Exhibit 236

From: [redacted]
To: [redacted]
Sent: Mon, 27 Feb 2017 23:37:44 -0800
Subject: Re: [redacted]
Cc: Jen Chai [redacted] Marlo McGriff [redacted]

[redacted]
REDACTED - PRIVILEGE

Let's say, if the users actually wanted this fine-grained control over which app gets what permission. [redacted] challenge you to sketch out a solution by which device-level permissions could be enforced server-side. Remember: besides Android, we have iOS, desktop, Chrome, and a bunch of other surfaces where notions of an "app" might not even exist. Personally, I can't think of a world where we do a good and thorough job with runtime permissions across Google apps that doesn't confuse the hell out of our users and make the lives of eng and PM folk hell - by fragmenting the user base into dozens if not hundreds of odd states where the data can't flow in some directions.

I'm down to brainstorm this more with L+C and Android Platform. Like I said, Google isn't the only one 'publisher' of apps where data is shared across. Facebook and Uber are very much in the same bucket, and it's definitely broader than just location (e.g., contacts between FB main app and FB messenger seems like a very sensible data to share).

[redacted]

On Mon, Feb 27, 2017 at 6:16 PM [redacted] wrote:

> The WAA opt-in covers gaia-keyed, long-term retention of user activity with 1st party Google products. It was originally known as the search history opt-in. In practice includes web search, maps, images, news, assistant, etc., and is used to control what winds up in personal logs for those products. It is an account-level control that is enabled by default for new accounts (except dasher). About 85% of signed-in search users have it on.

[redacted]
REDACTED - PRIVILEGE

> live independently of what is retained in location history.

On Mon, Feb 27, 2017 at 7:28 PM, [redacted] wrote:

> On Mon, Feb 27, 2017 at 3:32 PM, Jen Chai [redacted] wrote:

Got it, thanks [redacted]
What's WAAH opt-in? Is this a prompt that shows up during set-up? Device level or account level? One opt-in for all Google apps?

This is WAAH: https://myactivity.google.com/myactivity
This is [redacted]portal where all the toggles are for WAAH, LH, etc: link.

I believe WAAH opt-in happens pretty liberally from a number of places, including Android setup (when Google Now is activated).
Here are a couple of slides to give you an idea of what's in WAAH and who uses it. ([redacted] correct me if only a subset of those is WAAH)

Thanks!
Jen

On Mon, Feb 27, 2017 at 3:20 PM, [redacted] wrote:

Thanks, Jen.

Second scenario is not exclusive of ULR. It has to do with Web & App Activity History (aka WAAH) opt-in. A user can have both WAAH and LH - and many users do.

[redacted]
REDACTED - PRIVILEGE

Btw, don't view it as a bad thing. It's something that happens and is highly desired by Google apps - that's why we rolled out the unifying privacy policy a few years back.

Also, I'm fairly certain that this cross-app data sharing is it unique to Google.

Sorry, typo. I meant to say I'm fairly certain that the cross-app data sharing is NOT unique to Google.

Concrete (hypothetical, but very plausible) example: Uber and Uber Eats. Uber app has a permission to get your location. Let's say they analyze your usage and establish your daily commute routine (aka user location model). You install Uber Eats and decline it access to your location on device. Uber Eats can still determine where you are based on the user location model established from locations collected via the main Uber app.

Another example (doesn't even have to be different apps): let's say you have two devices, both running Google Now. One has location permission and one doesn't. Can Google Now give you push notifications to both devices based on locations collected on just one device?

Apps like Facebook and Uber are very likely already doing it as well, via own backends. I'm curious if Android Platform has a position on solving this more broadly.

On Mon, Feb 27, 2017 at 13:51 Jen Chai [redacted] wrote:

Sure, we'll follow up with [redacted] I'll include you in the review as well and you can fwd if needed (it's usually Fridays at 10am, but we have a backu[redacted]or the next few weeks, so I'll have to verify if he can attend this week).
Just so I'm clear - this use case is when the user has opted into ULR, but opted out of the app-level location permission for a Google app. That Google app could be getting location from the server through ULR/ PV/[redacted]. Correct?

What's the second scenario? If the user doesn't have ULR, but is using Google App 1 and that app is saving the location points it gets and shares it in the backend with Google App 2 (who does not have app-level location permissions)? What is [redacted] Footprints [redacted]

[redacted] is protobuf that encapsulates device location once it gets to a Google server. It's just a unified way to pass device location around servers. [redacted] part of [redacted]

Footprints is essentially where WAAH data is stored and processed.
[redacted] a backend built by Google Now team that essentially collects a bunch of signals (including current location) in both ephemeral and persistent way. I'm getting into the weeds here, so I hope [redacted] would make a drawing showing how all of these things relate to one another.

Thanks,
Jen

On Mon, Feb 27, 2017 at 12:16 PM, [redacted] > wrote:

On Mon, Feb 27, 2017 at 9:14 AM, [redacted] > wrote:

On Mon, Feb 27, 2017 at 8:49 AM, [redacted] n> wrote:

[redacted]

Great timing. [redacted]nd I just had a conversation about location logging (in WAAH or LH) and this particular question is about location use across products (via backend channels). [redacted]going to take a look at what products do the logging in WAAH (Footprints).

My understanding is that nothing is current stopping one Google product (A) from using location logged by another Google product (B) if [redacted]pproves the A<-B use. In fact, our landmark privacy policy came into existence pretty much for the very reason of enabling cross-product data use. That's why a separation on Android doesn't really make sense: Google products are currently allowed to [almost completely freely - with PWG oversight] exchange data[redacted]do you have any thoughts on this?

For runtime permissions, we explicitly ask the user "Allow XYZ to use location?" and the user can select Deny. I feel like we may hurt user trust if we are providing locations to XYZ via the [redacted]-loophole when the user has explicitly denied it. It's possible that it may not be a serious concern, but I'd be more comfortable if we have an "official policy" on this for M/N/O (in addition to planning for P/Q). Maybe discuss this in the next [redacted]neeting? **Jen**, could you please add this to the agenda?

Sounds good. To be clear - this loophole affects all apps/products sharing data on the back end.[redacted]

I don't have a very good understanding of (2) yet.[redacted]over that area and[redacted] (cc'd) was interested in following up as well. We captured some manifestations of (2) in [redacted]so we'd like to get into next level of detail there.

I doubt that we'll find any kind of runtime permission checking on the back end; e.g., between GMM, Now, and Websearch, I'm fairly certain the location data is shared freely regardless of where it came from. I don't even think there is a technically coherent way to implement cross-app permissions outside of mobile world (how would we even do it? would the users even want it? if we had it, would it be easy to understand for anyone?).

So, it stands to reason that we should try to get the user story right on Android; Apple/iOS will probably follow (e.g., how does Uber do this?).

Jen, [redacted]- would you keep us updated on wha[redacted]says?
Perhaps one or two of {myself, Marlo, [redacted] could attend.

Btw, [redacted]do you have a sense of how permissions are enforced by Context Manager in GCore? E.g., can a 1p app (GSA/GMM/etc) get recent locations collected by ULR via Context Manager if it doesn't have direct access to device location?

[redacted]

My thinking is that we should go in the direction of [redacted]

Agreed.

[redacted]

On Mon, Feb 27, 2017 at 8:20 AM, [redacted] wrote:

> Agree the issue is not necessarily specific to [redacted]
> Do we have a list of known products not checking for permission?

On Feb 27, 2017 08:04, [redacted] wrote:

> Hey [redacted] this loophole existed for 2+ years. There were a bunch of products already out of compliance, before even [redacted] existed.
> This is our chance to fix it.

[redacted]

On Sun, Feb 26, 2017 at 10:19 AM, [redacted] wrote:

> In addition to solving for P/Q, we need a stopgap solution [redacted] Can we list all the clients of [redacted] and make sure they are complying with the runtime permission?

On Fri, Feb 24, 2017 at 8:09 PM, [redacted] wrote:

> Hi [redacted] you are referring to this loophole, right? If so, kudos to you for you being aware of it!
> It has been nagging us for ages and I'd like to see how we could structure the new Android P/Q permissions such that the loophole is closed.

I suggested that we develop something along the lines of a [redacted] "can Google have

your Location?" - and that permission is applied, as an umbrella, to all other Google apps. There are some caveats, of course (e.g., does it apply to YouTube? Waze?), but I think this is the cleanest way to solve the problem.

Jen, if you aren't aware of it, let's chat some more. I'd really like to solve it all together as we move forward with the new Device Location permissions and [redacted]

[redacted]

On Fri, Feb 24, 2017 at 4:42 PM, [redacted] wrote:

> Following our conversation, I had a question about how server-side systems (including [redacted] or [redacted] are dealing with Android's runtime permissions.
> Suppose a user has disabled permissions to, say, GMM. With client-side location, GMM will not get location, as the user intended. However, they can still get a place card (e.g. Riddler) via ULR-[redacted]->GMM server --> GMM client. (ULR has GmsCore's location permissions, not GMM's). This seems like a bypass to Android's permissions model.
>
> - How are the various teams using [redacted] e.g. Riddler) dealing with this today?
> - Is there a solution other than apps "self-policing" ?
>
> Thanks,

[redacted]

--

[redacted]