

**From:** Freeman, D. Reed [Reed.Freeman@wilmerhale.com](mailto:Reed.Freeman@wilmerhale.com)   
**Subject:** [espc-announce] FYI: Medium.Com: The 2020 URL Querystring Data Leaks — Millions of User Emails Leaking from Popular Websites to Advertising & Analytics Companies  
**Date:** May 2, 2020 at 1:57 AM  
**To:** James Campbell [campbell@espcalition.org](mailto:campbell@espcalition.org)



[The 2020 URL Querystring Data Leaks — Millions of User Emails Leaking from Popular Websites to Advertising & Analytics Companies](#)

[Zach Edwards](#)

[Apr 29](#) · 24 min read

---

**Most popular websites on the internet are using 3rd party analytics and advertising Javascript code — and depending on how a website sets up their marketing systems, typically email systems and new user signup flows, the user emails can accidentally and/or purposefully leak to companies across the global data supply chain.**

The organizations included in this research have hundreds of millions of emails and real users between them — and only Wish.com, Mailchimp and The Washington Post took this report on their user email breaches seriously — Wish updated their email system within ~72 hours of the report being sent and the other two started taking actions relatively quickly — whereas many other organizations either didn't respond or have failed to take any actions for weeks or months.

All organizations need to be aware of this significant user data vulnerability, but more importantly, **there needs to be significant efforts by organizations sharing user emails in this way, to submit \*partner deletion requests\* to the 3rd party advertising and analytics companies who received the user emails.**

Throughout this research, some of the advertising companies that were tracked receiving the user emails are included — but this should not be considered 100% complete due to the fact that this has been going on for years on some websites, and it's impossible to know externally all of the organizations who synced data on a specific website or webpage at any historical point.

***All organizations included in this research leaking user emails should publicly post the list of all***

***their historical advertising and analytics vendors who could have received the user emails while their respective breaches were active.***

One important trend to notice is how often Google Analytics, Google's DoubleClick, Facebook, and Twitter are ingesting the user emails — these are organizations that should be receiving deletion requests en-masse and they should all have processes to handle this type of effort already (Facebook likely has this tech already based on conversations on this research and additional research from a private report from several years ago).

In this research, there are also “red flag organizations” who have ingested user emails that are small or relatively unknown organizations, yet likely receiving huge amounts of user emails in their request logs. These smaller organizations need a unique type of scrutiny due to the power that an advertising or analytics company can attain from ingesting millions of user emails from their enterprise clients — [the Cambridge Analytica effect if you will...](#)

---

### **Numerous Enterprise Organizations Leaking User Emails Through 3rd Party Javascript Request Headers Sent via Browsers to 3rd Party Advertising & Analytics Companies**

Each of these orgs leaked user emails by unsafely appending the user email to a URL in plain text (or encoded in base64) and then having the user emails leak to 3rd party advertising and analytics companies.

When any 3rd party Javascript code loads on a website, metadata from the user and the website can be transmitted to the 3rd party domain / company that controls that code — this is technically through the “Request Headers” sent through a browser — and this data can include what page a user is visiting, what type of device and browser they are using, their location, and other forms of fingerprinting / cookies / URL querystring/ URL parameters that are used by advertising and analytics companies.

This type of email user data in a URL bar synced into Javascript pixels is most typically blocked by a regular person through “Ad blockers” or through browsers like Safari, Brave, and Firefox — those browsers use Javascript/cookie blocking as a default features to protect users (each browser handles it slightly differently). This breach and research included here would impact all

Chrome users of these websites who went through these specific user flows and who didn't proactively block all Javascript (a rarely used option) or use a Chrome "Ad blocker" extension that blocked this type of Javascript. Some people using the other "safe" browsers (Safari/Brave/Firefox) could have been protected from the leak due to their 3rd party Javascript requests being blocked.

**Most of the data breaches that were found (some are still live breaches as of publishing) are caused by a sloppy and dangerous growth hack that is used to improve attribution tracking for analytics tools and used to optimize and segment retargeting advertising campaigns.**

Last build: 24 days ago - v9 supports multiple inputs and a Node API allowing you to program with CyberChef!

Recipe	Input
<b>From Base64</b> Alphabet A-Za-z0-9_. <input checked="" type="checkbox"/> Remove non-alphabet chars	ZWR3YXJ...tYWlsLmNvbQ
<b>Decode text</b> Encoding IA5 German (7-bit) (20106)	
	<b>Output</b> ...sgmail.com



Several of the breaches involve “plain text” user emails — this is when you can literally read the email in the URL with minimal changes/encodings.

Some of the breaches involve a form of plain text known as “base64 encoding” — in short, base64 is a programming language feature that is NOT a form of encryption and provides no user protections. A base64 string can be decoded through many tools, and there is even a free service from the nice folks at GCHQ called [CyberChef](#) for parsing custom base64 encodings.

Before I get into the details about how this breach happens, and the specific circumstances surrounding the examples, I want to briefly acknowledge and give credit to the team at Wish.com for how quickly they changed their entire email architecture after being informed of their breach — in less than 72 hours Wish had completely rebuilt their email architecture and they had built a completely new auto-login flow via email.

I believe the Wish.com breach was the largest out of all the examples in this research, and it lasted over a year and likely involved hundreds of millions of user emails in a base64 plain-text format being shared with analytics and advertising companies, but their work to quickly escalate the problem, realize the scope, and then pull the trigger to rebuild their systems was a dramatically better response than how other organizations handled these reports. I believe Wish and all organizations in this research should be requesting deletion of user emails from any 3rd party logs held by external advertising and analytics companies, but it appears no organization has submitted this request to their partners, even after being notified of their breaches.

For the most part, most of these user email data breaches are **still live** as of publishing this research — and in this research I’ll show you how to “breach yourself” by just using current

research — and in this research I'll show you how to "breach yourself" by just using current website signup flows and other normal website features on the specific websites in question.

I also want to thank [Eliya Stein](#) at Confiant.com for being a sounding board on these technical issues, and helping to provide an additional vet and other important context around the Wish.com breach (those details below).

---

### **3rd Party Javascript Collects a "Referrer" URL Field, Which Can Leak User Data and Email Addresses from a Website**

This research is focused on a specific type of user data breach that occurs due to how Javascript collects data on a website. When a user loads a web page, the URL that they are visiting, along with any URL parameters (extra tracking codes appended after a "?" in a URL) are shared with any advertising or analytics companies through the javascript code on that page and through a technical browser transmission "request header" known as a "Referrer" field.

---

### **Quibi Leaking New User Emails on Email Confirmation Webpage to Advertising and Analytics Companies**

*(Pre-Publishing Note: Quibi reached out hours before publication with an apology and several sentences explaining "how this happened" and what they were doing to fix it. Apparently they no longer leak user emails — I have doubts about some of their statements and will let other reporters publish their remarks)*

When you install the Quibi app, you are asked to submit an email to create your account, and then emailed a confirmation link that must be clicked to confirm the account. When a user clicks this email confirmation link, their email address is appended into the URL they are clicking in plain text, and sent to 3rd party advertising and analytics companies.

Quibi was informed of their user email data breach on April 17, 2020 but haven't responded to the details other than through their automated customer support system.

Here's a screen shot showing the Quibi New User Email Verification Webpage URLs and how this page was built to leak the user email in plain text to advertising and analytics companies:

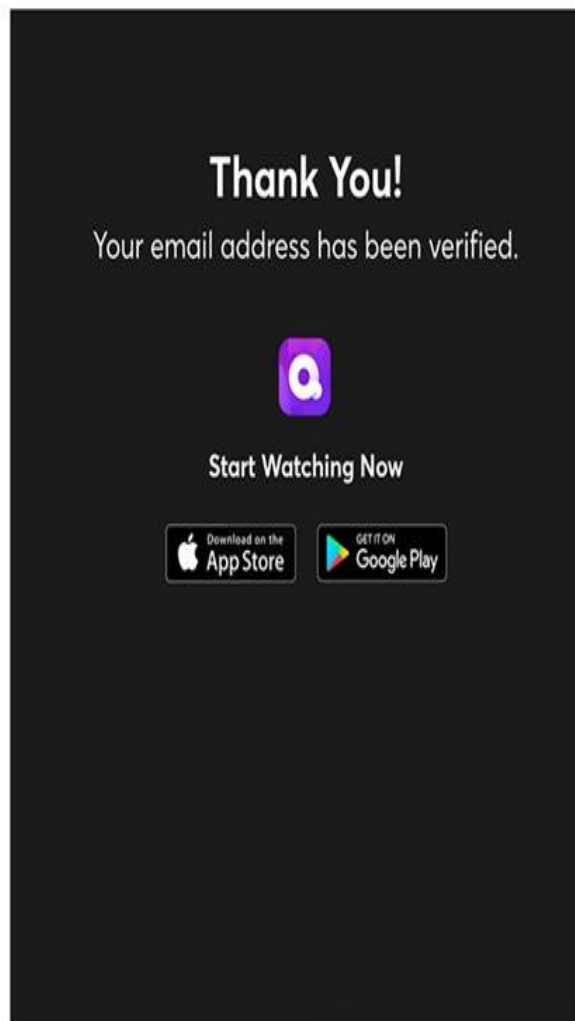
page was built to leak the user email in plain text to advertising and analytics companies.

## Leaking User Emails via URL Parameters

 [https://quibi.com/email\\_verified/?email=quibi%40victorymedium.com&message=This%20URL%20.....](https://quibi.com/email_verified/?email=quibi%40victorymedium.com&message=This%20URL%20.....)



After installing the Quibi mobile app, and entering their email, a person is asked to confirm their email address. After the person clicks to confirm their Quibi email, their email address is appended into the URL of the webpage they visit in plain text and then shared with 3rd party advertising and analytics companies.



That same "Email verification" webpage above from Quibi sends the data to advertising and

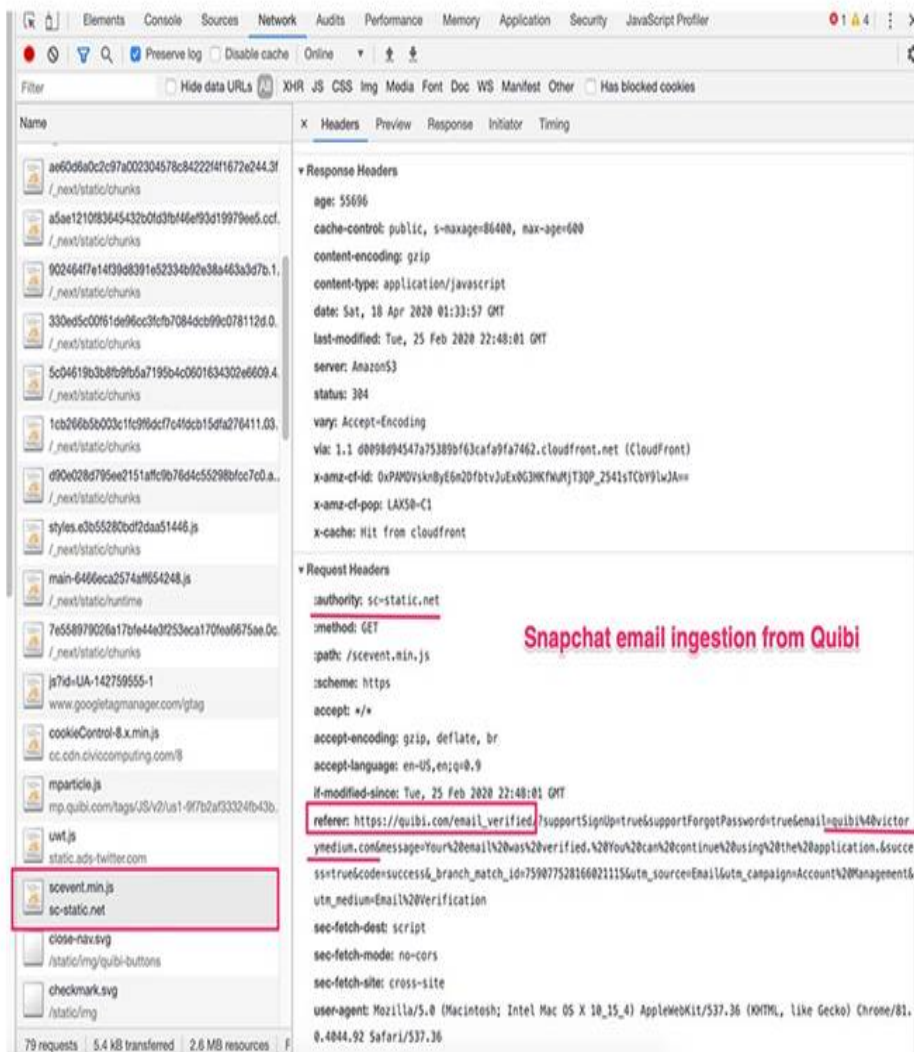
analytics companies through the referrer fields in the request headers — a screen shot below includes the user email sync from Quibi to Snapchat's *sc-static.net* advertising endpoint.

# User Emails in URL Parameters Passing to 3rd Party Analytics & Advertising Companies via Request Headers

 [https://quibi.com/email\\_verified/?email=quibi%40victorymedium.com&message=This%20URL%20....](https://quibi.com/email_verified/?email=quibi%40victorymedium.com&message=This%20URL%20....)



After installing the Quibi mobile app, and entering their email, a person is asked to confirm their email address. After the person clicks to confirm their Quibi email, their email address is appended into the URL of the webpage they visit in plain text and then shared with 3rd party advertising and analytics companies.



The screenshot shows the Network tab in a browser's developer tools. The selected resource is `sc-static.net`. The Request Headers section is expanded, showing the following information:

- authority: sc-static.net
- method: GET
- path: /scevent.min.js
- scheme: https
- accept: \*/\*
- accept-encoding: gzip, deflate, br
- accept-language: en-US,en;q=0.9
- if-modified-since: Tue, 25 Feb 2020 22:48:01 GMT
- referrer: [https://quibi.com/email\\_verified/?email=quibi%40victorymedium.com&message=Your%20email%20was%20verified.%20You%20can%20continue%20using%20the%20application.%20success=true&codesuccess&branch\\_match\\_id=75907752816602115&utm\\_source=Email&utm\\_campaign=Account%20Management&utm\\_medium=Email%20Verification](https://quibi.com/email_verified/?email=quibi%40victorymedium.com&message=Your%20email%20was%20verified.%20You%20can%20continue%20using%20the%20application.%20success=true&codesuccess&branch_match_id=75907752816602115&utm_source=Email&utm_campaign=Account%20Management&utm_medium=Email%20Verification)
- sec-fetch-dest: script
- sec-fetch-mode: no-cors
- sec-fetch-site: cross-site
- user-agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10\_15\_4) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/81.0.4044.92 Safari/537.36

A red box highlights the `referrer` header, and a red text annotation "Snapchat email ingestion from Quibi" points to it.

Here's a screen shot of the Twitter request as it receives the user email in the URL:

The image shows a browser window with a dark-themed website. The website content includes a 'Download Now' button, a 'Thank You!' message stating 'Your email address has been verified.', and a 'Start Watching Now' button. Below these are 'Download on the App Store' and 'GET IT ON Google Play' buttons. The browser's developer tools are open to the Network tab, showing a list of requests. The selected request is to 't.co/i' with a path containing user and transaction IDs. The request headers show an authority of 't.co' and a path that includes the user's email address. The query string parameters include 'p\_id: Twitter', 'p\_user\_id: 0', 'txn\_id: 03098', and 'events: [{"pageview", null}]'. A red box highlights the path parameter, and a red arrow points to the text 'Twitter ads sync' on the right side of the image.

Thank You!  
Your email address has been verified.

Start Watching Now

Download on the App Store

GET IT ON Google Play

Network Inspector:

- Name: t.co/i
- Path: /adsct?p\_id=Twitter&p\_user\_id=0&txn\_id=03098&events=%5B%5B%22pageview%22%2Cnull%5D%5D&=0&tw\_order\_quantity=0&tw\_iframe\_status=0
- Request Headers:
  - authority: t.co
  - method: GET
  - accept: image/webp,image/apng,image/\*,\*/\*;q=0.8
  - accept-encoding: gzip, deflate, br
  - accept-language: en-US,en;q=0.9
  - cookie: muc=c39edcac-be7a-46b1-808f-056d64b74db6
  - referrer: https://quibi.com/email\_verified/?supportSignup=true&supportForgotPassword=true&email=quibiumedia.com&message=Your%20email%20was%20verified.%20You%20can%20continue%20using%20the%20app%20as=true&code=success&branch\_match\_id=759077528166021115&utm\_source=Email&utm\_campaign=Account
  - sec-fetch-dest: image
  - sec-fetch-mode: no-cors
  - sec-fetch-site: cross-site
  - user-agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10\_15\_4) AppleWebKit/537.36 (KHTML, like Gecko) 0.4044.92 Safari/537.36
- Query String Parameters:
  - p\_id: Twitter
  - p\_user\_id: 0
  - txn\_id: 03098
  - events: [{"pageview", null}]
  - tw\_sale\_amount: 0
  - tw\_order\_quantity: 0
  - tw\_iframe\_status: 0



Here's what one of the email confirmation links looks like:

[https://quibi.com/email\\_verified/?](https://quibi.com/email_verified/?)

[email=quibi%40victorymedium.com&message=This%20URL%20can%20be%20used%20only%20once&success=false&\\_branch\\_match\\_id=759077528166021115&utm\\_source=Email&utm\\_campaign=Account%20Management&utm\\_medium=Email%20Verification#](https://quibi.com/email_verified/?email=quibi%40victorymedium.com&message=This%20URL%20can%20be%20used%20only%20once&success=false&_branch_match_id=759077528166021115&utm_source=Email&utm_campaign=Account%20Management&utm_medium=Email%20Verification#)

When initially tested, the user email address in plain text format was transmitted to:

1) *Google's DoubleClick.net endpoint*

2) *Google's updated ads endpoint @ [google.com](https://google.com)*

3) *Google Tag Manager (and therefore potentially custom tags could fire for specific visitors/geos/URL params, thus leaking this to more companies)*

4) *Twitter ads endpoint*

5) *Snapchat ads endpoint & the [tr.Snapchat.com](https://tr.snapchat.com) subdomain*

6) *Google Cloud infrastructure via [cloudfunctions.net](https://cloudfunctions.net)*

7) *CivicComputing.com, which redirects to <https://www.civicuk.com/> and appears to be a company based in the United Kingdom.. this raises big GDPR red flags....*

8) *Facebook events / custom audiences for ads*

9) *Google ads conversion pixel*

10) *Twitter ads conversion pixel*

11) *Google Analytics*

12) *Facebook analytics, Google Analytics, Twitter analytics (they fire at the end of the page load again)*

 Please verify your email address  

Quibi <help@quibi.com>

to quibiapril26test ▾



## Confirm Your Email

Thanks for signing up for Quibi. A whole world of quick bite entertainment awaits you.

Please take a moment to confirm your email to better secure your account.

Just hit the link below and you'll be all set.

[CONFIRM EMAIL](#)

Thanks!  
Team Quibi



[help@quibi.com](mailto:help@quibi.com)

The Quibi new account email confirmation flow was tested again on April 26, 2020 and it was confirmed that the user email is still being appended to the email confirmation page URL in plain text and leaked to 3rd party advertising and analytics companies.

Since the original test, several new advertising companies were found receiving the user data including LiveRamp.com, SkimAds, and Tapad — it seems likely that numerous ad tech orgs have been syncing the Quibi new user emails and the list included here could be incomplete.

*Quibi's user data breach is one of the most egregious in this research, because they are a new and extremely well-funded organization and were launched well after both GDPR and CCPA went into effect. In 2020, no new technology organizations should be launching that leaks all new user-confirmed emails to advertising and analytics companies — yet that's what Quibi apparently decided to do.*

Out of all the data breaches in this research, the Quibi research is the hardest to swallow due to how new this organization is, and how much money they had to push into their marketing and advertising to grow new users — it's an extremely disrespectful decision to purposefully leak all new user emails to your advertising partners, and there's almost no way that numerous people at Quibi were not only aware of this plan, but helped to architect this user data breach.

It's 2020, and this type of growth-hack needs to stop being green lit. Quibi needs to explain to their users why this was done and why it hasn't been changed even after being notified...

---

### **The Biggest Breach: Wish.com Likely Leaked Hundreds of Millions of User Emails for Over a Year, With the User Emails Encoded into Base64 Strings**

From July 2018 until January 2020 when this research was initially shared with Wish.com, Wish transmitted user emails to at least Google, Facebook, Pinterest, Criteo, PayPal and Stripe, and potentially other companies.

In July 2018, Wish.com deployed code that started their user email breaches — this was tracked due to user emails in base64 format being cached in systems like URLscan.io — the Wish.com developers deployed code that started to encode users emails in base64 plain text and then append that string into URLs sent to users via email in a URL parameter named “ee” — when users clicked on any marketing emails from Wish, their email was appended to the URL for any page/product-page they clicked from the marketing emails from Wish, and then when the user

visited the Wish page, their email in base64 format was transmitted to Wish's 3rd party advertising and analytics partners.

urlscan.io
Sponsored by SecurityTrails
34 running

## www.wish.com

54.183.78.202

Submitted URL: [http://www.wish.com/home?utm\\_campaign=5b515ddd36f7ae0e5cfa361c&utm\\_medium=email&utm\\_source=New+User+Mobile+Push&recvid=5b34ee2fc74c9712f5c20c66](http://www.wish.com/home?utm_campaign=5b515ddd36f7ae0e5cfa361c&utm_medium=email&utm_source=New+User+Mobile+Push&recvid=5b34ee2fc74c9712f5c20c66)

Effective URL: [https://www.wish.com/home?utm\\_campaign=5b515ddd36f7ae0e5cfa361c&utm\\_medium=email&utm\\_source=New+User+Mobile+Push&recvid=5b34ee2fc74c9712f5c20c66](https://www.wish.com/home?utm_campaign=5b515ddd36f7ae0e5cfa361c&utm_medium=email&utm_source=New+User+Mobile+Push&recvid=5b34ee2fc74c9712f5c20c66)

Submission: On July 23 via manual (July 23rd 2018, 2:51:20 pm) from CA

Summary
HTTP 81
Links 5
Behaviour
iOCs
Similar 126
DOM
Content
API

### Summary

This website contacted 13 IPs in 4 countries across 10 domains to perform 81 HTTP transactions.

The main IP is 54.183.78.202, located in San Jose, United States and belongs to AMAZON-02 - Amazon.com, Inc., US. The main domain is www.wish.com.

TLS certificate: Issued by Go Daddy Secure Certificate Authority... on January 5th 2017. Valid for: 2 years.

The main domain was scanned 6297 times on urlscan.io Show Scans 6297

126 structurally similar pages on different IPs, domains and ASNs found Show Scans 126

Verdict: No classification

Google Safe Browsing: Clean (Current Classification)

### Additional live information

Current DNS A record: 3.221.61.25 (AS14618 - AMAZON-AES, US)

Domain created: January 2nd 1995, 21:00:00 (UTC)

Domain registrar: MarkMonitor Inc.

### Domain & IP information

IP/ASNs	IP Detail	(Sub)Domains	Domain Tree	Links	Certificates
	IP Address	AS Autonomous System			
4	54.183.78.202	16509 (AMAZON-02 - Amazon.com)			
1	2a00:1450:4001:81d::200a	15169 (GOOGLE - Google LLC)			
21	2600:9000:20bb:c00:e37:e948:2981	16509 (AMAZON-02 - Amazon.com)			
2	2a00:1450:4001:806::200a	15169 (GOOGLE - Google LLC)			
2	2a00:1450:400c:00::9d	15169 (GOOGLE - Google LLC)			
3	2a03:2880:f11c:8186:face:b00c:0:50fb	32934 (FACEBOOK - Facebook)			
3	2a03:2880:f01c:8012:face:b00c:0:3	32934 (FACEBOOK - Facebook)			
2	2a00:1450:4001:81d::200e	15169 (GOOGLE - Google LLC)			
1	2a00:1450:4001:81d::2004	15169 (GOOGLE - Google LLC)			
1	2a00:1450:4001:81d::2003	15169 (GOOGLE - Google LLC)			

### Screenshot

Live screenshot Full Image

### Detected technologies

- TornadoServer (Web Servers) Website
- Criteo (Advertising Networks) Website
- Facebook (Widgets) Website
- Google Analytics (Analytics) Website
- Google Font API (Font Scripts) Website
- Modernizr (JavaScript Libraries) Website
- jQuery (JavaScript Libraries) Website
- jQuery UI (JavaScript Libraries) Website
- webpack (Miscellaneous) Website

### Stats

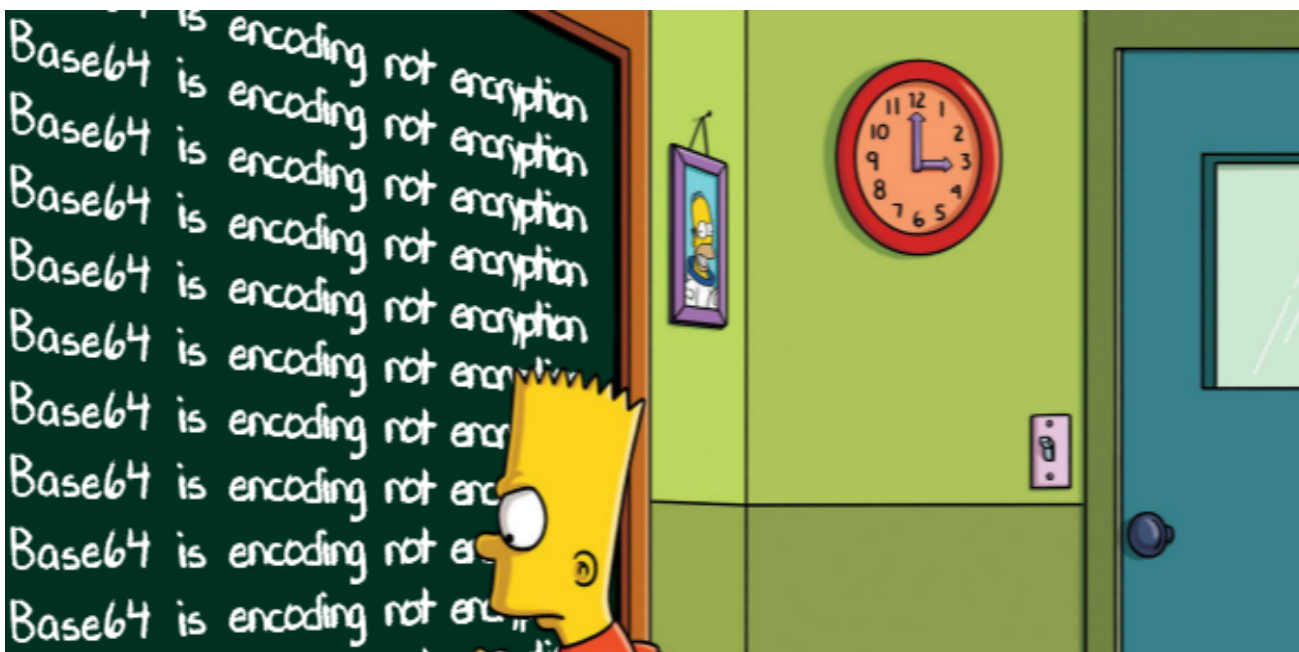
81	10	0	84%	79%
Requests	Ad-blocked	Malicious	HTTPS	IPv6
10	14	13	4	1,740
Domains	Subdomains	IPs	Countries	Transfer
3,550	9			
Size	Cookies			

A URLScan.io capture of a Wish.com page view from July 23, 2018 that captured a user's "ee" parameter and their email encoded in base64 plain text. The ee string is blurred for privacy due to it containing the user email.

Approximately ~72 hours after being informed of this research Wish rebuilt their entire email architecture and stopped appending the "ee" parameter with base64 user encoded emails into marketing emails. It does not appear Wish has informed their users of this user email breach, but they did take the issue very seriously and quickly agreed that the base64 email encoding was a practice they weren't going to continue. Minimal comments from Wish were received after the research was submitted, but they were more professional than the most organizations when confronted with this type of research.

Due to Wish.com being a massive multi-billion dollar company, [who in 2015 was Facebook and Instagram's #1 app advertiser over Christmas, spending upwards of \\$100 million](#), and their previous valuations, it's likely that tens of millions, if not hundreds of millions of user emails were pushed through the "ee" parameter and leaked to advertising and analytics companies.

**To repeat: from 2018–2020, most if not all of the Wish.com marketing emails appended user emails in a format that, if the user clicked on the email and they were using a browser that didn't block 3rd party javascript, then that user had their email in base64 plain text format leaked to 3rd party advertising and analytics companies including Google, Facebook, Pinterest, Criteo, PayPal and Stripe, and potentially other companies.**





*It didn't take long for Wish to send me a marketing email and I was able to confirm the finding immediately.*

*When a subscriber clicks a link in the email, the destination URL has several parameters in the querystring, including "ee" which is paired with the recipients base64 encoded email address.*

*This means that this entire URL, including the ee parameter can potentially be leaked to any 3rd party resources that are loaded on the page.*

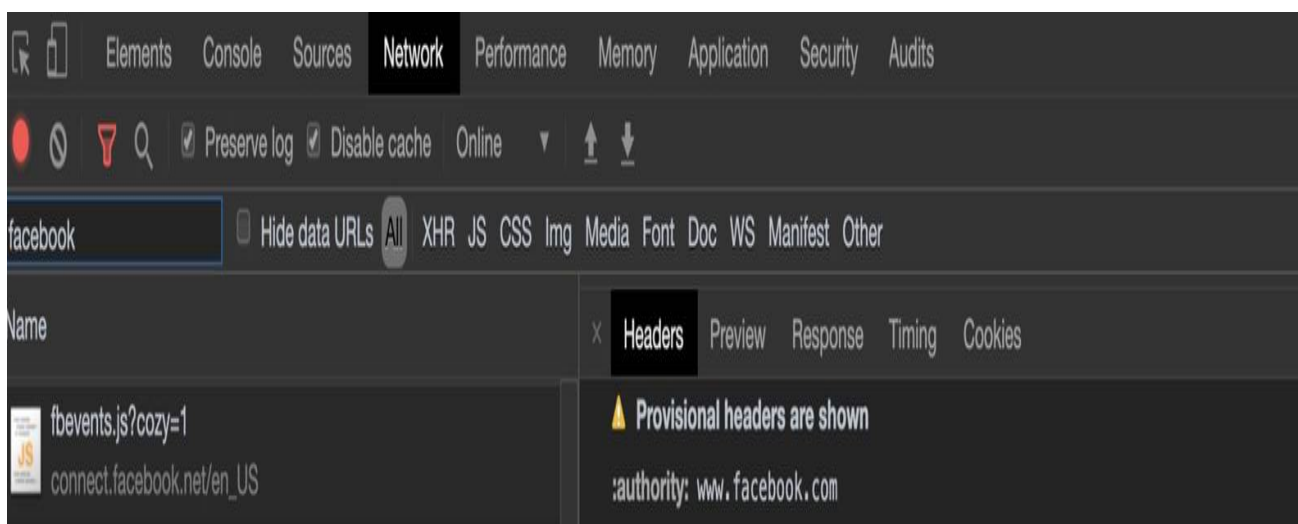
*In this case, I can confirm that it's being leaked at least to Facebook, Google, Pinterest, and Critico per Zach's observation, but also additional 3rd parties including Paypal & Stripe.*

*From my observations, it looks like these are mostly tracking endpoints and not actual ad slots on these pages. If they ever introduce display ads connected to rtb on these pages, then the impact of this leak has the potential to be quite large.*

*I'm not in a great position to comment on GDPR implications, because that's a little bit outside of my expertise, but for sure it's a terribly bad practice to pass around PII in plain text in the URL like this, and I do consider base64 encoding to be plain text.*

*One thing that we're not able to observe is if and how this data is being abused, but any ad tech company with integrity should scrub data like this if they recognize it as PII.*

*I've included a screenshot of the parameter being leaked to Facebook via the referer.*



<pre> sdk.js?cozy=1 connect.facebook.net/en_US tr?id=1491478797738271 www.facebook.com 1491478797738271?v=2.9.15&amp;r=stable connect.facebook.net/signals/config sdk.js?hash=42272dd37ca5caf2a2797a1147783a65&amp;ua=modern_es6 connect.facebook.net/en_US ?id=1491478797738271&amp;ev=PageView&amp;dl=https%3A%2F%2F...4772. www.facebook.com/tr status?client_id=227791440613076&amp;input_token&amp;origi...17aa5ab46b8. www.facebook.com/x/oauth ?id=1491478797738271&amp;ev=Microdata&amp;dl=https%3A%2F%2F...15789. www.facebook.com/tr tr?id=1491478797738271 www.facebook.com ?id=1491478797738271&amp;ev=PageView&amp;dl=https%3A%2F%2F...4772. www.facebook.com/tr </pre>	<pre> :method: GET :path: /tr?id=1491478797738271 :scheme: https accept: image/webp, image/apng, image/*, /*; q=0.8 accept-encoding: gzip, deflate, br accept-language: en-US, en; q=0.9 referer: https://www.wish.com/feed/blitz_buy__tab?utm_campaign=2020-01-13_WELCOM TION_EDUCATION_LARGE_46f7b96d91c642aba342027201988e47&amp;verification_code=369ef2 28376a9a571154922a&amp;user_id=5e1ceddd17aa5ab46b8beb28&amp;uid=46f7b96d91c642aba3420 e47&amp;cmpqid=2020-01-13_WELCOME_VARIATION_EDUCATION_LARGE_46f7b96d91c642aba3420 e47&amp;see=7[REDACTED]onQuY29t&amp;email_section=user_edu_v3_big-SPIN_NOW&amp;utm_sou OME_VARIATION_EDUCATION_LARGE&amp;utm_medium=email&amp;recvuid=5e1ceddd17aa5ab46b8beb2 sec-fetch-mode: no-cors sec-fetch-site: cross-site user-agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_14_6) AppleWebKit/537.36 like Gecko) Chrome/79.0.3945.117 Safari/537.36 </pre>
<pre> 14 / 505 requests 206 KB / 15.2 MB transferred 773 KB / 18.8 MB resource </pre>	<pre> ▼ Query String Parameters view source view URL encoded </pre>

Wish.com, like the other organizations included in this research, would ideally submit deletion requests to all of their advertising and analytics partners who received data during this period with requests to those partners to delete the request logs containing base64 user emails.

Ideally, Wish.com would also inform their own users about this breach too.

### JetBlue.com Still Leaking New User Emails to Advertising and Analytics Partners

Jetblue has known about their ongoing data breach since March 2020 and sent several email responses after being shown this research, but still haven't made any changes to their website or the ongoing leak of new user emails to 3rd party advertising and analytics companies



the ongoing leak of new user emails to 3rd party advertising and analytics companies.

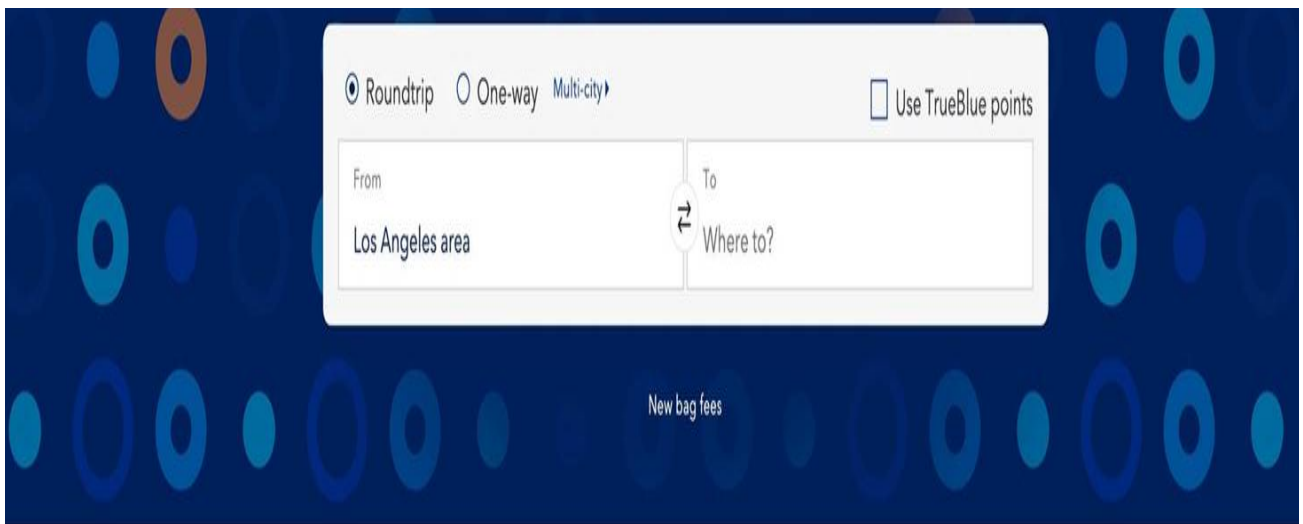
After being informed of the leak, Jetblue stated they would never do what they are doing because it would be against the law (\*NOTE: JetBlue wrote “Federal Passenger Privacy Act” in their response — this may be a reference to a 1974' privacy bill — or as [this Berkley Law paper on page 14 indicates](#), JetBlue has sent this statement before and is possibly referencing a nonexistent law), writing this in a response recently:

*We regret to hear of any disappointment you experienced when creating a TrueBlue account. We can assure you we don't share your information. **The Federal Passenger Privacy Act\* strictly prohibits the release of any information regarding our customers** or their travel to any other party. We even require specific security information to verify the identity of our customers before we're able to discuss their own information.*

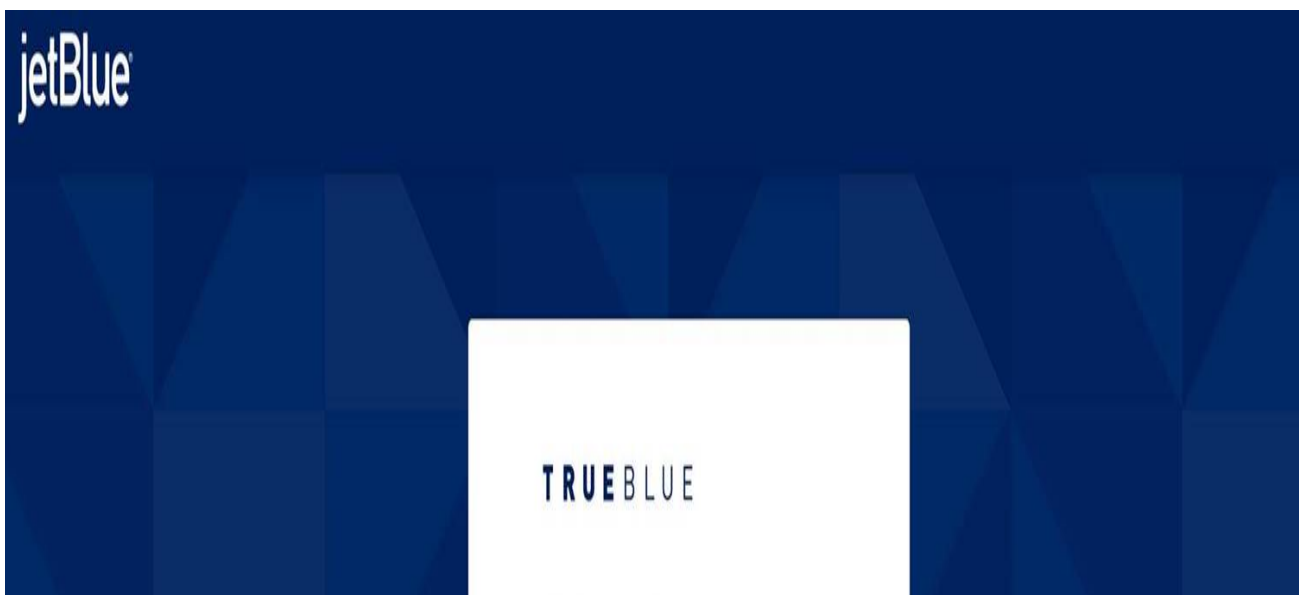
These details were tweeted out in March and then emailed to JetBlue, which still didn't have an impact to get them to change:

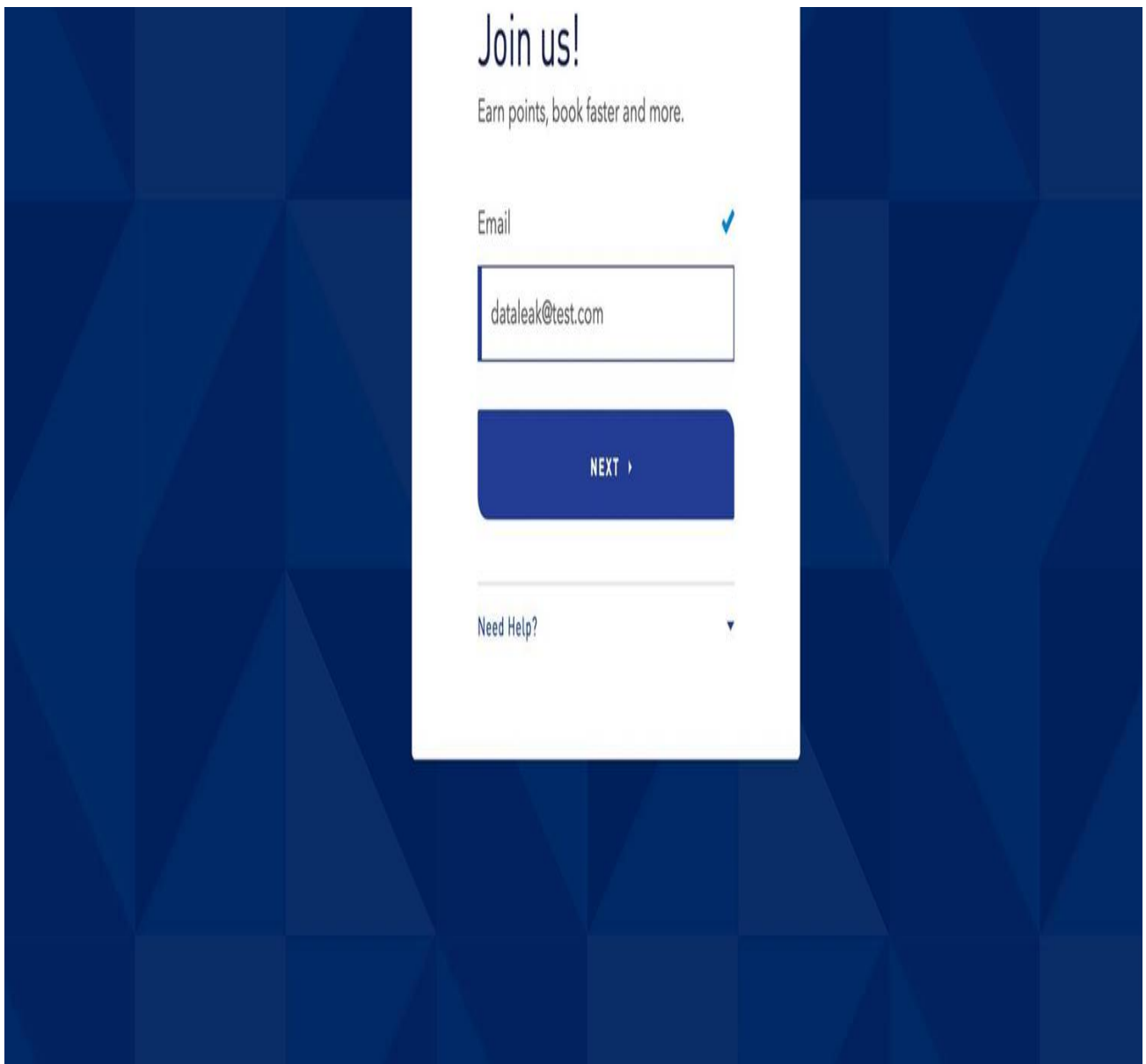
Here's the flow of how all new JetBlue users are having their email addresses leaked to 3rd party advertising and analytics companies, in violation of the Federal Passenger Privacy Act\* (and potentially other privacy laws)— step one, click “Join” in the menu bar on Jetblue.com from the homepage or any page on the site:





Then, you'll be prompted to enter your email — whatever you enter here, when you click the next step, your email is passed into the URL and subsequently leaked to the 3rd party advertising and analytics companies:





Here's a screen shot of the next step, with the user email being passed into the URL — the icon showing "45" is the Ghostery.com count of advertising & analytics companies receiving data on the webpage — it's not a complete list but this shows dozens of companies are receiving user emails from the current JetBlue.com data leak.





### What's your full name?

Please enter it as it appears on your government-issued ID.

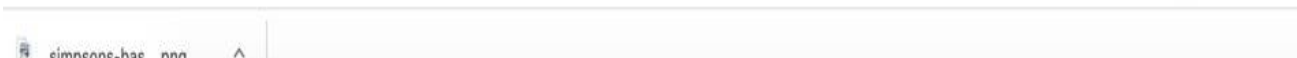
[+ Add a Title \(optional\)](#)

First Name

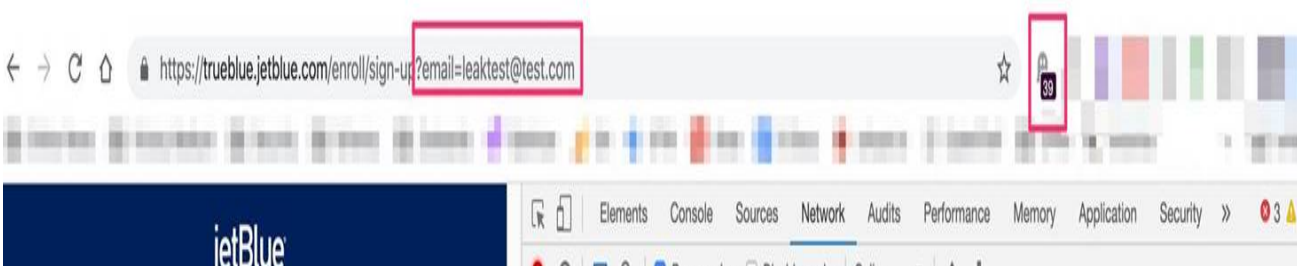
Middle Name(s) (optional)

Last Name

[+ Add a Suffix like Jr. or Sr. \(optional\)](#)



Here's a screen shot from a previous test showing one of the advertising pixels firing and how it receives the user data through the request headers (notice only 39 pixels were tracked on this page last month, April's test showed 45):





archived version

here: <http://web.archive.org/web/20190709195758/https://trueblue.jetblue.com/enroll/join-us>

Here's a screen shot of the July 2019 version of the JetBlue.com user account creation 2-step form that leaks user emails on the 2nd step:



Jetblue.com has been leaking user emails for about nine months for people creating new accounts.... it's unclear when JetBlue will update this but they have rejected the research even though being informed on multiple occasions.

---

### **The Bezos-Schmidt-Funded KongHQ.com (Formerly Known as Mashape) Using Common 2-Step Form That Leaks on the 2nd Step**

The company formerly known as Mashape, now known as KongHQ, was founded in 2007 and received \$1.5 million in seed funding in 2011 from a [round of investors that included Jeff Bezos and Eric Schmidt](#) through Innovation Endeavors.

KongHQ has a 2-step signup form similar to the JetBlue leak, but the KongHQ form starts on their homepage. When a user puts their email in the form on the homepage and hits enter, their email is immediately pushed into the URL bar and then transmitted to 3rd party advertising and analytics partners.



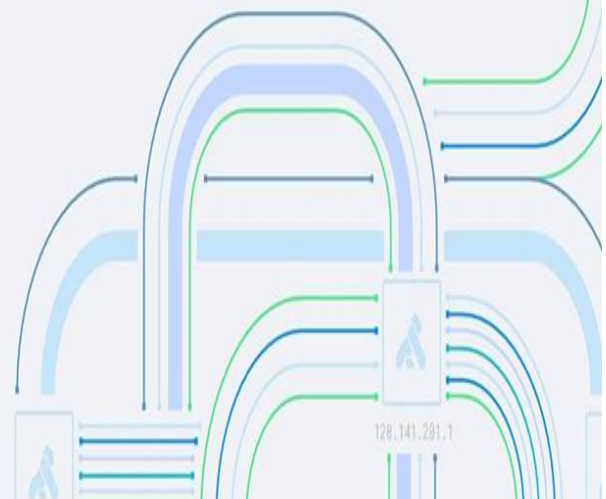
Products Solutions Open Source Docs Resources Company

Request De

NEWS Webinar - Microservices API Management: Best Practices for the Enterprise →

## Next-Generation API Platform for Multi-Cloud and Hybrid Organizations

Achieve architectural freedom by connecting all your microservices and APIs natively within and across clouds, Kubernetes, data-centers and more.



The image is a screenshot of the KongHQ website. At the top left, there is a search bar containing the email address 'breachme@test.com' and a green button labeled 'Request Demo'. The background features a complex network diagram with blue and green lines representing data flow, and several server icons with the Kong logo. Below the header, there are three distinct feature cards:

- Sub-Millisecond Latency:** Accompanied by a diagram of two boxes connected by a dense green line of vertical bars. Below the card, it says 'Accelerate your services. Dramatically'.
- 25K+ TPS / Node:** Accompanied by a diagram of multiple horizontal lines representing data streams entering a box. Below the card, it says 'Easily handle peak volume. Scale'.
- Platform Agnostic:** Accompanied by a diagram of a central box with lines extending to various points, representing different platforms. Below the card, it says 'Radically extensible. Connect APIs across'.

KongHQ was informed of this breach back in February 2020 but still haven't made any changes to their website and 2-step form — their response was similar to JetBlue in totally ignoring the issue.

In the original tests KongHQ transmitted data to:

- Google
- LinkedIn Twitter
- Facebook
- Drawbridge
- Mixpanel
- CrazyEgg



- New Relic
- Pardot
- Wistia

You can breach your own email address right now by filling out the form on the homepage of KONGHQ.com, but user beware!

After clicking “Request Demo” on the homepage, you are transmitted to the 2nd step of the form, with your email address added into the URL bar to auto-fill the form...

Request Demo

Schedule a demo today to see how the Kong Enterprise API Platform can help your organization more easily secure, manage, and optimize the performance of your microservices and APIs.

TRUSTED BY

YAHOO! JAPAN SOULCYCLE FUJITSU

Just a few more details needed to complete your demo request

First name

Last Name

Email

Phone

Job Title

Company

How can we help you?



Request Demo

© Kona Inc. 2020

Unfortunately, anywhere you can find a 2-step signup form where the 2nd step has some form of autofill, many of those systems are being built with insecure technology and sometimes the user emails are purposefully leaked to optimize retargeting advertising campaigns or improve analytics attribution data.

---

### **Democratic Data Broker NGPVan.com / EveryAction.com (& Their Clients) have been Pushing User Emails into Google Analytics & Other Systems for Years**

NGPVan.com/EveryAction.com are owned by the same company and provide a wide range of CRM/marketing services for political and nonprofit clients. These platforms have an enormous range of features — and similar to the Mailchimp-Mandrill email breach described in this research, NGPVan created a legacy URL field for “emailAddress” that is appended into URLs, mostly on unsubscribe pages, and this can lead to NGPVan/Everyaction clients leaking user emails to 3rd party advertising and analytics companies.

A typical NGPVan unsubscribe URL that has the user email in it, looks like this (The email is appended at the back of the URL):

email.everyaction(.)com/unsubscribeUnique/3d7e893c-921a-ea11-828b-

2818784d6d68/ad179bfc-9c1a-ea11-828b-2818784d6d68?

nvep=ew0KICaiVGVuYW50VXJpljogIm5ncHZhbjovL3ZhbI9FQS9FQTAwMS8xLzU1NDA3liwNCiAglkRpc3RyaWJ1dGlvbIvuaXF1ZUIkljogImFkMTc5YmZlTlJmWEtZWExMS04MjhiLTl4MTg3ODRkNmQ2OCIsDQogICJFbWFpEFkZHJlc3MiOiAiAiemFjaEB2aWN0b3J5bWVkaXVtLmNvbSINCn0%3D&hmac=

73-vUhguqpitg-

5DybUg7PmTNqxOTTLInLe8CYE0y0=&id=107423875&emailAddress=cats%40victorymedium.co

m



site:everyaction.com inurl:emailAddress



Settings

Too

About 234 results (0.18 seconds)

email.everyaction.com › unsubscribeUnique › emailAddr...

## To unsubscribe or change your email preferences ... - Français

No information is available for this page.

Learn why

email.everyaction.com › unsubscribeUnique › emailAddr...

## To unsubscribe or change your email preferences ... - Français

No information is available for this page.

Learn why

## <https://email.everyaction.com/unsubscribeUnique/89...>

No information is available for this page.

Learn why

email.everyaction.com › unsubscribeUnique › emailAddr...

## Manage Email Preferences - Français - EveryAction

No information is available for this page.

Learn why

## <https://email.everyaction.com/unsubscribeUnique/96...>

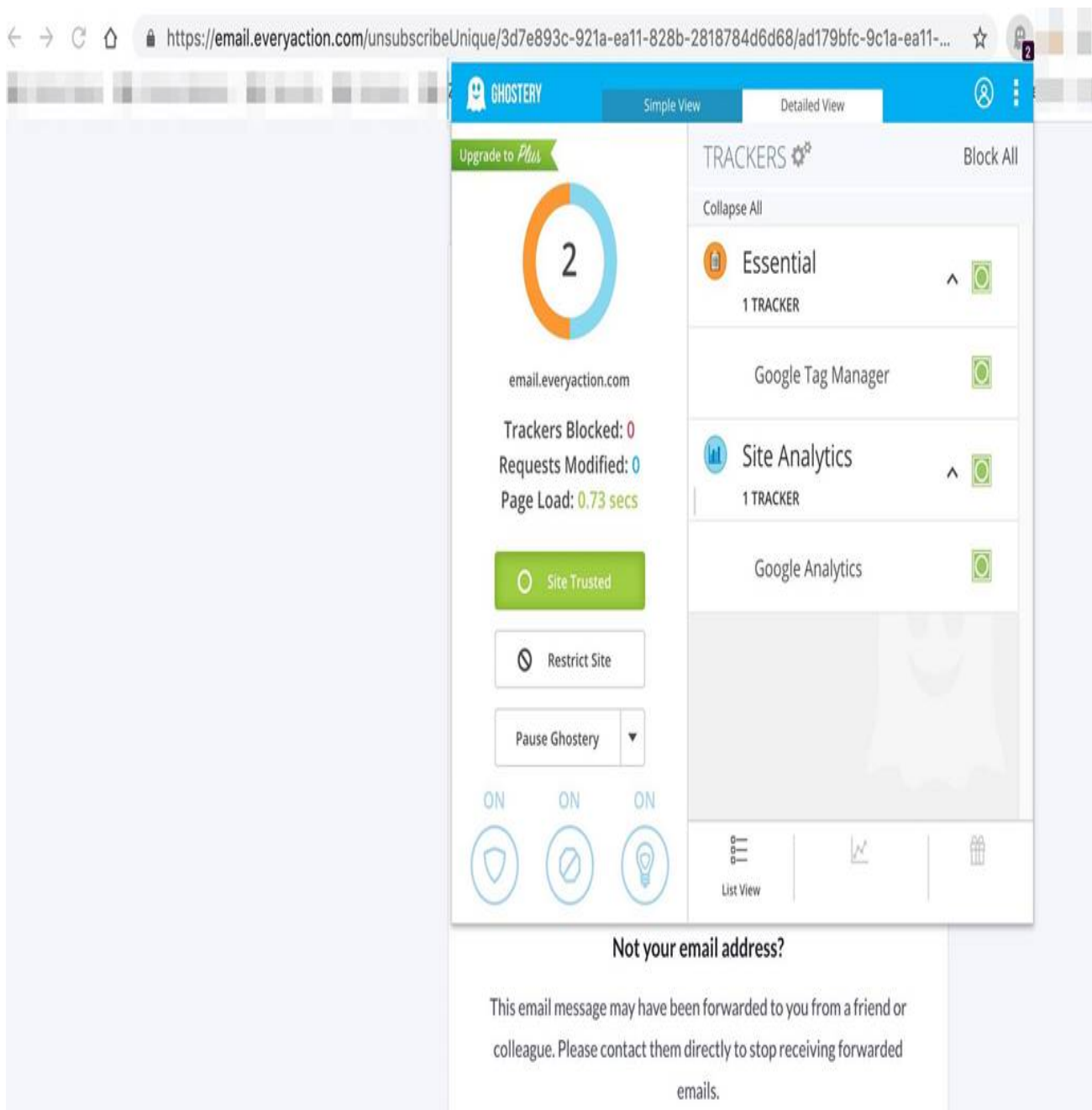
No information is available for this page.

Learn why

<https://email.evervaction.com/unsubscribeUnique/3d7e893c-921a-ea11-828b-2818784d6d68/ad179bfc-9c1a-ea11-...>

Unfortunately, not only are advertising and analytics companies ingesting the user emails on random unsubscribe pages all across the NGPVan client base, but those same URLs with user emails in plain text are also cached in [Google search results](#), [URLscan results](#), and in other repositories of cached user pages across the internet.

The primary company ingesting the user emails from NGPVan clients appears to be Google via their Google Analytics product, but a Microsoft endpoint also receives data. Here's what Ghostery picks up on one of the unsubscribe pages with a user email in plain text in it:



← → ↻ 🏠 🔒 <https://email.evervaction.com/unsubscribeUnique/3d7e893c-921a-ea11-828b-2818784d6d68/ad179bfc-9c1a-ea11-...> ☆ 🏠

**GHOSTERY** Simple View Detailed View

Upgrade to *Plus*

2

email.evervaction.com

Trackers Blocked: 0  
Requests Modified: 0  
Page Load: 0.73 secs

Site Trusted

Restrict Site

Pause Ghostery

ON ON ON

TRACKERS Block All

Collapse All

Essential 1 TRACKER

Google Tag Manager

Site Analytics 1 TRACKER

Google Analytics

List View

**Not your email address?**

This email message may have been forwarded to you from a friend or colleague. Please contact them directly to stop receiving forwarded emails.

Update

© 2020 EveryAction

And then here's an additional screen shot of the actual data transfer, showing that Microsoft is also receiving the user emails through their visualstudio.com endpoints.

The image shows a side-by-side comparison of a web form and its network traffic. On the left is a form titled "Email Preferences for Zach Edwards". It contains a "Subscribed email address" field with the value "cats@victorymedium.com". Below this, there are checkboxes for "Welcome Series" (checked), "Feature Workshops" (unchecked), and "Customer Experience" (checked). At the bottom, there is a "Please unsubscribe" checkbox (unchecked) and a "Not your email address?" link. On the right is a browser's developer tools network tab showing a POST request to "https://dc.services.visualstudio.com/v2/track". The request headers include "Accept: \*/\*", "Accept-Encoding: gzip, deflate, br", and "Accept-Language: en-US,en;q=0.9". The response headers include "Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Name, Content-Type, Sdk-Context", "Access-Control-Allow-Origin: \*", "Access-Control-Max-Age: 3600", "Content-Length: 96", "Content-Type: application/json; charset=utf-8", "Date: Sun, 26 Apr 2020 21:58:05 GMT", "Strict-Transport-Security: max-age=31536000", "X-Content-Type-Options: nosniff", and "x-ms-session-id: AD45A91B-8002-40CD-AB70-91CDE2D98ACB".

Email Preferences for Zach Edwards

Subscribed email address

cats@victorymedium.com

I am interested in the following topics :

Welcome Series

Feature Workshops

Customer Experience

Please unsubscribe the email address below from all future mailings

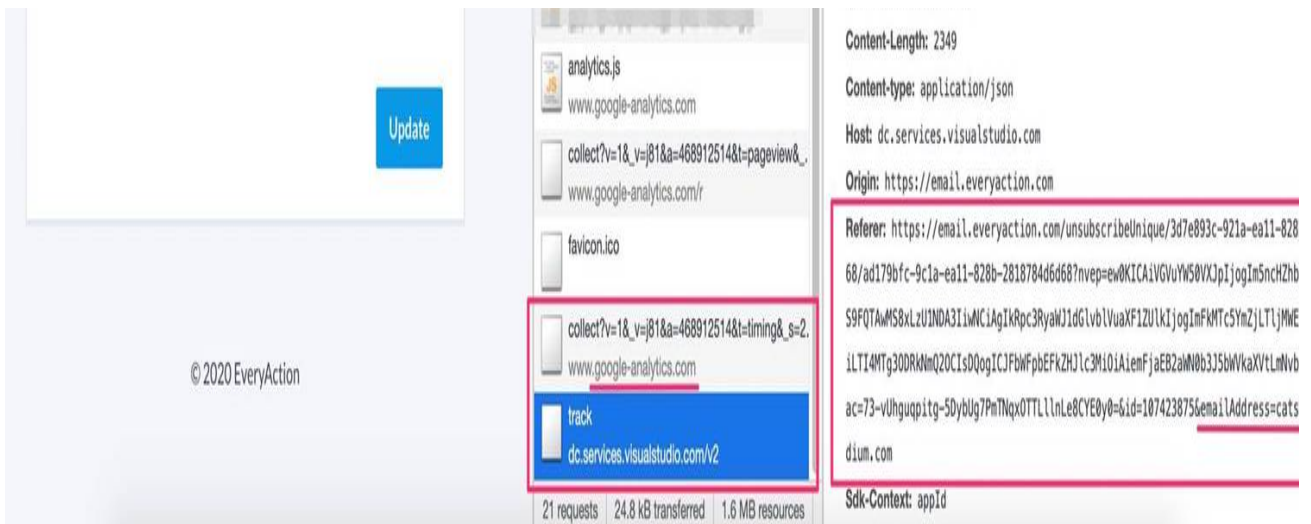
cats@victorymedium.com

Not your email address?

This email message may have been forwarded to you from a friend or colleague. Please contact them directly to stop receiving forwarded emails.

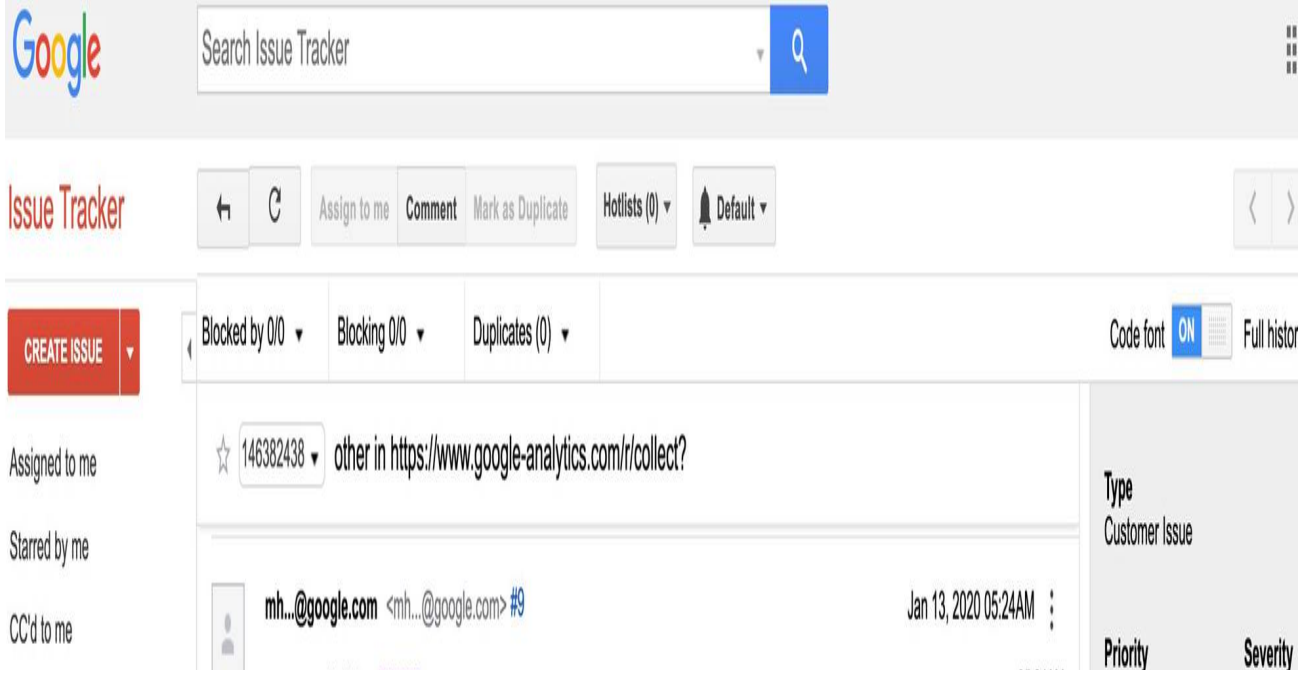
Network traffic details:

- Name: /bundles
- Request URL: https://dc.services.visualstudio.com/v2/track
- Request Method: POST
- Status Code: 200 OK
- Remote Address: 13.86.218.248:443
- Referrer Policy: no-referrer-when-downgrade
- Response Headers: Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Name, Content-Type, Sdk-Context; Access-Control-Allow-Origin: \*; Access-Control-Max-Age: 3600; Content-Length: 96; Content-Type: application/json; charset=utf-8; Date: Sun, 26 Apr 2020 21:58:05 GMT; Strict-Transport-Security: max-age=31536000; X-Content-Type-Options: nosniff; x-ms-session-id: AD45A91B-8002-40CD-AB70-91CDE2D98ACB
- Request Headers: Accept: \*/\*; Accept-Encoding: gzip, deflate, br; Accept-Language: en-US,en;q=0.9; Connection: keep-alive



NGPVan has been appending the user email address to unsubscribe links across their own emails, and client emails for several years — the start date isn't exactly clear but emails from 2018 have this same plain text email.

*Google has also been aware of the NGPVan user email leaks since January, and Google clarified their Google Analytics policy around this type of ingestion, which apparently requires a “mandatory remediation process with the customer where they must stop sending PII to Analytics and ensure all historical PII data must be removed.” This statement was sent by Google on January 13, 2020, and several organizations have been flagged for Google who are sending user emails into Google Analytics, yet it appears no actions have been taken by Google on any of these issues.*



Reported by me

To be verified

---

Bookmark groups

---

Saved searches

---

Hotlists

victorymedium

---

Create hotlist

Create bookmark group

Browse components

-Hotlist: 702027

Hey Zach,

Sorry this took so long.

Passing data that Google could use or recognize as personally identifying an individual, such as email addresses or mobile numbers, through Google Analytics is prohibited by our terms of service, and we take action on any account found doing so intentionally. We take this very seriously and whenever a report is made, we begin a mandatory remediation process with the customer where they must stop sending PII to Analytics and ensure all historical PII data must be removed.

Thank you, for the examples that you brought to our attention, we will work with [redacted] and [redacted] to correct this issue going forward.

If you have any further questions, please do let me know and if I can not answer it, then I will find someone that can.

Regards,

[redacted]

05:24AM

05:24AM

P4 S4

Status Assigned

Assignee Verifier

wo...@google.com --

CC

wo...@google.com  
zach@victorymedium.com

Reporter e-mail

--

Requested charity

--

Found In Targeted To

-- --

NGPVan is currently used on the <https://covid19responsefund.org/> website sponsored by the World Health Organization (WHO), the United Nations Foundation, the Swiss Philanthropy Foundation, and [with supporters including Google, Facebook, Microsoft, and others](#). It's unclear if people who donate money through the NGPVan donation form on the website are subsequently sent emails with their email address leaking via the unsubscribe links, but the form does provide options to join the email lists of several sponsoring organizations...

HELP FIGHT CORONAVIRUS

# COVID-19 Solidarity Response Fund for WHO

The world has never faced a crisis like COVID-19. The pandemic is impacting communities everywhere. **It's never been more urgent to support the global response.** The humanity, solidarity and generosity of people and organizations everywhere is also unprecedented. But we can't stop now.

The World Health Organization (WHO) is leading and coordinating the global effort with a range of partners, supporting countries to prevent, detect, and respond to the pandemic. **Donations support WHO's work, including with partners, to track and understand the spread of the virus; to ensure patients get the care they need and frontline workers get essential supplies and information; and to accelerate research and development of a vaccine and treatments for all who need them.**

Donors from [Japan](#), [Canada](#), or [Europe](#) have the option to make a tax-deductible donation to the Solidarity Fund by giving through our partners

## DONATE NOW

- 1 \$3
- 2 DETAILS
- 3 PAYMENT
- 4 FINISH

I certify that I am not making this donation on behalf of any party, organization, or individual that is in violation of the UN Foundation's Partner Due Diligence Policy.

View the [UN Foundation's Partner Due Diligence Policy](#)

## COMMUNICATION INFORMATION

I would like to receive emails from: (Optional)

- World Health Organization (WHO)
- United Nations Foundation

BACK

DONATE \$3 NOW

NGPVan is a for-profit company and just because their clients are largely political campaigns and nonprofits, it doesn't give them the right to leak user emails to advertising and analytics companies — hopefully this issue is resolved so that as the 2020 campaign heats up and users take advantage of unsubscribe forms more often, those user emails aren't also leaking en-masse to 3rd party companies.

**Growing Child, Popular Magazine for Parents, Leaking Emails on Unsubscribe Page to Google**



## Analytics, Google's DoubleClick, only Google Pixels Receiving Data

GrowingChild.com is a magazine founded in 1971 that describes itself as “serving millions of families in the United States and around the world...”

Unfortunately for the families who have subscribed to GrowingChild.com newsletters and then decided to unsubscribe, their unsubscribe pages print the user email in plain text into the URL and then share the user email in plain text to Google and several Google products including Google Analytics, Google Doubleclick and several other Google advertising endpoints.

The GrowingChild unsubscribe URLs are built like

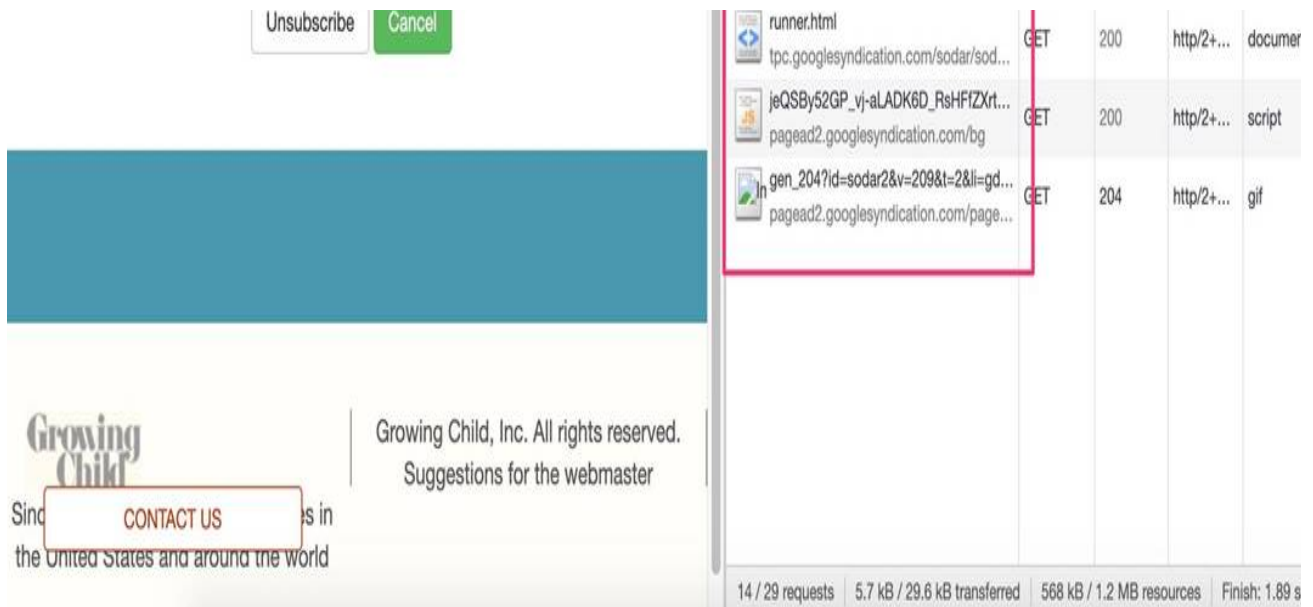
this: [https://growingchild.com/index.php/unsubscribe/unsubscribe.html?](https://growingchild.com/index.php/unsubscribe/unsubscribe.html?email=growingchild@victorymedium.com)

[email=growingchild@victorymedium.com](mailto:growingchild@victorymedium.com) and this leaks as a referrer to the Google pixels here:

The screenshot shows a web browser window with the URL <https://growingchild.com/index.php/unsubscribe/unsubscribe.html?email=growingchild@victorymedium.com> in the address bar. The page content includes the Growing Child logo, a large "UNSUBSCRIBE" button, and a form with the following text: "You are currently subscribed to the newsletters that have a check-mark in the box. To unsubscribe from any newsletter, please remove the check-mark and then click on Unsubscribe." Leave a check-mark in the box next to the newsletters you wish to receive. Below this are three checkboxes: "Growing Child (includes Growing Parent)", "Growing Up (includes Growing Parent)", and "Grandma Says" (which is checked). At the bottom, there is a text input field with the placeholder "We appreciate your constructive comments" and a green submit button.

The browser's developer tools are open to the Network tab, showing a list of requests. A red box highlights the following requests:

Name	Method	Status	Protocol	Type
adsbygoogle.js	GET	200	http/2+...	script
gtm.js?id=GTM-N96XWSV	GET	200	http/2+...	script
integrator.js?domain=growingchild.com	GET	200	http/2+...	script
show_ads_impl_fy2019.js	GET	200	http/2+...	script
zrt_lookup.html	GET	200	http/2+...	document
analytics.js	GET	200	http/2+...	script
osd.js?cb=%2F20100101	GET	200	http/2+...	script
ads?client=ca-pub-564403431925570...	GET	200	http/2+...	document
collect?v=1&_v=j81&a=166686178&=...	GET	200	http/2+...	gif
sodar?sv=200&tid=gda&tv=2020042...	GET	200	http/2+...	xhr
sodar2.js	GET	200	http/2+...	script



It's unfortunately too common for unsubscribe pages to be built this way from legacy organizations, but organizations like Google seem to almost capitalize on it sometimes, like in the requests above that trigger across numerous Google advertising endpoints.

---

### **MailChimp's Mandrill Legacy Email Redirect via their API Can Leak Mandril-Client-User Emails to Advertising and Analytics Companies**

Mailchimp's developer product Mandrill.com was [founded in 2012 and claimed 80,000 users by 2015](#) — lots of developers still use their products, but one of their legacy APIs still has some clients using it. This legacy Mandrill API has a feature that can be used and then it can potentially expose user email addresses on unsubscribe pages to 3rd party advertising and analytics companies.

This Mandrill API doesn't automatically leak user data but there is the option for Mandrill clients to redirect an unsubscribe URL sent via their API to include the user's email address in the URL bar.

Mailchimp was informed of this issue relatively recently, they acknowledged the report and mentioned it was already escalated, but haven't appeared to make many changes yet besides deleting old support articles which recommended the process that could potentially leak a user email to a 3rd party advertising or analytics company.

Here's an example MailChimp Mandrill API redirect URL; this will redirect into a business email address for a newsletter:

[https://launch.us2.list-manage.com/track/click?  
u=baefb9fcb23d26e0308254e5c&id=87ad1425d5&e=af88ddc5b8](https://launch.us2.list-manage.com/track/click?u=baefb9fcb23d26e0308254e5c&id=87ad1425d5&e=af88ddc5b8)

urlscan.io Home Search API Live About Login Sponsored by SecurityTrails 93 running

## Search for domains, IPs, filenames, hashes, ASNs

list-manage.com Search! Reload

[Help & Examples](#)

Search results (100 / 103688, sorted by date)

[Detail](#)

URL	Submitted	Size	IPs	
<p>1 URL: <a href="https://thehipsterengineer.com/broadcast/">thehipsterengineer.com/broadcast/</a> Redirect from: <a href="https://engineeringbroadcast.com">engineeringbroadcast.com</a> IP: 216.172.169.66 - PTR: 66-169-172-216.unifiedlayer.com - Server: nginx/1.14.1 GeolP:  Houston, US - AS46606 (UNIFIEDLAYER-AS-1, US)</p>	4 minutes ago Via: api	1 MB	133	8 2
<p>2 URL: <a href="https://thefringedpineapple.com/">thefringedpineapple.com/</a> Redirect from: <a href="https://thefringedpineapple.com.us2.cas.ms">thefringedpineapple.com.us2.cas.ms</a> IP: 23.227.38.65 - PTR: zagat.ssl.shopify.com - Server: cloudflare GeolP:  CA - AS13335 (CLOUDFLARENET, US)</p>	1 hour ago Via: automatic Source: certstream-suspicious	3 MB	92	33 6
<p>3 URL: <a href="https://thefringedpineapple.com/">thefringedpineapple.com/</a> Redirect from: <a href="https://thefringedpineapple.com.eu.cas.ms">thefringedpineapple.com.eu.cas.ms</a> IP: 23.227.38.65 - PTR: zagat.ssl.shopify.com - Server: cloudflare GeolP:  CA - AS13335 (CLOUDFLARENET, US)</p>	1 hour ago Via: automatic Source: certstream-suspicious	3 MB	92	33 5
<p>4 URL: <a href="https://thefringedpineapple.com/">thefringedpineapple.com/</a> Redirect from: <a href="https://thefringedpineapple.com.eu2.cas.ms">thefringedpineapple.com.eu2.cas.ms</a> IP: 23.227.38.65 - PTR: zagat.ssl.shopify.com - Server: cloudflare GeolP:  CA - AS13335 (CLOUDFLARENET, US)</p>	1 hour ago Via: automatic Source: certstream-suspicious	3 MB	92	32 5
<p>5 URL: <a href="https://thefringedpineapple.com/">thefringedpineapple.com/</a> Redirect from: <a href="https://thefringedpineapple.com.us3.cas.ms">thefringedpineapple.com.us3.cas.ms</a></p>	1 hour ago Via: automatic	3 MB	92	33 6

IP: 23.227.38.65 • PIR: zagat.ssi.shopy.com • Server: cloudflare  
GeolP: 🇺🇸, CA - AS13335 (CLOUDFLARENET, US)

Source: certstream-suspicious

6 URL: [broadwayofficefurniture.us12.list-manage.com/track/click?](https://broadwayofficefurniture.us12.list-manage.com/track/click?)

1 hour ago

1KB 1 1 1

If you visit “list-manage.com” you’ll be redirected to a Mailchimp error/details page

This “list-manage(.)com” domain is owned by MailChimp and numerous legacy Mandrill clients can be found who have various endpoints from this domain embedded and cached in URLscan.io like in this screen shot..

Google has also cached ~47,000 of the Mandrill unsubscribe pages via this search

@ <https://www.google.com/search?q=site%3Alist-manage.com%20unsubscribe> — not all of these results have user emails appended to them, which makes it clear that this is not a feature deployed by all Mandrill clients.

The process to add user emails into the Mandrill redirect URLs was covered in a legacy support article from Mailchimp. This support article was sent to Mailchimp with this research that they haven’t substantially responded to, yet they had time to delete the support article and try to hide what they were recommending to their clients — this page was

deleted: [https://mandrill\(.\)zendesk\(.\)com/hc/en-us/articles/205583017-Can-I-add-an-automatic-unsubscribe-link-to-Mandrill-emails-](https://mandrill(.)zendesk(.)com/hc/en-us/articles/205583017-Can-I-add-an-automatic-unsubscribe-link-to-Mandrill-emails-)

[Can I add an automatic unsubscribe link to Mandrill emails?](https://mandrill(.)zendesk(.)com/hc/en-us/articles/205583017-Can-I-add-an-automatic-unsubscribe-link-to-Mandrill-emails-)

[Yes, Mandrill provides an easy-to-use merge tag to automatically add an unsubscribe link to your Mandrill emails. The... mandrill.zendesk.com](https://mandrill(.)zendesk(.)com/hc/en-us/articles/205583017-Can-I-add-an-automatic-unsubscribe-link-to-Mandrill-emails-)

Even though Mailchimp deleted this support article sometime in the last week, it’s still available [in the Google Cache](https://www.google.com/search?q=site%3Alist-manage.com%20unsubscribe), you can see how Mailchimp showed how to put the user email into a specific query string:

This is Google's cache of <https://mandrill.zendesk.com/hc/en-us/articles/205583017-Can-I-add-an-automatic-unsubscribe-link-to-Mandrill-emails->. It is a snapshot of the page as it appeared on Apr 24, 2020 23:48:44 GMT. The current page could have changed in the meantime. [Learn more.](#)

[Full version](#) [Text-only version](#) [View source](#)

Tip: To quickly find your search term on this page, press **Ctrl+F** or **⌘-F** (Mac) and use the find bar.



[Sign in](#)

## Articles in this section

[About Bounces](#)[About Rejected Emails](#)[About the Rejection Whitelist](#)[About Unsubscribes](#)[Can I add a domain to the Rejection Blacklist?](#)[Can I add an automatic unsubscribe link to Mandrill emails?](#)[Can I receive bounce notifications via email?](#)[Do You Have a Global Suppression List?](#)[How do I export my Rejection Blacklist?](#)[How do I export my Rejection Whitelist?](#)[See more](#)

## Can I add an automatic unsubscribe link to Mandrill emails?

Yes, Mandrill provides an easy-to-use [merge tag](#) to automatically add an unsubscribe link to your Mandrill emails. The merge tag consists of the word **UNSUB**, followed by a colon, and then a full website address (with [http://](#) or [https://](#)) where recipients are redirected when the unsubscribe is processed:

```
*|UNSUB:http://mywebsite.com/unsub|*
```

If a recipient clicks the generated link, the message status changes to **Unsubscribed** and the recipient's address is added to the [Rejection Blacklist](#) in your Mandrill account. The redirect URL will be appended to include two query string parameters that can be used for processing the unsubscribe in your own system or database:

- **md\_id**: The `_id`, as provided in the API call and webhooks of the specific message where the unsubscribe link was clicked
- **md\_email**: A URL-encoded version of the recipient's email address

### Using the unsub merge tag

The unsubscribe merge tag generates only the URL to unsubscribe and can be used in both HTML and text emails. In HTML, you might construct the link like this to create a clickable link for recipients to unsubscribe:

```
<a href="*|UNSUB:http://mywebsite.com/unsub|*">Click here to unsubscribe.</a>
```

MailChimp also scrubbed their larger support article in Mandrill about unsubscribe pages which used to be @ [https://mandrill\(.\)zendesk\(.\)com/hc/en-us/articles/205582947-About-Unsubscribes](https://mandrill(.)zendesk(.)com/hc/en-us/articles/205582947-About-Unsubscribes) — the page is about appending user emails into the URL bar — [Google cached that page too here](#).

Here's a screen shot of that page before Mailchimp deleted it this past week:

## Articles in this section

[About Bounces](#)[About Rejected Emails](#)[About the Rejection Whitelist](#)[About Unsubscribes](#)[Can I add a domain to the Rejection Blacklist?](#)[Can I add an automatic unsubscribe link to Mandrill emails?](#)[Can I receive bounce notifications via email?](#)[Do You Have a Global Suppression List?](#)[How do I export my Rejection Blacklist?](#)[How do I export my Rejection Whitelist?](#)[See more](#)

## About Unsubscribes

You can add your own unsubscribe link, use Mandrill's unsubscribe merge tag, or use an automatic unsubscribe footer.

Mandrill automatically adds a [List-Unsubscribe header](#) to all emails that include a Mandrill-generated unsubscribe link. If recipients use an email program that supports the List-Unsubscribe header (like Hotmail, AOL, or Yahoo), they can use the option in their email program to unsubscribe.

### Add a Link with the Unsubscribe Merge Tag

The merge tag consists of the word **UNSUB**, followed by a colon, and then a full website address (with [http://](#) or [https://](#)). Recipients will be redirected to this website address when they click the link, so it should be an existing page on your site. Here's an example:

```
*|UNSUB:http://mywebsite.com/unsub|*
```

The redirect URL will be appended to include two query string parameters that can be used for processing the unsubscribe in your own system or database:

- **md\_id**: The `_id`, as provided in the API call and webhooks of the specific message where the unsubscribe link was clicked
- **md\_email**: A URL-encoded version of the recipient's email address

### Add an Automatic Unsubscribe Footer

The footer can be added in your [Sending Defaults](#) or using [Rules](#). The footer includes information about the address the email was sent to, along with an unsubscribe link. You do not need a website to process these unsubscribes. Mandrill will direct the recipient an unbranded web page confirming that the recipient address has been unsubscribed.

When someone unsubscribes using a Mandrill unsubscribe link, the information is sent back to Mandrill to process. The message status is changed to **Unsubscribe**, and a sender-level rejection is added for that recipient address.

While the email is on the Rejection Blacklist, if you're using the same 'from' address on emails sent through Mandrill, any emails sent to that recipient will be rejected. The rejection is temporary, and in many cases it doesn't make sense for people to unsubscribe from transactional emails. However, this allows senders to get feedback about how people are engaged with their emails and the rejection prevents continued sending to someone who has signaled they don't wish to receive more emails.

You may remove an email address from the [Rejection Blacklist](#) list for a small reputation hit. Recipients shouldn't be removed in bulk from the rejection list since they're there because of a bounce, unsubscribe, or abuse complaint. Mistakes happen, or recipients don't always understand, so removing recipients from the rejection list is possible.

#### Note

The Mailchimp `*|UNSUB|*` merge tag would need to be modified to include a URL for the Mandrill unsubscribe merge tag if you're already using a Mailchimp template.

Again, MailChimp received the report, they've obviously been scrubbing their content since receiving the research, but haven't sent any other details about their plans to notify users or rearchitect the Mandrill service.

---

## Washington Post Leaks Some User Emails in Base64 to Service Providers, Appears Not to Send Data to Any External Advertising Companies

The Washington Post was recently alerted to a base64 user email data leak to a limited number of analytics companies, primarily Chartbeat.com (and maybe a few others) — it appears no advertising companies received the base64 user email strings that several of their newsletters append to their unsubscribe links.

The Washington Post escalated the initial report quite fast and noted they were addressing their issues — Wapo's base64 user email sharing could be resolved by the time of publication or likely sometime soon after.

Here's one of the unsubscribe links that has had the user base64 email strings:



**We think you'll like this newsletter**

Check out **Opinions A.M. and P.M.** for the best of The Post's opinions and commentary, in your inbox every morning and afternoon. [Sign up »](#)

**The Washington Post**

[Manage my email newsletters and alerts](#) | [Unsubscribe from The Week in Ideas](#) | [Privacy Policy](#) | [Help](#)

You received this email because you signed up for The Week in Ideas or because it is included in your subscription.

©2020 The Washington Post | 1301 K St NW, Washington DC 20071

Not all of the Washington Post newsletters are built the same way — the leak occurs in the unsubscribe links for the “This Week in Ideas” newsletter and another one of their weekly newsletters — their core system for newspapers subscribers that sends daily emails does not seem to be built this same way and doesn’t seem to leak user emails.

The user emails are encoded in base64 plain text format and appended into a “bem” URL parameter — you can see one of these unsubscribe link via this link

@ <https://s2.washingtonpost.com/wp-unsubscribe/newsletters?>

[bem=ZWR3YXXXXXXXNoLnNjb3R0QGdtYWIsLmNvbQ%3D%3D&nlsendid=5e6e0c0bfe1ff6038cda4f2e](https://s2.washingtonpost.com/wp-unsubscribe/newsletters?bem=ZWR3YXXXXXXXNoLnNjb3R0QGdtYWIsLmNvbQ%3D%3D&nlsendid=5e6e0c0bfe1ff6038cda4f2e)

The base64 string is the “bem” param above — mine is slightly obscured above but these can be found across old emails and in some locations on the internet.

---

### **Facebook Manipulates URL Query Parameters (for Filtering), But Still has System That Can be Broken to Leak Emails to 3rd Party Advertising and Analytics Companies**

Nearly all modern websites use Javascript for advertising and analytics tracking, but it’s still very rare for organizations to “sandbox” their partner javascript pixels in a way that ensures that URL parameters don’t get transmitted accidentally to ad tech and analytics partners.

Facebook is one of the few organizations that regularly does “URL referrer filtering” for their javascript partners, and across most (or all?) of their websites. Facebook does this to filter certain URL parameters like the [“mkt\\_tok” parameter reported to them back in 2019 that could leak user emails through Adobe’s one-token-user-authentication architecture](#).

Facebook has never even sourced this filtering product of theirs, likely because it helps to cut



Facebook has never open sourced this filtering product of theirs, likely because it helps to cut down on spam and extra work they need to do internally, but their filters are largely built on their own domain, and they have certain Facebook business marketing pages with 3rd party advertising and analytics pixels, and even though Facebook's filtering product is deployed there, it still typically transmits user emails in some field into 3rd party systems — possibly in fields that can be easily purged, but ingested nonetheless.

Attached below is a screen shot example showing a type of filter where the unknown/unexpected URL parameters that I put into the Facebook URL were then passed into a unique Adobe Marketo field named “\_mchQp” — it's possible that this is a field Facebook parses to ingest unknown inbound data that may/may not get deleted based on some other criteria.

The screenshot shows a browser window with the URL `https://www.facebook.com/business/?sendmyuserdetails=zachFB%40victorymedium.com`. The page content includes a COVID-19 resource banner, the Facebook for Business logo, and a video advertisement for Zach. The developer console is open to the Network tab, showing a list of requests. The selected request is `visitWebPage?_mchHa=&_mchRe=&_mchQp=sen267-pvb-941_mktoresp.com/webevents`. The request headers are visible, including `Referer: https://www.facebook.com/`. A red text overlay on the right side of the console reads: "Facebook doesn't automatically push user's email appended URL parameters to 3rd party advertising analytics partners. A vast majority of these custom parameters are filtered/deleted, but a few of the business pages they push unexpected parameters into third parties hosted via Adobe Marketo software."

The image shows a browser window with a marketing banner and a network developer tool. The banner has the text "Everyone is a marketer with the right toolkit." and a "Sign Up Now" button. The network tool shows a list of requests, with one selected. The right pane shows the query string parameters for that request, including a parameter that appears to be a user email address: "\_mchQp: sendmyuserdetails=zachFB@victorymedium.com". Red annotations on the right side of the image point to this parameter, stating: "see the filter in ac below with the cu parameters being into a known ' \_m query string par".

Organizations that have Javascript advertising and analytics partners need to be aware of their own user data breaches but also plan for ways that attackers could inject bad data into 3rd party systems to corrupt retargeting campaigns or break analytics systems. And at some point, more organizations will need to look to architecture from orgs like Facebook who filter URL parameters and build internal sandboxes to protect user data from flowing across their global data supply legal exposure chain.

---

### What's Next? What Questions are Important?

The organizations included in this research can request any changes or comment additions via [URLdatabreach@victorymedium.com](mailto:URLdatabreach@victorymedium.com)

Individuals who use any of these services and who believe had their emails leaked, should be given an easy processes to request the deletion of their user emails that were sent to 3rd party advertising and analytics companies. The organizations involved in this research should provide that process in whatever format they can provide.

Each organization included in this research should be changing their systems to stop leaking user emails in plain text or base64 plain text formats, they should notify users who could have been impacted by the leaks, and also issue deletion requests for all their users to 3rd party advertising and analytics organizations they work with.

All organizations should be extremely careful about 2-step forms, email tracking that appends

encoded or plain text emails into URLs, and any process that “syncs a user email” to a 3rd party company. This process is almost assuredly not described properly in Terms of Service and Privacy Policies for organizations, and it’s obviously not a process that most users expect to occur.

Unfortunately, as auditors saw with the Cambridge Analytica scandal and Facebook’s inability to confirm that the data was completely deleted, the organizations involved in this research face a similar dilemma tracking down and deleting user emails that were sent to their 3rd party advertising and analytics partners — how can you actually ensure and know this data was deleted? How can users who were involved in these flows ensure their emails are deleted? Who is in charge of requesting that? Each user to every ad tech/analytics company? Each user to the original offending organization? Should organizations proactively request the deletion for all their users? Will the previously leaked user email data just stay with these 3rd party advertising and analytics companies with only a small minority of users requesting deletion?

Finally, many advertising companies have features they’ve built to sync user emails into retargeting lists and other audience advertising targeting strategies, without properly notifying users? How many of those organizations have user emails that were given without the user fully understanding what was occurring or having an ability to delete or modify that information after it was sent?

Hopefully, organizations will start to take a more proactive approach to trying to stop this type of data supply data breach, and a more responsible plan of action after being notified of significant problems.

---

Additional questions or concerns? Ping me on twitter @[thezedwards](#).

[Zach Edwards](#)

Follow

*Founded/Co-founded 6 companies (🍷, 📊, 🗺️, 📈, 🧑, 🏠, 🐔), digital team for Obama 08' + numerous other campaigns, motto = Research. Build. Test. Repeat. // whitehat*

[See responses \(1\)](#)

---

---

Do not reply to this message. Replies go only to the sender and are not distributed to the list.

To unsubscribe from this list, or change the email address where you receive messages, please use the "Modify" or "Unsubscribe Now" links at the bottom of this message.

Any views or opinions presented in this email are solely those of the attributed authors and do not necessarily represent those of the ESPC. The ESPC makes no representation as to the accuracy of the content of this email, and accepts no liability for the consequences of any actions taken on the basis of or in reliance on the information provided. Any discussion of law contained herein should not be construed as legal advice offered to the recipient. Where legal advice is required, recipients should consult independent counsel.

Email Sender & Provider Coalition, PO Box 478, Kennebunk, ME 04043

**ESPC Member Communications** / [espc-announce](#) / see [discussions](#) + [participants](#) + [delivery options](#)

[Permalink](#)

