CCE Journal

Cryptographic Centre of Excellence

TRUE SECURITY
IS THE FREEDOM TO
LIVE DANGEROUSLY.

The Internet is now secure for big business. Using digital certificates and unprecedented standards, we can provide the security and privacy services your business needs to thrive. To find out how we put it all together, visit beTRUSTed.com

beTRUSTed^{SM}

An e-security business of PricewaterhouseCoopers.

PRICE**W**ATERHOUSE**C**OOPERS

Join us. Together we can change the world.

CCE Journal

Cryptographic Centre of Excellence

The PricewaterhouseCoopers Cryptographic Centre of Excellence (CCE) was formed by the company's Global Risk Management Solutions practice in order to unite members from around the globe with unique expertise in cryptography and cryptographic services. This was done to build a network of highly skilled professionals who could assist clients and one another throughout an engagement's lifecycle. By establishing relationships with academic institutions, leading security vendors, cryptographic research organisations and leading cryptographers, we are in a truly unique position to offer our global clients the best solutions for their cryptographic security needs.

Global Risk Management Solutions, part of PricewaterhouseCoopers, has over 7,000 professionals worldwide, many of them industry specialists, and offers a comprehensive identification of risks, whether they are strategic, financial or operational in nature. Our solutions-based risk identification and analysis offers guidance on industry best practices and common training programmes, using state-of-the-art methodologies and tools consistently.

By addressing the changing needs of today's business leaders, we are able to help organisations identify, assess and manage complex issues and risks across the whole enterprise – or within in any part of it – whether they are strategic, financial or operational in nature. We help clients to develop risk management solutions that minimise hazard, resolve uncertainty and maximise opportunity.

Contact Information

Dr. Alastair MacWillson

Partner in PricewaterhouseCoopers London and joint CEO of beTRUSTed within the PricewaterhouseCoopers Global Risk Management Solutions practice.

alastair.macwillson@uk.pwcglobal.com

Geoffrey C. Grabow CISSP, MSc IS

Americas Leader – PricewaterhouseCoopers Cryptographic Centre of Excellence.
Chief Scientist – beTRUSTed.

geoffrey.c.grabow@us.pwcglobal.com

John Velissarios M.Comp.Sci.

EMEA Leader – PricewaterhouseCoopers Cryptographic Centre of Excellence.
Strategy and Business Development - beTRUSTed.

john.velissarios@uk.pwcglobal.com

Stephen G.Wilson BSc. BE Elec (Hons)

Asia Pacific Leader – PricewaterhouseCoopers Cryptographic Centre of Excellence.
Strategy and Business Development - beTRUSTed.

stephen.g.wilson@au.pwcglobal.com

More information and previous editions of the CCE Journal can be found at:

www.pwcglobal.com/cce

The views expressed in this publication are not necessarily the views of PricewaterhouseCoopers.

To SUBSCRIBE to CipherText, our weekly e-mail cryptographic newsletter, go to
http://www.pwcglobal.com/cce

In this Issue

Editor's Soapbox

Trust is not Digital in Nature

by Geoffrey C. Grabow CISSP

Public key systems have eliminated the 'key management' problem, which is of primary concern in symmetric key systems, and have introduced the problem of 'trust management' in its stead. To manage this environment, two of the most commonly used trust models, 'trust hierarchy' and the 'web of trust', have been implemented by various vendors.

The web of trust model spreads trust among the users of the system. Each user can be a sponsor for the validity of other users. This sponsorship permits two people who have a sponsor in common to trust each other. In other words, if Alice trusts Bob, and Claire trusts Bob, and Bob vouches for the identity of Alice and Claire, then Alice and Claire can trust each other. This model gets extended indefinitely such that if Claire then vouches for David, and David vouches for Fran, Alice should be able to trust Fran.

This model assumes a great deal. Trust between people is not 'digital' in nature. It does not directly hold true that Alice can trust Claire 'as much' as she trusts Bob, simply because Bob has trust in Claire. Trust between people is much more analogous to that of a document being photocopied, and the photocopy being copied, etc. After a few iterations through the copying process, the document gets hazy and eventually unusable. The same holds true for trust between people. A more practical example is that while you might trust a friend to drive your car, you would not necessarily loan your car to a friend of a friend.

This is not to say that the technology introduces any sort of problem, rather that this is a social condition that cannot be resolved through the use of cryptography.

Hierarchical trust systems are based on large, single points of trust, where many people all trust the same entity. In the Certificate Authority (CA) model, Alice, Bob, Claire, David, etc, all trust one central entity, such as a bank, who then vouches for the identity of everybody it knows.

The first catch to this is that the CA must 'get to know' each person individually through some sort of 'enrolment process'. That process, the opening of an account, requires the CA to prove its identity to the enrollee as well as the enrollee proving their identity to the CA. That proof then becomes the defining line of the amount of trust that can be placed in that enrollee.

For example, if Alice enrolls with the CA by sending an email message and receiving an email response, this provides a very low level of trust that the CA can actually guarantee Alice's identity to other parties. If Bob walks into the office of the CA with his passport, driver's license, some credit cards, letters of reference, DNA sample, etc, this provides an extremely high level of confidence that this is the genuine Bob. The CA can then feel very confident in declaring that this is really Bob.

This also holds true for times when a user's identity must be revoked. Bob can walk into the CA and indicate that his previous identity was lost or stolen and be reissued a new one after verification. The CA, knowing that this is really Bob, can then post indicate to other users that Bob's previous identity is now untrustworthy.

The second problem with the CA environment is determining what happens if Alice and Bob are enrolled with different CAs. If Alice is enrolled with CA #1, and Bob is enrolled with CA #2, neither Alice nor Bob can verify each other's identity simply because neither trusts the other's CA. To get past this problem, we need to build a hierarchy of CAs. We need a CA above CA #1 and CA #2 that will not vouch for Alice or Bob's identity, but rather will vouch for the identity of CA #1 and of CA #2. This means that CA #1 and CA #2 must each become an enrollee of CA #3. At this point, for Alice to verify Bob's identity, she would:

1. Ask CA #1 to verify Bob

2. CA #1 doesn't know Bob, but CA #1 trusts CA #3

3. CA #1 asks CA #3 to verify Bob

4. CA #3 trusts CA #2

5. CA #2 verifies Bob and returns that information to CA #3

6. CA #3 returns the verification to CA #1

7. CA #1 returns the verification to Alice

This starts to look very similar to the web of trust model where the trust starts to become distributed, but the amount of trust provided in the CA environment is higher if the CA's use significant means of identification during the enrolment process for people as well as other CAs. Additionally, since CAs are businesses or organizations, they can be kept to a higher degree of responsibility than an individual through the use of a Certificate Practice Statement and other legal documents.

So, which model is better? Actually, with the number of mergers and partnerships taking place in the industry, we are likely to see a combination of the two. Any system, which attempts to encompass a large number of companies, people or countries, will probably have a number of CA hierarchies. The CAs will need to be either part of another hierarchy or a member in a web of trust.

Comments on this topic are welcome and may be submitted to the CCE Journal via e-mail using the contact information at the beginning of this issue.

About the author

Geoffrey is the Americas leader of the PricewaterhouseCoopers Cryptographic Centre of Excellence, co-editor of the CCE Journal and Chief Scientist of beTRUSTed.

He can be contacted via e-mail at geoffrey.c.grabow@us.pwcglobal.com

Smartcards

by Dr. Kim R. Wagner,
PricewaterhouseCoopers

The future of the smartcard market has, over the last decade, been presented by more or less the same exponential curve. The point at which the curve starts to take off has consistently been 2-3 years after the current year, regardless of whether that year was 1990 or 2000.

Although we have seen significant growth in the smartcard market over the last years, it is a common perception that smartcards have underachieved compared to their potential. In this article we try to identify the reasons for this, and try to assess in particular whether these reasons are permanent or temporary.

We will discuss the main drivers and the main barriers for smartcards, and provide an outlook for how they might evolve, giving us some insight into the medium term future of smartcards.

Our article does not aim to be a technical introduction to smartcards, and people who are interested in learning more about the technology should refer eg. to Rankl and Effing: *Smart Card Handbook*, John Wiley & Sons, 2000.

What are smartcards?

Smartcards are credit card sized plastic cards with a micro processor chip embedded, such that the card can perform computational tasks and in many ways work like a (very small) computer. The smartcard chip will be *tamper resistant*, meaning that it is possible to store data on the chip which the chip can use in its internal computations, but where it is next to impossible to extract this data in an unauthorised fashion. Such storage space is used eg. for *secret cryptographic keys*, which enable the card to participate in cryptographic protocols. Tamper resistance is a crucial aspect of smartcards, and one that sets them apart from usual PCs.

Often smartcards will have data about their cardholder (such as name and credit card number) stored on them, or data such as electronic tickets for eg. rail systems or loyalty tokens for retailers, or even digital currency.

Smartcards distinguish themselves from ordinary computers in several ways. Firstly, they are much smaller and slower, they are, as we already pointed out, tamper resistant, and they do not (usually) have any resident power source, keypad, display, or clock. For all these elements they are reliant on an *interface device* (IFD), such as a point of sale device, a PC with a card reader, a mobile telephone with a smartcard slot, or an ATM.

Although there are still smartcards which do not have any inherent cryptographic capability (so-called memory cards, mainly used for telephone payment), we shall restrict our discussion purely to cryptographic smartcards.

We shall however, include the SIM cards of mobile telephones in our discussion, since they use the same chip technology as ordinary smartcards.

The Case for Smartcards

The case for smartcards is often presented purely in technical terms. In this article we will try also to identify clearly the business cases behind the technical arguments.

It is generally agreed that smartcards provide **security, convenience and mobility** for a relatively low cost to financial and other transactions.

The business drivers behind the technological aspects are associated with the benefits of being able to provide an electronic business environment with *increased trust* and *reduced fraud*, in addition to added value to customers in terms of convenience and mobility.

Security

Smartcards provide **security**, because they are able to *authenticate themselves*. This enables a smartcard issuer to give an identity to a smartcard and have some confidence that in subsequent transactions with that card, it will be possible to identify the card, and to verify that it is not a copy.

The capacity to support authentication in this sense, as having an identity which is *easy to validate* for relevant parties, but *difficult to copy*, is a property also found for instance in human beings, high quality bank notes and passports. Smartcards further have the commercially desirable property that their identity is *cheap to produce*, and for electronic commerce the crucial property that they can demonstrate their identity *remotely*.

Magnetic stripe credit or debit cards and personal signatures are examples of objects that are less able to support such reliable authentication. It is relatively easy to manufacture copies of magnetic stripe cards, which are difficult to distinguish from the originals.

The big difference between smartcards on the one hand and bank notes and passports on the other is that the authentication mechanism for smartcards is electronic and thus works remotely, eg. from a home PC over the internet or from an un-manned ATM to a bank.

Credit and debit card issuers suffer losses due to fraud in the hundreds of millions of dollars every year, mainly because magnetic stripe cards do not support remote authentication very well, and are easy to copy. These losses are expected to drop dramatically when credit and debit transactions can be done using the strong authentication supported by smartcards. Visa, MasterCard, and American Express are pursuing their common standard, EMV, as the means of migrating magnetic stripe cards to smartcards.

The big advantage of a card being able to authenticate itself is that it supports *two-factor authentication* of the cardholder. In order to perform a transaction the cardholder needs not only to know a PIN (which enables the smartcard or an underlying host application login), but also to possess their particular smartcard at the time of transaction. Therefore, it is not enough to spy on the communication line (since that does not bring you into possession of the card), and it is not enough to steal the smartcard (since that does not tell you the PIN – unless the PIN has been written on the smartcard).

Even though smartcards provide a big step in supporting secure remote financial transactions, it is still necessary for the cardholders to be security conscious. For example, present day smartcards do not have displays or keypads, which means that the cardholder will have to rely on external devices to transmit transaction details to the card, and to display relevant information from the card. If such an external device is compromised, it can engage in one transaction with the card while giving the impression via its display that it is engaging in a completely different one.

Security concerns like this may lead to smartcards having their separate display (and possibly keypad), but it seems more likely that mobile telephones or PDAs can be used as trusted personal devices for interacting with the smartcard.

The issue of trusted devices has traditionally been dealt with by the credit card organisations by type approval, and the issue of merchant side fraud by the fraud control measures exerted over merchants by their acquirers and the credit card organisations themselves.

As the Internet allows merchant side fraud on a larger scale and more easily across national

borders, the issue is bound to grow in importance. So far however, consumer side fraud accounts for the vast majority of reported fraud cases.

Smartcards can provide security in other ways than by supporting authentication. Since they have a general cryptographic capability, they can be used for instance for generation of cryptographic keys and for performing cryptographic tasks with the keys generated on them, such as digital signing and decryption.

Having a cryptographic key generated on the card, and never leaving the card, is important since it supports *non-repudiation*. Non-repudiation is the property of a signing party being held to their signature. In other words, if you use a cryptographic key to sign a message, and pass this message on to somebody else, then this other party can argue that it was *you* who signed the message. Obviously, non-repudiation is a very important property in any area of digital signatures where those signatures have real value.

Several things need to be in place for non-repudiation to be feasible. If cryptographic keys used for signing are resident in software, for example, it is easier to argue that they some-how may have been compromised and divulged to a third party, who could then have used them to produce the signature in question. If the keys are generated anywhere else than on the hardware module where they will eventually be used, a compromise of the generating party or of the line of communication between this party and the recipient (the signer) would have the same effect. Therefore, for non-repudiation, the most desirable architecture is one where signing keys are generated, stored and used in the same tamper resistant hardware.

In the absence of expensive hardware security modules, smartcards fill this requirement to near perfection. Near perfection and not perfection, because they are so small that they are much slower and not quite as tamper resistant as larger expensive hardware secure modules. Furthermore, random number generation, which is at the heart of generation of cryptographic keys, is not as good on smartcards as on larger hardware based units. Generally speaking, however, for the price of a smartcard you get a very attractive solution in this respect.

Technical solutions are not enough to carry a non-repudiation argument through a court of law. You have to argue that no third party had access to both components of the two-factor authentication provided *viz* the PIN and the smartcard.

For higher grade solutions (often on the server side rather than the client side) smartcards are also often used simply as intelligent storage devices, interfacing with a hardware secure module. In this case a cryptographic key is exported onto several smartcards, in a form such that all or a significant number of the smartcards are needed to recreate it. Here it is the tamper resistance of the smartcard which is the key feature.

For solutions where the PIN component of the two-factor authentication is not considered secure enough, some smartcards are now storing, or in one case even equipped to capture and recognise fingerprints, but with this comes the difficulty of providing a supporting infrastructure and example of preventing using captured fingerprints. Clearly the biometric area and its intersection with smartcards is developing quickly and new solutions are marketed frequently addressing the issue of providing reliable card holder and not just card verification.

> **Technical solutions are not enough to carry a non-repudiation argument through a court of law.**

Convenience

Smartcards can provide **convenience** in a number of ways. A multi-application smartcard can replace a train pass, a gym pass, a health insurance card, supermarket loyalty card, company credit card, private credit card, bank card, company access token and an airline ticket all within one smartcard. The smartcard would have a number of applications, each supporting one or more of the usages mentioned above.

The possibility of individual personalisation of smartcards provides convenience as well, for instance enabling you to store your profile on your card, so that you don't have to repeatedly enter those details when purchasing goods over the internet or TV, say.

Furthermore the chip allows a dynamic evolution of the card. You may be able to

download entirely new applications to your card over the web, or your bank may be able to update your card in the field, in response to changed risk management or application upgrades.

The convenience of smartcards is underpinned by the adherence to standards such as ISO/IEC 7816 parts 1-4 (or higher), ensuring that the smartcard has a standard physical format, position of contacts, etc. In the areas where there are no standards, or where there are several competing ones, such as in the cases of card operating systems and e-purses, convenience is hampered in the short term.

Mobility

Smartcards provide **mobility**, because you can take the smartcard with its applications, your profile and its ability to authenticate itself with you wherever you go, or, as pointed out above, you can perform the authentication remotely. This means that you can bring this key to establishing trust anywhere, say to an internet cafe, your mobile telephone or the set-top box of a friend's TV. From all those points you can use your card to access or pay for or receive products or services.

It is only coupled with the other advantages of the smartcard, security and convenience, that this becomes a real advantage over other solutions, say, based on user id and password. For most applications like Pay TV the security requirements are such that a hardware component for authentication is mandatory. Therefore TV service providers are faced with the choice of either providing a stationary set-top box or having a mobile smartcard solution. As of now, smartcards (and equivalent USB tokens, iButtons etc) are the only means of providing a truly mobile device which supports strong authentication.

Again, standards are crucial to enable true mobility, and we are seeing a number of initiatives in this area (eg. FINREAD for terminals and various GSM and set-top box standards).

Electronic Identity

In summary, the technological properties of smartcards such as tamper resistance and cryptographic capabilities combined with standardisation lead to security, convenience and mobility, which combine to offer an electronic identity, as illustrated by Figure 1.

Specific Industry Drivers

Smartcards started out life as telephone cards in France and moved into the banking sector with electronic purses by Mondex, Proton and others. By the emergence of multi-application operating systems and larger memory capacity on smart-card chips, smartcards can now be used in an ever growing number of areas. Figure 2, opposite, summarises the current main drivers in the smartcard arena, and illustrates how specific business interests lead to developments in this area, including the emergence of new issues such as how to brand a multi-application card.

Barriers – or 'What's gone wrong?'

Looking at the drivers in Figure 2, it is perhaps difficult to see why smartcards have not been a greater success than they have. The devil is as usual in the details. Looking at the specific examples of smartcard rollouts and trials all over the world, it is clear that there are several forces that have acted against the immediate success of smartcards.

Limited capacity

There is a threshold for non-volatile memory and ROM below which multi-application smartcards may be theoretically feasible but practically uninteresting. This threshold is around 16 K bytes for each category (ROM and EEPROM) of memory, and below this it gets difficult to have more than one or two interesting applications on one card.

This threshold has only recently been reached. There is therefore a backlog of smartcard applications to be written and implemented for
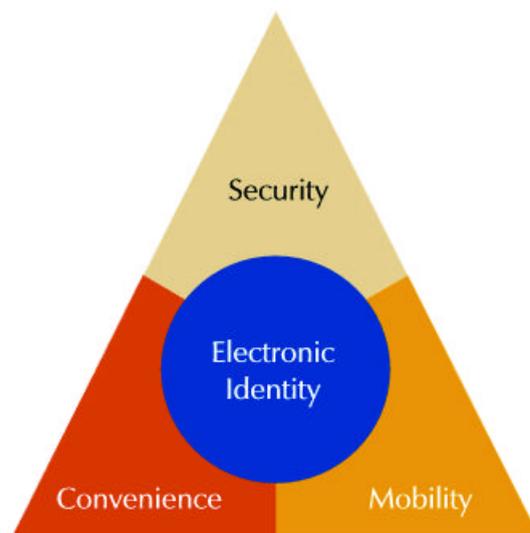

Figure 1 The main advantages of smartcards

versatile, standards based smartcards	public transport initiatives	← make ticket handling more effective, reduce fraud
government health and/or id card	← ease govt administration reduce social security fraud
mobile standards (GSM)	← enable secure interoperable mobile communications and transactions
EMV for credit/debit cards	← reduce credit and debit card fraud
PKI solutions (PKCS#11)	← increase security of networks and corporate computing systems
loyalty card	← enable data mining, ease admin of loyalty card
e-cash/purse	← enable electronic micro payments, reduce merchants cash handling costs & risks
→ extensive back-office systems for on-line and for off-line processing
→ complex card and application management process
→ problematic PIN handling
→ tricky branding

Figure 2

multi-application platforms such as MULTOS or JavaCard. Since the 32K byte limit is already being passed, this specific restriction is more or less now a thing of the past, although it is easily predictable that as capacity grows so will the size of applications.

	start 1999	start 2001
ROM	32K	135K
E2	16K	64K
RAM	0.5K	5K
bit	8	16

Table 1 – Development in chip capacity

Single application smartcards are no good!

Smartcards have received their share of poor publicity from trials with single application cards, such as Visa's e-cash or earlier versions of the Mondex e-purse. The main experience from a number of e-cash trials around the world is that the general public in collaboration with merchants without much incentive can be very unforgiving to a technology that brings them marginal benefits in return for marginal hassle.

Without either (and preferably both) multi-application cards and a built-up infrastructure catering for smartcards in home PCs, at merchants and at banks and ATMs, the benefits of smartcards do not seem to break the threshold required for large scale public uptake. The only time where single application smartcards could be feasible is as a component of a fully supported PC infrastructure if, say, the major PC vendors where to ship ready-made smartcards and readers with home PCs.

Not so secure after all?

Smartcards 5-10 years ago were sometimes said to be tamper proof when they were anything but. Later on the term has been prudently revised to 'tamper resistant', accepting the

thesis that with the right amount of skill, time and equipment in the hands of an interested party, it would be foolhardy to reject the possibility that a standard smartcard could be compromised. The trick now lies in making it sufficiently difficult that compromising smart-cards is more expensive than the feasible gain, and preferably the exclusive realm of large government agencies and the like.

Smartcards received some, often belated, bad publicity from having their secrets exposed. The hardware technology in smartcard manufacturing has moved ahead by leaps and bounds, and over the last 4-5 years there have been no reported successful cases of extracting secrets from current smartcards, except one.

The differential power analysis (DPA) attack reported in the press in 1998 sent shock waves through the smartcard community. Without penetrating the tamper resistance of the smart-card it was possible to extract the keys used for public key operations that were supposed to take place in the depths of the tamper resistant parts of the chip. The reason was that the electric behaviour of the chip depended on the binary digits in the key it was using in its cryptographic operation. Using advanced measure-ment equipment the small differences in electric profile over very short time spans could be used for this purpose, revealing enough of the key to eventu-ally deduce the entire key.

The differential power analysis (DPA) attack reported in the press in 1998 sent shock waves through the smartcard community.

The countermeasures to DPA have not been widely publicised in detail but most, if not all, smartcard vendors now assert the DPA resistance of their platforms. Furthermore, evaluations taking place according to the ITSEC, FIPS-140 or Common Criteria frame-works at a sufficiently high level will take DPA resistance into account, and depending on the evaluation level of the individual product or platform provide users some degree of assurance. Most assurance must come however, from the fact that DPA is still an advanced and costly attack to mount.

Costly and complicated infrastructure

Even for what is probably the biggest potential market for smartcards, viz credit and debit cards, the requirements for supporting infrastructures put a significant damper on the migration from magnetic stripe cards to smartcards.

There is already a credit and debit standard, EMV, for smartcards, backed by Visa, MasterCard and American Express. EMV uses the capabilities of smartcards to address most of the current methods of fraud. There are already many smart-card applications which implement this standard. The complication is however, that establishing the required infrastructure in terms of smartcard readers and back-office systems is quite costly.

In order to migrate a merchant from magnetic stripe cards to smartcards, all the point of sale devices of the merchants have to migrate to ones that can read smartcards as well as magnetic stripe cards. The staff must then be instructed to use the card reader part for all cards that have a chip embedded, and only use the swipe reader for cards without a chip and for cards where the chip does not work. We can assume that the card issuers will fund the migration of their customer population to smartcards (like Barclays and American Express Blue Card have done with their cards in the UK) It is a remaining issue, how to fund the migration of point of sale terminals.

As it stands, point of sale terminals are sometimes provided by the Acquirer as part of a package offered to the retailer, and sometimes owned by the retailer themselves. Most often retailers who own their terminals will be larger ones who will have the resources to manage that part of their business them-selves, but this is not to say that all large retailers go for this option. Ownership however, plays a big role in determining the likelihood of a terminal being migrated to chip. If a point of sale terminal is owned by the Acquirer and the Acquirer has a clear business case for upgrading the terminal to chip, then it is fairly likely that the terminal will be upgraded. However, as the credit and debit association rules specify, liability stays with the Issuer unless it shifts to the Acquirer under some specific circumstances. The limited nature of those circumstances means that generally speaking, Acquirers have less of an incentive to combat fraud than Issuers.

As the Issuers will often pass the liability on to the merchants, depending on the kind of transaction, merchants who do cardholder-not-present transactions will have a significant incentive to combat fraud, but ironically, EMV is not specifically tailored to Internet or MOTO (mail order/telephone order) transactions.

In the UK credit card fraud in general, and 'skimming' (capturing magnetic stripe information and copying it onto blank credit cards) in particular has had an alarming growth rate. For this reason the Issuers have had sufficient incentive and cohesion to (under APACS) set up a fund of 25 million GBP which has been used to encourage Acquirers to migrate terminals. The scheme was designed so that early entries were given the greatest rewards, and the scheme has had some effect. APACS state that there are now 7.5 million chip cards in the UK market, compared to 4 million at the end of 1999, and 115,000 chip terminals have been deployed, a figure expected to grow to 500,000 by the end of 2003.

Similar schemes do not exist in the USA, where there is less recognised credit and debit card fraud (because all transactions are online), and where the Issuers (and Acquirers) are less cohesive. In such cases, and for point of sale terminals that are owned by the retailers themselves, the path from the entities that would benefit from smartcards (mainly Issuers) to those who would have to instigate the necessary change (individual retailers or acquirers) is more indirect, and therefore offers more resistance.

Since Internet credit and debit card transactions are increasing rapidly and since credit and debit card fraud are especially rampant in this environment, the payment associations have paid a lot of attention to combating fraud on the Internet.

Two initiatives, Chip Electronic Commerce and 3D Secure are under development. CEC is essentially a combination of SET and EMV, using smartcards for authentication over the Internet. 3D Secure is not available yet, and it is therefore not clear if smartcards will play a central role. What is clear is that there is significant cost involved in mandating consumers using smartcards from their home PC in the near future. The cost is not just in providing readers, which are getting cheaper all the time, but in support for customers who have problems installing their software drivers for their smartcard reader etc.

Complicated Card Management

Concerning infrastructure, the lifecycle of a smartcard is quite complicated and a number of vendors have to collaborate in order to achieve anything like a streamlined manufacturing process. Take the credit and debit standard EMV, for example, and a typical smartcard application implementing that.

Data which goes on the smartcard for this single application comes from:

- the *card issuer* (customer name, account number, risk management parameters, signed customer key etc);
- *application provider* (application code, layout);
- *credit card association* (root's public key);
- *platform provider* (keys to control application load);
- *loading system* (processed data and possibly session keys); and
- *PIN generation system.*

Needless to say, when more than one application is involved, it will often be attractive to load the applications in the same process. However, the data flow gets even more complicated in those circumstances – especially if and when something goes wrong in the middle of the processing of a card.

Since this is a fairly new business it is still developing rapidly, and standardisation efforts are under way concerning formatting the kinds of personalisation data mentioned above. This should make the process easier for issuers and everybody else involved, and especially making it easier to carry lessons from one smartcard project to the next.

Who'll give end-users card readers?

Getting PC vendors to sell PCs with smartcard readers requires solid public demand which does not exist at present. The only reason solid public demand would arise would be if credit card companies, (online) banks or merchants would have better liability terms or prices for customers who use smartcards in their electronic transactions. Given the intense competition between the internet players at present it seems unlikely that such a preferential treatment of what initially would be a small minority would be tenable. Even the seemingly mundane step of a bank giving away card readers to its Internet customers would be fraught with risk, not by the direct cost of the card reader, but rather in terms of the customers not knowing how to connect the card reader (or not having a spare serial port/floppy drive/pc card/USB), or install it, and the increase in customer service costs.

Who owns the card?

Even assuming that there were an adequate infrastructure in place in terms of card readers,

PC software, PIN pads, POS terminals, network connections and back-office systems, there are still complications with eg. branding issues of multi-application smartcards. There are several models for multi-application smartcards in terms of who issues them and who controls which applications and data reside on the cards.

One model has a generic company as issuer, and others then being free to download software onto the card. This would mean that cards would be branded generically, but not by each individual organisation that had their application on the cards. Only a certain category of application providers can be expected to accept this kind of platform for their applications.

Another model has a designated card Issuer which itself loads some applications onto the card, and then invites other organisations to load their applications onto the card as well. This permits specific branding, but if it is still up to the cardholder precisely which applications reside on the card, branding is still an issue.

The third model has the card issuer deciding up front which applications go on the card and leaves no flexibility to the cardholder to put other applications on after issuance. In this case branding is easier, but a lot of the potential flexibility of the multi-application smartcard has been lost.

With current technology it seems impossible to retain the flexibility of allowing cardholders to download the applications they want onto their smartcards, and at the same time allow all parties who have applications on the users card to brand those cards. At the same time, certain applications, such as typically e-purse applications or even credit or debit applications will have security restrictions associated with them, which requires strict control on which other applications reside on the card.
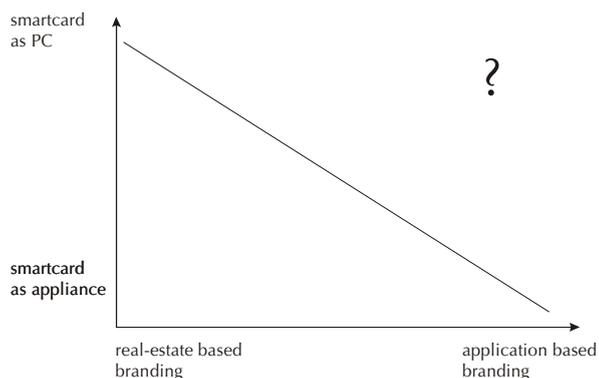

Figure 3

Somewhat independently of the ownership discussion, the fees structure of applications versus cards is also unsettled. One question that is continuously bounced around is that of a fees structure for a smartcard and its applications. Will fees be flat or will they somehow be linked to usage? No question, application providers and providers of back-office solutions would prefer licence fees that depend on volume and is somehow calculated on a per-transaction basis. However, it seems very difficult at present for them to prevail with this payment structure as the owners of the 'real estate', viz the card issuers insist on a flat fee per application, and a yearly fee (or flat fee) for back-office systems. Where you would think that first-to-market desire would enable per-transaction fees, this has not yet been the case.

Outlook

In summary, looking at the barriers to 'take-up' of smartcards, we conclude that none of them are long term barriers:

- **limited capacity:** already, smartcards are getting to the capacity that allows for a number of interesting applications, and their processing speed has grown to acceptable levels.

- **single application smartcards are no good:** not a problem anymore, since smartcards now support multiple applications.

- **not so secure after all:** with the maturing of the smartcard industry a reasonable level has been reached where the industry seems to keep slightly ahead of the attempts to circumvent smartcard security.

- **costly and complicated infrastructure:** with MasterCard, Visa and American Express committed to rolling out credit and debit cards on chip worldwide over the next decade, a large part of the necessary infrastructure is going to be established anyway, and other applications will be able to piggyback on this.

- **complicated card management:** as the industry gets used to launching multi-application smartcards, businesses spring up around providing smartcard management services. Furthermore, standards emerge, such as those promoted by Global Platform. In the future, rather than being a new and complex business to issue multi-application smartcards, it will be a matter of choosing between a few well established solutions. There is no doubt either, that this development will take time, but the

consolidation that we see presently in the smartcard market should help push this development over the next decade.

- **who'll give end-users card-readers for their PCs**: as smartcard solutions and infra-structure emerge which does not depend on users having card-readers at home, the overhead of equipping end-users with card-readers will diminish. At the same time the potential value-add will increase, as it becomes possible to utilise or tap into existing solutions. Already Freeserve is providing a mousepad with a built-in card-reader. Compared to the other steps, viz. credit and debit on chip, the card-reader in every home will undoubtedly lag a couple of years behind, though.

- **who owns the card**: two distinct models are likely to crystallise: the first one will be cards issued by banks, and here the bank will likely retain a high degree of control over the applications on the card. The second will be cards issued by others, eg. PC manufacturers, supermarkets etc, where the cards will be more 'open' in the sense of allowing download of other applications, eg. from a store or the Internet. At the same time we will see branding shift, especially for the smartcards which are not issued by banks, to utilise PC, mobile, PDA and point-of-sale interfaces rather than, or in supplement to the card surface. This will allow for a satisfactory branding of smartcards where applications can be dynamically loaded and deleted.

Concerning the main drivers for smartcards, clearly the dominant ones are the EMV initiative by MasterCard, Visa and American Express regarding credit and debit cards, and the mobile industry.

There are more than 2 billion credit and debit cards in the world today, and the gradual migration of those to smartcards will provide the market with a continuous supply of custom, in addition to providing a much sought after infrastructure.

Mobile telephones use smartcard chips for their SIM cards, and there is no question that the two strands will have a fruitful interaction over the next years, with smartcard technology and applications supporting mobile authentication and commerce.

Long term it is difficult to forecast how the mobile, PDA and smartcard products will interact and/or merge. In principle there is nothing preventing mobile telephones being used for all the applications we have discussed

for smartcards. This would involve entirely new 'card-readers' at retailers, which performed the transaction with the mobile phone instead of with a smartcard. It seems more likely however, that mobile phones with a smartcard slot, possibly in combination with PDA and Internet functionality will become more popular. It seems that in the short to medium term there are enough uses for the smartcard as a credit card sized piece of plastic, that it will not be reduced to a SIM card.

About the author

Dr. Kim Wagner is a Senior Manager at PricewaterhouseCoopers and is in the Strategy and Business Development team at beTRUSTed.

He can be contacted via e-mail at kim.wagner@uk.pwcglobal.com

Making Sense of your Authentication Options in E-Business

by Stephen Wilson, Director, beTRUSTed, Asia Pacific

When it comes to authentication, e-business planners and implementers are faced with a bewildering array of options. Alternatives include the traditional user name and password, various 'two factor' authentication tokens, digital certificates and public key infrastructure (PKI), smartcards, and an ever-growing range of biometric methods. In the light touch regulatory environment of the US, the European Union, Australia and elsewhere, the onus is on electronic service providers to select and implement authentication technologies that are fit for purpose. A risk analysis should be performed on the types of transactions to be undertaken, and authentication measures agreed upon that are commensurate with the potential for fraud, impersonation and identity theft. But where does one start such an analysis in practice?

This article surveys the major authentication methods available to e-business implementers today, and characterises their qualities and relative strengths. There is no one-size-fits-all authentication solution, and the paper aims to provide practical guidance in support of the specific risk assessments that will be needed case-by-case.

The objectives of authentication

It is often put simply that in e-business, authentication means that you know who you're dealing with. Authentication is inevitably cited as one of the four or five 'pillars of security' (the others being integrity, non-repudiation, confidentiality and, sometimes, availability).

To be a little more precise, let's examine the functional definition of authentication adopted by the Asia Pacific Economic Co-operation (APEC) E-Security Task Group; namely the *means by which the recipient of a transaction or message can make an assessment as to whether to accept or reject that transaction*.[1]

Note that this definition does not have identity as an essential element, let alone the complex notion of 'trust'. Identity and trust all too frequently complicate discussions around authentication. Of course, personal identity is important in many cases, but it should not be enshrined in the definition of authentication. Rather, the fundamental issue is one's *capacity to act* in the transaction at hand. Depending on the application, this may have more to do with credentials, qualifications, memberships and account status, than identity *per se*, especially in business transactions.

---

[1] See www.apii.or.kr/apec/atwg/preatg.html

Access control versus Signature

There are two important facets of authentication. The first is *access control*, which is the management of 'who gets through the door'. Once through, it might not matter much to the service provider what their users do. A user may be accessing information-only services, for free, or they may have pre-paid. In these cases, no lasting relationship is involved between the user and the service provider.

On the other hand, it may be important to track what the users get up to once they're through the door. For this we may need a *signature*. If the service provider wants to maintain a relationship with the user, perhaps in the form of a legal contract, or if they simply wish to track and audit what the user does in the system, then they need to tie the user to their electronic transactions.

Tying a user to their actions can be done a number of ways. In the simplest case, where conventional access controls are in force, a user may be linked indirectly to their on-line actions via event logs that record what they did during a session. The granularity of the event logs and their designer's decisions as to which events are of interest may impact the certainty with which the user can later be identified with their purported actions. And the quality of the identification may erode over time, especially if the logs are not well maintained or are open to abuse or tampering. To make a robust identification, IT forensics and expert opinion may be necessary to make a case from what amounts to circumstantial evidence.

A more robust way to tie someone to their on-line actions is to have them actively apply an *electronic signature* to all key messages or transactions involved during the session.

Some may claim that a particular form of electronic signature – a PKI-based digital signature for instance – brings special abilities to 'legally bind' a user to their on-line actions. But really there are no hard and fast rules about such a legal binding. A user can be bound in a number of ways, and the effectiveness of different approaches boils down to the quality and weight of the evidence linking one's identity to one's purported actions, as we shall see now, looking closely at the idea of non-repudiation.

Unpacking 'non-repudiation'

Non-repudiation is frequently held out as a *unique* benefit of digital signatures and PKI. It is true that the unique association of a private key with the matching public key provides very elegant evidence of origin. Yet non-repudiation is more a legal construct than a technological feature. It cannot be emphatically attributed to PKI and denied for other authenticators.

Non-repudiation is really all about the degree of difficulty you are likely to have if you attempt to argue that you did not sign a given document. Such an argument will factor in the technical properties of authentication and integrity. If significant doubt can be cast over either the origin or the integrity of a signed document, then it is more likely to be repudiable.

As Adrian McCullagh and Professor Bill Caelli point out in their article *Non-repudiation in the Digital Environment*[2], there is nothing stopping someone from flatly denying that a given document was signed by them. The real question is: *What is the likelihood that they can make their case?* In this regard, there is no technological monopoly over non-repudiation.

Consider today's EFTPOS system with its symmetric cryptography. Plainly, it would be very difficult to arbitrarily repudiate a regular EFTPOS transaction, despite the fact that it employs no PKI. Equally, lawyers agree that digital certificate subscribers can and will successfully repudiate transactions made wrongly in their name in the event that their private key is stolen[3].

So digital signatures and PKI cannot guarantee non-repudiation. Yet this technology does bring special advantages when it comes to establishing the likely origins of a document or message. The main issue is evidentiary weight. In contrast to PKI, with many authentication methods there is only circumstantial evidence of the origin of a document.

Take SSL for example, which notoriously is claimed by many PKI vendors to not offer 'non-repudiation'. A web client connecting over SSL is authenticated via a public key handshake at the start of the secure session. But individual transactions sent during a session are not signed as part of the SSL protocol, and so, when decrypted, stored at the server and perhaps passed on, carry no lasting direct

---

[2] Published in the Journal of the PricewaterhouseCoopers Cryptographic Centre of Excellence, Issue 2, 2000; see www.pwcglobal.com/cce.

[3] Whether or not the user may be found negligent in respect of the theft of their private key is another matter; the point is that there is nothing intrinsic to PKI that prevents repudiation.

evidence of their origin. Such transactions can certainly be ascribed to the client, but proving their origin (especially if the transaction data has become separated from the SSL server) typically entails significant information security forensics, winding back network events, and compiling circumstantial evidence.

In contrast, because a public key-based digital signature persists with the document, making the case for the origin of a signed document tends to be easier, cheaper and less risky.

Available authentication solutions

In most jurisdictions, the designers of electronic business systems have a wide choice of authentication technologies available to them. Even if electronic signature prescriptive legislation applies, there is likely to be many options for access control purposes.

Shared secret (password)

There are several approaches to password or 'shared secret' authentication, so called because they all rely on the registered user being able to furnish a piece of secret information, which should only be known to them and to the service which they seek to access. It is noteworthy that for all shared secret methods:

- they work in closed groups where the user is already known to the service (so that they may be registered and equipped with their shared secret).

- they provide no signature function but rather are for access control alone.

Password

The simplest, most widespread authentication solution relies on the user being registered with a static password which is selected so as to be difficult for anyone else to guess. Properly managed – that is, selected using some reasonably random process, and regularly changed or 'rolled over' – passwords or 'pass-phrases' can be very difficult to guess and hence secure enough for many e-business access control applications.

Passwords present both practical and fundamental limitations, as follows:

- Sound management procedures invariably make passwords hard to use, causing users to fall back on readily guessed phrases related to their personal circumstances.

- More fundamentally, no matter how well a password is managed, it cannot provide an

electronic signature, and so only provides circumstantial evidence to identify the originator of an electronic transaction; while adequate for low value transactions, circumstantial evidence can present risks in high value e-business.

- In the event that a password is guessed – or stolen, if it has been carelessly recorded somewhere – this may occur *without the user being aware of it*. Any significant delay in the detection of identity theft can expose the user to greater levels of fraud and losses.

On its own, a secret password is referred to as 'single factor' authentication, because it is based on the one factor – something you know. The problem of identity theft in a shared secret system can be mitigated through 'two factor' authentication, which relies not only on the thing that you know but also something you have – typically a hand-held electronic token.

There are two important approaches to two-factor shared secret authentication.

One-time PIN

A one-time PIN token contains a pseudo-random number generator, which generates and displays a new series of digits at a fixed time interval, usually about half a minute. Before being issued, the token is synchronised to a similar device located on the server, so that any time later, the two devices are expected to show the same random number. The one-time PIN is used in conjunction with a conventional password. At a log-in prompt, the user enters their name, their password, and the number showing on the token at the time. If the random number matches the value expected by the server, then the user is authenticated.

The one-time PIN has become popular in recent years, especially for remote access in large companies of mobile information workers. The devices bring a certain administrative burden, and can suffer from occasional failure if a token's clock loses synchrony with the server. It bears repeating that in a shared secret approach, the one-time PIN provides no electronic signature.

Challenge-Response calculator

The challenge-response calculator is based on asymmetric cryptography, built into a circuit in the device. Each user is thereby issued with a fixed private key linked to their token. The matching public keys are held on a central database. At the log-in prompt, the user first enters their name and a conventional password.

The server then generates a random number and sends it to the user, who types it into their calculator. The device processes the number using the private key and displays a result, which is entered into the log-in screen. If the server can reverse the process using the public key from the user database, and retrieve the original random number, then the user is authenticated.

This device is a little more robust in practice than the one-time PIN because there is no time sensitivity, but the total costs of the two approaches, including administrative overhead, are comparable. The challenge-response device does have the advantage of generally coming with a conventional calculator included.

Public Key Infrastructure (PKI)

Public Key Technology has emerged in recent years as one of the most powerful forms of electronic authentication. It is based on public key or 'asymmetric' cryptography, whereby users create their own unique digital signatures using a special pair of mathematical keys. Each user is equipped with a unique private key which is kept secure and confidential, and a matching public key. Copies of the public key are made widely available to anyone the user wishes to communicate with.

A digital signature is created by encrypting a given document or transaction with the private key. Any recipient of the document can use the matching public key to decrypt and verify the digital signature. Proof of ownership of a key pair is usually conveyed by *public key (or digital) certificates* issued by trusted third parties or 'Certification Authorities'. Naturally, these algorithmic operations are automated in the sender's and receiver's application software.

We use the term PKI here to mean a formal management structure wherein one or more Certification Authorities issue public key certificates to users. PKI encompasses the technologies and standards for registering users and creating certificates, the policies which govern the registration of users and the application of their certificates, and the audit of Certification Authorities. A PKI can be implemented on a small scale within an enterprise or trading community, or on a large scale, across a whole industry sector or an entire country.

> **Public Key Technology has emerged in recent years as one of the most powerful forms of electronic authentication.**

There are several ways to deploy PKI-based authentication.

A simple distinction is often made between 'soft certificates' where the certificate – or more correctly the private key – is stored on a computer disk, and 'hard certificates' where the private key is held in a secure token. Soft certificates are characteristically less secure than hard certificates, for it is difficult to prevent theft of a private key from magnetic storage. However, it is best to break down the classification a little further. There are multiple soft certificate options, offering varying degrees of protection, and there are two important types of private key token, each with distinct benefits.

Browser-based soft certificates

The most common commercial PKIs today involve browsers or commercial e-mail clients which store the user's private key on the hard disk of their PC. In almost all cases, the client software makes use of a standard private key store built into the operating system.

To date, the portability of private keys in this environment has been poor, and is a major cause of user dissatisfaction.

But the most serious limitation of browser-based keys is their vulnerability to un-apprehended identity theft. If they can gain physical access to your PC, the only thing standing between an attacker and your private key is, at best, a password. Worse, if your machine is on the Internet, it is a simple matter for malicious code – a virus or worm – to extract the private key from the standard store. For example in 1999 the Caligula Trojan Horse was reported, which affected the PGP secure email product and was said to extract private keys and transmit them back to the attacker.

Server-based soft certificates

With the aim of improving portability of soft certificates, several PKI vendors now offer a 'roaming' key management solution where the user's private key is stored centrally on a secure server, and is transmitted to the local PC as and when it is needed. The user first authenticates himself or herself to the server by a shared secret (preferably a two-factor token). An applet is then downloaded to the PC, which transfers the private key in encrypted form and activates

it, after which it is used in the same way as a regular local key. At the end of the session, the applet securely destroys the key.

The quality of the server-based soft certificate solution depends largely on the care with which the software is designed to securely transmit and later destroy the private key. Even taking this for granted, there may remain an in-principle objection to the method, in that it involves maintaining a copy of one's private key. This is traditionally presumed to weaken non-repudiation. Yet as discussed above, non-repudiation is not black and white, and should be treated in terms of evidentiary weight. If the solution is well designed, and the entity minding the private key is well trusted, then the server-based soft certificate probably provides the same (if not greater) evidentiary weight as a browser-based soft certificate.

Fat client-based soft certificates

While fat client software is not currently fashionable, it does provide potential benefits in protecting a locally stored private key. The main practical weakness of a browser-based private key is that anyone with an operating system manual can figure out exactly where the private keys are stored. But if the keys are managed by custom fat client software, they can be stored in a proprietary file structure, known only to the software developers. The method is by no means perfect, as it is still susceptible to a social engineering attack on the developer. But it does make it vastly more difficult in practice for private keys to be stolen on-line. The fat client approach has been implemented in the Australian Tax Office's eTax on-line personal income tax package.[4]

Smartcards

Given the fundamental vulnerability of any disk-based private key storage, the more common response to the risk of identity theft in PKI is to migrate to tamper resistant hardware media – classically the smartcard. The newer cryptographic smartcards have enough computing power for digital signature processing to be done on the card itself. This means the private key never has to leave the card, where it would be vulnerable to copying. The more sophisticated smartcards can also generate the user's key pair, meaning that the private key is never exposed at any time in its lifecycle. Notwithstanding the theoretical possibility for future cryptanalytical break-throughs[5], it is essentially impossible for a diligent smartcard user to lose control of their private key without knowing it.

Thus using smartcards to store and deploy private keys ensures the highest evidentiary weight. For now however, the cost of the requisite high performance cards, plus the card readers, has proven prohibitive in all but the most risky types of transactions.

Universal Serial Bus (USB) Dongle

More recently, the same cryptographic chips contained in smartcards have been re-packaged in a more utilitarian form – a token which fits the Universal Serial Bus (USB) port found today in most PCs and laptops. The so-called USB dongle thus requires no special reader and can be used as is with a large number of machines.

Biometric authentication

Biometrics refers to a number of methods that involve measuring and matching a physical characteristic of part of someone's body in order to identify them. The approach is some-times styled as 'three factor authentication' referring to the use of something you are, over and above the conventional two factors of what you know and what you have.

A wide range of biometric authentication solutions are now on the market and have captured the imagination of many in information security. Biometrics have great intuitive appeal,

> ... using smartcards to store and deploy private keys ensures the highest evidentiary weight.

---

4 It should be noted that the fat client design of eTax was chosen primarily for its protection of personal data. The software includes tax calculators which allow what-if scenarios to be tested by the tax payer in preparing their return. A thin client implementation of the calculator would necessitate trial data to be sent to the Tax Office, and this might reveal personal details the tax payer would prefer to keep private.

5 Over the years, several cryptanalytical attacks on smartcards have been put forward by researchers. Typically these attacks involve obtaining the card and measuring some particular aspect of its performance in order to reveal information about the private key contained within. Such attacks include Timing Analysis (where it is possible that the pattern of bits in the key is manifest in variations in the time it takes the chip to perform encryption), Differential Power Analysis (where the bit pattern might be discerned from glitches in the device's power supply), and the 'Belcore' Attack (which relies on radiation-induced temporary errors in the chip's firmware to reveal statistical information about the key). Not only do these attacks require illicit access to the card or peripheral equipment; they are usually easily thwarted by designers deliberately building noise into the cards.

for they promise absolute identification for high risk applications. Yet it is easy to overlook certain fundamental limitations of biometrics and wrongly assume that they can and perhaps will supersede PKI. In fact, the overwhelming majority of biometrics have the same limitations as shared secret authentication: (1) they only work in closed groups, where the user is already known and registered, and (2) they usually provide no signature function but rather are for access control alone.

This is not to deny the potential usefulness of biometrics in high risk access control applications. Indeed, one of the finest goals for biometrics is to replace PIN controlled access to cryptographic smartcards and the like.

Regular biometrics

Regular biometric authentication involves scanning a chosen body part or even body function, algorithmically reducing the scan to a set of numerical values, and comparing the result with a previously registered reference set or 'template'. Commercial biometrics are available which work on fingerprints, the iris or retina, the shape of the hand, the geometry of the face, the spectral characteristics of the voice, and many other features.

The trick is to pick out certain characteristic markers that can be reduced each time to a highly consistent set of values – within a very large space of possible values. This makes it unlikely for the same set of values to be found in measurements done on different individuals – but not impossible.

In practice, each time the biometric measurement is performed on the one person, the raw data that results will be slightly different. All sorts of factors make this so. The angle or position of the body part can vary, the voice can be affected by a cold, the face can age or be affected by injury. An effective biometric system therefore has to be tolerant of reasonable variations in the raw measured data, yet this inevitably introduces the risk of confusing two different people as one.

Therefore, each different biometric has to be engineered to balance the risks of (1) a false negative, where the system is so selective it can sometimes fail to properly detect a legitimate user, and (2) a false positive, where the system is so accommodating that it occasionally confuses a stranger with a registered user. A biometric system will be deliberately biased one way or the other, depending on whether the business application is more tolerant of

the risk of impersonation, or the risk of user inconvenience from being locked out.

It is just as well that fundamentally biometrics only work in closed groups of previously registered users, because the profusion of different approaches – and even of different algorithms with one approach like fingerprinting – means that in practice there is little hope of interoperability between any of them.

Signature Dynamics

Of all biometric approaches, one stands out for its ability to indeed produce an electronic signature – Signature Dynamics. This technique uses a stylus and digitising tablet, not to merely digitise the user's hand-written signature, but rather to measure its dynamical characteristics – such as the velocity and acceleration of the various curves, and the order in which various marks are made. Furthermore, the measurement can be used to seed a cryptographic algorithm and thence bind the user to an electronic document.

Proponents of signature dynamics argue that it introduces a clear element of 'ceremony' to electronic signatures, the creation of which can otherwise become rather robotic. This in turn better captures the user's intent to be legally bound by what they sign. However, adequate ceremony can always be crafted at the application level for any underlying electronic signature technology. And it seems unlikely that signature dynamics can scale up to high volume e-business transactions, like the signing of cheques, automated procurement messages, stock trades, and digital certificates.

Comparing authentication options

For comparison purposes, the table below shows each authenticator characterised or scored against the following qualities.

- **Access control (Y/N):** does the authenticator provide for access control?
- **Signature (Y/N):** does it provide an electronic signature?
- **Integrity (Y/N):** does it provide a test for the integrity of an authenticated transaction?
- **Open (O/X):** can the method authenticate parties in open groups where there is no prior dealing? Or does it only operate in closed groups who are already known to each other?
- **Number of factors (0, 1, 2 or more):** does the authenticator utilise one factor (what

you know), two (what you have), or more, including the third biometric factor (what you are)?

- **Vulnerability (scored from 0 to 5):** the susceptibility of the method to identity theft.

- **Evidentiary weight (0 to 5):** the relative degree of certainty, factoring in the provision of an electronic signature and the vulnerability to identity theft.

- **Availability (0 to 5):** how widely may the mechanism be deployed in the business computing environment?

- **Convenience (0 to 5):** assuming that the mechanism is available, how easy is it to use?

- **Robustness (0 to 5):** the relative resistance to routine wear and tear or malfunction.

- **Affordability (0 to 5):** factoring in the cost of any special software or hardware.

Conclusion

There is no one-size-fits-all authenticator for e-business. In most jurisdictions, service providers have flexibility to choose a technology that is fit for purpose, and furthermore may be expected to have performed a risk analysis in support of their decision. The core of such an analysis should be consideration of the need to bind users to the transactions they are executing on your system the ease of proving that binding via circumstantial evidence like audit logs or the more direct evidence of an electronic signature, and the risks to the business of failing to bind users.

For high value or high risk transactions, PKI provides especially clear binding through digital signatures. And the desired evidentiary weight of a digital signature should be used to drive the appropriate selection of private key mechanisms within any PKI.

Method	Access control?	Signature?	Integrity?	Open?	No. of factors	Vulnerability	Evidentiary weight	Availability	Convenience	Robustness	Affordability
Shared secret – Password	Y	-	-	X	1	*	*	*****	*****	*****	****
– One time PIN	Y	-	-	X	2	****	**	****3	*****	****14	**
– Challenge-Response	Y	-	-	X	2	****	**	****3	****	*****	**
PKI – Browser based soft certs	Y	Y	Y	O	1	**	***	****4	**11	**15	****
– Server based soft certs	Y	Y	Y	O	2²	***	****	****5	*****	*****	**
– Fat client based soft certs	Y	Y	Y	O	1	***	****	**6	*****	****	****
– Smartcard	Y	Y	Y	O	2	*****	*****	****7	*****12	*****	****
– USB dongle	Y	Y	Y	O	2	*****	*****	****8	***13	**16	****
Biometrics (regular)	Y	-	-	X	3	****	**	**9	*****	***	**17
Signature Dynamics	Y	Y	?¹	X	3	***	****	****10	****	***	*

Black: ratings for the current time
Red: ratings for the medium term (two to three years hence)

Explanatory notes

1. Signature dynamics algorithms are still under development. At the time of writing, it was not clear that all available solutions incorporated an integrity check on the document being signed.

2. It is assumed here that server-based soft certificates should only be deployed using two-factor access control, rather than simple password.

3. Two-factor shared secret tokens may be used anywhere there is a regular Internet connection.

4. Good quality certificate support is only available in relatively recent versions of the Intel/Windows platform browsers. Thus, certificate support cannot yet be said to be universal.

5. Server-based soft certificates may be used wherever there is Internet access from a relatively recent browser version.

6. Fat client software must always be specially distributed and so cannot be presumed to be installed wherever the user happens to be.

7. The smartcard's availability today is hampered by the ongoing lack of readers in existing PCs and other common appliances. This will change in time, with readers expected to eventually become a standard feature of all new machines.

8. The USB port is common in all recent PCs and laptops.

9. It is inconceivable that computer and peripheral vendors will converge on a small number of regular biometric methods. Even within the one class of measurement – fingerprinting for instance – there are multiple non-interoperable equipment vendors.

10. At least one signature dynamics vendor is aggressively targeting the palm computing platform, a strategy which, if successful, may greatly extend its availability.

11. The convenience of browser-based soft certificates is impaired by the relative difficulty of using the one certificate on multiple machines, which entails exporting one's private key from the hard disk to a diskette.

12. Once the user has access to a card reader, using a smartcard is as simple and as familiar as using a magnetic stripe card.

13. Many USB ports today are at the rear of the PC or laptop and are difficult to access. Even if the port is at hand, the USB dongle requires some care and attention in insertion. They are readily dislodged from the port.

14. Time-based one time PIN generators sometimes run foul of the server and the token losing synchrony.

15. Browser-based key stores may be fragile in the event of a PC crash or system software upgrade. Keys and certificates often need replacement in these cases.

16. The USB port was specified for more or less permanent connections to peripheral equipment, and not for frequent insertion and extraction. The connector has a design lifetime on the order of thousands of insertions. Dongles, if used daily, are therefore liable to fail in significant numbers over the lifetime of a private key.

17. A small number of regular biometrics can be expected to become standard features in PCs and laptops (fingerprint scanners are included in some models now) and therefore the cost may come down somewhat.

About the author

Stephen is the Asia Pacific leader of the PricewaterhouseCoopers Cryptographic Centre of Excellence and a Director of beTRUSTed Asia Pacific.

He can be contacted via e-mail at stephen.g.wilson@au.pwcglobal.com

Successful Security Officers do it in the Old-Fashioned Way

by Paul Rivers, e-Risk Limited

It must be every CIO's nightmare to wake up to headlines like this:

Animal rights activists cause major breach of Bank account security

August 1st, 2001

The Stop Huntingdon Animal Cruelty campaign has been leaked highly confidential documents, records and account information of some of the Bank's biggest corporate customers. This information has now been publicly published on the Internet, resulting in a major security breach for the bank and its customers.

The documents are internal records detailing the accounts of various big name customers. This could have disastrous consequences for the Bank in terms of damaged reputation and loss of confidence by its largest customers.

Closing the stable door after the horse has bolted is the original no win solution. But who will be the next person to leak sensitive information or when and how will the next attempt to break in occur? Where should we focus our preventative efforts, and how best should we prepare? The old-fashioned ways that led to good security are just as applicable today and if we, by a process of under-estimation and over-confidence, place too much trust in the tools of the trade, then we are cruising for a bruising. There is an old Russian proverb that was quoted by President Reagan during the SALT talks and which shows us the way to proceed: it is 'Trust, but verify'.

Automation, it has been said, takes a manual process that works, and turns it into one which nearly works, but is faster and cheaper, and different. The key point to bear in mind when we introduce computerised processes, is that we replace a manual and visible set of processes with a complex set of software procedures that run in a complex computer and network environment that we can no longer sense and understand directly.

In his seminal work, 'The Art of War', the Prussian General, Carl von Clausewitz, referred to 'the position of the interior'. Here, we must defend against every possible attack. The enemy, on the other hand, only needs to find one flaw in our defence in order to defeat us. The enemy can collude, conspire, and wait for technology to grant it additional tools. Our watchword must be, 'Know your enemy, and be prepared'.

Who then is the enemy? And are they already within the walls?

**Sources of computer risk
(Computer Security Institute)**

external
attacks
2%

viruses
4%

disgruntled
employees
10%

dishonest
employees
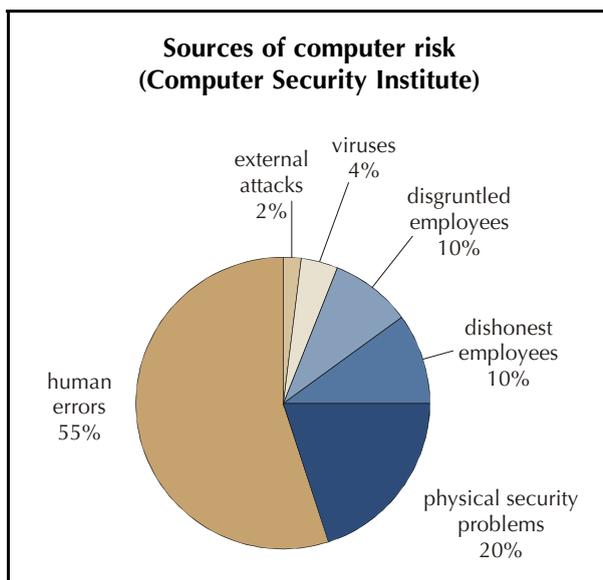10%

human
errors
55%

physical security
problems
20%

Figure 1

Figure 1 shows some interesting statistics on the sources of computer risk. 80% of the computer-related crimes and security incidents come from insiders – from error and direct attack. These statistics appear in a number of respectable texts on information security, but their origin is somewhat uncertain. Although the Computer Security Institute has been accredited with its source, Eugene Schulz in Information Security Bulletin (Vol 6, #2, March 2001) relates it to a seventeen year old FBI study. If this is true, and these statistics are so far out of date, are they still relevant to us today? Indeed, the Computer Security Institute regularly publishes summaries of reported incidents, and their recent statistics show that external attacks now outnumber internal attacks. This is unlikely to be a surprise if we pay any credence to the media reports on the effect of worldwide computer viruses and Eastern Block, Mafia-based or 'script-kiddy' hacking attacks.

However, the relevance of the data in Figure 1 is best viewed in terms of the impact of the risk on the organisation. The insider threat still presents the greatest **financial** impact on an organisation because the threat is directed solely against the organisation and it comes from within – a trusted environment, which is always open to damaging attack. Although external attacks are growing exponentially, and some are, no doubt, fuelled by international crime syndicates, we need to differentiate between those attacks that are targeted at our organisation in particular and those that are looking to exploit vulnerabilities in **any** organisation. The latter, once barred, will, in all likelihood, look for easier targets elsewhere.

So, are we really in control of our internal environment? Do we have the balance between internal and external threat mitigation right? In my experience within organisations that no longer fit into one room, and which have more than one computer, the answer is a categorical 'NO!'.

The human element

Given the statistic that disgruntled or dishonest employees cause about 20% of security incidents, it seems wise to pursue a policy that tightens personnel selection and security. Here are a few pointers that can only improve the quality of staff.

- Conduct pre-employment background checks – take up references and do it properly.

- Publish the security policy of the Company. Ensure that Management endorses it and that employees understand it.

- Emphasise the necessity for employees to be alert to suspicious activities (eg. money laundering is an important area to watch for).

- Ensure that there is clear separation between authorisation and operational functions.

- Be prepared for anti-company activities from disgruntled employees, ex employees, competitors and even customers.

- Ensure that all rules and regulations relating to systems access are explicitly stated, and those penalties for violation are equally clearly stated and seen to be imposed.

- Ensure that appropriate holiday policies are enforced, and rotate staffing where possible.

- Limit access to sensitive systems on a need to know basis. The starting position should be to deny all access and then to grant privileges, rather than the other way around.

- Following a dismissal or resignation, ensure that:

  - The employee's obligations are clearly and directly reviewed with the employee;

  - Remove computer access as soon as possible – review carefully any continuance of normal working during a notice period;

  - Collect keys, smart cards, etc;

– Ensure that the employee's personnel files are secured for future potential use;

– If the employee had administrator or superuser status, change all passwords on the system.

Authentication

When a user logs in to one of our computer systems, what checks are performed other than validation of user name and password? Most probably, no other validation check is performed at all, even though many privileged users might be on holiday for over 10% of the year, sick for an additional five days, on business trips or just out at lunch. It is very difficult to be 100% certain just who is in a building at any given time, and where they are, and whether they are authorised to be in a specific location, and most importantly, whether they are who they purport to be.

Door access mechanisms are standard features in financial institutions, but it is often too complex or onerous a task to restrict building access to the specific days worked by part-time and temporary staff, or adjusting access for the business trips, holidays and sickness periods of perma-nent staff. It is also unlikely, in my experience, that HR has the granularity in its record systems to identify in real time exactly who should be in the organisation on a given day.

Before a user is authenticated by a computer system, we should be able to confirm automatically and in real time that:

- The user is physically present in the building or permitted to access computers remotely, and that we know which is the case.

- The user is reasonably expected to connect with the computer system – ie not on holiday, or on a business trip, or sick, or otherwise not in a position to access the computer system – eg. in a meeting.

- The user actually is the correct person and not someone pretending to be the user.

It is generally accepted that each computer user should have a unique identifier and that it is bad practice to share a generic user name like 'operator', 'administrator', 'temporary', 'telex', 'engineer', etc., but it still happens and to the detriment of the computer access audit trail. It is critical that a computer user should be held

responsible for the actions performed under the computer user's user name (or identifier).

Proof of authentication – the conveyance and establishment of trust – is an interesting human exercise that is an integral and essential part of social interaction. When a new person is introduced to us, the establishment of trust is dependent on many signals:

- the stature of the person who performs the introduction, and the nature of our existing relationship.

- the environment of the introduction.

- the appearance of the new person.

- the presentation and establishment of credentials – the recognition of common acquaintances, schools in common, shared experiences.

After the first introduction, the degree of trust builds up as events confirm our initial basis of trust. It reaches a peak where it may remain steady, but continual input is received which enables us to readjust our initial judgement.

> **When a new person is introduced to us, the establishment of trust is dependent on many signals.**

Establishing authentication between computer users and our crown jewels is a similar exercise in the initial establishment of trust, but the resultant actions are between computer and computer user, without the ongoing complex biological trust-reinforcing signals and we should think clearly about the life cycle management of this trust. 'Life cycle' refers to the processes that are involved in the establishment, the ongoing maintenance and the end of life tear down of the trust mechanisms. It is only when we consider the whole life cycle that we begin to appreciate the overall cost of ownership.

Two party computer authentication requires the exchange and verification of a secret that is shared only between the two parties seeking to establish trust. We have three conceptual levels for these secrets:

- Something we know.

  – This is typically a verifiable password. This password does not need to be known by the second party, but needs to be verifiable through a trusted process.

- Something we have.

  – The SecurID token which generates a unique but verifiable (!) number is a

well-known example. A token is a device that can be lost or stolen, but it cannot be shared or compromised (easily). This is a major improvement over the use of secret passwords, as, other than by employing race conditions to hijack a login session (which, if successful, prevents the correct session from being established – an event which is immediately apparent to the user), a session cannot be started without it.

- Something we are.
  - For example, a smart card thumb print reader extends the power of a token so that, even if it is stolen or lost, the device is unusable. This assumes, of course, that the device is well made and cannot be compromised without breaking it.

Of course, none of these techniques are impervious to misuse through collusion between users and direct threat to the owner of the device. Equally, there are many interposed processes that rely on an implicit trust of the user to computer authentication process. In these days of rogue mobile code, it begs the question, 'how do we trust, and continue to trust, these verification processes?'

Can we trust a PC as the primary vehicle for initiating e-Business transactions? Can we ensure that e-Business transactions initiated at the PC are communicated over the internet in a trustworthy way?

There is a key paper that provides terrific insight into the requirements for secure (trusted) computing. It is:

> 'The Inevitability of Failure: the flawed assumption of security in modern computing environments' by Peter Loscocco et al. (published by the USA's National Security Agency)

In this paper, Loscocco presents the case for a secure operating system. Without the features of a secure operating system – eg. mandatory security and trusted paths – our applications, in particular, our security applications like SSL and IPSEC, cannot, in the final analysis, fulfil their design criteria.

So, in the final analysis, we cannot trust a PC running the standard Microsoft and Unix operating systems. Do we want a dancing paper clip, or security?

In January 2001, the Trusted Computer Platform Alliance (of computer big-names) released a document entitled, 'Design Philosophies and Concepts, version 1.0'. Broadly speaking, its

purpose is to encourage the use of computer platforms for critical purposes by improving the basis on which a computing environment may be trusted. Its thesis is that an entity can be trusted if it always behaves in the expected manner for the intended purpose.

1. The PC is turned on

2. The TCPA-compliant BIOS boot block and the trusted platform module (TPM) attests that the BIOS can be trusted

3. The BIOS loads and queries to ensure the user is authorised to use the platform

4. The BIOS and the TPM attests that the OS loader can be trusted

5. The OS loader extends its trust to the OS kernel, and so on.

So, in the final, final analysis, we need to look beyond secure operating systems with mandatory access control. Sounds very much like we ought to use a dedicated terminal locked inside a vault without a key! Clearly, this is not a useful way forward, but it illustrates that we need to understand the vulnerabilities of the tools we choose to use and ensure that these limitations do not offset the benefits from the process.

Access control to information file systems

When a new account is established, it is normal practice for business approval to be obtained in writing first, and a record of it and the subsequent actions retained. However, the match between a given business role, as it is understood by line management, and the permissions to access software, applications and file systems as granted by computer operations is difficult to make. Management often specifies that new user X should be given the same access rights as person Y, because X does the same job as Y. But person Y's access is not reviewed and verified at the time. It is generally true that access permissions are granted but rarely withdrawn. This is especially true in the dynamic changes that often occur in trading room front desks. Access to the new system, the new business tool, is important and urgent. Removal of the old access control is not as visible, nor as urgent, to line management.

Accurate access control is dependent on a clear identification of business role. The verification of business role is often attempted by checking the department reporting structures held by HR and Accounts Departments, but these departments use and maintain these data for different purposes – HR for payroll and for

personnel reviews and the Accounts Department for cost control – and often at only monthly or less frequent intervals. It is often possible for a person, eg. a shared PA, to be split between departments and data from these sources soon become unreliable.

For audit and control purposes, we need to be able to review application and file system access permission by business role and also by authorising department head. We need to be certain that these access permissions remain appropriate and that any change in department reporting structure or business role or application triggers a review of access permissions.

All of this administrative work is unexciting, mundane, and inglorious but arguably equally essential as the high profile, highly skilled and well-paid network monitoring 'propeller-heads' who try to keep our externally facing systems secure.

The use of directories in modern operating systems (eg. WIN2000 Active Directory) presents an excellent opportunity to establish centralised control. Directories are the glue that can hold together the many to one relationships between a computer user, the business roles approved for the user, and the user's place within the organisation and define the access control to the computer systems that enables the user to do the job.

There is a strong case to support the establishment and maintenance of clear and active documentation of business roles that can relate computer access permissions with business authorisation in a manner that is meaningful to all parties. These data need to be reviewed by both business and system management whenever there is a change in the business, the personnel involved with the business or the relevant computer systems. This change control process is expensive in terms of time and effort but is an essential part of the cost of ownership of computer systems. Its critical role within the overall scope of Operational Risk cannot be ignored.

About the author

Paul Rivers is a consultant with e-Risk Limited, specialising in information security, infrastructure architecture and its implementation, and technology risk management.

He can be contacted via e-mail at paul.rivers@e-risk.ltd.uk

Bringing XML to PKI

by Mark O'Neill, CTO, Vordel Ltd

XML represents one of the most important new waves to hit computing in recent years. XML is a subset of the Standard Generalised Markup Language (SGML), the ISO standard for defining electronic documents independently of how they are to be displayed and, as such, has been around in concept for many years. Now the concept of display-independent structured information is being applied to the Web and some exciting possibilities are emerging.

The World-Wide Web Consortium (W3C) is now following through on its longstanding aim to extend the present HTML-based Web, which is all about display, to a new XML-based Semantic Web, which is all about information. XML is a meta-language that can create different languages for business-to-business data exchange that are specific to vertical markets. It can also be used to separate content from the display of that content. The Hyper Text Markup Language (HTML) on which the Web is built binds the information to be displayed on a Web page with the instructions for displaying that information. XML describes the information, but makes it possible for it to be displayed in different formats, depending on the device that is displaying the information (say, a WAP phone, or interactive television).

Initiatives aiming to define the business documents that are to be sent across this XML network include RosettaNet, OASIS, and ebXML. These initiatives sit at a high level as over-arching frameworks – much like EDIFACT, ANSI/x12, and TRADACOMMS in the world of EDI – in fact ebXML is trying to redefine the UN/EDIFACT standards for XML.

Meanwhile, a number of low-level technologies provide the mechanism for sending XML pay-loads between computers – these technologies combine new protocols like the Simple Object Access Protocol (SOAP) with familiar protocols like HTTP POST. The XML security layer sits in between the over-arching frameworks and the low-level protocols. It does not belong up at the document layer (where standards such as ebXML sit) because using custom XML tags for security is unwieldy and time-consuming to implement. On the other hand, the low-level protocols are payload-independent and while they are designed to facilitate and not impede XML security, they do not implement it. Transport-level security standards like IPSEC that are not XML-specific are still applicable here, but, towards the end of 2000, a number of XML-specific security standards emerged that have the potential to provide an XML-specific security layer.

One very special reason why XML-specific security is important is the many different protocols that now move over Web ports, thus bypassing firewall restrictions. Networked applications use different ports – like gateways for information – to access data across an Internet connection. For an IT manager it is an appealing prospect to Internet-enable an application by opening it over Web ports using XML. Unfortunately, many firewalls block applications that try to access information across port 80, which is the port used for accessing Web pages.

Quite often, the fact that an application is blocked by a firewall looks to users as though the application just doesn't work. Users typically do not understand that a protocol is being blocked by the firewall for security reasons. This problem has held up the spread of PKI, because it blocks PKI-enabled applications from making essential Web port-based directory lookups to find end-user certificate and access privilege information over the firewall.

In an XML-based world, firewalls must be capable of dipping into XML streams travelling over Web ports to check their payloads, much as today's email virus checkers dip into email data streams on mail servers. In the case of XML signatures, this authentication can be done locally or by sending the signature block to an XKMS Trust Service (see below). However, if the XML stream is encrypted, then the traditional firewall is of limited use. Because it simply cannot read the data, the firewalling logic must move to the points where the XML document is decrypted and processed. This is known as a distributed firewall. Already we're seeing a consolidation of companies in the PKI and content-filtering areas, and the growth of personal firewalls. These are trends that are set to continue.

XML Signature

Varying standards are emerging that enable data to be digitally signed using XML. The first security-related XML initiative to reach consensus support is the XML Digital Signature (XML-DSIG) specification, written by a W3C working group. The importance of XML to signatures is indicated in the fact that the W3C's work on digital signatures focuses on XML signatures, and it emphasises that XML signatures may be applied to any digital content. The XML

> **Quite often, the fact that an application is blocked by a firewall looks to users as though the application just doesn't work.**

signature can be within the same document as the signed data (known as enveloping), or in a separate document (detached). XML Signature is designed to make use of many different encryption and hashing algorithms. It has been implemented in Baltimore's X/Secure toolkit and also as part of IBM's AlphaWorks program.

S2ML and AuthXML – Two become one?

In November 2000, two separate initiatives were announced to develop an XML standard for transporting security information between on-line commerce systems. The two initiatives are S2ML (Security Services Markup Language), led by Netegrity, and AuthXML, led by Securant Technologies. The goal of both initiatives is to implement Single Sign-On, one of the holy grails of computing, between on-line trading environments.

Single Sign-On is needed because on-line commerce typically involves many Web sites or Web services, which need to share information about a user. S2ML or AuthXML would enable partners and affiliates to link their exchanges together to share entitlement information – for example, credit limits and 'gold card' type profiles. Also, both protocols would eliminate the need for users to repeatedly enter registration information onto multiple Web sites. Participants in S2ML include webMethods, Sun, VeriSign, and Jamcracker. In addition, the ebXML working group has endorsed S2ML. Participants in AuthXML include Check Point, Novell, and Valicert. Some vendors have signed up to support both competing initiatives.

As S2ML and AuthXML address the same requirements but are not interoperable, OASIS set up a Technical Committee for XML-Based Security Services to merge the two initiatives into a single standard. It was felt that a single standard would be a more favourable outcome for the industry than two competing initiatives. After all, a standard should be something that everyone uses. The OASIS initiative to merge AuthXML and S2ML is still at the early stages, having started in December 2000 but, following the first few conference calls, the effort is starting to take shape. It will still take about a year before products based on the unified standard are available. Until then, companies such as Jamcracker, which supplies customer relationship management to ASPs, must contend with the

existence of many different proprietary authentication and authorisation systems.

Using XML for PKI Integration

In November 2000 Verisign, along with webMethods and Microsoft, announced a number of new XML specifications that apply XML to PKI integration. The new protocols are gathered together under the XML Key Management Specification (XKMS). XKMS aims to shield developers from the complexities of PKI, and the need to use PKI toolkits, by leveraging the Web services model.

Web services is a new concept, advanced by companies such as Sun, IBM and Microsoft. A Web service is a piece of software on-line that is largely shielded from the end-user or application accessing it. It takes a data input and returns an output that can be used by, say, an e-commerce application, and is an ideal way for companies developing software to minimise their efforts and decrease time to market by using existing on-line components.

Currently, developers who want to use digital certificates and signatures in their applications must use programming toolkits, and these toolkits generally only interoperate with that vendor's PKI software. This can cause 'toolkit bloat' in applications which must link with multiple vendor libraries, resulting in a complicated support matrix of supported certificate authorities and their associated key management protocols. A Web service-based approach to PKI integration aims to reduce the pain of implementation.

XKMS works by allowing an application that is processing a signature to act as a client, simply passing the KeyInfo block of an XML Digital Signature to a trusted Web service (called a Trust Service) that then processes the public key information. This Trust Service element of XKMS is called XML Key Information Service Specification (X-KISS). XKMS is a Web service defined using the Web Services Description Language (WSDL). It uses SOAP as the wire protocol, with S2ML authentication incorporated into the envelope header of the SOAP message.

It seems inevitable that XKMS is only the beginning of the use of XML in PKI. Many of the intra-PKI messages defined by PKIX are strong candidates to become XML-based, since XML is now the de facto means for defining any kind of meta-data. A number of PKI vendors are now using XML to define certificate issuance policies, although no inter-vendor standard that would facilitate cross-certification appears to be on the horizon. It will be interesting to see

how XKMS performs, and it looks likely to take off, not least because Microsoft is bundling it into its .NET offering. XML is the key to getting PKI products from different vendors to talk to each other but, at the end of the day, XML is just a file format and some desire from the PKI vendors for cross-certification is also required.

XKMS is the first application of Web services to PKI, but conversely PKI has a lot to add to the Web services architecture. The XML Service-Oriented Architecture (SOA) envisioned by people such as Steve Burbeck of IBM's Emerging Technologies Group, is ultimately moving towards the dynamic choice of business-to-business collaborators. This involves querying a broker or service database with a search pattern that allows a set of alternative service providers to be returned. An example is 'please find a cheap vendor of paperclips and order 10,000 units'. Here there is an obvious application of PKI technologies – in order to certify that the vendor is trustworthy, and that the purchaser has an appropriate credit rating. The OASIS-sponsored convergence of S2ML and AuthXML looks likely to provide the means of achieving this.

In summary, the XML and PKI worlds have a lot to offer each other. XML Signature and initiatives such as S2ML and AuthXML offer the means to secure XML-based e-commerce, and they fill in the current security gaps in the XML architecture owing to the bypassing of firewalls. XKMS is an interesting development that could represent the beginning of the XML-isation of the PKI industry – and hopefully this process will bring about some cohesion between various PKI vendor products.

Canonicalisation

The W3C Candidate Recommendation XML Signature throws up the issue of canonicalisation. Canonicalisation refers to the need for the signed XML document itself to be protected against changing subtly while it is in transit.

To understand why canonicalisation is important, it's important to understand what a hash is. A hash is a value produced by a one-way mathematical function that is run on a piece of data. If someone else runs the same hash function onto the data, they will obtain the same hash value. This is how signatures work – this hash value is encrypted with the private key of the signer, and then anyone with access to their public key can decrypt the original hash, compute a new hash based on the data they have received, and make sure that the two hash values are the same.

XML presents a number of problems for signature verification. An XML document may contain some white space between tags, for example, and this white space may be disregarded by a XML processing system. Similarly, the order in which tags or attributes occur in an XML document may be changed when it's loaded into a processor. The problem here is that when the receiving application computes a hash of the document, having disregarded the white space or the tag-order, the hash will not match the original hash and the signature will not compute. In addition, certain differences between file formats on different operating systems can cause XML documents to subtly change as they are sent between disparate machines. These are the issues which canonicalisation addresses by defining a standard way – the canonical way - to define XML information.

Sign what you see

The process of specifying the W3C's XML Signature standard has thrown up a number of interesting questions. Some of these touch on philosophical issues, and go to the core of the concepts behind structured data and its representation on-screen. Chief among these is the question of what should be signed. The Candidate Recommendation of 31 October 2000 requires that only what is 'seen' should be signed. The word 'seen' is quote marked because the user may perceive the information in another media rather than the visual media – for example, through sound. This is because the user's decision to sign is based on the visual representation of the XML data, not the underlying XML data itself. Remember that XML separates data from the way in which it is represented. The 'sign what you see' requirement can be implemented by taking a bitmapped image of a document when it is signed, although it is then difficult to programmatically process this bitmap later.

An alternative approach is to sign the tags that are generated by a style-sheet to display the XML to the user – these tags could be HTML tags, for example. The XML Signature Candidate Recommendation states that the data should be signed along with whatever filters, style sheets, client profile or other information affects its presentation. This is because the way in which these tags are presented on-screen depends on the rendering technology, which may display the same tags in different ways. It has been suggested that a hash could be taken of the software that is rendering the tags on-screen,

complete with its version number, but then the question is where to draw the line? Does the operating system need to be hashed also? And what about speech-rendering software?

So-called 'active' content presents an interesting case. The scenario goes like this: what if a user signs an XML document which is linked to a dynamic data source that includes functions based on the current time? When the recipient receives the document, the signature will compute fine. However, when the user now views the document, she will now see the document with the current date instead of the date on which the document was signed.

This is analogous to emailing a spreadsheet that contains a date-sensitive macro, so that when the signed spreadsheet is viewed by an email recipient at a later date, the spreadsheet will have changed. The macro won't have changed, but the output of the macro depends on the current time, and that is what changes. This is why it is important to sign the visual representation of the document at the time of signing, rather than the underlying document code itself.

About the author

Mark O'Neill is the CTO of Vordel Ltd.

Vordel produces the TalkXML suite of PKI-enabled XML applications.

He can be contacted via e-mail at mark.oneill@vordel.com

Upcoming Conferences

The following list of conferences has been brought to our attention. We would welcome any additions.

October 29-31, 2001
Las Vegas, NV, USA

Fall 2001 Biometrics Summit

http://www.aliconferences.com

October 29-31, 2001
Albany, NY, USA

Network Attacks & Countermeasures

http://www.networksecuritycorp.com

October 29-31, 2001
Washington, DC, USA

The CSI 28th Annual Computer Security Conference & Exhibition

http://www.gocsi.com/28th_annual/

October 31-November 2, 2001
Dallas, TX, USA

Secure Communications & VPNs

http://www.globalknowledge.com/imsec

November 1-2, 2001
Boston, MA, USA

Securing Your Enterprise's e-Commerce & m-Commerce

http://www.dci.com/events

November 5-7, 2001
Los Angeles, CA, USA

Internet & Web Security

http://www.misti.com

November 5-8, 2001
Philadelphia, PA, USA

Eighth ACM Conference on Computer and Communications Security

http://seclab.crema.unimi.it/~ccs8/

November 7-8, 2001
Ruhr-Universität Bochum, Germany

Workshop on Algebraic Methods in Cryptography

http://homepage.ruhr-uni-bochum.de/Lothar.Gerritzen/GKAMWork.html

November 8-11, 2001
Dallas, TX, USA

4th International Conference on Electronic Commerce Research

http://tecom.cox.smu.edu/icecr4

November 13-16, 2001
Xian, China

Third International Conference on Information and Communications Security (ICICS)

http://homex.coolconnect.com/member2/icisa/icics2001.html

November 15-16, 2001
San Jose, CA, USA

Designing Security Architectures

http://www.globalknowledge.com/imsec

December 3-5, 2001
Montreal, Canada

Privacy by Design 2001 Building Privacy for Better Business

http://privacy.zeroknowledge.com/privacybydesign2001/

December 3-7, 2001
Singapore

National University of Singapore Workshop on Applied Cryptology

http://www.ims.nus.edu.sg/Programs/coding

December 6-7, 2001
Seoul, South Korea

Fourth International Conference on Information Security and Cryptology (ICISC)

http://cnscenter.future.co.kr/icisc01/

December 9-13, 2001
Gold Coast, Queensland, Australia

Asiacrypt 2001

http://www.isrc.qut.edu.au/asiacrypt/

December 10-14, 2001
New Orleans, LA, USA

Annual Computer Security Applications Conference

http://www.acsac.org/2001/welcome.html

February 4-6, 2002
Leuven, Belgium

Fast Software Encryption Workshop 2002

http://www.iacr.org/workshops/fse2002/

February 18-22, 2002
San Jose, CA, USA

RSA Conference 2002

http://www.rsaconference.com/

March 16-22, 2002
Orlando, FL, USA

InfoSec World Conference and Expo/2002

http://www.misti.com/

Call for Articles

If you are interested in contributing to this publication, we invite you to submit articles containing your thoughts, ideas and concepts.

Contribution guidelines for papers being submitted to the Cryptographic Centre of Excellence Journal are:

- Topic must fall under the umbrella of cryptography, security and/or privacy;

- Articles should not be of a promotional or product marketing nature;

- All submissions will be reviewed for content and may be declined at the discretion of the editor (for example, if the tone and/or content is overtly promotional or product marketing-oriented);

- Maximum article length to be 5,000 words plus tables/graphics;

- Submissions must be original work and, where appropriate, give credit to the original author(s);

- The editor reserves the right to edit the text with the agreement of the author; and

- All submissions must be made in MS Word or .RTF format.

PricewaterhouseCoopers reserves the right to re-format for publication purposes and re-distribute as appropriate.

Authors maintain ownership of all submissions.

Completed submissions or abstracts should be submitted via e-mail to either:

geoffrey.c.grabow@us.pwcglobal.com

john.velissarios@uk.pwcglobal.com

beTRUSTed^{SM}

An e-security business of
PricewaterhouseCoopers

www.beTRUSTed.com