Me2B Alliance Product Testing Report: Deeper Look at K-12 School Utility Apps Surprisingly Uncovers Global Advertising Company From CBS/Viacom, Unexpected Security Risks

Research Performed by Zach Edwards, Kelsey Nordstrom, George Vo
Written by Zach Edwards, Kelsey Nordstrom, Lisa LeVasseur
Contributors: Andrea Ausland

#DataPrivacy #SchoolUtilityApps #MobileApps #DigitalAdvertising #DataSharing #DanglingDomains #DataBrokers #COPPA

Table of Contents

1 ABSTRACT

TLDR: Common technical frameworks and app templates used by hundreds of organizations, when combined with technical weaknesses built into devices and operating systems from Google and Apple, are leading to unregulated and out of control student and parent data sharing to unexpected online advertising companies.

After publishing Spotlight Report #1 report the Me2BA was contacted by the Student Data Privacy Project to examine apps used by 18 schools/districts for their FERPA Complaint with the Department of Education. Our data supply testers noticed significant network traffic, well beyond the SDK channels, as well as certain legacy development tactics that relied upon in-app browsers opening websites within the apps. The hypothesis was that many of the school utility apps were using in-app WebView methods to display content, and this was, indeed, the culprit. The WebView development technique allows external websites to open within an app, without launching a separate browser (see Appendix A for an example, and guidance on how to spot this technique). This process results in all of the vendors integrated into a website receiving user data in the context of the app that opened that webpage within their in-app browser. For school utility apps, this so-called "context" typically includes the name of their school, or school district. We took a closer look at the network traffic to confirm the assumption and to determine the scope and scale of this data sharing.

Within the domains in the sample, we noticed significant amounts of network traffic associated with school sports pages and discovered a vendor providing sports scores for K-12 schools across the U.S. and monetizing their "free service" with extremely aggressive advertising monetization schemes, baked right into these taxpayers funded school utility apps.

The company providing this "free service for sports coaches" monetized with online advertising is MaxPreps.com, a subsidiary of CBS/Viacom, who are also the owners of the popular kids' television channel, Nickelodeon.

We did not expect that our deep dive into WebView in-app browsers within K-12 school utility apps would end up requiring us to focus significant time researching a data supply chain owned by one of the largest media companies in the U.S., but we followed the data pipelines and the facts. As noted in Spotlight Report #1, in April 2021, Disney, CBS/Viacom and about a dozen other companies were parties to the largest settlement against brokers of kids' data in U.S. history. CBS and other parties were required to make changes to some of their products and delete certain data. Yet MaxPreps seems to have never come up in that lawsuit, it's never come up in any significant public reporting or research, and any changes to other CBS products as a result of the settlement do not appear to have not made their way to MaxPreps products. While MaxPreps was never mentioned in the California

settlement and the details seemingly would not have required CBS/Viacom to make changes to their subsidiary MaxPreps, it's clear that the behavior that the settlement pointed out, which CBS/Viacom agreed to stop, is similar if not worse within this subsidiary that offers free products for schools.

Our research took another unexpected turn in the course of the deep dive into MaxPreps, when we came across a handful of "dangling domains." We wrote about one such dangling domain in particular that Apple quietly purchased for $3,695 in late September 2021. The domain for sale was previously owned by a company that went bankrupt and the domain was integrated into a legacy SDK product across 159 mobile apps, with 155 of them being on Apple's iOS marketplace, with a potential install base of tens of millions of devices.

In addition to the dangling domains, we also observed several hijacked domains leading to malicious sites. In at least one instance, we observed in dismay when a dangling domain was purchased by an unknown actor over the course of a few days. The following apps/domains fell prey to hijackers before we could intervene:

- The Santa Monica-Malibu USD Android App from Blackboard Inc. had a dangling domain of "Malibuhigh.org" - this domain still to this day hosts a fake legal website, and there could still be risks from Business Email Compromise schemes or other ways to abuse the fact that this was a real domain used by a school district in one of the wealthiest counties in the United States. Here is a Google search result showing files where this legacy domain was referenced as being valid – other government agencies communicated with this domain at different points in the past.

- Maryland's largest school district's Android App, also from Blackboard Inc., already lost their sports domain by the time we figured it out, with Magruderathletics (WARNING).org being compromised and still hosting malicious redirects **to this very day.** After the Me2B Alliance alerted Blackboard Inc., they were able to quickly remove this domain from their active mobile app, reducing some of the risks. This is also an active domain, and Business Email Compromise risks for emails that originate from this domain (i.e. "@Magxxxxathletics.org") remain a real threat.

- The Quinlan, Texas School District had a domain that went up for sale for $30 that was integrated into their Android app, which was purchased before anyone could take action. After the Me2B Alliance alerted Blackboard Inc., the dangling domain link was removed from the app, and subsequently the Android app was pulled down from the Google Play Store

The research we are releasing today focuses on an intensive evaluation of 11 school utility apps (from an original pool of 18 apps) made by companies who support thousands of other schools with similar app frameworks.

In short, the use of WebView in school utility apps, and the operational challenges to maintain them, creates a significant channel for data sharing and also introduces serious security risks. If people using these mobile apps "can't choose their own browser" they can't make informed choices that empower them to block and stop some of these data transfers, which can be downright dangerous when an app for kids integrates dangling domains into WebView interfaces. If Google and Apple merely made a few changes to empower users over developers, these risks for schools, kids, parents, and administrators would nearly completely disappear.

These risks have been compounded by certain companies providing "free software for schools" that purposefully monetizes these free tools for apps and websites via data sharing and online advertising, with the new Me2B Alliance research focusing on the CBS/Viacom subsidiary called MaxPreps.com. Another way to think about this research is that we've attempted to point out a technical framework that numerous school utility apps are using, which utilizes a type of "Content Management System" (CMS) that allows school administrators to "add links" into an app, without actually submitting a new version of that app into the app store – and the links being added into the apps are merely web URLs, with the web content rendering within the app's in-app browser. These websites rendering in the apps contain advertising pixels/javascript code, which then collect data within the apps when opened by users, and are sharing the access and user data with new companies – hundreds of them, sometimes more. Rarely do app privacy labels account for these data transfers, and neither the app makers, the schools, or the vendors collecting data within those apps and WebView URLs are currently taking accountability for making these kids user data flows safer. We're surprised and alarmed by this "advertising for kids" architecture happening within school utility apps paid for by taxpayers, with some companies seemingly earning sizable revenues from these data pipelines. As a result, serious questions need to be asked of all the organizations participating in these schemes.

This report includes guidance on how to identify a school utility app with potentially unsafe WebView links in Appendix A, which we hope provides investigative journalists, data auditors, school administrators, parents, students, app developers, and everyone-in-between with a way to recognize when an app is opening web links.

If you're interested in having the Me2B Alliance take a closer look at your school's apps, please contact us at services@me2ba.org

2 INTRODUCTION

After publishing our research across 73 school utility apps for Spotlight Report #1, we were contacted by the Student Data Privacy Project to examine apps used by 18 schools/districts for their FERPA Complaint with the Department of Education. Our data supply testers noticed that data was being sent to third parties beyond just the SDK channels for data sharing. Based on the visual layouts of the apps and our awareness that in-app browsers can create unexpected and unique risks, our hypothesis was that in-app browsing using WebView methods was the core culprit for these advertising companies showing up in our network logs. A simple way to think about these data sharing methods are "websites opening within apps." Thus, we decided to take a closer look at the network traffic in a handful of apps from the FERPA research schools to validate the assumption and learn more about vendors operating in this space within school utility apps.

This research involved a smaller and deeper analysis, focused specifically on the data supply vendors within school utility apps.

What we found not only validated the hypothesis that school utility apps use WebView in-app browsers extensively to serve content -- indeed enabling more data sharing with third parties in unexpected ways --but we also discovered three other disturbing findings:

- A website for school sports called Maxpreps.com, used by a large number of schools in the US, shares student data with an untold number of advertising companies. MaxPreps is owned by Viacom/CBS, a public company, named in a landmark settlement. The settlement required Viacom to remove cross-site identifiers related to a children's app called "Llama Spit Spit" along with several other niche-requirements to comply with laws focused on kids' data sharing.

- We found several instances of hijacked domains previously owned by schools, leading directly to "parked domains" filled with online advertising.

- We documented malicious pop-ups within school utility apps due to advertising data supply chain attacks by malicious advertisers promising "free gift cards."

- In addition to hijacked domains, we found several dangling domains, including the one mentioned in our blog post from earlier this year.

During the auditing process, our team determined that the majority of school apps analyzed were built using app templates created and licensed by companies who provide app platforms (including the CMS-like backend administrative dashboard) to hundreds if not thousands of schools, and other types of community/religious organizations. *In short, the apps were built with a common technical framework, and this practice is widespread, yet*

neither Apple nor Google seem to account for this behavior with specific in-app browser-default controls – letting users override the use of an in-app browser -- a device feature, which would protect end-users from risks created by this legacy development practice.

Common technical frameworks – licensed app templates plus the CMS-like administrative portal - used across many organizations are leading to unregulated and out of control student and parent data sharing to unexpected online advertising and big tech vendors.

In order for many school apps to stay updated with relevant news, events and other details for students and parents, it is a common tactic for developers to integrate web pages directly into the apps, also known as a "WebView"[1] or an "in-app browser." This technique allows the school to create menus of several links (URLs/domains) within the school apps that will open an in-app browser to view a related webpage with particular details about the school or program. The schools keep the details updated by maintaining the web pages that host the content or by relying on third party websites to keep content updated.

This is a method that allows the school to include dynamic, fresh information in mobile apps without having to update the app itself, and it's particularly common for school calendars, lunch menus, and details about extracurricular activities and school events. The school's app administrators update websites, and the apps "auto-update" with the new content from the pages. This is a helpful process to allow more people to provide content updates for apps, but it becomes problematic when those websites share data with unique advertising and analytics vendors. It becomes an emergency (as found during this research) when a school loses control over one of the domains integrated into their school apps, and an even bigger problem when that domain is controlled by spammers and scammers.

This problem would be mitigated if people could designate a default (and privacy respecting) browser to be the default browser used for in-app WebViews, and if Apple and Google didn't make it possible for developers to override a user's browser preference.

2.1 What is WebView?

All mobile apps with an "in-app browser" are using "Webview" code/APIs, and the rules and protections for these in-app browsers are slightly unique between Apple iOS and Android platforms. Other devices / IOT use similar technical concepts, but Apple and Android have the most apps using this legacy practice as they have the oldest app marketplaces.

WebView is a low-level set of functions (APIs) provided by an operating system within a mobile device that allows native apps to present web pages, woven seamlessly into the

[1] https://developer.android.com/reference/android/webkit/WebView for Android, and https://developer.apple.com/documentation/uikit/uiWebView for iOS.

application without having to open or close a separate browser. WebView allows native apps to selectively become a browser while keeping the user in the app.

A 2011 paper on WebView attacks against Android devices explains this legacy development and data supply chain strategy:

"WebView is an essential component in both Android and iOS platforms, enabling smartphone and tablet apps to embed a simple but powerful browser inside them. To achieve a better interaction between apps and their embedded "browsers", WebView provides a number of APIs, allowing code in apps to invoke and be invoked by the JavaScript code within the web pages, intercept their events, and modify those events. Using these features, apps can become customized "browsers" for their intended web applications." [2]

WebView in iOS relies on WebKit, an open-source browser "engine" first introduced 23 years ago in 1998, which later evolved into the Safari engine used by Apple around 2013. The WebView APIs are therefore limited by WebKit capabilities supported in the OS. iOS WebView is restricted and protected by iOS WebKit (built on Safari) limitations, and Android WebView is restricted and protected by a slightly similar concept with Chromium limitations. However, they both ultimately support in-app browsing as a seamless user experience without giving users full control over their own data sharing preferences within these in-app browsers.

2.1.1 Apps That Open a Separate Browser

In contrast to the use of WebView, some apps will open external webpages by opening a separate browser. In this case, it's obvious to the user of the app that they are no longer in the app and that they have been redirected to an external website in their default browser of choice. When the app opens a separate browser, it will use the user-specified default browser (privacy settings intact) thus allowing the user more control and security. This is the case on both Apple iOS and Android, although on both platforms app developers "get to choose" and can override user choices by loading an in-app browser instead of letting the user choose.

On Android, there have been some preference shifts that over the past few years that make it slightly easier to default an in-app browser, but these options are largely hidden and don't seem to prevent app developers from deploying WebView content into apps.

[2] "Attacks on WebView in the Android System", T. Luo, H. Hao, et al, *ACSAC 2011*, December 5-9, 2011, Orlando Florida USA. https://web.ecs.syr.edu/~wedu/Research/paper/webview_acsac2011.pdf

2.1.2 WebView in iOS

In iOS, original capabilities were provided through UIWebView and WebView.

Last year however, Apple started a process to deprecate some of the oldest aspects of App WebViews, but the full deprecation deadline was paused seemingly due to Covid-19.

Apple also has a 30 minute video and support documentation explaining how developers can update their apps from this legacy WebView coding practice into their more modern "WKWebView" practices, but it appears that no time table to depreciate this has been reset.

On iOS, WebView APIs are restricted by certain rules written into Safari's Webkit browser, which has some tracking protections that are stronger than Android's default Chromium browser frameworks.

Additionally, Safari's WebView browsers don't have the ability to send a "Global Privacy Control" (GPC) signal to send signals to opt-out of data selling and sharing, even though this standard has made its way into both the Mozilla Firefox browser and the Brave browser. If Safari's WebView browsers supported the GPC, a "do not sell or share my data" signal would be sent to third party websites, which would express the preference of these users to opt-out of all data sharing and selling schemes they can legally opt-out of, which would help to keep users of all ages safer. A business can still "ignore" this GPC signal, but ignoring a preference creates unique exposure compared to no preferences being sent from a user via automated signals.

2.1.3 WebView in Android

For Android developers, Google's flagship opensource mobile operating system still heavily relies on WebView pages in apps, and to make matters worse, Google has detailed documentation about how to use JavaScript to increase the amount of user data being collected via in-app WebView pages.

Google's Android provides the ability for a user to set their default in-app browser, but app developers can still seemingly override this behavior and user choice.

2.1.4 User Control of WebView-Launched Browser

Originally, users on both iOS and Android had no ability to specify the browser used by WebView, nor did they have access to any controls or settings for these in-app browsers.

For iPhone users, there's no ability to set a default browser to be used by mobile apps that open in-app web pages. Such an ability would ensure that any links open in a preferred,

consistent environment, expressing data blocking preferences and automated signals of consent/opt-outs. While Safari's WebView implementation could be argued is "safer, with less external data sharing" than the Android Chromium WebView implementation, partially due to Apple's new iCloud Private Relay which protects network data sent within mobile apps. Yet Apple's Safari still doesn't have a way for users to transmit automated opt-out preferences via the Global Privacy Control or any other respected automated signals, even though this automated option is required under new California data privacy laws.

Unlike on Apple, for newer Android devices/users, there *is* an option to set a new default browser to open all in-app webpage links, though this feature is not heavily promoted. Moreover, Android apps appear to be able to override or ignore a user's preference when opening an in-app WebView page, as observed during our research.

It appears Android has gone further than iOS to give users control of WebView in-app browsers, but it doesn't appear that either platform gives full control to avoid these privacy pitfalls.

2.2 How Pervasive is WebView?

It's difficult to estimate what percentage of apps use WebView, but according to a WebView research paper from 2011 research:

"Currently, in the Android market, 86 percent of the top 20 most downloaded apps in 10 diverse categories use WebView." [6]

The use of WebView, however, is a very slowly fading coding practice, but there are a massive portion of legacy apps still using it. Even some new apps deploy in-app browsers due to the additional user data collection benefits they offer app makers.

We see the use of WebView in older apps that display web-based information, and for apps that conduct sophisticated user data profiling and advertising schemes —which is why Facebook, LinkedIn, Twitter and many other enterprise apps for large web services employed WebView.

Additionally, the use of WebView is popular in utility apps built from generalized templates, a business strategy that enables profitable distribution of custom apps to a wide customer base. ***Virtually every legacy school utility app is precisely this kind of app, where it's rare to not see an in-app browser being used in some way.***

In order for many school apps to stay updated, with relevant news, events and other details for students and parents, it is a common tactic for developers to integrate web pages directly into the apps - basically creating menus/links within the school apps that will open

an in-app browser to view a webpage with custom information about the school --- and the schools keep the details updated by maintaining the web pages that host the content.

In research released earlier this year, the Me2BA team found that the school apps in the research sample hadn't been updated in 11.6 months on average. This statistic may have been slightly confusing for some parents due to the fact that the content seen in these schools' apps was updated at a much higher frequency. This is due to these apps being built primarily with in-app WebView links, where the apps are mostly just opening website links.

Another way to explain it: the "app administrators" are actually updating websites, and the apps auto-update with the new content from the pages. This is a helpful process to allow more school personnel to provide content updates for apps, but there are risks involved for end-users because they can't choose their own browser with consistent data sharing / opt-out preferences.

For reference, this publicly available tutorial on Blackboard's CMS admin panel provides clear instructions for school administrators on how to add URLs to their app. The tutorial also highlights a feature that makes it possible for school administrators of Blackboard school utility apps to open web links within the app or the user's default browser (See Figure 1) – and by letting an admin turn off safety features with the toggle of a button, Blackboard has made all of the students, parents, teachers and administrators using their apps less safe.

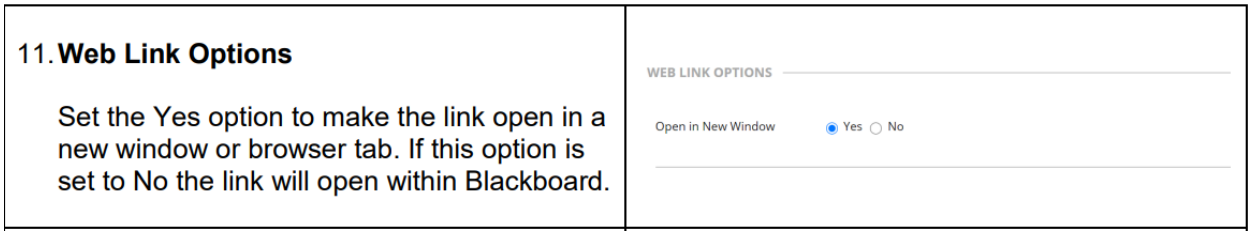


Figure 1 Blackboard CMS Setting to Open Web Links In-app or in Default Browser

2.3 How Does WebView Enable Data Sharing?

In our previous research, we examined third-party data sharing through the use of SDKs (Software Development Kits). Another way data can be shared with third parties is when an app uses WebView to access third-party websites.

User data is shared on a website opened via in-app browsers in exactly the same way it happens in a browser, but with fewer controls for end-users. Vendors integrated into these websites can collect network/IP address data, cookies, other local storage / mobile ID fields, and they can reshare this information to new vendors through the same developer tactics used on websites.

Many advanced mobile users have turned to ad blocker apps in order to try and minimize data flows within in-app browsers, but the vast majority of these ad blocker apps rely on "domain / request lists" in order to filter the extra sharing, which is a process that isn't perfect and can't protect against the most aggressive data collection efforts from unscrupulous vendors. The only way to truly stop this data sharing is at the device-level, with the support of the device operating system manufacturers.

2.4 What Information Gets Shared Using WebView?

For both iPhone and Android users, by default, the network data (IP address) is sent to most of the website vendors who collect data on any web pages that are triggered from in-app browsers. Including network data, these in-app web browsers can share cookies, fingerprint data and other local storage fields that are used to track users persistently over time.

Over the last few months, Apple began to test and roll out their new iCloud Private Relay, which obfuscates network data and should eventually protect against some types of tracking within WebView in-app browsers.

In general, just like with SDKs, any and all personal information available to the app can be shared with third parties.

2.5 Security Risks of Using WebView

There are several risks associated with using WebView in particular and using in-app browsers in general. We list only a couple here.

2.5.1 Dangling Domains

It is frequently the case that links (domains) included in mobile apps very simply expire and become available for purchase in the open market. We call these "Dangling Domains"—domains that are actively being linked to in mobile apps, but no longer have a viable service provider associated with them. We found two flavors of this happening, and there are undoubtedly more:

- Schools simply allow domains to expire, possibly due to personnel turnover, or simply due to credit card expiration.

- The service provider using the domain goes out of business. See our recent blog on this affecting over 150 iOS apps and potentially scores of millions of devices.

2.5.2 Hijacked Domains

Dangling domains become problematic when those websites share data with unique advertising and analytics vendors, and it becomes an emergency (which we observed during this research) when a school loses control over one of these domains integrated into their school apps. It becomes an even bigger problem when that domain is "hijacked", controlled by spammers and scammers as our research detailed below.

- The Santa Monica-Malibu USD Android App from Blackboard Inc. had a dangling domain of "Malibuhigh.org" - this domain currently hosts a fake legal website, and there could still be risks from Business Email Compromise schemes or other ways to abuse the fact that this was a real domain used by a school district in one of the wealthiest counties in the United States. Here is a Google search result showing files where this legacy domain was referenced as being valid.

- Maryland's largest school district's Android App, also from Blackboard Inc., already lost their sports domain by the time we figured it out, with Magruderathletics (WARNING).org being compromised and still hosting malicious redirects **to this very day.** After the Me2B Alliance alerted, Blackboard Inc., they were able to quickly remove this domain from their active mobile app, reducing some of the risks. This is also an active domain, and Business Email Compromise risks for emails that originate from this domain (i.e. "@Magxxxxathletics.org" remain a real threat.

- The Quinlan, Texas School District had a domain that went up for sale for $30 that was integrated into their Android app, which was purchased before anyone could take action. After the Me2B Alliance alerted Blackboard Inc., the dangling domain link was removed from the app, and subsequently the Android app was pulled down from the Google Play Store.

3 METHODOLOGY

3.1 Sample Selection

As with Spotlight Report #1, this research examined school utility apps, which are apps used for communication and sharing between students, parents, and school personnel. We used the list of 18 schools/districts that we investigated for the Student Data Privacy's FERPA Complaint with the Department of Education, where we explored the SDKs used in the apps (similar to Spotlight Report #1). Seven of the schools required accounts/logins in order to use the apps, which we were unable to obtain. Thus, we were able to examine some of the apps more deeply than others.

3.2 Data Collection

For this research, the data collection consists of two steps:

1. Determining which apps include WebView, and
2. Assembling a list of domains linked to by each app that uses WebView.

During the analysis portion of our research, our team realized that the vast majority of unique advertising requests were initiated within the apps through MaxPreps.com webpages integrated into the "Sports" tabs of the school utility apps. Thus, the testing focused on opening Sports tabs in the apps, in particular.

There are a few common methods to definitively detect WebView use in a mobile app. Please refer to Appendix A for tips on how you can determine if your app is launching an in-app browser.

- For more advanced developers or heavy app users, the visual rendering of content within a WebView in-app browser will be noticeable because it loads more like a website (sometimes in stages, sometimes with long load times for large pages), compared to a mobile app where the content has sometimes already been downloaded with the app and the content appears instantaneously. This method can be used as an early indicator prior to more sophisticated network capture tests.

- One method that works for Android apps and open-source apps (iOS app packages are encrypted and analyzing any source code is uniquely challenging compared to Android) entails analyzing the app source code to see if it is using WebView APIs. There are multiple ways to perform this analysis, one example here. Typically, this method involves examining the app source code for inclusion of the WebView class in the app, although this method should be double checked against the "live app experience" (actually download/use the app on a sandboxed device) because subtle

app code and platform changes can trigger these legacy code samples now opening links in the user's default browser and not within in-app browsers.

- Another method is to use a software tool to capture and log the network traffic via a proxy/Monster in the Middle (MiTM) connection, using a tool such as Charles Proxy helps to analyze network traffic from the app to third party sites and makes it possible to parse the vendor request and response headers and body payloads for unique ingestions or sharing of user data.

For this research, we utilized the last method to log and capture the network traffic, which we believe is more accurate as it captures the fine-grained details of the user data ingestions and sharing between the app and 3rd party vendors.

3.3 Data Collection Strategy

Our testers captured network traffic while using 36 apps representing 18 schools/districts. We conducted manual client-side testing which included navigating around the app, attempting to click all potential links within app menus, and exercising multiple functions in the app in order to find latent URLs triggered by deeper use of the app. Charles Proxy recorded all the domains connected to during the testing. See Appendix H for a sample list of 189 de-duped domains on the "Sports" tab powered by MaxPreps.com within just two Blackboard apps.

When we became aware of the high network activity on the sports pages in apps, we deliberately more deeply probed those pages, which we knew generated sizable ad related traffic.

The inherent limitations in this manual testing method are:

- It's not identical from app to app. Our purpose was to "click all the links" so our testers could attempt this until they were satisfied, yet this process does create unique results per app, and the order of clicking links is unique per test. This mostly impacts the analysis, and has minimal impact on whether advertising is shown or specific vendor selections within any one advertising supply chain.

- The testing didn't systematically execute every single possible path in the app user experience (UX). We may have missed other harmful links, so don't hesitate to let us know if you are familiar with any problematic kids' data brokers.

- Since external links branch off into numerous (potentially countless) other external sites, the testing didn't examine every externally linked branch of every possible path

in the app UX, down to its terminus. ***The numbers presented later in the findings section are illustrative only. The reality is that if we'd tested longer, the number of external advertising data transfers would continually rise.***

- The findings represent a single snapshot in time and would likely be different in quality if they were rerun today.

Automated testing could have removed the first limitation above but wasn't an option as automated testing would have prevented or impacted many of the ad syncs altogether, in addition to a number of other limitations. Automated testing would also have created other challenges and prevented our ability to detect key findings such as:

- Many schools have significant problems with their sports feature built via MaxPreps.

- Dangling domains are displayed to end-users in visually inconsistent ways, we would have likely missed one of the dangling domains being briefly available for sale for $30.

Tables 1 and 2 list the iOS and Android apps analyzed in our testing.

Table 1 iOS Apps

APP NAME	SCHOOL/DISTRICT NAME
Anchorage School District	Anchorage School District, Anchorage, AK
CHCCS	Chapel Hill-Carrboro City Schools, Chapel Hill, NC
Eanes ISD	Eanes ISD, Westlake Hills, TX
myMCPS Mobile	Montgomery County Public Schools, Montgomery County, MD
Santa Monica-Malibu USD	Santa Monica Malibu Unified School District, Santa Monica, CA
Thompson School District R2-J	Thompson School District, Loveland, CO

Table 2 Android Apps

APP NAME	SCHOOL/DISTRICT NAME
Anchorage School District	Anchorage School District, Anchorage, AK
Campus Parent	Hiawatha Leadership Academies, Minneapolis, MN
CHCCS	Chapel Hill-Carrboro City Schools, Chapel Hill, NC
MCPS	Montgomery County Public Schools, Montgomery County, MD
MCPS Connect	Montgomery County Public Schools, Montgomery County, MD
Montgomery Public Schools	Montgomery County Public Schools, Montgomery County, MD
ParentVue	Poudre School District, Ft. Collins, CO
Q Parent Connection	Dearborn Public Schools, Dearborn, MI
Santa Monica-Malibu USD	Santa Monica Malibu Unified School District, Santa Monica, CA
StudentVUE	Poudre School District, Ft. Collins, CO
Thompson School District R2-J	Thompson School District, Loveland, CO

3.4 Analysis

We conducted an analysis of the unique domains invoked by all the apps during the data collection.

It should be noted that once we realized the "Sports" tabs were full of online banner ads, and syncing unique vendors with every page load, several of the test sessions of these apps and data flows from the MaxPreps sports pages, were merely "refreshing the sports tab" for 5-10 minutes to capture a sample of the advertising vendors who received data within the app. We were not focused on capturing the brands buying ads within MaxPreps, but our screenshots below include major brands like Best Buy, Amazon, Cisco and Bertolli.

Note, however, that this was mainly an exercise in curiosity to provide a rough order of magnitude of the kinds of links being invoked. It should only be regarded as informational and not normative or prescriptive. If the tests had gone on longer, we would expect to find new advertising vendors, and our research into the data supply chains across MaxPreps content should be considered a "snapshot."

And important part of our analysis entails desktop traffic capture using the tool Trackermap from Evidon. Figure 2, below is an example of the "network traffic" captured from one of the MaxPreps school pages.

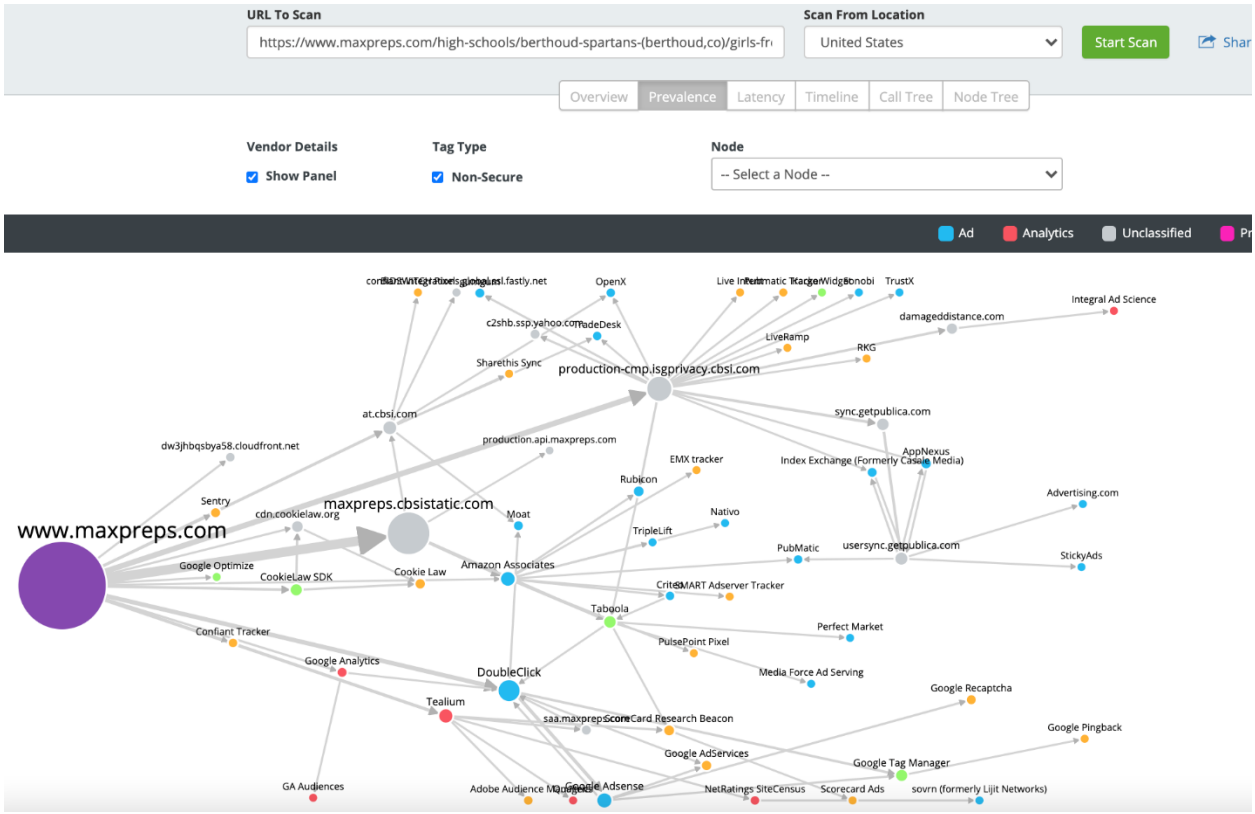


Figure 2 MaxPreps Network Traffic Map

This network traffic map merely demonstrates some of the controller, processor, and co-controller flows of user data across one page load and outside the context of the school utility mobile apps. Within the school utility apps, if a user refreshed the sports pages, numerous more vendors were being captured through the advertising syncs, which we captured in our live tests with real mobile devices.

When auditing any page or app with active advertising supply chains, it's important to remember that there are tens of thousands of advertising companies globally, and it's common for unique advertising vendors to receive data before an ad is shown, and then more unique vendors to receive data after an ad is shown. The advertising bidstream is also a constant flow of user data, and a whole separate can of worms. These advertising data flows create a significant sprawl of user data, and depending on where a test is being conducted, unique regional advertising companies are likely to receive data for any one specific advertising supply chain test. Due to these realities, the significant list of advertising companies who received user data from the MaxPreps advertising flows, within school utility apps, should be considered a baseline of data sharing with significantly more data sharing occurring for other users, in other locations, and at other times of the year. In short, what we found was very bad, but the reality is significantly worse.

For the advertising vendors we were able to capture with this process, we distilled all the unique domains invoked by the apps, identified the business owner of the URL, and assigned a risk rating similar to our SDK risk rating in Spotlight Report #1. Table 3 describes

the risk ratings for the unique domains. Note that "Malicious" maps to detected malicious redirects. (See this and this for more information on malicious redirects.)

Table 3: Domain Risk Categories

CATEGORY		RISK LEVEL	DESCRIPTION
M	Malicious	Extremely High	The URL maps to a hijacked domain, a malicious entity—such as a honeypot, designed to collect user data.
DD	Dangling Domain	Extremely High	The URL is currently for sale and at risk.
AD	Advertising	High	The URL is used for advertising, tracking, analytics, monetization.
LH	Legitimate, High	High	The URL is school-related service, but entails significant ad tracking, monetization, data brokers, etc.
LL	Legitimate, Low	Medium	The URL is school-related, and has moderate risks (par for websites)

4 FINDINGS

4.1 MaxPreps

Through our examination of the domains called by the school utility apps, we noticed one standout was present in every school utility app provided by Blackboard, Inc., which was MaxPreps.com. Maxpreps.com is a school sports scores and analysis platform owned by CBS Viacom, who are also the public media company who own the children's television channel Nickelodeon.

MaxPreps is ostensibly a valuable tool for parents, students, school staff and fans to follow their teams. The problem, however, is that it is a self-avowed advertising platform (see section 4.1.1), that provides no transparency, consent, or control for the aggressive advertising supply chains integrated into their pages, and subsequently integrated into school utility apps via WebView.

Troublingly, we found MaxPreps.com in apps for schools that had students likely under the age of 16, due to the inclusion of Freshman and JV sports data.[3]

MaxPreps provides a free service for coaches and team management tools, with marketing language on pages including: "High School coaches get FREE access to team management and communication tools." (source @ https://secure.maxpreps.com/utility/member/login.aspx)

[3] COPPA is 13 and under, California's CCPA expanded COPPA to 16 - and CPRA expanded it a bit further around data sharing.

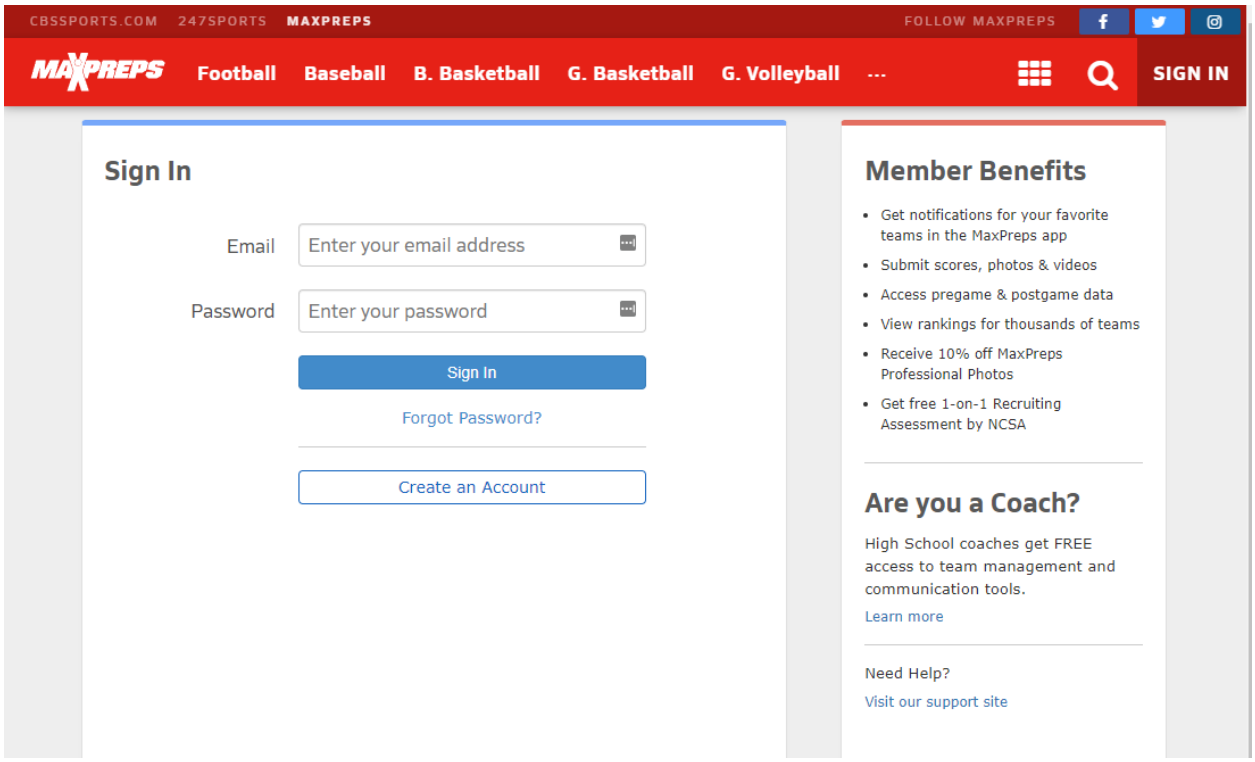
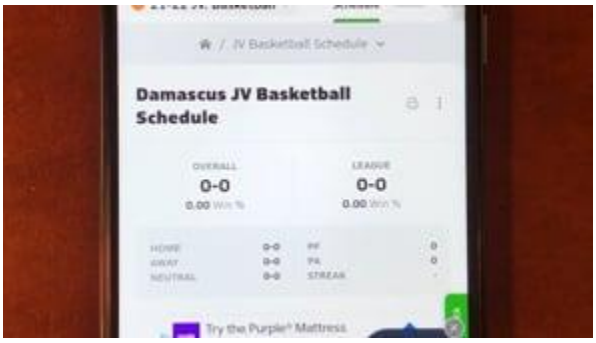


Figure 3 MaxPreps Sign In Page

4.1.1 Advertising on MaxPreps

Currently, MaxPreps offers 17 different advertising formats as shown in Appendix D.

The following is a video of navigating a MaxPreps-powered school sports page in a Blackboard app, with ads that were shown:



Video 1: MaxPreps-powered Sport Page in Blackboard App

From a high-level, there should be no taxpayer funded mobile apps for schools that have online advertising within them – this should be a hard and fast rule.

The inclusion of MaxPreps advertising vendors within in-app webpages within school utility apps demands that the advertising on these pages be "technically sandboxed" so that they

can't trigger pop-ups or malicious experiences/redirects/downloads -- and the types of ads shown in these apps should be contextual and significantly limiting any user data ingestions. Unfortunately, during our research, we captured both a malicious pop-up network that was breaking the functionality whenever it triggered, served via the Kargo SSP advertising vendor, as seen below in Figure 2, as well as malicious redirects and pop-ups within several of the "Sites" within the MaxPreps' Sports tabs that had custom domains, where apparently schools forgot to re-register the domain, and the domains were subsequently scooped-up by malicious advertising actors.

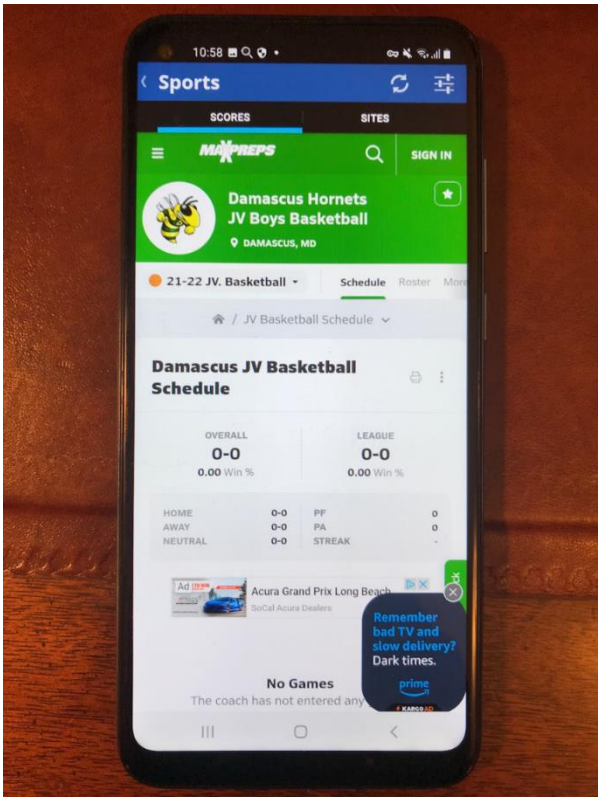
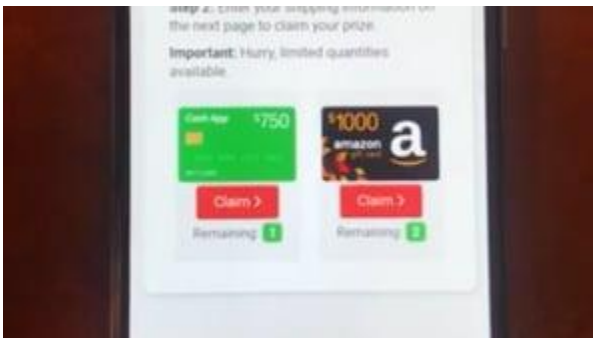


Figure 4 Kargo Pop-up on MaxPreps Page in Blackboard App

Here is a short video of the malicious pop-ups in a Blackboard app available to students, parents and teachers:



Video #2: Malicious Pop-ups in Blackboard School Utility App

MaxPreps also has advertising from Google on many of the pages, which includes behavioral retargeting ads and syncs people's data to numerous new advertising companies before and after the auctions. The full list of advertising vendors who received data can be requested by contacting the Me2B Alliance at services@me2ba.org.

The MaxPreps search pages (Figure 5, below, shown here on a desktop connection), which let you sort by "Schools," "Athletes" and "Videos" includes ads from the public advertising company Taboola, which consistently serves up inappropriate ads and links to low quality websites.

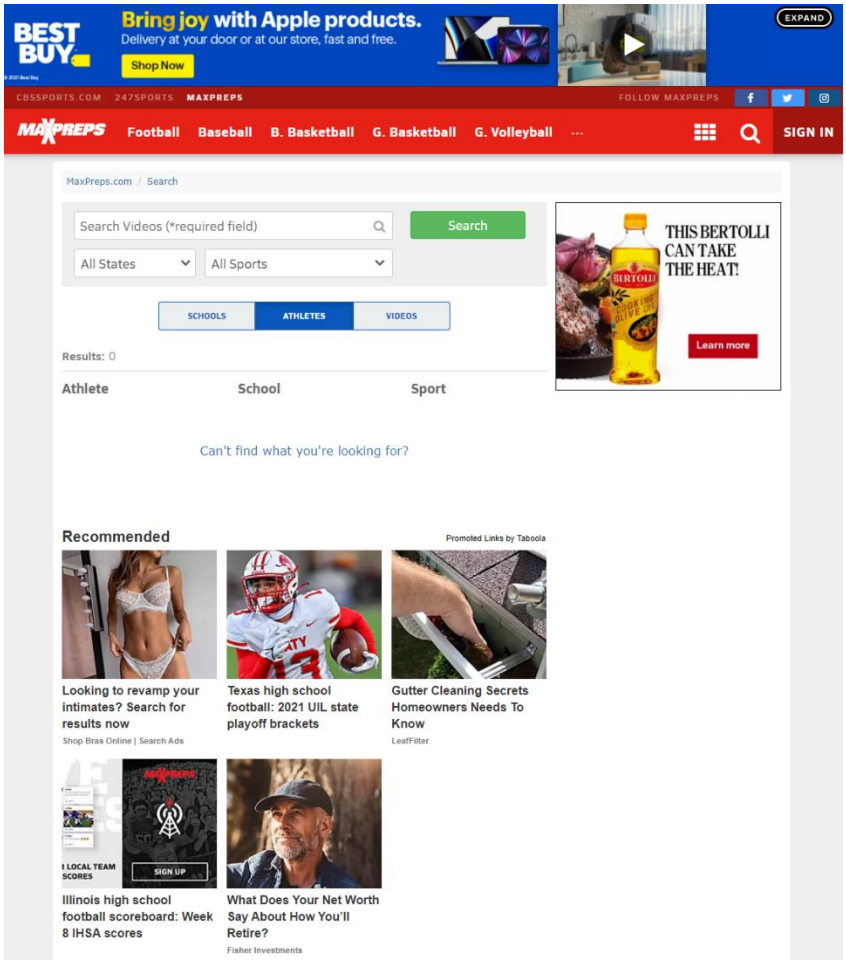


Figure 5 Example of Google Advertising on MaxPreps Website

4.1.2 How Many Schools Use MaxPreps?

According to the MaxPreps website, MaxPreps partners with over 200,000 school coaches throughout the United States (source: https://support.maxpreps.com/hc/en-us/articles/115004604407-About-MaxPreps). Based on the website's vast list of teams, thousands of schools in the United States use MaxPreps.

Several resources on the MaxPreps website flaunt the reach of the service. Figure 3 is a portion of the "Coach Handout", flaunting that the site "already has your school information.

Figure 6 MaxPreps Coach Handout

Note that you can use the public MaxPreps search interface (Figure 6) to find out if your school may be using MaxPreps within your school mobile apps.

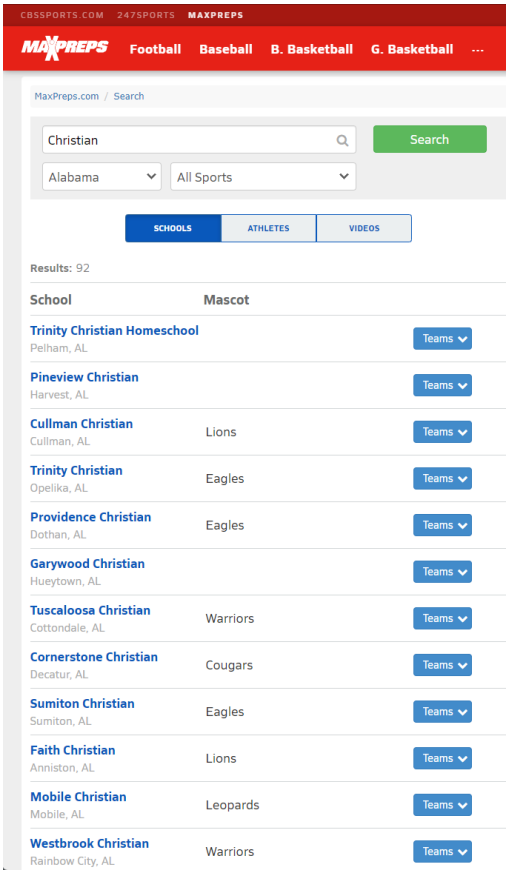


Figure 7 MaxPreps School Search

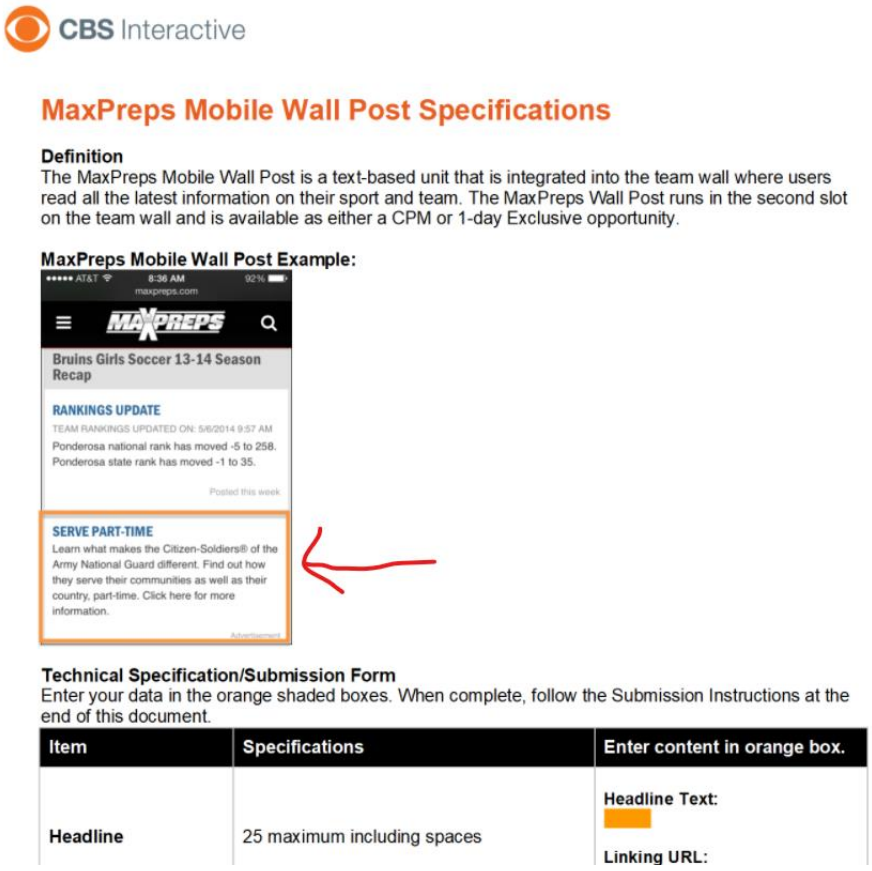
When attempting to determine the scale of schools potentially using this MaxPreps software, it became clear that MaxPreps claims to only work with High School students, but there are countless private and Christian schools who have K-12 curriculum, who are being "serviced" by MaxPreps. For example, Victory Christian Academy, in Chula Vista, California,

has a K-12 curriculum, and is frequently covered by MaxPreps. This raises questions about whether certain school apps have users under the age of 13 interacting with the “sports” tab connected to a MaxPreps scoreboard, filled with online advertising.

4.1.3 MaxPreps advertising being purchased by the U.S. Military

To our surprise, for at least some period of time, MaxPreps was used by the military to serve military recruitment ads to high school students, and one of the example advertising documents from CBS Interactive on MaxPreps is for the U.S. Army National Guard, with the text of the ad saying:

“Serve Part-Time
Learn what makes the Citizen-Soldiers of the Army National Guard different. Find out how they serve their communities as well as their country, part time. Click here for more information.”



CBS Interactive

MaxPreps Mobile Wall Post Specifications

Definition
The MaxPreps Mobile Wall Post is a text-based unit that is integrated into the team wall where users read all the latest information on their sport and team. The MaxPreps Wall Post runs in the second slot on the team wall and is available as either a CPM or 1-day Exclusive opportunity.

MaxPreps Mobile Wall Post Example:

Technical Specification/Submission Form
Enter your data in the orange shaded boxes. When complete, follow the Submission Instructions at the end of this document.

Item	Specifications	Enter content in orange box.
Headline	25 maximum including spaces	Headline Text: Linking URL:

Figure 8 MaxPreps Military Advertisement

Source: https://www.cbsinteractive.com/advertise/MaxPreps_Mobile_Wall_Post.doc

In 2015, MaxPreps was mentioned in 3 federal contracts from the Department of Defense, Department of the Army, with all three from the "...Readiness Center” - likely all focused on recruitment. (See Figure 9)

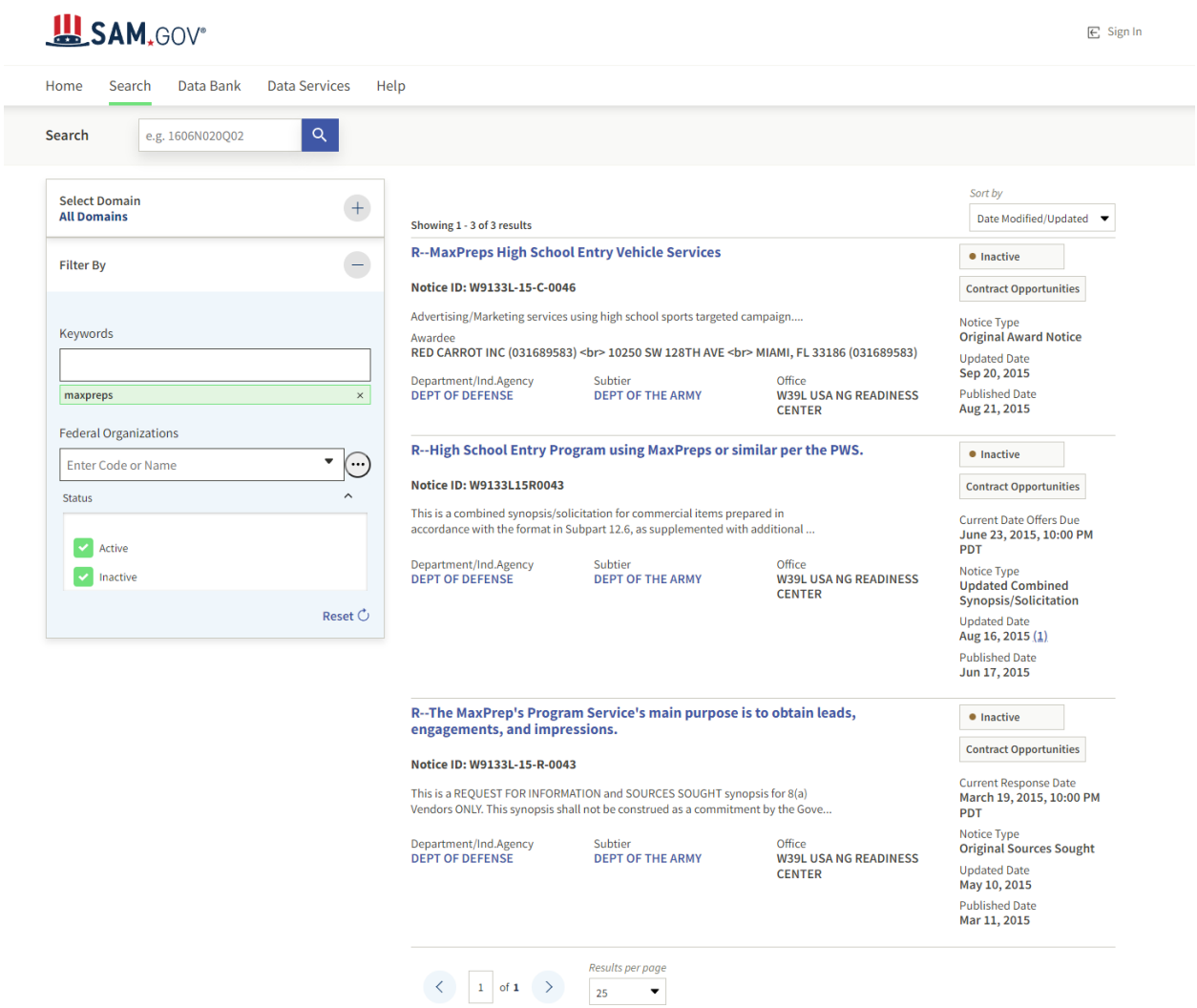


Figure 9 US Government MaxPreps Contracts

Source: https://sam.gov/search/?index=_all&pageSize=25&page=1&sort=-modifiedDate&sfm%5Bstatus%5D%5Bis_active%5D=true&sfm%5Bstatus%5D%5Bis_inactive%5D=true&sfm%5Bkeywords%5D%5B0%5D%5Bkey%5D=Maxpreps&sfm%5Bkeywords%5D%5B0%5D%5Bvalue%5D=Maxpreps

It appears only one contract that mentioned MaxPreps was awarded to a company called "RED CARROT INC" for $4,301,517 and the phrasing of the contracts makes it appear that only a portion of this $4.3 million would have gone to MaxPreps advertising, with the money being spent on other apps and websites. MaxPreps was however mentioned by name in the title of these Requests for Proposals. The details of the federal contracts can be seen here.

4.1.4 Obsolete Data Still Being Monetized with Online Advertising

MaxPreps seems to have profiles for private schools like the "Pathway Christian Academy" (see Figure 10 below) in Christiansburg, Virginia, and various pages for sports at this school (volleyball @ https://www.maxpreps.com/high-schools/pathway-christian-academy-

(christiansburg.va)/volleyball/home.htm) yet the Pathway Christian Academy seems to no longer exist and have been replaced by the Virginia Techniques Training Center .

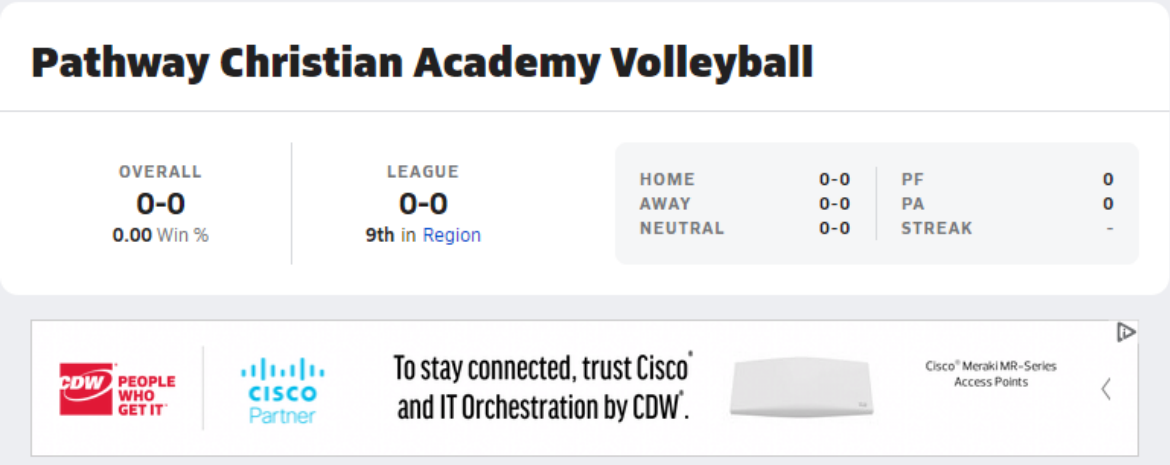


Figure 10 Obsolete School Data with Ads in MaxPreps

Why does MaxPreps continue to make new team pages for a school that seemingly doesn't exist anymore? Why are so many pages across MaxPreps empty, yet still serving advertising from major brands?

4.1.5 Other Noteworthy Behavior in MaxPreps:

- MaxPreps uses kids' photos on website pages that are filled with online advertising. For further example, please see Appendix E.

- We found it peculiar that MaxPreps heavily relies on freelance photographers, who license their photos to MaxPreps without compensation. Photographers are only paid a portion of sales when a photo is purchased from the MaxPreps website, and apparently not a portion of any advertising revenues. Please see Appendix F for more details.

- MaxPreps generates "profiles" for athletes without their knowledge or consent and prompts other users to "follow" the profile to get alerts for when new photos are added to the site. Please see Appendix G for more information.

4.2 Dangling & Hijacked Domains: An Unexpected Security & Privacy Risk

The research into the 18 apps unexpectedly revealed another serious and urgent set of risks caused by the use of WebView in mobile apps: expired, embedded URLs ("dangling domains") that have been acquired ("hijacked) by bad actors. When the domain is hijacked by a bad actor, the compromised website can result in an array of harms to students

through malicious website redirects[i] including flagrant collection of personal information for spam purposes, and links to inappropriate websites (such as pornographic content), or scam credit card/gift card offers.

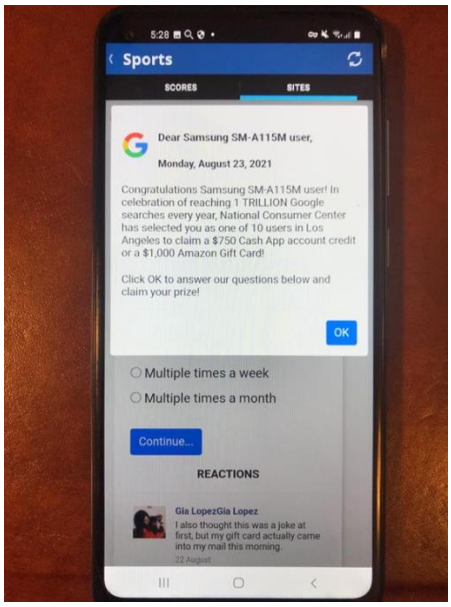
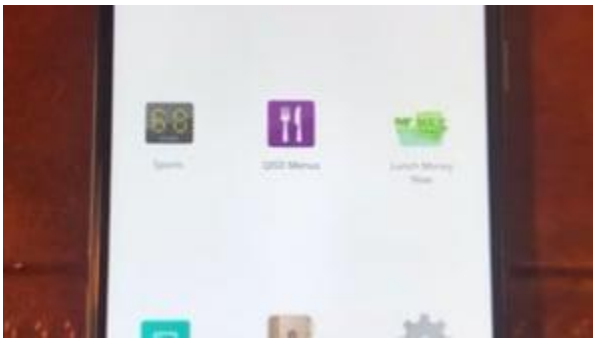


Figure 11 Hijacked Domain with an Amazon Gift Card Scam

Here is a video of a dangling domain within a Blackboard app:



Video 3: Dangling Domain in a Blackboard App

Figure 12 (below) shows a screenshot of a dangling domain within a Blackboard app for a school that was briefly available for purchase for $30. This was an example of a domain that was briefly lost and then the team at Blackboard, after being alerted to the problem by our team, were able to take action by removing the domain from the app and subsequently the Android app was pulled down from the Google Play Store.

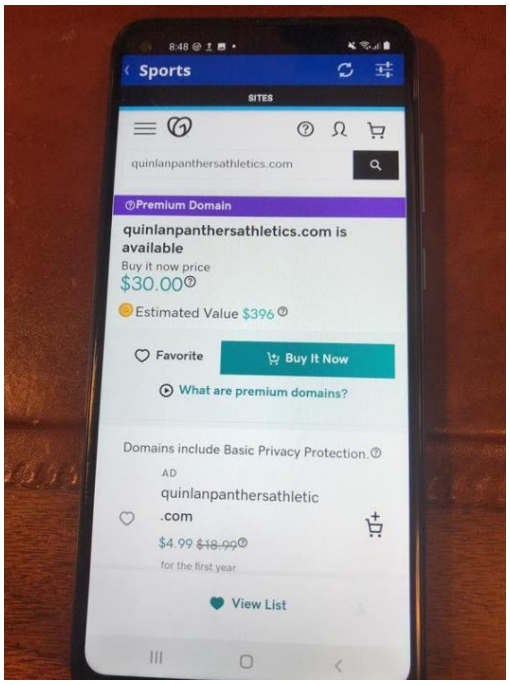


Figure 12 Dangling Domain

Our team reached out to the internationally recognized anti-malware researchers at Confiant.com with some early details about these compromised websites within apps. The Confiant team was able to confirm the problem existed and pointed our team to additional malware research that had found similar exploits in the past that triggered pop-up ads and other low quality advertising experiences. The Me2B Alliance deeply appreciates the support from Confiant and their work.

We are willing to discuss additional details and concerns with any U.S.-based cybersecurity companies who contact us.

After confirming that several U.S. schools had apps that were hosting compromised websites that injected scams and aggressively shared the student and parent user data, our team reached out to Blackboard's security team with detailed information about our disconcerting findings. Blackboard responded quickly, and we appreciated the thoroughness of their response: they confirmed early research, they alerted schools, and they removed links from existing apps.

It's unknown if any students or parents at the impacted schools were alerted, and the serious ongoing Business Email Compromise risks from the compromised school domains remain. This warrants more widescale notifications to school districts and states, as well as parents and administrators to alert them that emails from the compromised domains are illegitimate and potentially harmful.

Additionally, our team still has strong concerns about advertising companies who ingested data via these compromised apps, and we have concerns for any ad buyers who could have

been defrauded by these advertising companies allowing these monetization schemes to exist for criminal domain squatters and hijackers.

We could find no indication of a business relationship between Blackboard and MaxPreps, but MaxPreps was integrated into the "Sports" section of every Blackboard app we tested. Nonetheless, app template providers like Blackboard should be taking more steps to make sure their mobile apps are safe from extraneous data sharing from vendors like MaxPreps, or any other vendor who operates advertising networks. (We question whether any webpage that is run by an advertising network should ever be in a K-12 school app.) The safety of this supply chain and technical infrastructure is someone's responsibility (see also Section 4.4).

We support GDPR regulations and any privacy frameworks that place accountability squarely on the "data controllers", which we believe companies like Blackboard to be. The data flows that occur in the context of a WebView in-app browser are integrated into that app, no matter the domain integrated into that architecture by any administrator. The core app template architecture creates the risks, which we believe companies like Blackboard should acknowledge, so that they can make changes to their own app code to make these school utility apps safer. In particular, an app admin should not be able to turn off a privacy feature that impacts end-users of the app. Blackboard and any company operating in the school utility ecosystem needs to disable any features that allow this "safety toggling" concept.

As it stands, we're still waiting for companies like Blackboard to change their systems, but this process is deeply complex due to the shared ownership of their apps with client schools and school districts. Apple and Google can also make changes to their underlying operating systems in Android and iOS that would help all companies like Blackboard comply with user choice and preferences. ***If this change happened at the operating system level, it would save hundreds of millions of dollars in development costs for developers who want to give users the ability to select their own in-app browsers, without also having to rewrite all their legacy code and apps.***

We appreciate that companies like Blackboard are "between a rock and a hard place" and believe that raising these issues in a transparent way will help move these challenging debates forward.

4.3 Extensive Data Sharing Through WebView / In-app Browsers

Note: This section is provided as informational only and should be viewed as an illustrative "snapshot" of the findings from the manual testing sessions of the apps shown earlier in

Tables 1 and 2. This information is included to give a sense of the magnitude of the situation.

This current analysis exposed yet another data conduit to third parties, beyond the use of SDKs. Namely, the practice of including embedded browsers in apps, using WebView (in both Android and iOS apps). Mobile apps with embedded browsers can easily communicate and are sharing student data with an unbounded universe of third parties via URLs/requests from these WebView-invoked vendors – usually sent using JavaScript. The "context" of the app is sent with these third-party requests through a "referrer field" or "path field" which tells advertisers the name of the app being used, creating risks when the app is for K-12 students, parents, teachers and administrators.

As mentioned previously, users have no choice or control over the use of an in-app browser when the app developers overrule preferences, which means school utility app users can't configure browser privacy settings, or enable a privacy signal such as the Global Privacy Control signal. Both Apple and Android have some protections on in-app browsers, but they are minimal and the inability for users to always overrule an app and choose their browser of choice to open web links exacerbates the risks.

Linking to a single webpage within an in-app browser oftentimes results in numerous outbound requests to new vendors. This can go very badly (i.e. hundreds of potential third parties) when the linked webpage is run by an advertising entity who is hosting banner or video ads on the pages.

Troublingly, WebView is particularly common in school utility apps[4]. The sample size was small, but based on our past experience auditing school apps and awareness of vendors in this ecosystem, we believe that in-app browsers are prevalent due to this process costing less to build and maintain for the tech vendors, and being easier to maintain for non-technical school administrators.

To quantify the magnitude of the situation, through less than 3 "sessions" using each of the apps, we found:

- Over 650 *unique* external URLs were invoked in 11 Android apps, 73% of the links were advertising or analytics related, 9.5% of the links were ostensibly appropriate, but had an unacceptable level of advertising connected to them, and 3% of the links were hijacked domains leading to malicious sites.

[4] This will be further studied in our pending US benchmark of school utility apps.

- Over 550 *unique* external URLs were invoked in 6 iOS apps, 66% of the links were advertising or analytics related, 8% of the links were appropriate but had an unacceptable level of advertising connected to them

The number of external URLs students are sharing data with each day could easily number in the thousands.

Will each of these vendors be alerted that they could have ingested data from users under 13? Will the advertising companies like Google and Taboola, who hosted the banner ads on pages, be required to alert advertising vendors who synced data from any of these school apps? In California, will users under 16 have their data deleted? If the app can't tell the difference between an under 13/16 user and other users, should all the data be deleted and vendors encouraged to delete all the data they ingested under that "it could be under 13" threshold? Will these vendors who ingested data from these URLs be required to delete or modify any algorithms and models they optimized using data from these apps?

It's important to once again underscore the fact that this wasn't necessarily "apples to apples" testing, and the numbers, while large, are likely ***small subsets of the actual numbers***, given the limits of our testing and the reality of advertising data supply chains. There are no conclusions to be drawn with respect to "iOS versus Android" here. The value of the findings has to do with the sheer volume of third parties and the distribution of the *kinds* of URLs included. If our testing sessions were longer and exercised more functionality, it would have yielded an infinite list of linked URLs from advertising companies.

In short, if 73% of third-party URLs included in these apps are advertising related, this means that students' and/or parent's data is being shared with potentially thousands of advertising entities every time students use the app during the course of a school year.

4.4 Who is the Data Controller for Apps Built from Licensed School App Templates?

Our research from earlier this year showed that privacy policies for school utility apps were often inaccurate and inadequate. This current, deeper analysis, including a discussion with a team from Blackboard, underscored the reality of "data controller confusion" for licensed app templates.

In the case of school utility apps, the school is often the customer of the technology/platform provider, and in the supplier agreement, the school (licensee) typically is understood as the data controller for the app, and the technology provider (licensor) is absolved of any data controller responsibilities (following a Section 230 protections line of reasoning). Further, while the team conducting this research are not lawyers, functions like a radio button to toggle off security and privacy features in a CMS back-end dashboard may

possibly void Section 230 protections. The serious problem here, however, is that the school may not be aware of what those responsibilities are, and worse, may not really know they're responsible. Which begs a larger question of, *who is the data controller*—or perhaps more accurately—*who are the data co-controllers in the case of apps built from licensed school app templates? And who are the data processors? And when there are technical loopholes and missing safeguards to protect kids from being profiled by advertising companies, who along this data supply chain is required to take action?*

Important to note that this business model (licensed app templates) isn't unique to EdTech; this is a pervasive business model for which data controller/data processor roles between the platform provider and the licensee must—and likely will eventually—be clarified in the courts.

4.5 Key Insights - Summary

In summary, the key insights from this research are the following:

- Mobile apps using embedded browsers (both iOS and Android) present an alternate channel by which personal information can – systematically and at scale – be sent to third parties.
    - It is, therefore, insufficient to analyze only the SDKs in a mobile app to assess data flow to third parties. And due to the time-based reality of new advertising vendors being synced before and after advertising auctions, these types of data supply chains create *infinite risk* for students, parents, teachers and administrators.

- Mobile apps that include embedded browsers have been found to include hijacked domains with active malicious redirects. More care needs to be taken by schools both for the user data privacy and security implications, but also due to the massive potential harm from business email compromise schemes. It's our hope at the Me2B Alliance that security experts at the Department of Education, DHS, CISA and FBI are thinking about the risks from public and private schools losing access to their domains, and the ecosystem risks from this behavior being coupled with any criminal business email compromise schemes.

- There is a thriving practice for bad actors to constantly scan for expired URLs and purchase them, inserting malicious redirects for data collection (advertising/drive by download honeypots), or other harmful tactics. We observed in real time at least one school-related-dangling-domain get purchased by an unknown actor (i.e. not related to the school). Whose responsibility is it to prevent and who should be alerted when the domains are lost?

- Neither iOS nor Android app stores require developers to disclose either the use of WebView, or the included URLs in the app information label, making it impossible for users to understand with whom their data is being shared. Apps are also not required to prominently note the types of advertising within their apps, and whether advertising vendors who bid and win the chance to serve an ad in the app get an opportunity to sync additional data from the user.

- The providers of technology (including licensable mobile app templates) *are* accountable for data processing practices that occur within the context of their app template WebView pages, yet legacy development practices coupled with mobile operating systems not fully supporting user choice for preventing in-app browsers, are making it nearly impossible to keep non-technical customers' users safe from harm.

5 Me2BA Recommendations

5.1 Schools

1. All schools/districts that have students under the age of 13, and all schools in California that have students under the age of 16, who are using the Blackboard Mobile Communications App and who have added MaxPreps.com into the app:
    a. Immediately **remove the MaxPreps.com link** from their school/district apps.
    b. Immediately send an email to support@maxpreps.com and ask whether MaxPreps can support deletion requests for your schools' users to the advertising vendors like Google and Taboola.
2. School app administrators need to be trained on the risks of embedding URLs in apps.
    a. School app administrators should recognize that if they can't audit a page for advertising or analytics vendors, then they need to find help.
    b. Coaches and all school staff need to be warned about digital product monetization schemes and everyone should know the mantra, "if the product is free, you are the product."
3. Schools and school districts must have processes in place to:
    a. Keep track of all owned URLs, especially when embedded into mobile apps used by students, parents, teachers and administrators,
    b. Ensure that URLs aren't expiring (due to expired credit cards, etc.).
    c. Have a plan for what to do if a domain expires, because a compromised domain for a school could lead to significant Business Email Compromise or specific types of payment fraud against a State or the Federal government.

5.2 MaxPreps.com/Viacom

1. Immediately **remove all student data for children under the age of 16**, because it was obtained without consent or awareness and it could be impossible to determine what data is under 13 and what data is under 16,
    a. If it's impossible to determine which users are under the age of 13/16, all data should be purged that was received from school utility apps.
2. Immediately remove all of the online advertising from the pages integrated within all school utility apps until all behavioral retargeting ads are turned off, all inappropriate ads are turned off, and unsafe data transfers are turned off.
    a. We believe that a full removal of all advertising is the correct initial course and anything less than a full removal of the advertising from these pages will need to be coupled with concrete details about what is being done to give users control and transparency of the advertising data sharing.
3. Strongly consider not serving online advertising on MaxPreps webpages served within apps until Google and Apple update settings that gives students, parents,

teachers and administrators the ability to open in-app WebView links via a browser of their choice.

4. Consider ending the monetization of all MaxPreps.com content with online advertising, not only the content served within school utility apps, but the content on the website too.

5.3 Apple and Google

1. The fastest, most efficient way to reduce the serious risks of student data sharing to advertising companies, and malicious redirects through acquisition of expired URLS is for Apple and Google/Android to immediately **update their operating systems to allow users to designate a safe default browser for use by in-app embedded browsers, and not let developers overrule that choice.** The default browser choice that users should choose should be a browser that is configured to the highest level of privacy and security, such as the Safari, Mozilla's Firefox, or the Brave browser on iOS devices. Further, the default browser should also be one capable of sending the Global Privacy Control (GPC) signal, for maximum privacy (which Safari cannot do yet, although Mozilla's Firefox and the Brave browser both can send this additional GPC opt-out signal).
2. App privacy labels within the app marketplaces (or any marketplace with an approval process is the data controller for the user data flows within those apps that use marketplace/operating system-derived metadata) must include ***all*** third parties who are receiving data – including those receiving data via WebView data flows.
    a. This means the app label needs to include ***all*** URLs invoked in the app.
    b. Apps should not be able to claim that they don't track users, if they embed in-app WebView webpages with significant amounts of user data transfers to advertising companies, and active online advertising auctions within these pages.
    c. If an app is using online advertising, especially in school utility apps, there needs to be significantly more work done to document all advertising vendors within that data supply chain. If you are the data controller of the app and have partners who monetize this app with online advertising, you take on the obligation of documenting your data processors and co-data-controllers.

5.4 Developers of licensable, templatized apps (like Blackboard) with in-app browsers

1. Include security features in your app configuration service to keep your customers' users safe---especially when they are children. For example:

a. The CMS admin panel should default to—and possibly only allow—links to open in a separate browser. Why should admins be able to turn off a privacy and security feature?
b. System should automatically check if a newly entered URL is active before saving the entry.
c. System should automatically periodically (daily) check if client-entered URLs are still active and should alert the app administrator about any dangling domains, and disable any calls to them.
d. System should present highly visible and understandable warnings that expired URLs present serious security risks to their users.
e. System should warn users about the risks for embedded websites via in-app browsers, especially when training for these risks could be the difference between a safe way for students and parents to learn about their school sports program, and the current dystopic reality where an entire company has been created with "free products for coaches" which is actually a product being heavily monetized with online advertising – the kids were the product all-along.

5.5 Federal, State and Local Policy Makers

1. Schools need more training and support to understand and fulfill their duties as Data Controllers and Co-Data Controllers, especially when any student data could be shared with online advertising companies.
2. Schools need resources to be able to "check that their websites and apps are safe from unexpected external data transfers" - and if schools are using vendors who contract across numerous local, state and federal jurisdictions, providing software for a wide variety of schools, there should be federal standards to require these vendors to operate both in good faith and with an acknowledgement that when any utility app for schools is unsafe, it's usually just the tip of the iceberg.
3. The Department of Education should consider launching a "privacy bounty program" similar to technical bug bounty programs, to support the public spending time auditing school mobile apps and websites.
4. Schools and School Districts should be required to report lost or dangling domains within a specific period of time, so that the public and other government administrators know that emails from these domains can no longer be trusted. A public registrar of formerly-government-owned-domains would also be useful – the vast majority of these are on .com, .org & .us domain suffixes, as any .gov or .edu domains are apparently more easily recovered.

6 Appendix A – How to Spot WebView Pages in a Mobile App

Sometimes it's obvious to detect the launch of an external website from within a mobile app through the recognition of popular brand (like Facebook for example). Other times, it can be harder to distinguish. If you want to be exceedingly certain, we suggest following the methods outlined in Section 3.

Here are some ways to detect the use of WebView through the interface:

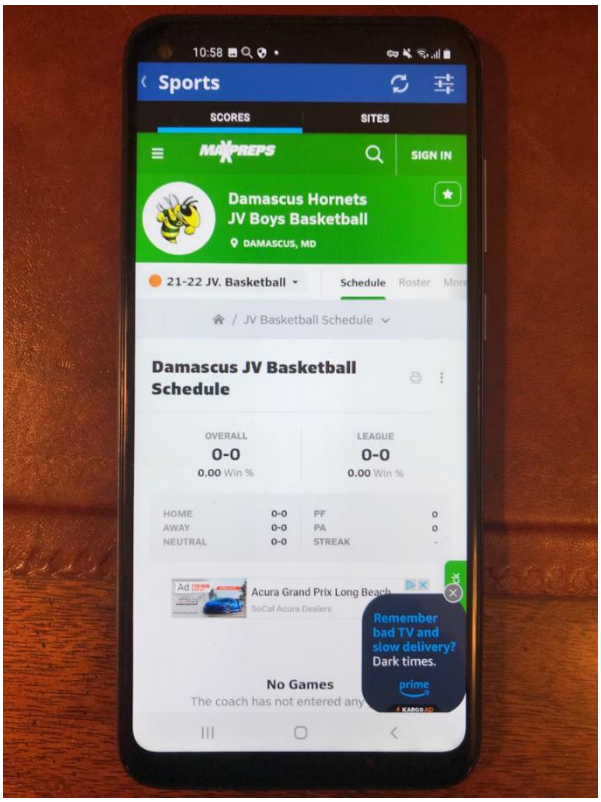


Figure 13 Example WebView Page

Things to look for on the page:

- Is there a different color scheme on the page?
- Is there a large, new logo and brand name on the page (like "MaxPreps" on Figure 13)?
- Did the page load differently / more slowly and piecemeal?
- Are there ads on the page (like the Acura ad in Figure 13)?
- Is there a new Sign-in button?
- Is there a separate "hamburger menu" button (usually 3 lines or dots)?
- Do you see pop-up advertising like the Amazon Prime ad from the Kargo SSP, seen above?

7 Appendix B – Data Flow Analysis Using TrackerMap

The screenshots below are from a paid tool from Crownpeak called "Trackermap" - the colored nodes are unique vendors receiving data. The vast majority of these data transfers would be "a surprise" to app makers and typically none of these data transfers are accounted for in Apple / Google privacy labels or other types of legal disclosures. Most lawyers don't know these data flows exist, and many app developers don't understand this either - and the people updating apps without technical knowledge rarely know about these risks.

When an in-app browser opens a WebView URL, this is represented in the images below with a large purple circle. All of the nodes flowing off this core circle, are the unique vendors who received data in the context of the app that opened the original URL within a WebView.

For some of the screenshots below, which are desktop simulated data flows from pages that opened within some of the WebView in-app browser sessions, some of the pages had minimal vendors receiving data – whereas on other pages, countless advertising vendors received data. When a page has numerous advertising vendors collecting data, this usually means there are banner ads on the page, and the potential for a near unlimited number of vendors to receive user data if refreshing a page and the ads on it regularly.

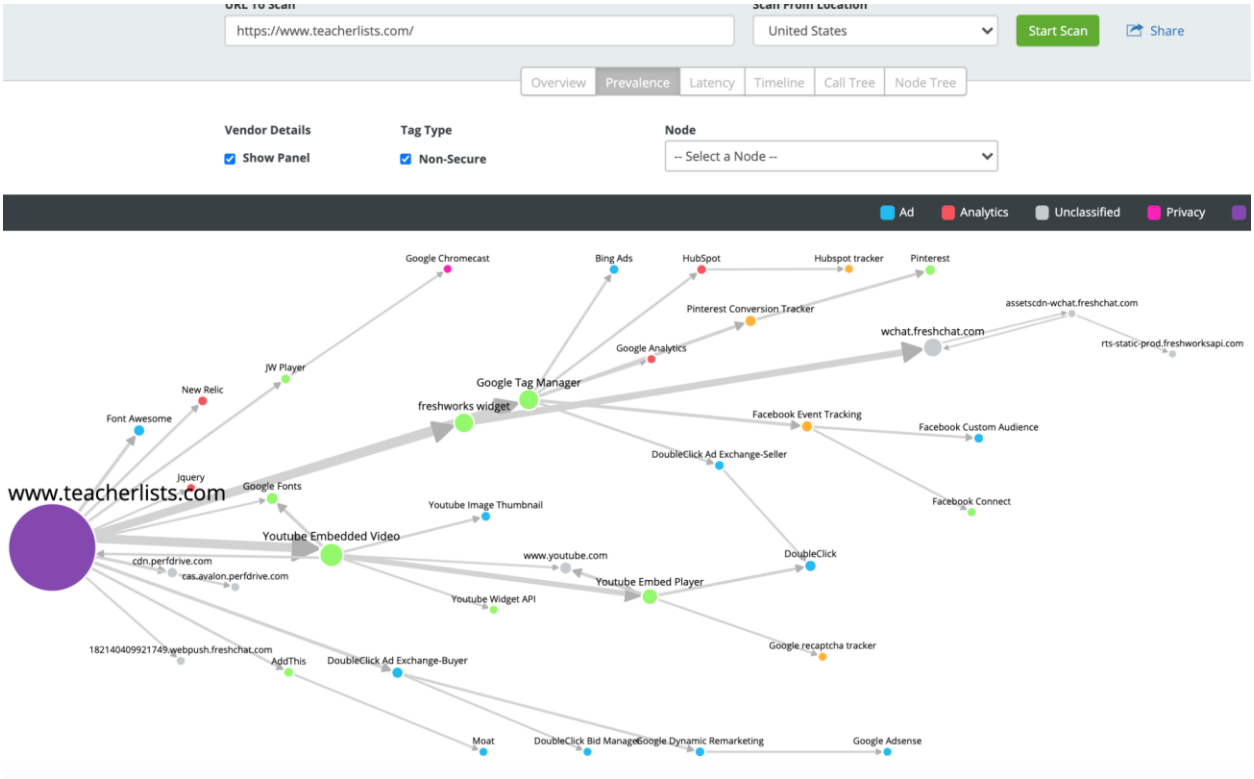


Figure 14 TrackerMap for www.teacherlists.com

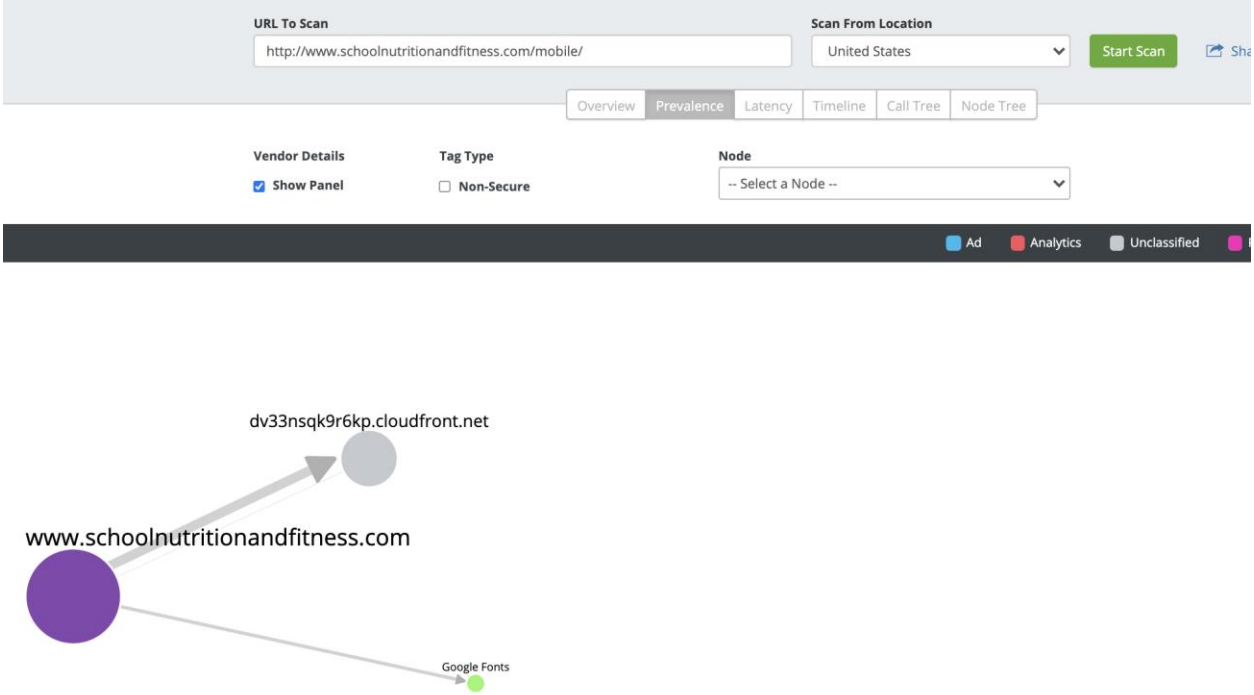


Figure 15 Trackermap for www.schoolnutritionandfitness.com

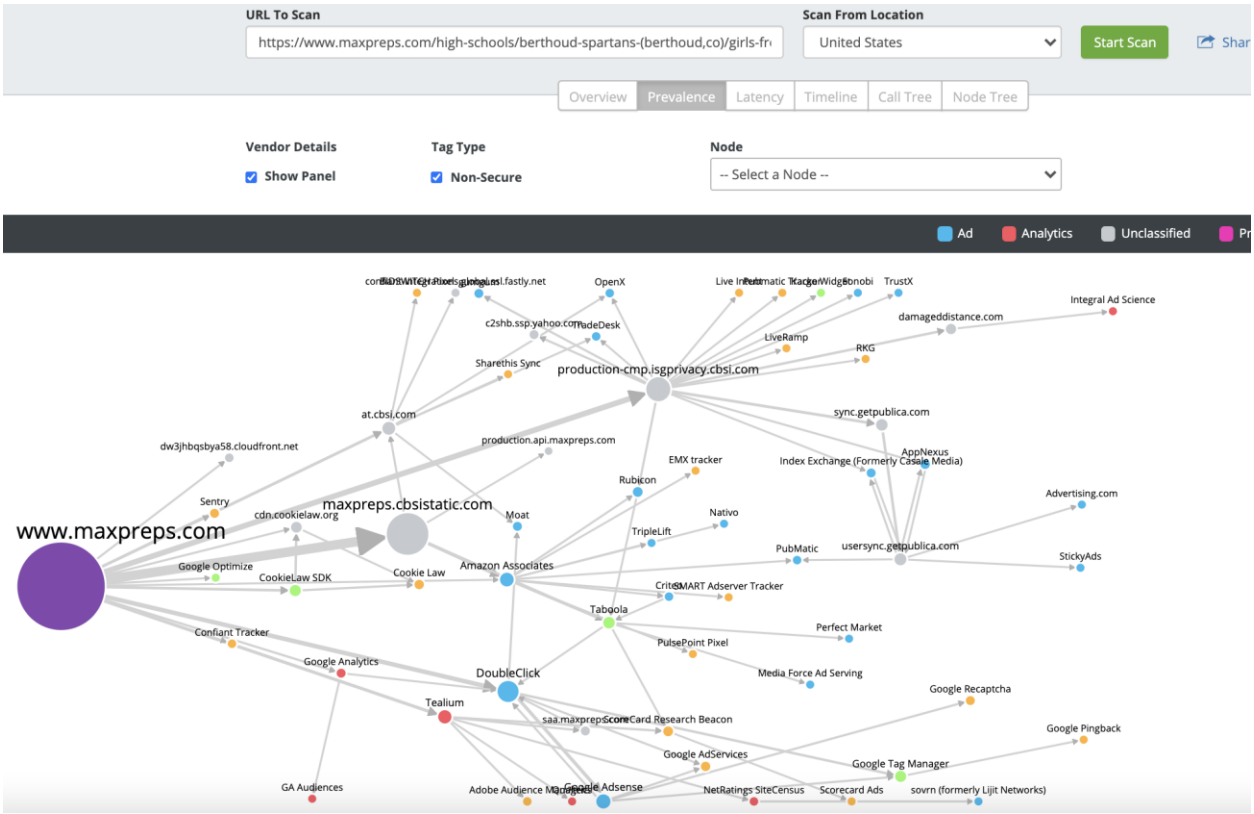


Figure 16 TrackerMap for MaxPreps

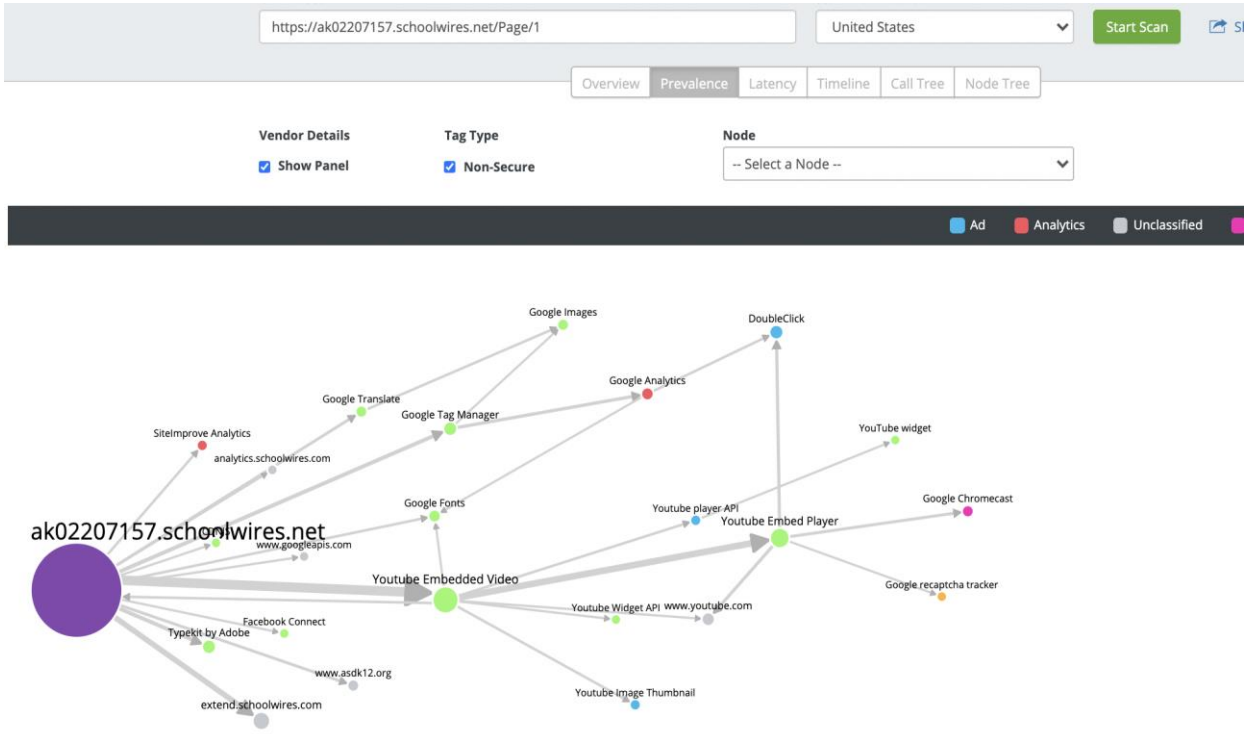


Figure 17 TrackerMap for ak02207157.schoolwires.net

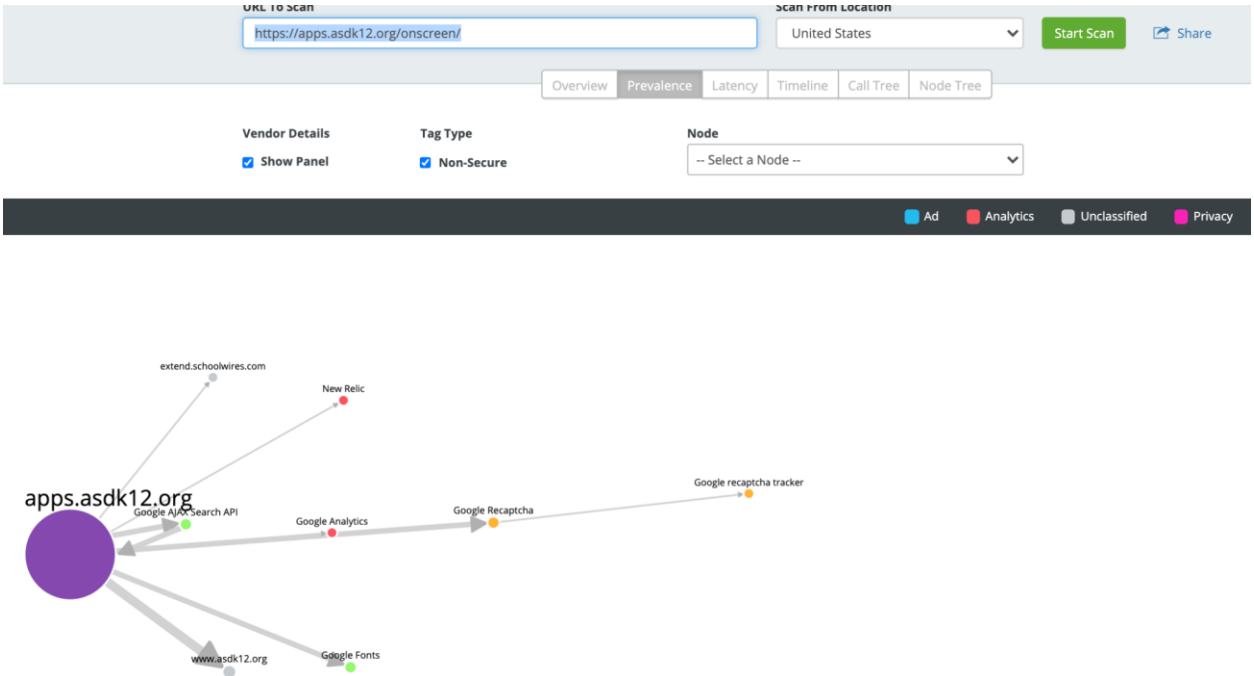


Figure 18 Trackermap for apps.asdk12.org

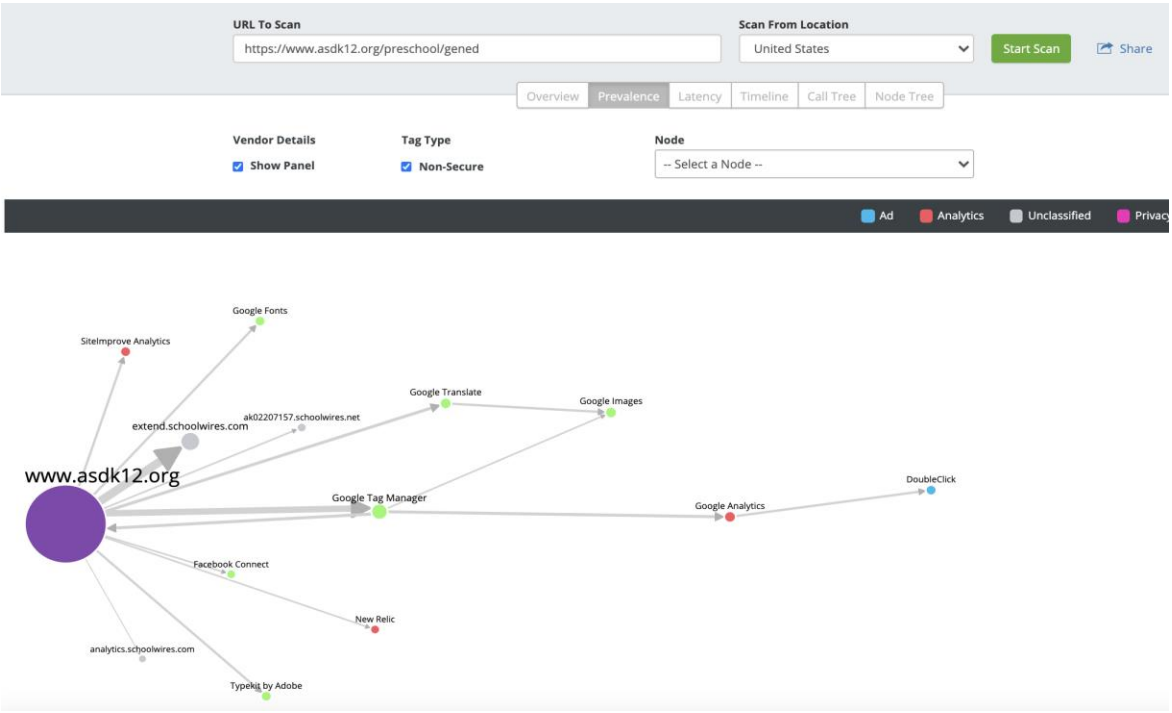
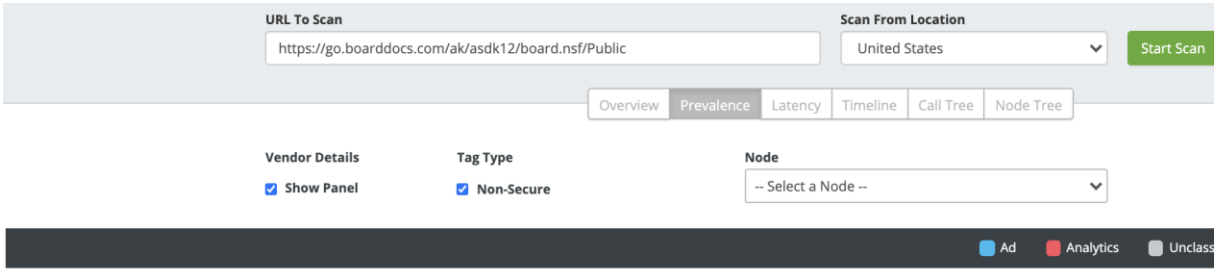


Figure 19 TrackerMap for www.asdk12.org



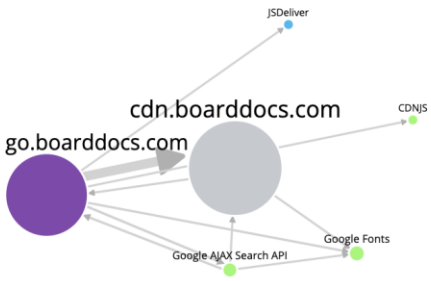


Figure 20 TrackerMap for go.boarddocs.com

8 Appendix C – Schools Included in Study

In this appendix we list the schools and school districts included in the research. Please contact us at services@me2ba.org if you'd like more contact information for these schools.

- Anchorage School District, Anchorage, AK
- Santa Monica Malibu Unified School District, Santa Monica, CA
- Thompson School District, Loveland, CO
- Poudre School District, Ft. Collins, CO
- Montgomery County Public Schools, Montgomery County, MD
- Dearborn Public Schools, Dearborn, MI
- Innovation Academy Charter School, Middlesex County, MA
- Hiawatha Leadership Academies, Minneapolis, MN
- Hasbrouck Heights Schools, Hasbrouck Heights, NJ
- Chapel Hill-Carrboro City Schools, Chapel Hill, NC
- Eanes ISD, Westlake Hills, TX
- Vashon Island School District, Vashon Island, WA
- Seattle Public Schools, Seattle, WA

9 Appendix D – MaxPreps Advertising Formats

MaxPreps offers 17 advertising formats (listed below). Of particular interest, note the "CBS Interactive Hosted Lightbox" (highlighted in yellow below), which is another target for mandatory deletion of student data.

- Display Standard
  - Half Page
  - Messaging Plus
  - Presenting
  - Super Leader

- Display Rich Media
  - CBS Interactive Hosted Lightbox
  - IAB Billboard
  - In-Banner Video
  - Network Pushdown
  - Third Party Expand

- Display Custom
  - Skybox
  - MaxPreps Wall Post
  - Native Ad

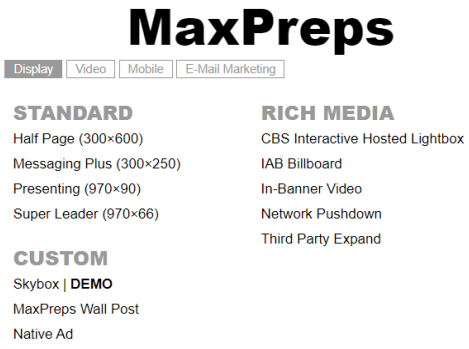


Figure 21 MaxPreps Wall Post Advertising Format

- Video:
  - CBS Interactive Hosted Video In-Stream Ad With Companion
  - 3rd Party Video Specifications

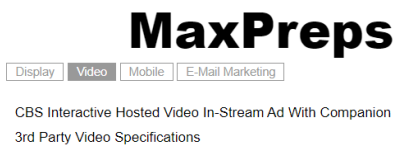


Figure 22 MaxPreps Video Wall Post Advertising Format

- Mobile
    - MaxPreps Mobile Wall Post
    - Mobile Opportunities

Figure 23 MaxPreps Mobile Wall Post Advertising Format

- E-Mail Marketing
    - Newsletter Opportunities

Figure 24 MaxPreps E-Mail Marketing Wall Post Advertising Format

MaxPreps and the CBS Interactive advertising "MaxPreps Wall Post Specifications" also showcases ads for "BeRecruited.com" - a website that claims to have 1,923,834 Active High School Athletes and 34,675 Active College Coaches - and likely over 2 million users total counting parents accounts.

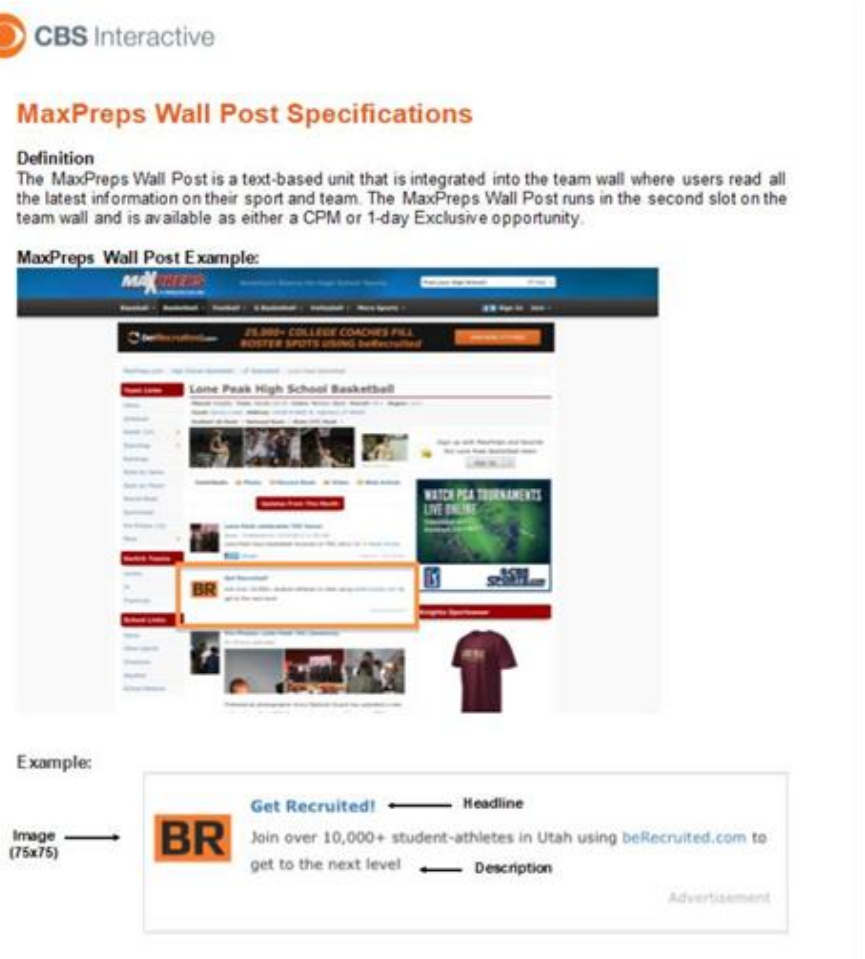
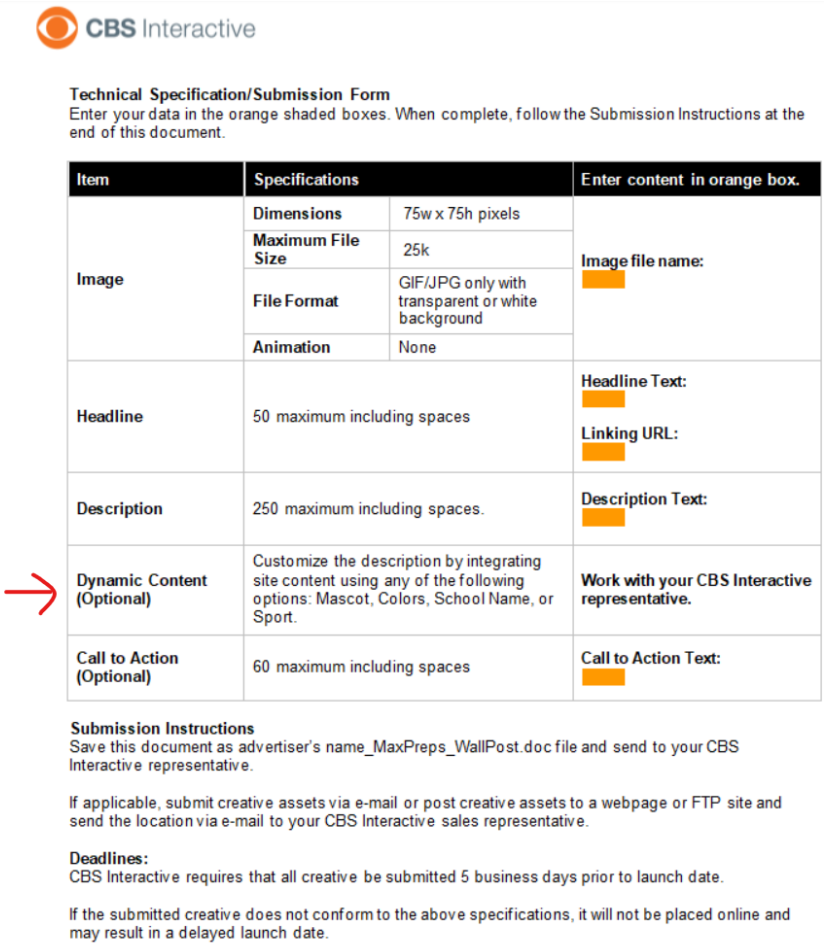


Figure 25 MaxPreps Wall Post Advertising

CBS Interactive's Technical Specification/Submission Form notes that advertisers targeting MaxPreps users are able to use "Dynamic Content (optional)" that lets the advertiser "Customize the description by integrating site content using any of the following options: Mascot, Colors, School Name, or Sport." MaxPreps advertisers are urged to "Work with your CBS Interactive representative" to use this dynamic content advertising feature.



CBS Interactive

Technical Specification/Submission Form
Enter your data in the orange shaded boxes. When complete, follow the Submission Instructions at the end of this document.

Item	Specifications		Enter content in orange box.
Image	Dimensions	75w x 75h pixels	Image file name:
	Maximum File Size	25k	
	File Format	GIF/JPG only with transparent or white background	
	Animation	None	
Headline	50 maximum including spaces		Headline Text: Linking URL:
Description	250 maximum including spaces.		Description Text:
Dynamic Content (Optional)	Customize the description by integrating site content using any of the following options: Mascot, Colors, School Name, or Sport.		Work with your CBS Interactive representative.
Call to Action (Optional)	60 maximum including spaces		Call to Action Text:

Submission Instructions
Save this document as advertiser's name_MaxPreps_WallPost.doc file and send to your CBS Interactive representative.

If applicable, submit creative assets via e-mail or post creative assets to a webpage or FTP site and send the location via e-mail to your CBS Interactive sales representative.

Deadlines:
CBS Interactive requires that all creative be submitted 5 business days prior to launch date.

If the submitted creative does not conform to the above specifications, it will not be placed online and may result in a delayed launch date.

Figure 26 CBS Interactive Advertising - Targeting MaxPreps Users with Dynamic Content

Source: https://www.cbsinteractive.com/advertise/MaxPreps_Wall_Post.doc

10 Appendix E – MaxPreps Uses Online Advertising with Kids Photos

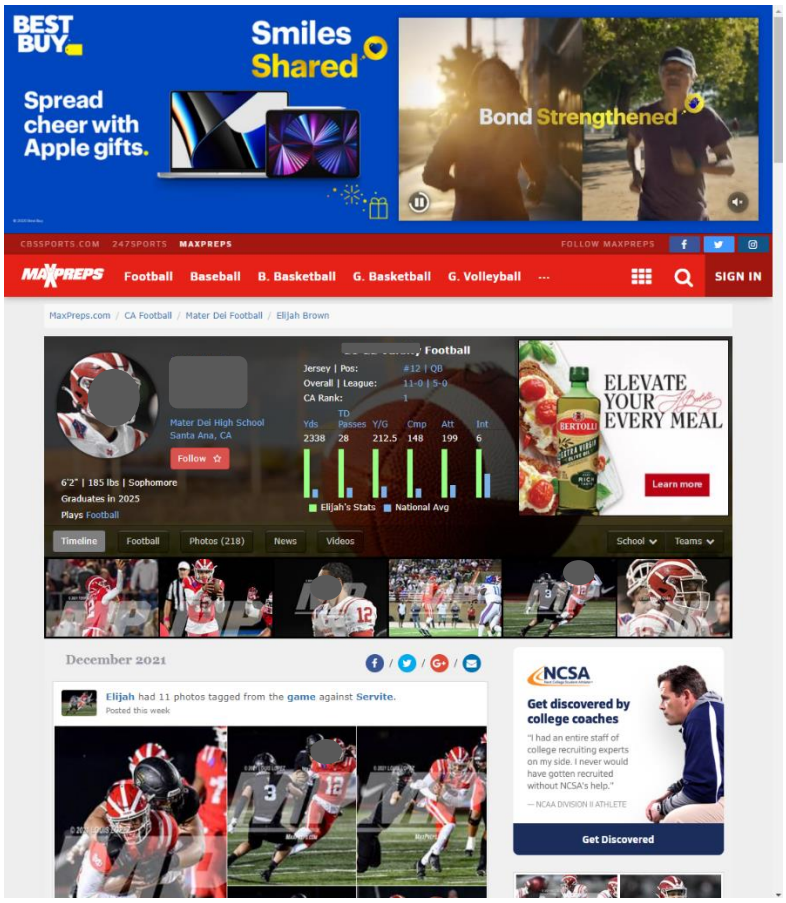


Figure 27 Example MaxPreps Screen from School Utility App

Source: https://www.maxpreps.com/athlete/elijah-brown/y4NzDVht_0-bAB3evH1_CA/default.htm

11 Appendix F – MaxPreps Freelance Photography

All MaxPreps photographers are considered freelance and do not receive a salary for any photos. There are over 290 photographers who have licensed photos to MaxPreps without receiving compensation.

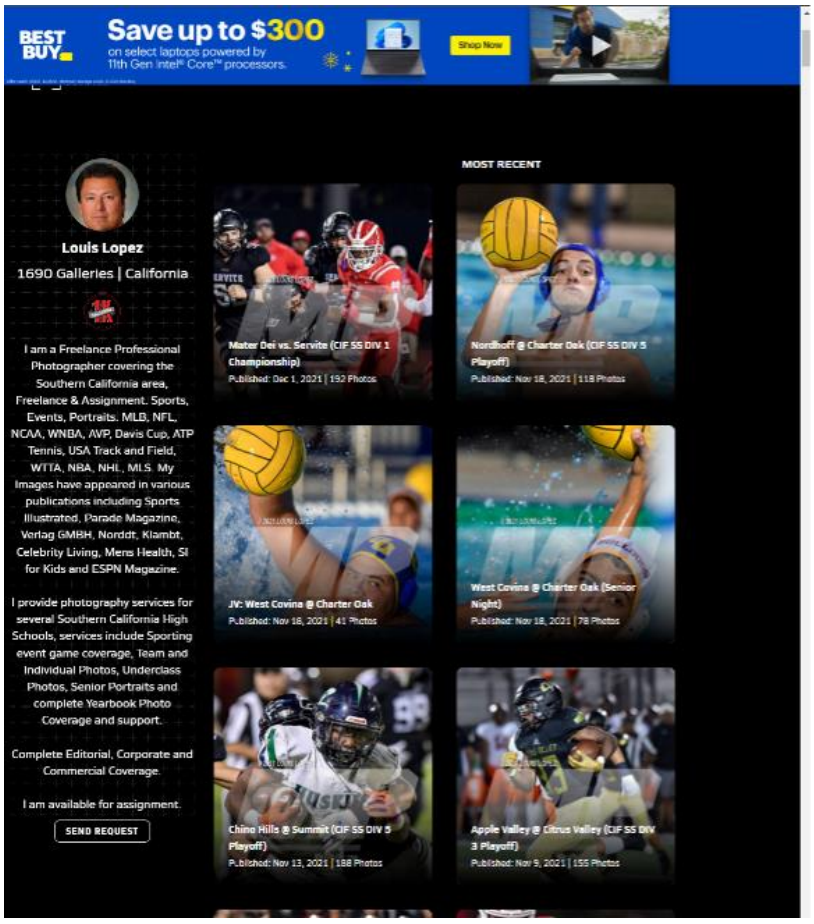


Figure 28 MaxPreps Freelance Photography Page

Source: https://www.maxpreps.com/photography/photographers/

12 Appendix G – MaxPreps Athlete Profile Page

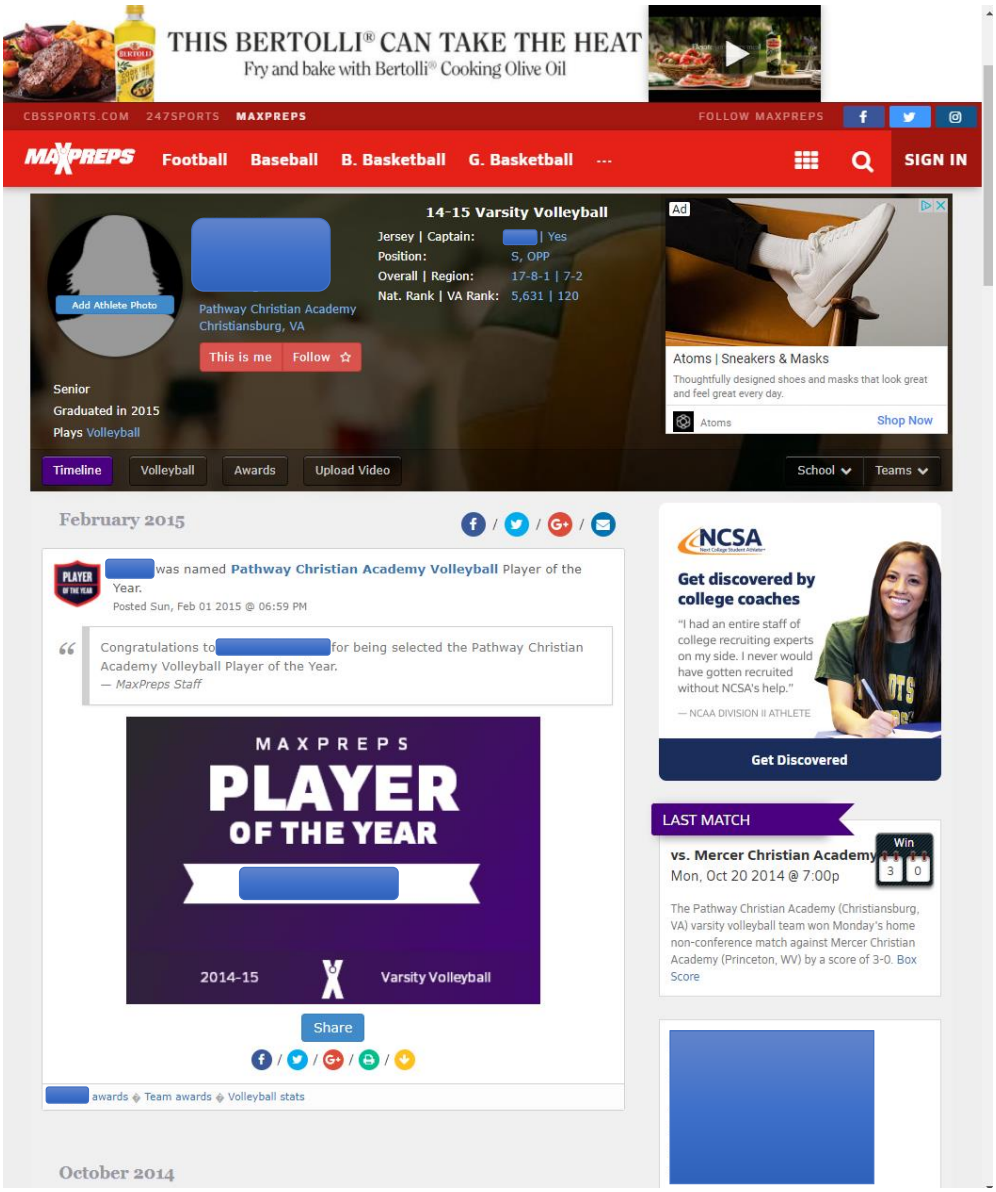


Figure 29 MaxPreps Player Profile Claim Page

MaxPreps supports the ability to "claim" a profile (example: https://www.maxpreps.com/athlete/emily-turpin/TqYpSxCoEeOZ5AAmVebBJg/default.htm) with a button that says "this is me" right next to the (questionably appropriate) button "follow" which shows up for most or all of the student athlete pages.

13 Appendix H - De-duped Advertising Domains Captured within Two Blackboard Apps in MaxPreps "Sports" Tab

The following list of 189 domains, primarily advertising and advertising related, were found in just two Blackboard apps, within the Sports tabs, both hosted by MaxPreps:

15.com
1rx.io
2mdn.net
360yield.com
3lift.com
9zpg.net
accesscontents.com
acuityplatform.com
addthis.com
addthisedge.com
adform.net
admanmedia.com
adnxs.com
adobedtm.com
adpredictive.com
adrta.com
adsafeprotected.com
adsrvr.org
advangelists.com
advertising.com
agkn.com
allin1-digitalcontent.net
alphonso.tv
amazon-adsystem.com
ampproject.org
app.link
apxlv.com
at2010.net
atdmt.com
azurewebsites.net
b2c.com
bazaarvoice.com
beevakum.net
bestmegaoffer.com
betweendigital.com
bfmio.com
bidr.io
bidswitch.net
bigteams.com
blackcrow.ai
bluekai.com
bnmla.com
bootstrapcdn.com
bounceexchange.com
bouncex.net
branch.io
bttrack.com
btttag.com
casalemedia.com
cbsi.com
cbsistatic.com
cdn925.com
cdninstagram.com
cdnwidget.com
celtra.com
claimprizesnow.com
clarium.io
clicken.us
clicktale.net
clientgear.com
cloudflare.com
cloudfront.net
cnnx.link
cnstrc.com
cogocast.net
contextweb.com
cookielaw.org
crashlytics.com
creativecdn.com
criteo.com
criteo.net
crwdcntrl.net
damageddistance.com
demdex.net
doubleclick.net
doubleverify.com
emxdgt.com
everesttech.net
exelator.com
extremereach.io
facebook.com
facebook.net
fastly.net
fksnk.com
getpublica.com
gglcdn.net
glotgrx.com
google-analytics.com
google.com
googleadservices.com
googleapis.com
googleoptimize.com
googlesyndication.co
googlesyndication.com
googletagmanager.com
googletagservices.com
gstatic.com
gumgum.com
igodigital.com
impactradius-event.com
imrworldwide.com
imtwjwoasak.com
inmobi.com
inmobicdn.net
instagram.com
js7k.com
kampyle.com
kargo.com
krxd.net
lightboxcdn.com
linkedin.com
loopme.me

lporirxe.com
magruderathletics.org
mathtag.com
maxpreps.com
media6degrees.com
mfadsrvr.com
microsoft.com
moatads.com
moatpixel.com
mvhsathletics.org
nationalconsumercenter.com
omtrdc.net
onetrust.com
openx.net
parentlink.net
perfectmarket.com
petco.com
pinterest.com
pippio.com
pleasewait.ws
pointmediatracker.com
pubmatic.com
pushnami.com
qualtrics.com
quantcount.com
quantserve.com
resetdigital.co
retailproductsusa.com
rewardzoneusa.com
rlcdn.com
rocketgate.com
rtactivate.com
rubiconproject.com
rzucscenter.com
sc-static.net
scanscout.com
scorecardresearch.com
semasio.net
sentry.io
simpli.fi
sitescout.com
smaato.net
smartadserver.com
smrtb.com
snapchat.com
sonobi.com
spnccrzone.com
spotxchange.com
springserve.com
stickyadstv.com
survata.com
surveysandpromoonline.com
syn-cdn.com
taboola.com
tapad.com
technoratimedia.com
therewardboost.com
tiqcdn.com
tk0x1.com
tremorhub.com
trueleadid.com
truoptik.com
trustx.org
tudum.co
tvpixel.com
twimg.com
twitter.com
unltdentertainment.co
unrulymedia.com
videohub.tv
whatfix.com
wknd.ai
xg4ken.com
yahoo.com
yimg.com
your.vet
zemanta.com

[1] Footer