HCE Synergies with Public Transport

CIPURSE™ and HCE open up new possibilities
for Transit Ticketing Systems

www.osptalliance.org

About OSPT Alliance

The OSPT Alliance is an international association chartered to provide a new open standard for secure transit fare collection solutions. It provides industry education, creates workgroup opportunities and catalyzes the development and adoption of innovative fare collection technologies, applications and services. The OSPT Alliance was founded by leading technology companies, and membership is open to technology providers, transit operators, consultants, solution vendors, government agencies and other stakeholders in the transit ecosystem. For additional information, please visit www.osptalliance.org.

Legal

OSPT Alliance
Prinzregenten str. 159
D-81677 Munich
Germany

Version	Date	Major changes since previous revision
1.0	***18.2.2016***	***Initial version***

Table of Contents

1. Executive summary

OSPT Alliance's open ticketing standard CIPURSE™ has been designed with security and ease of use in mind. It is continually evolving to meet market needs and can be implemented on a broad range of media ranging from single application tickets and cards to multi-application smart phones and hosted systems, including HCE. However, there are a number of approaches to implementing HCE. This paper explores the options and how they relate to CIPURSE.

Host Card Emulation (HCE) is a technology that enables software emulation of a traditional tamper resistant plastic smart card (such as those used in transit, banking, access, identity). However public transport can impose structural constraints that may impact the appropriate choice of HCE approach. Some legacy proprietary ticketing systems cannot support HCE at all. Nor do all mobile phones support HCE. In addition, today, online HCE cannot meet throughput requirements in most cases. Coverage is also an issue at present. HCE must also be able to meet security and business rules requirements.

HCE, however, can provide transport operators with multiple benefits, especially where throughput is not key. Major benefits are openness, security, investment protection, respect of standards, affordability and the prospect of leveraging a 'killer app' to create broader acceptance of digital services. It facilitates affordable interoperability between closed loop networks and with other application areas. Many of HCE's specific security concerns are already covered by measures put in place by transport operators dealing with proprietary security risks. In fact, implementing CIPURSE adds considerable value to any public transport HCE use case by facilitating multi-application options, enhancing openness and security and protecting investment through adherence to standards and multi-platform compatibility.

The paper reviews different types of HCE implementation, including tokenization, and their advantages and disadvantages – without SE, hybrid, SE in the cloud, with offline support and account based.

It suggests that at present asynchronous secure transaction systems as used in transit are best supported by secure element technologies such as CIPURSE or hybrid HCE leveraging CIPURSE. However, a HCE solution with offline support and without the presence of a secure element provides significant additional flexibility and reach.

Indeed many operators already have the most important components of such a solution in place. It is this real life combination of HCE, offline credentials, NFC mobile phones and transport network infrastructures built upon open standards that OSPT Alliance is working towards.

OSPT Alliance is also working on adding additional simplified authentication methods to the CIPURSE Specification that would further optimize its use with HCE. In addition OSPT Alliance will further develop CIPURSE to meet other HCE related developments, for example a move towards use of elliptic curve cryptography.

2. Scope

In 2014 HCE received a lot of attention, thanks to moves by conventional payment networks, banks and some of the largest internet players. Although public transport is not just about payment, it is useful to understand if and how HCE can be leveraged within public transport missions.

In order to do so, this white paper first acquaints the reader with some basic information about the CIPURSE™ open standard and how Host Card Emulation (HCE) works. Afterwards it considers the context in which HCE may be relevant, specifically:

- Public transport (PT) constraints that are structural to HCE;
- Specific public transport opportunities created by HCE;
- CIPURSE strengths any HCE implementation would require.

Once these key points are addressed, CIPURSE based HCE implementations and the associated security concepts are introduced and further described in the remainder of this document.

2.1 Audience

This document is intended for:

- OSPT Alliance members implementing CIPURSE based on HCE;
- Decision makers at public transit organizations;
- System integrators;
- Consultants.

3. Definitions

3.1 Terminology

PICC *Proximity Integrated Circuit Card*

HCE *Host Card Emulation*

SE *Secure Element*

PT *Public transport/transit*

TEE *Trusted Execution Environment*

OS *Operating System*

APDU *Application Protocol Data Unit*

3.2 References

[CIPURSE_OpInt]
OSPT™ Alliance: CIPURSE V2, Operation and Interface Specification
Revision 2.0 / 2013/12/20

[CIPURSE_T]
OSPT™ Alliance: CIPURSE V2, CIPURSE T Profile Specification

[HCE 101]
Smart Card Alliance – HCE 101 – MNFCC-14002
August 2014

[GOOGLE_HCE]
https://developer.android.com/guide/topics/connectivity/nfc/hce.html#SecureElement

[GOOGLE_HCE_CX]
https://developer.android.com/guide/topics/connectivity/nfc/hce.html#Coexistence

[GSMA_MC]
OpenID Connect Mobile Connect Profile 1.0,
February 2015

[MOBISECSERV]
Two-factor Authentication for Android Host Card Emulated Contactless Cards

4. Introduction to OSPT Alliance and CIPURSE

The aim of the OSPT Alliance is to help the transit community move towards the next generation of secure, cost-effective, and flexible fare collection solutions through a global, multi-provider community.

Its charter is to leverage the recently defined new open standard for secure transit fare collection solutions, while providing industry education and creating workgroup opportunities, and to be a catalyst for the development and adoption of innovative fare collection technologies, applications, and services.

The OSPT Alliance is also building a global ecosystem of transit operators, transit consultants and integrators, technology solution providers, and government agencies to stimulate development and delivery of next-generation fare collection solutions.

The CIPURSE™ open security standard addresses the needs of local and regional transit authorities to have future-proof fare collection systems with more advanced security than that of those currently in use. CIPURSE can be implemented on a broad range of media ranging from single application tickets and cards to multi-application smart phones and hosted systems, including those supporting HCE. CIPURSE supports a range of ticketing applications such as single journey, daily tickets, account based tickets and season tickets as well as loyalty applications, micro-payment, and other value added services. The mobile phone is able to combine all such traditional card applications into a single device serving multiple applications.

The OSPT Alliance HCE white paper presents CIPURSE, as used with a variety of different security approaches for HCE.

5. What is CIPURSE and what can CIPURSE be used for?

The CIPURSE™ open standard was designed to address the needs of local and regional transit authorities to have future-proof AFC systems with higher performance and more advanced system security than that of those currently in use. These systems are capable of enabling commuters to use a single fare medium—from simple, standalone tickets to multi-application cards, microSD cards and NFC mobile phones—seamlessly across several modes of transport in different locations, even across different regions and systems.

This platform agnostic approach to realize fare media using above mentioned devices and technologies brings new value to the ecosystem and enhances customer experience. Through independent testing, the open standard provides optimized interoperability of fare media from multiple sources to enable simple, fast and cost-effective integration into public transport schemes or for value added services.

To enable CIPURSE Advanced Security, the standard builds upon existing, proven, open standards—the ISO 7816 smart card standard, as well as the 128-bit advanced encryption standard (AES-128) and the ISO/IEC 14443-4 protocol layer—to provide a platform for securing both new and legacy transit fare collection applications, and has the potential to be used within existing application frameworks around the world. At the same time, because it is an open standard, it promotes vendor neutrality, cross-vendor system interoperability, lower technology adoption risks, higher quality and improved market responsiveness, all of which result in lower operating costs and greater flexibility for transport system operators. Furthermore, the CIPURSE security architecture was designed for ease of use and with performance in mind.

The CIPURSE standard is continuously evolving to be in line with latest trends and technological developments. For example, a recent notable addition to the standard is the Multiple Proximity System Environment (PxSE). PxSE offers efficient application identification and selection in contactless access control environments. Within a multiple application ecosystem, PxSE will improve product performance and optimize times to enter a transit network, event or building.

The CIPURSE standard also addresses terminal side components to facilitate easy integration of the CIPURSE fare media into existing AFC systems. For example, the CIPURSE SAM (Secure Access Module) specification defines the feature set to be supported by SAM or terminal firmware. This allows the enhancement of any ISO 14443 compliant card reader to support the CIPURSE solution.

6. What is HCE?

Host Card Emulation (HCE) is a technology that enables software emulation of a traditional tamper resistant plastic smart card (such as those used in transit, banking, access, identity). By using HCE, NFC transactions no longer need to be performed directly between validator and a physically embedded tamper resistant chip, called the secure element (SE). The mobile device can now act as a virtual representation of a contactless smart card.

Nowadays, many NFC mobile devices already support NFC card emulation. In most cases, the card is emulated by an installed SE. But with the introduction of Android 4.4, an additional method of card emulation that does not involve a secure element, called host-based card emulation was made possible for a significant mobile market share. This allowed any Android application to emulate a card and talk directly to the NFC reader [GOOGLE_HCE]. In essence, HCE bypassed the need for the physical presence of the SE. However, bypassing the SE or implementing some proprietary, non-standard SEs can make systems very vulnerable and exposed to external malware threats. In order to avoid malware and provide complete data security it is necessary for some kind of secure element to be introduced alongside the HCE technology. Mobile devices usually implement the SE in form of SIM, microSD or embed HW directly in the device. The alternative approach consists of a pure software solution on the handset, in combination with the use of remote SE, or other forms of risk mitigation measure.

When taking into consideration that some forms of standardized SE are used with HCE, providing mobile ticketing alongside classical contactless smart cards without changes to the existing validation infrastructure is no longer as challenging as it used to be. It should be considered by various operators providing such services because of many advantages described below.

6.1 Public Transport constraints affecting HCE

Public transport systems are often more complex that one would imagine… for good reasons. What are the public transport constraints that are structural to a projected HCE implementation in public transport (PT)?

First and foremost, the ability to fully respect HCE related technical standards: if a public transport infrastructure is already locked into a technology that allows proprietary native commands[1], AIDs routing will simply not be an option; i.e. HCE is not an option, unless PT acceptance infrastructures are upgraded to support ISO/IEC 14443-4 compliant applications, such as CIPURSE, as well as legacy RFID technologies already in place.

Smart Card Alliance – HCE 101 – MNFCC-14002 August 2014 states: "5.3 Transit […MIFARE Classic[2] does not support the use of simple AIDs, which is a core component of an HCE implementation. MIFARE DESFire has usage modes that do support AIDs, but much of the infrastructure in place does not use an AID…][… instead using native commands;]".

Secondly, a fragmented compatible use base:

- Without operationally sound analog contactless communication, a user transport credential is useless. Every transport authority is faced with this and relies on specific analog contactless communication specifications (such as ISO/IEC 14443 for contactless smart cards and NFC for mobile phones) to procure compatible technology;
- Unfortunately, to date, NFC mobile phones were designed to work in a multitude of different environments, irrespective of the public transport-specific environment. As a result, only a small portion of existing NFC mobile phones (and NFC SIM, SD cards, etc.) pass the test (due to differences in antenna design, powering, and so on);
- Furthermore, some of the leading mobile manufacturers, Apple in particular, have so far retained full control of their NFC chip and thus prevent their mobile phones from being used with existing public transport applications;
- These significant user bases cannot be served by HCE in public transport. Nevertheless, the industry trend is clearly towards international standardization and openness. Today, the OSPT Alliance is already performing benchmark tests as part of its certification program reference implementation. OSPT Alliance is also able to certify CIPURSE for NFC mobile phones using international NFC analog contactless communication standards and intends to do so to assist public transport authorities as soon as these standards are universally accepted;
- Once leading public transport authorities start accepting compatible NFC Mobile phones, using the OSPT Alliance certification program, just like other CIPURSE form factors, the existing trend towards standardization and openness will be reinforced, and the lucky owner of the right NFC mobile phones will enjoy greater convenience.

Thirdly, transaction throughputs:

- In most mass transit implementations around the world, throughput during a usage use case (implicit use of a transport service through contactless user interactions at the gate or upon boarding a public vehicle) is a matter of safety and also often has an impact on revenue. In such a context, HCE would need to be 'always on' (which is not the case), exclusive (not competing with other running processes), and performing well under 500ms (which is not yet the case for 'online HCE', considering blue sky scenarios for 3G or 4G latencies of about 100ms +/- 40%, and a minimum of two round trips);
- In other words, currently, 'online HCE' is not compatible with the most common use cases of mass transit. On the other hand, using HCE where transaction throughput is not as critical is a real option. Furthermore, such investment should be worthwhile since 5G networks will most probably reduce latency dramatically and thus enable 'online HCE' across PT use cases. Finally, this white paper explores work-around alternatives to 'online HCE'.

Wireless Network Signal coverage is also a factor. When using online forms of HCE implementation, the PT network wireless network coverage should first be verified. Most major urban transport networks are investing in their infrastructures to provide online connectivity; however, in 2016 this is still not true for the vast majority of locations.

[1] Trend Micro – Hacking RFID Payment Cards Made Possible with Android App – November 24, 2014 – By Veo Zhang.

[2] http://www.smartcardalliance.org/wp-content/uploads/HCE-101-WP-FINAL-081114-clean.pdf

PT specific security issues are also relevant:

- Public transport requires both high value and limited use credentials; there is no one-size-fit-all security solution in PT. This often implies that only certain PT products may be supported by HCE;
- Secondary levels of protection are usually required for various reasons; one of the most common reasons is the need to mitigate the residual lack of trust between various business entities participating into a mutualized revenue collection and customer service. HCE is very well positioned to meet such a constraint;
- Interoperability between transport and non-transport is found in many PT networks; this is conventionally enabled using a common smart card purse application which HCE will need to cope with.

So are business rules:

- Travel rights take many different forms: entitlements, pre-specified fares, dynamically specified fares (Pay-as-you-go), and so on. HCE does not need to support all travel rights but it should target those that are the most relevant to a city or region;
- Many diverse use cases make up the functional scope of PT systems: from loading transport products, availing rights to travel upon boarding, complying with travel right inspections, fulfilling inquiries and complaints, and so on. HCE needs to support a meaningful scope.

6.2 HCE opportunities for PT

Leveraging an ever growing population of smart NFC phones as personal or third party terminals is a sizeable opportunity for PT; whether for adding PT credentials to an existing travel contract, or inquiring about one's travel journey plan, dedicated lanes, regional travel services, and so on. Whenever transaction throughput is not paramount, 'online HCE' is worth considering as an alternative to costly ticket vending machines, self-service kiosks and manned point-of-sales.

PT networks are mostly closed loop networks where the same business entity is both the issuer and acquirer. HCE specifically enables such a business entity to autonomously deploy a solution. Interoperability is of the essence for many international and regional travelers; HCE can be leveraged as an elegant way to provide such interoperability between separate closed loop transport networks, in particular when coupled with account-based transport credentials.

HCE solutions do not require massive investments and are often supported by cloud based service providers that have already made investments in IaaS, Saas, or even PaaS (Infrastructure, Software, and/or Platform as a Service). Public transport networks have the potential to bring such service provider value-add to a critical mass of public transport users. They form a unique win-win to better serve a broad user base.

Interoperability between PT and other ecosystems (especially payment, mobile, and retail) is not always part of a public transport's core mission. Furthermore, when such interoperability is enabled by prepaid electronic money purse solutions, they have had negative implications on non-transport infrastructures (such as requiring dedicated proprietary terminals and acquiring networks).

HCE can be leveraged to remediate such situations by linking side by side a general purpose account (supported by a mobile terminal) to a PT ticket or smart cards (compatible with existing legacy PT infrastructure).

Having had to deal with risk models of intermittently connected systems and compromised proprietary secure elements, PT authorities have often implemented secondary levels of protections, such as shadow account management, or electronic signing of transactions. These secondary levels of protections are precisely the capability needed for 'hybrid HCE'; one where host based services are delivered through HCE while offline based services are provisioned through tokens or pervasive CIPURSE secure elements.

7. Why run CIPURSE on HCE?

7.1 CIPURSE strengths for any HCE implementation

- **Contactless killer app** – A CIPURSE™ based HCE service offering can be positioned as an opportunity to leverage the only proven contactless killer app to date: public transport. Indeed, PT authorities are challenged by the imbalance between ever increasing infrastructure costs and the limited elasticity of transport tariffs. One way to channel new investments into public transport is to leverage the ticketing application as a killer app. Daily multiple usage of the ticketing application changes how a user considers other value added services (essentially payment, mobile, and retail services);
- **Openness** - Online CIPURSE HCE bypasses the need for the physical secure element. This makes contactless infrastructures (whether related to payment, ticketing, events, etc.) independent of third party providers, such as network carriers or device manufacturers. Consequently, newcomers and startups can easily participate and take part in the development of such solutions. They can enter markets that are otherwise not accessible;
- **Secure** – OSPT Alliance believes that HCE will propagate and promote new services through value added NFC solutions; however, neglecting security would simply stop such a momentum. In order to avoid malware and provide complete data security it is required that a careful approach be considered. OSPT Alliance is committed to focus on such an approach;
- **Investment Protection** - CIPURSE based HCE investments are considered safe in 2016 since CIPURSE is a pure software solution with no hardware lock-in, open to all participants, with a broad cross industry support; i.e. providing a unique level of investment protection;
- **Full respect of ISO standards** – We have seen that this is paramount to acceptance in public transport but this also applies to other ecosystems. Whenever proprietary means are used, whether in the form of non ISO commands or hardware lock-in, HCE promises will not come to fruition. OSPT Alliance is dedicated in respecting standards;
- **Free to be embedded** - In the payment world, HCE is often synonymous with tokenization. In an offline context, HCE with tokenized credentials often requires the presence of a secure element. CIPURSE licensing policy and non-discriminatory alliance statuses were designed to produce broad availability of such secure elements (including limited use form factors, smart cards, eSE, UICC, μSD, etc.)

In other words, PTs are best positioned to take advantage of HCE with CIPURSE. Online and hybrid HCE implementations can both leverage the pervasiveness of OSPT Alliance and the power of CIPURSE. Such a solution architecture is pivotal to the realization of PT missions through win-win collaborations outside of transport.

8. Implementations of CIPURSE on HCE

Since CIPURSE™ is an open specification built on top of widespread underlying open standards it is possible to implement applications that follow the CIPURSE protocols in an HCE environment.

Using HCE however creates a security problem due to a lack of a tamper proof execution environment for data processing requiring protection. In public transport implementations, several mitigations strategies are commonly used to protect data from being manipulated and can be leveraged.

The following sections first expose the security problem posed at the CIPURSE protocol level. Then, possible mitigation strategies are explored.

At a protocol level, the main challenge posed by HCE relates to the following characteristics of the CIPURSE protocol:

- Session initiation in CIPURSE depends on symmetric key pairs and this fixed pair is used to create session keys that subsequently are used to encrypt session APDUs;
- In an HCE environment there is no physically tamper proof product (device or feature to deter tampering, such as sealed IC chips) to store the symmetric key pair. This limitation of HCE consequently exposes the CIPURSE cryptographic protocol to attacks otherwise not possible with CIPURSE certified products.

At system level, several mitigation strategies, with varying security characteristics, can be used, including:

- Security by obfuscation (such as hiding sensitive data inside the software code, possibly fragmented);
- Use of a remote secure element (leaving the mobile application acting as a relay for encrypted APDUs);
- Introduction of a second factor to protect or generate the key materials;
- Use of limited transportation credentials (such as one-time use cryptograms provided by a cloud based tokenization infrastructure and defeating the risk of spying on and cloning emulated card data);
- Use of account based security measures where security risks are managed and mitigated by the cloud system (such as shadow account management and blocking/black-listing of cloned/tampered cards).

Some concrete examples will be given in this chapter. Depending on the strength of the security mechanism chosen and the non-functional requirements of the use-case at hand, fraud detection and risk management will play a complementary role.

Furthermore, OSPT Alliance is considering all of the above strategies and market needs and is working towards adding additional simplified authentication methods to the CIPURSE Specification that would address the mentioned problems and be especially relevant for HCE (account-based/account-linked) implementations. Why CIPURSE is the most suitable standard for HCE products is explained in chapter 9 where these additional authentication methods are further developed.

8.1 HCE without SE (Pure HCE)

HCE in its pure form does not use a SE of any kind. The mobile device routes commands from a NFC controller to its CPU where the mobile application is running. If software emulation of CIPURSE commands is implemented in the HCE application, then the same system level infrastructure as for the classic plastic smart card solutions can be used.

On the other hand, malware applications running on the device can intercept commands and keys used during the execution of encryption/decryption. Furthermore, the mobile device cannot protect access to the memory of the device. This means that credentials stored in the mobile device could be accessed and possibly distributed by a malware application.

Additional measures could be used (e.g. white box cryptography, biometric data) to obfuscate the implementation of the cryptography or key storage, but even the use of enhanced security measures cannot provide the level of security needed to protect the keys. Keys and sensitive data can be cloned or altered and used by an unauthorized entity. However, this approach could still be used in cases where the keys are changed often, the level of security needed is not so high and the handling of unauthorized intrusions is enforced in the back end systems (e.g. blacklisting of device or credential).

8.2 HCE with physically present SE (Hybrid HCE)

Pure HCE solutions can be upgraded to a higher level of security by storing keys and confidential data in a SE, which is physically present on the device. Only when the credentials or encrypted data are needed, are they retrieved from the SE. In such models, the SE is providing only key storage and cryptographic services to the HCE application. Malware cannot see deciphered data or keys used in the protocol. A SIM card, microSD or embedded SE can act as a tamper resistant secure storage.

This approach works as long as there is a way to access the SE element from the mobile application and the app is able to run the commands needed in the CIPURSE security protocol (authentication, encryption/decryption, key and data storage, etc.) The SE needs to tailor its services to a specific HCE application (to avoid supporting potentially hacked applications loaded on the same mobile phone – a problem that seems not yet resolved).

On the system level (validation side), integrators should be aware of the changes required to support such a solution. Another disadvantage of Hybrid HCE solutions is that the SE is owned by either the OEM or MNO, which requires all the participants to execute contractual agreement between each other. This complicates the business model even further.

8.3 HCE with remote SE (Online HCE)

Nowadays, implementations of SEs in form of a SIM, microSD or embedded HW on the device are not standardized, nor are they simple to integrate and deploy. Moreover, the business model complexity of the physical presence of the SE grows exponentially for global M-to-N value-added services which involve more than one service operator, more than one carrier, and more than one trusted service provider. Thus, an alternative approach based on the SE being securely stored on the remote, tamper resistant cloud environment has been introduced. Because of the elimination of problems that are present with the physical SE on the mobile devices, such HCE solutions (SE in the cloud) are becoming ever more popular. The solution takes away the complexity of dealing with the physical SEs and SE issuers (mobile carriers, mobile device manufacturers, trusted service managers) thus, reducing the development and deployment cost, shortening time to market and making the business case more appealing by simplifying the global M-to-N implementation challenge. The main benefits of the cloud SE solution are:

- Providing a mobile ticketing solution alongside the classical contactless smart card without any need to upgrade the existing ticket validation infrastructure;
- It is independent from SE issuers (phone manufacturers, MNOs and TSMs);
- All mobile devices running Android 4.4 or above with NFC and an internet connection can be used (a great proportion of new smart phones and devices);
- Global M-to-N service problems can be easily solved without increased cost for many involved parties in the solution;
- Issuing new cards is easy and comes with no additional costs to users or service providers;
- Possibility of providing open API architecture that takes advantage of HCE and enables third party applications to use CIPURSE features through its open API;
- Cards on cloud can never get lost or stolen; user can reset or move card to new device at any time;
- Different services can coexist on the same infrastructure;
- Fewer or no ticket vending machines which are expensive to maintain;
- Unlimited number of 'CIPURSE applications' for different service providers and operators can coexist;
- Possibility to create an open API for third party service providers for card management (key management, ADF management, etc.);
- Additional card services and added value services can easily be provided alongside: ticket vending, browsing etc.

An Android application with embedded HCE functionality forwards APDU commands to the SE in the cloud. A secure SSL connection between the mobile device and SE in the cloud ensures protection of the additional data that is transferred alongside APDU commands (account balance, user data, etc.) The Android application only acts as a 'relay', routing the APDU commands from validator to SE in the cloud and back, meaning it does not have access to keys and thus cannot process APDUs locally. This approach makes mobile application malware safe, transferring the security focus to the CIPURSE application running SE in the cloud. Consequently, the solution is as secure as the cloud infrastructure is secure.

There are three major challenges when using SE in the cloud:

- Internet availability;
- Internet latency;
- Authentication.

The authentication challenge is a security threat when more than one mobile device addresses the same SE at the same time. A cloud SE solution should provide an additional security check to prevent mobile device intrusion and cloning of user's credentials to access the account on his remote SE. The random, obfuscated token generated when the Android application is installed on the mobile phone and registered on the SE is one way to solve this. The token is regenerated and updated on the server side and mobile application with every transaction. Therefore, only the Android application that sends the expected token can communicate to the requested SE and use the associated data structure. In addition, for extra security, the token can be generated using fingerprint id, SMS check, etc. After a successful application download

and successful user registration, the Android application sends the user's data and random generated token to the application server where the user's data is stored. The application server sends the request to the cloud to create a user associated account in the cloud and map the generated random number to the newly created SE.

This mechanism protects the user's credentials but does not prevent the possibility of the user sharing his or her account, either intentionally or by malware installed on his device.

Two other concerns deal with the availability and latency of the internet. In order to successfully execute validation when using a cloud based SE approach, the mobile application has to be connected to the internet. One of the possible solutions to address this connectivity issue is that service providers or PTs offer internet access points to their customers in the areas where ticket validation is performed. The concern is becoming less and less of an issue with the growth in the percentage of users having prepaid data plans, allowing them to connect to the mobile internet network without additional cost.

Finally, the SE cloud solution faces a latency issue. When validation is performed, commands are sent from the validator to the remote server and backend, causing a slower validation process compared to the standard physical SE solutions. With the future introduction of 5th generation mobile networks (5G), data rates, speed and latency issues should decrease dramatically.

8.4 HCE with offline support (offline HCE)

Besides the benefits of HCE implementations previously described, a multitude of possibilities enabled by the flexibility of software (on the mobile phone and on the host) are waiting to be leveraged by transport networks.

These transport network systems, regardless of whether they are currently supported by a smart card solution or not, are intermittently connected by nature and therefore asynchronous for their most part. In most real life transport scenarios, holding transport credentials within a mobile phone without requiring online connectivity (meaning speed, network coverage, etc.) while boarding on transport services is therefore a fundamental requirement.

Such asynchronous secure transaction systems are best supported by secure element technologies such as CIPURSE or hybrid HCE leveraging CIPURSE, as discussed earlier. However a HCE solution with offline support and without the presence of a secure element provides significant additional flexibility and reach.

Many people have written about the fact that such implementations might lack security which is correct in absolute terms, especially if one considers simplistic versions of HCE; but considering that most leading transport networks already have secondary levels of protection in place (due to the fact that many legacy secure elements have been hacked); these protection mechanisms can be coupled with others, at the mobile phone software level, to provide acceptable risk levels for an HCE implementation with offline capabilities.

It is important to highlight that one should envision use cases that are well supported online through earlier forms of HCE implementation; e.g. registration and sales of transport services would be best performed with an NFC phone connecting to its service host. Offline HCE should not be considered as a generic solution for all use cases (e.g. including registration and sales) but in preference a mix of online and offline depending on security requirements.

To date, only a highly secure element such as CIPURSE can provide the level of genericity and security to support all use cases.

Another important preliminary note is that, in most transport service usage use cases, implementations can be supported by CIPURSE smart cards, in combination with sales use cases provided by offline HCE services. One instance is when the NFC mobile phone serves as a personal terminal to update the smart card with limited value cryptograms that are dynamically managed by the host on an intermittent basis.

Nevertheless, providing a HCE solution with offline capabilities without requiring a smart card is a true game changer for many transport networks. CIPURSE being software based and transport systems being multi-tiers systems, it is entirely possible. The mobile phone environment is a mix (OS, TEE, Apps, etc.) and so is the profile of transport as a service (service registration, sales, usage, product loading, purse reloading/top-up, inspections, customer services, etc.) As a consequence, the number of possibilities offered by CIPURSE is exponential and may create a feeling of complexity at first.

However, each part of the system required to support offline HCE in transport already exists commercially. What does not exist yet is the real life combination of HCE, offline credentials, NFC mobile phones and transport network infrastructures built upon open standards; but many transport networks, solution providers, and OSPT Alliance are busy with making this a reality.

8.4.1 The case for HCE with offline support

For many transport networks, implementing a solution with transportation credentials available offline through a NFC mobile phone will enable not just broad coverage of their supported customer base and compatible infrastructure base; it will also create the potential for collaborations otherwise impossible, thanks to business to business (host to host) decoupled interfacing. For instance:

- Connecting a transport account with a cloud based payment service already supported by international payment networks, banking networks, or alternative payment service providers (e.g. as source of funds for a transport account) and thus alleviating the infrastructure cost of sales services otherwise provided through expensive ticket vending machines, manned POS, and so on;
- Connecting a transport account with a mobile network operator services (e.g. as source of mobile money, online account services such as registration, e-money transfers, load and reloading services, etc.);
- Connecting a transport account with a merchant retailer loyalty program;
- But also enabling direct interactions between a transport service user and other service providers (such as willingly sharing geo-location information to benefit from specifically catered insurance services, promotional offers, and so on).

Another important reason why offline HCE is relevant to many existing transport networks is that transport networks are usually already their own issuers and their own acquirers. In effect they are often providers of tokenized transport credentials in the shape of public transport products loaded onto smart cards. In other words, they have already the most important components of an effective HCE solution with offline support in place, although under different security assumptions.

HCE may therefore be considered as a complement to existing smart cards (as long as they comply with open standards), or as a smart card replacement (when the transport product risk profile and related host based protection measures are compatible).

Replacing the transport account with an offline tokenized transport credential is a challenge that should be explored to protect the transport network assets/transport product. It is clearly a way forward that:

- Resolves the complexity that dooms most account-based/linked experimentations;
- Provides the flexibility required by transport authorities and operators.

This is very different from other experiments that have added to conventional smart card systems the infrastructure and services to replace a transport credential with a payment credential. Indeed, these experiments are entirely different implementations altogether, they do not provide offline support per se; and are not the subject of this white paper.

8.4.2 What about tokens?

As previously mentioned, many transport use cases are best supported by either a secure element (with added security, offline support, no battery support, etc.), pure HCE, or online HCE implementations (e.g. registration, account management, sales, reloading/top-up, etc.). However, the transport service usage use case is central to public transport and thus should drive the goal for a HCE implementation; this implies providing offline support while limiting (or avoiding entirely) impacts on the validation and inspection infrastructures already in place. In most cases it all comes down to:

- The provision of tokenized transport credentials;
- The protection of the token within mobile phone environment; and
- The management of whatever residual risks using a hosted system.

While HCE does not necessarily require tokenization of an account identifier, especially in transport services where the issuer and the acquirer are often the same entity; tokenization should be looked at very carefully as it may be pivotal for your transport network. There are many reasons for this, such as:

- Transport products are often already a form of token for a passenger customer account and account-based/linked credentials; in effect already providing the framework for asset devaluation necessary for an effective tokenization approach;
- Additional levels of protection, such as through shadow account management, are usually already in place and can prove sufficient to mitigate the risks involved with mobile phone based tokens;

- Tokenization is increasingly being adopted by payment networks and technology giants, and thus is desirable as a preparatory step for interoperability, or at least for interfacing between networks hosts;
- Offline is a must-have capability for usage transport use cases and usually implies limited use/value transport credentials so as to mitigate the lack of tamper proof physical security protection;

Here is an example of an offline transport credential token use case:

- CIPURSE uses standard APDU commands to talk to terminals; as long as the transport credential is created while respecting the structure of these commands and considering limited use boundaries to match an acceptable risk profile, the solution is good to go;
- In this example, the transport credential takes the form of a temporary transport card and a temporary transport product, so really we are talking about two tokens rather than one;
- Tokenization creates a temporary ID and associated key(s) and links this temporary ID to a real account. The fare product is tied to this temporary card;
- The terminal derives the keys from the temporary ID and considers this a normal card. The temporary transport product is perceived by the terminal as the normal fare product for a normal card;
- The temporary ID returned as part of the transaction message allows linking of the provided service to the real account;
- Unfortunately, both the temporary card credentials and the associated fare product can be cloned (they are not protected by tamper proof physical security). In an offline context they should therefore be considered where secondary levels of protection are available (e.g. blocking of cloned credentials) or where the fraud business case is negative (e.g. high prosecution cost in case of inspection);
- The temporary card must be provisioned to the phone prior to interacting with the terminal and provisioning the temporary transport product;
    - The life span of the temporary card can be longer than that of the temporary product since its attributes would probably not need location aware data elements;
    - This provisioning is transparent to the users and includes the transport account number masked by a random number generated by the host using issuer key materials only known to the transport infrastructure (e.g. terminal SAM).
- Transport networks that have control over their terminal SAMs are in a good position to leverage offline HCE; terminal SAMs accept whatever appears as a valid card, freshly generated via a temporary ID or previously during account creation;
- The temporary transport product should then be provisioned by the cloud, on the fly while initiating a transport journey;
    - Such provisioning should call for the passenger attention for several reasons; one important reason is that this interaction usually represents the implicit acceptance of the 'transport contract' formed between the passenger and the transport service provider;
    - Through this interaction, limited use boundaries are embedded into the APDU loading the transport credential onto the mobile phone: for example, a limited lifespan of one journey (a consumable token through check-in/check-out or check-in only for pre-specified trip). There are normal fare product constraints, not linked to tokenization.
- In the case where either the temporary card and/or transport product have reached the end of their lifespan, the user would need to go online to authenticate as the account holder (and transparently have its tokens renewed). As long as this happen only occasionally, the balance between security and convenience would still be satisfactory. This can also be an automated process where the user gets informed only in case of insufficient connectivity.

8.4.3 Using payment network Token Service Providers or your own transport network Token Service?

While it's hard to say just how soon Token Service Providers will become an option for transport networks, the trend is clearly set with the general development of the payment tokens market: Samsung Pay, Apple Pay, Google Pay, EMVCo preliminary EMV tokenization specification, Visa, MasterCard, etc., just to mention a few. These

tokenization services are specifically geared towards replacing payment credentials; but replacing transport credentials with payment ones is an entirely different topic with far reaching impacts.

- Payment infrastructures, authorization and processing have a cost;
- Payment ecosystems enable unmatched interoperability but also constraints (e.g. payment scheme participation, business rules, processing costs, terminal specifications, and so on) that have nothing to do with the public transport's mission;
- Transport authorities are obliged to serve all, even those without bank account, payment cards, and so on;
- Transport networks, being both their own issuers and acquirers of a lot more than just payment credentials, will find that their total cost of ownership cannot be computed solely based on payment related solutions (e.g. including the cost of accepting concessionary transport products, the cost of issuing payment credential for those without bank account, and so on).

On the other hand, many global and local independent solution providers are offering tokenization services (as well as HCE, TEE, SE life cycle management, etc.), including for transport. Not all merchants and banks are prepared to be intermediated by a payment network at a time when mobile technology enables more intimacy with their customers and thus better branding of their own services. This is good for the transport ecosystems as it results in a rich offering of effective independent solutions.

Last but not least, a transport network with tokenization capabilities of its own will open up its collaboration possibilities; such as bridging its network with other digital service networks such as:

- Payment and banking networks;
- Mobile networks;
- Merchant networks.

Will transport networks create tokenization services enabling collaborations that serve their public transport mission, leverage their local infrastructures, provide benefits to their entire local ecosystems, and eventually provide the basis for standardization at mobile software and host services levels? Only the future will tell; but transport networks are possibly the best candidate to do so considering that they hold both the critical mass of daily users and infrastructures that are 'HCE ready'. The key for such collaboration to be enabled is to build upon open standards, such as CIPURSE and OSPT Alliance system-level open specifications.

8.4.4 Risk management and offline support

In an offline HCE implementation, risk management, account-holder verification, and transaction authorization are not performed online. Instead, check and balance processing are performed over a window of time that is driven by settlement agreements between the transport network participants (transport operators and retailers); within 24 hours is usually viewed as sufficient.

During this window of time, a shadow of the credentials and their impact on customer accounts are computed; transactions are verified; tokens are tracked; manipulation of card balances are tracked by comparing value top-ups and value usage, possible cloning of card data are monitored. In the case where illegitimate use of a card (real or temporary) has occurred, the transport network infrastructure devices are sent blocking instructions for that card; thus mitigating the risk to journeys travelled on that same day with cloned cards/products.

It should be noted that the situation is entirely different should the token be used for tradeable goods such as with a payment application. On the other hand, because the transport credential token has limited value in the first place, its manipulation would not provide a business case as attractive as the hacking of a weak secure element in the cases of those hacked so far.

This model has been around in transport for literally 20 years and works according to specification. Why should it be different with a mobile based virtual card or transport product? In essence it is the same, with two important additions:

- The illegitimate manipulation of a software based virtual card or a virtual transport product in a mobile phone environment is more likely than of a real card protected by a tamper proof hardware;
- Such manipulations are potentially economically viable for the fraudster as there is no physical product involved (i.e. less or no cost) to scale the hack.

Therefore, transport networks considering offline HCE would be wise to invest in:

- Additional card and product tracking solutions (such as through usage pattern analysis, velocity checks, and so on);
- Remotely upgradeable mobile software protection mechanisms (e.g. to defeat the purpose of hacked mobile applications);
- Remotely upgradeable acceptance infrastructures (e.g. to block cloned tokens).

The good news is that such investments would also benefit conventional smart cards that are still relying on hacked proprietary products.

8.4.5 Implementation options to protect tokens offline

Several implementation options are available to protect a token offline. They should be considered together with the asset devaluation brought by the token definition as both, together, define the level of security that an offline HCE implementation will provide.

Clearly many transport networks will not be comfortable with such leading edge hi-tech projects. But considering the massive success of purse based smart card systems for transport in places like Asia, and the potential benefits of bringing offline HCE to the picture, these projects may prove strategic for many.

In order to achieve the same magnitude of adoption than purse based smart card systems, tokens should be pervasive enough (available for the majority of mobile phones). Unfortunately, standardization of handling tokens offline on the mobile phone and standardization of linking private host based services to MNO verification capabilities are still at their early stages.

Nevertheless, some of the possible implementations are:

- Obfuscation and white-box cryptography (not ideal but with asset limitations of transport credentials, it may be good enough and pervasive enough);
- Verification using mobile phone factors such as a mobile phone/application unique 'signature' or watermark (mobile phone industry associations are yet to release an industry wide standard for practical and effective content protection);
- Device specific and application specific security (not ideal considering the market's fragmentation);
- Another alternative is through mobile phones TEE (Trusted Execution Environment) complying with GlobalPlatform TEE Specification: less secure than a CIPURSE secure element (the TEE resides in unsealed hardware processing environment and relies on software security); but more secure than OS/App based alternatives). However the fragmentation of mobile phones and the need to procure a proprietary software solution are significant drawbacks that may defeat the purpose of implementing an offline HCE in transport;
- Of course a CIPURSE secure element based hybrid HCE model is always preferred and would probably prove more cost effective, especially since CIPURSE embedded secure elements are available nowadays; but this section assumes that no secure element is issued in the first place.

8.5 Account based HCE

Account based functionality is a critical requirement for many transportation agencies as it allows for a solution to support mobile devices readily. In contrast to a stored value approach, account based HCE allows for processing to take place on the back end, rather than the front end. This change essentially allows users to pay as you go, and the mobile device becomes the key for readers to access transportation and have their device become the trigger for billing later, rather than decrementing value from a balance stored on the device.

The benefits of the approach are many, as the transit agency can readily support many new device types provided they can be securely recognized at the fare media reader level, without the need to access back end systems to check balances. This is especially useful for transportation agencies that support variable fares based on usage. Account based is usually either threshold based (users keep a balance which is recharged when it goes below a certain level) or pay as you go (users are charged to their payment method of choice either as they use the system or at pre-determined intervals).

Transportation agencies gain through account based approaches in several key ways. First, they can use reusable secure fare media such as a mobile device using HCE or the SE to keep the UID secure. Next, they can allow for users to interact with the system without using ticket vending machines which are expensive to

maintain. Finally, account based improves customer service as there is no need for near real time data exchange between the ticket vending mechanism and the fare gate. Transit agencies sometimes adjust for that delay by allowing customers to use the system without knowing what their balance really is, leading to the risk that some riders could scam the system once they become aware of the delay. Furthermore, using HCE to implement account based models does create risks (impersonation, stealing and reusing associated keys, and so on) that need to be addressed. CIPURSE with its uniform way of handling multiple fare media types supports account based functionality readily as the feature set required for account based is a subset of the feature set for stored value. CIPURSE can be used to recognize the user ID as valid in real time. Additional CIPURSE functionality such as the ability to recognize key variables can be used to manage risks associated with an account such as whether the account has been recently recharged, has repeatedly been flagged for bad debt, or is invalid.

9. Impact on CIPURSE Specifications

One possible impact is when HCE is used in combination with account-based services that do not require the user data to be dynamically modified while interacting with the system. This implies that the emulated card in use during such host based transaction does not necessarily require the same trust to be established (as opposed to in a conventional secure element interaction implementation). The acceptance network (the reader/validator/gate) must verify that it is interacting with an authentic (remote) account (to authorize access, as long as the account is white-listed); on the other hand it does not update the data. Consequently, in the particular case of account-based transactions, the credential used to interact with the acceptance network does not need to verify the authenticity of the acceptance network.

Moreover, HCE needs to be even faster than a physically present SE given the additional transaction time overheads implied by the mobile network latency.

A faster and simpler authentication method would therefore be welcome to support such account-based HCE implementations; for example, CIPURSE™ support for one-way-authentication.

Another possible impact is the logical evolution related to collaborations between transport and non-transport network based on account-based systems. In such scenarios, accepting an account credential without requiring the use of symmetric keys would enable new business models and unleash the potential of public transport networks as a central part of today's digital life. For account based model where no data are written to the SE, use of asymmetric algorithm(s) would allow for acceptance devices without a SAM; nevertheless, the security of the private key used for signature generation should be considered. Considering that OSPTA's SEs are all recent and can include support for Elliptic Curve Crypto protocol (ECC), this further evolution of the CIPURSE protocol is another desired goal.

Please note however that such an implementation (involving ECC) would result in longer transaction times. Indeed, both signature generation (by the transport credential) and signature verification (by the acceptance network) contribute to the overall transaction time. If several card issuers are to be considered, certificate (chain) verification by the acceptance network is to be added. If network authentication is also required, transport credentials also need to verify a certificate (chain).

10. Implementation examples

10.1 Medius Cloud SE

Medius CloudSE is an open API architecture that implements CIPURSE™ specification as a cloud based secure element (SE). CloudSE exposes CIPURSE functionality to third party applications through secure web services. The architecture takes advantage of Android's KitKat 4.4 Host-based Card Emulation (HCE) library for selecting CIPURSE applications stored on CloudSE and tunneling APDU commands to/from CloudSE.

The whole solution is independent of third party SE issuers, such as carriers and device manufacturers and does not require any upgrade of the existing ticket validation infrastructure. Furthermore, this API architecture enables third party developers to implement their own applications and value added services that can easily access web services exposed by CloudSE. The following table illustrates the correlation between the usual CIPURSE smartcard implementation and the CloudSE:

CIPURSE smart card	CIPURSE CloudSE
CIPURSE application as Native/JavaCard applet	CIPURSE application as a Web Service
PxSE for application selection	HCE + HTTPS (REST) + token
Tamper resistant EPROM	High Availability, cluster data storage in the cloud (HSM implementation)
Native/GlobalPlatform oriented interfaces for card administration	CloudSE card management Web Service or WEB application
Multiple ADF's as multiple applets	Unlimited instances of CIPURSE ADF database structures

Figure 1 CIPURSE smart card and CIPURSE CloudSE correlations

The following picture represents the overall CloudSE architecture. It consists of three major components:

- **CloudSE** - High availability redundant computer clusters that provides continued service of executing APDU commands for multiple CIPURSE applications stored on the tamper resistant data infrastructure. The infrastructure is fully protected and audited. CloudSE exposes SSL REST web services that enable communication between the third party client applications (Android, web, etc.) and the CIPURSE SE functionality;
- **WEB Application server** – A cloud portal web application infrastructure that enables certified users (service providers, system integrators) to use CloudSE infrastructure for administration needs. The web application enables CIPURSE card creation, key and ADF management, default value creation, etc. The web application server also runs the web application for end user requests such as card instantiation, personalization, card refill, etc.;
- **Mobile HCE application** - Android application that incorporates a HCE library to enable communication between NFC validators and CloudSE web services. Application is a bridge that tunnels the APDU commands to/from CloudSE core;

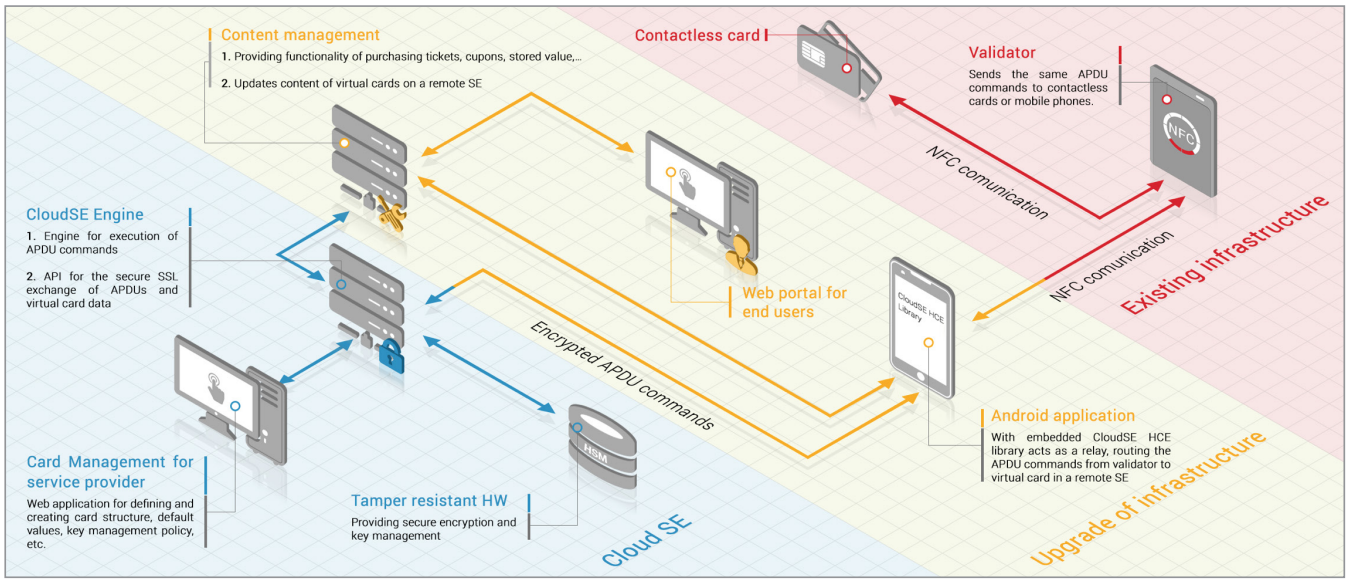

Figure 2 Medius CloudSE architecture

- REST Web service access API for accessing CloudSE, handling card lifecycle and exchanging APDUs between ticket validators and CloudSE SE via HCE technology on the mobile phone;
- Web application and open Web service API for card and key management for service providers and integrators. Using the web interfaces, service providers and system integrators can define and create their own CIPURSE card structure, default values, key management policy, etc.;
- Android library on top of HCE for the third party mobile application enabling third party application developers to simply implement HCE communication with NFC validators.

10.2 Telenor HCE approach

With HCE it is still possible to use a secure element on the mobile phone as a provider for security. One example of how this can be done is illustrated by a two factor authentication approach using something you have and something you know [MOBISECSERV]. In this user scenario a CIPURSE HCE application and a PIN (something you know) is used together with a secure element applet (something you have) on the SIM (i.e. UICC).

In the scenario, the user is registered at the CIPURSE application specific back-end with a unique triplet: the CIPURSE card ID, a 128 bit AES key generated from the PIN code and a MSISDN number (phone number). The user has paid up front for a number of uses (e.g. paid for ten trips or for a monthly ticket) in the CIPURSE HCE application. The two-factor authentication scenario starts when the owner of the CIPURSE HCE card intends to use an available transportation (e.g. bus, metro or boat). The user opens the CIPURSE HCE card application on his/her mobile device and is prompted for a personal PIN code. To get a valid ticket, the user selects the 'get valid ticket' option in the application and holds the phone to the contactless card/ticket validator. A beep signal is given from the validator after less than 0.5 seconds and the user retracts the phone and sees a verbatim two word secret displayed in the HCE application (e.g. 'Cold Milk'). While waiting for the transport to arrive (or during the transport), the user is, in a separate pop up window on the phone, asked to confirm that the popped-up verbatim secret is the same as the one previously presented in the CIPURSE HCE card application. The user can confirm by typing his/her PIN again or deny by pressing the cancel button. If the user confirms that the secrets are equal, the ticket is valid. If the user does not confirm the secret or inserts a wrong PIN code, a valid ticket is not issued. The separate pop up window with the repeated verbatim is presented by the secure element applet on the UICC and not by the CIPURSE HCE application on the mobile phone. The secure element applet on the UICC is a SIM Toolkit application running on the UICC. The applet has GUI elements that are visible on the mobile phone screen. As with tokenization solutions this method relies on the mobile phone to be connected. In this user scenario, 'connected' means being able to send and receive binary SMS messages (used to communicate with the UICC) and to be able to communicate with the back-end system using a data connection (e.g. provided by the mobile operator or a Wi-Fi connection).

Mobile Network Operators are accelerating the deployment of Mobile Connect[3] [GSMA_MC]. Mobile Connect can be used as the second factor in order to solve the security issues for a CIPURSE HCE application as described in the two factor authentication user scenario above. With Mobile Connect, the mobile internet device (e.g. mobile phone) can act as a local server authenticating the (ticket) transaction. Note however that the mobile internet device must have an internet connection and be able to send/receive binary SMS messages.

[3] http://www.gsma.com/personaldata/mobile-connect

11. Conclusion

This paper has demonstrated that while there are clearly a number of areas where public transport requirements may prove challenging for HCE, for many operators, the benefits could outweigh the challenges, especially where implemented in conjunction with CIPURSE™.

There are multiple methods of implementing CIPURSE with HCE. Each method has its own advantages and disadvantages, but CIPURSE is able to support many types of HCE implementations in a way that proprietary approaches cannot. Offline HCE, for example, does introduce additional risks but mitigations are available that would in any case be highly beneficial to system operation. An ideal approach is a combination of online and offline HCE, depending on security requirements. However each implementer will select the approach best suited to their particular transport system.

There is no solution that would be the best for every situation. What is clear is that leveraging HCE and tokenization, along with CIPURSE, will put transport operators in a key position not just to provide significant benefits to their local customer base but to also play a key role in open standards based public private collaborations going forward.

12. Appendices

12.1 Lifecycle of CIPURSE Application hosted on Android and Windows 10 HCE

[GOOGLE_HCE_CX] Android's HCE implementation is designed to work in parallel with other methods of implementing card emulation, including the use of secure elements. This coexistence is based on a principle called 'AID routing': the NFC controller keeps a routing table that consists of a (finite) list of routing rules. Each routing rule contains an AID (application identifier) and a destination. The destination can either be a specific application (HCD service) running on the host CPU or a connected secure element.

Android applications that implement a HCE service or that use a secure element do not have to worry about configuring the routing table - that is taken care of by Android automatically. Android merely needs to know which AIDs can be handled by HCE services and which ones can be handled by the secure element. Based on which services are installed and which the user has configured as preferred, the routing table is configured automatically.

Even though AID registration is defined by ISO/IEC 7816-5, Google does not provide a mechanism to verify reserved AIDs. This means that two or more HCE applications installed on user's device could register the same AID addresses. In such situations Android OS prompts the user to select the appropriate application for the transaction. AIDs must also be registered for a specific category. Currently, Android OS only supports two options: payment applications and other. A conflict can occur when the same AID is used within the same category. Android OS allows the user to select the default application for each category; meaning the user is not prompted when collision occurs. Android OS simply forwards the commands to the default application.

More information on how to declare AIDs for applications that use SE for card emulation can be found here [GOOGLE_HCE_CX].

Microsoft also added HCE support for the mobile devices running Windows 10 OS. Before that, Windows with version 8.1 only supported NFC with SIM based SE. Windows 10 HCE implements an architecture that is like Google's. As with from Android OS 4.4 version onwards, AIDs used in the HCE application have to be registered at the time of installation. APDU commands following the 'SELECT command' are routed to the application that registered that AID. Handling AID collisions is also performed the same way as with HCE on Android.

- Installation
  - Objective: assign AID;
  - Mapping of CIPURSE™ application installation onto HCE;
  - Dependency on partitioning alternatives should be considered.
- Initialization and Personalization Phase
  - Registration @PxSE;
  - Population of security credentials;
  - Creation of EFs;
  - Population of EFs.
- Operational phase
  - CIPURSE command execution.
- Application Deletion
  - De-registration from PxSE;
  - Releasing any eSE / server connection/instance.

12.1.1 PxSE

The PxSE applications reference in their SELECT response the AID list of the CIPURSE applications registered on them. The PxSE AID specified by the OSPT Alliance in the 'CIPURSEV2 Registered PxSE AIDs' document can be used according to the purpose of the application to be registered. For example the PTSE AID can be used to create a PxSE referencing a transport application. In parallel, the terminal (Validator, Vending Machine etc.) will use the PTSE AID to retrieve all the AIDs of the registered and activated CIPURSE transport application(s) on the device.

Only one instance of a dedicated PxSE (e.g. a PTSE) can be present on a device, as a consequence an HCE based CIPURSE application must be registered on a HCE based PxSE application having an AID not already used on the device (e.g. a PTSE instantiated on a security domain of the UICC of the same device, or a PxSE AID already referenced in the HCE application registry).

The way an HCE based CIPURSE application is registered on a HCE based PxSE application is being standardized by OSPT Alliance Mobile Working Group.

- Partitioning Options (S/W and H/W combinations);
  - Alternative implementations with/without eSE, remote server based, etc.
- Perspectives: security, operational, architecture, personalization alternatives;
  - Specific to each implementation.
- Risk Assessment of Partitioning Options.

12.2 OSPT HCE Certification Process

An independent third party certification process has been established to certify products compliant to CIPURSE™. A product can be called CIPURSE certified only if it has successfully passed the functional evaluation. The CIPURSE certification program provides transport authorities with the confidence that CIPURSE products provided by different vendors are interoperable and compatible with transport and ticketing ecosystems implementing CIPURSE. The program tests all consumer products including different form factors such as limited use tickets, plastic cards, stickers, key fobs and NFC smartphones for both file system oriented and Javacard-based CIPURSE products. The OSPT Alliance is also considering the certification of HCE based CIPURSE products. The HCE based CIPURSE product must support an administration phase based on Global Platform oriented CIPURSE product as specified by the OSPT Alliance, but limited to the INSTALL for INSTALL, STORE DATA and DELETE commands. In fact the clear mode will be used without any authentication as this personalization method will be activated and used only for CIPURSE certification purpose. As a consequence, all the tests of the 'CIPURSE V2 Certification Program Conformance Test Plan' related to the personalization method itself will not be executed by the Test Laboratory for the certification of a HCE based CIPURSE product. The vendor will provide the AID of the emulated Card Manager in the ICS.

For more information please visit www.osptalliance.org