November 16, 2026

EMERGING THREAT: OPTIMIZELY AND THREE OTHER PROVIDERS POISONED IN IOS EXPLOIT AND SKIMMING CAMPAIGN

Threat actors injected malicious code into third-party scripts from Optimizely, Kissmetrics, StorifyMe and minute-ly on major websites, including BBC, CNN, Zoom, and Fox News. Visitors were served a payment card skimmer and a cryptocurrency address swapper,

with some also delivered iOS exploit kits.

EXECUTIVE SUMMARY

On November 16th, 2026, the consumer privacy publication PrivacyAffairs and an independent security researcher reported that attacker JavaScript had been added to Optimizely Web Experimentation files served to visitors of major websites, including CNN, Zoom, and Optimizely's own site. The injected code loads a script that swaps cryptocurrency wallet addresses, captures payment card details, and sends cookies and page screenshots to attacker infrastructure. On iPhone and iPad visitors, it also loads additional stages that researchers have linked to delivery infrastructure for Coruna and DarkSword iOS exploit kits.

Optimizely Web Experimentation is an A/B testing platform that serves one JavaScript file per customer from Optimizely's content delivery network. Because the malicious code sits in this trusted third-party script, a site's own servers can be clean while its visitors' payment and wallet data is exposed. Optimizely has not said whether its platform or customer accounts were breached. The same campaign also delivers scripts from three other providers, Kissmetrics, StorifyMe, and minute-ly. The earliest of these observations dates to November 13th, three days before the Optimizely reports, so the activity likely began earlier than first reported.

Any organization that loads Optimizely Web Experimentation should treat itself as potentially affected. Given a live card skimmer and cryptocurrency clipper served through a third-party script and an unknown root cause, Beazley Security recommends affected organizations apply the mitigations below immediately and conduct a thorough review for any signs of compromise.

AFFECTED SYSTEMS OR PRODUCTS

Product	Affected Versions
Optimizely Web Experimentation (customer project JavaScript served from cdn.optimizely.com)	Some customer project files observed on November 16th, 2026

Product	Affected Versions
Kissmetrics	Shared loader and at least one customer script observed November 13th, 2026
StorifyMe	Shared web component and customer widgets observed November 13th to 15th, 2026
minute-ly	Loader observed November 16th, 2026

MITIGATIONS / WORKAROUNDS

At the time of writing, Optimizely has not published a statement, root cause, or remediation guidance. Researchers found that not every Optimizely customer was affected, and several files rebuilt during the same day were clean. While the cause remains unknown, a file that is clean at one point in time may not stay clean.

Until Optimizely provides guidance, the following steps may help to reduce risk:

- Download your Optimizely project file exactly as `cdn.optimizely.com` serves it, and review the "Project JavaScript" settings in your Optimizely project. Look for the domains listed under Indicators of Compromise or any script loader you did not add. A plain-text search can miss encoded variants: CNN's file carried a second copy that hid the address in Base64, and TalkTalk's carried a longer loader.
- Consider removing the Optimizely snippet from the site until Optimizely explains what happened, starting with checkout, login, and payment pages.
- Block `1688360[.]cc`, `5vobauua3ojpklx[.]icu`, `eil786npto9u5bq[.]icu`, and `je4ft3gn8vxudum[.]icu` in your Content-Security-Policy (CSP), and alert on `script-src` violation reports that name unknown hosts. On two affected sites, CSP blocked the payload from loading.
- Review recent changes to Optimizely Project JavaScript and the accounts that made them. Limit write access to the people who need it, rotate Optimizely account credentials, and enforce multi-factor authentication. Either compromised customer accounts or a platform compromise could explain the injected code.

- If payment card numbers pass through form fields that your own pages host, treat this as a possible card data breach and notify your payment processor. The Payment Card Industry Data Security Standard (PCI DSS) requires merchants to inventory the scripts on payment pages and monitor them for unauthorized changes.

INDICATORS OF COMPROMISE

Exploitation was confirmed on November 16th, 2026. That day, PrivacyAffairs and an independent researcher found the injected loader in live Optimizely project files and recovered the payload it fetches. At the time of writing, neither source has reported that the injected code was removed. The primary payload domain, `1688360[.]cc`, was registered at 05:23 UTC on November 16th, and PrivacyAffairs notes that visits before that time could not have reached it. NOW TV's file, last rebuilt at 10:52 UTC that day, already contained the loader. The attacker has not been identified.

Search for requests to `1688360[.]cc` (the paths `/track.js`, `/hit.php`, `/card.php`, and `/html2canvas.min.js`) and to the `.icu` domains in the table below. Review the following sources:

- Proxy, DNS, and firewall logs.
- CSP violation reports.
- Client-side monitoring data, including `sendBeacon` calls to unknown hosts on checkout pages and `html2canvas` requests to hosts other than your own.
- Your Optimizely project files, including historical copies if you keep them.

To check the file your site serves, use the exact Optimizely URL your pages load. It takes one of two forms:

```
cdn.optimizely.com/js/<project_id>.js
```

or

```
cdn.optimizely.com/public/<account_id>/s/<name>.js
```

A match on `atob(` is not malicious by itself, but it flags Base64 decoding for manual review.

```
curl -s "<optimizely_script_url>" | grep -E -i "1688360|5vobauua3ojpklx|eil786npto9u5
```

The attacker replacement wallet addresses below are listed in the header of `track.js`. Financial and fraud teams can use them to trace diverted transfers.

Indicator	Type	Des
1688360[.]cc	Domain	Payl data for th injec Optir regis 16th, UTC
hxxps://1688360[.]cc/track.js	URL	Main skim crypt clipp scre cook
hxxps://1688360[.]cc/card.php	URL	Card endp
hxxps://1688360[.]cc/hit.php	URL	Exfilt for p cook and addr
hxxps://1688360[.]cc/html2canvas.min.js	URL	Scre serve infra
5vobauua3ojpklx[.]icu	Domain	iOS s

Indicator	Type	Des
		host
hxxps://5vobauua3ojpklx[.]icu/boot.js	URL	iOS s scrip
eil786npto9u5bq[.]icu	Domain	iOS s host
hxxps://eil786npto9u5bq[.]icu/boot.js	URL	iOS s scrip
hxxps://eil786npto9u5bq[.]icu/JIUJIU888/weifile.html	URL	Attac on th stage block resea
hxxps://webshell[.]lol/get.js	URL	Load iOS e serve poisc Kissr Stori and a Optir
hxxps://1688360[.]cc/px.gif	URL	Error in the payl
hxxps://5vobauua3ojpklx[.]icu/D11/b.html	URL	Entry as Da deliv
hxxps://eil786npto9u5bq[.]icu/A8/a.html	URL	Entry asso Coru

Indicator	Type	Des
hxxps://ios-hack[.]icu/api/ip-sync/sync	URL	IP sy requ visitc
hxxps://l7bcsnwc7nzr3uo[.]icu/api/iptj	URL	Addi endp trace
je4ft3gn8vxudum[.]icu	Domain	Dom reco CSP alon camp infra
c2a34c579e4c2e8802752c93b3a9cca0230610f3e4dd6688f4272459c7c7c162	SHA256	Emb payl bytes the E UAE
0f2f2a27d19112e5800541daecc91eb6c77e49f9b43ed23c890b876631978533	SHA256	Pois load
c154f3b644cd51f2446483c7ffd2ae4f178c4b0cb54846cfb5b83215367349cf	SHA256	Pois Kissr shar bytes
0408c5e199664c7841c4dfdb6e5065ca394f11247e5ed992ce28ecaafc936455	SHA256	Pois stori elem bytes
123f813b18cf5a3b513beabd52a110e19f1c7834f2aa9b972633b2214c57e737	SHA256	webs

Indicator	Type	Des
8d69f32fff37208e52dc573111e8db99f031f4913845a24fb2e14755747ac860	SHA256	boot 5vob
e410b6509f33defce16ad7896cbe2fb3cc689f2cc53d87c87d75efba6b57e290	SHA256	boot eil78 (arch
f7b84b9a94f749f8ced3f0112d0a66894e8bbc3b4aa5592021d1b8976a9bad40	SHA256	track
2f69879f1c5bdfe9c075ef67ddb6405afe992aff89dcf42863567f83237d3ba6	SHA256	boot eil78
0xC4026b06D82b0F0B363eda0642D61CfbDc17884e	Other	Attac Ether addr
19g9rZcPeiz8rUvyWpTRyXXhPnjsWrc3kW	Other	Attac Bitcc addr
THvs6p56XkUiMkJeSc51cAP64u2RsTYqxa	Other	Attac Tron
UQAgM1VqSZQiJLSG5w-ovurq5CZdxLEK3_7CMmwALTA5e09p	Other	Attac TON (frier
0:2033556a49942224b486e70fa8beeaeae4265dc4b10adffec2326c002d30397b	Other	Attac TON form
FVtL8NMYS9PA1v3HE67XUGCtGA9JaZUvqav2fU2LrV5n	Other	Attac Solar addr
ltc1q3wp7g6ds0ggg6fl6kncpa4sepn0q3qfacgred5	Other	Attac Litec

Indicator	Type	Des
		addr
qzkaaf8n6grmpr0d9lx8xutn3r9lk722f566hhpe58	Other	Attac Bitcc addr
DBEGrxRF7KMSuhTS4c1MFviStU6efdMnu5	Other	Attac Doge addr

THREAT INTELLIGENCE

Researchers found the loader in Optimizely files on news, software, telecommunications, and insurance websites. Through other routes, the payload also loaded on news, retail, and travel websites. PrivacyAffairs downloaded each file between 20:33 and 20:36 UTC on November 16th. It confirmed the loader in the Optimizely files of Optimizely itself, NOW TV, CNN, Virgin Media, and TalkTalk, and found the files of Channel 4, American Eagle, and Marks & Spencer clean. In separate page captures, the independent researcher found the same loader in Optimizely files used by CNN, Zoom, BBC, Salesforce, GitLab, Hootsuite, Smartsheet, Virgin Media O2, Stamps.com, Tricentis, Mutual of Omaha, and HSBC Insurance Singapore. That researcher also observed the payload loading through other routes on three sites: an advertising wrapper on Fox News, a first-party script on American Eagle, and a Jscrambler-protected script on Air France's Italian site. No other source has confirmed these other routes.

Neither source has attributed the activity. PrivacyAffairs noted that targeting only older iOS versions resembles the iPhone exploit kits behind recent malicious advertising on Dutch news sites, but it stated that it has not confirmed this link.

This incident follows several recent attacks that turned shared website scripts against downstream sites:

- A cryptocurrency clipper injected into an Adform tracking script.
- ClickFix lures served through Brevo's embedded scripts.

- A card skimmer planted in Convertr's hosted analytics tracker, identified in October 2026.

TECHNICAL DETAILS

Attackers inserted malicious code into JavaScript files served from the content delivery networks of four third-party providers including Optimizely Web Experimentation, Kissmetrics, StorifyMe, and minute-ly. Optimizely is the most widely observed, with poisoned customer files confirmed on sites including BBC, CNN, Zoom, GitLab, and Salesforce. Optimizely Web Experimentation delivers each customer's experiments through a single JavaScript file. Sites load it from

```
cdn.optimizely.com/js/<project_id>.js
```

or a

```
cdn.optimizely.com/public/<account_id>/s/<name>.js
```

path. The file contains the customer's tests plus any custom code entered as "Project JavaScript", and it runs on every page that includes it. In the infected files, the last line of that custom code is the same one-liner. It is a self-invoking function inside a `try`/`catch` block that creates a `script` element with its source set to

```
hxxps://1688360[.]cc/track.js
```

, marks it `async`, and appends it to the document head. The independent researcher found the line in GitLab's project file (`5113954737848320.js`) and in ten other project and snippet IDs belonging to unrelated organizations. On Zoom's site, a OneTrust consent stub controlled when the Optimizely file loaded but did not clean its contents.

The root cause is not known. PrivacyAffairs observed that the infected files were last rebuilt at different times through November 16th, from 10:52 UTC (NOW TV) to 20:32 UTC (TalkTalk), while other customers' files rebuilt that day stayed clean. It concluded that someone with write access to either Optimizely's systems or the affected customers' accounts inserted the code. Optimizely has not said which.

The independent researcher describes `track.js` as one self-contained script with four modules. PrivacyAffairs describes it as about 30 KB of plain, commented JavaScript.

- **Cryptocurrency clipper.** The script uses regular expressions to find Bitcoin, Ethereum, Tron, Solana, Litecoin, Bitcoin Cash, Dogecoin, and TON addresses, then validates each match with the correct checksum for that currency. It replaces matches in page text every 50 milliseconds, but only on pages that contain payment keywords. It also rewrites form fields (including framework data models), link and image targets, copy events, clipboard reads and writes, `navigator.share`, and the URLs and bodies of `fetch` and `XMLHttpRequest` calls.
- **Screenshot and cookie exfiltration.** The script posts the following to `/hit.php`:
  - The page URL (up to 500 characters).
  - Cookies (up to 8,000 characters).
  - A JPEG screenshot rendered with `html2canvas`, which it fetches from the same attacker host.
  - The original wallet addresses it replaced.

PrivacyAffairs states that this happens the first time the clipper swaps an address.

- **Card skimmer.** The script hooks form submissions and POST bodies sent through `fetch` and `XMLHttpRequest`, including JSON, `FormData`, and nested objects. It looks for card numbers that pass the Luhn checksum and a card brand check, together with a valid MM/YY expiry date and a 3- or 4-digit security code. It removes duplicates within a session and sends results to `/card.php` with `sendBeacon`. PrivacyAffairs notes that the script cannot read card fields that a payment provider hosts in its own iframe (for example Stripe or Adyen), so sites that collect card numbers in their own forms face the highest risk.
- **iOS second stage.** On iPhone, iPad, and iPod devices only, the script injects `boot.js` from two `.icu` domains. On iOS or iPadOS 18.6 and earlier, it also adds a hidden iframe that, according to PrivacyAffairs, loads further code. The recovered `boot.js` is obfuscated, and neither source has published an analysis of it or of the iframe content at the time of writing. PrivacyAffairs reports that one of the `.icu` domains was registered at 16:38 UTC on November 16th.

The analysis comes from PrivacyAffairs, which read `track.js` as text without running it, and from the independent researcher who published it as "OPTIMJACK" on amirulcyber.com. That researcher recommends testing staging pages with a canary wallet address to detect the clipper: paste the address, submit the form, and compare the submitted value. The researcher notes that the page

text sweep only runs on pages with payment keywords, so a canary test on a page without them can miss part of the clipper's behavior.

HOW BEAZLEY SECURITY IS RESPONDING

Beazley Security is monitoring client perimeter devices through our Exposure Management Platform to identify impacted devices and support organizations in remediation of any issues found.

We are also conducting threat hunts across our MDR environment to detect potential exploitation attempts against our clients.

If you believe your organization may have been impacted by this attack campaign and need support, please contact our Incident Response team.

SOURCES

- PrivacyAffairs: Optimizely's A/B-Testing Code Is Loading a Card Skimmer on Major Websites
- amirulcyber.com: CNN, Zoom, Fox News Served Visitors a Crypto-Stealing Skimmer (OPTIMJACK)
- The Hacker News: Hackers Poison Adform Script to Swap Crypto Wallet Addresses Across Customer Sites
- BleepingComputer: Brevo supply-chain attack injected ClickFix scripts on customer sites
- BleepingComputer: Hackers hijack Convertr analytics script to steal payment card data
- HudsonRock: Poisoned Scripts on BBC, Zoom and Other Major Websites Linked to iPhone Exploit Loaders