Open About the Open-Rate?

Shirin Kalantari

To cite this version:

Shirin Kalantari. Open About the Open-Rate?. 15th IFIP International Summer School on Privacy and Identity Management (Privacy and Identity), Sep 2020, Maribor, Slovenia. pp.187-205, 10.1007/978-3-030-72465-8_11 . hal-03703758

Open about the open-rate?
State of email tracking in marketing emails and its effects on user's privacy

Shirin Kalantari

imec-DistriNet, KU Leuven Leuven, Belgium
shirin.kalantari@kuleuven.be

Abstract. While many suspected that commercial email communications would be obsolete by 2020, email still continues to prevail over other mediums in terms of bringing revenue. In order to take full advantage of this valuable channel, commercial emails are tagged with tracking measures that at the very least enable senders to obtain individual read receipts for their emails. The collection of these read receipts, referred to as the *open-rate*, is used to measure the success of the campaign. In this paper we investigate the implications of email tracking, as it is used for obtaining open-rates, on recipients' privacy. In addition, we demonstrate the prevalence of email tracking in marketing emails of 736 websites and provide suggestions for mitigating its privacy risk.

1 Introduction

While web traffic dominates our modern usage of Internet, email is another inevitable part of our online life. Many suspected that email would fade away by the advent of alternative technologies [6,37,42] but in reality, email still continues to be among the most fundamental building blocks of our online life. In addition to delivering day-to-day communications, emails are widely used for distributing marketing contents [20]. It is estimated that marketing emails account for 44% of all emails in a user's inbox [1]. Popularity of email for distributing marketing content is partly due to its high Return of Investment (ROI) rate. As a recent report suggests, marketers can expect an average ROI of €42 for every €1 they spend on email marketing [46].

Being a strategic channel in terms of bringing revenue, marketing emails often include analytics measures that enables marketers to measure the effectiveness of their email marketing strategy. Stimulating user engagement is considered as one of the main goals of email campaigns [46]. The campaign *open-rate* is a metric used to represent user engagement by indicating the proportion of the recipients who opened a certain marketing email.

Techniques for increasing campaign open-rates are highly sought after. Several email marketing businesses offer dedicated services for increasing open-rates. For example, the company, *phrasee*[1], offer personalized machine generated sub-

[1] https://phrasee.co/

ject lines that promise to boost email open-rates. Similarly, various email marketing platforms offer *send time optimization* services [41,32] that personalize email delivery time to be the “optimum” time in which each recipient is most likely to open the email. In fact, even from an academic perspective, techniques are proposed for increasing campaign open-rates such as the research by Sahni et al. [44].

To calculate the open-rate of their email campaign, marketers take advantage of *HTTP requests for remote images to track their emails*. Therefore, instead of directly embedding images in the email message, they host images on a remote web server and include their URL address in the email message. At the moment when the recipient opens one of these emails, a series of HTTP requests are initiated by the email client for loading these remote images. These HTTP requests carry the information required for tracking email open-rates in three forms: *meta-data*, *headers*, and *URL parameters*. While meta-data and headers are generic to all HTTP requests, URL parameters are chosen by the sender and can be *personalized* to carry identifying information about the recipient. A primitive (yet still widely used) example for such personalization is a remote image that uses the recipient email address as part of its URL.

The information carried in HTTP *meta-data*, *headers*, or *URL parameters* allow senders to infer more than just the campaign open-rates. Especially, with personalized parameter the HTTP requests act as *individual read receipts* for emails enabling senders to obtain fine grained information about a user’s interaction with their email such as time of opening in addition to devices and software used for opening the email. Concerns about the privacy implications of email tracking using personalized URLs has been raised since 1995 [45] and previous academic works revealed the prevalence of this method of tracking in marketing emails [4,21,11,30].

In this paper we aim to provide a comprehensive overview of marketing email eco-system. We demonstrate the prevalence of email tracking techniques in a corpus of 237,741 marketing emails that we collected from 736 websites. Based on this corpus we try to give some additional insights about email tracking techniques and also highlight some misapprehension regarding email tracking techniques.

2 Background

Email started out very simple, the Simple Mail Transfer Protocol (SMTP) can only carry textual messages that are represented by US-ASCII. In 1991, Multipurpose Internet Mail Extensions (MIME) relaxed this restriction by defining algorithms that encode the email message to US-ASCII. Although the first motivation for using MIME was to support European characters in email [39], its introduction also enabled sending emails with richer text formatting such as HTML. With HTML, email messages are no longer restricted to textual content as they could contain well-designed messages with integrated multimedia contents that render consistently across different email clients.

In addition to having styling and richer text formatting, an HTML email often contains references to remote resources. These resources are loaded by the email client, when the recipient opens the email. Email tracking is inherently linked to emails with HTML content, which consequently generates HTTP requests for all embedded resources. This HTTP request leaks information in three forms: *meta-data*, *headers*, and *URL parameters*. In this section we give an overview of email marketing eco-system and the protocols used in email, further elaborate on each of the three forms of HTTP tracking and discuss the effectiveness of existing countermeasures in stopping them.

2.1 Overview of the Email Marketing Eco-system

Main Entities: The main entities that drive the email marketing ecosystem are:

- *Campaign owners:* These are the businesses that reach their audience via email to deliver services. For example news services that send headlines, news digest and briefs via email and e-commerce platforms that promote sale campaign or transactional emails. The end users either explicitly subscribe to newsletters or receive transactional emails as part of the services they are using (e.g., social media updates, online purchase confirmation, reminders, etc.). The campaign owner might want to include a number of third parties into their campaign emails to improve their services or to monetize their audiences. For example they might integrate advertisement partners such as Facebook or Instagram Ads; email optimization services such as mailing list sanitization tool *ZeroBounce*[2] or subject line personalization service *phrasee*; marketing platforms such as *Salesforce*[3] or *Google Analytics*[4].
- *Email Service Providers (ESPs):* While sending emails is very important for campaign owners, it is not part of their core business. As a result, they involve ESPs to manage and send out their campaign emails. The ESP manages the mailing list, provides email templates, and most importantly sends out the campaign emails [5]. In addition, an ESP provides integration tools, allowing marketers to seamlessly integrate third parties into their marketing platform. The ESP Mailchimp offers more than 250 integration tools to its customer.[6]
- *Mailbox providers:* Each email address is registered with a mailbox provider. It offers email hosting for users to send, receive and store their email messages. Gmail and Yahoo! are examples of widely used mailbox providers. Additionally, mailbox providers offer email security services such as spam filtering, malware detection, and transport layer encryption to protect users from malicious content.

[2] `https://www.zerobounce.net/`

[3] `https://www.salesforce.com/`

[4] `https://analytics.withgoogle.com/`

[5] Mailchimp, Selligent, and Campaign Monitor are examples of well-known ESPs.

[6] `https://mailchimp.com/integrations/`

- *Email Clients:* Also called Mail User Agents (MUAs), email clients are software such as Thunderbird and Microsoft Outlook that recipients use to open, read and interact with their emails.

Email Delivery and Protocols: Figure 1 is an overview of key components and protocols that are used in sending and retrieving an email. To send an email the ESP submits it over SMTP to the marketers' (i.e., email marketer) Mail Transfer Agent (MTA) who is in charge of transmitting emails and relaying each email to its recipient network via SMTP (step 1). Before routing an email, each MTA performs certain checks like validating email message format, spam filtering, and malware detection on the email. If the email fails the checks, MTA sends a bounce message with a status code, like those described in RFC 3463 [49], to the ESP (step 2&3). Finally, the email arrives at the final MTA, which is the recipient's mailbox provider. Afterward, the recipient can use an email client to retrieve the newly arrived email via email access protocols such as POP3 or IMAP (step 4). Before rendering the email, the email client performs a *preprocessing* step to sanitize the email message (step 5).

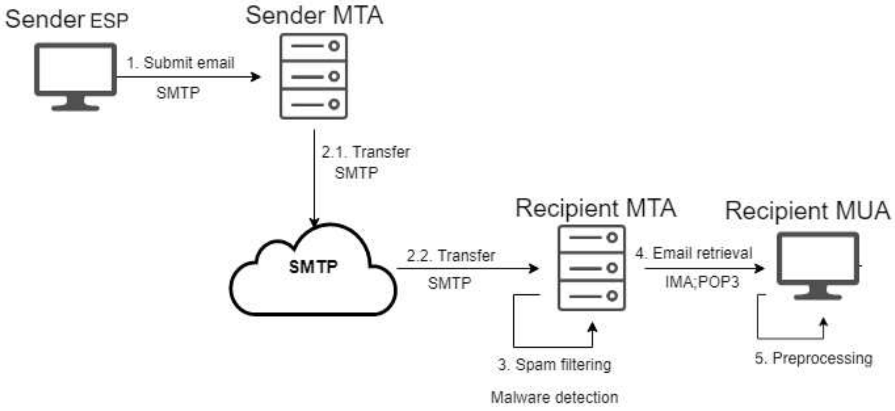


Fig. 1. Overview of software and protocols used in sending marketing emails.

Email Marketing Guidelines: For campaign owners, *deliverability* and *consistent rendering* of their emails are of outmost important. These two factors are directly affected at two points in the email transportation process: *spam filtering performed by the MTA (step 3)* and *preprocessing by the email client (step 5)*:

- *Spam filtering:* Compliant with the spam filtering guidelines that is enforced by mailbox providers and MTAs, ensures that each email gets delivered to its intended recipients and spam emails are filtered out. CAN-SPAM Act [16] is an example of such regulations that is currently in place in the US. Getting the recipient's consent in form of confirmation email and providing an opt-out option, through an unsubscribe link or List-Unsubscribe header [2], are among the basic requirements for sending bulk and transactional emails. To classify an email as spam, most spam filters use the text of the email message [43,5]. A technique that spammers use to circulate these textual filters is a so-called *image spam* in which spammers format their whole messages inside images [5,25]. Figure 2 is an example of an image spam email. Thus, including embedded images in an email alerts spam filters that the email might contain an image spam. Gmail use optical character recognition (OCR) techniques to extract the text from an image and run their spam filters on it [19]. Email service providers advise against embedded images[7] and recommend external images instead [33,22,36].

- *Preprocessing by the email clients:* Consistent rendering is another important concern for email marketers which is affected by the *HTML preprocessing step of email clients.* In the preprocessing step, based on email client policy, some HTML tags are removed (*HTML stripping*) and certain elements are overwritten (*HTML overwriting*) [27]. The HTML stripping removes HTML tags that cause serious attacks in email. For example, `<script>` tag is strictly removed by all email clients. This is due to the *Reaper exploit* that was discovered in 1998 by Carl Voth [51] and demonstrated that by running javascript, an attacker can wiretap email communications. Some email clients also remove external CSS files since they open an attack surface that can be exploited to change the content of an email. The exploit called *Ropemaker* enables a malicious attacker to change the content of an email after it is sent, just by changing the content of the external CSS that is used inside an email [17]. In the HTML overwriting step the email client overwrites parts of the HTML email for example, to block remote contents the email client change the URL of remote images to prevent the HTML rendering engine from automatically requesting them (see Figure 3 as an example of HTML overwriting which prevents remote images from automatically loading). It should be noted that email clients *make different choices* about HTML stripping and overwriting. For example, Apple email clients such as Apple Mail and iOS Mail block images by HTML overwriting but do not block remote CSS files (no `<link>` stripping or overwriting) [7], Thunderbird blocks images but does not remove remote CSS (no `<link>` stripping) which exposes recipients to attacks such as Ropemaker when they choose to load remote contents. Campaign owners format their email templates with an eye on these differences, and they use ESPs' testing services to ensure that their email renders properly across different email clients.

[7] There are two methods for embedding images in emails: data URI and Content-ID (CID). With data URI, the `src` attribute of an `<img>` include the 'immediate data' [34]. CID images come as attachments to emails and the image `src` attribute reference to the MIME Content-ID of the attachment[34].

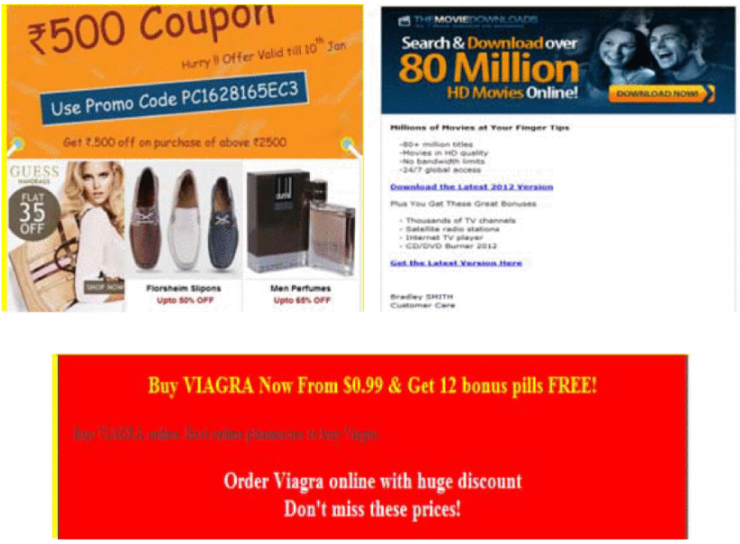

₹500 Coupon
Hurry !! Offer Valid till 10th Jan
Use Promo Code PC1628165EC3
Get ₹.500 off on purchase of above ₹2500
GUESS
FLAT 35 OFF
Florsheim Slipons
Upto 50% OFF
Men Perfumes
Upto 65% OFF
THEMOVIEDOWNLOADS
Search & Download over
80 Million
HD Movies Online!
DOWNLOAD NOW!
Millions of Movies at Your Finger Tips
-80+ million titles
-Movies in HD quality
-No bandwidth limits
-24/7 global access
Download the Latest 2012 Version
Plus You Get These Great Bonuses
- Thousands of TV channels
- Satellite radio stations
- Internet TV player
- CD/DVD Burner 2012
Get the Latest Version Here
Bradley SMITH
Customer Care
Buy VIAGRA Now From $0.99 & Get 12 bonus pills FREE!
Order Viagra online with huge discount
Don't miss these prices!

Fig. 2. Examples of image spams taken from the study by Ketari et al. *A Study of Image Spam Filtering Techniques*[25]

2.2 Remote resources and HTTP requests

The HTTP requests for remote resources are the seeds of email tracking. However, it could be argued that remote resources are unavoidable in marketing emails. Marketing emails are in nature call-to action and they depend on the linkability of the HTML links. For displaying images, while embedded alternatives exist that do not require a HTTP request, their usage is discouraged as they affect deliverability of the email. In this part we take a closer look at privacy implication of HTTP requests in email and discuss the effectiveness of existing countermeasures in stopping them. As already mentioned an HTTP request carries three categories of information that are interesting for trackers: *meta-data, HTTP headers and personalized URL tokens.* An HTTP request can be generalized in the following form:

```
GET request-URL [request-header]*
```

Meta-data: Since HTTP is an application layer protocol, it depends on transport layer protocols such as TCP/IP. These protocols reveal meta-data information about the email client. For instance, the IP address, ports and packet size. Xu et al. [52] demonstrate that by combining IP address and other tracking methods, long term surveillance attacks can be launched upon recipients revealing their geolocation information and their email reading habits. Information about recipients' timezone can also be inferred based on meta-data. This is used by ESPs to deliver emails according to users' timezone [8,31].

HTTP headers: The `request-header` is one or more HTTP headers that the email clients attach to the request to help the server provide a tailored response.

HTTP headers are used for email tracking. Englehardt et al. [11] demonstrated that email clients send the `Cookie` along with the HTTP requests which enables the sender to link the requests with the recipient's web profile. Bender et al. [3] show that `User-Agent` header is used by sender to infer information about the recipient's device to deliver advertisement accordingly.

Personalized URL tokens: The `request-URL` in the representation above, is the address of the remote resource. In email, it can also be personalized to contain identifying information about the recipient. This could be any string that map to the recipient email address at the server side. Englehardt et al. [11] considered the email address of the recipient, or a combination of hashing and encoding functions applied on it. They considered these tokens as Personally Identifying Information (PII) and looked for cases where they are shared with third parties. Haupt et al. [21] and Maass et al. [30] use multiple subscription to find personalized URLs, by comparing the URL structure of emails sent to multiple users.

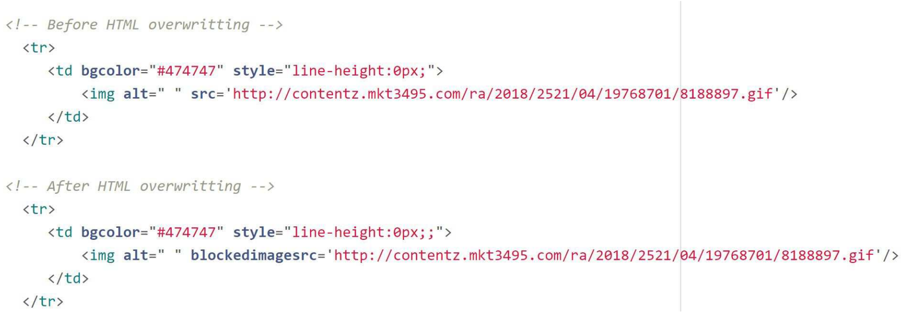
2.3 Countermeasures:

Blocking Remote Contents: This countermeasure is deployed in every modern email client, either by default or through user settings. To block remote contents, the email client changes the URL of remote resources in the HTML overwriting step which prevents the rendering engine from automatically requesting them. Figure 3 is an example of HTML overwriting to block remote images in Outlook web. While blocking remote contents stops all HTTP-based tracking, it imposes negative effects on the user experience. Especially since some email clients take a rough approach to blocking remote contents. For example, blocking remote contents also implies *blocking embedded images* in several email clients such as Yahoo! [22]. Note that loading embedded images does not require any HTTP requests, though it can trigger image spam emails and can be used to expose recipients to a sophisticated phishing attack [26,35]. Currently the only email client that provide more fine-grained content blocking is ProtonMail[8], which offers a multi-level option for loading embedded and remote images as shown in Figure 4.

Disabling HTML: Most email clients allow users to disable HTML as a countermeasure against email tracking [48]. If the HTML part never gets rendered, there will be no HTTP requests. The email client then ignores the `text/html` MIME parts and will use the `multipart/alternative` text parts. According to MIME specification in RFC 2046, *"... the content of the various parts are interchangeable"* [15]. However, this is under the assumption that the senders do provide alternative MIME parts for their emails.

Content Proxies: This is the most effective, existing countermeasure for minimizing the risks of email tracking. Content proxies are currently only deployed

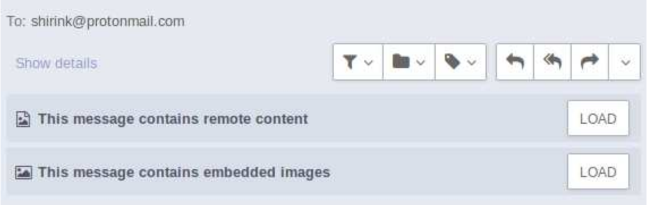
[8] `https://protonmail.com/`



```
<!-- Before HTML overwritting -->
  <tr>
     <td bgcolor="#474747" style="line-height:0px;">
         <img alt=" " src='http://contentz.mkt3495.com/ra/2018/2521/04/19768701/8188897.gif'/>
     </td>
  </tr>

<!-- After HTML overwritting -->
  <tr>
     <td bgcolor="#474747" style="line-height:0px;;">
         <img alt=" " blockedimagesrc='http://contentz.mkt3495.com/ra/2018/2521/04/19768701/8188897.gif'/>
     </td>
  </tr>
```

Fig. 3. HTML overwriting in Outlook web, the `src` attribute of the image is overwritten.



To: shirink@protonmail.com

This message contains remote content

This message contains embedded images

Fig. 4. Providing a multi-level option for loading embedded and external image in ProtonMail.

by Google [18] and Yandex and can fully mitigate tracking based on *meta-data* and *HTTP headers*. The proxy make the HTTP requests for remote contents in email and serve the response back to the email client. The request has the meta-data and the HTTP headers of the proxy and reveals nothing about the recipient. However, content proxies do not change the URL of the requests. Hence, they *cannot protect against tracking using personalized tokens*.

Browser Extensions: There are several browser extensions that aim to prevent email tracking by identifying and blocking beacons images in the email. Beacons are images in the size of a few pixels that incorporate third parties into email. Two examples of plugins that block pixels are *Ugly Mail*[9] and *Trocker*[10]. Using advertisement blocking extensions also mitigate some of the privacy risks of email tracking as it blocks all the requests to known third party trackers [11].

[9] `https://uglyemail.com/`

[10] `https://trockerapp.github.io/`

3 Data collections

3.1 Collecting newsletter corpus with multiple subscription

To illustrate how the prevalence of URL personalization in marketing emails, we set up a mail server and collected a corpus of commercial newsletter emails by subscribing with multiple identities to each newsletter service. In order to find newsletters forms we crawled Alexa top 10K sites[11]. We adapted the crawler specification of Englehardt et al. [11] to subscribe with multiple email addresses to each newsletter. This multiple subscription enables us to find personalization tokens by comparing the URL structure of emails sent to multiple users.

The crawler and mail server communication is summarized in Figure 5. A web server is set to act as the intermediary between the crawler and the mail server. First, the crawler searches each site for a subscription form (step 1). Once a potential subscription form is found, the crawler requests a new email address from the web server along with information about the website in which this email address is going to be submitted (step 2). The web server generates a unique email address for this site and registered this email address on the mail server (step 3). The web server then sends the newly generated email address to the crawler to submit it to the subscription form. In order to subscribe with multiple email addresses to each newsletter, step 2-4 are then repeated.

When the subscription form has been submitted, the website will often send a *confirmation email* to the specified email address. This email contains a *confirmation link* and only after this link is clicked, the subscription is considered finalized and newsletter emails start to arrive. Confirmation emails intend to reduce the risk of newsletter emails being identified as spam. By clicking on the *confirmation link*, the entity in possession of that email address confirms that (s)he is willing to receive further emails from this sender. The confirmation link extraction scheme is specified in the next section. Once the mail server finds the confirmation link, it submits it to the web server so that the crawler can find it and click on it.

3.2 Email preprocessing

For each incoming email, the mail server extracts the HTML message by retrieving the MIME part `text/html`. It stores the images, links, and checks whether the email include an embedded image. It stores links to external files (`<link>`), checks if there are any `<script>` tag in the HTML structure and also searches for URLs within `<style>` tag.

To extract the text part of an email, the mail server stores both the text alternative part and the text extracted from the HTML part. It also extracts some meta-data related to the email header.

To find the confirmation link, the mail server searches the *first incoming email* of each inbox (i.e., email address. It extracts all `<a>` tags from the HTML

[11] From Alexa top 1 Million list used in [11] available at `https://github.com/citp/email_tracking/blob/master/crawler_mailinglists/data/top-1m.csv`

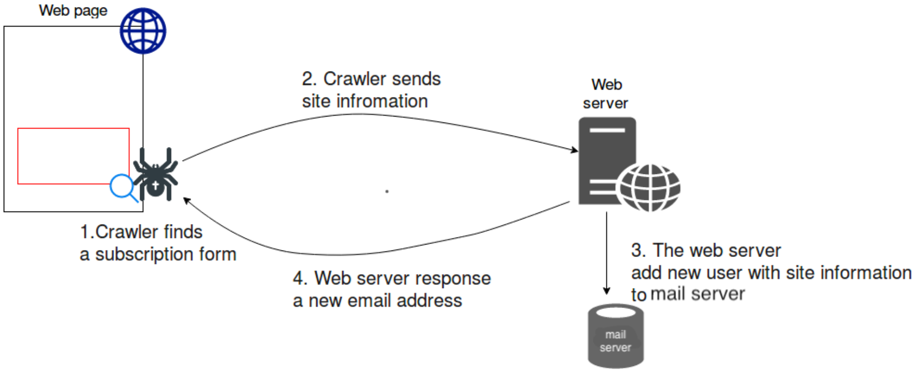
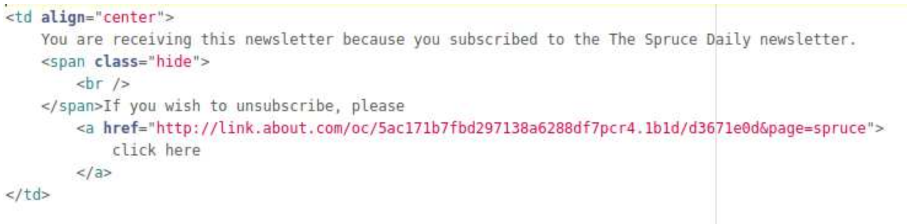


Fig. 5. Crawler communicates with our web server.

message and checks if the text of the tag contains one of the keywords *confirm, validate, finalize, subscribe, activate, step*. If no such link is found, it checks if the URL in the `href` attribute contains one of these keywords. In case no link is found, the mail server is going to adapt a bottom-up approach for finding the confirmation link: for each link in the email, it is going to search for the keyword in the text of the enclosing HTML element (see Figure 6).

The mail server also checks every incoming email for implementation of anti-spam practices. It first searches for evidence of *email address sharing*. It checks whether the sender email address has the same domain name as the website on which the email address was registered. Providing opt-out options is also checked using the `List-Unsubscribe` header, and the unsubscribe link within the HTML message. For finding the unsubscribe link the same approach as finding confirmation link is repeated, using keywords *opt-out*, *unsubscribe*, *opt out* and *unsub*.



```
<td align="center">
    You are receiving this newsletter because you subscribed to the The Spruce Daily newsletter.
    <span class="hide">
        <br />
    </span>If you wish to unsubscribe, please
        <a href="http://link.about.com/oc/5ac171b7fbd297138a6288df7pcr4.1b1d/d3671e0d&page=spruce">
            click here
        </a>
</td>
```

Fig. 6. An unsubscribe link found by searching the text of the parent element of `<a>` tag.

3.3 Identifying personalized parameters

We identify personalized URLs parameters using *differential testing*. In this method multiple email addresses are subscribed to the same newsletter. Personalized URL parameters are identified by comparing the URL structure of the same campaign email that is sent to these different inboxes. To do so we perform the following steps:

- We first identify email addresses that are subscribed to the same site.
- For websites on which multiple users are subscribed we calculate the cosine similarity of pairwise emails. We use the text extracted from the HTML part, to calculate text similarity. These steps avoids comparing emails that are subjected to A/B testing. If the similarity of two emails is more than 95%, we consider them to be the same campaign email.
- When two emails are identified as identical, we compare the URL structure of their images and links based on their order of appearance in HTML structure.
- We then compare the two URLs. If they are different in a substring of 5 or longer, we consider them personalized. We assume that such string has enough entropy to identify the recipient.

4 Results

In total our crawler made 187,886 subscription attempts (and, hence, so many requests for a new email address on the mail server). The crawler might have filled in a contact form, or a comment section. 6,160 users received at least two emails during the period. The mail server was running between 2018/04/01 and 2018/06/10. During this period, we received 237,741 newsletter emails from 736 websites.

External images and links are the most common remote contents in email: Analyzing the prevalence of remote resources in our corpus shows that links and images together account for 97.25% of remote resources in newsletter emails.[12] However, in terms of privacy risks of images are more hazardous since they are made without user involvement as soon as the client renders an email.

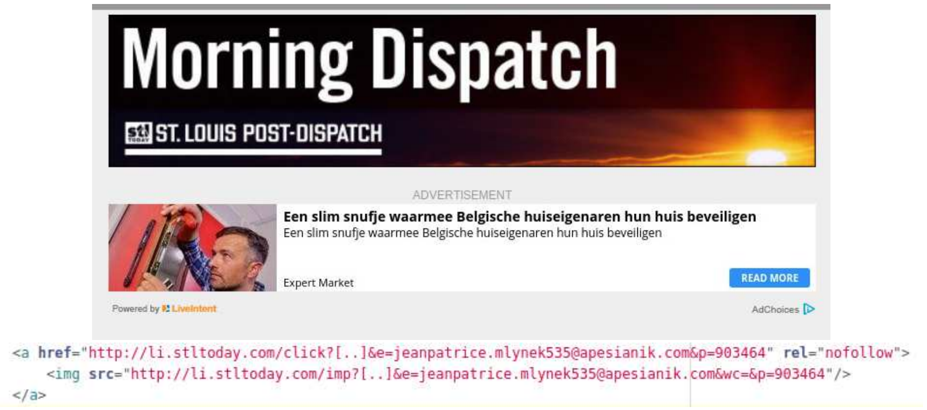
Marketing emails follow anti-spam regulation and guidelines: The corpus confirms that anti-spam practices are followed by email marketers:

- No instances of email address distribution were detected (i.e., each inbox received emails only from the website on which it was registered).
- An unsubscribe link is found in 87.58% of all emails. In addition, the `List-Unsubscribe` header is used in 89.48% of emails.

[12] HTML `<a>` tag account for 52.26% and `<img>` tag for 44.99%

URL personalization is very common: Almost every marketing email in our corpus contains images or links with personalized URLs. For 58% of senders, the personalized token was persistently used in all the emails they sent (for the remainder the URL parameter changes in different emails). Our dataset indicates no instances of URL personalization for other remote resources such as CSS files.

Distinction between third party and first party is blurry in email: Previous studies consider the distribution of third-parties in emails as indication of email tracking [4,11,30,23]. In these studies, the URL of third-party contents *use a different domain name* than the email sender, or the website on which the email address was registered. Most web tracking protections use domain blacklisting for identifying tracking content. However, our corpus shows that this distinction does not always hold in email as we find instances of *third-party advertisement using the first-party domain.*



Een slim snufje waarmee Belgische huiseigenaren hun huis beveiligen
Een slim snufje waarmee Belgische huiseigenaren hun huis beveiligen

Expert Market

```
<a href="http://li.stltoday.com/click?[..]&e=jeanpatrice.mlynek535@apesianik.com&p=903464" rel="nofollow">
    <img src="http://li.stltoday.com/imp?[..]&e=jeanpatrice.mlynek535@apesianik.com&wc=&p=903464"/>
</a>
```

Fig. 7. A LiveIntent advertisement and its HTML code snippet which users a subdomain of the sender *stltoday.com.*

We find 14 senders that use their own domain to serve third party advertisement. These advertisements are served by LiveIntent[13], a Supply Side Platform (SSP) that enables publishers to receive revenue by managing their advertising space inside their emails. LiveIntent is among trackers that receive the highest number of PII information in the paper by Englehardt et al. [11]. LiveIntent advertisements are served through a so-called LiveTag that uses a dedicated subdomain of the email marketer to serve the advertisement. Figure 7 is an example of a LiveIntent advertisement that uses a subdomain of the sender, *stltoday.com.* To find the advertisement element, we use the general HTML and URL structure of LiveTag [28]: find everu `<a>` element that has an `<img>` tag as

[13] `https://www.liveintent.com/`

its immediate child in the HTML structure, check if URLs in the `src` and the `href` attributes of these elements contain the corresponding query parameters of a LiveTag (p=, and either e= or m=). In addition, for each of these senders we rendered one email and manually verified whether the identified tag is serving an advertisement. Table 1 illustrates the results. To check whether these images could get blocked by current countermeasures, we checked the domains in Table 1 in trackers list of the ad-blocker Disconnect.me[14], and EasyList[15]. None of these domains are among the online trackers that will be blocked by ad-blockers that are using these lists.

Domain	Advertisement URL	Presence in email
al.com	eads.al.com	59.61%
alternet.org	li.alternet.org	49.69%
cleveland.com	eads.cleveland.com	62.27%
dealnews.com	c3.dealnews.com	96.38%
nj.com	eads.nj.com	61.38%
nola.com	eads.nola.com	58.33%
philly.com	li.philly.com	79.10%
realtor.com	li.realtor.com	17.24%
seriouseats.com	li.seriouseats.com	17.14%
stltoday.com	li.stltoday.com	40.08%
tigerdirect.com	li.tigerdirect.com	88.76%
timesofisrael.com	nl.timesofisrael.com	89.62%
townhall.com	li.townhall.com	14.53%
travelocity.com	content.travelocity.com	20%

Table 1. The domain names that were used to serve LiveIntent advertisement.

5 Discussion

5.1 Privacy concerns of email tracking

The result shows that using personalized parameters is common in marketing emails which enables them to obtain an HTTP-based read receipt for their emails. It is reasonable to compare this form of email tracking with norms of obtaining read receipts in similar applications. This is in-line with the contextual definition of privacy [38] that uses informational privacy norms for assessing privacy risks. Email standards provision protocols for obtaining read receipt emails through Message Disposition Notification (MDN) [47]. Different privacy concerns of MDN have been discussed in RFC 8098 for example it emphasizes on obtaining user consent before sending MDNs: *"[...]While Internet standards normally do not specify the behavior of user interfaces, it is strongly recommended that the*

[14] `https://github.com/disconnectme/disconnect-tracking-protection`

[15] `https://easylist.to/easylist/easylist.txt`

user agent obtain the user's consent before sending an MDN. [...] The purpose of obtaining user's consent is to protect user's privacy. The default value should be not to send MDNs." [47]. For this reason, email clients use a very explicit user interface before sending an MDN report.
Another application that involves sending read receipts are mobile messaging applications. Email has been compared with messaging applications quite often in the web and in academic research [13,50]. Table 2 shows policy of popular messaging apps in regard of read receipt. Unlike email, in messaging applications user often have control over sending read receipt and can reject them.

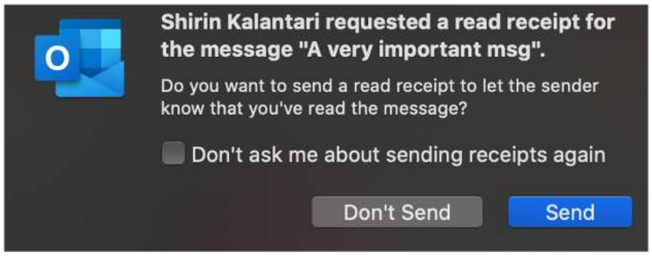
Application	Read receipt	Disabling read receipt
WhatsApp	✓	✓
Messenger	✓	✗
Skype	✓	✓
Telegram	✓	✗
Hangout	✗	✗
iMessage	✓	✓

Table 2. Popular messaging apps and their policy regarding read receipts.

Obtaining read receipts in email protocols and messaging applications is *unambiguous* and comes with *explicit interface* and *fair denial consequences*:

1. *Unambiguous:* In both MDN protocol and messaging applications, a read receipt is exactly what the name suggests: it indicates whether the recipient has opened a message. The same cannot be said about HTTP-based read receipts in email since remote contents are loaded *every time the email is opened*. Moreover, each request contains fine-grained meta-data such as the exact time at which the email was read, devices and software that were used and the location of the recipient.
2. *Explicit interface:* Read receipts use a clear and explicit user interface in messaging applications (e.g. a double blue check mark next to the message) and in MDN protocol (see Figure 8). However, the user survey of Xu et al. [52] revealed that the majority of participants had no awareness about the information leaks of loading remote contents in email.
3. *Fair denial consequences:* Recipients can reject sending MDN and disable read receipts in most messaging applications without sacrificing functionality. However, to prevent HTTP-based read receipts a user should either block remote contents, or disable HTML emails which both have a significant negative impact on the user experience.

While most of the discussions in this paper are around marketing emails, note that the same techniques for obtaining read receipts can be and *are being used* in day-to-day, conversational email communications. There are several email tracking services targeting personal email communications for example,



Shirin Kalantari requested a read receipt for the message "A very important msg".

Do you want to send a read receipt to let the sender know that you've read the message?

□ Don't ask me about sending receipts again

Fig. 8. Obtaining a read receipt through the MDN protocol in Outlook.

the renowned modern email client *Superhuman*[16] includes tracking pixels in every outgoing email and shows the read status of sent emails, *MixMax*[17], *Streak*[18] and *Yesware*[19] offer browser extensions for integration of email tracking services to popular email clients such as Gmail or Outlook. The privacy implications of email tracking for personal email communication is even greater, since it conveys additional information as illustrated in the following example by M. Davidson [10]:

"An ex-boyfriend ... pens a desperate email Subject: "I've been thinking about us". He sends it to his former partner. She reads it when she gets to work in Downtown Los Angeles at 9am. She reads it again before dinner with friends in Pasadena at 7pm. She reads it again at home in Santa Monica ... She decides not to answer the email ... [but] her email is always communicating, and it's sharing info she does not want to send and doesn't even know she is sending." [10]

While in this paper we mainly focus on privacy implications of email tracking for obtaining privacy invasive HTTP-based read receipts, note that email tracking is also a stepping stone to pervasive online tracking. Although performed by hackneyed techniques, email tracking is very effective to persistently track users over time and across devices. In fact, there are a number of online tracking businesses that are built around email tracking, promoting it as *the solution for cookie-less tracking across devices* [29,9,12]. Moreover, email tracking often includes third parties which receive yet more data about the users such as newsletters they are subscribed to, online services they are using, and their email reading habits. This information is valuable to spammers for *personalizing* their phishing and spamming attacks. Adding personalized context to phishing attacks amplifies its impact, as shown in the recent *Emotent* phishing campaign, which is at the moment of writing among the *"top malware threats affecting Europe"* [14]. Furthermore, there are businesses interested in sneaking into users' inboxes to gain business insights from their bulk emails. In 2017 *Unroll.me*, a free service that allows users to manage their newsletter subscriptions, sold parts of its users' data to Uber [24].

[16] https://superhuman.com/
[17] https://www.mixmax.com/
[18] https://www.streak.com/
[19] https://www.yesware.com/

Senders' Justification: adjusting sending rate. It is argued that some senders use email tracking in order to adjust their sending rate. Bender et al. [3] subscribed multiple email addresses to different newsletters but exhibited *different email reading behavior* including different email reading/opening frequencies for each account. Their findings confirm that email marketers respond to the user's behavior and adjust their sending frequency, i.e., they send fewer emails to less-active inboxes.

However, this practice mainly serves to the interest of sender as it aims to prevent users from reporting their emails as spam, which jeopardizes their reputation and delivery rate. Note that for recipients there is a difference between receiving fewer emails and unsubscribing from a newsletter. Unsubscribing from email communications has strict legal bindings that often protects the recipient such as mandating the sender to delete all the user data. In contrast, when a sender reduces sending frequency or even stops sending emails, the user data remains within the system. Instead of infecting emails with tracking tokens to steadily monitor users' interactions senders should put user in control and allow them to adjust their preferences.

5.2 Toward mitigation: Multi-level content blocking

While blocking remote contents can stop HTTP-based tracking, it is a rough approach with heavy impact on user experience. De Paula et al. [40] emphasis on providing multi-level countermeasures that enable users to choose different levels of risks according to their tasks. Similarly, the privacy concerns of email tracking could be reduced by involving the user. For example, giving users the option to only load certain images in email (e.g., only loading the banner of a newsletter). While this would not fully mitigate email tracking risks, it could minimize the scope of tracking. Despite its simplicity, practical implementation of a multi-level countermeasures is still missing in most email clients.[20]

6 Conclusion

In this paper we demonstrate the prevalence of email tracking in a corpus of 237,741 marketing emails. Our results illustrate that marketers are strictly following email communication guidelines by providing opt-out options and obtaining consent for sending emails. Though these guidelines have not progressed as the email eco-system evolves and do not protect users against the current methods of email tracking. As work in progress, we aim to assess the implications of email tracking with respect to European General Data Protection Regulation and ePrivacy directive.

[20] As mentioned in Section 2.3, ProtonMail is the only email client that we could find which provides a multi-level blocking for remote and embeded images.

References

1. The Direct Marketing Association (UK) LTD: Consumer email tracker 2019. https://dma.org.uk/uploads/misc/consumer-email-tracker-2019---v5.pdf, last accessed on 2020-12-15 (2019)
2. Baer, J., Neufeld, G.: The Use of URLs as Meta-Syntax for Core Mail List Commands and their Transport through Message Header Fields. RFC 2369 (July 1998)
3. Bender, B., Fabian, B., Haupt, J., Lessmann, S., Neumann, T., Thim, C.: Track and treat - usage of e-mail tracking for newsletter individualization. In: Twenty-Sixth European Conference on Information Systems (ECIS2018), Portsmouth,UK, 2018 (06 2018)
4. Bender, B., Fabian, B., Lessmann, S., Haupt, J.: E-mail tracking: Status quo and novel countermeasures. In: Proceedings of the thirty-seventh international conference on information systems (ICIS). Dublin,Ireland 2016 (12 2016)
5. Biggio, B., Fumera, G., Pillai, I., Roli, F.: Image spam filtering using visual information. In: 14th International Conference on Image Analysis and Processing (ICIAP 2007). pp. 105–110 (Sept 2007). https://doi.org/10.1109/ICIAP.2007.4362765
6. Brandon, J.: Why email will be obsolete by 2020. https://www.inc.com/john-brandon/why-email-will-be-obsolete-by-2020.html, last accessed on 2020-12-15 (April 2015)
7. CampaignMonitor: The ultimate guide to css. https://www.campaignmonitor.com/css/, last accessed on 2020-12-15
8. Clare, V.: Introducing time zone sending. https://www.campaignmonitor.com/blog/new-features/2017/04/introducing-time-zone-sending/, last accessed on 2020-12-15 (2017)
9. Conversant: Five building blocks of identity management. https://www.conversantmedia.com/hubfs/US%20Conversant/IMAGE%20ILLUSTRATIONS%20and%20VIDEOs/Resource-center-assets/PDFs/Five_Keys_to_Identity_Resolution_24Apr2019.pdf, last accessed on 2020-12-15
10. Davidson, M.: Superhuman is spying on you. https://mikeindustries.com/blog/archive/2019/07/superhuman-is-spying-on-you, last accessed on 2020-12-15 (2019)
11. Englehardt, S., Han, J., Narayanan, A.: I never signed up for this! privacy implications of email tracking. Proceedings on Privacy Enhancing Technologies **2018**(1), 109–126 (2018)
12. Epsilon: The way the cookie data crumbles: People-based profiles vs. cookie-based solutions. https://www.epsilon.com/hubfs/Cookie%20Crumbles.pdf, last accessed on 2020-12-15 (2019)
13. Ermoshina, K., Musiani, F., Halpin, H.: End-to-end encrypted messaging protocols: An overview. In: Bagnoli, F., Satsiou, A., Stavrakakis, I., Nesi, P., Pacini, G., Welp, Y., Tiropanis, T., DiFranzo, D. (eds.) Internet Science. pp. 244–254. Springer International Publishing (2016)
14. Europol: Internet Organised Crime Threat Assessment (IOCTA) 2020. European Union Agencyfor Law Enforcement Cooperation (Europol) (2020)
15. Freed, N., Borenstein, N.: Multipurpose Internet Mail Extensions (MIME) Part Two:Media Types. RFC 2046 (November 1996), https://tools.ietf.org/html/rfc2046
16. FTC.gov: Can-spam act: A compliance guide for business. URL: https://www.ftc.gov/tips-advice/business-center/guidance/can-spam-act-compliance-guide-business, last accessed on 2020-12-15

17. Gardiner, M.: Ropemaker email security weakness - vulnerability or application misuse? https://www.mimecast.com/blog/2017/08/introducing-the-ropemaker-email-exploit/, last accessed on 2020-12-15
18. Gmail: Turn images on or off in gmail. https://support.google.com/mail/answer/145919, last accessed on 2020-12-15
19. Google Workspace Admin Help: Use optical character recognition to read images. https://support.google.com/a/answer/6358855, last accessed on 2020-12-15
20. Handley, A., Stahl, S., Rose, R., Moutsos, K., McPhillips, C., Beets, L.M., Kalinowski, J., Reese, N.: B2b content marketing benchmarks,budgets, and trends—north america. https://contentmarketinginstitute.com/wp-content/uploads/2019/10/2020_B2B_Research_Final.pdf, last accessed on 2020-12-15 (2019)
21. Haupt, J., Bender, B., Fabian, B., Lessmann, S.: Robust identification of email tracking: A machine learning approach. European Journal of Operational Research **271**(1), 341 – 356 (2018). https://doi.org/https://doi.org/10.1016/j.ejor.2018.05.018
22. Hodgekiss, R.: Embedded image support in html email. https://www.campaignmonitor.com/blog/email-marketing/2019/04/embedded-images-in-html-email/, last accessed on 2020-12-15 (2019)
23. Hu, H., Peng, P., Wang, G.: Characterizing pixel tracking through the lens of disposable email services. In: 2019 IEEE Symposium on Security and Privacy (SP). pp. 365–379 (2019)
24. Isaac, M., Lohr, S.: Unroll.me service faces backlash over a widespread practice: Selling user data. https://nyti.ms/2pYH0Eb, last accessed on 2020-12-15 (2017)
25. Ketari, L.M., Chandra, M., Khanum, M.A.: A study of image spam filtering techniques. In: 2012 Fourth International Conference on Computational Intelligence and Communication Networks. pp. 245–250 (2012)
26. Klevjer, H.: Phishing by data uri (2012). https://doi.org/10.13140/2.1.4088.0007
27. Litmus: Why do some email clients show my email differently than others? https://help.litmus.com/article/158-why-do-some-email-clients-show-my-email-differently-than-others, last accessed on 2020-12-15
28. LiveIntent: How to implement livetags. https://support.liveintent.com/hc/en-us/articles/360001247043-How-to-Implement-LiveTags-, last accessed on 2020-12-15 (2020)
29. LiveIntent: Overview of custom audiences. https://support.liveintent.com/hc/en-us/articles/204889644-Overview-of-Custom-Audiences, last accessed on 2020-12-15 (2020)
30. Maass, M., Schwär, S., Hollick, M.: Towards transparency in email tracking. In: Naldi, M., Italiano, G.F., Rannenberg, K., Medina, M., Bourka, A. (eds.) Privacy Technologies and Policy. pp. 18–27. Springer International Publishing, Cham (2019)
31. Mailchimp: Use timewarp. https://mailchimp.com/help/use-timewarp/, last accessed on 2020-12-15
32. MailChimp: Insights from mailchimp's send time optimization system. https://mailchimp.com/resources/insights-from-mailchimps-send-time-optimization-system/, last accessed on 2020-12-15 (2014)
33. MailPoet: Why we don't allow embedded images. https://docs.mailpoet.com/article/26-embedding-images-in-emails-bad-idea, last accessed on 2020-12-15

34. Masinter, L.: The ”data” URL scheme. RFC 2397 (August 1998), https://tools.ietf.org/html/rfc2397
35. Maunder, M.: Wide impact: Highly effective gmail phishing technique being exploited. https://www.wordfence.com/blog/2017/01/gmail-phishing-data-uri/, last accessed on 2020-12-15
36. Mitchell, T.: Everything you know about email content filtering is wrong. https://blog.returnpath.com/everything-you-know-about-email-content-filtering-is-wrong/, last accessed on 2020-12-15
37. Nisen, M.: The future of work won't include email. https://www.businessinsider.com/why-email-should-become-obsolete-2013-1, last accessed on 2020-12-15 (2013)
38. Nissenbaum, H.: Privacy in Context: Technology, Policy, and the Integrity of Social Life. Stanford University Press (2009)
39. Partridge, C.: Partridgecraig2008ttdo. IEEE Annals of the History of Computing **30**(2), 3–29 (2008)
40. de Paula, R., Ding, X., Dourish, P., Nies, K., Pillet, B., Redmiles, D.F., Ren, J., Rode, J.A., Filho, R.S.: In the eye of the beholder: A visualization-based approach to information system security. International Journal of Human-Computer Studies **63**(1), 5 – 24 (2005). https://doi.org/https://doi.org/10.1016/j.ijhcs.2005.04.021
41. Roberts, C.: Announcing send time optimization. https://www.campaignmonitor.com/blog/new-features/2017/05/announcing-send-time-optimization/, last accessed on 2020-12-15 (2017)
42. Rogers, A.: As workplace communication evolves, email may not prevail. https://www.forbes.com/sites/ciocentral/2017/02/15/as-workplace-communication-evolves-email-may-not-prevail/, last accessed on 2020-12-15 (February 2017)
43. Sahami, M., Dumais, S., Heckerman, D., Horvitz, E.: A bayesian approach to filtering junk e-mail. In: Learning for Text Categorization: Papers from the 1998 workshop. vol. 62, pp. 98–105. Madison, Wisconsin (1998)
44. Sahni, N.S., Wheeler, S.C., Chintagunta, P.: Personalization in email marketing: The role of non-informative advertising content. Marketing science (Providence, R.I.) **37**(2), 236–258 (2018)
45. Storm, D.: The hidden privacy hazards of html email. https://strom.com/awards/192.html, last accessed on 2020-12-15
46. The Direct Marketing Association (UK) LTD: Marketer email tracker 2019. https://dma.org.uk/uploads/misc/marketers-email-tracker-2019.pdf (2019)
47. Tony, H., Alexey, M.: Message Disposition Notification. RFC 8098 (February 2017), https://tools.ietf.org/html/rfc8098
48. TorBirdy: Towards a tor-safe mozilla thunderbird reducing application-level privacy leaks in thunderbird. https://trac.torproject.org/projects/tor/attachment/wiki/doc/TorifyHOWTO/EMail/Thunderbird/Thunderbird+Tor.pdf, last accessed on 2020-09-07 (2011)
49. Vaudreuil, G.: Enhanced Mail System Status Codes. RFC 3463 (January 2003), https://tools.ietf.org/html/rfc3463
50. Vella, H.: Interview: messaging apps take on work email. https://www.raconteur.net/hr/messaging-apps-take-on-work-email, last accessed on 2020-12-15
51. Voth, C.: Reaper exploit. http://web.archive.org/web/20011005083819/http://www.geocities.com/ResearchTriangle/Facility/8332/reaper-exploit-release.html, last accessed on 2020-12-15

52. Xu, H., Hao, S., Sari, A., Wang, H.: Privacy risk assessment on email tracking. In: IEEE INFOCOM 2018 - IEEE Conference on Computer Communications. pp. 2519–2527 (April 2018). https://doi.org/10.1109/INFOCOM.2018.8486432