FTC PrivacyCon 2017
January 12, 2017
Segment 2
Transcript

JUSTIN BROOKMAN: With the second session now. So I am Justin Brookman. I am the Policy Director of our Office of Technology Research and Investigation, OTEH, here at the FTC. This session is going to focus on mobile privacy issues. This obviously has been an area where the FTC has been active for quite some time, whether it's policy guidance, like our report on mobile disclosures. Whether it's tools. We have a tool for mobile health app developers, to test to see what sort of health privacy laws might be implicated. And enforcement work, as well, starting with the Golden Shores case. But also, more recently, cases like InMoby, Turn.

And we certainly recognize these mobile devices are, obviously, incredibly useful and really amazing. I mean, right? We're all kind of obsessed with our phones, maybe a little too much, but it's because they're incredibly practical, and useful, and functional devices. At the same time, we recognize there are these heightened privacy concerns, that we've been aware of for some time. But I think we're still grappling with in a lot of ways. Obviously mobile applications and libraries have access to information that's maybe a little bit harder to get in other contexts, like the web. Access to sensitive data, like geolocation, sensors like microphones, and cameras.

Obviously we've seen the platforms, kind of, evolving to try to address some of these concerns, over time. Into the development of specific advertising identifiers. Even more recently, you've seen Apple starts to move away from device identifiers for people who try to opt out of targeting. You've seen Google try to adjust, and moved more to a first time a permission model, for some sensitive permissions. Again trying to iterate, trying to get a grasp on how consumers expect these devices to operate.

But I think this is still evolving. It's still a work in progress in a lot of ways. So we are very fortunate to be joined here by four leading researchers who've been spending a lot of time looking at how mobile devices work, and thinking about some of the privacy threats, and concerns, in this ecosystem. So we're going to have four presentations. First we're in here from two presentations that present mobile tools, that kind of look and see what some of the data flow is, and data capabilities are, for applications, from Dave Choffne, from Northeastern University, and Narseo Vallina-Rodriguez, from IMDEA.

Then we're going to hear from Sebastian Zimmeck, from Carnegie Mellon, about a tool they have to "automatedily" look at privacy policies, and try to match up what those applications actually do, compared to what they assert in their privacy policy. And finally, we're going to hear from Primal Wijesekera, who's going to talk about some of the research he's been doing at University of British Columbia, thinking about ways to change what sort of permissions are ask for from applications, based on the context, and thinking about maybe new models for trying to match consumer expectations there.

So we're going to hear these four presentations. And we're going to go into a short Q&A afterward. Again, after the presentation, feel free to line up at either mic if you have questions

you'd like to ask the panelists. You can also ask on Twitter, we have question cards that will be sent up here. I may try to follow it as well. And with that, let's get going. Dave Choffnes, please.

DAVE CHOFFNES: All right thanks for the introduction. Let's see if this works. Great. So today I'm going to be talking about Recon, a tool we built for revealing, and controlling personal information leaks from mobile network traffic. And just to get started I want to make this a little bit interactive, wake you guys up. So how many in the room have used your mobile device to access the internet today? Just raise your hands. So for those on the web, pretty much everyone. OK. Now, how many of you have used it just since this session started? In just the past couple of minutes. We have a lot of honest people here. About half the room has their hands up.

So I mean, the point I'm trying to make here is, obviously mobile devices have become essential, and we're addicted to them. And we love the services that they provide. They have rich sensors, they're always connected to the internet. But one of the issues with this internet connection and all of the information that's available to them is that they can share that information with other parties. And we've learned over the years that this is in fact pervasive. And in this environment, consumers are at a huge disadvantage. It's generally known that information is shared, but users often have very little visibility into what exactly is being shared, and almost no control.

So before we got started on this project, we actually wanted to answer a simple question, which is what exactly is being shared by some of the most popular apps when you use them from a mobile device? So we did some controlled experiments, and what we did is we put very conspicuous personal information on these devices. Things that if you search for these names, or passwords, usernames, e-mails, they wouldn't have occurred by chance, just randomly in network traffic. And we did manual tests. We interacted-- well, I should say, my grad student Jing Jing Ren interacted with each of these apps, spent several weeks of her life she won't get back trying to understand how these devices are using the network to transmit personal information. So we did this for all platforms, all of the major ones, iOS, Android, and even Windows Phone.

So what did we find? The high level takeaway was that information leakage was pervasive. So on this graph, on the y-axis is the fraction of apps that are leaking personal information. On the x-axis is the different categories of personal information that we saw being leaked. And then each bar denotes a different app store. So we have iOS in blue. We have Google Play Store in red. And Windows Phone Store in orange. So some of the things that jump out immediately, there's run of the mill, basic tracking, is quite common. Device identifiers, very large fraction of apps are leaking it. Locations are also leaked. This is probably not surprising to those of you who have been paying attention.

What was a little more disconcerting is very personal information. So actual user identifiers, such as names, and potentially security sensitive information, such as credentials, were being exposed by these apps. And what I haven't mentioned yet is, everything that is in this graph was exposed in plain text, which means that anybody could have seen it. It could have been your ISP. It could have been a stranger at a coffee shop, on an unencrypted WiFi access point. So these are results from a little more than a year ago. Fortunately, a lot of these now have moved to TLS, so they're encrypted, but a lot of the information is still being shared with other parties.

So given this is the case, a question that's already come up in the first session is, what do we do to solve some of these problems that we see? So one thing you might do is try to fix the devices. That's where the problem originates. And you could say we could have better software. But it runs into a number of challenges, which is you need OS vendors to comply, or you need the app store to be able to enforce, and none of these are particularly easy to do for a variety of reasons. But one thing that's common about information that's shared with other parties is that it's all shared over the internet. And there's a common language spoken there, and it's IP. So what we can do is actually look into network traffic, and try to understand, identify, and potentially block personal information leaks in network traffic. And it's actually really easy.

So if I know what my personal information is, I can just search for it. But the challenge is when it's not mine. So what we do in this work is we actually automate the process of identifying personal information, even when we don't know it, in advance. And the key hypothesis is that information tends to be shared in a common format. So if you were to look at network traffic, you would see things like name equals Choffnes, or zip equals 02115, which is the zip code for north-eastern. And so what our system does is it actually learns the format automatically. And the nice thing is that to provide protection for consumers we don't need them to tell us anything in the first place.

On top of that, as these formats change over time, our system can adapt. So I'm not going to go into technical details, because I don't think that's appropriate for this audience, but at a very high level, we analyze network traffic, apply machine learning, and we can actually refine this using crowdsourcing. So the end result is what I'll show you. And there's obviously, it's too small for many of you to read, but what I'm showing is a dashboard that you can access via web page, from your mobile device. And this dashboard will tell you what information has been leaked about you. You can tell us if we're right or wrong, and you can also apply controls to it. So you might see something like, your name has been exposed. In some cases we've seen even passwords exposed in plain text.

You can also see other types of interfaces, like pervasive location tracking, or this interface. We call it, where they know you've been, because it's one thing to know that an app is requesting your GPS location. It's another to know just the extent of where you are, and when you're there, that these other parties know about you. So most people that we show this interface when they see their own data they find it pretty creepy, which tells you that this is something that users might want to take control over.

We've also done studies comparing app versus website, so if you're using a service on a mobile device you could use the website, or an app. We found surprising differences between the two, and they're not consistent. So we built a tool that allows you to look at several popular apps, or online services that we studied, and actually get a custom recommendation as to whether you should use the app, or website, depending on your personal privacy preferences. So there's other parts of the system. I won't go into too much more detail. You could provide feedback, telling us if we're right or wrong, and you can also decide to block, or change, or modify in some way, the information that is exposed to other parties, according to your preferences.

So high level, one question you might ask, I said machine learning. There's just some technology I'm throwing at this. Does it work? In our lab experiments we found it's very accurate, with very few false positives, and false negatives. So in case you're wondering, generally you don't see too much bad information, and the system learns over time. We've also given this out to users. And we've had almost 400 users who have enrolled in our user study. As part of this we surveyed users at an early stage. They found it useful. In fact, some modified their behavior based on the information they learned.

On top of that, we found over 27,000 cases of information being leaked by various apps. And there's been numerous cases that are suspicious, or in some cases actually just simply dangerous. So I'll focus on some of those, which is, in the process of trying to understand privacy, we saw that passwords are being exposed in plain text. Or sometimes, they were encrypted, but exposed to third parties that never should've gotten them. So far we've identified-- I should now update this to 26 apps that have exposed passwords, because we found another one last Friday, in an app that is used by-- well, they claim over 100 million installs.

We do responsible disclosure, which is why I'm not going to name them right now. So we wait until they fix the problem, before we go public. And we've got a variety of responses that are somewhat interesting from developers. And many, particularly at the major companies, will act quickly, but some don't necessarily understand that this is a huge security problem. Some actually did it by design. Some aren't even able to fix the problem, because they don't have access to the source code, and the vendor that they used doesn't exist anymore. So these are some of the problems that we face. These are persistent problems. So it's not a matter of just fixing privacy and security concerns at a certain moment in time. You have to keep monitoring. And you have to keep being vigilant. And you have to have systems that react, even when the app vendors won't be able to do it themselves.

So if any of you are interested in some of the results, we only publish the ones where we're 100% certain which app is actually responsible for leaking information. You can find information about this on our website. We also have a version that looks at websites that are leaking information. It is not our focus, but it's something that we also see as part of this study. And so to wrap up, with this project, what we're trying to do is improve transparency, and control, over personal information. So what we do is we learn what information is being leaked. We use crowdsourcing to determine if were correct. And also give us hints as to what matters to consumers. And we allow those consumers to block or change what's leaked. This is an ongoing project. You're about to hear from Narseo about Lumen. And this is something that we are talking about integrating into that environment. You could also build this into home routers. And we'd also like to apply this analysis to IOT devices, which we just learned about in the previous session, to understand what personal information is being exposed by those, as well.

So before I wrap up, I just want to thank my collaborators. In particular my Ph.D. Student Jing Jing Ren, who is behind most of the work. And if any of you are interested in learning more about Recon, you can visit this website right here, where you can also sign up to participate in our study, and use the system yourself. That's it.

JUSTIN BROOKMAN: Narseo. NARSEO VALLINA-RODRIGUEZ: I hope that you can hear it. So I am Narseo Vallina--Rodriguez, and I'm going to talk about our ongoing efforts to illuminate the [INAUDIBLE] system, within the Lumen Privacy Monitor. This works is done in collaboration with a lot of colleagues from the International Computer Science Institute. That goes from Verne Jackson, to Mark Calmet, Kristin Cliby, Sir Cheoman Edwin Driaz, and also Primal. And other colleagues at UMass, and Stony Brook University.

So whenever we run a mobile application, we know that they are accessing certain pieces of information, which they can use later to create an accurate profile about our persona. And we know that they are accessing this information because we are granting them permissions to access these pieces of data. The problem is that most users will believe that this information is only shared with the application developer. But the truth is that this information is also shared with a large number of third party services for analytics, and advertising purposes. Unfortunately, as opposed to the desktop context., we cannot rely on existing ad-blockers, because those are specifically targeting web apps. And in the case of the mobile applications, advertisement downloads are integrated in the app.

So in this project, we have three specific goals. The first one is to define the third party ecosystem that exists on mobile systems. Then we want to evaluate their impact that they have to use their privacy. And finally we want to promote more transparency, by releasing the data, and also giving access to the tools to other researchers interested in investigating that space. And finally, we also want to enable user control. We want to give users the opportunity to control where their mobile applications are actually talking to.

And how do we actually achieve it? Well, we have the Lumen Privacy Monitor, which is an application available for free on Google Play, that has been actually available for almost a year. In a nutshell Lumen leverages the Android VPN permission to intercept, and analyze mobile application's traffic, on userspace, and locally at the device. So this allows us to access personal information directly from the device, so we have access to the truth. In addition to that, with used consent, we are also doing in TLS interception. So we can look at encrypted traffic, to see whether any application is also leaking data over encrypted channels.

In this small video you can see this short tutorial about how Lumen exposes this information to the user. As you can see, if you're clicking any of the applications that has a profile, then you can look at the different pieces of that information that they are collecting, or clicking, as well as the organizations accessing this information. And in addition to that, we want to educate the users about why they should care about this specific piece of data.

But in addition to giving a tool for users, we are also using the data we collect from them to conduct research studies. And so far we have had access to more than 2,800 applications. And this has an important-- there's something that we should clarify here, because we're analyzing the applications with real user [INAUDIBLE], and we're therefore being able to analyze, for instance, what happens when users are at their homes, and how the mobile applications are acting with other IOT devices.

The challenge that, first we have to tackle is, identify the domains that are related with first party tracking services, or with third party tracking get services. And for that we represent the interactions between mobile applications, and domains as separate. And here you can see two examples, accuweather.com, which is a well-known weather service, and [INAUDIBLE], which is an analytics service. And the basic heuristic will be analyzing, or considering a domain as a third party service if at least than one application is talking to it. But, as you can see in those examples, the Accuweather app, is also accessing accuweather.com, in addition to the HTC weather wizard. So how can we distinguish between those interactions, when they are actually first party apps, or not? And for that, what we do, we analyze that tokens that exist on the application package names, as well as on their domain.

But there is one more challenge that we have to tackle, which is, how can we distinguish between networks and tracking services, from modern third party services, such as content delivery networks. And for that, the problem is that we cannot rely on existing data. Unfortunately the main blacklist, such Easy List, which is the one that Ad Block Plus, is using are targeting only web analytics, and advertisement services. And in addition to that, our existing euro-classification services, even commercial ones, are inaccurate or incomplete. Here, you can see an example. Flurry.com, which is one of the most popular services on mobile apps, is labeled as an internet servers.

So we had to create our own custom classifier. And for that we used two different approaches. The first one is defying the ligature of unique identifier, and the data flows between applications, and their domains. So that ranges from the IMEI code, or MC code, or even the mega-address, and the serial number of the device. And in addition to that, we are analyzing the content of the landing pages for each of the domains, using a word processor, and basic natural language processing.

Here, you can see that out of 3,261, seven level domains, we have identified 336 harvesting those unique identifiers. Only 9% of them were previously reported as advertising and tracking services by the other data services. But 31% of them are clearly third party services. This translates into 109 domains, which were not previously labeled as tracking services by previous methods. I will skip that slide for the sake of time. But here we can see, as well the distribution of the popularity of the different domains across the mobile apps, we can see that the most popular one is crashlytics.com This is very popular because its a service that allows application developers to somehow obtain very accurate back reports. So they know when the application is actually crashing. But its also an analytic service.

And the second one is the Facebook graph API, which is very popular because it not only allows application developers to logging in Facebook, or even to integrate the application with Facebook, social network, but also to print ads from Facebook, and also benefit from Facebook's servers. If we take a look at the top 10 domains we can see that many of them are fairly mobile specific. But we were curious about what's the prevalence of cross-platform tracking. So we called up the 1000 Alexa sites, which is a ranking that measures the popularity of different websites, and we can see that over 68% of them are actually present in the web as well.

We wanted to measure, as well, the prevalence of those apps, depending on the app category. This is very hard to see in this light, but I will highlight those boxes over there. Over 70% of the applications connect at least to one tracker. And 25% of them connect to at least five trackers simultaneously. Those categories that you see here are game apps. So they're potentially used by sensitive populations, such as children. And surprisingly, those are the ones that have a higher prevalence of tracking services. So in collaboration with Sir Chekoman, as well as with Primal, we are now constructing a test build that will allow us to check whether our mobile applications are also complying with regulations.

In the scope of the COPA study, that we are conducting, we have also found certain abusive practices. The first one is that, as opposed to certain permissions, which are protecting unique identifier such as the IMEI, we have found evidence of tens of applications, and trackers, tracking users without consent. And they manage to do it by invoking they get prompt command, which is a system level command that contains system protected information, but also unique identifiers, as you can see here. Like the Mac address, their serial number, or even the device fingerprint. And also the network to which they're connected from. And as we said the problem is that we reported that to Google, and as part of our responsible to disclosure, and they said, that the command, unfortunately, is working as expected.

In addition to that, we have also found evidence of something more while applications which are COPA compliant, not honoring, or basically using third party libraries that are not honoring users choices for anonymity. Here you can see, in red, how this specific tracker is uploading that the user has disabled Google's target advertising services. And in green you can see that they are actually uploading the user's unique ID. And this is actually going against Google's best practices, in terms of privacy.

In addition to that, we're keeping developing new features to the app, so that we can provide more services for the user. And we are currently enabling system wide user control. I'm only showing how we are blocking ads, because this is the only one that you can easily perceive, but we are incapable of blocking ads, and trackers, without actually affecting applications performance. So here we can show how basically we're blocking both in the web, as well as mobile application ads.

And in addition to that, we are also making the data we are collecting publicly available. And the first case is what we call, the Haystack panopticon, which you can see on this URL. The dots in the center represent tracking services. The dots around it represent different types of mobile applications. So if the user enters one of the application names in the box, he or she will be able to understand the different trackers that are connecting to these from this application. And then by clicking on the different trackers, they can see, as well, the network, and how interconnected everything is. We're also are hoping to make this data available for researchers, and that will be, very likely, soon available on our website. And, in the meantime I would like to invite you to go to our website, take a look at the project, and try the application, if you're interested, and give us any feedback about it. That will be very useful for us. Thank you very much.

JUSTIN BROOKMAN: Sebastian, please.

SEBASTIAN ZIMMECK: Thank you, Justin. And it's a pleasure to be here. I'm talking about the automated analysis of privacy requirements for mobile apps. And I was fortunate to work with a shipload of great collaborators on this. And all of the people who worked on this were, at some point, part of the usable privacy policy project, which was funded by NSF, DARPA, and the Air Force Research Laboratory. So when you're using a phone many different types of data are sent to first, or third parties. For example, device IDs, location information, e-mail addresses. And these types of information should be described in a privacy policy, you know, what is collected, what is shared, how long are these data retained. And the idea of this project is to look at the privacy policy on one side, and analyze what is stated there, and on the other side, look at the apps, and see whether what is said in the privacy policy actually is happening in the apps. And we call that a privacy requirement compliance.

So we need to look at both sides. The privacy policies, as well as the mobile apps. And because we want to make this automatic, as automatic as possible, we decided to analyze the privacy policies using machine learning. Essentially we are looking at the text fragments, individual words, to analyze the practices. And for the mobile apps we are looking at the source code. So we are not actually running the apps, but rather we are downloading the apps from the Play Store, decompiling them, and look at the source code. And then we compare the results of the two.

I mentioned the word privacy requirements. And privacy requirements are something we came up with. So these are self-defined, and derived from laws. The reason why we are not comparing our results directly to the law is that there are laws that are not applicable to every app. For example, as you all know there are special laws for children, for financial institutions, and this allows us to define ourselves a set of requirements that we want to analyze without necessarily getting into the difficult question of whether an app actually violates the law.

And on the right side of the slide, you see some of the privacy requirements that we analyze. First of all, we require that an app has a privacy policy. And then there are various notice requirements. For example, the notice of policy changes. That is something that we took from the California law. So users have to be notified in case of material policy changes. How they are informed of these changes. And the notices, are something that the privacy policies themselves have to comply with. On the right side you see collection and sharing practices. And those have to be talked about in the privacy policy, as well as implemented in the app. So that is something that applies to both of the policies, as well as the apps.

The first finding that was surprising to me that we have is that many apps don't have a privacy policy. Although they should probably have one. And about half of the apps that we analyzed, did not have a policy. We had a total of 17,991 apps, and out of these, 71 percent did not have a policy, despite processing PII. We used, for the policy analysis, machine learning methods. And for the analysis, static code analysis. I don't want to go into the details here. But I'd be at the post session later, so if you're interested in the details, please stop by, and I go into that.

I just want to talk a little bit briefly about the results that we received here. And the first point I want to make is that the inconsistencies between apps and privacy policies are quite numerous actually. So if, for example, you look at the first row, CID means the collection of device IDs. So

that means that a first party, an app developer, uses an API, to get a device ID. For example, an IP address, or the actual device ID, from an Android phone. You can see that 50% of apps are actually doing that without stating so in their privacy policy, or omitting to write anything about device identifiers. And that is true for all the practices that we looked at. Maybe the sharing of contact information, which you see in the last row is the exception here, but for all the other ones we certainly have higher numbers than we initially expected.

The second point to note here is, again, looking at the first row of collection of device identifiers is that we are able to find all the problems, which you can see from the recall value in the third column, which is one, but we have some false positives. So that means we identify apps that are actually covered by the privacy policy, or the app analysis goes wrong in those cases. And that is what this number of 0.75 in the precision column, of the first row means. And I think that is something we have to improve. But the good news is that by manual work this can be still helpful. And it's probably better that way, to not miss anything, and have some false positives. As opposed to missing problems.

So this is just to give you an idea of the results that we have. And these results I just mentioned were for individual apps. And what you see here is a graph that relates to a group of apps. So if you are interested in finding apps where you have a high chance of finding inconsistencies between a policy, and an app, then this graph tells you should look at apps that do not have a top developer badge, and that have very few user ratings. So these two things, on the Android store, identify apps that have, more often, problems than apps that have a badge, and that have use ratings. So that is something that can save some time if you already know, OK this group of apps is particularly interesting to me.

We have tested our system with California office of the Attorney General, and we are continuing this collaboration. And so far we're obtain promising results. I focus mainly here, and just talk on helping regulators. But our system is much more general. So the techniques are, in principle, also helpful to appstore owners, developers. For example, if they are included in software development tools, most developers are not malicious, and simply either don't know, or are not able to work this out, because they just don't have the resources. And so I think it would be important going forward to also help them.

Generally we work with Android apps. But the techniques we used, for the most part, could be also applicable to other platforms, for iOS. Probably we would need a different analysis of approach for the apps. And for the internet of things, that's another interesting use case. For example, one could start here with fitness trackers, and similar things like that, getting different types of information. That's all I want to say. Thank you very much.

JUSTIN BROOKMAN: Thnak you, Sebastian. And Primal.

PRIMAL WIJESEKERA: Hello, everyone. My name Primal Wijesekera. The work that I'm going to present here is a collective of a bunch of people in UC Berkeley, and then UC in Canada. So most of our recent work focused on just understanding how people make privacy decisions, and how we can accommodate those requirements, and improved the mobile platform. So last year in PrivacyCon, we presented a work showing that asking users every single time

when there is a privacy sensitive request is just infeasible, because of insanely high frequency. But in the same work we found out that people do want to have a more final level control over regulation. Not them just being prompted once, and then let the system make the decision. They want to make the decision more and [INAUDIBLE]

The question is that if we ask too many benign questions it's going to habituate the user for future privacy questions, which could be more concerning and harmful. So in an idealistic set up, user should only be involved or prompted only when they are likely to care about that question, about that privacy sensitive request. So that we don't habituate the users. Or else, when the system doesn't know how to react, or doesn't know how to act on behalf of the user. So by prompting the user, or by involving the user, the system make sure that they don't take a wrong decision, or make a mistake.

So to collect ground truth on user's privacy expectations, or how they want to react under different circumstances we conducted a field study last year where our participants used a heavily modified Android platform for a period of six weeks. During the study period this modified Android platform collected all the privacy sensitive requests, all the access requests originated from third party applications to resources such as location, contacts. And we also logged security, or privacy related behaviors. We thought those could be useful, as well.

While logging all these privacy sensitive requests, we probabilistically prompted each user once a day on a selected privacy sensitive request. If you can see on the screen, the prompt mentioned the application had just accessed a sensitive resource, and the type of resource that was just accessed, and the important question, if they had the option, how would they want to react? Whether they want to allow, or whether they want to deny it. So we use these responses as the ground truth for a classifier that I'm going to expend later on.

So after the study period, from 133 mobile Android smartphone users we have collected 176 events, data points. So these data points include board sensitive request, privacy sensitive request, and their behavioral traits. And we have also managed to collect more than 4,000 prompt responses. So based on our collected data set we were able to simulate ask on first use. So ask on first use is the latest permission model, or the access regulation in the mobile platform. It came out as a solution to many of the questions found in ask on install, that users were forced to make the privacy decision at the installation. Whereas, in ask on first use, users were prompted when an application requests a certain resource for the first time, so they have a better context.

But what we found out, based on our result, is that 15% of the time ask on first use made the wrong decision. So like the scenario is that users are OK with granting access, when they were prompted at the first instance under a certain set of circumstances, but in subsequent cases when those circumstances changed they actually want to react differently. Very bad decision. But ask on first use is not capturing how they change their decision, or where that decision based on the surrounding circumstances. Hence the 15% error rate. So based on our latest data set, 15% error rate means the ask on first use is likely to make a wrong decision once in every minute. So we need to do better.

So the important question is that, can we actually capture how people varied that decision based on the surrounding context? So was there surrounding circumstances? So we explored the feasibility of using machine learning to figure out how the context helped users to make decisions. So the grouped the collected data set into three different categories. The first one is just from machine information, the meta data about the permission type, the visibility of the requesting application, or the time of day. And we also, as I mentioned earlier, we also collect that sort of behavioral traits. Their mobile browsing habits. Their audio preferences in the mobile phone. Their screen login habits. For an example, how often they let the screen to timeout, versus how often they actually manually lock the screen. Whether they have password, or any sort of security mechanism in place as well.

And then there are contextual preferences. How they want to react to certain privacy questions, under different circumstances. For an example, how they want to react to a location request, when they have a clue that the requesting application is running, versus when they didn't have a clue that the requesting application is not running at all. Or how they want to react to a location request when they were using Facebook, versus when they were using a banking application.

So we had two different machine learning models. The first one is behavior, and the permission information. The second one is the contractual preferences, and the permission information. So the model based on just behavioral traits, and the permission version had an error rate of 24% versus 15% on ask on first use. The significance of this result is that the ask on first use, on average, had 12 prompts per user during the study period. But the behavioral model did not require any user involvement, because all of the data, the behavioral, and the permission information are passively observable.

So this is a very promising, and interesting first step towards decreasing the user involvement in the learning fit, so that we avoid the risk of user habituation. And the ML model that were using contextual preferences had an error rate of only 3.2%, which is an 80% reduction of ask on first use. And even more importantly, they ML model on conditional preferences only required the same number of problems, as in ask on first use. Even when we reduce the number of prompts, we still had a significantly better rate error rate, compared to ask on first use.

So what's the take away message? The contextual preferences are the most predictive feature group. So as a platform, when they are acting on behalf of the users, when they are making decisions on behalf of the users, if they want to make the correct decision, they need to take surrounding context into account. Right now we only have two different features. The foreground application, when the request is made, and whether they have an idea whether the application is running or not. This is not a comprehensive list. But we believe this is our first step toward understanding how context help, and what are the different kind of contextual cues that people are using.

So in the beginning of the presentation I mentioned it's important to figure out when to involve the user in the process. So use of machine learning, or the use of a classifier as an added advantage, when it produces the decision, it also produces a confidence score. So if the confidence is high enough it can just let the classifier, or the platform, take the decision on behalf of the user. But if the confidence is below a certain threshold, we can prompt the user, involve

the user in the process. So this prompting not only makes sure that the system won't make a mistake. It also helps or trains the classifier in subsequent cases, in future similar cases, the system can make the decision on behalf of the user correctly, without involving the user.

So we still have questions to answer. When systems are making decisions on behalf of the user, there is always this chance the system can make the wrong decision. So the question is how we can increase the transparency of this automated decision making, so that the users can go back, and check whether the decisions are being made correctly, whether they are aligned with their own preferences. If not, how they can fix it. The second one is there is the observation of using passively observable traits. This is very significant in the domains of variables, and [INAUDIBLE]. The user environment is very, very restrictive, or impossible. but we still need to learn that preferences. So we can use these passively observable traits in those domains, to learn their preferences, without actually confronting them on every single use case.

While most of the permission models, or the access regulations are moving towards being more restrictive, but as a community we don't have a clear strategy how we can deny access. Are you going to completely cut off an application from the resource? Are we going to feed fake data? Can we feed less granular data, as in for location? So the question is how can we deny access, without actually compromising the usability? Because if the usability is compromised, the user will be more likely to be forced on a lenient approach in their privacy choices. So while we are working on most of these questions, hopefully we'll have better answers to present in the next PrivcayCon. Thank you.

JUSTIN BROOKMAN: Thank you very much, Primal. Thank you to all of the speakers. We're going to a relatively short period of Q&A now. But if folks in the audience would like to ask things I invite you guy to come up. Some early folks on Twitter, feel free to tweet something, and we'll try to-- someone will get it up here for me. Actually, someone came to the podium right now. So go ahead. Sorry. And if you could, identify yourself before you talk, if you're willing to.

SPEAKER 1: Yes. My name is Fur O'Neill. I'm a Ph.D. Student at Texas Tech, and also a tech writer at Eset, a security software company. I have a question for three of you, and first is for Sebastian. I just had a question about your research method. Specifically, how did you correlate your inconsistencies with the user ratings? If you have a specific like figure, or table, in your paper that you can refer to. I wasn't sure what you were referring to there.

SEBASTIAN ZIMMECK: The figured that I showed Was a logistic regression model, and applied to these dependent, and independent variables that you saw. Is that what you're asking?

SPEAKER 1: Did you specifically talk about that in the paper? Can you reference which figure in the paper it was, you were mentioning? From your slides, I couldn't identify that paper.

SEBASTIAN ZIMMECK: OK. Yeah. I mean it's definitely in the paper, it could be figure eight, but it's pretty much at the end.

SEBASTIAN ZIMMECK: All right. I'll get with you afterwards.

SPEAKER 1: I'll send you an e-mail about that.

SPEAKER 1: And then the next question I have is for Narseo. Is that how you-- Yeah. OK. I'm just wondering, with your app, which is really cool. I really like the idea.

NARSEO VALLINA-RODRIGUEZ Thanks.

SPEAKER 1: But I'm wondering like, from the like ethical hacking type of perspective, now we're introducing another app that is basically looking at personal identifiable information of somebody, which can cause problems in itself, right? I'm wondering if there's a way to get that technology, what it's doing, which is good, has useful purposes, and maybe put that at the app level, as, you know, from a FTC perspective, like how are we going to use this technology across all apps, right? So we can't-- we don't want everyone to be downloading the app. What if you could integrate that into existing apps, or maybe have it somehow on the app store level? So before an app passes through to the user's mobile phone, or something, you could have a way of, I mean, to have to connect to the cloud. I'm just thinking out loud here of ways to not have to have another app that's accessing all this information in another way. I just wonder if you've thought about that.

NARSEO VALLINA-RODRIGUEZ We did. And that's actually a really nice question. By the time that we'd released the application, we had to be very careful about the ethical considerations, because we were accessing all user's traffic. So the approach that we decided to follow was to do all the processing locally in that device. And we are only uploading [INAUDIBLE] traces, which are very simplified. It's application X talks to xyz.com, and it linked the IMEI over TLS. But we don't care about the value of the IMEI. We don't care about the IP address of the user. So we are just interested in measuring how applications actually behave. And we are also happy to give a different versions of the application, which is printing all the traffic that it's analyzing. But we are very cautious about giving that tool to others, because if they deploy it, then the ethical considerations also show up.

SPEAKER 1: That's why I was-- there was a part of it, where it's useful for everyone to see it, but also, it's going to add a level of--

NARSEO VALLINA-RODRIGUEZ So in that case, basically if you are interested in using it, we can ship it to you. But we would like to know that you're not releasing it with real users. If you're going to use it for your own despot with a fake account whatsoever, that should be fine. But we have to be careful about collecting any personal information of the users. We are not uploading that payload at all. We don't care about it.

SPEAKER 1: OK. Thank you. That answers my question. And then my last question is for Primal. Is that good? Your talk, it reminded me of the last panel with from The Democracy Of Technology, where you said that you use the two terms, likely, and probably for what notices that we should ask about meaning that-- because you said we can't ask users for every notice, because there's just too many, right? So then you say what's likely or probable, and then you go on to describe your machine learning algorithm that's going to go in and make some decisions. You use the term contextual and what made me think of the last panel was contextual, you know

I think a synonym could be, inferred as well. Contextual or inferential. So I'm just wondering if there's a possibility that, you know, your machine learning could have elements of algorithm bias, where what you're going to be asking is going to be based on these things that we might not be wanting to ask people about. Is again, is that going to be different for different groups of people, different stereotypes, that maybe are identified from different groups of people. So different people would be asked different things, if that's something that could be possible.

PRIMAL WIJESEKERA: Yes. So to answer the question, are we going to use certain things that they don't want to be accounted for. When I say context, so if I speak on the contextual model, we'll be making the decision just based on the run-time information. Those information won't be dependent on the specific user. So, for an example, the visibility. When I said the visibility of the application, that care is whether the user had a clue that the application is running or not. So we don't take the features, like the gender of the user, or the inferred age of the user, the inferred profession of the user. We don't take those categories at all. We just based our decision on the run-time information. So if I directly answer the question, I would say we don't use any information that would probably be harmful, or violate users expectations, because these are just run-time information.

SPEAKER 1: My follow up question doesn't apply then So I'm done.

JUSTIN BROOKMAN: I have a follow up question on that, actually. I was kind of curious what the practical implications of your paper were. Do you think it is the right approach for platforms to maybe move to this more, kind of black box system of trying to, even in the best of faith, kind of guess when someone would want-- given some of the potential bias, or accountability issues, that you mentioned in your paper, or there are more, maybe more modest recommendations based on your paper, that they would like to see the platforms implement.

PRIMAL WIJESEKERA: So let me comment on that. The notion of when people say-- I think, whenever we mention machine learning, the first things come out, it's a the black box. No one understands what goes in. No one understands what comes out. They know if you put it in, you'll get something. So that's why--

JUSTIN BROOKMAN: Pejoratively

PRIMAL WIJESEKERA: Yeah. So that's why I mentioned the first open question is that we need to increase the transparency of this other automated decision making. We're actually, right now, working on it, like how we can let the user know, that this particular decision was made because of these reasons. Your application X requested location. It was denied because it was not running in the foreground. So you earlier had denied a location request, when an application was running in the background. So that's why we denied it. So to tackle that question, I think the solution is how we can increase the transparency so that users are kept informed.

And the other question is that, users' privacy preferences can change over time, right? So how we can accommodate those changes? So when we have an interface. Or something that users can go back and audit the decisions that are made, they can actually fix certain things that were not actually aligned with their preferences.

JUSTIN BROOKMAN: Feel free to jump in if anyone else wants to join, but I have a follow up to the gentleman's second question about observing personal information. And this was the thing we talked about on the planning call, which I thought was interesting, which is about the challenges of encryption, because we like encryption. We recommend encryption as something to safeguard traffic from outside attackers. But in some ways, researchers are sometimes the attackers, right? In, kind of, both of your presentations. And something that we encountered at OTEC, when we looked at smart TVs, we could see the smart TV was phoning home, something, but it's actually sometimes really challenging to do man in the middle, especially on an operating system people don't know very well. So maybe talk a little bit about what some of the challenges are, you guys have seen, as far as encryption. And then whether it does interfere a lot with the research you've been doing. And then, kind of maybe, what the right balance is to kind of make sure that these black boxes, right, are accountable. We can kind of find out what they are saying about us. But still we also like the safeguarding from other people's attacks.

DAVID CHOFFNES: Yeah. I can speak to that. When we started this study in 2015. A lot of things were in plain text. I think over the past year, increasingly, we see information flows transitioning into encryption. And that's good, in terms of the man in the middle eavesdropper. But it is true that, as researchers, we have to go to more and more extreme measures, to be able to understand what is happening inside that traffic. Sorry, inside those encrypted connections. And so both myself, and a number of my colleagues. I'll let Narseo speak for himself. But increasingly we're thinking about ways that we could address this problem through maybe changing how we treat different parts of the data flows. So for example, if a device is leaking information about me, do I have a right to see if that information is leaking about me? And if that's the case, there are technical solutions that would allow you to encrypt it in a way, that the owner of that data, and only the owner of that data, would be able to see that.

But there's definitely going to be some challenges in moving to this environment, because often, when information is sent over the network, some of it may be about a user. Some of it may belong to the app, or to the company, it may be considered sensitive, and they don't want to expose that to researchers or others. So in terms of actually making it happen, I think there would probably need to be a push in terms of policy. But from a technology perspective, we certainly have the basics, and the elements in place to achieve something like this.

NARSEO VALLINA-RODRIGUEZ I second all of what Dave said. But in our experience, we are seeing around 70% of the apps using TLS. And only a handful of them cannot be intercepted. So just with a basic, man in the middle attack, we can decrypt them easily. And I think that the advantage in our site is that many of those applications want to run on corporate environments, where their ideas are deployed. So they have to allow a third party to inject a certificate, and somehow perform man in the middle attacks. So we should take advantage of that for a while.

And, in any case, there are other cases in which you can take more extreme measures. Like, if your Samsung TV is talking to a domain that you're not trusting completely, then you can completely block, and act as a flow firewall, to some extent.

JUSTIN BROOKMAN: The gentleman, at the fifth mic.

SPEAKER 2: Yeah. I have a question for Primal. When you did your field study, did you find that users generally answered for each of the prompt for the permission the same way? And is it possible to maybe crowdsource some of the decision making?

PRIMAL WIJESEKERA: I think that, I feel that [INAUDIBLE] they've already been looked into how to crowdsource these privacy decisions. What we found out with based on our field study is that if they are using-- right now, we don't know what are the full spectrum of contextual circumstances they use to make those decisions. So the question remains, like if we crowdsource, can we account all these circumstances. So when they deny a certain request, can we know they denied because of this? So when we do like experience something, or like the field study, we have the full spectrum of data, that they have, and they make the decision. So it's easier for us to figure it out, or at least step closer, for us to understand what are the things that they use to make the decision. So that's why I say, well crowdsourcing is useful, but I'm not sure how we can capture all these contextual circumstances, if we do the crowdsourcing.

SPEAKER 2: For some applications though, like ones that ask for location information for doing like a Bluetooth low energy beacon to figure out stuff about you, there's no reason to do that. Some of that you could crowdsource, right? Yes. So in the previous work, one of the reasons that we found out that when they denied certain request, the functionality it wasn't cited as one of the main reasons. But the thing is that, right now, we don't have that project. On the platform, we don't know why they were requested So visibility is, one way or another that we're using to figure out whether this function will be created or not. But even, that would be like the case where Google Maps could be accessing location, even when they're running invisible, right? So it is a very gray area. There is no black and white. So, as you said, there are some very direct cases. There are very gray cases, that can't really make a decision, based whether they're base related, or not.

SPEAKER 2: Thank you.

JUSTIN BROOKMAN: I have one more follow up question for Sebastian Then I'll get to the gentlemen. Your tool looks at privacy policies, and tries to kind of match up whether what they're doing is consistent with what the privacy policy says. And there are kind of different schools of thought from what privacy policy should do, right? The conditions should be really long and detailed, and say all the stuff that goes on, maybe like privacy requirements, that you kind of talked about in your slides. Or should they be more accessible, and very easily readable, understandable by a lay audience. And it occurred to me, sitting here, that last year, during PrivacyCon, one of the presenters kind of made a strong pitch that the audience for a privacy policy really shouldn't be the general public. It should be like us here at the table. It should be like the academics, or the press, or researchers, who can then kind of demonstrate some external accountability for actual practices. I was curious, I mean, if you kind of thought-- I mean, obviously, you guys showed that even half of them don't even have a privacy policy. So maybe this debate's a little early. But I was wondering if you had really thought about what the role the privacy policy should be, and kind of developing your tool for testing it.

SEBASTIAN ZIMMECK: Yeah. That's a very good question. And we thought a lot about it. And I don't know, to be honest with you. I tend to think of private privacy policies as mainly

something for lawyers. And, you know, the very interested user might read it, or might look something up, but my feeling is that that's probably not the major thing that users do. And that's one reason why we came up with these privacy requirements. And I think that probably it will become more clear in the future, that, you know, privacy policies are really addressed at a certain audience. And I think one thing that I could see coming out of this work, of comparing apps practices, and what is written in policies is that maybe you could alert users if there is a discrepancy between the two. And then that's a different way of notice and choice, right? So then the user could say, OK. I don't know the privacy policy. But as long as I'm not alerted, I know, the app behaves as the policy says. And if I'm interested, I look it up. And if I receive a message, then I know there is something wrong. Maybe that's a new idea.

JUSTIN BROOKMAN: I think we're over time, but I did promise you the chance to have the last question. So yours will be the last question, and then you can come confront us afterwards.

SPEAKER 3: Are we seeing the promise of machine learning local to the smartphone or the device, in order to manage permissions? Is that something that is feasible now? Or will it require regulation, or legal changes?

PRIMAL WIJESEKERA: I'll probably have a very definite answer in four to five weeks. Because we are right now trying implement all of these ML models in Android itself. The question is that if we use a server, or something, then that is a data concern, that we are shipping user's data into a third party service. So that's why we want to have everything in the smartphone itself. Well, there result so far is promising. We might have to compromise certain things if you want to have everything in the phone itself. Well, there could be performance degradation. We'll have to cache certain results. But we should have everything in the smartphone itself to avoid the privacy issues that shipping out of the phone. Thank you.

JUSTIN BROOKMAN: So we are over time. But thank you all very much. Hopefully the presenters will stick around for a while to answer questions. We're now going to a 10 minute break. After the break at 11:55 Lorrie Kraner will be moderating a panel on consumer privacy expectations. Again, there will be a lunch poster session, where there will be, I think about 10, to 15 other research presented to folks. If folks want to grab lunch, and walk around, and see what's going on. And the last caveat, you are not allowed at all to bring food or drink in this room. With that, enjoy your time. Thank you. Bye. Bye.