Attacks against GSMA's M2M Remote Provisioning

Maxime Meyer[1], Elizabeth A. Quaglia[2], and Ben Smyth[3]

[1] Vade Secure Technology Inc., Paris, France
[2] Information Security Group - Royal Holloway, University of London, UK
[3] Interdisciplinary Centre for Security, Reliability and Trust, University of Luxembourg, Luxembourg

Abstract. GSMA is developing and standardizing specifications for embedded SIM cards with remote provisioning, called eUICCs, which are expected to revolutionise the cellular network subscription model. We study GSMA's "Remote Provisioning Architecture for Embedded UICC" specification, which focuses on M2M devices, and we analyze the security of remote provisioning. Our analysis reveals weaknesses that would result in eUICCs being vulnerable to attacks: we demonstrate how a network adversary can exhaust an eUICC's memory, and we identify three classes of attacks by malicious insiders that prevent network operators from providing service. These attacks arise due to weaknesses in the specification. We disclosed our findings to GSMA; they confirmed the validity of these attacks and acknowledged their potential to disrupt the cellular industry and its customers. We propose fixes, which GSMA are incorporating into their specification. Thus, we have improved security of next generation telecommunication networks.

Keywords: eUICC, cellular networks, security, standardization, telecommunications.

1 Introduction

Machine to Machine (M2M) devices (i.e., machines communicating together without human intervention) are ubiquitous. Some of these devices communicate using cellular networks [14]. To access a 4G cellular network, an M2M device authenticates using an embedded MFF2 SIM [5], which is issued by a Mobile Network Operator (MNO). This authentication is achieved using the Authenticated Key Agreement (AKA) protocol [3], which defines seven cryptographic algorithms, a unique identifier for the subscriber's device, and a symmetric key shared between the MNO and the device. These algorithms and the symmetric key are embedded in MFF2 SIMs. The physical security of SIMs ensures the confidentiality and integrity of symmetric keys. Moreover, confidentiality and integrity of algorithms is also ensured, which is important for proprietary implementations of algorithms. MFF2 SIMs are limited as follows: they are neither re-programmable (i.e., they are tied to a *single* subscription during their lifetime)

nor remotely personalizable (i.e., MNOs have to *physically* upload the symmetric key onto MFF2 SIMs).

ETSI proposed a specification for embedded SIMs that are both re-programmable and remotely personalizable, thereby overcoming the limitations of MFF2 SIMs [6]. Following ETSI's proposal, industrial researchers, e.g., [2, 7, 17], and academic researchers, e.g., [20], presented proposals for secure remote provisioning schemes. Building upon ETSI's specification on secure remote provisioning, GSMA[1] released a specification for a next generation SIM, namely, an *embedded UICC*. This next generation SIM can support multiple MNOs simultaneously. Data from each MNO is stored in a *Profile*. Profiles are remotely provisioned, and installed into eUICCs. The mechanisms and protocols for remote provisioning and managing profiles on eUICCs for M2M devices are described by GSMA's *"Remote Provisioning Architecture for Embedded UICC"* specification [12].

Remote provisioning is a core aspect of GSMA's specification. To realize remote provisioning, this specification introduces a Subscription Manager (SM), which acts as an intermediary between MNOs and the eUICC. (This evolution is motivated by business cases [10, Section 3]. The SM oversees the remote management of the eUICC and of the profiles installed on this eUICC. The SM is further separated into two roles: Secure Routing (SM-SR) and Data Preparation (SM-DP), as depicted in Figure 1.

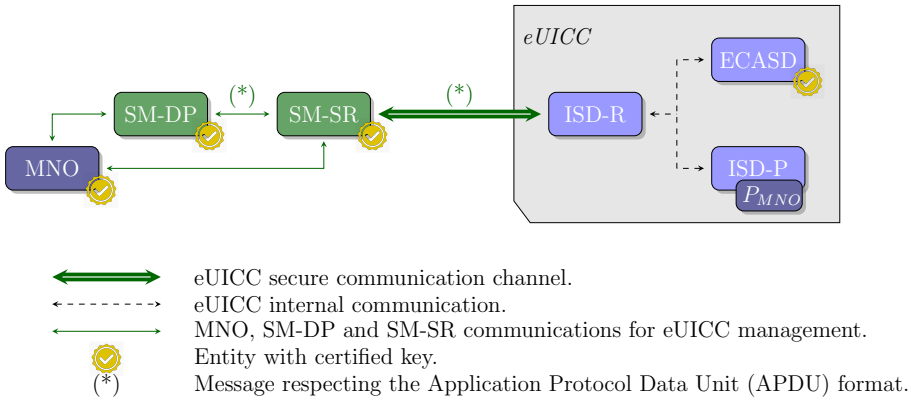

Fig. 1: eUICC remote provisioning interfaces and communication channels, adapted from [18]

Unlike MFF2 SIMs, which are tied to a single subscription, eUICCs support several subscriptions simultaneously. This is made possible by eUICC architecture, which is based on GlobalPlatform's smart card standard [8] and compatible

[1] Groupe Spéciale Mobile Association (GSMA) is a consortium of stakeholders in the telecommunication industry representing operators' interests worldwide.

with the MFF2 SIM file system.[2] GlobalPlatform's standard specifies the architecture of smart card platforms and their management. Internal communication is defined by GlobalPlatform's OPEN framework, which uses the Application Protocol Data Unit (APDU) message format. The eUICC architecture makes use of privileged applications, called *Security Domains*. These domains have access to keys and can setup secure channels with external entities. GSMA's specification relies on these security domains to isolate profiles (denoted P_{MNO} in Figure 1) into dedicated applications, each residing in a separated container (denoted ISD-P in Figure 1).

To secure external communication, eUICCs contain a controlling application (ECASD) which handles authentication of remote entities based on their certificates. Furthermore, remote messages pass through a secure channel between the SM-SR and its representative security domain (denoted ISD-R in Figure 1) on the eUICC. After processing by the ISD-R, messages are handled by the OPEN framework and relayed to their destined profile or application, characterized by a unique number, the AID (application Identifier).

GSMA is promoting their specification for standardization [24, 11]. Any weakness or flaw in the specification or subsequent standard could have disastrous consequences on the secure deployment of eUICCs. As such, security of remote provisioning must be analyzed. Indeed, finding and fixing specification flaws is paramount, because the cost of fixing problems increases exponentially once production commences.

Contribution. We study version 3.1 of GSMA's M2M remote provisioning specification and we present the first security analysis of remote provisioning. Our analysis reveals flaws which would make eUICCs susceptible to attacks and we recommend fixes to avoid such attacks. We disclosed our results to GSMA's Embedded SIM Working Group. They have acknowledged the validity of the attacks and they have already revised the latest release of the specification [13]. Thus, this research improves security of next generation telecommuncation networks.

Structure. Section 2 describes the mechanisms behind the creation of a profile and its corresponding ISD-P. It also describes how profiles are remotely uploaded through a secure channel onto the destined eUICC. Section 3 presents a memory exhaustion attack against eUICCs. The attack works by dropping an acknowledgment message sent during ISD-P creation. Such an attack leads to the creation of an empty and undeletable ISD-P. The attack can be repeated to exhaust an eUICC's memory. Section 4 introduces an attack that can be launched by a malicious SM-SR to prevent operators from installing new profiles on eUICCs, and Section 5 presents a similar attack that can be launched by a malicious MNO (or a malicious SM-DP). These attacks work by modifying remote management messages. Section 6 shows how a malicious operator can abuse an eUICC profile policy rules to block all other operators. Section 7 details our reporting of those attacks to GSMA and their reaction to it. Finally, Section 8 presents a brief conclusion.

[2] GlobalPlatform is a consortium that maintains and promotes smart cards standards.

2 Preliminaries

2.1 Profile Download and Installation

GSMA's eUICC specification defines a remote provisioning procedure, called `Download&Install`, to transmit profiles from an MNO to an eUICC, and install these profiles onto the eUICC [18]. Communication between an eUICC and an external entity uses a secure channel between the SM-SR and its interface on the eUICC, the ISD-R. The procedure can be summarized as follows (see Figure 2):

1. An MNO initiates the process by making a `DownloadProfile` request to an SM-DP containing a profile description (e.g., profile size, type, network capabilities). The SM-DP is responsible for creating the profile according to the description.
2. The SM-DP uses the `GetEIS` function to obtain data about the destined eUICC from the SM-SR. This data contains mutable and immutable information about the eUICC.
3. The SM-DP makes a `CreateISDP` request to the SM-SR. After processing the request (labelled (3a) in Figure 2), the SM-SR creates an ISD-P on the eUICC (labelled (3a)). This request creates a container on the eUICC, called ISD-P, that will hold the profile. (We will describe procedure `CreateISDP` in Section 2.2.)
4. The SM-DP establishes a secure channel with the ISD-P (labelled (4a)), and sends the profile to the ISD-P over that secure channel (labelled (4b)).
5. The ISD-P installs the profile and sends an acknowledgment message to the SM-DP, which relays it to the MNO.

2.2 ISD-P creation

ISD-P creation (Step 3 of the `Download&Install` procedure described in Section 2.1) precedes the upload of the profile onto the eUICC. During this phase, memory is assigned, and the profile's unique Application Identifier (AID), are set. The creation proceeds as follows (see Figure 3):

1. The SM-DP initiates the procedure by sending a `CreateISDP` request to the SM-SR.
2. The SM-SR receives the request and establishes a secure channel with the eUICC's ISD-R.
3. The SM-SR instructs the ISD-R to create an ISD-P, specifying its creation parameters including the application identifier of the ISD-P (`ISD-P AID`) in an APDU command message.
4. The APDU is past from the ISD-R to by the smart card framework (labelled (4a) in Figure 3) which creates the ISD-P (labelled (4b)) and returns an acknowledgment message to the ISD-R (labelled (4c)).

5. The ISD-R sends a Response APDU reporting the success of the ISD-P creation to the SM-SR.
6. The SM-SR updates the eUICC Information Set (EIS) file[3].
7. Finally, the SM-SR returns the `ISD-P AID` to the SM-DP.

3 Memory exhaustion attack by network adversary

We analyzed the security of GSMA's remote provisioning protocol [12] by considering potential adversaries and their motivations. In this section, we consider a network adversary, i.e., an adversary that is able to read, modify and delete messages sent over the network, and also inject messages into the network.

3.1 Message timeout

Error handling in step 3 of the `Download&Install` procedure (see Section 2.1) is limited to timeouts, i.e., response messages sent back when the response message from the eUICC is not received by the SM-SR during a specific time frame. Hence, in the event of a lost response message from the ISD-R (labelled (7) in Figure 3), the flow of the protocol can be represented as follows (see Figure 4):

- Following the ISD-P creation on the eUICC, the return APDU containing the `ISD-P AID` is lost (1).
- The SM-SR awaits for the response APDU during T seconds (fixed during implementation), then sends a timeout response message to the SM-DP (2).[4]

3.2 Memory exhaustion attack

It is possible to launch an attack that fills part of an eUICC's memory with an empty ISD-P. Moreover, the eUICC's memory could be exhausted by repeating the attack. Indeed, a malicious adversary could drop the ISD-R's response to the SM-SR (see (7) in Figure 3) as is common in denial of service attacks [27, 28]. As a result, the SM-SR does not receive the return message from the eUICC and cannot update the `EIS` file. This is because the status of the ISD-P creation on the eUICC is unknown by the SM-SR. The ISD-P created remains on the card, without any association to an SM-DP nor MNO, thus the ISD-P is *orphaned* and memory space on the eUICC has been reserved for it.

[3] Characteristic data and information concerning an eUICC are stored in the EIS (eUICC Information Set) file, which is stored by the SM-SR responsible for managing the eUICC.

[4] The process ends here, however, as an operator must possess a profile on an eUICC to provide network services to the device containing this eUICC, the process could start again. In this case, when the SM-SR tries to create an ISD-P with the same `ISD-P AID`, the smart card framework would issue an error message since an application on the eUICC with the same AID (Application Identifier) already exists.

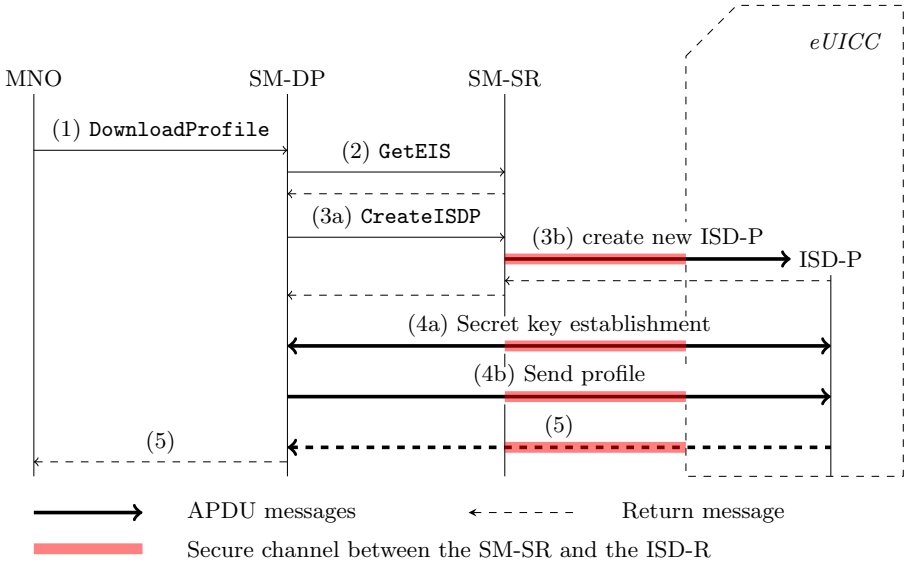

Fig. 2: Profile download and installation flow

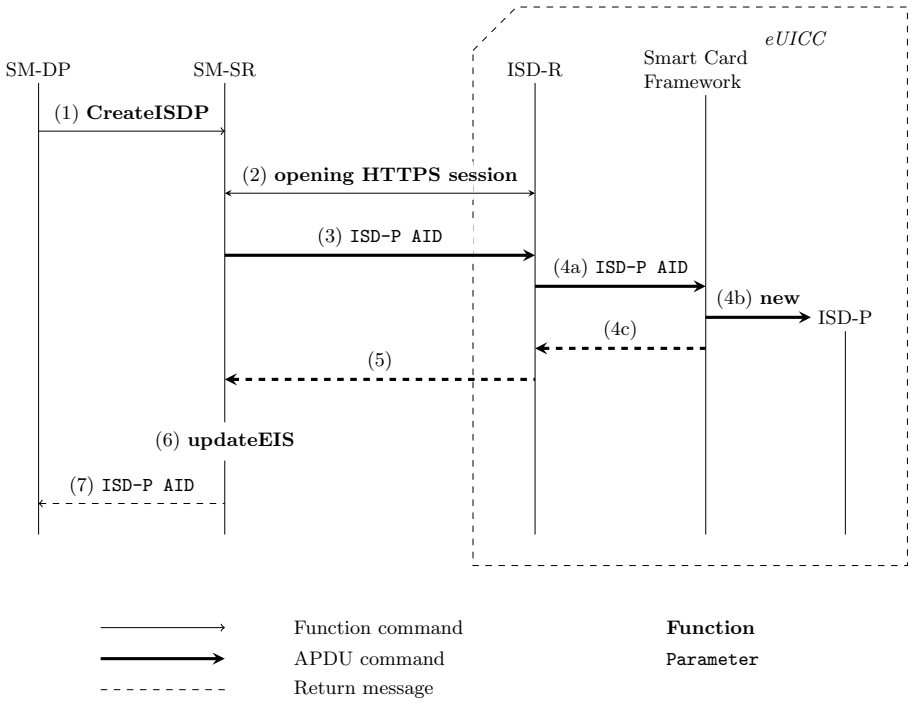

Fig. 3: ISD-P creation flow

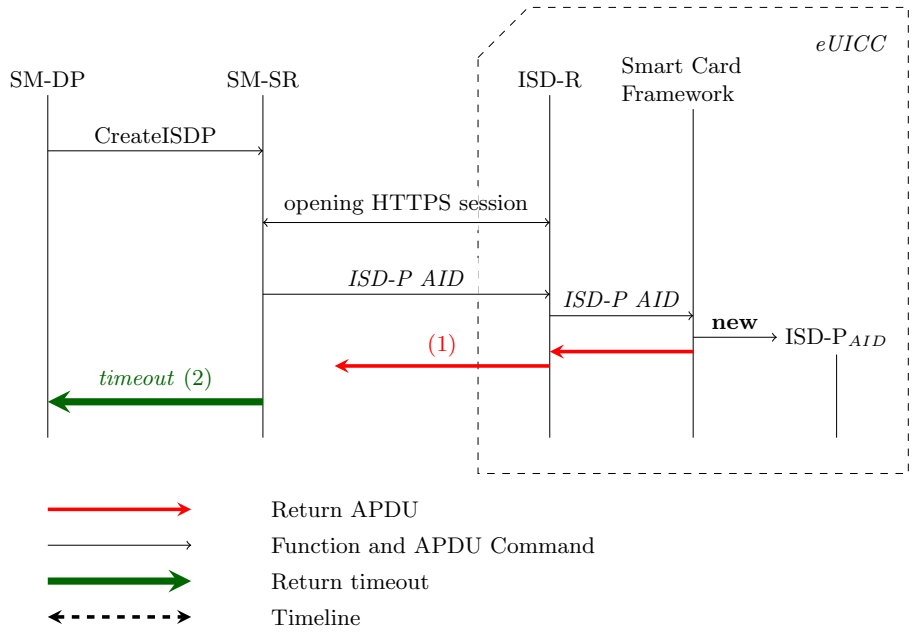

Fig. 4: GSMA timeout countermeasure and resulting creation of an orphan ISD-P

It is not possible to recover from this attack (e.g., by deleting the created ISD-P), since the specification restricts profile deletion to MNOs or SM-DPs. And, to delete a profile, an MNO or an SM-DP has to send a `DeleteProfile` command containing the `ISD-P AID` of the concerned profile to the SM-SR. However, as depicted by (9) in Figure 3, the SM-DP receives the `ISD-P AID` from the return message of the `CreateISDP` command. As this return message is dropped by the adversary, the `ISD-P AID` is lost, thus, neither the MNO nor the SM-DP can delete an orphaned profile. A procedure called *Master Delete* is defined in the specification to allow an SM-SR to delete profiles not maintained by operators anymore [12, Section 3.10]. However, this mechanism only works for fully installed profiles, which is not the case here. Consequently, the master delete procedure cannot be applied. In fact, there is no mechanism defined in the specifications for the SM-SR or the eUICC to delete orphaned ISD-Ps. If the attack is repeated to exhaust the eUICC's memory, only profiles existing on the eUICC before the attack can be used later.

This attack would cause financial loss, because it prevents operators from providing network service. Moreover, it is not possible to recover from it and any trace of the attack is minimal (only a lost message), making the attack even valuable. For example, an operator, with a profile on an eUICC, could collude with a network adversary and deliberately fill the eUICC memory. The attack is similar to a permanent denial of service, a class of attack already targeting connected devices [4].

3.3 Countermeasure

The attack can be prevented by creating a mechanism on the card to manage ISD-P creation. Once an ISD-P is created, this mechanism awaits the next logical instruction of the `DownloadProfile` process. If the awaited instruction is not received on time, the ISD-P is automatically deleted by this mechanism and a notification is sent to the SM-SR. As such, even if the notification is dropped too, the orphaned ISD-P is deleted. This mechanism could be implemented as an extension of the GlobalPlatform framework.

4 Undersizing memory attack by SM-SR

The remote provisioning framework for eUICCs introduces a new ecosystem involving a new entity, namely, a subscription manager. The subscription manager is divided into two roles: an SM-SR and an SM-DP. The subscription manager division was motivated by a desire to have one entity managing eUICCs, and another one creating standardized profiles. GSMA also defined new business models based on the SM-SR and SM-DP [10], describing relations between entities and business mechanisms for remotely provisioning SIMs with profiles.

In this section, we present an attack that can be performed by a malicious SM-SR, behaving as a *malicious insider* [23, 15]. Such an adversary is dishonest, perhaps due to greed. Malicious entities have the capabilities of their honest counterparts, plus they will try to modify the protocol or the messages sent without being suspected of being dishonest, resulting in what is considered as low cost attacks. See Anderson et al. [1] for further information on low cost attacks. In our case, we are considering a valid network entity owning a valid certificate. Staying undetected is important for this malicious entity, as it is prone to sanctions if it is detected [9, 25].

4.1 Preparation for ISD-P's creation

To create a suitable profile, the SM-DP must check that the parameters requested by the MNO are compatible with the destined eUICC. In order to perform this verification, the `Download&Install` procedure proceeds as follows:

- The SM-DP first sends a `GetEIS` request to the SM-SR (see step 2 of Section 2.1).
- The SM-SR returns an APDU response message containing data from the EIS file.
- The SM-DP uses this information to check the validity of the eUICC and to learn the value of the eUICC's mutable technical characteristics.[5]

[5] Several characteristics of eUICCs which are instantiated and set at manufacture time, are immutable (i.e., constant throughout the eUICC life cycle) and signed by the manufacturer. These characteristics, as well as other mutable information about the eUICC, are issued by the manufacturer to the first SM-SR responsible for the eUICC as the EIS file.

– The SM-DP constructs a `CreateISDP` request containing the following input data: the EID of the destined eUICC, the identification of the profile to be downloaded and installed, the identity of the MNO requesting the profile creation and the *required memory* to be allocated to the ISD-P created.

4.2 Undersizing memory attack

It is possible for the SM-SR responsible for an eUICC to launch an attack that prevents operators and SM-DPs from installing profiles on that eUICC. After receiving a `GetEIS` request from the SM-SD, the SM-SR returns the EIS file, after setting the value of field `remainingMemory` to 0. (This cannot be detected because the field isn't signed.) By doing so, the SM-SR prevents an SM-DP from creating an ISD-P required for uploading a new profile on the eUICC, because the process of downloading and installing a profile onto an eUICC would halt, as the eUICC would be considered by the SM-DP as unable to receive a new profile. Therefore, an SM-SR can deny operators from installing profiles on an eUICC.

This attack is feasible due to mutable fields from the EIS file that are not signed by the EUM nor by the eUICC. This is especially the case of fields whose values change each time a profile is added or deleted from the eUICC, in particular the field `remainingMemory`. An SM-SR performing this attack could be suspected of being malicious by the SM-DP or the operator if the device is quite new as its eUICC's memory should be almost empty. However, in the case of a change of subscription for an old device, that might have several profiles installed on it, the SM-SR will likely probably not be detected. To increase its chances of staying inconspicuous, the malicious SM-SR could set the value of the remaining memory to a number between 0 and the minimal size of a profile.

4.3 Countermeasure

This attack can be prevented by protecting the eUICC's mutable characteristics. To achieve this, the values sent by the eUICC to the SM-SR during an `AuditEIS` could be signed using the eUICC's private key.[6] Hence, the SM-DP can be assured of the value's integrity. The signature should also contain a timestamp to prevent replay attacks by the SM-SR.

5 Inflated profile attack by network operators

5.1 Inflated profile attack

When making a `DownloadProfile` request (see Section 2.1), a malicious operator could write a profile description such that the corresponding profile, when created

[6] The function `AuditEIS` is used here instead of `getEIS` as the former function is between the SM-SR and the eUICC directly, while the latter one originates from an SM-DP and returns the EIS file stored in the SM-SR.

by the SM-DP, exhausts the remaining memory. (The operator can learn how much memory is available from using the getEIS function.) By doing so, no other operator can store a profile on the eUICC. The SM-SR might suspect that an operator has performed an attack, but the operator could always deny it. Moreover, to increase success, the MNO could define a profile such that some memory would be left free on the eUICC, but not enough for a new profile to be installed. This attack can similarly be initiated by an SM-DP during the `createISDP` request.

5.2 Countermeasure

This attack can be avoided if an upper bound on profile size is defined. When a request for a new profile is made by an operator, the SM-DP and the SM-SR would check the size of the profile to be created with this maximal size.[7]

6 Locking profile attacks by network operators

6.1 Profile Policy Rules and eUICC lock

GSMA's specification defines a set of policy rules for managing the life cycle of profiles. These policy rules are stored in a file (POL1) inside each profile. They specify whether a profile can be disabled, can be deleted, or should be deleted once it is disabled. A profile's policy rules are initialized by the MNO during profile creation and can only be modified when the profile is enabled. An unsynchronized copy of these rules is maintained in the EIS file stored by the SM-SR. Policy rules can only be changed by the MNO owning the profile, that MNO is also responsible for updating EIS accordingly.

The policy rule `CannotBeDisabled` locks an eUICC to a profile. This lock forces devices to connect to a specific network, it is a feature of the existing subscription model[26]. It is typically used in the context of subsidizing subscriptions. The eUICC lock feature can be set multiple times, while, for 4G networks, once a device is unlocked, it is not possible to lock it again.

At a high level, policy rule `CannotBeDisabled` is either *true* or *false* for the enabled profile. We show that this rule introduces a weakness that can result in an eUICC being locked to an undesirable operator's profile, without regard for the initial value of the rule. We demonstrate that a malicious MNO can launch an attack when rule `CannotBeDisabled` is false (§6.2.1) and that an opportunistic MNO can take advantage of its position when the rule is true (§6.2.2).

6.2 Locking profile attacks

[7] The SM-SR should perform the check to prevent the SM-DP launching a similar attack

6.2.1 Rule `CannotBeDisabled` is *false* Suppose all of an eUICC's profiles have set the policy rule `CannotBeDisabled` to *false*. Further suppose a malicious MNO is interested in blocking other operators' profiles. For this, the malicious operator installs its profile (see Section 2.1), enables it and sets policy rule `CannotBeDisabled` to *true*. The MNO owning the previously enabled profile will receive a notification from the SM-SR indicating that its profile has been disabled, but, the operator cannot re-enable its profile as the new one cannot be disabled. The eUICC is locked to the malicious operator's profile. Moreover, this new profile cannot be disabled and, consequently, cannot be deleted. The following examples present scenarios whereby eUICCs might be locked:

Cyberwarfare. In the case of a conflict arising between countries, one country could use a national operator to remotely attack and block the other country's eUICCs. Such attacks are termed *state sponsored hacking* by Rounds and Pendgraft [21]. (See Schneier [22] for further discussion on cyberwar.)

Hackers. Hackers might obtain a certificate corresponding to an operator. Such attacks have been observed in other domains, e.g., [19, 16]. Thus, it is feasible that hackers might block all devices without an enabled profile.

Supply chain attack. Assuming devices are powered-on once manufactured, and then shipped to their destination, and further assuming that devices are passing along the border of a country where operators have an aggressive market strategy, one operator could install a profile on all devices inside the container. Such attacks could also occur while devices are in production or in storage.

6.2.2 Rule `CannotBeDisabled` is *true* An issue might arise when a subscriber wants the operator to unlock M2M devices. Indeed, M2M device owners are likely to initiate the remote unlocking of eUICCs. This setting, where the client ask the MNO to unlock devices, is problematic in the presence of an opportunistic MNO. The opportunistic MNO can delay the unlocking process, thus preventing other MNO's from enabling their profile on the locked eUICC. Furthermore, the locking profile cannot be deleted without the operator's approval.

6.3 Countermeasure

We present several countermeasures that can be combined, if desired. First, a mechanism to automatically unlock the eUICC, once a lock expires. Secondly, specifying an upper bound on the locking period (e.g., two years), to prevent abuse. Finally, permitting locking only once during the life of an eUICC. This can be achieved by using a counter set to a specific value once a lock is used on a profile.

7 GSMA response

We reported our findings to GSMA under their Coordinated Vulnerability Disclosure Programme. GSMA assigned the GSMA Embedded SIM Working Group (ESIMWI4 Group) to investigate and, based upon their findings, GSMA provided the following response:[8]

> our remote SIM provisioning experts have acknowledged the existence of the four attacks and confirmed the vulnerabilities have the ability to impact the mobile industry and its customers

Moreover, GSMA are working with us to incorporate our fixes into their specification. Indeed, GSMA has released an updated specification [13] which includes some of our fixes. Thus, we have improved security of next generation telecommunication networks.

8 Conclusion

GSMA are striving towards standardization of remotely provisioned, embedded SIMs. Their efforts have resulted in specifications for remote provisioning. In particular, they have specified remote provisioning for M2M devices. This evolution towards next generation telecommunications is exciting, but not without risk. Indeed, we have studied release 3.1 of GSMA's specification and discovered that the proposed evolution is insecure. More issues might well exist and it is crucial that the specification is studied further to ensure security of next generation telecommunications.

Acknowledgments

This work was largely performed at Huawei's Mathematical and Algorithmic Sciences Lab in France.

References

1. Anderson, R., Kuhn, M.: Low cost attacks on tamper resistant devices. In: International Workshop on Security Protocols. pp. 125–136. Springer (1997)
2. Berard, X., Gachon, D.: Method for remotely delivering a full subscription profile to a uicc over ip (December 2013), US Patent App. 13/991,846
3. Blom, R., Norrman, K., Naslund, M., Rommer, S., Sahlin, B.: Security in the Evolved Packet System. Tech. rep. (February 2010)
4. Cimpanu, C.: New malware intentionally bricks iot devices. `https://www.bleepingcomputer.com/news/security/new-malware-intentionally-bricks-iot-devices/` (April 2017), accessed: 2017-04-11

[8] Email communication, August 2017.

5. ETSI: Smart Cards; Machine to Machine UICC; Physical and logical characteristics (Release 9). Technical Specifications 102 671 (April 2010)
6. ETSI: Smart Cards; Embedded UICC; Requirements Specification (Release 12.0.0). Technical Specifications 103 383 (September 2013)
7. Girard, P., Proust, P.: Method for managing content on a secure element connected to an equipment (November 2013), US Patent App. 13/991,823
8. GlobalPlatform: Card Specification (Version 2.3). Technical specifications (October 2015)
9. Gow, D.: Telefónica hit by record €152m anti-trust fine. https://www.theguardian.com/business/2007/jul/04/media (July 2007), accessed: 2016-12-06
10. GSMA: Business Process for Remote SIM Provisioning in M2M. Technical Specifications 1.0 (February 2015)
11. GSMA: GSMA announces mobile industry initiative to create a global remote provisioning specification for consumer devices (March 2015)
12. GSMA: Remote Provisioning Architecture for Embedded UICC. Technical Specifications 3.1 (May 2016)
13. GSMA: Remote Provisioning Architecture for Embedded UICC. Technical Specifications 3.2 (June 2017)
14. GSMA Intelligence: Cellular M2M forecasts: unlocking growth. Tech. rep. (February 2015)
15. Jiang, S., Smith, S., Minami, K.: Securing web servers against insider attack. In: Computer Security Applications Conference, 2001. ACSAC 2001. Proceedings 17th Annual. pp. 265–276. IEEE (2001)
16. Langley, A.: Further improving digital certificate security. https://security.googleblog.com/2013/12/further-improving-digital-certificate.html (December 2013), accessed: 2017-01-16
17. Merrien, L., Berard, X., Gachon, D.: Method for transmitting a sim application of a first terminal to a second terminal (May 2014), US Patent App. 13/991,542
18. Meyer, M., Quaglia, E.A., Smyth, B.: Overview of GSMA Remote Provisioning Specification (2017), https://bensmyth.com/publications/2017-eUICC-overview/
19. Microsoft: Fraudulent digital certificates could allow spoofing. https://technet.microsoft.com/library/security/2607712 (August 2011), accessed: 2017-01-16
20. Park, J., Baek, K., Kang, C.: Secure profile provisioning architecture for embedded uicc. In: Availability, Reliability and Security (ARES), 2013 Eighth International Conference on. pp. 297–303. IEEE (2013)
21. Rounds, M., Pendgraft, N.: Diversity in network attacker motivation: A literature review. In: Computational Science and Engineering, 2009. CSE'09. International Conference on. vol. 3, pp. 319–323. IEEE (2009)
22. Schneier, B.: Cyberwar. https://www.schneier.com/blog/archives/2007/06/cyberwar.html (June 2007), accessed: 2016-10-12
23. Schultz, E.E.: A framework for understanding and predicting insider attacks. Computers & Security 21(6), 526–531 (2002)
24. Sierra Wireless: The eUICC opportunity: harness the power of IoT eSIMS. White paper (2017)
25. Thomas, D.: France hits Orange with €350m antitrust fine. https://www.ft.com/content/d20d7882-a4b7-11e5-a91e-162b86790c58 (December 2015), accessed: 2016-12-06

26. Vermeulen, J.: Why it is legal for FNB to SIM-lock its smartphones. https://mybroadband.co.za/news/smartphones/178714-why-it-is-legal-for-fnb-to-sim-lock-its-smartphones.html (September 2016), accessed: 2017-01-16
27. Wood, A.D., Stankovic, J.A.: Denial of service in sensor networks. computer 35(10), 54–62 (2002)
28. Xie, L., Zhu, S.: Message dropping attacks in overlay networks: Attack detection and attacker identification. ACM Transactions on Information and System Security (TISSEC) 11(3), 15 (2008)