computers are bad

https://computer.rip - me@computer.rip - fax: +1 (505) 926-5492

2023-09-03 plastic money

You will sometimes hear someone say, in a loose conceptual sense, that credit cards have money in them. Of course we know that that isn't the case; our modern plastic card payment network relies on online transactions where the balance tracking and authorization decisions happen within a financial institution that actually has the money (whether it's your money or credit). There is an alternate approach though, one which has historically been associated with terms like "epurse" in the technology industry: what if the balance tracking and authorization decisions were actually made inside of the card?

Ten years ago this proposal might have seemed more absurd in the United States (amusing since, of course, the technology to facilitate this is much older). Payments in the US were made using magnetic cards with two tracks totally hundreds of bits. Debit cards could contain a challenge value used to verify the PIN, but even then the decision of whether or not to accept a PIN offline was made entirely by the payment terminal. Fraud related to these cards became a problem quickly enough that offline processing of card transactions (for example by the use of a "kerchunker" impression machine) became very rare, and all transactions were conducted online. Even then cards were vulnerable to duplication and this was a fairly common form of fraud.

Europeans, though, had been using smart card technology dating back to as early as the '80s in France. These cards had an onboard microcontroller that could make decisions and even run applications. Inside of the card there is nonvolatile storage that can retain a cryptographic key, allowing the card to participate in a cryptographic challenge-response process that made duplication very difficult. Even better, PIN verification was performed inside the card, meaning that even a malicious terminal could not accept an invalid PIN during an offline transaction.

And today, that's the most widespread application of smart card technology: cryptographic challenge-response authentication. The technique is ubiquitous both in payment systems and in access control and ID verification, spanning a wide gamut of capabilities from DESfire keycards to the United States Government's behemoth of an identity credential standard, PIV.

It's sort of interesting that these less ambitious applications of smart cards are about as far as they've gotten in the United States. Their capabilities are much greater than modern applications suggest. Smart cards were, from the very beginning, conceived as much more powerful multi-application devices that were capable of enough internal accounting logic to implement true stored value cards, or SVCs. Cards which "contained" money in a balance that could be debited and credited fully offline, just by a terminal communicating with the card.

First, bit of history of the smart card. One of the reasons that smart cards have made relatively little inroads in the US is their European origin. Nearly all of the development of smart card technology happens in European companies companies like Gemplus (Netherlands) and Axalto (France), today merged into Gemalto, part of French defense conglomerate Thales. Not to be understated either is the German company Giesecke+Devriant. Many early developments happened within the French Bull group as well, which through merger into Honeywell continues to make related products. Identity technology vendor Morpho, later Safran Morphotrust, today Idemia, forms the backbone of the TSA and Border Patrol's ubiquitous travel surveillance from their headquarters in the suburbs of Paris. They are further accused of providing identification technology to Chinese government agencies for purposes of oppression. Identity is a sticky business.

These companies have a long-running relationship with secure identity. G+D has long been a major international center of expertise in currency manufacturing and security, and the US Federal Reserve System for example relies on G+D equipment to detect counterfeits. Gemalto became one of the primary vendors in secure digital identity technology, and Thales carries this on today, providing major components of the US federal government's USAccess/HSPD-12 scheme.

It all began with payphones. Well, that's not true, there were plenty of developments in smart card technology before they were applied to payphones, but payphones introduced the technology to the French masses in 1986. France also pioneered chip-based online transactions, with a nationwide ATM network based on smart cards in 1988 and ubiquitous issuance of a precursor to EMV in 1993. We have to be careful to differentiate online and offline systems, though. One of the confusing issues around SVCs is their functional similarity to online transaction cards using an integrated chip for authentication purposes. To understand more clearly, let's take a closer look at one of the most common SVC applications in the US: the laundry card.

Laundromats are conceptually simple; each machine needs a coin acceptor (often limited to quarters only) and a coin vault. In practice, it's a little more complex. Most customers don't walk in with enough quarters any more, so the laundromat has to provide a change machine. Change machines, being stocked with hundreds of dollars in coins and bills, are an attractive target for theft. Besides, they aren't that reliable. Emptying the coin vaults on each machine daily is a time sink for staff, especially when the risk of theft requires multiple staff members as a precaution. Wouldn't it be easier if laundromat payments were electronic?

Today there are a number of ways to achieve that end, most of them worse than the old system of rolls of quarters, involving some combination of QR codes, smartphones, Bluetooth, and "The Cloud." These approaches were a nonstarter in the '00s, though. Wireless networking was in its infancy, the cost of putting network connectivity in every machine was very high. A solution was needed that allowed the billing devices in machines to be offline, operating totally independently. Any case of offline payment terminals calls for SVCs.

So in many laundromats even today, there is a device on the wall somewhere called a value transfer machine, or VTM. Actually the term VTM is a trademark of one of the major vendors of these systems, ESD, but it's such a good generic term that I will disregard their claim and use it across vendors. At the VTM, a customer either inserts their smartcard or presses a button indicating that they want to purchase one. The VTM accepts a payment by either cash or payment card, and then "transfers" that "value" to the inserted smart card---or a new one dispensed from an internal stack. Pricing details vary, but smart cards aren't as cheap as anyone would like and so it's common to charge a few dollars for a new card. Customers are encouraged to keep their card for the long term.

What happens internally? A very simple implementation suffices to explain the concept. On the smart card, there is a value in nonvolatile memory that represents the amount of money on the card. When you add money, the VTM increments that value. When you insert the smart card into a laundry machine and start a cycle, the billing device in the machine (usually a drop-in replacement for a coin acceptor with the same electrical interface) decrements that value. And there you have it: the card is just like cash, representing value on its own, with no online operations required.

Of course you can see the problems with this scheme: couldn't anyone just write a bigger number to the card? The earliest implementations tried to prevent this with simple password schemes or very elementary cryptography, and results were poor. The French payphone system of the late '80s, for example, was known to be vulnerable to duplication of cards and so naturally a black market emerged.

The history of early SVCs, mostly of the '80s and '90s, being vulnerable to at least duplication if not outright forgery gave them a poor reputation for security that persists to this day. It doesn't need to be that way, though, and excepting some obsolete systems still in use it isn't. If we can make the blockchain work we can certainly make SVCs work (admittedly this somewhat self-defeating argument presages the failure of SVCs to catch on for general purpose use). The problem with early SVC systems was the limited computational capabilities of the smart card, no match for the high complexity of strong cryptographic algorithms. Smart card technology advanced, though.

The term "smart card" is not very precisely defined but tends to refer to any card with an Integrated Circuit Chip (ICC) compliant with one of several specifications for physical and electrical interface, mostly ISO 7816 for contact operation and ISO 14443 for contactless operation. It's important to understand that while the term "smart card" is most often used to refer to contact operation, that's not a limitation of the technology. Historically some cards implemented contact and non-contact operation by having two separate chips, but that method is well obsolete. Modern smart cards, especially payment cards, are usually dual-interface cards where the same ICC is capable of communicating the same logical protocol over either the contact interface ("insert") or the noncontact interface ("tap"). Since the noncontact interface is compatible with NFC, smartphones are able to use their secure element to run an application similar to the one that runs on EMV cards.

If these cards are so smart, what do they actually do? Well, that part has varied a great deal over time. The earliest smartcards, developed in the '70s, were essentially memory and nothing else. Later on, though, smart card software evolved to multi-application cards in which a smart card operating system provides services and manages the selection of applications.

Perhaps the most famous smart card operating system is Java Card, a platform that allows smartcards to run constrained Java applets. Java Card was developed by French conglomerate Schlumberger, whose identity and card division spun out to form a major part of Gemplus (now part of Thales). Besides supporting very constrained devices, Java Card was designed for the high-security applications typical of smart cards. It provides full-featured cryptography up to ECC on modern devices, but more importantly enforces security isolation of applets and their communications and memory.

Java Card is particularly widely known because of its role in the "Java Ring," a chunky fashion accessory that presents a Java Card environment in the onewire-based "iButton" form factor. iButtons are a topic for their own post one day, being surprisingly widely used in a couple of niches where their improved durability over ICC-type smart cards is an advantage.

Java Card is also widely used, being one of the most common operating environments on practical smart cards. There is a good chance that you have more than one Java Card environment on your person at this moment. Discussing the full scope of Java Card applications requires a bit more rambling on the smart card as a physical object, though.

If you are a dweeb about identity documents, you have probably read into ISO 7810. This standard describes a set of physical form factors for identification documents. Most notable is the ID-1 form factor, which is widely used for payment cards, driver's licenses, and in general any standard-sized wallet card. Size ID-3 from the same standard is the norm for passports. But then there's an apparent oddity, size ID-000, a small 25x15mm card with a notch out of one corner. Sound familiar? ISO 7810 ID-000 is the physical description of a conventional SIM card.

SIM cards are just smart cards. Big reveal, I know! GSM was standardized by an organization out of Paris in the same time period that France Telecom adopted smart cards for payphone payment. When looking for a transportable means of authenticating the phone owner, it was an obvious choice. SIM cards no longer conform is ISO 7810 in most cases (having migrated to the smaller micro and nano formats), but continue to be compliant with ISO 7816 for electrical and protocol compatibility. It is no coincidence that SIM cards are often shipped in an ISO 7810 ID-1 compliant carrier, since these make personalizing and testing in the factory easy to do with standard smart card interfaces.

ISO 7816, the standard for smart cards specifically, describes the physical position and layout of the contacts on the ICC. It also describes an electrical interface [1] and logical protocol for communication with smart cards. Smart card communication is based on APDUs, or application protocol data units, packets exchanged between the reader and card. APDUs can indicate a standardized cross-vendor operation code, or a proprietary operation specific to some application on the card. This is a little network protocol used within the confines of the card slot, and smart card applications specify which APDU commands must be supported by cards.

The abstraction of the fairly well-defined APDU protocol creates a healthy degree of separation between smart card uses and implementations. This is all to say that the software running on smart cards often varies by vendor, even within a common application. Java Card is very common, but not universal, for both SIM cards and EMV payment cards. It competes with "native" operating systems like MULTOS. These native operating systems tend to leave more memory and processor time for applications because of the lack of a bytecode interpreter (yes, Java Cards actually run a very constrained JVM), but usually lead to application development in C which is less appealing than "weird constrained Java" to many organizations.

As you might imagine given this range of applications, security expectations for smart cards are high. In fact, the modern concept of a "secure element" largely originates with smart cards, and many secure elements in things not shaped even remotely like cards continue to use the ISO 7816 logical interface and Java Card. The SIM card is really just a portable secure element, capable of running multiple applications with nonvolatile storage, and in some countries (mostly European) they have been used for broader identification and authentication purposes. Smart cards are expected to be resistant to both electronic and physical tampering. Smart cards were historically a common form factor for cryptographic secure elements, being used to protect key material of sensitivity ranging from satellite TV scramble codes to military communications equipment---although for reasons of both durability and not having been invented overseas, the US NSA has historically preferred more homegrown form factors for cryptographic elements.

Putting this all together, you can probably see that it is indeed possible to build a reasonably secure stored value smart card system. All increment and decrement operations can be cryptographically authenticated. Unique secret keys, "burned in" to cards as part of personalization and not readable from outside of the secure element, can be used to authenticate the card and prevent duplication. While it is conceptually possible to duplicate stored value cards through laboratory analysis, the cost is unlikely to be less than the value cap imposed by the SVC service.

In the '90s, SVCs started to catch on. A marquee implementation went on display to the world in 1996: at the Summer Olympics, held in Atlanta, three banks partnered with the Olympic committee and businesses to offer an SVC payment system. It was particularly appealing to international visitors: debit and credit cards rarely worked overseas in 1996, and tourists in the US for the duration of the Olympics could hardly be expected to open US bank accounts. SVCs provided a convenient alternative to cash. Visitors could buy them in fixed denominations with cash or travelers cheque, and value could be reloaded at kiosks around the Olympics sites. The SVC nature of the system allowed the offline payment terminals to be deployed to area businesses relatively cheaply, without a requirement for a phone line like the credit card terminals of the era.

The Olympics SVCs were manufactured by the usual suspects: Gemplus, Schlumberger, and G+D. The cards ran a cryptographic application generally based on the existing French payment card system, a precursor to EMV that was focused on supporting offline use-cases. The Olympics experiment was mostly considered a success, with few technical problems. The banks involved were apparently underwhelmed at the number of cards issued, and it was speculated at the time that they were perhaps more popular with collectors than users. One can imagine that the SVC technology, entirely new to locals and visitors not from Western Europe (and, to be fair, some from Australia), faced some challenges in gaining consumer confidence.

SVCs became a standard feature of the Olympics for a few years, making their last appearance (as far as I can tell) in 2002 at Salt Lake City. This was reportedly a very limited system based on magnetic stripe cards, and so I assume that it was not an SVC system at all but just a gift card system with the heritage of the 1996 and 1998 SVCs. It is likely impossible to design magstripe SVCs that are not vulnerable to trivial duplication, I know of only one method and it is experimental (characterization of weak permanent magnetic fields acquired by the magstripe during the manufacturing process, which seem to be unique enough to differentiate individual cards).

SVCs saw other experimental applications at the same time. The University of Michigan deployed a smart card SVC in 1996 as well, allowing students to load funds and spend them on campus and at nearby businesses. This type of program became fairly popular at large universities, but beware a terminological challenge: many universities still refer to their student ID payment card program as a stored value card for historic reason, but none that I'm aware of today actually are. With universal acceptance of payment cards, it is far more cost effective to make an arrangement with a bank and processor to encode student IDs as Visa or MasterCard cards. They then function as specialty prepaid cards with whitelisted merchants and purchase types, a service readily available from the prepaid card issuance industry.

Another nascent application of SVCs in the US were welfare programs like SNAP and WIC, implemented through a system called Electronic Benefits Transfer or EBT (EBT replaced the physical "stamps" in "food stamps"). Once again, while a few states adopted SVCs and may even still call their EBT cards SVCs, every example that I know of today is processed on the Visa or MasterCard network as a prepaid card.

Why is it that SVCs gained so little traction for payments in the US? A 1999 Spectrum article rounds up the state of SVCs at the time, optimistically opening that the contents of your wallet "might be replaced by just two or three smartcards." One look at the typical wallet will show that this hasn't gone as hoped. The true promise of multi-application cards, that you could have your government ID, payment card, health information, etc. all as applications on a single physical card, is virtually nonexistent in practice. Outside of specialty systems like PIV, the multi-application capability of smart cards is mostly only used to interact with different kinds of payment networks. Perhaps the most common smart card application in the US is called "CHASE VISA" and it is basically the reference EMV application with the name changed [2]. If there's even a single other application on the card, it's probably for interaction with an EFT network.

It's fairly easy to see why this happened: different applications are issued by different organizations. The thought of your driver's license and credit card being one physical object almost certainly induces nightmares of having your credit card number stolen and then having to interact with the DMV (or, as it is pronounced in the New Mexico vernacular, the MVD). The practical logistics of multi-application cards are difficult to manage, and the cards are cheap enough that it's easier for everyone to keep different applications separate.

What of payment cards, though? Smart cards for payments are now the norm even in the backwards United States [3], but stored value systems are harder to find today than they were in 1999. Spectrum elaborates, after discussing the popularity of smart card systems (broadly defined) in Europe:

> Why has it taken so much longer for smartcards to take off in the United States? In the first place, some of these cultural and political drivers are absent. The country has an excellent telecommunications infrastructure. There is no governmental or centralized mandate in any of the traditional application areas of smartcards. But the industry is evolving. The activities of Europay, MasterCard, and Visa (EMV) in developing specifications for financial-transaction cards will have a major impact on the U.S. market and the rest of the world. Nonetheless, it is felt that a smartcard will have to be able to handle several applications for the technology to gain widespread acceptance in the United States.

EMV sure did have an impact in the US, even if it took a solid decade. Multi-application cards seem dead in the water from a practical perspective; even though many more sophisticated smart card systems (like MULTOS) are designed for remote issuance and updating of applications. Anyone who has had access to their office and email at the mercy of the USAccess/HSPD-12 PIV scheme can attest that its Thales-built remote personalization system is... not exactly ready for the average consumer.

Besides, the telecommunications point is not to be underestimated. By the time SVC technology competed in America, telephone connected payment card terminals were already becoming the norm (mostly from American Verifone, although French Ingenico was a major player). Rates of telephone service and, not long after, internet service in the US were very high. These factors made offline systems much less attractive: merchants unhappy with the risk of offline processing of non-chip credit cards were just moving to online processing, not to smart cards.

The lack of a standards body to set the direction is also undeniably a factor. The introduction of EMV took as long as it did in large part because of the fragmentation of the payment card industry; different components of the market had different objectives and there was no one to push them along. To be fair, US payment card issuers cope with fraud better than most overseas observers seem to give them credit for. The inconvenience of card fraud is relatively low; I recently had credit card information stolen (how, I can only speculate) and there was no action involved on my part beyond responding to a text message alert and receiving a new card in the mail. Because of the card information update service the processors provide to qualified merchants, I haven't even had to reenter my card information on any subscriptions. A lot of effort has been put into smoothing over the fraud that occurs, even if it does seem that one of my cards is used fraudulently every two years or so.

Despite the lackluster adoption of SVCs in the US, they have a few strongholds, both here and abroad. First, although a somewhat minor detail, I cannot help but note the military overseas SVC program that my career once incidentally involved. The EagleCash system, operated by the Department of the Treasury, provides SVCs to members of the armed forces (sometimes branded NavyCash or Armed Forces EZPay due to variants of the program rules). The cards are mostly used in overseas military installations and aboard ships, situations in which offline processing can remain a big advantage. EagleCash was considered the most prominent deployment of SVCs in the US, and probably still is. EagleCash reaches nearly a billion dollars in annual turnover, mostly in the Middle East.

Much more widespread, though, are transit cards. Many transit systems globally use some sort of SVC for fare payment, under different names in different cities. Prominent US examples include Clipper (SF Bay area), Ventra (Chicago), MetroCard (New York City), and SmarTrip (national capital region). Overseas, Oyster (London), Rav-Kav (Israel), and Octopus (Hong Kong) are well-known. Many of these systems were pioneering when implemented, and some remain pioneering payments technologies today.

Many early US systems, such as Clipper, were implemented at least in part by the Cubic Corporation. If that sounds like an ominous defense contractor, it is. Cubic produces a wide range of C4ISR systems for the US military, but because of its location (in the Bay Area) and early involvement in transportation technologies, Cubic became a major US vendor in transit fare collection systems. The nearly identical fare gates of BART and the DC Metro, for example, were early models designed and built by Cubic (BART and DC Metro are twin projects in many ways). They originally used magstripe tickets, and I have read that they were controlled by PDP computers although I am unsure if this factual or just confusion with the better documented use of PDP/8E computers to drive the train arrival signs and announcements.

Cubic came back in the '00s with noncontact SVC payment systems, which are now widely deployed in major US cities and many overseas systems. Of the systems I listed above, most had Cubic as at least a member of the consortium that implemented them, if not as the prime contractor. Oyster, for example, was implemented by Cubic alongside EDS, now a division of HP perhaps best remembered for the political career of its founder Ross Perot.

How do these systems work exactly? Offline systems simplify payment networks in some ways, but also add complexity, which is often apparent in transit systems that combine offline terminals (for example in buses) and online terminals (for example at train platforms). I will walk through a description of the operation of Clipper, with which I am most familiar. The details vary from system to system depending on architectural decisions made during the original system design and modernization efforts that have been performed since, so details vary. For example, some newer systems especially abandon offline operation almost entirely and have even in-vehicle terminals perform online transactions via either public or municipal LTE networks.

A Clipper card can be purchased from a number of vendors, either first-party ticket windows in certain stations or private convenience stores that have opted to participate in the program. These stores can also add value to an existing Clipper card, from cash or a payment card transaction, by entering the value to add into a device very much like a credit card terminal (it is, running custom software as many do) and then tapping the card to it to allow the write operation to complete. Similarly, cards can be purchased or value added via vending machines at stations, which usually require the card to be tapped twice: once to read the current value and determine eligibility to add value (there is a value cap, for example), and a second time to write the added value.

Because these transactions involve writing to the card, the new value is available immediately. It can be spent on fixed terminals like station fare gates, but also on vehicle terminals as in buses. Either way the terminal reads the value from the card, determines eligibility, and writes the new (decreased) value back to the card.

Things get a little bit strange, though, when you consider one of the most common user patterns in the modern era: you can create an account online and associate the card with your account, and then you can add value online. This is convenient, but confronts the offline nature of the system. You add value to the card, but there's no way to write the new value to it.

The solution, or at least partial solution, to this problem looks something like this: fare payment terminals have to receive a streaming log of value-add operations so that they can apply them the next time they are presented with the relevant card. Online systems, like vending machines and fare gates in train stations, find out about online value-adds almost immediately. If you mostly use a train, the operation is almost completely transparent, as you add value online and it is written to your card next time you pass through a fare gate.

For the offline terminals in buses, though, things aren't so smooth. These terminals operate fully offline while the vehicle is on route. At the end of the day, as vehicles are stored in yards, the payment terminals connect to a local area wireless network (traditionally 802.11a). They upload on-vehicle transaction logs for reporting, but also download logs of online transactions. If you add value to a card with a zero (or near zero) value and then try to board a bus, it is likely that you will be rejected: the value-add hasn't been written to the card yet, and the bus terminal hasn't been told about it. The transit operator often sets an expectation of one business day for online value adds to be available if your first trip is an offline terminal.

It may be that Bay Area transit operators are transitioning to online vehicle terminals to address this problem, it wouldn't surprise me as IP connectivity in transit vehicles is becoming the norm for multiple reasons. But, of course, in an environment where all devices are online the value of SVCs as a technology is greatly reduced. At some point the SVC nature of the system becomes more vestigial than anything else, although it can provide valuable fault tolerance.

The case of transit is more complex than just incrementing and decrementing, though. Passes (including automatically "earned" passes in many systems) and transfer discounts between operators can make fare logic surprisingly complex, and that's before considering the many rail systems that charge fare per zone traveled. To accommodate this kind of fare tabulation logic, transit SVCs typically store a history of the most recent transactions and cumulative counters of different types of transactions. This allows the system to compute distance-based fare (by comparing the current and previous transaction for their location), offer transfer discounts (by comparing the last two or three transactions to a table of discounts between operators), and automatically change to passes when they become most economical (by checking registers of accumulated fare per operator per time period).

Put together these programs can make fare calculation very complex, which is one of the advantages of computerized fare collection with usage history: the software can ensure that the fare paid is optimal, in the sense of being the lowest fare the customer is eligible for. Prior to these systems features like transfer discounts often went unused because of the added complexity of presenting a ticket and payment or determining validation procedures between transit agencies.

Even transit agencies are moving away from SVCs as IP connectivity to vehicles becomes more affordable and more common. Centralized systems, while they require network infrastructure, can be more flexible and more user-friendly.

It doesn't look like SVCs have much of a future. Despite being the dream of the '90s, they have gone the way of, well, so many other dreams of the '90s technology industry. By the time the ingredients for SVCs to succeed became widely available, they were somewhat of a solution in search of a problem. Network connectivity was spreading rapidly for other reasons, online processing of payments offered other advantages, there just weren't that many reasons to go the SVC route.

Smart cards are an important part of payments infrastructure today because of the EMV standard, and they continue to have applications in both their traditional form factor and embedded variants. Despite the power available from multi-application cards, MIFARE with its simple cryptographically protected read/write operation is more common in practice. So pour one out for SVCs, or more accurate to the tradition, put $10 on a laundry card, put it in a drawer, and move to a different city.

[1] The topic of electrical interface is actually slightly confusing because the standard describes 5v, 3v, and 1.8v logic levels. Modern cards are nearly always 1.8v, but fully compliant readers need to detect and provide the correct operating voltage to the card. This complexity is one of the factors that has lead to occasional security vulnerabilities in smart cards around supply voltage.

[2] Most payment card terminals query the name of the selected application and display it. Often it is only "VISA" or "MASTERCARD" but a few issuers customize their card loads to brand the application name. Just a bit of trivia.

[3] Outside of certain more niche applications like cardlock fuel cards, which are broadly compatible with payment cards for ease of implementation but don't seem to be interested in making the move to chip-and-whatever.