seek equitable exchange relationships in which cooperative behavior is rewarded. Future research may explore such social interaction over time with a dyadic research setting, studying how exchange patterns of repeated reciprocation develop over time.

H2b is partially supported. Transactional reciprocity is associated with the frequency of SIS ($P < 0.01$), but not with its intensity. This may imply that transactional rewards such as bonuses or promotion motivate individuals to share knowledge they already possess with others in order to signal a high level of productive activity vis-à-vis their superiors.

H3 is fully supported. Consistent with our theoretical expectation, executional cost is negatively associated with both the frequency ($P < 0.05$) and the intensity ($P < 0.001$) of SIS. This not only signals that executional cost constitutes a form of transaction cost that may deter individuals from sharing, as we hypothesized. The negative association with intensity is much stronger, suggesting that the negative association of executional cost is larger when the focal individual reacts to information shared by others. In other words, in the presence of high executional cost, individuals seem to be punished for reacting. Since our research design only accounted for the presence of executional cost, more research is required to identify the institutional or organizational sources of this cost.

H4 is not supported. Contrary to what we hypothesized, we find no support for the claim that an individuals' expectation to increase his or her status or social esteem is associated with SIS. Our measure of reputation is neither significantly associated with the intensity nor with the frequency of SIS. This negative result may be due to the fact that Ref. 106 introduced their measure of reputation (which we use in our empirical study) in the context of public knowledge sharing among private individuals who vie for public social esteem. In contrast, we study a population of security professionals in the context of a private setting in which sensitive and classified information is shared. This may imply that, insofar as security information sharing is concerned, future research should propose alternative measures of reputation that are congruent with this context.

H5 is partially supported. The extent to which the focal individual trusts the information received will be useful is positively associated with the frequency ($P < 0.01$), but not with the intensity of SIS. This may imply that a focal individual who has such trust would be more willing to share knowledge s/he already possesses. In this respect, more research is required regarding the relationship between initial trust among individuals and the evolution of such trust as exchange relationships unfold.

As regards the interaction effects, we find that H6a is partially supported. The extent to which the focal individual trusts the information received will be useful negatively moderates the relationship between attitude and the intensity ($P < 0.05$), but not the frequency of SIS. This may imply that trust can function as a partial substitute for attitude, in that the focal individual needs to convince him- or herself to a lesser extent that SIS is useful in general if that individual trusts the particular information s/he is about to receive is useful.

H6b is not supported. The extent to which the focal individual trusts the information received will be useful neither moderates the positive association of social reciprocity with the intensity of SIS nor that with the frequency of SIS. This may imply that, unlike in the above case for H6a, the focal individual's trust that any particular unit of information is useful cannot function as a substitute for the importance of social reciprocity in the exchange relationship as such.

H6c is fully supported. The extent to which the focal individual trusts the counterparty provides valuable information negatively moderates both the association of transactional reciprocity with the frequency ($P < 0.01$) and with the intensity ($P < 0.05$) of SIS. In line with our theoretical reasoning, this result may suggest that trust can help the focal individual to convince him- or herself that the exchange relationship is equitable (since the information s/he is about to receive is trusted to be useful), such that the focal individual has to rely less on the expectation that s/he will be compensated by monetary or career benefits whenever s/he participates in exchange relationships.

Table 3: Descriptive statistics on all variables

Variable	Obs	Mean	SD	Min	Max
Frequency	240	2.68	0.78	1	5
Intensity	228	2.34	1.20	1	7
Attitude	208	4.10	0.53	3	5
Reciprocity (social)	195	3.88	0.60	1.66	5
Reciprocity (transactional)	195	2.16	0.75	1	4
Executional cost	208	3.14	0.65	1.25	5
Reputation	190	3.46	0.47	1.5	5
Trust	190	3.82	0.55	1.25	5
Gender	260	1.04	0.20	1	2
Age category	261	2.87	0.86	1	4
Education category	260	2.58	1.25	1	6
Membership duration	260	7.05	5.35	1	18

Finally, the fact that we find partial support for H1, H2b, H5, and H6a suggests that a differentiation of the theoretical construct SIS into different measurement constructs is productive. Future research may further develop the measures of frequency and intensity we have proposed here or develop yet other detailed operationalizations.

As regards our control variables, we find no significant association of respondents' demographic heterogeneity, length of membership in MELANI-net, or industry affiliation with SIS. The latter non-finding also alleviates concerns of overrepresentation of a particular industry or firm among the responses. For the controls "age," "industry," and "education," a benchmark category was automatically selected during estimation for every control (viz. footnotes b to Tables 5 and 6).

The only significant association we find relates to the control "education" in the model for the frequency of SIS. Since the education category "other" is used as the benchmark, the results suggest that in comparison to individuals with an education captured by "other," the remaining individuals in all other education categories share significantly less in terms of frequency ($P < 0.01$, respectively), whereas no association with intensity is presented. Since all other categories capture academic degrees and the case of no education, this may imply that individuals who have a non-academic education (e.g. vocational training) share knowledge they possess more often with other individuals, probably because they are industry practitioners who wish to propagate information they possess throughout and across industries to strengthen organizational practice.

Discussion

Building on prior research in the field of the economics of information security, and adopting a behavioral framework to organize our theoretical reasoning, we have proposed how and why human behavior should be associated with SIS. To the best of our knowledge, this study is the first that associates the self-reported sharing of sensitive information among real individuals inside a private Information Sharing and Analysis Center (ISAC) with the behavior of these individuals. We also provide a dual empirical

Table 4: Correlations among dependent and independent variables[a]

	Frequency	Intensity	Attitude	Reciprocity (social)	Reciprocity (transactional)	Executional cost	Reputation	Trust
Frequency	1							
Intensity	0.3547***	1						
Attitude	0.2436***	0.2742***	1					
Reciprocity (social)	0.2602***	0.2750***	0.3798***	1				
Reciprocity (transactional)	0.1836**	0.0456	−0.0901	0.000	1			
Executional cost	−0.2238**	−0.1694*	−0.0976	−0.0314	0.1533*	1		
Reputation	−0.0226	0.0968	0.1227	0.3069***	0.0270	0.1148	1	
Trust	0.2279**	−0.0101	0.2471***	0.0269***	−0.1321	−0.1857*	0.1042	1

[a] Spearman correlations.

*$P < 0.05$; **$P < 0.01$; ***$P < 0.001$.

Table 5: Models for intensity of SIS (ordered probit estimation)[a,b]

	Baseline Coefficient (robust standard error)	Main effects Coefficient (robust standard error)	Full model Coefficient (robust standard error)
Attitude		0.4973 (0.1609)**	0.3627 (0.1672)*
Reciprocity (social)		0.3481 (0.1549)*	0.4045 (0.1526)**
Reciprocity (transactional)		0.2254 (0.1138)*	0.1860 (0.1118)
Executional cost		−0.3949 (0.1198)***	−0.4833 (0.1314)***
Reputation		0.0083 (0.1905)	0.0932 (0.1895)
Trust		−0.2250 (0.1577)	−0.1847 (0.1501)
Attitude × trust			−0.6544 (0.2874)*
Reciprocity (social) × trust			0.1969 (0.2431)
Reciprocity (transactional) × trust			−0.4561 (0.2119)*
Gender	0.2045 (0.3712)	−0.1507 (0.4480)	−0.2106 (0.4788)
Age 21–30	−0.0920 (0.3434)	−0.1286 (0.4063)	−0.1361 (0.4204)
Age 31–40	0.0567 (0.2031)	0.0896 (0.2220)	0.1139 (0.2293)
Age 41–50	−0.0001 (0.1762)	−0.0138 (0.1777)	0.0096 (0.1820)
Education none	−0.2253 (0.4789)	−0.7976 (0.5208)	−0.7239 (0.6388)
Education Master/Diploma	−0.3512 (0.4649)	−0.8990 (0.4964)	−0.8336 (0.6368)
Education Bachelor	0.0206 (0.4635)	−0.3347 (0.4924)	−0.3198 (0.6202)
Education PhD	−0.4581 (0.4984)	−0.8959 (0.5322)	−0.9997 (0.6382)
Membership duration	0.0257 (0.0134)	0.0184 (0.0165)	0.0184 (0.0164)
Government	−0.1539 (0.2729)	−0.2662 (0.3125)	−0.2945 (0.3082)
Banking and Finance	−0.0672 (0.2098)	−0.1598 (0.2527)	−0.1515 (0.2472)
All other industries	−0.0472 (0.2473)	−0.1649 (0.2977)	−0.1576 (0.2982)
Energy	0.0283 (0.2931)	−0.0650 (0.3260)	−0.1007 (0.3217)
Health	−0.3250 (0.2638)	−0.2498 (0.3260)	−0.2958 (0.3528)
Log pseudolikelihood	−318.98	−249.82	−246.50
Pseudo R^2	0.0214	0.0773	0.0896
Wald χ^2 (df)	16.10 (14)	55.43 (20)***	64.02 (23)***
Observations	225	188	188
AIC ‖ BIC	677.97 ‖ 746.29	551.65 ‖ 635.80	551.02 ‖ 644.87

[a] Two-tailed tests.

[b] Age category "above 50," education category "other" and the telecommunication/IT industry serve as the respective control variable benchmarks.

*$P < 0.05$; **$P < 0.01$; ***$P < 0.001$.

operationalization of SIS by introducing the measures of SIS frequency and intensity. Finally, our study confirms that interdisciplinary approaches which attempt to integrate thinking from economics and psychology are useful when SIS is studied [6].

Our study also contributes to prior work that has both theoretically predicted and descriptively noted that SIS, while beneficial, is underutilized [16, 32, 33, 44, 47, 72, 73, 103]. We provide some first empirical evidence on the association of particular human behaviors with SIS among individuals in a private ISAC setting. The study also contributes to understanding the theoretical prediction that actual SIS may not reach its societally optimal level [41, 47] by suggesting that human behavior may be at the core of this problem. At the same time, we would caution regulators and researchers to infer that SIS should be mandated (i.e. that individuals should be forced to share) as a consequence of this problem. Adjusting sanction levels for failure to comply with mandatory SIS could be difficult, if not impossible [65]. Moreover, regulation that attempts to solve the "sharing dilemma" in SIS should try to fix causes, not symptoms [19]. Our study has collected cross-sectional data, and hence we cannot establish causal relationships between human behavior and SIS. Nevertheless, the negative and significant association between executional cost and both the frequency and intensity of SIS that we identify confirms prior research that finds that institutions shape human interaction and behavior. Institutions are formal and informal rules which govern human behavior by rewarding desirable actions and making undesirable actions more expensive or punishable [12, 75, 76]. The organization of an ISAC is shaped by both internal institutions (i.e. rules voluntarily agreed to among ISAC participants and organizers) and external institutions (i.e. rules imposed onto them by government and regulatory authorities). Since high executional cost can be attributed to both effects, legislators, and regulators should be careful to predict the impact and consequences of intended regulation for the executional cost of SIS. The association between executional cost and SIS that our study identifies suggests that humans are likely to assess the economic consequences of external institutions in terms of executional costs and adapt their behavior accordingly. Moreover, we find that both social and transactional reciprocity are positively associated with both the frequency and the intensity of SIS. Since reciprocity is a social norm, it cannot be enforced by formal regulation and constraint, and the attempt to do so may induce individuals to comply with the letter rather than the spirit of the law by sharing irrelevant, non-timely, or false information [23].

We believe that the future study of these issues opens up promising paths for research that can both explain why individuals attempt to circumvent SIS regulation and suggest more conducive institutions. In this way, our study provides a stepping stone on which future research can build. The extant literature has documented well that actual SIS, while considered highly useful in general, is at low levels, and that individuals attempt to circumvent regulation that makes SIS mandatory [5, 32, 33, 71, 72]. Our study adds to these findings by suggesting that this economic problem of

Table 6: Models for frequency of SIS (Tobit estimation)[a,b]

	Baseline	Main effects	Full model
	Coefficient (robust standard error)	Coefficient (robust standard error)	Coefficient (robust standard error)
Attitude		0.2797 (0.1214)*	0.1895 (0.1111)
Reciprocity (social)		0.1807 (0.1195)	0.2150 (0.1046)*
Reciprocity (transactional)		0.2734 (0.0824)**	0.2361 (0.0816)**
Executional cost		−0.1872 (0.0911)*	−0.2336 (0.0962)*
Reputation		−0.1827 (0.1243)	−0.1121 (0.1232)
Trust		0.2689 (0.1058)*	0.2964 (0.1036)**
Attitude × trust			−0.3490 (0.2311)
Reciprocity (social) × trust			0.2055 (0.1813)
Reciprocity (transactional) × trust			−0.3839 (0.1378)**
Gender	0.4851 (0.1681)**	0.2412 (0.1791)	0.1837 (0.1773)
Age 21–30	0.1852 (0.2131)	0.2595 (0.2387)	0.2057 (0.2378)
Age 31–40	−0.0365 (0.1567)	0.0218 (0.1528)	0.0051 (0.1513)
Age 41–50	0.0294 (0.1222)	0.0040 (0.1264)	0.0171 (0.1243)
Education none	−0.6274 (0.1705)***	−0.9126 (0.2210)***	−0.8152 (0.2441)**
Education Master/Diploma	−0.6063 (0.1462)***	−0.8749 (0.2291)***	−0.7984 (0.2671)**
Education Bachelor	−0.5872 (0.1531)***	−0.8089 (0.2062)***	−0.7678 (0.2324)**
Education PhD	−0.5392 (0.2667)*	−0.8892 (0.2976)**	−0.9345 (0.3181)**
Membership duration	0.0277 (0.0112)*	0.0211 (0.0118)	0.0213 (0.0112)
Government	−0.1629 (0.2039)	0.0130 (0.2373)	−0.0097 (0.2288)
Banking and Finance	−0.0613 (0.1694)	0.0328 (0.2142)	0.0304 (0.2064)
All other industries	−0.5292 (0.1947)**	−0.4016 (0.2430)	−0.3748 (0.2395)
Energy	0.1054 (0.2236)	0.2191 (0.2485)	0.1867 (0.2399)
Health	−0.0909 (0.2115)	0.0767 (0.2787)	0.0465 (0.2759)
Constant	2.6652 (0.2705)***	3.0954 (0.3520)***	3.0939 (0.3577)***
Log pseudolikelihood	−274.73	−202.46	−197.92
Pseudo R^2	0.0538	0.1370	0.1564
F (df)	4.10 (14, 223)***	5.25 (20, 168)***	5.25 (23, 165)***
Observations (left ‖ right censored)	237 (12 ‖ 1)	188 (10 ‖ 1)	188 (10 ‖ 1)
AIC ‖ BIC	581.47 ‖ 636.96	448.93 ‖ 520.13	445.84 ‖ 526.75

[a] Two-tailed tests.

[b] Age category “above 50,” education category “other” and the telecommunication/IT industry serve as the respective control variable benchmarks.

*$P < 0.05$; **$P < 0.01$; ***$P < 0.001$.

underutilization is difficult to resolve unless regulators and lawmakers consider the association of human behavior and SIS outcomes. At this time, we speculate that a liberal institutional environment that attempts to make individuals comply by “nudging” them is probably more conducive than the attempt to enforce compliance by coercion [91]. We leave it to future research to either corroborate or refute this speculation, suggesting that irrespective of any particular institutional arrangement, human behavior is significantly associated with SIS and hence likely responds to changes in institutional configuration. All in all, our study suggests that future research can productively employ behavioral theory and methods as it attempts to further develop SIS research by considering the human interaction that precedes actual acts of sharing.

In a broader sense, our work develops prior conceptual ideas that human aspects matter at least as much as technological ones when SIS is concerned [19]. Our empirical approach takes the technological context as a given and focuses on identifying associations between human behavior and SIS. Cybersecurity managers in organizations can benefit from these results as they attempt to make individuals comply with organizational goals. Our results suggest that both the frequency and the intensity of SIS are associated with human behavior. Managers should therefore be careful to study these associations when they define organizational goals and accept that individual human behavior does not necessarily comply with these unless appropriate goal alignment is provided [57, 66]. For example, managers may facilitate an individual’s participation in SIS by reducing the executional cost of information exchange, or they may provide the focal individual with intelligence on counterparties to help them assess the likelihood with which information sharing may be reciprocated.

Our study is pioneering in the sense that it studies real human beings and their self-reported behavior in the context of a real ISAC. Nevertheless, it merely studies a single, centrally organized ISAC in a single country. Hence, future research should generalize our approach to alternative models of ISAC organization and explore diverse national and cultural settings by replicating our study with different ISACs and nation-states. We believe our approach is conducive to such generalization since neither our theoretical framework, nor any one of our behavioral constructs, nor the empirical measures we used to operationalize these are context-specific to any particular national or cultural context. Our measures and the theory in which they are grounded rather represent fundamental aspects of human behavior which, in our view, should apply globally. Thus, future work could complement our study with data from different ISACs, such that a transnational continuum of sharing intensities and frequencies could be constructed. This continuum would allow researchers to identify commonalities and differences in information exchange patterns and use these insights to propose expedient policy options.

Finally, the ISACs that exist as of today have evolved from trade associations, government agencies, and public–private partnerships. However, the evolution of such historical trajectories is subject to technological change [74]. We therefore believe that novel technologies could facilitate human interaction in future ISAC configurations. For example, since the cost of reputation losses upon security breaches can be interpreted as privacy risk [19], insights from privacy research and secure distributed computation and interaction [35] might be used to construct distributed ISACs with safe real-time participation. Future research may use our study to consider the impact of such novel technological approaches on human behavior to prevent unintended consequences.

From a broader perspective, our study design has some limitations that point to opportunities for future research.[4] First, both as regards the level and the unit of analysis, our study focuses on the individual. This implies that interactions between the individual and the organizational and institutional contexts within which the focal individual acts are beyond the scope of this study. Nevertheless, our setting may be expanded both theoretically and empirically to incorporate such multilevel interactions. For example, the organizational-level performance implications of SIS could be studied, in that future research would analyze the association of individual behavior with organizational results, such as increased cybersecurity or increased financial performance.

In particular, future research may analyze the extent to which different organizational processes, cultures, and risk management approaches are associated with SIS by way of human behavior. For example, critical infrastructure providers who face significant risks of business interruption and going concern if their cybersecurity is compromised may emphasize more than other organizations that SIS is desirable and hence direct their employees to act accordingly. Thus, organizational policy may moderate the association between human behavior and SIS. Future research could build on our approach by developing more complex multilevel study designs that can incorporate such additional sources of variance.

Finally, our study design is cross-sectional, implying that we can only claim association, but not causation. While we believe this is acceptable given the pioneering nature of this study, controlled experiments are required to establish causality. We encourage future work to introduce such methods. Further, future studies could also ethnographically analyze human interaction within an ISAC over time, log how and why behavior changes, and infer how this behavioral evolution operates on SIS outcomes.

Conflict of interest statement. None declared.

Acknowledgments

Comments received from Tyler Moore, Thomas Maillart, Mauro Cherubini, and three anonymous reviewers all helped to improve previous versions of this article. We thank the Swiss Federal Institute of Technology Zurich for providing free software and IT support during the survey implementation phase. The usual disclaimer applies.

References

1. Ajzen I. The directive influence of attitudes on behavior. In: Gollwitzer PM and Bargh JA (eds), *The Psychology of Action: Linking Cognition and Motivation to Behavior*. New York, NY: Guilford Press, 1996, 385–403.
2. Ajzen I, Fishbein M. *Understanding Attitudes and Predicting Social Behavior*. Englewood Cliffs, NJ: Prentice-Hall, 1980.
3. Ajzen I, Madden TJ. Prediction of goal-directed behavior: attitudes, intentions, and perceived behavioral control. *J Exp Soc Psychol* 1986;**22**:453–74.
4. Amayah AT. Determinants of knowledge sharing in a public sector organization. *J Knowledge Manage* 2013;**17**:454–71.
5. Anderson R, Fuloria S. Security economics and critical national infrastructure. In: Moore T, Pym D, Ioannidis C (eds), *Economics of Information Security and Privacy*. Boston, MA: Springer, 2010, 55–66.
6. Anderson R, Moore T. The economics of information security. *Science* 2006;**314**:610–13.
7. Andreoni J. Cooperation in public-goods experiments: kindness or confusion? *Am Econ Rev* 1995;**85**:891–904.
8. Aviram A, Tor A. Overcoming impediments to information sharing. *Ala Law Rev* 2003;**55**:231–80.
9. Ba S, Pavlou P. Evidence of the effect of trust building technology in electronic markets: price premiums and buyer behavior. *MIS Quart* 2002;**26**:243–68.
10. Bauer J, van Eeten M. Cybersecurity: stakeholder incentives, externalities, and policy options. *Telecommun Policy* 2009;**33**:706–19.
11. Baumeister RF, Leary MR. The need to belong: desire for interpersonal attachments as a fundamental human motivation. *Psychol Bull* 1995;**117**:497–529.
12. Baumol WJ. Entrepreneurship: productive, unproductive, and destructive. *J Political Econ* 1990;**98**:893–921.
13. Bazerman MH. *Judgement in Managerial Decision Making*. New York, NY: Wiley, 2005.
14. Belletier C, Robert A, Motak L, *et al*. Toward explicit measures of intention to predict information system use: an exploratory study of the role of implicit attitudes. *Comput Human Behav* 2018;**86**:61–8.
15. Bénabou R, Tirole J. Incentives and prosocial behavior. *Am Econ Rev* 2006;**96**:1652–78.
16. Bisogni F. Data breaches and the dilemmas in notifying customers. In: *Workshop on the Economics of Information Security (WEIS)*, Delft, 2015.
17. Bock GW, Zmud RW, Kim YG, *et al*. Behavioral intention formation in knowledge sharing: examining the roles of extrinsic motivators, social–psychological forces, and organizational climate. *MIS Quart* 2005;**29**:87–112.
18. Bodin LD, Gordon LA, Loeb MP, *et al*. Cybersecurity insurance and risk-sharing. *J Account Pub Policy* 2018;**37**:527–44.
19. Böhme R. Back to the roots: information sharing economics and what we can learn for security. In: *Second Workshop on Information Sharing and Collaborative Security (WISCS)*, Denver, CO: ACM, 2015.
20. Bolton GE, Ockenfels A. ERC: a theory of equity, reciprocity, and competition. *Am Econ Rev* 2000;**90**:166–93.
21. Brennan G, Pettit P. *The Economy of Esteem*. Oxford: Oxford University Press, 2004.
22. Brosnan SF, de Waal F. Monkeys reject unequal pay. *Nature* 2003;**425**:297–99.
23. Burr R. *To Improve Cybersecurity in the United States through Enhanced Sharing of Information about Cybersecurity Threats, and for Other Purposes*. Washington, DC: 114th United States Congress, 2015.
24. Chaiken S. Heuristic versus systematic information processing and the use of source versus message cues in persuasion. *J Pers Soc Psychol* 1980;**39**:752–66.
25. Chang HH, Chuang S-S. Social capital and individual motivations on knowledge sharing: participant involvement as a moderator. *Inf Manage* 2011;**48**:9–18.
26. Cohen J, Cohen P, West SG, *et al*. *Applied Multiple Regression/Correlation Analysis for the Behavioral Sciences*. 3rd ed. London: Taylor & Francis, 2002.

[4] We thank two anonymous reviewers for sharing ideas about how our approach may be expanded and generalized.

27. DellaVigna S. Psychology and economics: evidence from the field. *J Econ Lit* 2009;**47**:315–72.
28. Dillman DA, Smyth J, Christian LM. *Internet, Phone, Mail, and Mixed-Mode Surveys: The Tailored Design Method*, 4th edn. Hoboken, New Jersey: John Wiley & Sons, 2014.
29. Dunn Cavelty M. *Cybersecurity in Switzerland. Springer Briefs in Cybersecurity*. Cham: Springer International Publishing, 2014.
30. Eagly AH, Chaiken S. *The Psychology of Attitudes*. Fort Worth, TX: Harcourt et al., 1993.
31. Emler N. A social psychology of reputation. *Eur Rev Soc Psychol* 1990; **1**:171–93.
32. ENISA. *Incentives and Barriers to Information Sharing*. Heraklion: European Union Agency for Network and Information Security, 2010.
33. ENISA. *Information Sharing and Common Taxonomies between CSIRTs and Law Enforcement*. Heraklion: European Union Agency for Network and Information Security, 2016.
34. ENISA. *Information Sharing and Analysis Centres (ISACs). Cooperative Models*. Heraklion: European Union Agency for Network and Information Security, 2017.
35. Ezhei M, Ladani BT. Information sharing vs. privacy: a game theoretic analysis. *Expert Syst Appl* 2017;**88**:327–37.
36. Fehr E, Gächter S. Fairness and retaliation: the economics of reciprocity. *J Econ Perspect* 2000; **14**:159–81.
37. Fehr E, Gächter S. Altruistic punishment in humans. *Nature* 2002;**415**: 137–40.
38. Fehr E, Gintis H. Human motivation and social cooperation: experimental and analytical foundations. *Annu Rev Sociol* 2007;**33**:43–64.
39. Fehr E, Schmidt K. A theory of fairness, competition, and cooperation. *Quart J Econ* 1999;**114**:817–68.
40. Gabaix X, Laibson D, Moloche G, *et al*. Costly information acquisition: experimental analysis of a boundedly rational model. *Am Econ Rev* 2006;**96**:1043–68.
41. Gal-Or E, Ghose A. The economic incentives for sharing security information. *Inf Syst Res* 2005;**16**:186–208.
42. Gal-Or E, Ghose A. The economic consequences of sharing security information. In: Camp LJ, and Lewis S (eds), *Economics of Information Security. Advances in Information Security*, Vol. **12**. Boston, MA: Springer, 2004.
43. Gefen D, Karahanna E, Straub DW. Trust and TAM in online shopping: an integrated model. *MIS Quart* 2003;**27**:51–90.
44. Ghose A, Hausken K. A strategic analysis of information sharing among cyber attackers. *SSRN Electron J* 2006;**12**:1–37.
45. Gordon LA, Loeb MP, Lucyshyn W, *et al*. Externalities and the magnitude of cyber security underinvestment by private sector firms: a modification of the Gordon-Loeb model. *J Inf Secur* 2015;**6**:24–30.
46. Gordon LA, Loeb MP, Lucyshyn W, *et al*. The impact of information sharing on cybersecurity underinvestment: a real options perspective. *J Account Public Policy* 2015;**34**:509–519.
47. Gordon LA, Loeb MP, Lucyshyn W. Sharing information on computer systems security: an economic analysis. *J Account Public Policy* 2003;**22**: 461–85.
48. Gordon LA, Loeb MP, Sohail T. Market value of voluntary disclosures concerning information security. *MIS Quart* 2010;**34**:567–94.
49. Gordon LA, Loeb MP, Zhou L. Investing in cybersecurity: insights from the Gordon-Loeb model. *J Inf Secur* 2016;**7**:49.
50. Gouldner AW. The norm of reciprocity: a preliminary statement. *Am Sociol Rev* 1960;**25**:161–78.
51. Granovetter M. Economic action and social structure: the problem of embeddedness. *Am J Sociol* 1985;**91**:481–510.
52. Hair JF, Black WC, Babin BJ, *et al*. *Multivariate Data Analysis*, 5th edn. Upper Saddle River, NJ: Prentice Hall, 2009.
53. Harrison K, White G. Information sharing requirements and framework needed for community cyber incident detection and response. In: *2012 IEEE Conference on Technologies for Homeland Security (HST)*, Waltham, IEEE, 2012, 463–69.
54. Hausken K. A strategic analysis of information sharing among cyber attackers. *J Inf Syst Technol Manage* 2015;**12**:245–70.
55. Hausken K. Information sharing among firms and cyber attacks. *J Account Public Policy* 2007;**26**:639–88.
56. Hsu M-H, Ju TL, Yen C-H, *et al*. Knowledge sharing behavior in virtual communities: the relationship between trust, self-efficacy, and outcome expectations. *Int J Human-Comput Stud* 2007;**65**:153–69.
57. Hume D. *A Treatise of Human Nature*. New York, NY, Oxford University Press, 2000.
58. Kahneman D, Tversky A. Prospect theory: an analysis of decision under risk. *Econometrica* 1979;**47**:263–91.
59. Kahneman D, Tversky A. Prospect theory—an analysis of decision under risk. *Econometrica* 1979;**47**:263–91.
60. Keith B, Babchuk N. The quest for institutional recognition: a longitudinal analysis of scholarly productivity and academic prestige among sociology departments. *Social Forces* 1998;**76**:1495–1533.
61. Kolm S-C, Ythier JM. *Handbook of the Economics of Giving, Altruism and Reciprocity*. Amsterdam: Elsevier, 2006.
62. Kroenung J, Eckhardt A. The attitude cube—a three-dimensional model of situational factors in IS adoption and their impact on the attitude–behavior relationship. *Inf Manage* 2015;**52**:611–27.
63. Kwahk K-Y, Park D-H. The effects of network sharing on knowledge-sharing activities and job performance in enterprise social media environments. *Comput Human Behav* 2016;**55**:826–39.
64. Laube S, Böhme R. Strategic aspects of cyber risk information sharing. *ACM Comput Surv (CSUR)* 2017;**50**:1–77.
65. Laube S, Böhme R. The economics of mandatory security breach reporting to authorities. *J Cybersecur* 2016;**2**:29–41.
66. Lindenberg S, Foss N. Managing joint production motivation: the role of goal framing and governance mechanisms. *Acad Manage Rev* 2011;**36**:500–25.
67. Luiijf E, Klaver M. On the sharing of cyber security information. In: Rice M, Shenoi S (eds), *Critical Infrastructure Protection IX*. Cham: Springer, 2015, 29–46.
68. Malhotra D. Trust and reciprocity decisions: the differing perspectives of trustors and trusted parties. *Org Behav Human Decis Process* 2004;**94**: 61–73.
69. McElreath R. Reputation and the evolution of conflict. *J Theor Biol* 2003;**220**:345–57.
70. McEvily B, Perrone V, Zaheer A. Trust as an organizing principle. *Org Sci* 2003;**14**:91–103.
71. Moore T. The economics of cybersecurity: principles and policy options. *Int J Crit Infrastruct Prot* 2010;**3**:103–17.
72. Moran T, Moore T. The Phish-Market protocol: securely sharing attack data between competitors. In: Sion R (ed.), *Financial Cryptography and Data Security*. Berlin-Heidelberg: Springer, 2010, 222–37.
73. Naghizadeh P, Liu M. Inter-temporal incentives in security information sharing agreements. In: *2016 Information Theory and Applications Workshop (ITA)*. IEEE, 2016, 1–8.
74. Nelson RR, Winter SG. *An Evolutionary Theory of Economic Change*. Cambridge: Belknap Press, 1982.
75. North DC. *Institutions, Institutional Change and Economic Performance*. Cambridge: Cambridge University Press, 1990.
76. North DC. *Understanding the Process of Economic Change*. Cambridge: Cambridge University Press, 2005.
77. Nunnally JC, Bernstein I. 2017. *Psychometric Theory*, 3rd edn. New York: McGraw-Hill.
78. Oliver P. Rewards and punishments as selective incentives for collective action: theoretical investigations. *Am J Sociol* 1980;**85**:1356–75.
79. Olson M. *The Logic of Collective Action*. Cambridge, MA: Harvard University Press, 1965.
80. Paese PW, Gilin DA. When an adversary is caught telling the truth: reciprocal cooperation versus self-interest in distributive bargaining. *Pers Soc Psychol Bull* 2000;**26**:79–90.
81. Park JH, Gu B, Leung ACM, *et al*. An investigation of information sharing and seeking behaviors in online investment communities. *Comput Human Behav* 2014;**31**:1–12.

82. Petty RE, Cacioppo, JT. The elaboration likelihood model of persuasion. *Adv Exp Soc Psychol* 1986;**19**:123–205.
83. Podsakoff PM, MacKenzie S, Podsakoff NP. Sources of method bias in social science research and recommendations on how to control it. *Annu Rev Psychol* 2012;**63**:539–69.
84. Podsakoff PM, Organ DW. Self-reports in organizational research: problems and prospects. *J Manage* 1986;**12**:531–44.
85. Reinholt MIA, Pedersen T, Foss NJ. Why a central network position isn't enough: the role of motivation and ability for knowledge sharing in employee networks. *Acad Manage J* 2011;**54**:1277–97.
86. Ridings CM, Gefen D, Arinze B. Some antecedents and effects of trust in virtual communities. *Strategic Inf Syst* 2002;**11**:271–95.
87. Safa NS, von Solms R. An information security knowledge sharing model in organizations. *Comput Human Behav* 2016;**57**:442–51.
88. Siegrist J, Starke D, Chandola T, *et al*. The measurement of effort–reward imbalance at work: European comparisons. *Soc Sci Med* 2004; **58**:1483–99.
89. Simon HA. *Administrative Behavior: A Study of Decision-Making Processes in Administrative Organization*. New York, NY: Free Press, 1976.
90. Smith EA, Winterhalder B. (eds) *Evolutionary Ecology and Human Behavior.* New York, NY: Routledge, 2017.
91. Thaler RH, Sunstein CR. *Nudge: Improving Decisions about Health, Wealth, and Happiness*. New York, NY: Penguin Books, 2009.
92. Tom S, Fox C, Trepel C, *et al*. The neural basis of loss aversion in decision-making under risk. *Science* 2007;**315**:515–518.
93. Tomasello M, Carpenter M, Call J, *et al*. Understanding and sharing intentions: the origins of cultural cognition. *Behav Brain Sci* 2005;**28**: 675–91.
94. Trevor CO, Nyberg AJ. Keeping your headcount when all about you are losing theirs: downsizing, voluntary turnover rates, and the moderating role of HR practices. *Acad Manage J* 2008;**51**:259–76.
95. Tricomi E, Rangel A, Camerer C, *et al*. Neural evidence for inequality-averse social preferences. *Nature* 2010;**463**:1089–1091.
96. Tversky A, Kahneman D. Loss aversion in riskless choice: a reference dependent model. *Quart J Econ* 1991;**106**:1039–61.
97. Tversky A, Kahneman D. Advances in prospect theory: cumulative representation of uncertainty. *J Risk Uncertainty* 1992;**5**:297–323.
98. Vakilinia I, Louis SJ, Sengupta S. Evolving sharing strategies in cybersecurity information exchange framework. In: *Proceedings of the Genetic and Evolutionary Computation Conference Companion*. New York, NY: ACM, 2017, 309–10.
99. von Hippel E, von Krogh G. Open source software and the "private-collective" innovation model: issues for organization science. *Org Sci* 2003;**14**:209–23.
100. Wang W-T, Hou Y-P. Motivations of employees' knowledge sharing behaviors: a self-determination perspective. *Inf Org* 2015;**25**:1–26.
101. Wang JH, Wang C, Yang J, *et al*. A study on key strategies in P2P file sharing systems and ISPs' P2P traffic management. *Peer-to-Peer Network Appl* 2011;**4**:410–19.
102. Watzlawick P, Bavelas JB, Jackson DD. *Pragmatics of Human Communication*, New York, Norton & Company, 2011.
103. Weiss E. *Legislation to Facilitate Cybersecurity Information Sharing: Economic Analysis*. Washington, DC: Congressional Research Service, 2015.
104. Williamson OE. The economics of organization: the transaction cost approach. *Am J Sociol* 1981;**87**:548–77.
105. Xiong L, Liu L. PeerTrust: supporting reputation-based trust for peer-to-peer electronic communities. *IEEE Trans Knowledge Data Eng* 2004;**16**: 843–57.
106. Yan Z, Wang T, Chen Y, *et al*. Knowledge sharing in online health communities: a social exchange theory perspective. *Inf Manage* 2016;**53**: 643–53.
107. Zhao W, White G. A collaborative information sharing framework for Community Cyber Security. In: *IEEE Conference on Technologies for Homeland Security (HST)*, Waltham, MA: IEEE, 2012, 457–62.

OpenSky Report 2019: Analysing TCAS in the Real World using Big Data

Matthias Schäfer[¶‡*], Xavier Olive[¶||], Martin Strohmeier[¶†§], Matthew Smith[¶†], Ivan Martinovic[¶†], Vincent Lenders[¶§]

[¶]OpenSky Network, Switzerland
lastname@opensky-network.org

[*]TU Kaiserslautern, Germany
schaefer@cs.uni-kl.de

[†]University of Oxford, UK
firstname.lastname@cs.ox.ac.uk

[‡]SeRo Systems, Germany
schaefer@sero-systems.de

[§]armasuisse, Switzerland
firstname.lastname@armasuisse.ch

[||]ONERA, Université de Toulouse, France
xavier.olive@onera.fr

Abstract—**Collision avoidance is one of the most crucial applications with regards to the safety of the global airspace. The introduction of mandatory airborne collision avoidance systems has significantly reduced the likelihood of mid-air collisions despite the increase in air traffic density.**

In this paper, we analyze 250 billion aircraft transponder messages received from 126,700 aircraft by the OpenSky Network over a two-week period. We use this data to quantify equipage and usage aspects of Traffic Alert and Collision Avoidance System (TCAS) as it is working in the real world. We furthermore provide an overview of the methods used by OpenSky to collect, decode and store this data for use by other researchers and aviation authorities.

We observe that around 89.5% of the ADS–B-equipped aircraft have an operational TCAS. We further analyze the concrete usage of TCAS by examining several case studies where a loss of separation between aircraft has happened.

I. Introduction

Collision avoidance is one of the most crucial applications with regards to the safety of the global airspace. Since the introduction of mandatory airborne collision avoidance systems (ACAS) in the 1980s [1], they have helped reduce the likelihood of mid-air collisions despite a significant increase in air traffic density. In a recent survey among aviation professionals, it has been considered one of the most safety-relevant communications technologies on-board an aircraft [2].

Whilst there is no doubt as to the principal efficacy of ACAS, or more specifically its implementation, the Traffic Alert and Collision Avoidance System (TCAS), many details about its large-scale usage are not available. Under the current analytics system, pilots have to fill in a report if they encounter a TCAS resolution advisory during flight. Naturally, like any system relying purely on human reporting, the number of unreported cases is unknown and potentially very high. Collecting the true TCAS data broadcast by the aircraft themselves can help address these issues and improve the common knowledge about the efficacy of the current collision avoidance implementations.

With regards to collision avoidance, resolution advisories are naturally highly interesting, as they give direct insights into potential safety incidents and loss of separation.

Besides resolution advisories, traffic advisories can provide early indications of potential issues with the system. Finally, the equipage statistics regarding TCAS are of interest as they provide important information about the type and versions used by aircraft in the wild as well as the speed of upgrades and adoption.

In this paper, we provide unique insights into the global functioning of the collision avoidance system, and the data collection challenges that we encountered during the 7 years of operation of the OpenSky Network. We use a large set of crowdsourced surveillance data gathered by the network to analyze and quantify equipage and usage aspects of TCAS as it is working in the real world. We furthermore provide an overview of the methods used by OpenSky to collect, decode and store this data for use by researchers and authorities.

Concretely, we provide insights into the following topics:

- **TCAS Data Collection:** We explain our method of data collection as much of the relevant TCAS data is not simply broadcast (such as ADS-B) but instead has to be extracted from the Ground-Initiated Comm B (GICB) Registers.
- **TCAS Equipage:** Further, we analyze the OpenSky data set with regards to the TCAS versions (if any) used by the tracked planes.
- **TCAS Usage:** Finally, we look at the usage of TCAS in practice, providing several case studies. We analyze resolution advisories transmitted by aircraft and look at characteristics of the situations and the type of aircraft involved.

The remainder of this paper is organized as follows. Section II outlines the necessary background on the TCAS technology. Section IV describes the current state of the OpenSky Network and its newly-realized TCAS integration. Section VI provides statistics on the real-world equipage of TCAS while Section VII analyzes the usage and impact of its collision avoidance functions by examining several case studies. Section VIII discusses our experiences and finally Section IX concludes this work.

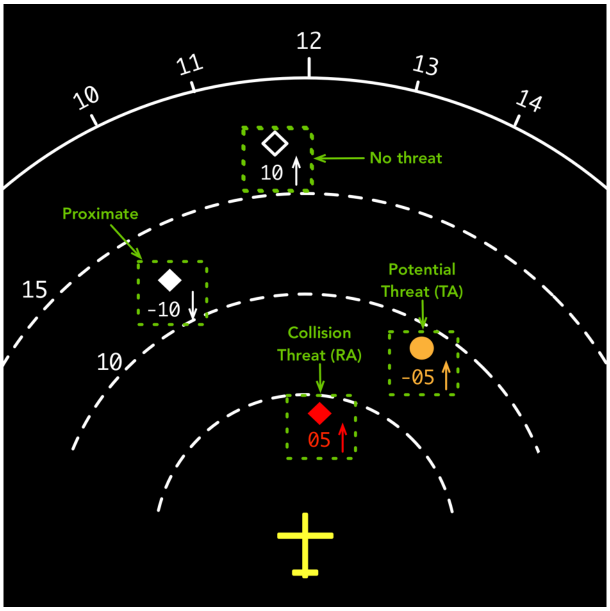


Fig. 1. A representation of TCAS data as seen by the pilot in the cockpit of an airliner. This is based on the Airbus Navigation Display (ND). Dashed semi-circles represent intervals at a range selected by the pilot, and numbers around the solid semi-circle are heading values.

II. Background: The Traffic Alert and Collision Avoidance System

Although airspace is tightly controlled by air traffic control (ATC), situations can arise where aircraft come too close to each other. This has resulted in mid-air collisions, such as the 1996 Charkhi Dadri crash, where an aircraft unduly descended and collided with another whilst under ATC control [3]. Incidents such as these have led to many regulators requiring aircraft to be equipped with collision avoidance systems, which may take over from ATC control when a dangerous situation arises.

TCAS is an implementation of the Airborne Collision Avoidance System (ACAS), designed to help reduce the chance of a mid-air collision [4], [5]. It has been required in some form on many aircraft since 1993, with TCAS II being introduced in 1998 [6]. In a situation where the risk of a mid-air collision is unacceptable (i.e. two aircraft are on course to collide soon), TCAS on each aircraft will communicate and negotiate actions for each aircraft [5].

The system on board the aircraft uses Mode C and S transmissions to detect and notify nearby aircraft of its existence, the responses from which are then processed and displayed to the crew. This will typically be presented as in Figure 1, with threats ahead of the aircraft being shown. Other aircraft can be *no threat*, *proximate*, a *potential threat* or a *collision threat*, depending on their distance, rate of closure and altitude difference. If an aircraft is a *potential threat*, a Traffic Advisory (TA) is given to crew, warning them of a potential intruder. If the intruding aircraft gets closer, a Resolution Advisory (RA) alert is given: the crew must ignore ATC instructions and follow the RA instructions.

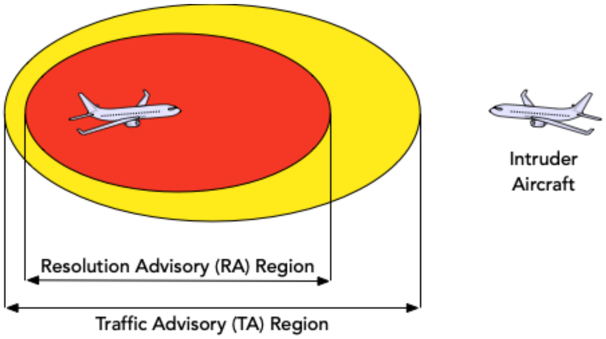


Fig. 2. Representation of TCAS Traffic (TA) and Resolution Advisory (RA) zones.

Although ATC manage airspace with high precision, aircraft can still end up closer than is safe. This is called a *loss of separation*, and in the worst case, can result in a mid-air collision. One such example occurred in March 2011, where a Delta aircraft took off with an inactive transponder, becoming too close to three other aircraft before resolving the issue [7]. TCAS provides a technical means by which to avoid this, and has been mandated on aircraft with more than 30 seats since 1993 [5], [6].

System Description

Establishing nearby aircraft with Mode S simply requires the object aircraft to listen for Mode S transmissions or ‘squitters’, the latter being messages transmitted periodically without prior interrogation. These contain the International Civil Aviation Organization’s (ICAO) transponder IDs, so the object aircraft follows up with Mode S interrogations to establish the position of the nearby aircraft. Heading and range are determined using the object aircraft’s directional antenna and the response time and altitude data is provided by the nearby aircraft from its instruments. Based on these data, the potential for conflict is calculated on the object aircraft. Depending on the proximity and closing speed of the target the interrogation rate will vary; at a large distance this will be once per five seconds, increasing to once per second when an aircraft is close [5]. An abstracted protocol diagram for Mode S can be seen in Figure 3 (top).

Mode C operates slightly differently, represented in Figure 3 (bottom). The object aircraft will issue Mode C-only all-calls, causing nearby aircraft with Mode C transponders to respond, at a rate of once per second. If the target has an altimeter then it will respond with its altitude, else TCAS onboard the object aircraft will use response characteristics to estimate altitude as well as range and bearing [5]. TCAS will only provide full alerting as below if Mode C-equipped aircraft provide altitude.

Through one of these methods, TCAS ascertains how close the nearby aircraft is both laterally and vertically, before deciding if it is necessary to alert the flight crew. For most systems, especially those on commercial aircraft, alerts are composed of two steps as shown in Figure 2. First comes a *traffic advisory* (TA), in which the traffic is typically displayed to the pilot as

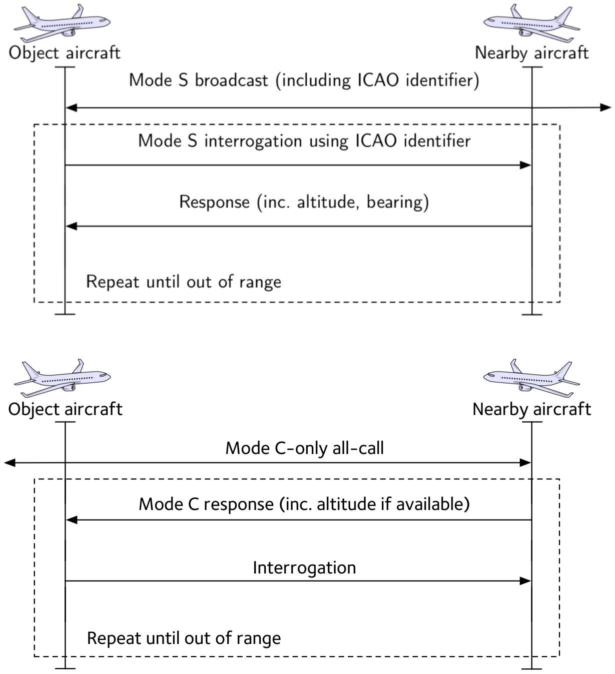

Object aircraft
Nearby aircraft
Mode S broadcast (including ICAO identifier)
Mode S interrogation using ICAO identifier
Response (inc. altitude, bearing)
Repeat until out of range
Object aircraft
Nearby aircraft
Mode C-only all-call
Mode C response (inc. altitude if available)
Interrogation
Repeat until out of range

Fig. 3. Representation of TCAS interrogation protocols of nearby aircraft using Mode C and S transponders.

amber and an aural alert of ‘traffic’ is given. If the intruder becomes closer to the aircraft, a *resolution advisory* (RA) is given. An RA will contain specific instructions for the flight crew, i.e., to climb or descend at a given rate, or hold vertical speed. These instructions are decided between the two aircraft automatically and aim to deconflict the situation. Crew must follow the instructions of an RA within seconds.

In the cockpit, crew have some control over the sensitivity level; they can select *standby*, *TA-ONLY*, or *TA/RA*. For most of a flight, TCAS will be set to TA/RA, which automatically calculates sensitivity based on altitude. TA-ONLY is limited to the lowest sensitivity level and does not issue RAs, whereas standby performs no TCAS interrogations and will not resolve conflicts [5].

Whilst in TA/RA, TCAS will calculate the sensitivity based on altitude, with higher altitudes assigned higher sensitivities. This then defines the *tau* value for issuing a TA or RA. Tau is calculated as the time in seconds to the Closest Point of Approach (CPA) between object and nearby aircraft, either laterally or vertically. When the nearby aircraft is within tau, the relevant alert is given.[1] For example, between 5000 and 10,000 ft, tau for a TA is 40 s [5].

III. Importance of Separation

Adequate separation is a crucial component of effective and safe airspace with TCAS being in place to protect it. ICAO define vertical separation minima in Doc 4444, namely [8]:

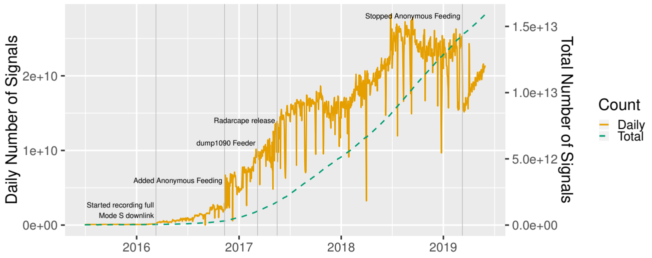


Fig. 4. The growth of OpenSky’s dataset over time from June 2013 to May 2019

- Under Vertical Separation Minimum (VSM), 1000 ft below 29,000 ft or 2000 ft above,
- In Reduced Vertical Separation Minima (RVSM) airspace in an RVSM-approved aircraft, separation above 29,000 ft is 1000 ft depending on conditions.

These are used as a basis by regional ATC in defining their requirements for VSM. Horizontal separation is a more complex definition which depends not only on the horizontal distance between aircraft but also on the vertical separation, type of navigation being used and whether the aircraft is climbing or descending [9].

Whilst losses in separation are not consistently penalized, they are treated as serious due to a potential ‘snowball effect’ if not corrected. A lack of separation allows for significantly smaller—or in some cases no—margin for error. In some cases these will be treated as an ‘airprox’, which requires a report on the incident to be submitted to a regional board.

Aside from the most serious consequence of separation loss, the mid-air collision, a number of other potential consequences can arise:

- Flight through wake vortex from other aircraft, causing extreme turbulence or loss of control,
- Causing other aircraft to take avoidance action, triggering airspace inefficiency,
- Requirement for extreme avoidance manoeuvres at short notice, risking injury to passengers or crew [10].

IV. The OpenSky Network

The OpenSky Network is a crowdsourced sensor network collecting air traffic control (ATC) data. Its objective is to make real-world ATC data accessible to the public and to support the development and improvement of ATC technologies and processes. Since 2012, it continuously collects air traffic surveillance data. Unlike commercial flight tracking networks (e.g., Flightradar24 or FlightAware), the OpenSky Network keeps the raw Mode S replies as they are received by the sensors in a large historical database which can be accessed by researchers and analysts from different areas.

The network started with eight sensors in Switzerland and Germany and has grown to more than 2000 receivers at locations all around the world. As of this writing, OpenSky’s dataset contains six years of ATC communication data. While the network initially focused on ADS-B only, it extended its

[1] Some adjustments are made to this at lower altitudes, and are covered in detail in [5].

data range to the full Mode S downlink channel in March 2017, which is also the base for this present work. The dataset currently contains more than 15 trillion Mode S replies and receives more than 20 billion messages per day. Fig. 4 shows the growth and development over the past several years with milestones highlighted, including the support of the dump1090 and Radarcape feeding solutions and the integration of non-registered, anonymous receivers, which has recently been discontinued. Besides the payload of each Mode S downlink transmission, OpenSky stores additional metadata. Depending on the receiver hardware, this metadata includes precise timestamps (suitable for multilateration), receiver location, and signal strength. For more information on OpenSky's history, architecture and use cases refer to [11], [12] or visit http://opensky-network.org.

V. Data Collection

We decoded the Mode S replies using the latest version of OpenSky's open-source decoding framework libadsb[2]. Since OpenSky collects downlink transmissions only, the respective uplink interrogations containing the requested Ground-Initiated Comm B (GICB) register numbers are missing. Therefore, we have updated our decoding library with routines for detecting the registers that are relevant to analyse TCAS/ACAS advisories, mainly BDS 1,0 for equipage and capability information and BDS 3,0 for active resolution advisories.

The data set considered in this work is a snapshot of the unmodified data ("raw data") that came into OpenSky between May 10, 2019 and May 23, 2019. During this two-week period, almost 1000 sensors from over 90 countries reported around 250 billion Mode S signal receptions by 126,700 different aircraft to the network. Based on the reported altitude, we found around 44.6% of these aircraft to be capable of flying in Class A airspace, thus assuming that these flights operated under instrument flight rules (IFR). Aircraft that were only seen below flight level 180 are assumed to operate under visual flight rules (VFR). Fig. 5 shows the distribution of all replies across the different reply types and by IFR/VFR aircraft. VFR aircraft seem to be responsible for only a negligible fraction of the communication happening on the 1090 MHz frequency. This is not surprising since many rely on FLARM in Europe and UAT in the US and Mode S ground interrogators have very limited range on lower altitudes.

A. TCAS RA Detection

In order to find cases of active threat resolutions for our analysis, we searched the raw data for aircraft transmitting BDS 3,0 registers. Over the two weeks period, we found 147 situations where the TCAS units exchanged active resolution advisories. Besides the aircraft's own transponder ID and altitude, long ACAS replies containing BDS 3,0 registers also provide information such as the threat's transponder ID (or its range, bearing and altitude), whether there are multiple threats, and detailed information on the issued RA itself. The latter contains flags such as whether the RA is corrective or preventive, the "sense" of the RA (downward or upward), whether is constitutes a sense reversal, and others. Also complements to the RA such as "no pass below/above" or "no turn left/right" are included. In 106 of the 147 cases, the aircraft also transmitted ADS-B which enabled us to further investigate the spacial situation and behaviour before and after the issuance of the RA (see Sec. VII).

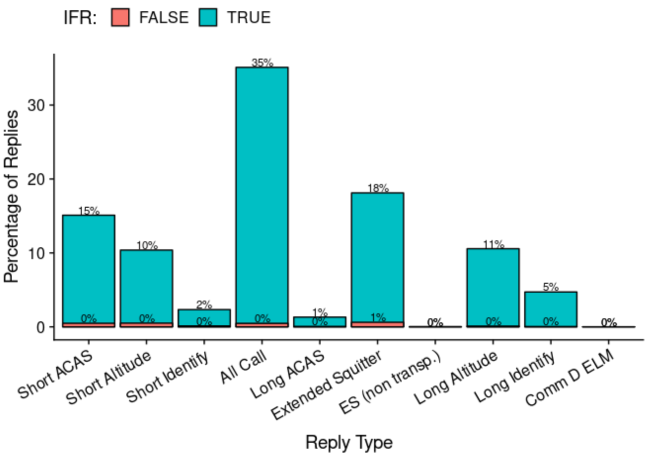


Fig. 5. Distribution of 249,310,882,072 Mode S replies collected by OpenSky during a two-week period 10-23 May 2019.

B. Aircraft Metadata

We have used OpenSky's own aircraft database to identify the metadata about an aircraft based on the received unique ICAO 24-bit identifiers. The aircraft database currently consists of 498,910 airframes (May 30, 2019), including about 1,500 different commercial airlines and many additional non-airline operators. The database has initially been built from many available online and offline sources, which are discussed in detail in the OpenSky Report 2017 [13]. It is now updated daily from several authoritative sources and also integrates crowdsourced information as it is curated by the wider OpenSky community. The full database is available for download and online use at https://opensky-network.org/aircraft-database.

C. Limitations

Despite collecting all analyzed data to the best of our possibilities, there are some natural limitations to the datasets used in our OpenSky reports. The most natural limit of our data is OpenSky's coverage. The OpenSky Network currently only fully covers the European continent (at least in the en-route airspace), while America, East Asia, Australia and New Zealand are covered partly. Our analysis explicitly does not cover or represent the situation in the non-covered airspaces.

Moreover, since receiving Mode S and ADS-B signals requires a line of sight between receiver and aircraft, the ranges of receivers are limited by the radio horizon. For

[2] https://github.com/openskynetwork/java-adsb

example, if the aircraft is in the en-route airspace, i.e. at a high altitude, and the receiver is not obstructed by the geographical environment (e.g., in coastal areas), the radio horizon and thus the range can be up to 700 km. Aircraft at lower altitudes, however, remain difficult to track due to their reduced line of sight. As a consequence, lower altitudes are only covered if there is a sensor nearby and aircraft trajectories may be incomplete in many areas.

Another important limitation is the data quality. ADS-B is still in its deployment phase and there are no guarantees that transponders are functioning according to the specification. In fact, a small number of transponders broadcast erroneous or invalid positions, or wrong ICAO 24-bit addresses. Furthermore, most OpenSky receivers are not certified. Due to missing implementations of proper tracking techniques, erroneous messages can pass the error detection mechanism of Mode S and therefore end up in our data. Although we have a multitude of plausibility checks to filter most of these invalid data, a small amount may still remain in the data used for this work. Nevertheless, based on our experience from working with Mode S and ADS-B for many years, we are confident that the portion of erroneous data is negligible compared to the overall size of the dataset and that the numbers provided in this work are accurate estimates of the TCAS situation within OpenSky's coverage area.

VI. TCAS Statistics

A. *Aircraft Equipage with TCAS*

There are several possibilities to get information on TCAS equipage on the Mode S downlink. One way is to decode the ADS-B Operational Status reports of an aircraft, which include information such as ADS-B version, TCAS availability, ADS-B availability, use of multiple antennae or position accuracy information. This information, however, is only broadcast by an aircraft if the transponder supports it, i.e., only ADS-B version 1 and 2 transponders. Overall, we found that only about 43% of the aircraft in the data set reported their operational status.[3]. The other 57% used either ADS-B version 0 (11%) or no ADS-B at all (47%). Note that these numbers cover VFR flights as well as flights in countries with no ADS-B mandate. Because of the limited nature of the operational status reports, it is not meaningful for a broader analysis of TCAS. We thus propose a more accurate method to estimate TCAS equipage by analyzing the data set for the number of aircraft that were actually seen replying to TCAS interrogations. Overall, we observed replies from 89.47% of all transponder-equipped aircraft using this method, with minor differences between IFR (91.04%) and VFR (88.21%).

To get more details about the equipage, we also extracted BDS 1,0 GCIB registers from the 2-weeks Mode S data set. Among other things, BDS 1,0 provides information about the version of the TCAS transponder. Note that this method has limitations as well, since it requires the transponder to support Comm B data link transmissions and it requires a nearby interrogator requesting this specific BDS register. Nevertheless, we found information on 26,100 transponders. The distribution of the indicated TCAS versions of these transponders is shown in Fig. 6.

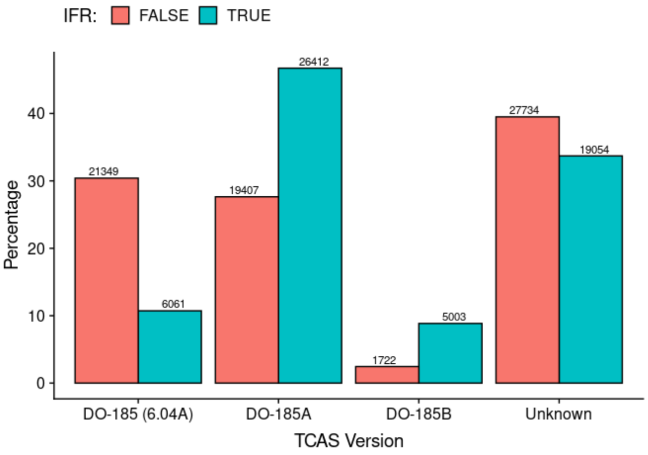


Fig. 6. Distribution of the TCAS versions indicated by 26,100 transponders in the respective BDS 1,0 GCIB transmissions.

TABLE I
Distribution of TCAS RAs by aircraft type (aircraft types with more than three occurrences).

A319	A320	A321	B737	B738	B739	B752
11	12	13	19	29	11	5
CRJ2	CRJ7	CRJ9	E75L			
17	5	10	21			

B. *TCAS Usage Statistics*

TCAS RAs by Aircraft Type: Table I shows the distribution of TCAS RAs by aircraft type. We have seen the most RAs for the B738 family, followed by the E75l and the B737. Naturally, these are absolute numbers, which must be seen in context, e.g. miles flown by these aircraft families within the OpenSky coverage.

TCAS RAs distribution by country: During the considered two-week period, TCAS RAs were collected for pairs of aircraft flying where OpenSky offers a coverage. 70 alerts were decoded over the United States, 15 over various Europe countries, 1 over Australia, 2 over Malaysia and 1 over Russia.

These figures have to be considered with caution. They should probably be normalised by a measure of traffic density and overall coverage of the considered region.

TCAS RAs distribution by altitude: Figure 7 plots the distribution of the altitudes where TCAS RAs occurred in the dataset and Figure 8 relates these occurrences to major neighbouring airports. Two major peaks in the distribution occur for altitudes very close to the ground, less than 3 nm from a major airport (see Section VII-A about parallel landings), around 10,000ft (see Section VII-B regarding intersections between traffic taking off and landing in neighbouring airports)

[3] An analysis of these status reports can be found in the OpenSky Report 2016 [14]

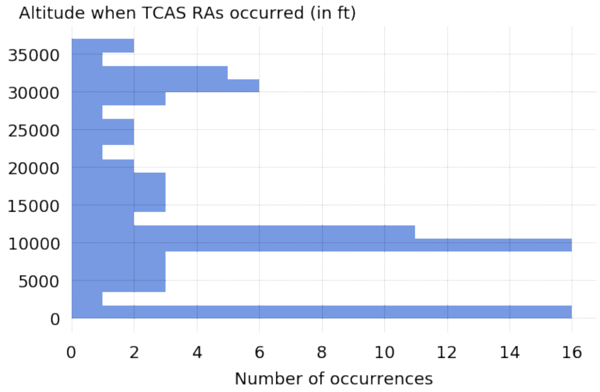


Fig. 7. Distribution of altitudes where TCAS RAs occurred.

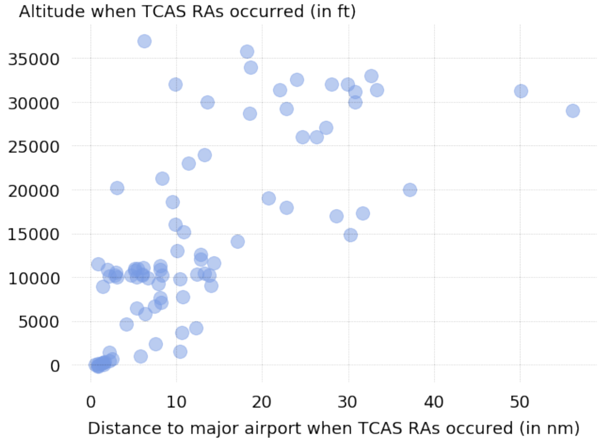


Fig. 8. Distribution of altitudes where TCAS RAs occurred vs. distance to major airport

with few occurrences around 30,000ft, near top of climbs and/or beginning of descents (see Section VII-C regarding the intersection between en-route traffic and climbing/descending aircraft).

VII. Case Studies

Our analysis has highlighted some examples of TCAS RAs which warrant further discussion. In this section we look at parallel approaches, at the intersection between standard arrival (STAR) and departure (SID) procedures, and near the top of climb/beginning of descent of trajectories.

A. Parallel Approaches

One situation in which TCAS appears to raise alarms under normal conditions is during parallel approaches. This is somewhat expected due to the relatively close proximity of aircraft at similar phases of parallel approaches and is usually safe considering that they are under close ATC management at this point.

In the case shown in Fig. 9 and 10, this is likely to have been triggered due to both horizontal and vertical proximity, coupled with the fact that the aircraft were at a sufficient altitude to use a higher sensitivity level. It appears that one aircraft was issued with a descend RA to increase separation. In this instance, whilst TCAS judged the situation to be a risk, it was not. According to METAR information at that time at Frankfurt airport `EDDF 111020Z 36009KT 9999 -RA FEW004 BKN009 11/09 Q1009 BECMG BKN010=`, moderate wind came from the North, which most probably lead to a heading of the aircraft not aligned with the runway (crab approach), possibly leading to an interpolation raising an alert.

In Fig. 11 and 12 we see another example of a TCAS RA on parallel approach. In this case, the RA is triggered as the

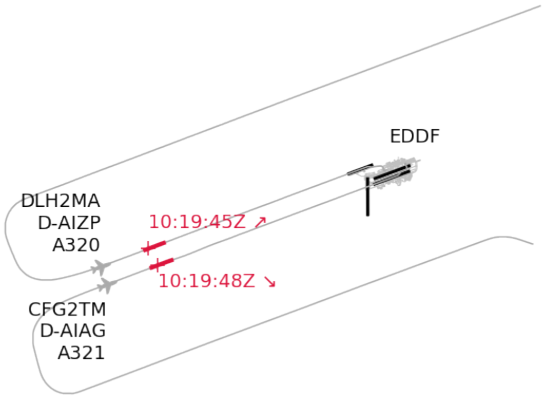


Fig. 9. Flight paths of DLH2MA and CFG2TM into Frankfurt Airport (EDDF) on parallel approaches. Parts of path marked in red indicate distance covered during a TCAS RA.

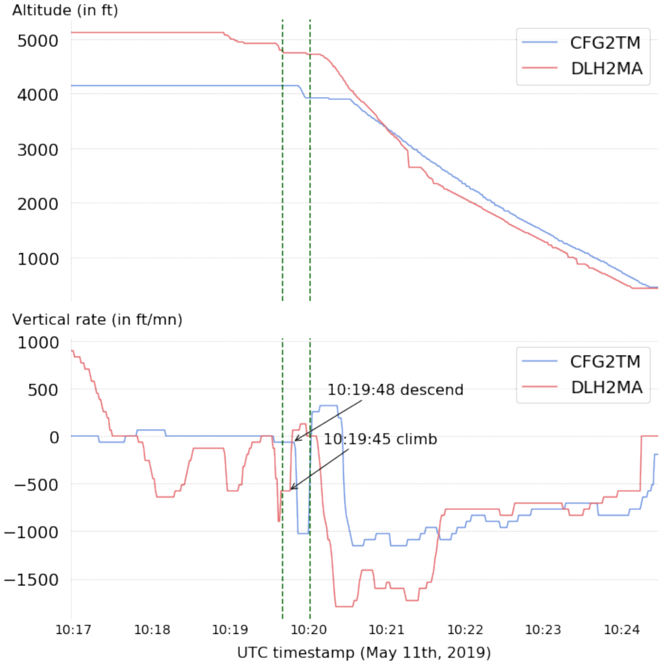


Fig. 10. Altitudes of DLH2MA and CFG2TM along with horizontal and vertical separation between the aircraft during the time period surrounding the RA on approach to Frankfurt Airport (EDDF). RA time period is denoted by green vertical lines.

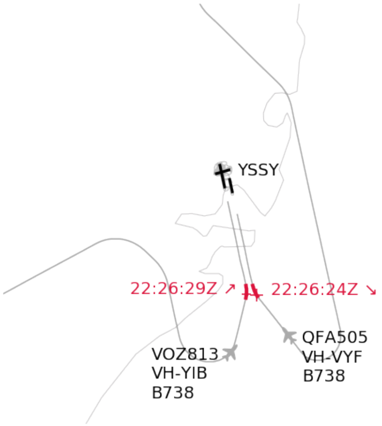


Fig. 11. Flight paths of VOZ813 and QFA505 into Sydney Airport (YSSY) on parallel approaches. Parts of path marked in red indicate distance covered during a TCAS RA.

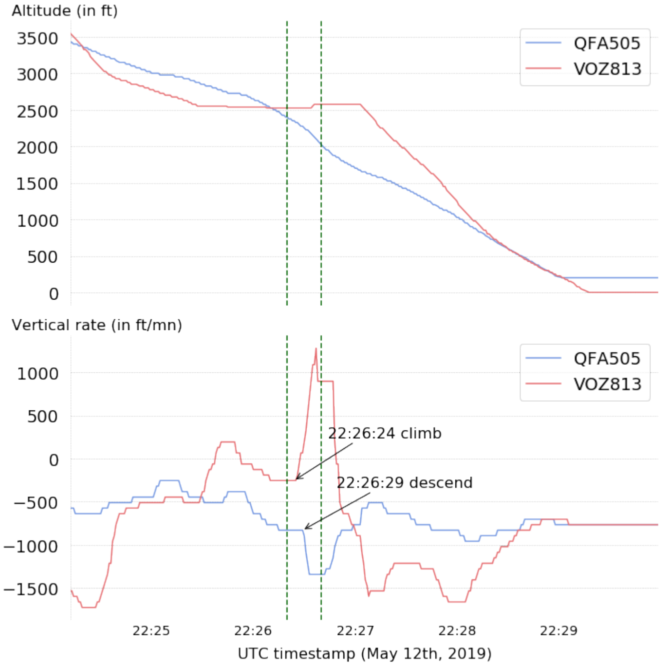


Fig. 12. Altitudes of VOZ813 and QFA505 along with horizontal and vertical separation between the aircraft during the time period surrounding the RA on approach to Sydney Airport (YSSY). RA time period is denoted by green vertical lines.

aircraft turn onto the localizers of their respective runways causing one aircraft to be issued with a maintain vertical speed (descend) RA and the other to maintain vertical speed (level) RA. Here, TCAS will have anticipated that the aircraft would have been continuing their localizer intercept paths, hence on course for collision. As with the previous case, the situation is safe and closely managed by ATC, but this RA is not spurious since the system accounts for intended future behaviour.

Unusual and anomalous final approaches have been addressed in [15] from a safety risk assessment point of view. In this specific situation, QFA505 reached 2500ft in order to catch the glide path before landing. When the RA was triggered, they had to climb again, and came above the glide path. After the RA was resolved, the vertical rate came down to -1500ft/min and the aircraft caught the glide path from above around 750ft above ground. Losing that amount of total energy in the last miles before the runway threshold can be uncomfortable for the pilot and is a common cause of failed approach. It is also worth noting that final approaches and low altitudes are among the most common scenarios where RAs are not followed by the crew, leading to more detailed safety studies [16].

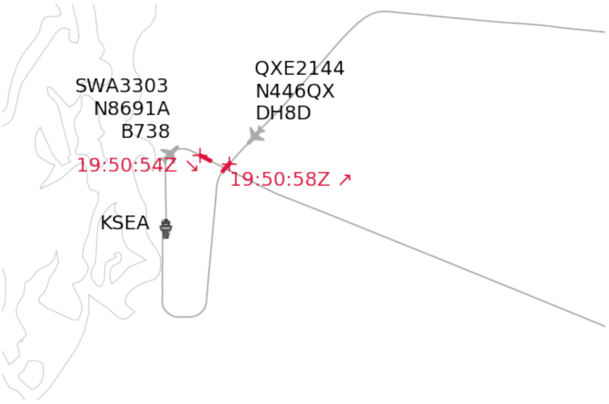


Fig. 13. Flight paths of QXE2144 and SWA3303 around Seattle–Tacoma Airport (KSEA) on intersecting departure and arrival trajectories. Parts of path marked in red indicate distance covered during a TCAS RA.

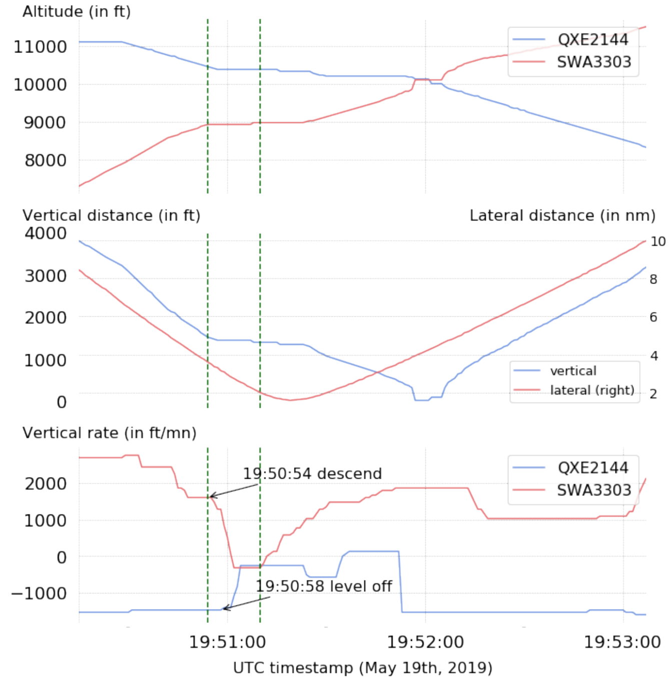


Fig. 14. Altitudes of QXE2144 and SWA3303 along with horizontal and vertical separation between the aircraft during the time period surrounding the RA on departure from/arrival to Seattle–Tacoma Airport (KSEA). RA time period is denoted by green vertical lines.

B. STAR/SID Conflict

A situation which gives rise to low-altitude RAs is the crossing of departure and arrival patterns. As with parallel approaches, these occur in regions under close ATC monitoring but here, the potential for harm is higher. This is due to the aircraft in these situations having opposite intended trajectories rather than ultimately travelling on similar horizontal and vertical trajectories as with parallel approaches.

In Fig. 13 and 14, we can see QXE2144 and SWA3303 near Seattle–Tacoma Airport. With SWA3303 climbing and QXE2144 descending, both aircraft received ‘level off’ RAs to maintain vertical separation until they had better horizontal separation. In this situation, the aircraft would have passed very close to each other without TCAS intervention. Notably, the RA occurs prior to the horizontal crossing.

Similarly, Fig. 15 and 16 show RPA3725 on departure from and AAL2436 on arrival to Dallas–Fort–Worth Airport. Here, the RA occurs during the horizontal crossing. As with the previous example, TCAS issues ‘level off’ RAs here due to the intended vertical and horizontal crossing of the aircraft. These were important to follow as otherwise the aircraft would have lost a considerable amount of separation.

Monitoring the regularity of cases such as these could be useful in identifying regularly conflicting departure and arrival paths, which could be adjusted to reduce the chance of conflicts in the future. Narrow 1000ft separations between STAR and SID procedures at the point they cross is a common cause of false alerts on ATC and TCAS systems, which lead some airports to adapt their procedures with a 2000ft separation between these paths.

C. Top of climb and/or beginning of descent

In Fig. 17 and 18 FIN7HL and AZA1491 cross their paths above Italy. A possible loss of separation seems to have been anticipated by the local ATC[4] who gave clearance to FIN7HL to descend to FL310 and to AZA1491 to climb to FL300, ensuring a conflict-free situation. Climbing and descending rates were interpolated by TCAS systems, yielding RAs to prevent a possible loss of separation. Indeed, TCAS in its current configuration is neither aware of ATC clearances nor does it take into account the altitude setting in the MCP. Recommendations have been issued in the European ATM Master Plan in order to take this setting into account in future ACAS systems.

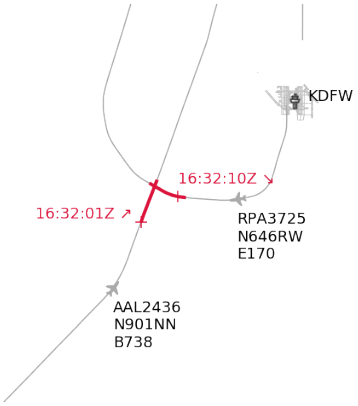


Fig. 15. Flight paths of RPA3725 and AAL2436 around Dallas–Fort–Worth Airport (KDFW) on intersecting departure and arrival trajectories. Parts of path marked in red indicate distance covered during a TCAS RA.

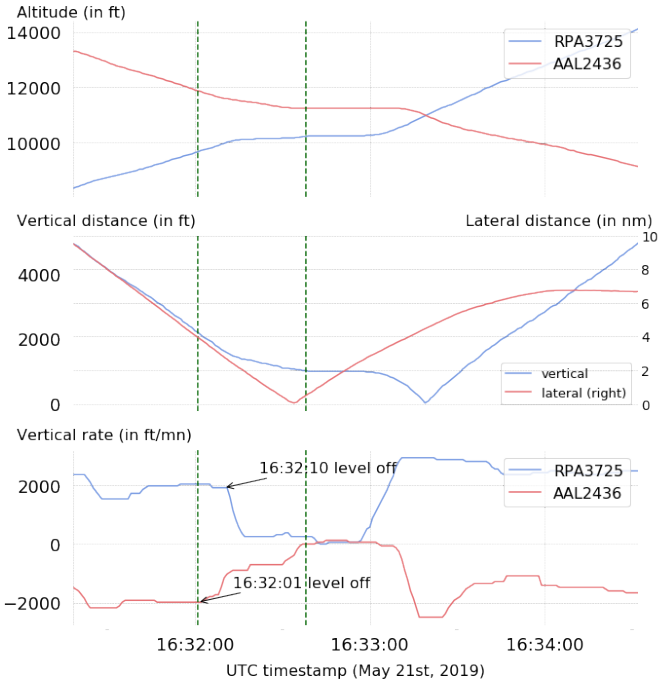


Fig. 16. Altitudes of RPA3725 and AAL2436 along with horizontal and vertical separation between the aircraft during the time period surrounding the RA on departure from/arrival to Dallas–Fort–Worth Airport (KDFW). RA time period is denoted by green vertical lines.

VIII. Discussion

Collision avoidance systems are a crucial cornerstone of managing modern air traffic and have helped to improve safety in increasingly busy airspaces. However, not much independent research has been done by the scientific community on the inner workings and the efficacy of the system. Recently, Aireon, providers of the first space-based ADS-B receiver system have conducted some preliminary analysis of TCAS data collected with their global satellite constellation [17]. The work shows that it is possible to receive TCAS RAs in space and analyze losses of separation using Aireon’s receiver system. While this proof of concept shows that satellite receivers can be a helpful supporting system, in particular in Oceanic airspace and other non-surveillance regions, the received data is not freely available for independent researchers to work on.

In contrast, besides the presented analysis of typical RA situations and a first look at wider statistics surrounding TCAS, the present work aims to facilitate future research in the area of collision avoidance. With the decoder open

[4] MCP altitude setting is available as part of the Mode S Comm-B standard, in the BDS 4,0 fields. The availability of such messages depends on the ACC and on the configuration of local secondary radars. The usage of such messages is rather consistent across Europe.

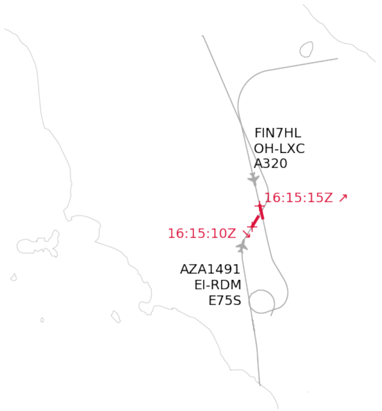


Fig. 17. Flight paths of FIN7HL and AZA1491 above Italy. Parts of path marked in red indicate distance covered during a TCAS RA.

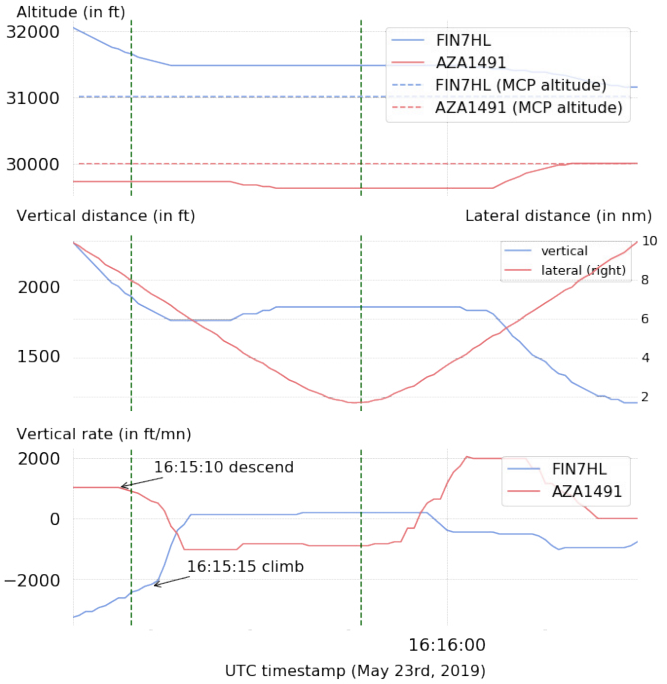


Fig. 18. Altitudes of FIN7HL and AZA1491 along with horizontal and vertical separation between the aircraft during the time period surrounding the RA. RA time period is denoted by green vertical lines. MCP altitude (corresponding to ATC clearance) is denoted by horizontal dashed lines.

sourced and the collected historical TCAS data freely available to researchers through the OpenSky Network, we hope that many others will look more deeply into this crucial and safety-critical area in the future.

IX. Conclusion

In this paper, we have analyzed the current usage characteristics of TCAS, the Traffic Alert and Collision Avoidance System, by using global data from the crowdsourced research network OpenSky. We have gathered statistical data and anecdotal case studies, which provide insights into the use of TCAS worldwide. We have developed an open source decoder for TCAS messages to conduct this research and enable other interested parties to gather their own data. Based on this decoder, the OpenSky Network also offers existing historical data going back to 2016, which can facilitate more detailed TCAS research in the future even for researchers without their own collection sites.

References

[1] T. Williamson and N. A. Spencer, "Development and operation of the traffic alert and collision avoidance system (tcas)," *Proceedings of the IEEE*, vol. 77, no. 11, pp. 1735–1744, 1989.

[2] M. Strohmeier, A. K. Niedbala, M. Schäfer, V. Lenders, and I. Martinovic, "Surveying aviation professionals on the security of the air traffic control system," in *Security and Safety Interplay of Intelligent Software Systems*. Springer, 2018, pp. 135–152.

[3] A. Pratap and J. McIntyre, "Officials say hundreds feared killed in airline collision over India," Online, Nov. 1996, accessed 2019-05-08. [Online]. Available: https://bit.ly/2NrB3zn

[4] International Civil Aviation Organization, *Annex 10 to the Convention on International Civil Aviation—Surveillance and Collision Avoidance System*, 4th ed., Jul. 2007, vol. 4.

[5] Federal Aviation Administration, *Introduction to TCAS II Version 7.1*. U.S. Department of Transport, 2011.

[6] S. Henely, *Digital Avionics Handbook*, 3rd ed., C. R. Spitzer, U. Ferrell, and T. Ferrell, Eds. CRC Press, 2015.

[7] Eurocontrol, "Flying without a transponder—10 minutes is all it can take," *NetAlert*, no. 19, p. 5, May 2014. [Online]. Available: https://www.eurocontrol.int/sites/default/files/publication/files/NetAlert-19.pdf

[8] International Civil Aviation Organization, *Procedures for Air Navigation Services—Air Traffic Management*, 16th ed., 2016, ch. 5, p. 5.2.

[9] ——, *Procedures for Air Navigation Services—Air Traffic Management*, 16th ed., 2016, ch. 5, pp. 5.4–5.39.

[10] SKYbrary, Online, Aug. 2017, accessed 2019-05-29. [Online]. Available: https://www.skybrary.aero/index.php/Loss_of_Separation

[11] M. Schäfer, M. Strohmeier, V. Lenders, I. Martinovic, and M. Wilhelm, "Bringing Up OpenSky: A Large-scale ADS-B Sensor Network for Research," in *Proceedings of the 13th IEEE/ACM International Symposium on Information Processing in Sensor Networks (IPSN)*, Apr. 2014.

[12] M. Strohmeier, M. Schäfer, M. Fuchs, V. Lenders, and I. Martinovic, "Opensky: A swiss army knife for air traffic security research," in *Proceedings of the 34th IEEE/AIAA Digital Avionics Systems Conference (DASC)*, Sep. 2015.

[13] M. Schäfer, M. Strohmeier, M. Smith, M. Fuchs, V. Lenders, M. Liechti, and I. Martinovic, "OpenSky report 2017: Mode S and ADS-B usage of military and other state aircraft," in *2017 IEEE/AIAA 36th Digital Avionics Systems Conference (DASC)*. IEEE, 2017, pp. 1–10.

[14] M. Schäfer, M. Strohmeier, M. Smith, M. Fuchs, R. Pinheiro, V. Lenders, and I. Martinovic, "OpenSky report 2016: Facts and figures on SSR mode S and ADS-B usage," in *2016 IEEE/AIAA 35th Digital Avionics Systems Conference (DASC)*. IEEE, 2016, pp. 1–9.

[15] X. Olive and P. Bieber, "Quantitative Assessments of Runway Excursion Precursors using Mode S data," in *Proceedings of the International Conference for Research in Air Transportation*, 2018. [Online]. Available: https://arxiv.org/abs/1903.11964

[16] Eurocontrol, "Operational safety study: TCAS RA not followed," Tech. Rep., 2017. [Online]. Available: https://www.eurocontrol.int/publications/operational-safety-study-tcas-ra-not-followed

[17] A. Hoag, J. Dolan, and M. Garcia, "Identifying Collision Avoidance Resolution Advisories and Anomalies in Aircraft Avionics globally with space-based ADS-B Data Observations," in *International Symposium on Enhanced Solutions for Aircraft and Vehicle Surveillance Applications*, 2018.

2019 11th International Conference on Cyber Conflict
Silent Battle
T. Minárik, S.Alatalu, S.Biondi,
M.Signoretti, I.Tolga, G.Visky (Eds.)
2019 © NATO CCD COE Publications, Tallinn

Machine Learning-based Detection of C&C Channels with a Focus on the Locked Shields Cyber Defense Exercise

Nicolas Känzig
Department of Information Technology
and Electrical Engineering
ETH Zürich
Zürich, Switzerland
kaenzign@student.ethz.ch

Roland Meier
Department of Information Technology
and Electrical Engineering
ETH Zürich
Zürich, Switzerland
meierrol@ethz.ch

Luca Gambazzi
Science and Technology
armasuisse
Thun, Switzerland
luca.gambazzi@armasuisse.ch

Vincent Lenders
Science and Technology
armasuisse
Thun, Switzerland
vincent.lenders@armasuisse.ch

Laurent Vanbever
Department of Information Technology
and Electrical Engineering
ETH Zürich
Zürich, Switzerland
lvanbever@ethz.ch

Abstract: The diversity of applications and devices in enterprise networks combined with large traffic volumes make it inherently challenging to quickly identify malicious traffic. When incidents occur, emergency response teams often lose precious time in reverse-engineering the network topology and configuration before they can focus on malicious activities and digital forensics.

In this paper, we present a system that quickly and reliably identifies Command and Control (C&C) channels without prior network knowledge. The key idea is to train a classifier using network traffic from attacks that happened in the past and use it to identify C&C connections in the current traffic of other networks. Specifically, we leverage the fact that – while benign traffic differs – malicious traffic bears similarities across networks (e.g., devices participating in a botnet act in a similar manner irrespective of their location).

To ensure performance and scalability, we use a random forest classifier based on a set of computationally-efficient features tailored to the detection of C&C traffic. In order to prevent attackers from outwitting our classifier, we tune the model parameters to maximize robustness. We measure high resilience against possible attacks – e.g., attempts to camouflaging C&C flows as benign traffic – and packet loss during the inference.

We have implemented our approach and we show its practicality on a real use case: Locked Shields, the world's largest cyber defense exercise. In Locked Shields, defenders have limited resources to protect a large, heterogeneous network against unknown attacks. Using recorded datasets (from 2017 and 2018) from a participating team, we show that our classifier is able to identify C&C channels with 99% precision and over 90% recall in near real time and with realistic resource requirements. If the team had used our system in 2018, it would have discovered 10 out of 12 C&C servers in the first hours of the exercise.

Keywords: *malware, botnets, machine learning, digital forensics, Locked Shields, network defense*

1. INTRODUCTION

Large enterprise or campus networks handle data from a vast set of different applications, protocols, and devices. Identifying malicious traffic in such networks is similar to the figurative problem of finding a needle in a haystack, raising the need for effective tools to automate this process and to support defenders such as computer emergency response teams (CERTs) in their operation. As network traffic is not only voluminous but also very diverse, these tools need to adapt to different contexts.

Recent alarming examples of malicious software exploiting a remote infrastructure in order to issue directives to steal or modify data or performing distributed denial-of-service attacks include CryptoLocker [1] or the Mirai botnet [2].

Machine learning-based models have repeatedly been proven to outperform humans in tasks involving large data volumes and high-dimensional feature spaces. However, training these models to detect malicious activity in networks is a particularly challenging task, because the methods used by modern threat actors are continuously evolving. Moreover, the profiles of legitimate background traffic can vary strongly among different networks and their users. Consequently, such solutions might perform well in the environment they have been trained in, while failing in new deployments.

In this paper, we focus on one particular type of malicious traffic: communication between compromised hosts and their Command and Control (C&C) servers. C&C traffic only depends on the botnet (i.e. the communication scheme between the C&C server and the bots) and is invariant to the networks to which the bots are connected. This makes the development of machine learning-based models that perform reliably in different contexts more feasible. We argue that identifying this type of traffic is fruitful because it means that compromised hosts can be identified (and eventually blocked, isolated or patched) before an actual attack is launched.

The work that we present in this paper is based on data from Locked Shields [3], the world's largest cyber defense exercise. While Locked Shields is only an exercise, it reproduces critical infrastructure under the intense pressure of severe cyberattacks. Moreover, it provides a setting that closely matches the real world: in practice, defenders have limited resources to protect a large, heterogeneous network against unknown attacks. And because it is an exercise, we obtained a ground-truth of logs from the attackers describing when and where they were active, something which is hardly possible for real incidents.

Problem statement: Given the constraints (e.g. in terms of computational resources and lack of familiarity with the network) that defending teams face during the Locked Shields exercise, we aim to design a system that can identify C&C traffic and compromised hosts.

Challenges: Solving this problem is challenging for the following reasons:

- Benign and malicious traffic profiles can vary considerably between different Locked Shields exercises.
  This requires a solution with high generalization and robustness.
- Defenders have a very limited budget for computational resources.
  This requires an efficient classification technique.

- Defenders have a small amount of storage capacity.
  This prevents them from storing large amounts of network traffic.
- Defenders have a small bandwidth to access the attacked network.
  This makes it impossible to send large amounts of data to an external system.

Our approach: Our key idea is to use data from past iterations of Locked Shields to efficiently identify similar-looking C&C traffic in future exercises. We do this by creating a labeled dataset containing flow-based features extracted from raw Locked Shields traffic captures, which we then use to train a supervised classifier (random forest) to flag C&C traffic. Our approach is efficient enough to be deployed during future Locked Shields exercises.

Novelty and related work: Detecting C&C traffic has been the focus of many research papers in recent years (cf. surveys in [4] [5]), many of which also pursue classifier-based approaches using machine learning algorithms. [6] proposes a two-stage system for identifying P2P C&C traffic using a decision tree and a random forest classifier. To train a random forest classifier, [7] leverages the fact that malware-related domains are likely to have an inconsistent pool of requesting hosts. [8] develops a system for classifying malicious C&C servers using NetFlow data, extracting features related to flow sizes, client access patterns and temporal behavior.

In contrast to these approaches, we use a new set of flow-based features and evaluate our models on two new and completely labeled datasets (Locked Shields 2017 and 2018). While most studies train and evaluate their models on different parts of the same dataset, we use train- and test-sets that have been acquired independently in different setups. This provides strong evidence for the ability of our system to perform in new environments. Moreover, a minority of the solutions proposed in past investigations claim to run in real time [4]. In our approach, we combine quickly computable features (e.g. number of packets per flow) with an efficient random forest algorithm, which makes real-time calculation feasible.

Contributions: The main contributions of this paper are:

- A selection of features that allow identifying C&C channels while being fast and efficient to compute.
- An efficient random forest model that classifies between C&C traffic and normal traffic with high accuracy.
- An implementation of the system that is suitable for deployment in future Locked Shields exercises.

- An evaluation based on real data from Locked Shields 2017 and 2018, which shows that our system allows defenders to identify C&C traffic, C&C servers and compromised hosts.

Organization: The remainder of this paper is organized as follows. In Section 2, we provide background information on the Locked Shields exercise and define the attacker model. In Section 3, we present our system to identify C&C traffic before we evaluate it in Section 4. In Section 5, we discuss the outcome and finally, we conclude in Section 6.

2. BACKGROUND ON LOCKED SHIELDS

In this section, we explain how Locked Shields is organized and give details about the roles of defenders and attackers.

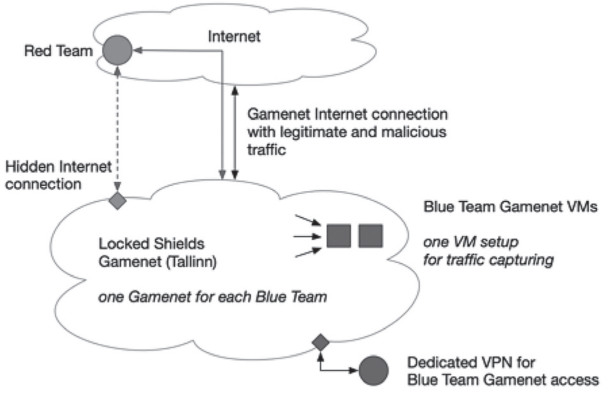
A. Exercise Organization

Locked Shields is the largest and most complex live-fire global cyber defense exercise, with more than 1000 participating cyber experts from 30 nations [9]. It takes place every year and is organized by the NATO Cooperative Cyber Defence Centre of Excellence (CCDCOE) in Tallinn (Estonia) [3].

For the exercise, participating countries send *Blue Teams*, which represent response teams whose main task is to secure and protect the network infrastructure. Whereas each Blue Team operates in an isolated instance of the network (*Gamenet*), a *Red Team* runs attacks against all these networks in order to compromise or degrade the performance of the connected systems.

In Figure 1, we illustrate the environment during an exercise.

FIGURE 1. LOCKED SHIELDS ENVIRONMENT OVERVIEW.


Red Team
Internet
Gamenet Internet connection with legitimate and malicious traffic
Hidden Internet connection
Blue Team Gamenet VMs
one VM setup for traffic capturing
Locked Shields Gamenet (Tallinn)
one Gamenet for each Blue Team
Dedicated VPN for Blue Team Gamenet access

The environment simulated during the exercise changes every year. In this paper, we focus on the last two occurrences of Locked Shields (2017 and 2018). In 2017, the Blue Teams had to maintain the services and networks of a military air base; in 2018, a major civilian Internet service provider, a military base and other critical infrastructures of a fictional country were targeted in cyber attacks.

B. Environment and Constraints for the Defenders

Prior to the exercise, the defenders (Blue Teams) receive an architecture scheme of the original Gamenet that shows the topology and connected devices. However, the scheme does not show changes put in place by the Red Team (e.g. additional connections between the Gamenet and the Internet to bypass the main gateways).

In addition, each Blue Team obtains two virtual machines (VMs) inside the Gamenet, which it can use during the exercise to install its own tools (e.g. to perform forensics or deploy patches). Moreover, the traffic exchanged in the Gamenet is forwarded to one VM in order to allow the Blue Team to perform on-site analysis and detection. However, the performance of this VM is limited and access to it is only possible via a low-bandwidth VPN tunnel. In order to rapidly counter Red Team activity, the Blue Team has to deploy efficient analysis tools (given the constraints on computation and bandwidth), intrusion detection systems, and to avoid sending voluminous data to an external infrastructure. The system that we present in this paper is designed to work in such a restricted environment.

After the exercise, the Red Team delivers reports to the Blue Teams summarizing their malicious activities.

C. Attacker Model

The attackers (Red Team) perform their activities according to a tight schedule of missions and goals. Waves of attacks hit the Blue Team for the entire duration of the exercise. Some attacks are limited to a specific phase of the exercise while others are repeated during the entire exercise.

Prior to the exercise, the Red Team knows the configuration of the entire Gamenet and can use this knowledge to prepare suitable attacks (e.g. leveraging outdated systems).

In order to systematically orchestrate the large number of attacks on all Gamenets, the Red Team uses Cobalt Strike as a C&C framework. This allows automatizing injections, deployment of malicious code and C&C datalink management.

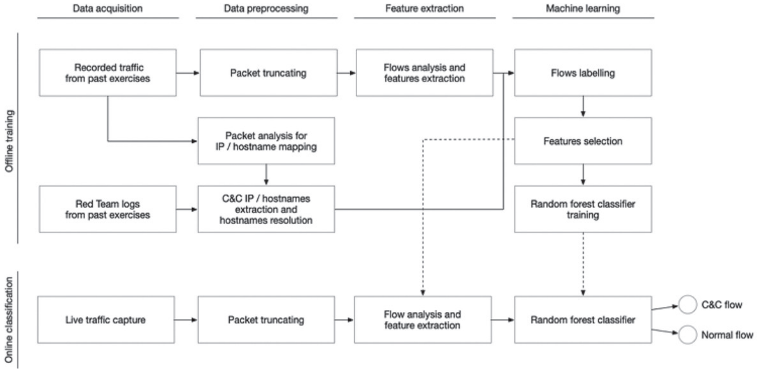
3. SUPERVISED MACHINE LEARNING FOR DETECTING C&C CHANNELS

In this section, we explain how we use supervised machine learning to identify C&C channels in the Locked Shields exercise. First, we provide an overview of our approach. Afterwards, we describe the data and labeling that we used. Finally, we explain how we selected the features and the machine learning model for this task.

A. Overview

Our system consists of two basic phases (Figure 2): offline training and online classification. In the offline training phase (which was done prior to the exercise), we used data from past Locked Shields exercises and processed them in order to obtain a labeled dataset to train a supervised classifier that could be used for live classification of C&C flows during the exercise.

FIGURE 2. SYSTEM OVERVIEW.


Data acquisition
Data preprocessing
Feature extraction
Machine learning
Offline training
Recorded traffic from past exercises
Packet truncating
Flows analysis and features extraction
Flows labelling
Packet analysis for IP / hostname mapping
Features selection
Red Team logs from past exercises
C&C IP / hostnames extraction and hostnames resolution
Random forest classifier training
Online classification
Live traffic capture
Packet truncating
Flow analysis and feature extraction
Random forest classifier
C&C flow
Normal flow

B. Data Analysis and Enrichment

In this section, we describe the data sources we used for labeling and training and the preprocessing steps we applied.

1) Available Data Sources

We built our labeled dataset from two sources: raw traffic captures and Red Team logs.

a) Raw Traffic Captures

We obtained pcap traffic traces containing the Gamenet activity recorded during Locked Shields 2017 and 2018 (LS17, LS18) from a participating country (Switzerland). The packets are not sampled, anonymized, or truncated. We extracted the features used to train our models from this data.

b) Red Team Logs

The activities of the Red Team are logged in different documents generated by the Cobalt Strike framework [10]. Among others, these documents contain *indicators of compromise* (e.g. IP addresses and domain names of C&C servers) and an *activity report*, which contains a timeline of all Red Team activities (e.g. commands that were executed on compromised machines). We used these log files to label the C&C flows.

2) Data Preprocessing

Before extracting features, we preprocessed the dataset in three ways: *(i)* we truncated packets to reduce the size of the dataset; *(ii)* we aggregated packets to flows; and *(iii)* we mapped domain names and IP addresses in the traffic capture.

a) Truncating Packets

Since capturing and analyzing full packets in real time is difficult for the Blue Team during the exercise, our approach does not require packet payloads. We used only the first 96 bytes (enough to capture everything up to the header of the transport layer) of each packet, which reduced the size of our dataset by approximately 75 percent. The performance of our final models did not decrease due to the truncation.

b) Flow Extraction

We aggregated the packets from the packet trace into flows, since our model operates at the flow level. A flow is defined by its 5-tuple (source IP, destination IP, source port, destination port, transport layer protocol). It starts with a TCP SYN packet and ends when the first TCP FIN packet is sent or after a timeout of 15s. We used CICFlowMeter [11] to extract flow-based features from the raw traffic traces.

c) Domain Name Resolution

The Red Team logs list some devices only by their domain name, thus we needed a mapping from these domain names to the associated IP addresses. We used Bro [12], a network analysis framework, to resolve the domain names to IP addresses from the packet traces, using information contained in the HTTP (host header), TLS (with server name indication), or DNS.

C. Data Labeling

After extracting a list of IP addresses and domain names of C&C servers from the Red Team logs, the labeling process was straightforward: we labeled all flows where at least one endpoint was a C&C server (i.e. listed in the Red Team logs) as malicious and all other flows as benign. The intuition behind this approach was that there was no benign reason for any device to contact a C&C server. It is safe to assume that any device communicating with a C&C server is compromised.

D. Feature Selection and Extraction

In this section, we explain how we selected and extracted the features that our classifier would use to identify C&C flows.

1) Feature Extraction

For computing the features, we used CICFlowMeter [13] (version 3.0), an open source tool for extracting flows from packet traces and computing large sets of features. CICFlowMeter focuses on time-related features such as the inter-arrival time of packets, active and idle times separately for packets in each direction, while including minimum, maximum, mean and standard deviation [11]. These features are suitable for our purposes because they can be extracted with little computational effort.

To capture the fact that C&C servers are typically located outside the internal network, we added an additional feature (Int/Ext Dst IP), indicating whether the destination IP address of a flow is within the internal address space.

Table I lists all features that we considered in our selection process.

TABLE I: COMPLETE LIST OF FEATURES CONSIDERED IN THE FEATURE SELECTION.
ONE ROW CAN DESCRIBE MULTIPLE FEATURES (E.G. THE MINIMUM, MAXIMUM, MEAN AND STANDARD DEVIATION OF A PROPERTY)

Nr	Feature	Description
1	Flow Duration	Duration of the flow in microseconds
2-3	Tot Fwd/Bwd Pkts	Total packets in the fwd/bwd direction
4-5	TotLen Fwd/Bwd Pkts	Total size of packets in fwd/bwd direction
6-13	Fwd/Bwd Pkt Len	Min, Max, Mean, Std size of packet in fwd/bwd direction
14-23	Fwd/Bwd IAT	Total, Min, Max, Mean, Std time between two packets sent in the fwd/bwd direction
24-35	Flag Counts	Flag Counts PSH, URG, SYN, FIN, RST, ACK, URG, CWE, ECE in Fwd/Bwd and both directions. (0 for UDP)
36-37	Fwd/Bwd Header Len	Total bytes used for headers in the fwd/bwd direction
38-40	Fwd/Bwd/Tot Pkts/s	Number of fwd/bwd/tot packets per second
41	Flow Byts/s	Number of flow bytes per second
42-45	Pkt Len	Min, Max, Mean, Std packet length of a flow
46-49	Flow IAT	Min, Max, Mean, Std packet inter-arrival time in fwd/bwd direction
50	Down/Up Ratio	Download and upload ratio
51	Pkt Size Avg	Average size of packet
52-53	Fwd/Bwd Seg Size Avg	Average size observed in the fwd/bwd direction
54-55	Fwd/Bwd Byts/b Avg	Average number of bytes bulk rate in the fwd/bwd direction
56-57	Fwd/Bwd Pkts/b Avg	Average number of packets bulk rate in the fwd/bwd direction
58-59	Fwd/Bwd Blk Rate Avg	Average number of bulk rate in the forward direction
60-61	Subflow Fwd/Bwd Pkts	Average number of packets in a subflow in the fwd/bwd direction
62-63	Subflow Fwd/Bwd Byts	Average number of bytes in a subflow in the fwd direction
64-67	Active Time	Min, Max, Mean, Std time a flow was active before becoming idle
68-71	Idle Time	Min, Max, Mean, Std time a flow was idle before becoming active
72-73	Init Fwd/Bwd Win Byts	TCP window size in the fwd/bwd direction
74	Fwd Act Data Pkts	Count of fwd packets with at least 1 byte of TCP payload
75	Fwd Seg Size Min	Minimum segment size in the forward direction
76	Int/Ext Dst IP	0 if Dst-IP of a flow belongs to Blue Teams subnet, 1 if external
77	L3/L4 Protocol	0 for TCP, 1 for UDP, 2 for ICMP

2) Feature Selection

To identify the best set of features, we removed correlating and irrelevant features by applying a recursive feature elimination scheme based on random forest Gini importance scores [14].

In each iteration, we trained a random forest classifier with the dataset from LS17 and all the considered features. Afterwards, we removed the feature with the lowest score from the set of considered features. Thus, we obtained a feature ranking, where the one that is first removed has the lowest rank. Eliminating features one by one is crucial, as importance scores can spread over multiple features with redundant information (i.e. if multiple important features are strongly correlated, their scores can all be low in a particular iteration).

The 20 most important features according to our feature selection are listed below in descending order of importance (except for the last two features, which we included in the feature set based on preliminary evaluations).

Tot Fwd Pkts, Flow IAT Mean, Fwd IAT Max, Flow Pkts/s, Bwd Pkt Len Min, FIN Flag Cnt, Init Fwd Win Byts, Active Mean, Bwd IAT Mean, Bwd Pkt Len Std, Fwd Seg Size Min, Fwd Pkt Len Std, Tot Bwd Pkts, Bwd Header Len, Subflow Fwd Byts, Subflow Bwd Pkts, Fwd IAT Tot, Flow IAT Max, Int/Ext Dst IP, L3/L4 Protocol

E. Model Selection

We tested a variety of different supervised models on our data: Artificial Neural Network, Support Vector Machine, Logistic Regression, Naive Bayes, K-Nearest Neighbors and Random Forest (RF). The main difficulty in our task was that the distribution of the background traffic was different in the LS17 and LS18 data, as benign and attack traffic profiles change every year. However, the distribution of the C&C session features hardly varies, due to the fact that the same tool (Cobalt Strike) is used to maintain these sessions. We found that RF performed best under these circumstances. Furthermore, RF models are highly efficient and require low training and inference times, which is decisive for real-time deployments.

1) Model Configuration

As a baseline model, we used an RF classifier with default configurations from scikit-learn [15] (i.e. an ensemble of 10 fully expanded trees). However, this resulted in large trees (30,000 nodes for the model trained on LS17, 70,000 for LS18) and we found that constraining the maximal tree-depth significantly increased the robustness of our model. We empirically found that a maximum tree-depth of 10 drastically reduced the node count (to 700 for LS17 and 900 for LS18). However, reducing the depth further had a negative impact on the performance. Moreover, we found that increasing the number of trees to 128 further improved the robustness and prediction quality with negligible impact on computational cost. In the following, we refer to configurations with a maximum depth of 10 and 128 trees as "tuned" configurations.

2) Robustness Against Camouflage

In the following, we analyze possible attack vectors against our model, assuming a white-box scenario where the attacker has full knowledge of the model and the features we deploy. We focus on two strategies that the attacker can follow: modifying Cobalt Strike's C&C configuration, and altering the C&C flows by other means (e.g. by changing the network stack on the infected machines).

a) Changing the appearance of the C&C sessions using Cobalt Strike

As our model detects C&C sessions maintained using Cobalt Strike, we first analyze the options this framework provides to alter their appearance. The two main parameters the Red Team can use during the exercise are the sleep-period and jitter of a C&C session. The sleep-period defines the time interval used to periodically contact the C&C server. The jitter configures the deviation from this periodicity. Our features are invariant to both of these parameters, as they focus on timing statistics within single connections and do not depend on the time elapsed between the periodic connections of a C&C session.

Cobalt Strike’s Malleable C2 tool [16] allows the custom design of the HTTP headers of the packets exchanged within C&C sessions to avoid detection. However, our model does not rely on features extracted from HTTP headers.

We conclude that bypassing detection of our model by altering Cobalt Strike’s C&C configurations is infeasible as our features are invariant to the options the framework provides.

b) Identifying attack vectors for manipulating feature values
Our classifier identifies flows that look like Cobalt Strike C&C channels. To avoid this, an attacker might attempt to camouflage these C&C flows as normal traffic for the given network.

We observe that most of the feature values can be altered either by injecting additional packets (to manipulate statistics such as inter-arrival time or packet counts) or by altering the packet sizes (which affects features such as the download size). Many of these tampering attempts could be prevented by additional checks in the feature extraction phase (e.g. sequence number checking for packet injections). However, since this is computationally expensive, we assume that the defenders cannot do this.

To simulate the robustness of our model in such scenarios, we conducted experiments involving tampering with the feature values, as described in Section 4.D.

4. EVALUATION

In this section, we evaluate our classifiers based on data recorded by the Swiss Blue Team from Locked Shields 2017 and 2018. After providing more details about the methodology (Subsection A), we evaluate precision and recall (Subsection B), runtime (Subsection C), robustness against camouflaging (Subsection D) and incomplete traffic captures (Subsection E).

A. Methodology

In this section, we summarize the datasets that we used for the evaluation, the environment in which we conducted the experiments and the parameters that we used.

1) Datasets

To evaluate the performance of our models, we used the complete LS17 dataset for training and the LS18 dataset for testing and vice versa. Therefore, our evaluation corresponds to a case where our classifier is used for classifying previously unseen data in a different network. In the following, we will refer to models trained on the full LS17 or LS18 datasets as LS17-models and LS18-models, respectively (cf. Table III).

In Table II, we summarize the baseline information about the datasets that we used for the evaluation.

TABLE II: BASELINE INFORMATION ABOUT THE DATASETS USED.

Dataset	Size	Packets	Flows	C&C Flows
LS17	114 GB	288'940'662	9'070'828	1'239'041 (13.7%)
LS18	216 GB	557'783'930	16'379'346	1'818'006 (11.1%)

2) Environment

We conducted all experiments and calculations on a virtual machine running Ubuntu 16.04 (64 bit), with 10 Intel Xeon E5-2699 cores and 16 GB RAM. The implementation was based on Python 3.6 and scikit-learn (0.19.2) [17].

3) Parameters and Models

We evaluated two configurations of our classifier: one with the default scikit-learn parameters [15], and the other with the tuned parameters described in Section 3.E. We refer to these configurations as “baseline” and “tuned” and summarize them in Table III. We trained all models using the 20 features obtained from the recursive feature elimination scheme described in Section 3.D.

TABLE III: CHARACTERIZATION OF MODELS USED IN OUR EVALUATION.

Model	Training data	Testing data	RF size	RF depth
LS17-baseline	LS17	LS18	10 trees	unconstrained
LS17-tuned	LS17	LS18	128 trees	10
LS18-baseline	LS18	LS17	10 trees	unconstrained
LS18-tuned	LS18	LS17	128 trees	10

B. Precision/Recall

We used widespread metrics precision (i.e., the percentage of reported C&C flows that are actual C&C flows) and recall (i.e., the ratio between the correctly identified C&C flows and all the C&C flows present in the dataset) to measure the prediction quality of our models. High precision is particularly important in the given task because a high number of false positives would mislead the defenders during their operation.

Table IV lists the precision and recall scores for all models. We repeated the evaluation ten times with different random seeds to train the models, and we report the medians of the results. The results show that all models achieve high precision and recall while tuned configuration clearly outperforms the baseline configuration.

TABLE IV: THE TUNED MODELS ACHIEVE HIGH PRECISION AND RECALL (MEDIANS)

Model	Precision	Recall
LS17-baseline	0.94	0.98
LS17-tuned	0.99	0.98
LS18-baseline	0.98	0.86
LS18-tuned	0.99	0.90

C. Runtime

In this experiment, we evaluate the runtime of three phases:

1. Extracting features from the training dataset
2. Training the model
3. Applying the model on the testing dataset

In Table V, we report the time it takes to extract features from both datasets (using CICFlowMeter). We note that the feature extraction tool extracts all 77 features from Table I. The runtime could be significantly improved by calculating only the 20 selected features and by using a more efficient implementation.

TABLE V: FEATURE EXTRACTION TAKES LESS THAN 45 MIN (LS17) AND LESS THAN 90 MIN (LS18) FOR DATASETS CONTAINING ABOUT 38 HOURS OF NETWORK TRAFFIC.

Dataset	Runtime	Extracted Flows
LS17	42 min	9'070'828
LS18	85 min	16'379'346

In Table VI, we report the time it takes to train and test the model on both datasets. As above, we point out that the training phase is not time-critical as it is done prior to the exercise. As the results show, running predictions on the whole dataset takes less than one minute. In a practical deployment, the inference would be performed on much smaller sets of samples, which makes real-time detection feasible.

TABLE VI: TRAINING AND INFERENCE TIMES OF THE BASELINE AND TUNED MODEL CONFIGURATIONS

Model	Training time	Inference time
LS17-baseline	120 s	6 s
LS17-tuned	1117 s	50 s
LS18-baseline	390 s	4 s
LS18-tuned	2828 s	30 s

D. Robustness Against Camouflaging

In this experiment, we simulate an attacker attempting to camouflage C&C flows as normal traffic. To model an attack against a particular feature, we replace the feature values in the malicious samples (i.e. the C&C flows) with values randomly subsampled from benign samples. As a result, this feature no longer helps in distinguishing C&C flows from normal flows.

In Figure 3 (LS17) and Figure 4 (LS18), we plot the precision and recall of the respective models depending on the number of tampered features. The results hold under the assumption that an attacker that attacks n features would target the n most relevant features according to Section 3.D. (which is a promising strategy). We evaluate the impact of tampering with 5 to 14 features on each model with 10 different random seeds and plot the median as well as the 95% confidence interval.

The results show that the tuned model reacts much less sensitive to camouflaging attempts and achieves high performance even if many features are tampered with (precision falls below 90% when manipulating >12 features). Recall of the LS18 model drops sharply when attacking more than 5 features, however, its precision remains high, meaning that the predictions the model makes are still reliable. Further, we observe that the variance among the tuned models is much lower than that of the baseline models.

FIGURE 3. ACHIEVED PRECISION AND RECALL FOR LS17 IF AN ATTACKER TRIES TO CAMOUFLAGE C&C FLOWS. OUR TUNED MODEL IS ROBUST AGAINST TAMPERING, FOR UP TO 10 FEATURES.

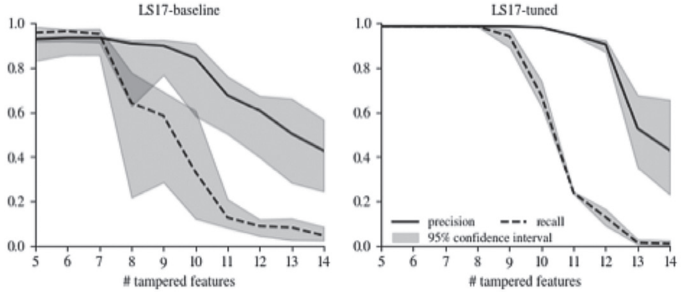
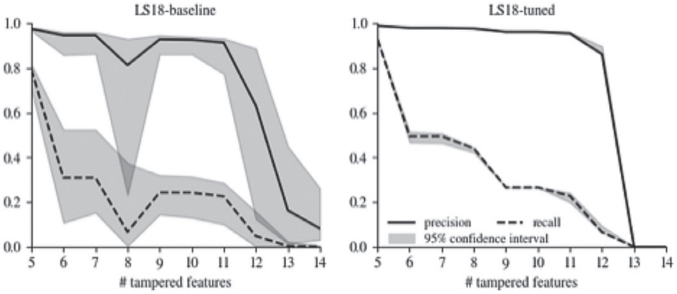


FIGURE 4. ACHIEVED PRECISION AND RECALL FOR LS18 IF AN ATTACKER TRIES TO CAMOUFLAGE C&C FLOWS. OUR TUNED MODEL ACHIEVES A HIGH PRECISION EVEN IF 12 FEATURES ARE ATTACKED BUT THE RECALL DROPS.

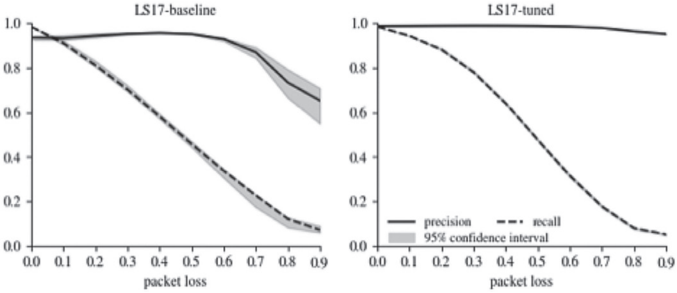


E. Robustness Against Packet Loss

In this experiment, we evaluate the impact of packet loss, which could occur due to the limited resources of the defenders to capture packets in real time during the exercise. We simulate this by randomly dropping between 10 and 90 percent of the packets.

The results in Figure 5 show that the tuned models achieve high precision (> 95%) even for 90% packet loss. This means that even for high losses, the raised alerts stay accurate. However, the recall decreases approximately linearly with the packet loss. Presumably, this is because C&C flows with too many dropped packets are no longer recognized as such, while the model still detects less affected flows.

FIGURE 5. IMPACT OF PACKET LOSS ON THE LS17-MODEL. THE CURVES SHOW THE MEAN VALUES OVER 10 MEASUREMENTS. IN OUR TUNED MODEL, PACKET LOSS HARDLY IMPACTS PRECISION.



5. DISCUSSION

In this section, we discuss the outcomes of the experiments conducted in this paper as well as details of possible real-world deployments and potential extensions.

A. Identifying C&C Servers

The ability to detect individual C&C flows can obviously be used to identify C&C servers (the destinations of such flows) and compromised hosts (the sources of the flows). In an additional experiment, we observed that running our system for a short time period of 30 minutes at the beginning of the exercise (11am-12 pm in Locked Shields 2018) was enough to identify most of the C&C servers (10 out of 12 listed in the Cobalt Strike reports). We further observed 5 different source IP addresses from the Blue Team's network communicating with these servers, suggesting that these hosts had been compromised at this point in time.

B. Running Multiple Models in Parallel

In this paper, we used datasets from two occurrences of Locked Shields: one to train the model, and the other to test it. In the future, when more datasets are available, we suggest training multiple models and conducting live classification during the exercise on all of them. This would make it even harder for the Red Team to camouflage C&C traffic as benign flows, because it needs to match the features of benign flows in multiple different models (while the features of C&C flows are similar in each model). Performing the inference only slightly increases the computational cost and is thus feasible during the exercise. Since we have data from only two iterations of Locked Shields, we could not evaluate this approach.

C. Practical Deployment for Future Locked Shields Exercises

In order to use our system in the next Locked Shields exercise, a Blue Team needs to perform three steps:

1. Train one or multiple models with labeled data from past exercises
2. Prepare the VM to record network traffic and compute the features
3. Run the trained models with the recorded features during the exercise

Step 1 is not time-critical and can be done at any time prior to the exercise. To counteract camouflaging attempts by the Red Team, we suggest using data from different years and training multiple models (cf. Section 5.B).

For Step 2, the Blue Team can use any tool to capture the traffic (no payloads required) and calculate the flow features. In our experiments, we used CICFlowMeter; however, more efficient implementations are possible.

Step 3 consists of feeding the extracted features to one or more models. Information about detected C&C flows can be passed to an intrusion alert system used by the defenders to coordinate security responses.

As our evaluation shows, our classifier is able to predict C&C flows with 99% precision and over 90% recall. By evaluating the system on two datasets originating from two different occurrences of Locked Shields (2017 and 2018), we provided strong evidence for the success of a deployment in future exercises on previously unseen data.

In an additional experiment, we simulated a real-case deployment, where we applied our system for a short 30 minutes time interval in the first phase of the LS18 exercise. There, our system unveiled almost the complete C&C infrastructure used by the Red Team (10 out of 12 C&C Servers).

D. Challenges and Deployment in Other Environments

In this paper, we have focused on a very specific use case for C&C detection (Locked Shields, Cobalt Strike). One of the main limitations of supervised-learning-based systems is that while they are highly effective in detecting anomalies that were labeled in the training set, they fail to detect new and unknown attacks. A further challenge is that the distribution of the legitimate background traffic may strongly vary among different networks.

By expanding the training data with more C&C traffic types and including a wider range of legitimate traffic profiles, our approach could be adapted for deployment in other environments. Moreover, data augmentation techniques such as domain randomization – currently applied with great success in the deep learning domain – are other promising paths towards broader generalization. For instance, OpenAI recently developed a human-like robotic hand to manipulate physical objects with unprecedented dexterity [18]. The training was performed solely in a simulated environment, but by randomizing the physical properties in the simulation, the final model generalized well enough to be deployed on a real physical hand. Although our application is very different, the same concepts could be applied to network traffic data to obtain richer training sets leading to more robust detection systems.

6. CONCLUSION

In this paper, we present a system for identifying C&C channels using supervised machine learning. As a typical use case for such a system, we focus on Locked Shields, the world's largest cyber defense exercise. Our evaluation shows that the system could be deployed by defenders in this exercise and that it identifies C&C traffic with high precision and recall. We use real data from one participating Blue Team and show that if this team had trained the classifier with the data from 2017, it would have identified C&C channels in Locked Shields 2018 with 99% precision and 98% recall. Further, running the system during a time interval of just 30 minutes in LS18 would have been enough to identify 10 out of 12 C&C servers used by the Red Team.

Acknowledgments

We thank the Swiss Blue Team for sharing their data and expertise with us and for their constant support throughout this project.

REFERENCES

[1] "How a British SMB survived a nightmarish cryptolocker ransom attack | Security | Computerworld UK," [Online]. Available: https://www.computerworlduk.com/security/how-british-smb-survived-nightmarish-cryptolocker-ransom-attack-3677593/.

[2] M. Antonakakis, T. April, M. Bailey, M. Bernhard, E. Bursztein, J. Cochran, Z. Durumeric, J. A. Halderman, L. Invernizzi, M. Kallitsis and others, "Understanding the mirai botnet," in *USENIX Security Symposium*, 2017.

[3] "Locked Shields 2017," [Online]. Available: https://ccdcoe.org/ locked-shields-2017.html.

[4] A. H. Lashkari, G. D. Gil, J. E. Keenan, K. Mbah and A. A. Ghorbani, "A Survey Leading to a New Evaluation Framework for Network-based Botnet Detection," in *Proceedings of the 2017 the 7th International Conference on Communication and Network Security*, 2017.

[5] M. Feily, A. Shahrestani and S. Ramadass, "A survey of botnet and botnet detection," in *Third International Conference on Emerging Security Information, Systems and Technologies, 2009. SECURWARE'09*.

[6] B. Rahbarinia, R. Perdisci, A. Lanzi and K. Li, "Peerrush: Mining for unwanted p2p traffic," in *International Conference on Detection of Intrusions and Malware, and Vulnerability Assessment*, 2013.

[7] M. Antonakakis, R. Perdisci, W. Lee, N. Vasiloglou and D. Dagon, "Detecting Malware Domains at the Upper DNS Hierarchy," in *USENIX security symposium*, 2011.

[8] L. Bilge, D. Balzarotti, W. Robertson, E. Kirda and C. Kruegel, "Disclosure: detecting botnet command and control servers through large-scale netflow analysis," in *Proceedings of the 28th Annual Computer Security Applications Conference*, 2012.

[9] "CCDCOE News (26 April 2018)," [Online]. Available: https://ccdcoe.org/more-1000-cyber-experts-30-nations-took-part-locked-shields.html.

[10] "Cobalt Strike Reporting," [Online]. Available: https://www.cobaltstrike.com/help-reporting.

[11] G. Draper-Gil, A. H. Lashkari, M. S. I. Mamun and A. A. Ghorbani, "Characterization of Encrypted and VPN Traffic using Time-related," in *Proceedings of the 2nd international conference on information systems security and privacy (ICISSP)*, 2016.

[12] "The Bro Network Security Monitor," [Online]. Available: https://www.bro.org/.

[13] "CICFLOWMETER A network traffic Biflow generator and analyzer," [Online]. Available: http://www.netflowmeter.ca/.

[14] G. Louppe, L. Wehenkel, A. Sutera and P. Geurts, "Understanding variable importances in forests of randomized trees," in *Advances in neural information processing systems*, 2013.

[15] "sklearn RandomForestClassifier," [Online]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html.

[16] "Cobalt Strike 3.11 Manual," [Online]. Available: https://www.cobaltstrike.com/downloads/csmanual311.pdf.

[17] "scikit-learn Machine Learning in Python," [Online]. Available: https://scikit-learn.org.

[18] OpenAI, "Learning dexterous in-hand manipulation," 2018.

5 Other publications from the Cyber-Defence Campus in 2019

Below we provide a list of the scientific papers published by the Cyber-Defence Campus in 2019.

- Crowdsourced Wireless Spectrum Anomaly Detection, Sreeraj Rajendran, Vincent Lenders, Wannes Meert, and Sofie Pollin, *IEEE Transactions on Cognitive Communications and Networking (TCCN)*, 2019.

  Abstract - Detecting anomalous behavior in wireless spectrum is a demanding task due to the sheer complexity of the electromagnetic spectrum use. Wireless spectrum anomalies can take a wide range of forms from the presence of an unwanted signal in a licensed band to the absence of an expected signal, which makes manual labeling of anomalies difficult and suboptimal. We pre-sent, spectrum anomaly detector with interpretable features (SAIFE), an adversarial autoencoder (AAE)-based anomaly detector for wireless spectrum anomaly detection using power spectral den-sity (PSD) data. This model achieves an average anomaly detection accuracy above 80% at a con-stant false alarm rate of 1% along with anomaly localization in an unsupervised setting. In addition, we investigate the model's capabilities to learn interpretable features, such as signal bandwidth, class, and center frequency in a semi-supervised fashion. Along with anomaly detection the model exhibits promising results for lossy PSD data compression up to 120× × × and semi-supervised signal classification accuracy close to 100% on three datasets just using 20% labeled samples. Finally, the model is tested on data from one of the distributed electrosense sensors over a long term of 500 h showing its anomaly detection capabilities.

- (Self) Driving Under the Influence: Intoxicating Adversarial Network Inputs, Roland Meier, Thomas Holterbach, Stephan Keck, Matthias Stähli, Vincent Lenders, Ankit Singla, and Laurent Vanbever, ACM Workshop on Hot Topics in Networks (HotNets), Princeton, New Jersey, USA, November 2019.

  Abstract - Traditional network control planes can be slow and require manual tinkering from operators to change their behavior. There is thus great interest in a faster, data-driven approach that uses signals from real-time traffic instead. However, the promise of fast and automatic reac-tion to data comes with new risks: malicious inputs designed towards negative outcomes for the network, service providers, users, and operators. Adversarial inputs are a well-recognized problem in other areas;we show that networking applications are susceptible to them too. We characterize the attack surface of data-driven networks and examine how attackers with different privileges—from infected hosts to operator-level access—may target network infrastructure, applications, and protocols. To illustrate the problem, we present case studies with concrete attacks on recently proposed data-driven systems. Our analysis urgently calls for a careful study of attacks and de-fenses in data-driven networking, with a view towards ensuring that their promise is not marred by oversights in robust design.

- Higher than a Kite: ADS-B Communication Analysis Using a High-Altitude Balloon, Matthias Schaefer, Roberto Calvo-Palomino, Franco Minucci, Brecht Reynders, Gérôme Bovet, and Vincent Lenders, OpenSky Workshop (OSN), Zurich, Switzerland, November 2019.

  Abstract - Receiving signals on the 1090 MHz frequency, one of the most important radio frequencies used in aviation, is typically done using ground-based receivers. However, an increasing number of airborne or even space-based receivers also aim to receive these signals for appli-

cations such as air traffic surveillance and collision avoidance. In this paper, we present our results from a high-altitude radio frequency measurement campaign with the goal to gain insights about the challenges and limitations of receiving 1090 MHz signals at high altitudes. We used a high-altitude balloon equipped with a software-defined radio to collect 1090 MHz signal data. In an extensive analysis of these data, we identify several challenges and provide a first impression of the radio environment at altitudes up to 33.5 km.

- Jamming/Garbling Assessment and Possible Mitigations in the OpenSky Network, Mauro Leonardi, Martin Strohmeier, and Vincent Lenders, OpenSky Workshop (OSN), Zurich, Switzerland, November 2019.

  Abstract - The Automatic Dependent Surveillance-Broadcast (ADS-B) technology is one of the pillars of the future surveillance system for air traffic control. However, its many fundamental vulnerabilities are well known and an active area of research. This paper examines two closely related ADS-B radio frequency channel issues, jamming and garbling. Both jamming and garbling produce the same physical effect: the reception of mixed signals, coming from different sources (usually not co-located). In this paper, we assess the impact of these reception problems and examine three separate mitigation techniques. Through the use of theoretical evaluations, simulations and real-world analysis based on data collected by the OpenSky Network, we compare their effectiveness and establish a first baseline for their use in modern low-cost, crowdsourced ADS-B networks.

- 28 Blinks Later: Tackling Practical Challenges of Eye Movement Biometrics, Simon Eberz, Giulio Lovisotto, Kasper Rasmussen, Vincent Lenders and Ivan Martinovic, ACM Conference on Computer and Communications Security (CCS), London, United Kingdom, November 2019.

  Abstract – In this work we address three overlooked practical challenges of continuous authentication systems based on eye movement biometrics: (i) changes in lighting conditions, (ii) task dependent features and the (iii) need for an accurate calibration phase. We collect eye movement data from 22 participants. To measure the effect of the three challenges, we collect data while varying the experimental conditions: users perform four different tasks, lighting conditions change over the course of the session and we collect data related to both accurate (user-specific) and inaccurate (generic) calibrations. To address changing lighting conditions, we identify the two main sources of light, i.e., screen brightness and ambient light, and we propose a pupil diameter correction mechanism based on these. We find that such mechanism can accurately adjust for the pupil shrinking or expanding in relation to the varying amount of light reaching the eye. To account for inaccurate calibrations, we augment the previously known feature set with new features based on binocular tracking, where the left and the right eye are tracked separately. We show that these features can be extremely distinctive even when using a generic calibration. We further apply a cross-task mapping function based on population data which systematically accounts for the dependency of features to tasks (e.g., reading a text and browsing a website lead to different eye movement dynamics).

  Using these enhancements, even while relaxing assumptions about the experimental conditions, we show that our system achieves significantly lower error rates compared to previous work. For intra task authentication, without user-specific calibration and invariable screen brightness and ambient lighting, we achieve an equal error rateof3.93%withonlytwominutesoftraining-data.Forthesame setup but with constant screen brightness (e.g., as for a reading task) we can achieve equal error rates as low as of 1.88%.

- Event Detection on Microposts: a Comparison of Four Approaches, Akansha Bhardwaj, Albert Blarer, Philippe Cudré-Mauroux, Vincent Lenders, Boris Motik, Axel Tanner, and Alberto Tonon, *IEEE Transactions on Knowledge and Data Engineering (TKDE)*, October 2019.

  Abstract - Microblogging services such as Twitter are important, up-to-date, and live sources of information on a multitude of topics and events. An increasing number of systems use such services to detect and analyze events in real-time as they unfold. In this context, we recently proposed *ArmaTweet*—a system developed in collaboration among armasuisse and the Universities of Oxford and Fribourg to support semantic event detection on Twitter streams. Our experiments have shown that *ArmaTweet* is successful at detecting many complex events that cannot be detected by simple keyword-based search methods alone. Building up on this work, we explore in this paper several approaches for event detection on microposts. In particular, we describe and compare four different approaches based on keyword search (*Plain-Seed-Query*), information retrieval (Temporal Query Expansion), Word2Vec word embeddings (*Embedding*), and semantic retrieval (*ArmaTweet*). We provide an extensive empirical evaluation of these techniques using a benchmark dataset of about 200 million tweets on six event categories that we collected. While the performance of individual systems varies depending on the event category, our results show that *ArmaTweet* outperforms the other approaches on five out of six categories, and that a combined approach offers highest recall without adversely affecting precision of event detection.

- Secure Location Verification: Why you Want your Verifiers to be Mobile?, Matthias Schäfer, Carolina Nogueira, Jens B. Schmitt and Vincent Lenders, *Esorics Workshop on Attacks and Defenses for Internet-of-Things (ADIoT)*, Luxemburg, September 2019.

  Abstract - The integrity of location information is crucial in many applications such as access control or environmental sensing. Although there are several solutions to the problem of secure location verification, they all come with expensive requirements such as tight time synchronization, cooperative verification protocols, or dedicated hardware. Yet, meeting these requirements in practice is often not feasible which renders the existing solutions unusable in many scenarios. We therefore propose a new solution which exploits the mobility of verifiers to verify locations. We show that mobility can help minimize system requirements while at the same time achieves strong security. Specifically, we show that two moving verifiers are sufficient to securely verify location claims of a static prover without the need for time synchronization, active protocols, or otherwise specialized hardware. We provide formal proof that our method is secure with minimal effort if the verifiers are able to adjust their movement to the claimed location ("controlled mobility"). For scenarios in which controlled mobility is not feasible, we evaluate how more general claim-independent movement patterns of verifiers affect the security of our system. Based on extensive simulations, we propose simple movement strategies which improve the attack detection rate up to 290% with only little additional effort compared to random (uncontrolled) movements.

- Classi-Fly: Inferring Aircraft Categories from Open Data, Martin Strohmeier, Matthew Smith, Vincent Lenders and Ivan Martinovic, *arXiv:1908.01061 [cs.LG]*, July 2019.

  Abstract - In recent years, air traffic communication data has become easy to access, enabling novel research in many fields. Exploiting this new data source, a wide range of applications have emerged, from weather forecasting to stock market prediction, or the collection of intelligence about military and government movements. Typically these applications require knowledge about the metadata of the aircraft, specifically its operator and the aircraft category. *armasuisse Science + Technology*, the R&D agency for the Swiss Armed Forces, has been developing Classi-Fly, a novel approach to obtain metadata about aircraft based on their movement patterns. We validate Classi-Fly using several hundred thousand flights collected through open source means, in conjunction with ground truth from publicly available aircraft registries containing more than two million aircraft. We show that we can obtain the correct aircraft category with an accuracy of over

88%. In cases, where no metadata is available, this approach can be used to create the data necessary for applications working with air traffic communication. Finally, we show that it is feasible to automatically detect sensitive aircraft such as police and surveillance aircraft using this method.

- Safety vs. Security: Attacking Avionic Systems with Humans in the Loop, Matthew Smith, Martin Strohmeier, Jon Harman, Vincent Lenders, and Ivan Martinovic , *arXiv:1905.08039 [cs.CR]*, May 2019.

  Abstract - Many wireless communications systems found in aircraft lack standard security mechanisms, leaving them fundamentally vulnerable to attack. With affordable software-defined radios available, a novel threat has emerged, allowing a wide range of attackers to easily interfere with wireless avionic systems. Whilst these vulnerabilities are known, concrete attacks that exploit them are still novel and not yet well understood. This is true in particular with regards to their kinetic impact on the handling of the attacked aircraft and consequently its safety. To investigate this, we invited 30 Airbus A320 type-rated pilots to fly simulator scenarios in which they were subjected to attacks on their avionics. We implement and analyse novel wireless attacks on three safety-related systems: Traffic Collision Avoidance System (TCAS), Ground Proximity Warning System (GPWS) and the Instrument Landing System (ILS). We found that all three analysed attack scenarios created significant control impact and cost of disruption through turnarounds, avoidance manoeuvres, and diversions. They further increased workload, distrust in the affected system, and in 38% of cases caused the attacked safety system to be switched off entirely. All pilots felt the scenarios were useful, with 93.3% feeling that simulator training for wireless attacks could be valuable.

- On the Applicability of Satellite-based Air Traffic Control Communication for Security, Martin Strohmeier, Daniel Moser, Matthias Schäfer, Vincent Lenders and Ivan Martinovic, *IEEE Communication Magazine (COMMAG)*, 2019, September 2019.

  Abstract - As air traffic control communication moves toward digital systems, there is an emerging trend toward supplementing or even fully substituting the traditional air-ground link in favor of communication between aircraft and satellites. In this article, we analyze coverage and security against wireless attacks of the novel satellite-based version of the Automatic Dependent Surveillance-Broadcast (ADS-B) technology. We compare it to the widely deployed terrestrial ADS-B system, which is known to be insecure and is inherently unable to provide coverage in some parts of the global airspace, such as oceans and polar regions. Our analysis shows that satellites can provide vast advantages in such non-surveillance areas. However, they are as fundamentally insecure as terrestrial ADS-B.

- Unsupervised Wireless Spectrum Anomaly Detection with Interpretable Features, Sreeraj Rajendran, Wannes Meert, Vincent Lenders and Sofie Pollin, *IEEE Transactions on Cognitive Communications and Networking (TCCN), Volume: 5 , Issue: 3* , September 2019.

  Abstract - Detecting anomalous behavior in wireless spectrum is a demanding task due to the sheer complexity of the electromagnetic spectrum use. Wireless spectrum anomalies can take a wide range of forms from the presence of an unwanted signal in a licensed band to the absence of an expected signal, which makes manual labeling of anomalies difficult and suboptimal. We present, spectrum anomaly detector with interpretable features (SAIFE), an adversarial autoencoder (AAE)-based anomaly detector for wireless spectrum anomaly detection using power spectral density (PSD) data. This model achieves an average anomaly detection accuracy above 80% at a constant false alarm rate of 1% along with anomaly localization in an unsupervised setting. In addition, we investigate the model's capabilities to learn interpretable features, such as signal bandwidth, class, and center frequency in a semi-supervised fashion. Along with anomaly detection the model exhibits promising results for lossy PSD data compression up to 120× × × and semi-supervised signal classification accuracy close to 100% on three datasets just using 20% labeled samples.

Finally, the model is tested on data from one of the distributed electrosense sensors over a long term of 500 h showing its anomaly detection capabilities.

- Digital Radio Signal Cancellation Attacks - An Experimental Evaluation, Daniel Moser, Vincent Lenders and Srdjan Capkun, *ACM Conference on Security and Privacy in Wireless and Mobile Networks (WiSec)*, Miami FL, USA, May 2019.

  Abstract - Attacker models are the cornerstone of any security assessment. As attacker’s capabilities evolve overtime, it is keytore-evaluate periodically if attacker models that were deemed unrealistic in the past might not pose a possible threat today. In this work, we evaluate the threat of wireless radio signal cancellation attacks in the face of recent advancements in software-defined radio attacker capabilities. Unlike classical radio interference or jamming attacker models which add noise to the legitimate communication, signal cancellation attacks aim at interfering destructively with the legitimate signal in order to remove those signals from the spectrum. While signal cancellation attacks were deemed unrealistic in the analogue domain, we analyse the system requirements to perform such attacks digitally using SDRs and evaluate the feasibility to launch such attacks against wireless communication systems such as GPS. Our evaluation reveals that signal cancellation attacks that manage to attenuate up to 40dB of the signal at the receive rare feasible over the air. We further show that even complex CDMA signals such asGPScanbeattenuatedby30dB,evenbelowareceiver’snoise floor. These results indicate that digital signal cancellation attacks –especially against systems like GPS– should not be considered impossible per se, but deserve consideration when assessing the threat of attacks on wireless communication systems.

- Secrets in the Sky: On Privacy and Infrastructure Security in DVB-S Satellite Broadband, James Pavur, Daniel Moser, Vincent Lenders and Ivan Martinovic, *ACM Conference on Security and Privacy in Wireless and Mobile Networks (WiSec)*, Miami FL, USA, May 2019.

  Abstract - Demands for ubiquitous global connectivity have sparked a satellite broadband renaissance. Secure satellite broadband is vital to ensuring that this growth does not beget unanticipated harm. Motivated by this need, this paper presents an experimental security analysis of satellite broadband signals using the Digital Video Broadcasting for Satellite (DVB-S) protocol. This analysis comprises 14 geostationary platforms encompassing over 100 million square kilometers of combined coverage area. Using less than e300 of widely available equipment, we demonstrate the ability to identify individual satellite customers, often down to full name and address, and their web browsing activities. Moreover, we find that these vulnerabilities may enable damaging attacks against critical infrastructure, including power plants and SCADA systems. The paper concludes with a discussion of possible confidentiality protections in satellite broadband environments and notes a need for further cryptographic research on link-layer encryption for DVB-S broadband.

- BlackWidow: Monitoring the Dark Web for Cyber Security Information, Matthias Schäfer, Martin Strohmeier, Marc Liechti, Markus Fuchs, Markus Engel and Vincent Lenders, *NATO CCD COE 11th International Conference on Cyber Conflict (CyCon)*, Tallinn, Estonia, May 2019.

  Abstract - The Dark Web, a conglomerate of services hidden from search engines and regular users, is used by cyber criminals to offer all kinds of illegal services and goods. Multiple Dark Web offerings are highly relevant for the cyber security domain in anticipating and preventing attacks, such as information about zero-day exploits, stolen datasets with login information, or botnets available for hire.

  In this work, we analyze and discuss the challenges related to information gathering in the Dark Web for cyber security intelligence purposes. To facilitate information collection and the analysis of large amounts of unstructured data, we present BlackWidow, a highly automated modular system that monitors Dark Web services and fuses the collected data in a single analytics framework. BlackWidow relies on a Docker-based micro service architecture which permits the combination

of both preexisting and customized machine learning tools. BlackWidow represents all extracted data and the corresponding relationships extracted from posts in a large knowledge graph, which is made available to its security analyst users for search and interactive visual exploration.
Using BlackWidow, we conduct a study of seven popular services on the Deep and Dark Web across three different languages with almost 100,000 users. Within less than two days of monitoring time, BlackWidow managed to collect years of relevant information in the areas of cyber security and fraud monitoring. We show that BlackWidow can infer relationships between authors and forums and detect trends for cybersecurity-related topics. Finally, we discuss exemplary case studies surrounding leaked data and preparation for malicious activity.

- Detection of Malicious Remote Shell Sessions, Pierre Dumont, Roland Meier, David Gugelmann and Vincent Lenders, *NATO CCD COE 11th International Conference on Cyber Conflict (CyCon)*, Tallinn, Estonia, May 2019.

  Abstract - Remote shell sessions via protocols such as SSH are essential for managing systems, deploying applications, and running experiments. However, combined with weak passwords or flaws in the authentication process, remote shell access becomes a major security risk, as it allows an attacker to run arbitrary commands in the name of an impersonated user or even a system administrator. For example, remote shells of weakly protected systems are often exploited in order to build large botnets, to send spam emails, or to launch distributed denial of service attacks. Also, malicious insiders in organizations often use shell sessions to access and transfer restricted data. In this work, we tackle the problem of detecting malicious shell sessions based on session logs, i.e., recorded sequences of commands that were executed over time. Our approach is to classify sessions as benign or malicious by analyzing the sequence of commands that the shell users executed. We model such sequences of commands as n-grams and use them as features to train a supervised machine learning classifier. Our evaluation, based on freely available data and data from our own honeypot infrastructure, shows that the classifier reaches a true positive rate of 99.4% and a true negative rate of 99.7% after observing only four shell commands.

- Design and Evaluation of a Low-Cost Passive Radar Receiver Based on IoT Hardware, Daniel Moser, Giorgio Tresoldi, Christof Schüpbach and Vincent Lenders, *IEEE Radar Conference (Radar)*, Boston, Massachusetts USA, April 2019.

  Abstract - Recent years saw an increase in computation power on Internet of things devices such as the Raspberry Pi. It is now common for such platforms to boast multiple CPU-cores with clock rates of 1 gigahertz and higher. We have taken this evolution as a motivator to see how far we can push the limit in performing complex operations on a large amount of data by implementing a passive radar system on the Raspberry Pi. To keep the costs of our system further down, we evaluated the use of low-cost RTL-SDR receivers. Our work shows that today's IoT devices allow real-time processing for passive radar applications for both, FM and DAB signals. With our low-cost receiver, we were able to receive echos of aircraft several kilometers away.

- Collaborative Wideband Signal Decoding using Non-coherent Receivers, Roberto Calvo-Palomino, Héctor Córdobes de la Calle, Domenico Giustiniano, Fabio Ricciato, Vincent Lenders, *ACM/IEEE Conference on Information Processing in Sensor Networks (IPSN)*, Montreal, Canada, April 2019.

  Abstract - In recent years we are experiencing an important growth of interest for sensing the electromagnetic spectrum and making its access more agile. Emerging initiatives use low-cost receivers in large deployments for sensing the radio spectrum or collecting air-traffic signals at large scale. One of the major drawbacks of low-cost spectrum receivers is their limited sampling rate, which does not allow to decode wideband signals. In order to circumvent the hardware limitations of single receivers, we envision a scenario where non-coherent receivers sample the signal collaboratively to cover a larger bandwidth than the one of the single receiver and then, enable the signal reconstruction and decoding in the backend. We present a methodology to enable the signal re-

construction in the backend by multiplexing in frequency a certain number of non-coherent receivers in order to cover a signal bandwidth that would not otherwise be possible using a single receiver. We propose a method that does not use the knowledge of the modulation scheme, and has been designed to be transparent to the subsequent decoding process. As such, it is equivalent to the reception of the signal by a high-end receiver. We demonstrate and evaluate our approach with two non-coherent receivers which collaboratively sample an aviation signal of almost twice the bandwidth of each receiver. The experimental results show that, using two non-coherent receivers, our method is able to reconstruct and decode correctly more than 80% of data.