



# Cross Miniapp Request Forgery: Root Causes, Attacks, and Vulnerability Detection

**Yuqing Yang**

Yang.5656@osu.edu

Yue Zhang

Zhang.12407@osu.edu

Zhiqiang Lin

Lin.3021@osu.edu

11/08/2022



# What is Super app?

- ① Super apps: apps integrating multiple services

# What is Super app?

- ① Super apps: apps integrating multiple services
  - | WeChat (China): 1.26 billion users

# What is Super app?

- ① Super apps: apps integrating multiple services
  - | WeChat (China): 1.26 billion users
    - | *'It's sort of like Twitter, plus PayPal, plus a whole bunch of other things. And all rolled into one great interface.'* | Elon Musk
  - | AliPay (China): 330 million users

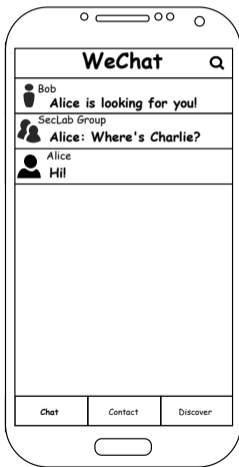
# What is Super app?

- ① Super apps: apps integrating multiple services
  - | WeChat (China): 1.26 billion users
    - | *'It's sort of like Twitter, plus PayPal, plus a whole bunch of other things. And all rolled into one great interface.'* | Elon Musk
  - | AliPay (China): 330 million users
  - | Line (Japan): 178 million users

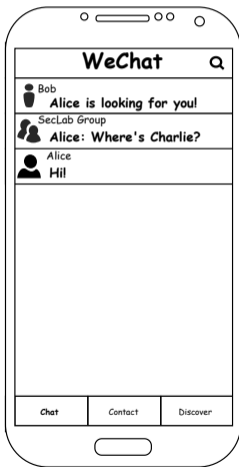
# What is Super app?

- 1 Super apps: apps integrating multiple services
  - | WeChat (China): 1.26 billion users
    - | *'It's sort of like Twitter, plus PayPal, plus a whole bunch of other things. And all rolled into one great interface.'* | Elon Musk
  - | AliPay (China): 330 million users
  - | Line (Japan): 178 million users
- 2 Miniapp, a light-weight and full-fledged app, executed inside a JavaScript engine created (or virtualized) by the super app.

# What is Mini-app?

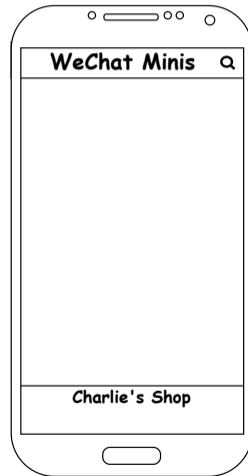
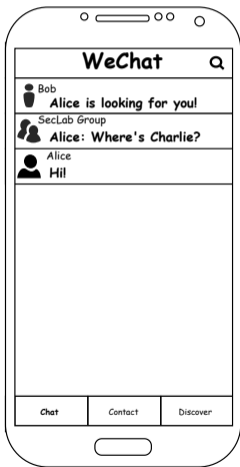


# What is Mini-app?

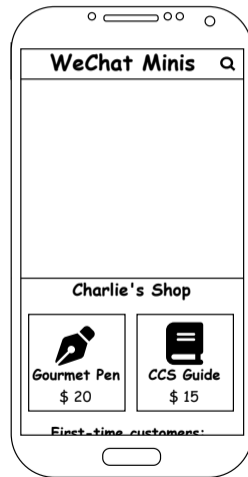
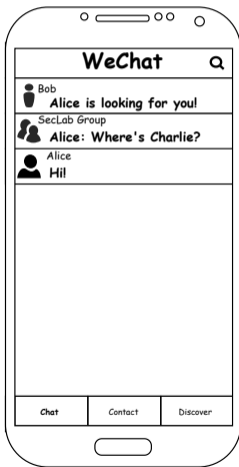




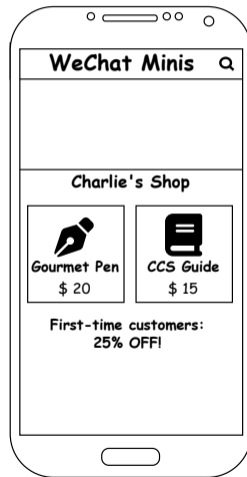
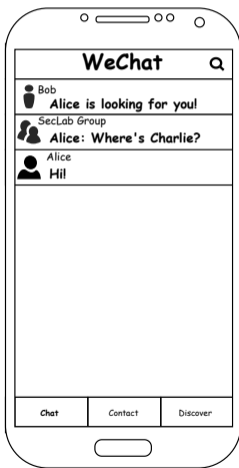
# What is Mini-app?



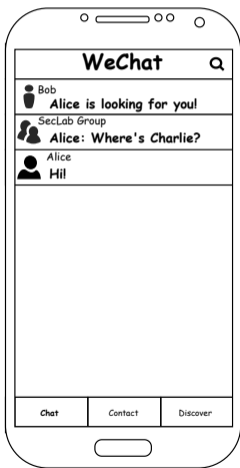
# What is Mini-app?



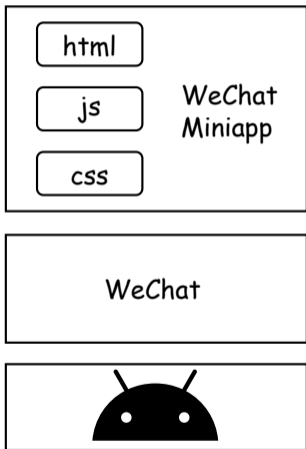
# What is Mini-app?



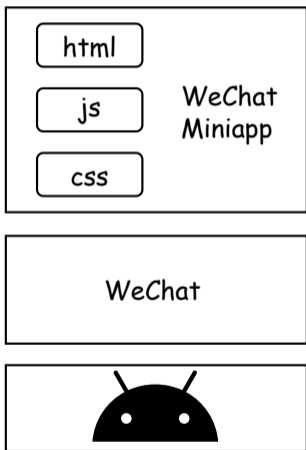
# What is Mini-app?



# Summary: Miniapp

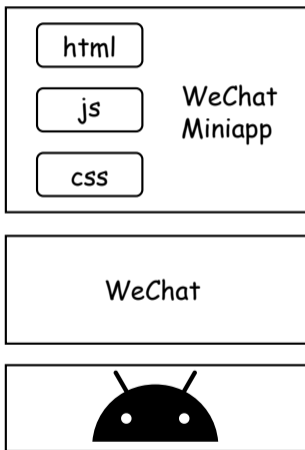


# Summary: Miniapp



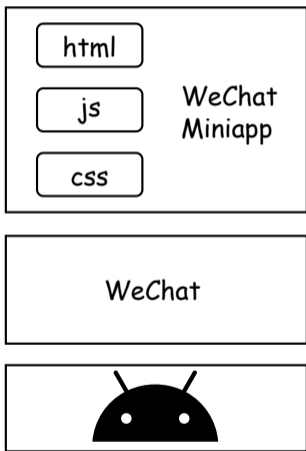
- | Miniapp: third-party apps *inside* super app
- | Super app:
  - | Provide Execution Environment

# Summary: Miniapp



- | Miniapp: third-party apps *inside* super app
- | Super app:
  - | Provide Execution Environment
  - | Manage OS Resources

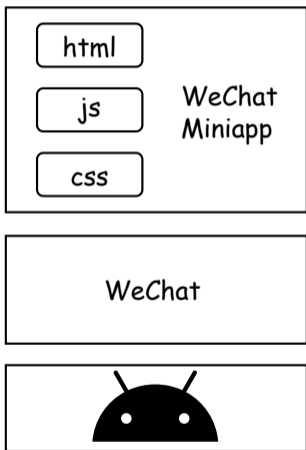
# Summary: Miniapp



- | Miniapp: third-party apps *inside* super app
- | Super app:
  - | Provide Execution Environment
  - | Manage OS Resources
  - | Handle Cross-miniapp Messages



# Summary: Miniapp



- | Miniapp: third-party apps *inside* super app
- | Super app:
  - | Provide Execution Environment
  - | Manage OS Resources
  - | Handle Cross-miniapp Messages
  - | ...

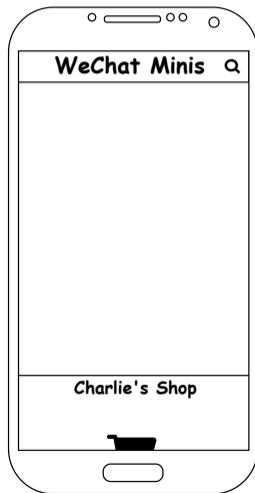
# Collaboration between miniapps



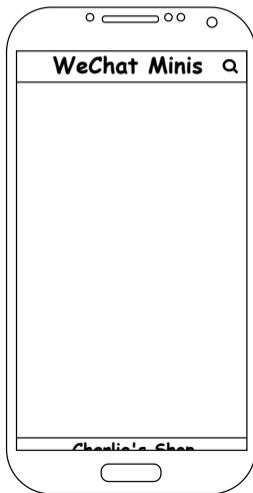
# Collaboration between miniapps



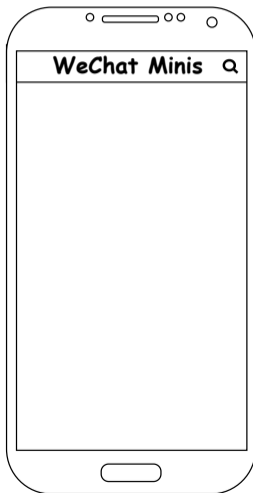
# Collaboration between miniapps



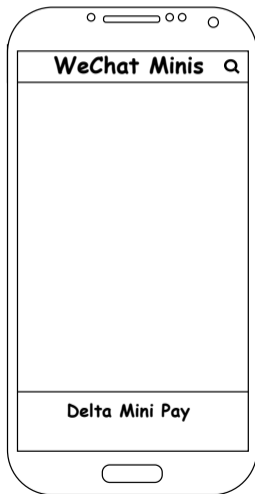
# Collaboration between miniapps



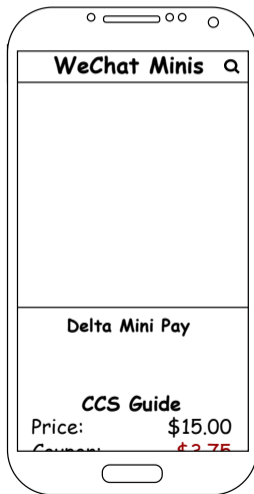
# Collaboration between miniapps



# Collaboration between miniapps



# Collaboration between miniapps

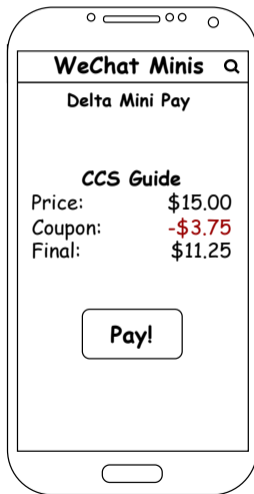




# Collaboration between miniapps



# Collaboration between miniapps



# Real-world Case

```
1 // sender (shopping miniapp) ID: wxd7c977843ebe7a64
2 submitOrder: function(){
3   price = self.getPrice();
4   tt.navigateToMiniProgram({
5     appId: "wx2d495bf4b2abdecef",
6     path: "paymentpage",
7     extraData: {
8       Price: price
9       orderID: orderid,
10    }
11  });
12 }
13 onLaunch(o){
14   var e=this;
15   o.referrerInfo && (e.globalData.paymentState
16     = o.referrerInfo.extraData.paymentState) &&
17     (e.globalData.couponCode
18     = o.referrerInfo.extraData.couponCode)
19   if(e.globalData.paymentState == "Success")
20   {
21     shiptheProducts() //ship the products
22   }
23   saveCouponCode(e.globalData.couponCode)
24 }
```

```
1 // receiver (payment miniapp) ID: wx2d495bf4b2abdecef
2 var e=getApp();
3 onLaunch(o){
4   o.referrerInfo && (e.globalData.price
5     = o.referrerInfo.extraData.Price) &&
6     (e.globalData.appId
7     = o.referrerInfo.appId)
8   e.globalData.orderID = o.referrerInfo.extraData.orderID
9 }
10 Pay:function() {
11   var price = e.globalData.Price
12   wx.requestPayment({price, ...}) //pay the order
13   if(e.globalData.appId == "wxd7c977843ebe7a64"){
14     e.globalData.coupon = 'MYCOUPON'
15   }else{
16     e.globalData.coupon = null
17   }
18 }
19 wx.navigateBackMiniProgram({
20   extraData: {
21     paymentState: 'Success',
22     couponCode: e.globalData.coupon
23 }
24 }
```

# Sending Cross-app Message

Launch me with parameters!

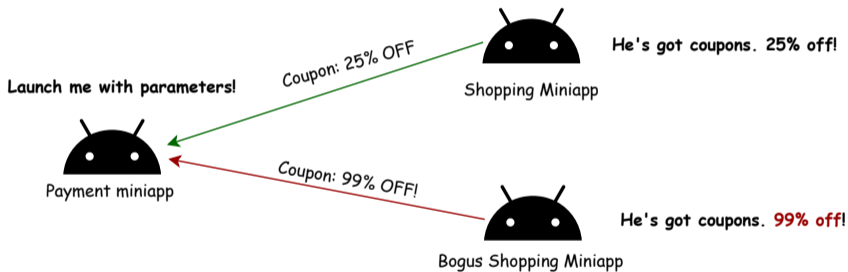


Coupon: 25% OFF

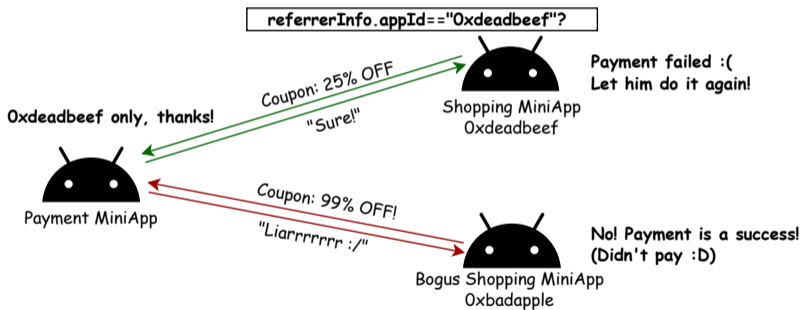


He's got coupons. 25% off!

# Sending Cross-app Message



# Sending Cross-app Message



# CMRF Vulnerability



- | CMRF: Cross-Miniapp Request Forgery
- | Consuming data without checking sender's identity can be dangerous!

# Attack Threat Model

## Victim:

- | On-market miniapp's back-end
- | Platform users' privacy

## Assumptions:

- | Front-end code is safe
- | Back-end is trusted



# Vulnerability Detection

## Dynamic analysis?

- | Over 2.5M miniapps to scan
- | Not scalable

## Static analysis?

- | Message Usage?  
referrerInfo.extraData.\*
- | ID Check?  
referrerInfo.appId

# Implementation

## Challenges

- | Packages are obfuscated
- | Handle Variable Aliases
- | Cross-Function Invocations

## Data-flow Analysis

- | DoubleX [1]

# Data Collection

## WeChat

- | 2,571,490 Miniapps from WeChat with MiniCrawler [2]
- | 6.29 TB space

## Baidu

- | 148,512 miniapps
- | 81 GB space

# Miniapp Stats (WeChat)

# Affected Miniapps (WeChat)

wechat						
Category	No Use		Checked		Vulnerable	
	# app	%total	# app	%	# app	%
Business	131,078	5.1	81	8.07	923	91.93
E-learning	10,271	0.4	4	5.19	73	94.81
Education	240,077	9.34	184	3.72	4,756	96.28
Entertainment	29,442	1.14	140	33.02	284	66.98
Finance	3,509	0.14	6	6.67	84	93.33
Food	114,675	4.46	332	8.07	3,780	91.93
Games	88,056	3.42	10	2.09	469	97.91
Government	31,432	1.22	33	9.02	333	90.98
Health	27,716	1.08	37	5.44	643	94.56
Job	21,773	0.85	16	7.02	212	92.98
Lifestyle	394,493	15.34	269	4.23	6,092	95.77
Photo	9,039	0.35	3	4.41	65	95.59
Shopping	989,498	38.48	743	2.56	28,304	97.44
Social	20,671	0.8	6	2.99	195	97.01
Sports	15,980	0.62	69	22.48	238	77.52
Tool	261,467	10.17	122	3.72	3,161	96.28
Traffic	35,412	1.38	53	9.28	518	90.72
Travelling	10,524	0.41	5	3.62	133	96.38
Uncategorized	83,983	3.27	0	0.0	18	100.0
Total	2,519,096	97.96	2,113	4.03	50,281	95.97

# Case Study

Category	Variable Name	Vulnerable w/o Check	Vulnerable w/ Incomplete Check	Total
Payment Info	for_pay_back	355	0	355
	payStatus	313	2	315
	pay	178	9	187
	isPay	118	0	118
	isLecturePay	115	0	115
Order Info	orderId	132	11	143
	orderInfo	80	0	80
	order_id	42	0	42
	jtOrderId	36	3	39
	hpj_jsapi_order_id	21	0	21
Phone Number	mobile	6,627	7	6,634
	phone	53	0	53
	userPhone	8	0	8
	phoneNumber	6	1	7
	partnerMobile	2	0	2
Promotion Info	cardId	25	0	25
	user_coupon_id	2	0	2
	couponCode	1	0	1
	coupon_id	1	0	1
	coupon_no	1	0	1
Device Info	deviceId	2	0	2
	uuid	2	0	2
	deviceId	1	0	1
	devicenum	1	0	1
	UUID	1	0	1

# Case Study: Shopping-for-free

Figure: Xixiu Group Purchase Backend (WeChat)

# Case Study: Device Manipulation

Figure: Suyuan Webcam (WeChat)



# Case Study: Promotion Abuse

Figure: Aurora Vision (Baidu)

# Takeaway

Root Cause: failure to check miniapp identity before consuming data is dangerous.

The Attack : CMRF attack can both cause financial loss to vendors and privacy leakage to users.

Detection : we detected that over 95.97% WeChat and 99.8% Baidu miniapps consuming cross-miniapp data are vulnerable.

Responsible disclosure we informed Tencent (Oct 2021) and Baidu (Apr 2022) and received acknowledgements from both.

<https://github.com/OSUSecLab/CMRFScanner>

# Thank You

## Cross Miniapp Request Forgery: Root Causes, Attacks, and Vulnerability Detection

Yuqing Yang  
Yang.5656@osu.edu

Yue Zhang  
Zhang.12407@osu.edu

Zhiqiang Lin  
Lin.3021@osu.edu

11/08/2022

# References I



Aurore Fass, Doliere Francis Some, Michael Backes, and Ben Stock.

Doublex: Statically detecting vulnerable data flows in browser extensions at scale.

In *Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security*, pages 1789{1804, 2021.



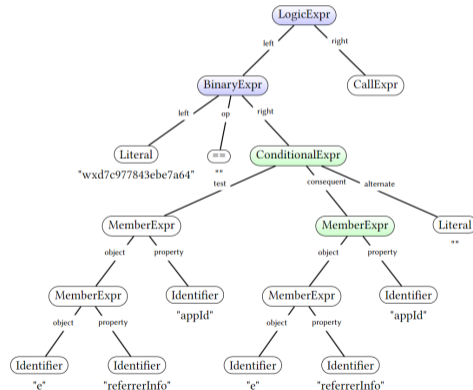
Yue Zhang, Bayan Turkistani, Allen Yuqing Yang, Chaoshun Zuo, and Zhiqiang Lin.

A measurement study of wechat mini-apps.

In *Proceedings of the 2021 ACM SIGMETRICS/International Conference on Measurement and Modeling of Computer Systems*, 2021.

# Identifying ID Checks

```
1 //code fragment of a vulnerable miniapp
2 "wxd7c977843ebe7a64"==(
3   e.referrerInfo.appId?e.referrerInfo.appId:"")
4   && checkPayStatus(param).then(function(a){...})
5
6 }
```



# Applicability

Super App	Vendors	AppID	Sending Request APIs	Location	Audio	Bluetooth	Camera	Multi-Media	Sport	UserInfo	Address	Invoice	File	Data Cache	Payment	AccountInfo	Coupon	PhoneNumber	Network
QQ	Tencent	appld	navigateToMiniProgram, navigateBackMiniProgram	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
WeChat	Tencent	appld	navigateToMiniProgram, navigateBackMiniProgram	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
WeCom	Tencent	appld	navigateToMiniProgram, navigateBackMiniProgram	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Baidu	Baidu	AppKey	navigateToSmartProgram, navigateBackSmartProgram	✓	✓		✓	✓		✓	✓	✓	✓	✓	✓			✓	✓
Taobao	Alibaba	appld	navigateToMiniProgram, navigateBackMiniProgram	✓	✓	✓	✓	✓		✓	✓	✓	✓	✓	✓			✓	✓
Alipay	Alibaba	appld	navigateToMiniProgram, navigateBackMiniProgram	✓	✓	✓	✓	✓		✓	✓	✓	✓	✓	✓			✓	✓
Tiktok	Bytedance	appld	navigateToMiniProgram, navigateBackMiniProgram	✓	✓		✓	✓		✓	✓	✓	✓	✓				✓	✓
JINRI Toutiao	Bytedance	appld	navigateToMiniProgram, navigateBackMiniProgram	✓	✓		✓	✓		✓	✓	✓	✓	✓				✓	✓
Watermelon Video	Bytedance	appld	navigateToMiniProgram, navigateBackMiniProgram	✓	✓		✓	✓		✓	✓	✓	✓	✓				✓	✓
Pipixia	Bytedance	appld	navigateToMiniProgram, navigateBackMiniProgram	✓	✓		✓	✓		✓	✓	✓	✓	✓				✓	✓
Toutiao Lite	Bytedance	appld	navigateToMiniProgram, navigateBackMiniProgram	✓	✓		✓	✓		✓	✓	✓	✓	✓				✓	✓