Can Systems Explain Permissions Better? Understanding Users' Misperceptions under *Smartphone Runtime Permission Model*

Bingyu Shen¹, Lili Wei², Chengcheng Xiang¹, Yudong Wu¹, Mingyao Shen¹, Yuanyuan Zhou¹, and Xinxin Jin³



Permissions control sensitive data

• We can perform various activities with apps on the smart phones.



• For certain functionalities, apps may need certain **permissions** to access **sensitive data or operations**.

Runtime permission model

- Users need to comprehend the permission requests to make decisions.
- System mainly provide brief information in the dialogs.

Allow Snapchat to make and manage phone calls?
Allow
Deny

Research Questions

- 1. (Permission model change) How commonly do users still have apps with install-time permission model?
- 2. (Runtime comprehension) Can the current **system-provided information** help users comprehend the permissions and their capabilities?
- 3. (Extra system information) What extra information (if the systems can provide) would impact users' decisions?

Research Questions

- 1. (Permission model change) How commonly do users still have apps with install-time permission model?
- 2. (Runtime comprehension) Can the current **system-provided information** help users comprehend the permissions and their capabilities?
- 3. (Extra system information) What extra information (if the system can provide) would impact users' decisions?

Android permission model change



Android permission model change



How commonly do users use low-version apps? Can users differentiate the low-version apps?

$\widehat{}$ Wi-Fi connection information \sim	Deny	Google Play G Pay	ACCEPT
Google play		photography controls	to Android.
optimized for your Android phone. You can access, view and edit your Microsoft Word. Microsoft Excel and Microsoft		READ MORE	
PowerPoint documents from virtually anywhere. Documents look like the originals, thanks to support for charts, animations.	Microsoft	Rate this app Tell others what you think	

Methodology 1: permission data analysis

- Analysis of Android permission model change
 - Dataset:
 - The lists of installed apps from 4,636 Permission Checker [1] users
 - App metadata and user's real permission settings (allow or deny) for the apps
 - Threat:
 - The demographics of Permission Checker users are unknown

Low-version apps are pervasive

- Among the 4,636 real Android users in our study, a large percentage (61.8%) have at least one such app installed.
 - Many apps are popular and still updating!

Top five commonly used low-version apps by PerChecker users						
App Name	# install	Update date*				
TextNow	10M +	09/19/18				
ES File Explorer	100M +	09/17/18				
Settings DB Editor	100K +	09/01/18				
WiFiMan	100K +	08/30/18				
Advanced Tools	100K +	07/27/18				

* (Date retrieved from Google Play on 09/20/2018)

Users cannot differentiate lowversion apps

- More than **one-third (38.3%)** of users believe low-version apps will ask for permissions at runtime.
- Even though users can revoke lowversion apps' permissions, only **one** user revoked its permission.



* Permission dialog for low-version apps on Android P.

Research Questions

- 1. (Permission model change) How commonly do users still have apps with install-time permission model?
- 2. (Runtime comprehension) Can the current **system-provided information** help users comprehend the permissions and their capabilities?
- 3. (Extra system information) What extra information (if the system can provide) would impact users' decisions?

Permission groups

- A permission group has a set of related functionalities.
- Users can only make decisions by groups.



Can users comprehend all the permissions and capabilities inside the permission group?

Methodology 2 – Survey

- Survey 1: Permission group comprehension (n=359 iOS & Android)
 - Choose **all** operations included in the permission group

	L.
Allow and manage	to make e phone calls?
A	llow
D	Deny

- Get your phone number
- Get your phone unique ID (e.g. IMEI)
- □ Make phone call
- □ Answer phone call
- □ Read your location
- □ Know whether the phone is making phone calls
- □ Read call history
- □ None of these
- I don't know

Hard to infer the accurate scope from the system-provided information

• Only 1 in 20 (6.1%) can infer the accurate scope of permission groups from info. in the system dialogs.



Reasons for misunderstand the permission group

• Reason 1: Hard to infer from the system description



Reasons for misunderstand the permission group

- Reason 1: Hard to infer from the system description
- Reason 2: Function-related permissions



	Anaroia	102
Take pictures and record video	88.6%	90.3%
Read pictures and video	41.4%	33.3%

:nc

Developer-provided explanations

- iOS requires developers to provide permission rationales for their permission request.
 - App store will audit the rationales.
 - But rationales mostly focus on benefits or make wrong claims [1,2].





ICC 2018

Misunderstanding for permission rationale

 More than half (54.7%) of users do not know that the explanations are provided by the developers.





■ By developers ■ By iOS ■ I don't know

 Uniform appearance across apps
Don't trust apps would help them understand the permissions.

Research Questions

- 1. (Permission model change) How commonly do users still have apps with install-time permission model?
- 2. (Runtime comprehension) Can the current **system-provided information** help users comprehend the permissions and their capabilities?
- 3. (Extra system information) What extra information (if the system can provide) would impact users' decisions?



Decision factors in granting permissions

- We find several factors related to users' decisions.
- Internal factors (how the data will be used)
 - Background access
 - Data transmission
- External factors (people's opinion on the app)
 - Brand reputation
 - Rating
 - Reviews

More results in the paper (Section 6)

Recommendations

- Security vs backward compatibility
 - System should provide clear notices for security enhancements
 - Joint-effort from app stores.
- Address common permission misunderstandings
 - Design permission groups that fit into user expectations
 - Developer-provided rationale should be regulated/highlighted!
- Extra information to address user concerns
 - Internal factors may be obtained with OS level support
 - External factors require crowdsourcing efforts

Conclusion

- Users have common misunderstandings in the permission model.
 - Low-version apps are still prevalent three years after the introduction of runtime permission model.
 - Many users mistakenly believe that the low-version apps still need to request permissions at runtime.
- Runtime permission group comprehension
 - User commonly misperceived the scope of permission groups from the the limited information provided by systems.
 - Many users mistakenly believe that iOS provides the rationale messages.
- Systems can provide more information that impact users' decisions.
 - Comparison of factors: background access, data transmission, brand reputation, rating, review and grant rate
 - Negative messages can have a stronger impact on users' permission decisions.

Thank you!



byshen@ucsd.edu

https://github.com/ucsdopera/PermissionStudyUsenix21



Backup slides

Permission model change



Install-time model

2021-04-12



Runtime model (Android 2015-)

Participants

- UCSD participants, flyers, Amazon Mturk workers
- Interview (n=20)
- Survey 1 (n=359)
- Survey 2 (n=1200)

