Understanding the Miniapp Malware: Identification, Dissection, and Characterization

Yuqing Yang [1] Yue Zhang [2] Zhiqiang Lin [1]

[1] The Ohio State University

[2] Drexel University

Feb 25th, 2025

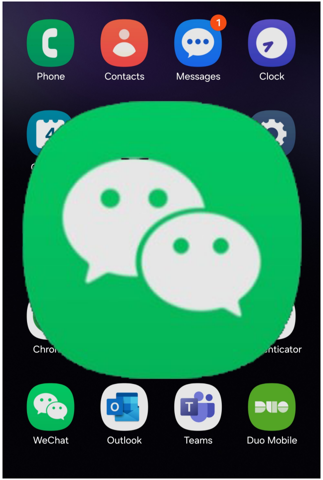
In Short...

- The first miniapp malware dataset
- Taxonomy of miniapp payloads
- Characterization of miniapp malware

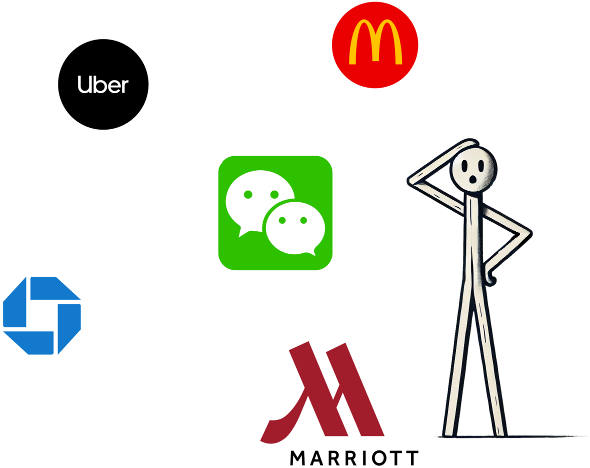
In Short...

- The first **miniapp malware** dataset
- Taxonomy of miniapp payloads
- Characterization of miniapp malware

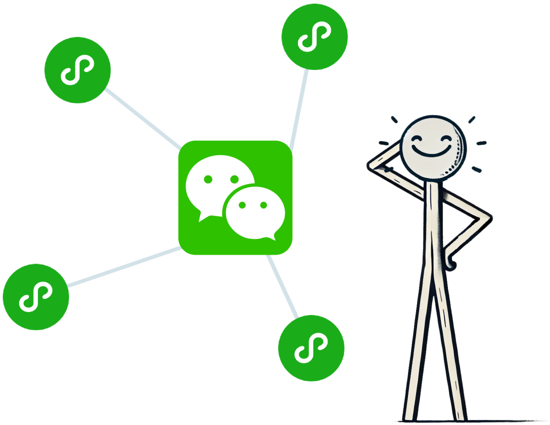
The Miniapps



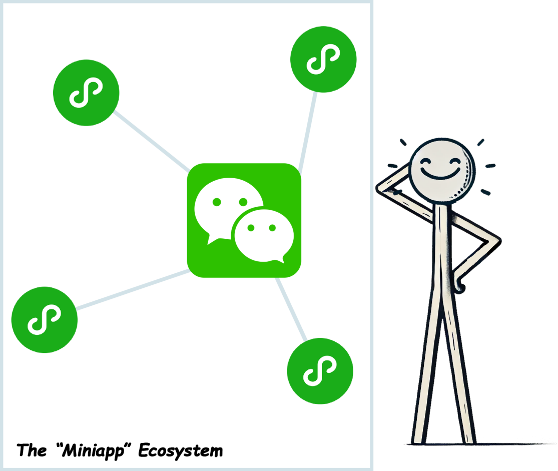
The Miniapps



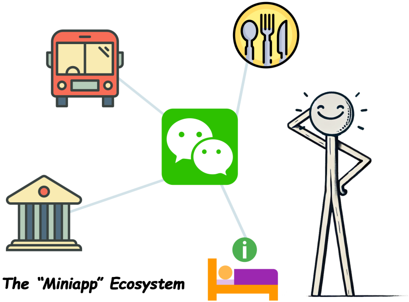
The Miniapps



The Miniapps

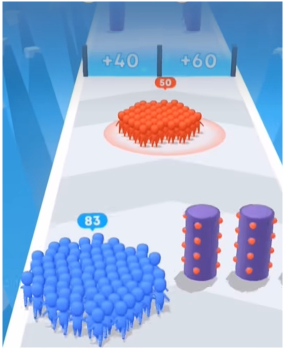


The Miniapps

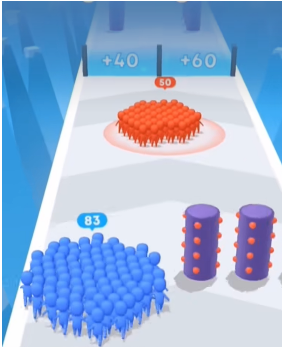


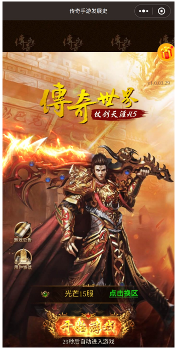
- A cross-platform solution
- Optimized versatility and functionality
- A product that "meets specific users' needs that really exist"
- Merges convenience in PC webpage and mobile QR code

The Malware Defined



The Malware Defined

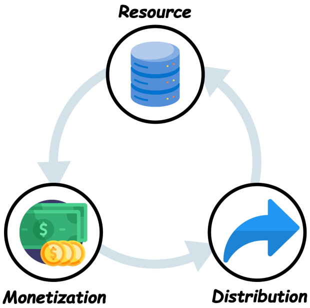




The Malware Defined

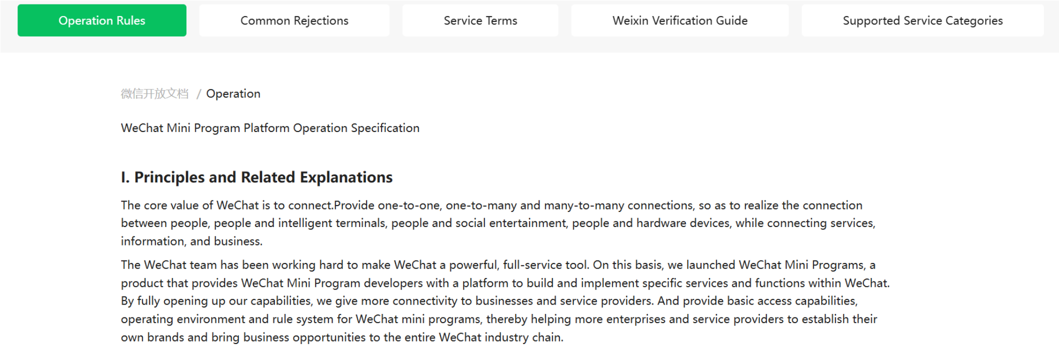
- Malicious application that:
  - Violate regulations and legislations
  - Inflict financial or privacy losses

Why We Care?



- Billions of user data
- Millions of revenue
- Rapid propagation among social network

Finding Malware Is Challenging



微信开放文档 / Operation

WeChat Mini Program Platform Operation Specification

I. Principles and Related Explanations

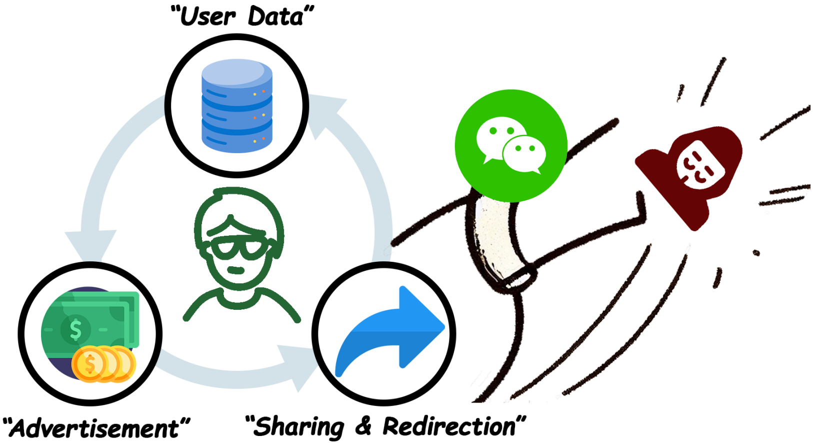
The core value of WeChat is to connect.Provide one-to-one, one-to-many and many-to-many connections, so as to realize the connection between people, people and intelligent terminals, people and social entertainment, people and hardware devices, while connecting services, information, and business.

The WeChat team has been working hard to make WeChat a powerful, full-service tool. On this basis, we launched WeChat Mini Programs, a product that provides WeChat Mini Program developers with a platform to build and implement specific services and functions within WeChat. By fully opening up our capabilities, we give more connectivity to businesses and service providers. And provide basic access capabilities, operating environment and rule system for WeChat mini programs, thereby helping more enterprises and service providers to establish their own brands and bring business opportunities to the entire WeChat industry chain.

Finding Malware Is Challenging

- Mechanism abuse (Sharing, Privacy data, Ad...)
- Fraud schemes (Net earning, Fraud gaming)
- MLM (Pyramid selling, Reciprocal promotion)
- Intellectual property violation
- ...

Evasive Behavior: Smuggling Through the Western Wall of

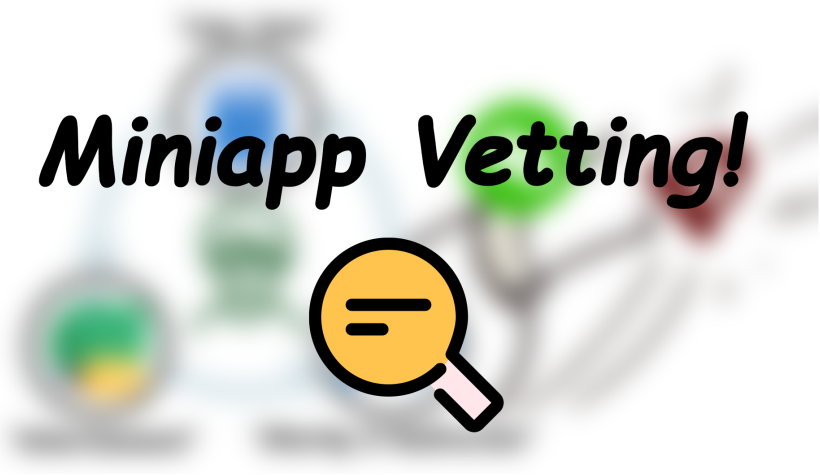


"User Data"

"Advertisement"

"Sharing & Redirection"

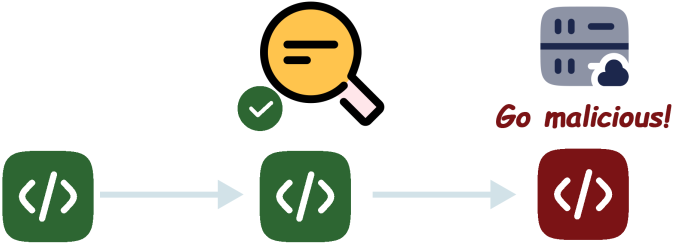
Evasive Behavior: Smuggling Through the Western Wall of



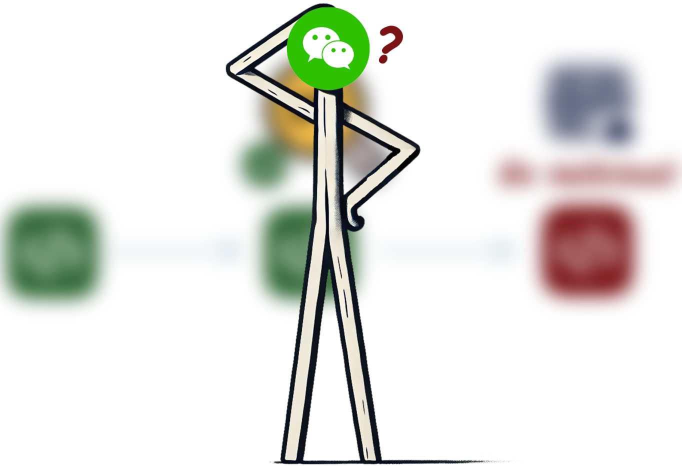
Miniapp Vetting!

Evasive Behavior: Smuggling Through the Western Wall of

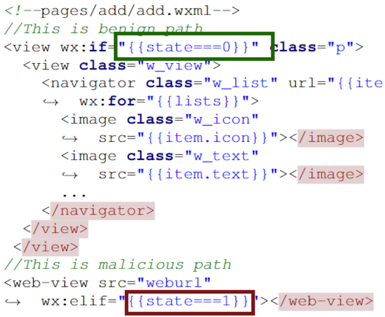
Vetting Evasion: The Malware With Two Faces



Vetting Evasion: The Malware With Two Faces

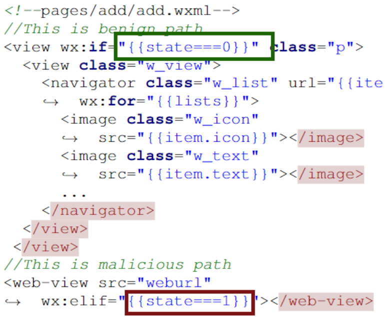


Content Vetting Evasion

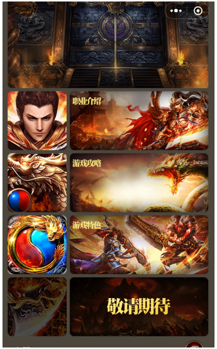


```
<!--pages/add/add.wxml-->
//This is benign path
<view wx:if="{{state===0}}" class="p">
  <view class="w_view">
    <navigator class="w_list" url="{{ite
    ↪  wx:for="{{lists}}">
      <image class="w_icon"
      ↪  src="{{item.icon}}"></image>
      <image class="w_text"
      ↪  src="{{item.text}}"></image>
      ...
    </navigator>
  </view>
 </view>
//This is malicious path
<web-view src="weburl"
↪  wx:elif="{{state===1}}"></web-view>
```

Content Vetting Evasion

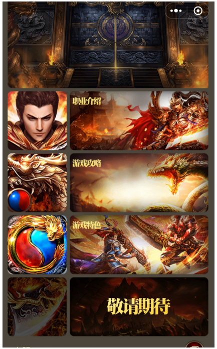


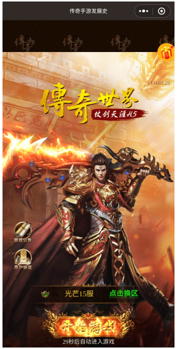
```
<!--pages/add/add.wxml-->
//This is benign path
<view wx:if="{{state===0}}" class="p">
  <view class="w_view">
    <navigator class="w_list" url="{{ite
    ↪   wx:for="{{lists}}">
      <image class="w_icon"
      ↪   src="{{item.icon}}"></image>
      <image class="w_text"
      ↪   src="{{item.text}}"></image>
      ...
    </navigator>
  </view>
 </view>
//This is malicious path
<web-view src="weburl"
↪   wx:elif="{{state===1}}"></web-view>
```



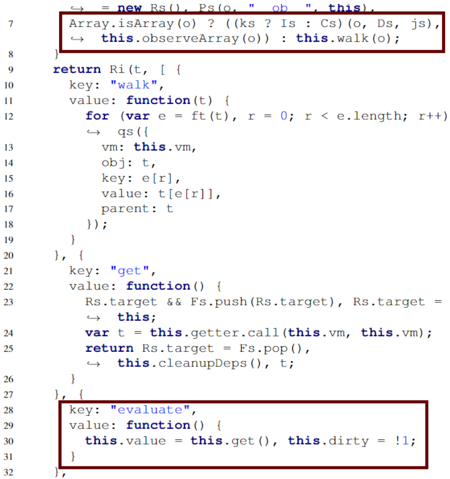
Content Vetting Evasion

```
<!--pages/add/add.wxml-->
//This is benign path
<view wx:if="{{state===0}}" class="p">
  <view class="w_view">
    <navigator class="w_list" url="{{ite
    ↪   wx:for="{{lists}}">
      <image class="w_icon"
      ↪   src="{{item.icon}}"></image>
      <image class="w_text"
      ↪   src="{{item.text}}"></image>
      ...
    </navigator>
  </view>
 </view>
//This is malicious path
<web-view src="weburl"
↪   wx:elif="{{state===1}}"></web-view>
```



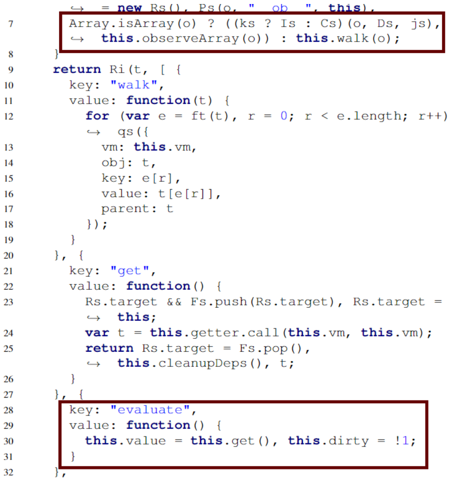


Code Vetting Evasion

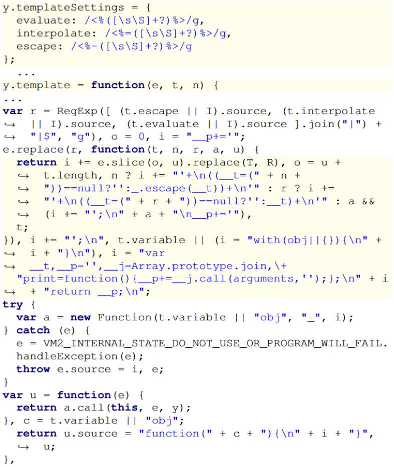


```
    ↪  = new Rs(), Ps(o, "  ob  ", this),
    Array.isArray(o) ? ((ks ? Is : Cs)(o, Ds, js),
    ↪  this.observeArray(o)) : this.walk(o);
  }
  return Ri(t, [ {
    key: "walk",
    value: function(t) {
      for (var e = ft(t), r = 0; r < e.length; r++)
      ↪  qs({
        vm: this.vm,
        obj: t,
        key: e[r],
        value: t[e[r]],
        parent: t
      });
    }
  }, {
    key: "get",
    value: function() {
      Rs.target && Fs.push(Rs.target), Rs.target =
      ↪  this;
      var t = this.getter.call(this.vm, this.vm);
      return Rs.target = Fs.pop(),
      ↪  this.cleanupDeps(), t;
    }
  }, {
    key: "evaluate",
    value: function() {
      this.value = this.get(), this.dirty = !1;
    }
  },
```

Code Vetting Evasion

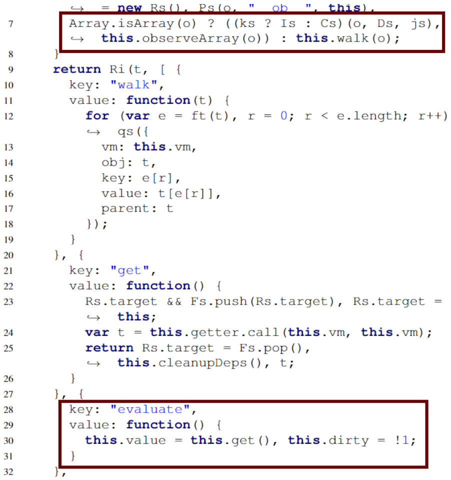


```
      ↪  = new Rs(), Ps(o, "__ob__", this),
      Array.isArray(o) ? ((ks ? Is : Cs)(o, Ds, js),
      ↪  this.observeArray(o)) : this.walk(o);
    }
    return Ri(t, [ {
      key: "walk",
      value: function(t) {
        for (var e = ft(t), r = 0; r < e.length; r++)
        ↪  qs({
          vm: this.vm,
          obj: t,
          key: e[r],
          value: t[e[r]],
          parent: t
        });
      }
    }, {
      key: "get",
      value: function() {
        Rs.target && Fs.push(Rs.target), Rs.target =
        ↪  this;
        var t = this.getter.call(this.vm, this.vm);
        return Rs.target = Fs.pop(),
        ↪  this.cleanupDeps(), t;
      }
    }, {
      key: "evaluate",
      value: function() {
        this.value = this.get(), this.dirty = !1;
      }
    },
```



```
y.templateSettings = {
  evaluate: /<%([\s\S]+?)%>/g,
  interpolate: /<%=([\s\S]+?)%>/g,
  escape: /<%-([\s\S]+?)%>/g
};
  ...
y.template = function(e, t, n) {
...
var r = RegExp([ (t.escape || I).source, (t.interpolate
↪  || I).source, (t.evaluate || I).source ].join("|") +
↪  "|$", "g"), o = 0, i = "__p+='";
e.replace(r, function(t, n, r, a, u) {
  return i += e.slice(o, u).replace(T, R), o = u +
  ↪  t.length, n ? i += "'+\n((__t=(" + n +
  ↪  "))==null?'':_.escape(__t))+\n'" : r ? i +=
  ↪  "'+\n((__t=(" + r + "))==null?'':__t)+\n'" : a &&
  ↪  (i += "';\n" + a + "\n__p+='"),
  t;
}), i += "';\n", t.variable || (i = "with(obj||{}){\n" +
↪  i + "}\n"), i = "var
↪  __t,__p='',__j=Array.prototype.join,\+
  "print=function(){__p+=__j.call(arguments,'');};\n" + i
↪  + "return __p;\n";
try {
  var a = new Function(t.variable || "obj", "_", i);
} catch (e) {
  e = VM2_INTERNAL_STATE_DO_NOT_USE_OR_PROGRAM_WILL_FAIL.
  handleException(e);
  throw e.source = i, e;
}
var u = function(e) {
  return a.call(this, e, y);
}, c = t.variable || "obj";
  return u.source = "function(" + c + "){\n" + i + "}",
  ↪  u;
},
```

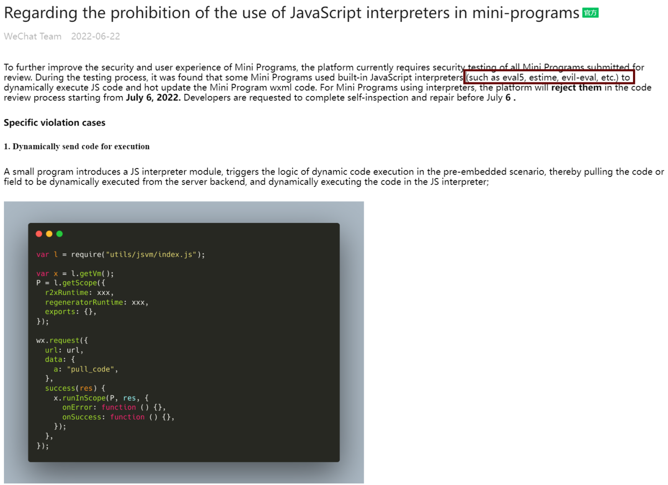
Code Vetting Evasion



```
    ↪  = new Rs(), Ps(o, "  ob  ", this),
    Array.isArray(o) ? ((ks ? Is : Cs)(o, Ds, js),
    ↪  this.observeArray(o)) : this.walk(o);
  }
  return Ri(t, [ {
    key: "walk",
    value: function(t) {
      for (var e = ft(t), r = 0; r < e.length; r++)
      ↪  qs({
        vm: this.vm,
        obj: t,
        key: e[r],
        value: t[e[r]],
        parent: t
      });
    }
  }, {
    key: "get",
    value: function() {
      Rs.target && Fs.push(Rs.target), Rs.target =
      ↪  this;
      var t = this.getter.call(this.vm, this.vm);
      return Rs.target = Fs.pop(),
      ↪  this.cleanupDeps(), t;
    }
  }, {
    key: "evaluate",
    value: function() {
      this.value = this.get(), this.dirty = !1;
    }
  },
```

- Implements APIs to evaluate node value
- Resembles relevant code in hot update libs

Oracle: Hot-update Libraries Banned Since 2022



Regarding the prohibition of the use of JavaScript interpreters in mini-programs 官方

WeChat Team 2022-06-22

To further improve the security and user experience of Mini Programs, the platform currently requires security testing of all Mini Programs submitted for review. During the testing process, it was found that some Mini Programs used built-in JavaScript interpreters (such as eval5, estime, evil-eval, etc.) to dynamically execute JS code and hot update the Mini Program wxml code. For Mini Programs using interpreters, the platform will **reject them** in the code review process starting from **July 6, 2022.** Developers are requested to complete self-inspection and repair before July **6 .**

Specific violation cases

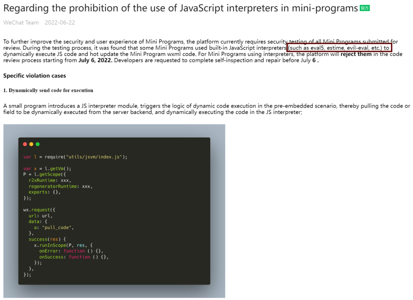
1. Dynamically send code for execution

A small program introduces a JS interpreter module, triggers the logic of dynamic code execution in the pre-embedded scenario, thereby pulling the code or field to be dynamically executed from the server backend, and dynamically executing the code in the JS interpreter;

```
var l = require("utils/jsvm/index.js");

var x = l.getVm();
P = l.getScope({
  r2xRuntime: xxx,
  regeneratorRuntime: xxx,
  exports: {},
});

wx.request({
  url: url,
  data: {
    a: "pull_code",
  },
  success(res) {
    x.runInScope(P, res, {
      onError: function () {},
      onSuccess: function () {},
    });
  },
});
```

Oracle: Hot-update Libraries Banned Since 2022

- Hot-update is complex to implement
- Developers tend to reimplement libraries
- Function signatures are kept (e.g., name and params)



Regarding the prohibition of the use of JavaScript interpreters in mini-programs 官方

WeChat Team 2022-06-22

To further improve the security and user experience of Mini Programs, the platform currently requires security testing of all Mini Programs submitted for review. During the testing process, it was found that some Mini Programs used built-in JavaScript interpreters (such as eval5, estime, evil-eval, etc.) to dynamically execute JS code and hot update the Mini Program wxml code. For Mini Programs using interpreters, the platform will **reject them** in the code review process starting from **July 6, 2022.** Developers are requested to complete self-inspection and repair before July **6 .**

Specific violation cases

1. Dynamically send code for execution

A small program introduces a JS interpreter module, triggers the logic of dynamic code execution in the pre-embedded scenario, thereby pulling the code or field to be dynamically executed from the server backend, and dynamically executing the code in the JS interpreter;

```
var l = require("utils/jsvm/index.js");

var x = l.getVm();
P = l.getScope({
  r2xRuntime: xxx,
  regeneratorRuntime: xxx,
  exports: {},
});

wx.request({
  url: url,
  data: {
    a: "pull_code",
  },
  success(res) {
    x.runInScope(P, res, {
      onError: function () {},
      onSuccess: function () {},
    });
  },
});
```

The Analysis Protocol

- Insight: evasion techniques leave traces in code

The Analysis Protocol

- Insight: evasion techniques leave traces in code
- The “Evasive signature” check:

The Analysis Protocol

- Insight: evasion techniques leave traces in code
- The "Evasive signature" check:
    - Code-based evasion: JS function signatures of evasive libraries
    - Content-based evasion: WXML signatures on webviews in conditional rendering

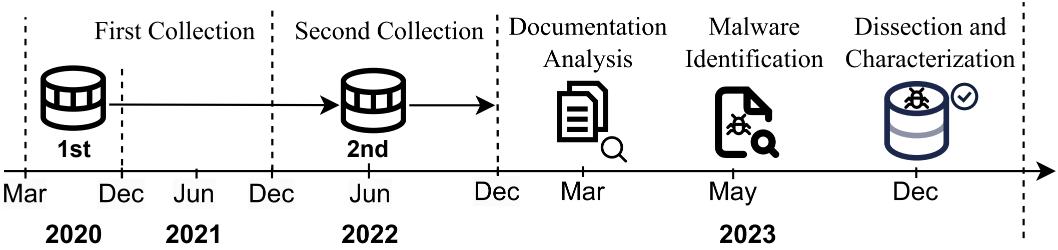
The Analysis Protocol

- Insight: evasion techniques leave traces in code
- The "Evasive signature" check:
  - Code-based evasion: JS function signatures of evasive libraries
  - Content-based evasion: WXML signatures on webviews in conditional rendering
- The "Platform removal" check:

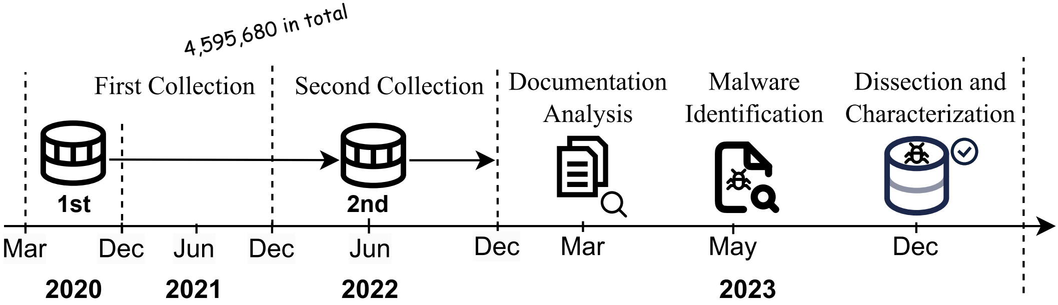
The Analysis Protocol

- Insight: evasion techniques leave traces in code
- The "Evasive signature" check:
  - Code-based evasion: JS function signatures of evasive libraries
  - Content-based evasion: WXML signatures on webviews in conditional rendering
- The "Platform removal" check:
  - Delisted miniapps are highly likely to violate regulation
  - Finding delisted miniapps helps to certify "evasive signature" check

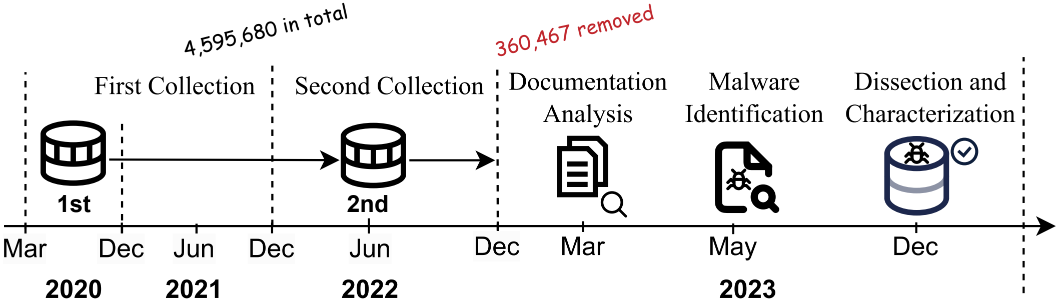
Detection


First Collection
Second Collection
Documentation Analysis
Malware Identification
Dissection and Characterization
1st
2nd
Mar
Dec
Jun
Dec
Jun
Dec
Mar
May
Dec
2020
2021
2022
2023

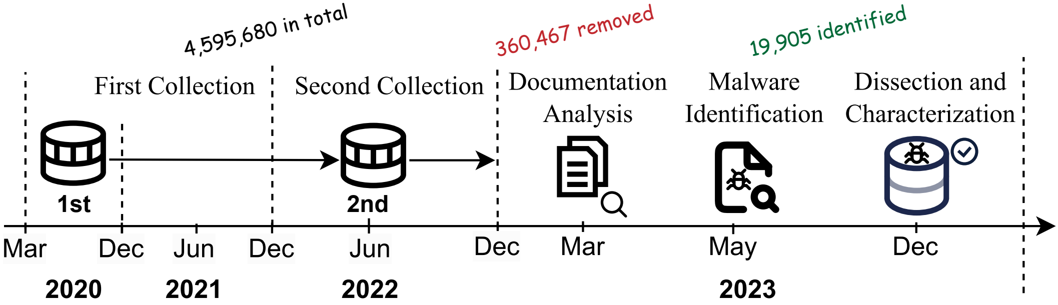
Detection


4,595,680 in total
First Collection
Second Collection
Documentation Analysis
Malware Identification
Dissection and Characterization
1st
2nd
Mar
Dec
Jun
Dec
Jun
Dec
Mar
May
Dec
2020
2021
2022
2023

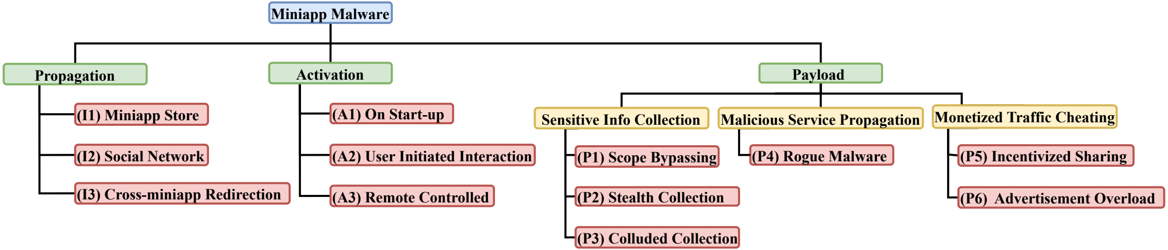
Detection


4,595,680 in total
360,467 removed
First Collection
Second Collection
Documentation Analysis
Malware Identification
Dissection and Characterization
1st
2nd
Mar
Dec
Jun
Dec
Jun
Dec
Mar
May
Dec
2020
2021
2022
2023

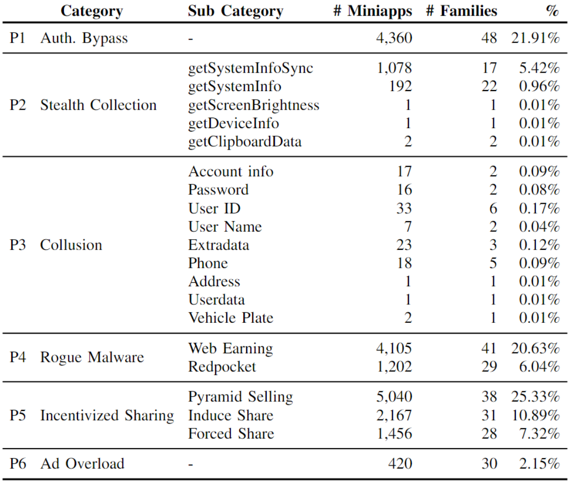
Detection


4,595,680 in total
360,467 removed
19,905 identified
First Collection
Second Collection
Documentation Analysis
Malware Identification
Dissection and Characterization
1st
2nd
Mar
Dec
Jun
Dec
Jun
Dec
Mar
May
Dec
2020
2021
2022
2023

Dissecting The Lifecycle

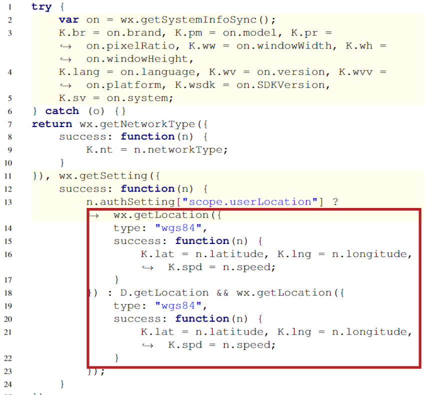


Privacy Malware



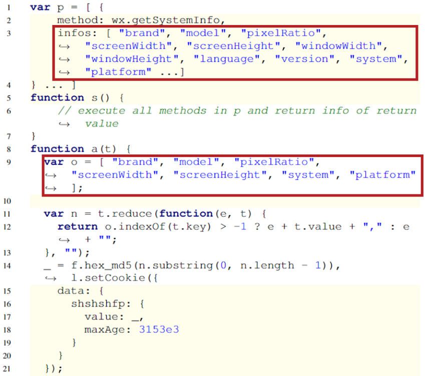
	Category	Sub Category	# Miniapps	# Families	%
P1	Auth. Bypass	-	4,360	48	21.91%
P2	Stealth Collection	getSystemInfoSync	1,078	17	5.42%
		getSystemInfo	192	22	0.96%
		getScreenBrightness	1	1	0.01%
		getDeviceInfo	1	1	0.01%
		getClipboardData	2	2	0.01%
P3	Collusion	Account info	17	2	0.09%
		Password	16	2	0.08%
		User ID	33	6	0.17%
		User Name	7	2	0.04%
		Extradata	23	3	0.12%
		Phone	18	5	0.09%
		Address	1	1	0.01%
		Userdata	1	1	0.01%
		Vehicle Plate	2	1	0.01%
P4	Rogue Malware	Web Earning	4,105	41	20.63%
		Redpocket	1,202	29	6.04%
P5	Incentivized Sharing	Pyramid Selling	5,040	38	25.33%
		Induce Share	2,167	31	10.89%
		Forced Share	1,456	28	7.32%
P6	Ad Overload	-	420	30	2.15%

Privacy Malware



```
try {
    var on = wx.getSystemInfoSync();
    K.br = on.brand, K.pm = on.model, K.pr =
    ↪  on.pixelRatio, K.ww = on.windowWidth, K.wh =
    ↪  on.windowHeight,
    K.lang = on.language, K.wv = on.version, K.wvv =
    ↪  on.platform, K.wsdk = on.SDKVersion,
    K.sv = on.system;
} catch (o) {}
return wx.getNetworkType({
    success: function(n) {
        K.nt = n.networkType;
    }
}), wx.getSetting({
    success: function(n) {
        n.authSetting["scope.userLocation"] ?
        ↪  wx.getLocation({
            type: "wgs84",
            success: function(n) {
                K.lat = n.latitude, K.lng = n.longitude,
                ↪  K.spd = n.speed;
            }
        }) : D.getLocation && wx.getLocation({
            type: "wgs84",
            success: function(n) {
                K.lat = n.latitude, K.lng = n.longitude,
                ↪  K.spd = n.speed;
            }
        });
    }
}),
```

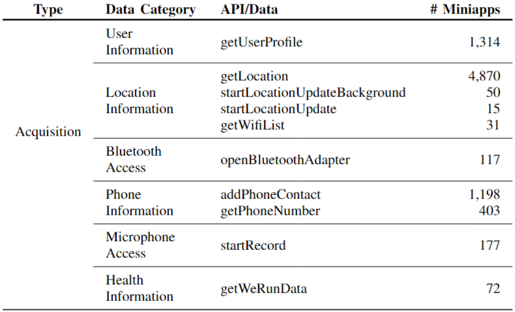
Collection upon start-up



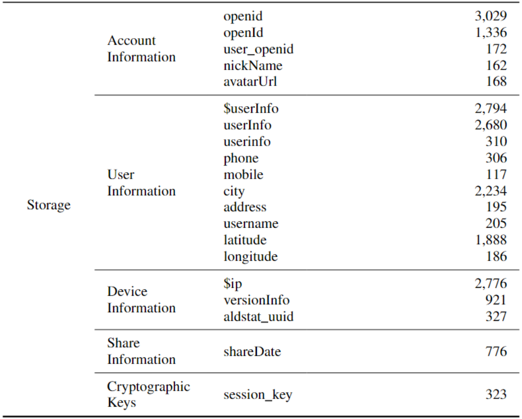
```
var p = [ {
    method: wx.getSystemInfo,
    infos: [ "brand", "model", "pixelRatio",
    ↪  "screenWidth", "screenHeight", "windowWidth",
    ↪  "windowHeight", "language", "version", "system",
    ↪  "platform" ...]
} ... ]
function s() {
    // execute all methods in p and return info of return
    ↪  value
}
function a(t) {
  var o = [ "brand", "model", "pixelRatio",
  ↪  "screenWidth", "screenHeight", "system", "platform"
  ↪  ];

  var n = t.reduce(function(e, t) {
    return o.indexOf(t.key) > -1 ? e + t.value + "," : e
    ↪  + "";
  }, "");
  _ = f.hex_md5(n.substring(0, n.length - 1)),
  ↪  l.setCookie({
    data: {
      shshshfp: {
        value: _,
        maxAge: 3153e3
      }
    }
  });
}
```

Fingerprinting user device info

Privacy Malware

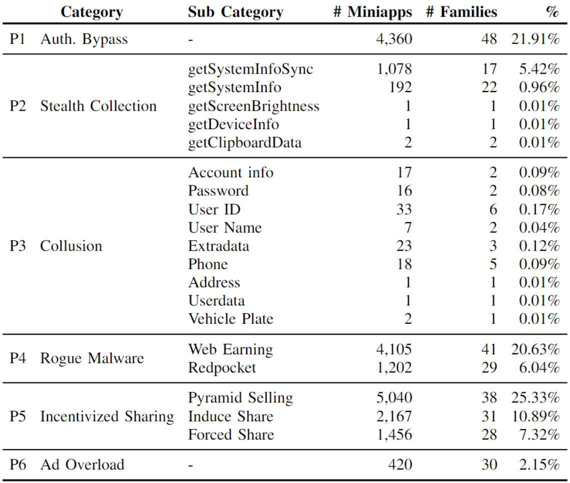


Type	Data Category	API/Data	# Miniapps
Acquisition	User Information	getUserProfile	1,314
	Location Information	getLocation startLocationUpdateBackground startLocationUpdate getWifiList	4,870 50 15 31
	Bluetooth Access	openBluetoothAdapter	117
	Phone Information	addPhoneContact getPhoneNumber	1,198 403
	Microphone Access	startRecord	177
	Health Information	getWeRunData	72



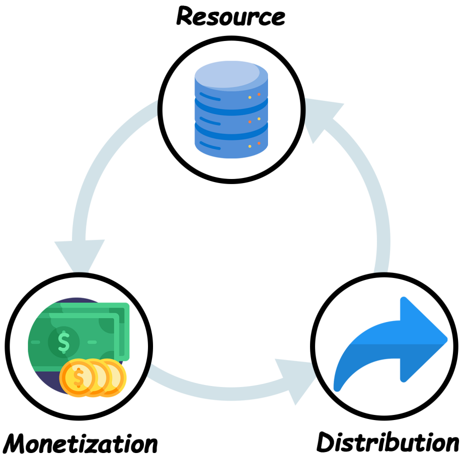
Storage	Account Information	openid openId user_openid nickName avatarUrl	3,029 1,336 172 162 168
	User Information	$userInfo userInfo userinfo phone mobile city address username latitude longitude	2,794 2,680 310 306 117 2,234 195 205 1,888 186
	Device Information	$ip versionInfo aldstat_uuid	2,776 921 327
	Share Information	shareDate	776
	Cryptographic Keys	session_key	323

Monetizing Malware

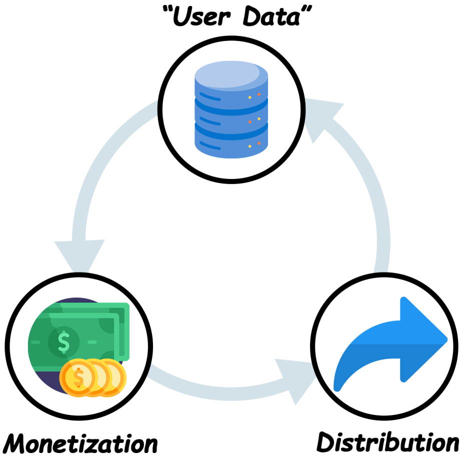


	Category	Sub Category	# Miniapps	# Families	%
P1	Auth. Bypass	-	4,360	48	21.91%
P2	Stealth Collection	getSystemInfoSync	1,078	17	5.42%
		getSystemInfo	192	22	0.96%
		getScreenBrightness	1	1	0.01%
		getDeviceInfo	1	1	0.01%
		getClipboardData	2	2	0.01%
P3	Collusion	Account info	17	2	0.09%
		Password	16	2	0.08%
		User ID	33	6	0.17%
		User Name	7	2	0.04%
		Extradata	23	3	0.12%
		Phone	18	5	0.09%
		Address	1	1	0.01%
		Userdata	1	1	0.01%
		Vehicle Plate	2	1	0.01%
P4	Rogue Malware	Web Earning	4,105	41	20.63%
		Redpocket	1,202	29	6.04%
P5	Incentivized Sharing	Pyramid Selling	5,040	38	25.33%
		Induce Share	2,167	31	10.89%
		Forced Share	1,456	28	7.32%
P6	Ad Overload	-	420	30	2.15%

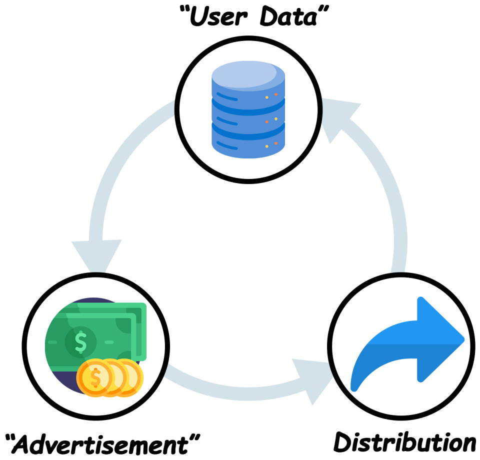
Monetizing Malware



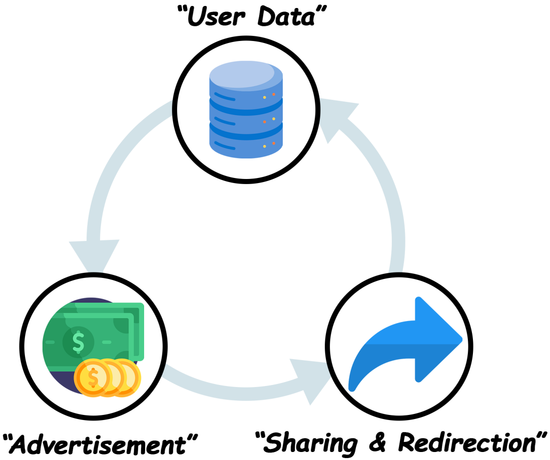
Monetizing Malware



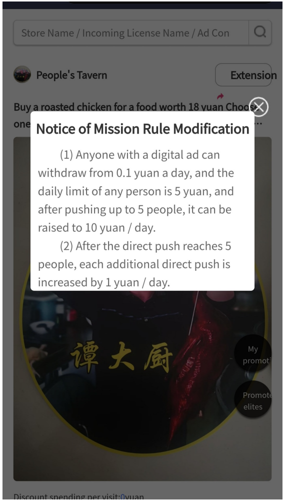
Monetizing Malware



Monetizing Malware



Monetizing Malware



Store Name / Incoming License Name / Ad Con

People's Tavern

Extension

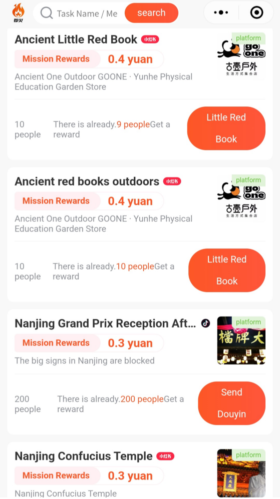
Buy a roasted chicken for a food worth 18 yuan Choose one ...

Notice of Mission Rule Modification

(1) Anyone with a digital ad can withdraw from 0.1 yuan a day, and the daily limit of any person is 5 yuan, and after pushing up to 5 people, it can be raised to 10 yuan / day.

(2) After the direct push reaches 5 people, each additional direct push is increased by 1 yuan / day.

Discount spending per visit:0yuan



Task Name / Me search

Ancient Little Red Book

Mission Rewards 0.4 yuan

Ancient One Outdoor GOONE · Yunhe Physical Education Garden Store

10 people There is already.9 peopleGet a reward

Little Red Book

Ancient red books outdoors

Mission Rewards 0.4 yuan

Ancient One Outdoor GOONE · Yunhe Physical Education Garden Store

10 people There is already.10 peopleGet a reward

Little Red Book

Nanjing Grand Prix Reception Aft...

Mission Rewards 0.3 yuan

The big signs in Nanjing are blocked

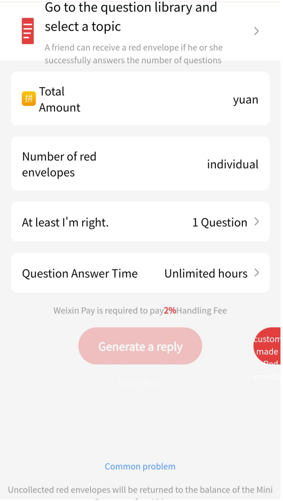
200 people There is already.200 peopleGet a reward

Send Douyin

Nanjing Confucius Temple

Mission Rewards 0.3 yuan

Nanjing Confucius Temple



Go to the question library and select a topic

A friend can receive a red envelope if he or she successfully answers the number of questions

Total Amount yuan

Number of red envelopes individual

At least I'm right. 1 Question

Question Answer Time Unlimited hours

Weixin Pay is required to pay2%Handling Fee

Generate a reply

custom made Red envelo

Common problem

Uncollected red envelopes will be returned to the balance of the Mini

Threat to Validity

- Sampled 500 miniapps

Threat to Validity

- Sampled 500 miniapps
  - 34 content vetting evasion
  - 466 code vetting evasion

Threat to Validity

- Sampled 500 miniapps
  - 34 content vetting evasion
  - 466 code vetting evasion
- 13 false positives (2.6%)

Threat to Validity

- Sampled 500 miniapps
  - 34 content vetting evasion
  - 466 code vetting evasion
- 13 false positives (2.6%)
  - 10 semantic issue on evasive API
  - 3 non-malicious webview displaying

Recap

- **Main contribution**: identified, dissected, released a miniapp malware dataset

Recap

- **Main contribution**: identified, dissected, released a miniapp malware dataset
- **Maliciousness leave traces**: vetting evasion leave identifiable code signature

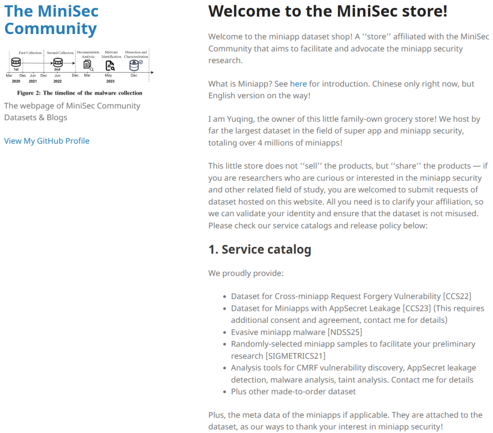
Recap

- **Main contribution**: identified, dissected, released a miniapp malware dataset
- **Maliciousness leave traces**: vetting evasion leave identifiable code signature
- **Enhanced devastation**: wider privacy impact, faster propagation

Recap

- **Main contribution**: identified, dissected, released a miniapp malware dataset
- **Maliciousness leave traces**: vetting evasion leave identifiable code signature
- **Enhanced devastation**: wider privacy impact, faster propagation
- **Domain-specific uniqueness**: the platforms can be victims!

Dataset Release



The MiniSec Community

Figure 2: The timeline of the malware collection

The webpage of MiniSec Community Datasets & Blogs

View My GitHub Profile

Welcome to the MiniSec store!

Welcome to the miniapp dataset shop! A ''store'' affiliated with the MiniSec Community that aims to facilitate and advocate the miniapp security research.

What is Miniapp? See here for introduction. Chinese only right now, but English version on the way!

I am Yuqing, the owner of this little family-own grocery store! We host by far the largest dataset in the field of super app and miniapp security, totaling over 4 millions of miniapps!

This little store does not ''sell'' the products, but ''share'' the products — if you are researchers who are curious or interested in the miniapp security and other related field of study, you are welcomed to submit requests of dataset hosted on this website. All you need is to clarify your affiliation, so we can validate your identity and ensure that the dataset is not misused. Please check our service catalogs and release policy below:

1. Service catalog

We proudly provide:

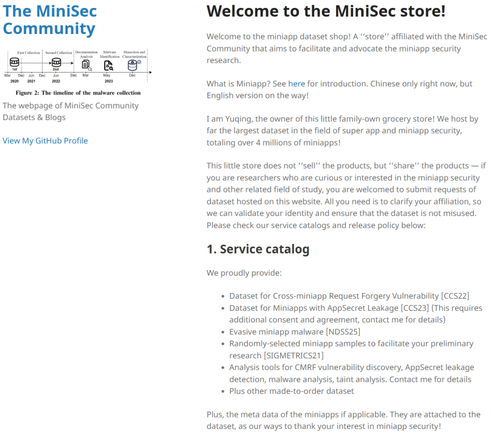
- Dataset for Cross-miniapp Request Forgery Vulnerability [CCS22]
- Dataset for Miniapps with AppSecret Leakage [CCS23] (This requires additional consent and agreement, contact me for details)
- Evasive miniapp malware [NDSS25]
- Randomly-selected miniapp samples to facilitate your preliminary research [SIGMETRICS21]
- Analysis tools for CMRF vulnerability discovery, AppSecret leakage detection, malware analysis, taint analysis. Contact me for details
- Plus other made-to-order dataset

Plus, the meta data of the miniapps if applicable. They are attached to the dataset, as our ways to thank your interest in miniapp security!

Dataset Release

- Nanjing University, China
- Xidian University, China
- Rochester Institute of Technology, USA
- Johns Hopkins University, USA
- Xi'an Jiao Tong University, China
- University of Science and Technology Beijing, China
- Chinese Academy of Sciences, China
- Peking University, China

Dataset Release



The MiniSec Community

Figure 2: The timeline of the malware collection

The webpage of MiniSec Community Datasets & Blogs

View My GitHub Profile

Welcome to the MiniSec store!

Welcome to the miniapp dataset shop! A ''store'' affiliated with the MiniSec Community that aims to facilitate and advocate the miniapp security research.

What is Miniapp? See here for introduction. Chinese only right now, but English version on the way!

I am Yuqing, the owner of this little family-own grocery store! We host by far the largest dataset in the field of super app and miniapp security, totaling over 4 millions of miniapps!

This little store does not ''sell'' the products, but ''share'' the products — if you are researchers who are curious or interested in the miniapp security and other related field of study, you are welcomed to submit requests of dataset hosted on this website. All you need is to clarify your affiliation, so we can validate your identity and ensure that the dataset is not misused. Please check our service catalogs and release policy below:

1. Service catalog

We proudly provide:

- Dataset for Cross-miniapp Request Forgery Vulnerability [CCS22]
- Dataset for Miniapps with AppSecret Leakage [CCS23] (This requires additional consent and agreement, contact me for details)
- Evasive miniapp malware [NDSS25]
- Randomly-selected miniapp samples to facilitate your preliminary research [SIGMETRICS21]
- Analysis tools for CMRF vulnerability discovery, AppSecret leakage detection, malware analysis, taint analysis. Contact me for details
- Plus other made-to-order dataset

Plus, the meta data of the miniapps if applicable. They are attached to the dataset, as our ways to thank your interest in miniapp security!

Thank You

Understanding the Miniapp Malware: Identification, Dissection, and Characterization

Yuqing Yang [1] Yue Zhang [2] Zhiqiang Lin [1]

[1] The Ohio State University

[2]Drexel University

Feb 25th, 2025