eSIM security

CHOOSE A PROJECT

General info

In a result of its research investigation efforts, Security Explorations, a research lab of AG Security Research company, conducted security analysis of eSIM technology.

This section of our website presents initial information regarding the project.

Notes

We broke security of Kigen(*) eUICC card with GSMA consumer certificates installed into it.

The eUICC card makes it possible to install the so called eSIM profiles into target chip. eSIM profiles are software representations of mobile subscriptions. For many years such mobile subscriptions had a form of a physical SIM card of various factors (SIM, microSIM, nonoSIM). With eSIM, the subscription can come in a pure digital form (as a software bundle), it can also carry Java Card applications.

According to Kigen:

1) eSIMs are "as secure and interoperable as SIM cards [...] thanks to the multi-layered GSMA eSIM certification scheme that protects device makers, device owners and mobile network operators (MNOs)" (source)

2) "Kigen OS offers the highest level of logical security when employed on any SIM form factor, including a secure enclave" and "Kigen SIM OS features help differentiate, scale and grow revenues with zero compromise security" (source)

The hack proves that our research on Java Card from 2019 did matter. Oracle indicated the vulnerabilities we reported to the company in 2019 were rather irrelevant (the company referred to them as "security concerns") / did not affect their production Java Card VM. These are now proved to be real bugs.

This is likely the first successful public hack against:

consumer GSMA eUICC
Kigen eSIM (Kigen press releases and web pages implicate over 2 billion SIMs enabled by Kigen secure SIM OS)
EAL(**) certified GSMA security chip (SLC37 chip based on 32-bit ARM SecurCore SC300 processor from Infineon)

The attack against Kigen eUICC relies both on physical access to sample card along knowledge of the keys used for malicious Java app installation. The remote over-the-air (OTA) vector can't be excluded - our Proof of Concept code mimics a malicious applet installation over OTA SMS-PP protocol (Short Message Service Point to Point) on a target Kigen eUICC. In that context, knowledge of the keys is a primary requirement for target card compromise.

From a technical point of view, both voice and data only (traveller's) eSIM profiles may carry potentially malicious JavaCard applications.

The hack proves no security / isolation for the eSIM profile and Java apps (no security for eUICC memory content).

It's worth to note that while this work builds on our past Java Card research from 2019 (along [25] [years] [of] [Java] [hacking] experience), it required development of some new exploitation techniques / know-how.

We hope the hack brings eSIM security along associated security risks to the focus of mobile network operators (MNOs), vendors, security researchers and security companies. This is important in the context of somewhat bold security claims / overconfidence of eUICC vendors (vide leaf eUICC cert valid for 100 years) and MNO assumptions pertaining to profile trust and its storage in a tamper-proof security element (MNO profile integrity / no compromise / no tampering assumed).

(*) as denoted by eUICC / EUM certificates and card EID

(**) RSP specification impose minimum EAL4 level ("4.14.1 eUICC Certification Requirements", page 55), Infineon pages for SLI37CMXXX chip featuring Kigen eSIM OS in the context of SLx17 product line mention CC EAL6+ high certification

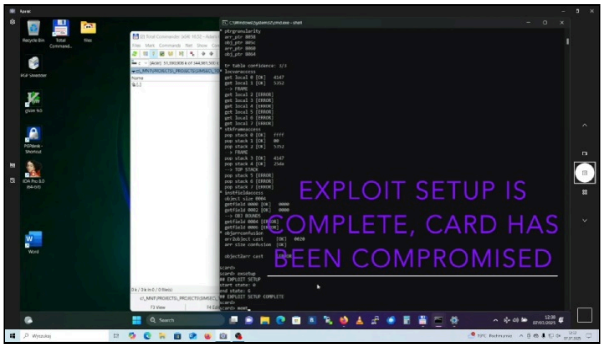
Crypto Proof

As a result of eUICC compromise, we were able to extract private ECC key for the certificate identifying target GSMA card.

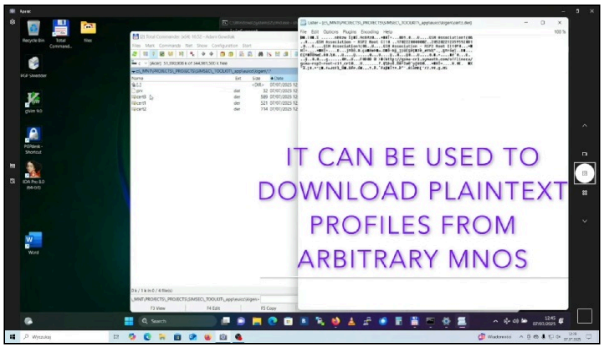
The cryptographic proof for a successful eUICC compromise can be downloaded from this location. The proof was provided to Kigen on Mar 17, 2025 (it was confirmed by the company on Mar 20, 2025).

Demonstration movies

"Kigen eUICC simulated over-the-air app install and card compromise", MP4 movie file, 39MB



"Kigen eUICC identity certificate and private key extraction", MP4 movie file, 13MB



GSMA certificate theft implications

The theft of arbitrary GSMA consumer certificate has the following implications:

1) one can download arbitrary profiles ("software" SIM cards)* from MNOs in cleartext, we have used Kigen eUICC cert to download decrypted eSIM profiles for various MNOs (ATT, Vodafone, O2, Orange, Bouygues Telecom, DTAC, China Mobile, CMHK and T-Mobile in particular), see sample profile output

2) GSMA cert theft makes UICC security element (certified, tamper-proof one) irrelevant - no need to hack the HW / UICC card of which goal is to keep MNO secrets secure

3) profiles contain MNO secrets (subscriber / network configs, secret / OTA keys, Java apps), some apps might be sensitive (call control, etc.), some keys might be shared across profiles / UICCs (for over-the-air SMSPP and/or Cell Broadcast)

4) network operators key OPc and Authentication Management Field (AMF) are two very critical secret keys embedded in eSIM profiles, these should be safeguarded by the network operators at any cost (see this Medium post on LTE security for explanation)

5) the apps can be extracted for inspection (for bug hunting / secrets, etc.), the functionality of our codes can extract and disassemble the apps in an automatic fashion (see sample app extraction output from `loadBlock` profile section)

6) the downloaded profile can be modified and put into arbitrary eUICC, we have verified that such a modified profile (with custom Java STK app injected) works fine (for calls, messaging, etc.). Currently, there is no ability for the MNO to detect their profile has been tampered. Modified Orange profile loaded into test eUICC card and used with Samsung 5G phone is illustrated by screenshot 1 and screenshot 2.

7) the downloaded profile can be potentially modified in such a way, so that the operator loses control over the profile (no ability for remote control / no ability to disable / invalidate it, etc.), the operator can be provided with a completely false view of the profile state (the result of its remote management ops in particular) or all of its activity can be subject to monitoring (vide rogue apps mimicking / proxying behavior of real ones)

In our opinion, the ability for a single broken eUICC / single eUICC GSMA cert theft to peek into (download in plaintext) eSIMs of arbitrary MNO constitutes a significant eSIM architecture weak point (we see no base for extending MNO trust beyond its own network).

(*) all the testing were done for profiles we owned (purchased), access to eSIM profiles of other users should not be possible for eSIM profiles that have been already downloaded (bound to eUICC card indicated during the first download request)

Kigen notification / reports

We initiated contact with Kigen on Mar 17, 2025 (crypto proof delivery to media addr due to missing dedicated security contact). On Mar 21, 2025 Kigen was provided with a technical document describing successful compromise of its eUICC and secrets theft.

The document was provided to Kigen completely free of any charge and without any conditions. We however indicated to Kigen that it should not publish / should not share the info received with any other 3rd party (outside of Kigen) as it contains our unpublished know-how / IP.

Per response received from Kigen on Mar 21, 2025, the company has all the information required to start work aimed at addressing the issues (*"The level of detail in your findings has been incredibly helpful - your work has played a crucial role in enabling us to fully understand the issue and begin crafting effective mitigation strategies"*).

On Mar 30, 2025, Kigen was provided with additional information about multiple (100+) security vulnerabilities affecting its Java Card VM implementation.

On Apr 1, 2025, we provided Kigen with a brief technical doc explaining potential approach to the addressing / mitigation of the 2 core issues exploited in the attack.

On Apr 17, 2025, Kigen asked for "the list of entities we intend to name (planned to be referenced in the July 2025 disclosure), so that Kigen could ensure timely coordination with those organizations. On Apr 18, 2025, we provided Kigen with a full list of MNO / EID corresponding to eSIM profiles downloaded with the use of a compromised Kigen eUICC identity.

On Jun 10, 2025, we informed Kigen that we developed a Proof of Concept code that is able to reliably and in an automatic fashion exploit Kigen eUICC cards (card compromise, memory access, eUICC leaf cert extraction, secret keys dump in plaintext). We also provided Kigen with a few ideas that are worth considering implementation in Kigen Sim OS as these could mitigate the exploitation scenario. Finally, we notified Kigen about a key instance shared by ECASD domain of arbitrary Kigen eUICC cards (cards available under different brands).

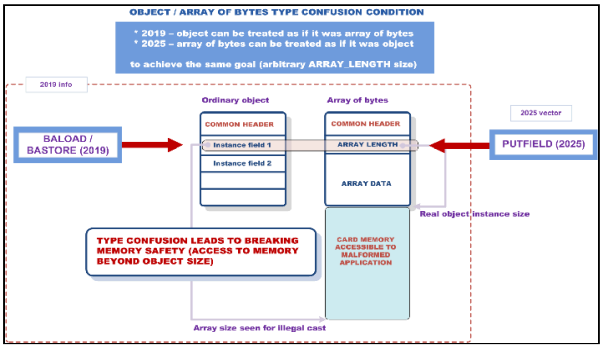
Reward from Kigen

On Mar 31, 2025, Kigen informed us that it is willing to reward us with $30K "for the detailed work we have performed to identify the vulnerability and to establish a 90-day non-disclosure period".

The reward has been paid in full by Kigen in three installments (on Day 1, Day 45 and Day 90).

The core issues

The core issues exploited in the attack against Kigen card rely on the same type confusion condition as Issues 1 and 2 from 2019:



The only difference is in the way (direction) target exploit (type confusion) condition is set up.

One might argue that this is a completely new / unknown issue (missing checks in *getfield* / *putfield* bytecodes), but we stick to the rationale that it is not as:

- 2019 issues clearly pointed out the possibility of a type confusion between an object and array instance
- 2019 issues signalled the problem of an overlap between array size and object instance field
- 2019 issues were mostly around missing type safety checks in bytecode implementation

As a minimum, the vendor should have implemented security measures that would detect type casts between arrays of bytes and object instances (in either direction) and that would hide array size field (made it part of an object header). None of that took place, thus our conclusion that the case is about old / known bugs and the impact of not addressing them (or bad addressing).

Kigen "mitigation"

Kigen decided to implement type safety checks aimed to detect arbitrary casts between object and integer types. The checks were introduced for almost all JavaCard bytecode instructions (there are ~180 of them). Unfortunately, the design / overall idea was completely broken due to the incomplete type state information (only type for the stack top checked), no real control-flow tracking and introduction of an arbitrary union / univesal type (corresponding to both integer and object type). The end result could not be different (100+ broken bytecode locations). In other words, Kigen fell the victim of anyone trying to implement some form of a bytecode verification without proper background / "state of the art" knowledge on the topic.

GSMA and Oracle notifications

Due to the potential impact of some parts of this research to the whole telecom / mobile industry, on Mar 31, 2025 we have reached out to GSMA security and provided it with a pre-disclosure notification document (Apr 07, 2025). The notification was destined for the review by the CVD Panel of Expert Companies consisting of major eUICC / mobile phone vendors.

We indicated to GSMA that the case was not directly applicable to GSMA CVD program as it was about old / known bugs and the impact of not addressing them (or bad addressing) by the industry and/or technology owner. In that context, it was more about an early notification with respect to the planned public disclosure.

We gave GSMA permission to share the notification document among the organisation and its members. We indicted that this is GSMA responsibility and decision regarding who gets early notification / info prior to the public disclosure (the 2nd week of Jul 2025, in accordance with the 90-day non-disclosure period agreed with Kigen).

On Apr 10, 2025, we also shared a pre-disclosure notification with Oracle Java Card team located in Germany. Oracle got back to us on Apr 18, 2025 via its Security Alert. The company informed that its Java Card team was reviewing the report and that it will get back to us soon once the team completes the review.

Oracle has not expressed interest in a brief technical doc explaining possible approach to address / mitigate the core issues that are used in the new exploit and offered to be shared with the company completely for free (Apr 24, 2025). The company provided results of its analysis / review of the notification doc on Jul 03, 2025. Oracle response can be seen here.

GSMA Acknowledgments

GSMA infomed us that the issues (research) never formally became a GSMA CVD case. It was closed and then referred to the eSIM group for resolution.

Whilst the case was technically out of scope for the CVD programme, GSMA indicated that it should list the CVD and our details on the acknowledgements page "in recognition of our significant efforts in this research along communication".

Vulnerable Kigen products / fixing status / patch details

The following information was received from Kigen on Jun 30, 2025 with respect to the vulnerability impact and its fixing:

> Impacted Product: Kigen ECu10.13, an eSIM product
> Severity: The compound vulnerability has been classified as CVSS v3.1 score: 6.7 (Medium)
> Vector:
> AV:P/AC:H/PR:N/UI:N/S:C/C:H/I:H/A:H/E:F/RL:O/RC:C/CR:H/IR:H/AR:M/MAV:P/MAC:H/MPR:N/MUI:N/MS:C/MC:H/MI:H/MA:H

The CVSS score provided by Kigen is an Environmental score (Base score has severity 7.1 High). This score assumes physical access vector only. With Network access vector assumed (the issues can be exploited over-the-air* upon keys knowledge as indicated in our reports provided to Kigen), the Base CVSS score is 9.1 (Critical). We notified Kigen of this potential CVSS miscalculation on Jul 01, 2025.

Kigen informed us that the company has coordinated with the GSMA to update the TS.48 Generic Test Profile specification, addressing the chain-of-trust weakness that enabled Remote Applet Management by an unauthorised actor. Simultaneously, Kigen issued a patch across all affected eSIM variants as a precautionary measure, both in-field and in inventory.

The fix disallows applet installation in TS.48 Test Profiles, in line with TS.48 v7. It introduces JavaCard Virtual Machine hardening to prevent the specific bytecode manipulation pathway outlined in our reports.

Kigen indicated that this patch has already been distributed to millions of eSIMs. The company said that it has also engaged indirectly with other eSIM vendors via the GSMA CVD programme and communicated directly with all known impacted mobile networks and customers to ensure the value of our findings is well understood across the ecosystem, and that stakeholders are equipped to mitigate accordingly.

Kigen has not provided the CVE number(s) corresponding to the reported issues. The company justified this as following:

Given the extent of private coordination and outreach - including direct notification to over 100 organisations and indirect notice to hundreds more through the CVD programme - we have opted not to request a CVE at this time. We feel the objectives of responsible disclosure have been met without introducing public registry overhead. We remain open to future CVE registration if ecosystem needs evolve.

(*) SMSPP over-the-air vector is simulated by our toolkit for malicious applet installation

Kigen security bulletin / confusing root cause

On Jul 09, 2025, Kigen issued a security bulletin with information about patches and mitigations for the reported issues.

Kigen bulletin focuses on GSMA Generic Test Profile as if this was the source for the vulnerability ("GSMA TS.48 Test Profile Vulnerability" phrasing). There is no reference to any JavaCard vulnerabilities. These seem to be hidden under the term "JavaCard Runtime Hardening". Finally, the bulletin indicates most eUICCs are not vulnerable. The term "Vulnerable" is used in the context of "GSMA TS.48 Test Profile Vulnerability" (applet loading possibility with the use of known keys), not JavaCard. Finally, Kigen bulletin doesn't carry any information about vulnerable / immune product versions. It indirectly denies the remote (OTA SMSPP) vector by indicating that "an attacker must first gain physical access to a target eUICC and use publicly known keys".

We do not agree with Kigen's bulletin content / point of view. As original vulnerabilities' discoverers, we perceive JavaCard issues as instrumental for the attack conducted against Kigen eUICC card. The possibility to load attacker's application is just the code deployment vector.

Kigen bulletin may be confusing for all 3rd parties trying to conduct threat analysis / evaluate the impact of the issues to their networks and devices. It also lets Kigen claim that the issues affected "a specific eSIM product variant tailored towards development use" when providing comments to media, which may indicate the issues have been affecting development cards only. While this may be true from a "GSMA TS.48 Test Profile Vulnerability" point of view, this might not be correct from a point of view of Java Card issues. These could affect all Kigen eSIM products due to GSMA spec requirements for JavaCard support on eUICC and Kigen's insecure JavaCard VM implementation. These could be exploited regardless of the presence of the test profile (upon keys knowledge and through other code deployment vectors).

Kigen eUICC FW check

According to information provided by Kigen, "ECu10.13" string indicates vulnerable eUICC card. We found out that this sequence is returned by the GET DATA command for an unpublished 0xDF1F item ("80 ca df 1f 00" APDU). Sample log illustrating FW version retrieval for Kigen eUICC through OTA SMSPP and our toolkit can be checked here.

Initial research impact

The research [once] [again] questions the value carried by various certification schemes.

Kigen's guide to GSMA eSIM certification indicates that GSMA eUICC (eSIM) must be certified to an assurance level of EAL4+. Assessment and issuance of the certification report is performed by a security laboratory qualified by a Common Criteria certification body (i.e. ANSSI or BSI), which conducts penetration tests aimed to verify whether designed security features are properly implemented. This is all done with respect to both an off-card actor or an on-card application.

The research also demonstrates that eSIM hacking makes things easier from a point of a backdoor / tampered profile (no need for low level, eUICC specific tampering / exploitation, which cannot be excluded - malicious Java app can still target underlying eUICC).

The hack should trigger attention of mobile network operators to the topic (there are 500+ of them according to GSMA pages).

It should revise MNOs trust / change their perception with respect to eUICC security and eSIMs in general (vide single broken eUICC / single eUICC GSMA cert theft and its power).

The warning call for mobile phone vendors

The eUICC (eSIM) chip is embedded (is part of a silicon) in most modern smart phones. While default access seems to be restricted to trusted parties only (i.e. access to Apple NFC & SE Platform and Samsung eSE SDK) in our opinion this is irrelevant (*).

If target eUICC chip runs vulnerable Java Card VM implementation, its security should be taken with cautious (vulnerable Java Card VM can lead to chip compromise and secrets extraction as illustrated by this research, backdooring of a target eUICC can't be excluded too).

Target eUICC chips may run some sensitive applications (digital wallets / payment, digital car keys, transportation cards, access / identification cards, etc.). In case of a successful eSIM compromise, the security / credibility of such apps may be affected.

With respect to the above, the research and demonstrated attack should serve as an early warning call for mobile phone vendors.

(*) our experience indicates that a target card (i.e. SIM / USIM / eUICC) running on top of a vulnerable JavaCard VM can be always compromised (100% success rate)

GSMA inquiry around eUICC / eSA certification scheme

We inquired GSMA on the issues pertaining to eUICC / eSA certification process, its costs(*), mandatory status, liabilities and "privileged" parties (with a power to provision potentially malicious apps into eSIMs). Our questions along answers provided by GSMA staff members on Jun 27, 2025 can be found in this document.

(*) in the context of the exploit presence and Kigen investments to ensure the company is building secure products

GSMA spec changes / action

As a result of this research, GSMA decided to shut down all test profiles.

The new TS.48 Generic Test Profile specification aims to address the issue by preventing the ability for an unauthorized actor to install potentially malicious Java Card applications on eSIM.

We fear this might not be enough (this might not address the core of the issue) due to the following:

> the problem has its origin in insecure Java Card VM architecture (split verification process, unverified bytecode execution)
> per specs, eUICC need to support Java Card apps (eSIM profiles can embed Java Card apps)
> there are many parties which theoretically might be able to provision arbitrary Java Card apps into eSIM (MNOs, RSP server owners, eUICC vendors, GSMA sub/certificate owners, etc.).

That alone creates a significant threat surface from a point of view of potentially bad actors (including rogue MNO / eSIM provider). One cannot exclude rougue eSIM store too.

One needs to keep in mind that as of Jul 2025, there are 44 (SAS-SM accredited sites) with a privilege to provision arbitrary eSIM profiles to eUICC chips. There are also 74 SAS-UP accredited sites that can create arbitrary eUICC identities. See this list for reference.

Guidance for the industry

On Jul 9, 2025, GSMA published Application Note to provide guidance for the industry in order to avoid the misuse of Remote Application Management (RAM) keys to install, within Profiles, a malicious Java Card Applications in an eUICC supporting Java Card technology.

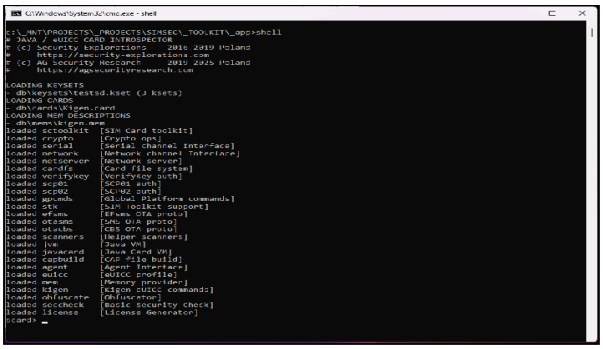
GSMA indicates this is a first set of recommendations to be used for products based on existing versions of the eSIM specifications. All concerned parties are encouraged to monitor GSMA pages for further recommendations.

Telecom expert's point of view

A summary / opinion piece regarding our findings done by Harald Welte, a reknown / long-time telecom expert with an in-depth knowledge on all sorts of telecom / eUICC / security related issues can be found in this blog post.

eUICC / Java Card exploitation toolkit

Our custom toolset makes it possible to test Java Card capable chipsets (SIM / USIM / eUICC) for the presence of security weaknesses and to investigate and exploit these in a convenient way through commands / scripting.



The toolkit implements BSC command (Basic Security Check) that provides a brief summary(*) of a target Java Card capable eUICC from a point of view of bytecode security:

```
scard> bsc
- castcheck         [Check for arbitrary int / obj casts]
- ptrgranularity    [Check for ptr granularity / tr table presence)]
- locvaraccess      [Check for arbitrary local variable access]
- stkframeaccess    [Check for arbitrary stack frame access]
- instfieldaccess   [Check for arbitrary instance field access]
- objarrconfusion   [Check for arbitrary object / array type and size field confusion]
```

Sample output / "report" for the checks conducted for Kigen eUICC card can be seen here.

The toolset makes it possible to download arbitrary eSIM profile in a plaintext and user friendly form with the use of a SW eUICC identity (a bundle of certificate chains, eUICC info and private key is completely sufficient for that).

Upon extracting the private eUICC key, the target card is not needed any more as the tool relies on a software (digital) form of the eUICC for its operation.

(*) which can't be perceived in terms of any comprehensive evaluation, but rather as providing an initial "feel" about target chip's Java Card VM security strength

Remote SIM provisioning (RSP) servers

We verified that arbitrary eSIM profiles could be downloaded from various production RSP servers including those operated by Oberthur (IDEMIA) and Thales (major SIM / USIM / eUICC card vendors).

It seems, no detection is made by RSP servers with respect to the rogue / compromised eUICC identity as of now (vide the confirmed use of a single eUICC identity for unusual(*) eSIM profile download patterns).

It's worth to note that EID of a target eUICC was required (most likely for auth / profile binding purposes) by only one of tested operators prior to eSIM activation (T-Mobile USA).

(*) understood in a context of multiple downloads, downloads conducted within a given time frame, with respect to same MNO or with respect to MNOs spread across various countries, all done with respect to same eUICC identity (EID)

Potential for RSP servers' / protocol abuse

Compromised eUICC identity could be theoretically used for crawling / scanning arbitrary RSP server in a search for eSIM profiles that haven't been downloaded yet (if weak / predictable matching id space is used by RSP server).

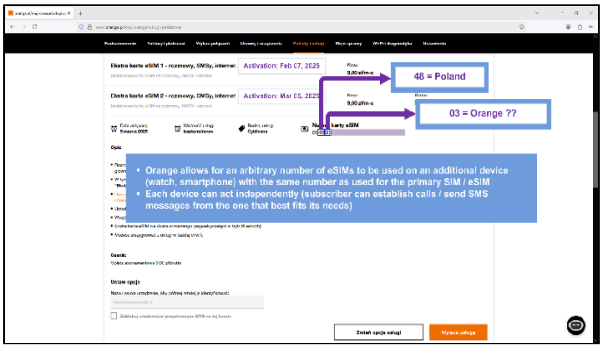
We haven't tested any RSP server in that context due to the possibility of hitting arbitrary eSIM profile in READY state. This would give us access to profile that hasn't been downloaded yet. This would make this profile not available to target user (due to eUICC identity binding). This would directly implicate access to / hijack of profiles we did not own.

In general, the key thing here is the trust of RSP servers to eUICC identities. So far, nobody assumed that an eUICC element could be compromised (that a malicious actor could be positioned at this location). This might indicate a need for revision / a more thorough look at the security / threat model used by RSP (the implication to RSP / crypto protocol* of a malicious actor abusing compromised eUICC identity). We signaled this to GSMA on Jul 14, 2025.

(*) such as the use of a fixed / predictable eUICC challenge, multiple / flood of requests from the same eUICC identity, etc.

Orange Poland mirrored eSIMs test

Orange Poland makes it possible for its subscribers to request an eSIM profile (or several of them) that acts as a mirror for subscriber's primary number (eSIM can receive all SMS and calls as if it was the primary number):



Shared secret keys test

We used a compromised Kigen eUICC identity to download the content of two Orange Extra eSIM profiles associated with a given subscriber in plaintext.

Investigation of the profiles' content revealed the following:

network operators key / OPc parameters responsible for security of the subscriper's identity were different (each eSIM was bound to a different phone number, the network routed eSIM numbers as if they were a primary one)
OTA keys used for remote management were also different (no key sharing)

eSIM spoofing / cloning test

On Jul 04, 2025 we conducted a test in the Orange network of which aim was to verify the impact of a spoofed / cloned eSIM profile.

For the purpose of the test, we installed an identical (cloned) Orange eSIM profile (downloaded in plaintext with the use of compromised Kigen eUICC identity) on two different physical eUICC cards. These two cards (associated with the same mobile network subscriber) have been then inserted into two different Samsung phones (PHONE 1 and PHONE 2).

The players / test information:

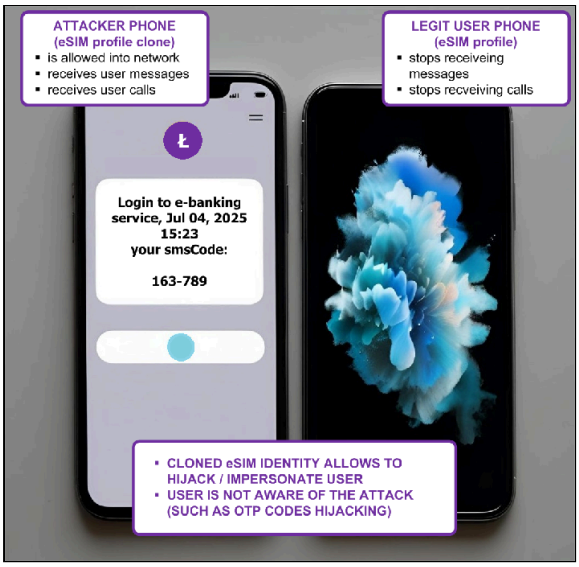
PHONE A (eSIM profile) - legit user
PHONE B (eSIM profile clone) - rogue / malicious user

The test proceeded in the following steps:

1) PHONE A was turned on, we confirmed that arbitrary phone calls / SMS messages could be established with it (legit user could use the phone as usual)

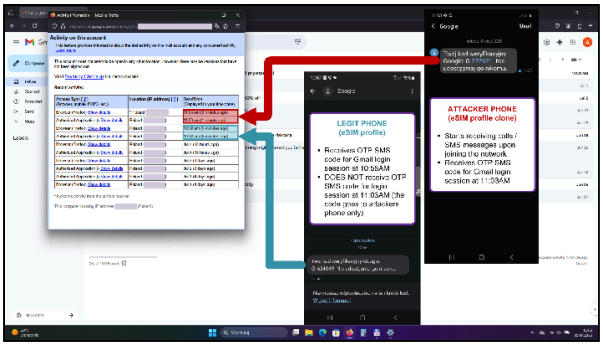
2) PHONE B was turned on by the malicious user, we confirmed that arbitrary phone calls / SMS messages started to be established with PHONE B, not PHONE A, we verified that PHONE A stopped receiving SMS messages / calls destinated to real subscriber number

3) PHONE B was turned off, PHONE A could start receiving calls / SMS messages upon network re-attach (such as through phone reset or plane-mode on / off procedure), PHONE A did not receive SMS messages (or their copies) received by malicious PHONE B in step 2 (at the time of hijacking) - the network assumed these messages have been already successfully delivered (they were).



The above test confirmed that eSIM cloning makes it possible to hijack identity of legitimate MNO subscribers. The hijacking can be conducted for arbitrary time period (attacker's choice). The user might not be aware* of the fact that its MNO identity has been hijacked and that it is not receiving calls / SMS messages (no visible trace at user end).

The possibility to hijack phone of a target MNO subscriber can potentially have serious implications. Multiple 3rd parties (i.e. Google Mail, e-banking sites) rely on SMS message for 2FA / OTP authentication. We verified that cloned eSIM can be abused for GMail.



According to some sources, the network should not allow access for duplicate SIMs (corresponding to same IMSI number) / SIMs already marked by the network as active. This does not seem to be the case for Orange network.

GSMA eSIM Security Group and ecosystem is well aware of SIM cloning threat (page 10 of this presentation). Yet, we think it is worth to bring the results of this real-life test to the public attention to emphasize broad importance of eSIM security / the other impact of compromising a single eUICC with the use of Java Card issues.

It's worth to note that a compromise of the eSIM chip embedded in the mobile phone may lead to eSIM cloning as malicious applet may access (steal by the means of memory reading / key decryption) content of all other eSIM profiles installed on the phone (eSIM chip). This makes eSIM cloning issue relevant.

(*) the copies of messages delivered to each mirrored eSIM are received by the main physical card though (vide trace leaving), the sole hijack possibility is key though

The ultimate (pending) goal

The actual goal of this research was to find remote vulnerability affecting eUICC for arbitrary mass-market mobile phone device (such as Samsung or Apple). One cannot exclude remote chip compromise - per GSMA standards, eUICC are obliged to implement remote management / remote SW update functionalities, which alone constitute a significant threat surface (beyond the threat surface for SIM / USIM cards).

We failed to gain funding for this research through our project designated for that purpose (*). We also failed to attract 3rd parties to the topic of SIM / USIM cards security analysis. Thus, somewhat incomplete (initial) research results / goals.

(*) in that context Microsoft could have had a real, potentially positive and wider impact, with proper funding we would also consider providing GSMA members with research results completely for free

Other eUICC vendors

Please, note that the hack should not be perceived in terms of a sole failure by Kigen company as it was primarily possible due to a ~5 years old flaw in Java Card VM (there has been some mitigations observed, it's not clear if these aimed to block the old issues though).

We believe there are systemic issues in the telecom ecosystem that make this (and can make future hacks like this) possible. Please, see our SIM / USIM cards project page for further information / insights on the topic.

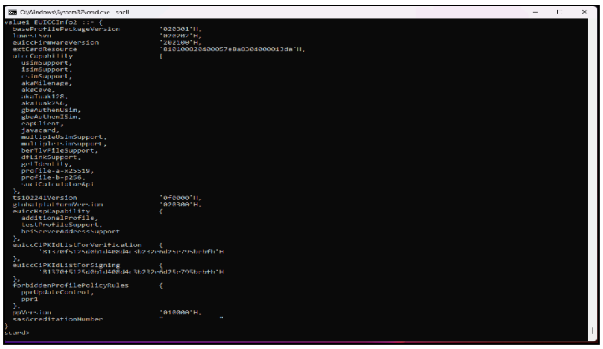
While there are other vendors of eUICC elements such as NXP and Thales in particular, we have limited means to verify if their products are secure / conform to vendors' claims (obstacles regarding access to card samples, no sale policy to security companies, access through NDA only, etc.).

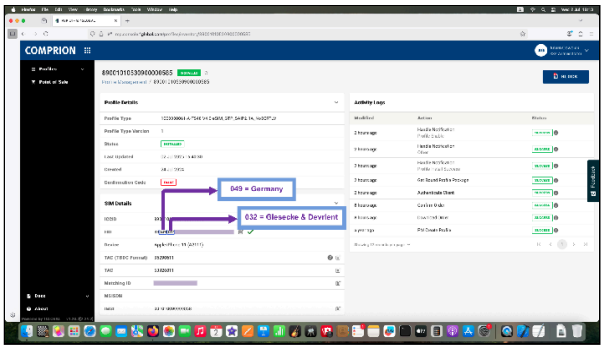
We have learned that no eSIM does bytecode verification (per eSIM protection profile described in SGP.25 and the Java Card protection profile in particular). This implicates an architectural weakness. This implicates that all eSIMs are likely vulnerable.

With respect to Java Card vulnerabilities from 2019, we suggest to inquire respective eUICC vendor directly for the fixing / impact status.

G&D eUICC

On Jul 02, 2025 we have used Comprion eSIM Test Profile Service(*) to install TS.48 test profile on a consumer GSMA eUICC from Giesecke Devrient:





We then used our toolkit and profile's known OTA key sets to install the malicious (test) applet into the chip over simulated OTA SMSPP protocol.

Initial analysis indicates the following:

- while the chip allows for arbitrary casts, it implements some form of a memory safety similar to the idea from our report
- the size field of the array object does not overlap with Object instance data

This makes the attack / bytecode sequence used for Kigen eUICC not directly applicable to tested G&D eUICC. This also goes along GSMA answer to question 1 ("some eUICC have additional proprietary mechanisms within their JVM implementations that, while not a full on-card bytecode verification, would stop some attacks and as a minimum require further, highly complex work to be successfully attacked").

BSC command summary for G&D chip can be seen here.

It's worth to mention that target GD eUICC contained a leaf certificate valid for 7984 years.

(*) for the purpose of this research, we subscribed to Comprion service a year ago (Jul 2024), but have not used it so far (on purpose)

Comprion statement

On Jul 09, 2025, Comprion company provided a statement in relation to the use of its eSIM Test Profile Service for the purpose of this project (testing of arbitrary eUICC cards for the presence of Java Card security issues).

Summary of security mechanisms / features of a target card

The following lists security mechanisms / features that are either implemented or has been encountered by us during the research (and hacking) of a target Kigen eUICC card:

- EAL4/5 certification
- side-channel / fault injection attacks countermeasures
- Java Card Runtime security mechanisms / type-safe language features
- additional mitigation against arbitrary Java casts

indirect Java pointers (pointers translation)
integrity / Java pointers validation checks
applet firewall (application isolation)
eSIM profiles isolation
stack isolation (separate java / native stack)
non-executable memory (stack)
tamper-proof silicon / secure processor (chip specific HW key for secrets storage)
time of check/time of use (TOCTOU) checks for security sensitive runtime / config structures

The attack demonstrated through this research showed these did not matter.

Per information received from Kigen, target card did not implement Infineon's Integrity Guard (security feature specific to unsupported chip variant).

Research complexity

This work started in Jul 2024 and initially took us ~8 months of work*. It required significant reverse engineering and R&D efforts (see details for information regarding size of developed code).

Each of our projects contains notes directory (`_re`). The notes directory for eSIM security research contains ~4200 files (the files we created to note down / record any important or relevant information spotted across the analysis / reverse engineering process). For a comparison, the `_re` directory for our previous project contains ~6800 files.

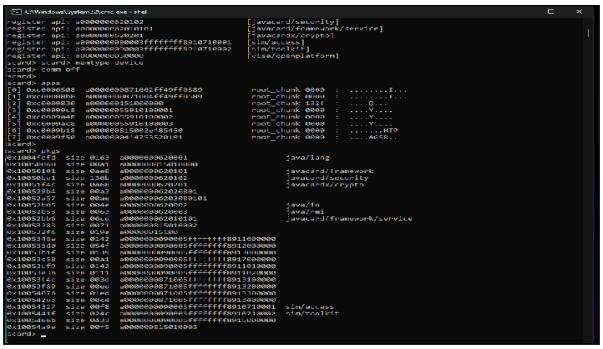
The above should provide a better picture on the overall work complexity / R&D effort conducted at our end while investigating security of target products.

(*) we also spent 2 months of the non-disclosure period for various toolkit developments, additional Kigen reversing, shared keys investigation, reliable / automatic exploit development, spoofing test

Backdoor feasibility / further analysis

During reverse-engineering process of Kigen eUICC card, we have gained significant know-how pertaining to its internals such as the following:

low-level system structures used for storing Java application and package data
low-level memory representation for Java Card VM objects
security domains, keys, associated memory chunks
core crypto subroutines
certificate parsing
JavaCard VM bytecode handlers
JavaCard VM native calls handlers
APDU command dispatchers (including semi-privileged instructions)
arbitrary checksums
persistent (flash) memory writing



The above should constitute a solid base for a more in-depth security analysis (vide the number / nature of initial issues affecting JavaCard component only) or prototyping (verifying feasibility) of an eSIM backdoor in the environment of Kigen eUICC card.

On technology vendor's responsibility

Kigen took full responsibility for both the presence and resolution of the JavaCard vulnerabilities that have been identified in its eUICC OS. The company informed that it does not use Oracle's JavaCard Reference Implementation in its commercial eUICC products. Its JavaCard Virtual Machine and Operating System were both developed entirely in-house from scratch to meet its performance, size, and security objectives.

Kigen acknowledged that similarities - conceptual or structural - may have been inadvertently introduced with respect to Oracle RI (Reference Implementation), which explained some overlap with the findings we made public in 2019.

Kigen indicated that its team was not made aware of those findings at the time. It's worth to note that according to our communication archive we have reached out to Kigen (web form) in Nov 2020 as a follow up of a Java Card Forum webinar conducted by the company with Orange.

Regardless of whether a licensee uses Oracle RI directly or builds a new VM from it, reference implementation should have security issues addressed, so that arbitrary parties / licensees such as Kigen do not replicate / introduce these in their own implementation.

Oracle is the primary source for all the information pertaining to what is needed to build a secure Java VM, what pitfalls / attacks to avoid. It is also the technology owner and licensor. In that context, it should be Oracle responsibility to provide state of the art, secure RI or at least notify licensees of any security problems affecting it.

Java security can be complex and full of pitfalls. Building a secure Java VM can constitute a significant challenge for Java vendor itself. Leaving this job to licensees is like asking for trouble.

On the value of independent security research

Java Card vulnerabilities disclosed in 2019 by our company (and reported to the vendor) have been instrumental for the attack against Kigen GSMA eUICC card.

If these vulnerabilities hadn't been downplayed by the vendor and instead been addressed at Java Card VM level (in runtime), the attack might have not been possible.

In that context, the research highlights both the need for and value of independent security research along its impact to security of real-life systems. It also highlights the need for SW / HW vendors to listen to / take action and provide financial recognition for the work and contributions of external parties such as ours (vide 127 security issues reported to Oracle over the years and not rewarded financially in any way).

Some recommendations / take aways for MNOs and/or vendors

We recommend to take the following into account in the context of eSIM security:

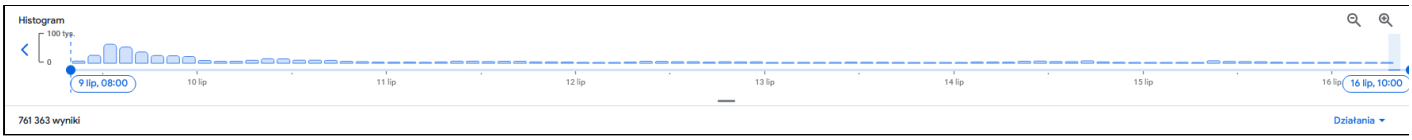
- never rely on any shared secret keys / credentials for eSIM security
- request and/or implement JavaCard hardening / runtime fixes from your JavaCard / eUICC vendor (memory safety at least)
- request and/or implement bytecode verification at RSP server (Java Card application scanning prior to bundle delivery)
- always assume your apps, their logic, associated secrets and/or some eSIM content could be revealed (one compromised eUICC identity can be used to download and peek into eSIM of any MNO)
- keep in mind that code extraction from an arbitrary eUICC may facilitate a more in-depth analysis (and new bug discoveries) along side channel attacks (known code sequence)
- do not take vendors' claims for granted, the telecom / mobile ecosystem in most cases relies on "security through obscurity"
- keep in mind that we have not tested eSIM chips used in mobile phones coming from major vendors, we haven't tried all of the ideas we have around the topic

One does need to keep in mind that eSIM insecurity primarily impacts end users. Arbitrary eSIM compromise may lead to compromise of secrets responsible for subscriber's privacy and security (security of voice / data communication in the mobile network).

Both eSIM and mobile phones' security architecture do not make it possible for users to inspect the content of the applications carried by eSIM profiles. The assumption is that these are safe / of a trusted orgin. This might not be the case. Same for the communication (SMSPP / CBS / BIP channels), which in most cases is done without any user consent. That architecture alone makes eSIM a perfect attack target and backdoor location for nation states / cybercrime groups.

The noise / ripples

We've recorded a stunning 75K visits (fetches) to this web page alone within the first week following project publication!



We are extremely excited that eSIM security triggered such a huge interest. This creates hope that the topic around eSIM / UICC / telecom security gets out of the *niche* and is explored further by security community, industry and academia.

Thank you to all media outlets and individuals that helped us spread the news about eSIM security related risks.

Details

The following technical materials are associated with respect to the security analysis conducted for eSIM technology:

Materials

Technical reports (as shared with Kigen company) describing security issues discovered in Kigen eUICC GSMA certified card along possible mitigations (Java Card VM / runtime hardening ideas):

- Card compromise and eUICC cert theft
- Broken VM type safety checks
- Idea for possible mitigation (memory safety)
- MNO / RSP server list used during testing
- Exploit reliability / automation, some mitigation and shared key
- Mirror test card info

Software and tools:

- eUICC / Java Card introspector (~1700KB Java source code)
- Generic applet for Java Card security testing (~150KB Java source code)
- Kigen eUICC card exploit module illustrating automatic compromise and secrets extraction (~72KB Java source code)