PKI considered harmful

Table of Contents

$Revision: 1.32 $

$Date: 2006/05/07 12:07:26 $

Abstract: Public Key Infrastructure (PKI) is now in its tenth year of activity on the Internet, but has yet to break out of pathetically small revenues. Reasons and factors contributing to the failure are many and varied. Is it that the PKI is a solution looking for a problem? That it doesn't solve the problems that it claims? Or that it is to expensive? Perhaps the military objectives failed to cross-over to the commercial sector?

This review lists all of the issues that are unresolved, contentious, or questionable, as known to this author.

Introduction

This paper serves as an attempt to broadly but briefly catalogue the list of serious issues that are unresolved with the concept of Public Key Infrastructure [1] [2].

This is by way of a working draft that might be expanded into a published document one day.

Some Terms

Trust

The general act of a user in accepting the risk of dealing with a party. Trust is individual to each user, and cannot be sold, passed or signed. "I trust you!" is a claim I can make strongly, "you trust me?" has no such strength and is no more than a hope.

Evidence

A statement or fact of some reliability that can be sold, passed or signed. This is what CAs do when they sign a statement of identity to a key - create an evidence of that identity. Users may or may not rely on that evidence as a basis of their trust.

Reliance

The user’s act of performing some act based on an expectation that the evidence to hand is good.

Web of Trust (WoT)

The regime that permits open statements to be signed onto a user’s key by many other users. As the statements are open, it is up to the user to decide what value each statement has. Commonly synonymous with PGP. x.509 does not generally support WoT.

Trusted Third Party (TTP)

A party that makes it its business to perform acts that can be relied upon in a cryptosystem, where the cryptosystem specifically needs this act to be performed for lack of capability elsewise. Typically, the PPKI (below) assumes that TTP is synonymous with CA, but this is deceptive as it implies that the CA delivers trust rather than evidence which results in muddled thinking.

Certificate Authority (CA)

A party that makes it its business to make statements on a public key and sign these into certificates. The CA purports that the statements can be relied upon. Typically, the PPKI (below) assumes that this statement is of the ’identity’ of the key holder, but this is their business model and not essential.

Certificate ("cert")

A public key that carries a statement from a CA. The statement and key are both signed by the CA’s key, and the statement is intended to be relied upon.

Public Key Infrastructure (PKI)

The regime that assumes that public keys must be signed by TTPs. Commonly synonymous with x.509. Specifically excluded is PGP and WoT, which do not make this assumption.

The Public PKI (PPKI)

The PKI formed by commercial CAs acting as signers of statement of identity for essentially three purposes: secure browsing, the S/MIME email system, and code signing. PPKI uses x.509.

SPKI is another system of note, but I am unfamiliar with it. It attempts to resolve many of the flaws of PKI.

A Polemic on Mission

It is commonly assumed by PKI proponents that any criticism of PKI is simply a prelude to unwinding and removing it. This is not the case.

Although there are many, including this author, who point to other methods as more efficacious in protecting user interests, there is a wider mission here: security.

In order to protect users, we deploy the tools that we have to hand. We do not build our ideal tools from scratch because such an approach is simply too expensive. Hence, the need to appreciate and work with tools that are not ideal is *the norm*; but to pretend that these tools are not ideal is simply unscientific. In an academic setting, pretence of ideality is unprofessional, in a security setting, it may well result in losses and thus be negligent.

In order to seriously deliver security to users, we have to seriously understand the weaknesses of every tool. We have to strengthen the weak areas and balance the costs with the strengths, not paper over the cracks and sell on strengths that are irrelevant.

PKI is no exception, and as we are now faced with a security situation of grave concern on the Internet (phishing, viruses, malware and identity data loss) the security community can no longer afford the luxury of blindly supporting a technology just because it sells.

For these motives, the ensuing discussion is overly aggressive, and deliberately so. By aggresively focussing on the weaknesses of PKI, I intend to surface those weaknesses so that we can all see their possibilities. I leave it to others to defend and downgrade these weaknesses, and to present the strengths in PKI.

This paper does not propose removing or unwinding any PKI or any aspect of PKI. What it does is lay the groundwork for future work, hopefully on a more scientific and security-oriented basis. If the reader chooses to repair their PKI, so be it. Likewise, enhancement is a possibility and many good suggestions have come out of the current debate on phishing.

It is far easier to repair and enhance a tool when the flaws are laid bare, and that is what this paper is about.

Credits

Where possible and known and acceptable the original author of the issues is listed. No serious research into these references has been done to date, and any errors remain those of this present author.

The following have by one means or another contributed to the work in this paper: Nicholas Bohm, Stefan Brands, Ian Brown, Roger Clarke, Don Davis, Carl Ellison, Dan Geer, Simson Garfinkel, Brian Gladman, Mark Granovetter, Philipp Gühring, Peter Gutmann, Frank Hecker, Robert Hettinga, Gervase Markham, Gary Howland, J. Marchesini, Eric Rescorla, Ron Rivest, Bruce Schneier, Adi Shamir, S.W. Smith, Nick Szabo, Anne & Lynn Wheeler, Bryce Wilcox, Peter Williams, Jane Kaufman Winn, M. Zhao, and some sources who preferred not to be named.

The Business Case

Revenue Capture

The contracts underlying the business of certificate authorities are aligned towards selling certs, not towards nominal purpose of providing something to users [3] [4].

Further, the economic model for business adoption of PKIs is skewed to very high costs up front, for claimed benefits in the future [5]. This advantages the seller of PKI technology, at the expense of high risks of project failure for the buyer. As few PKI projects have demonstrated return on investment (ROI) this risk remains serious.

In the retail side of PKI for secure browsing (SSL), Certificate Authorities happily re-certify the same key every year, yet over time the risk of key exposure results in decreased security. Further, they will happily sell a certificate for multiple years, at multiples of the one year price. This might indicate that all their costs are based on revocation management, or it might indicate that they are more oriented to generating revenue than delivering security.

If rollover of keys were a service to offer, then a shortterm key rollover service would make some sense. Instead, nominally relying parties are supposed to update from certificate revocation lists (CRLs) yet this is rarely done.

Alternatively, if security were a priority, issuers could issue subordinate authority certificates to server operators, and these local root keys could sign new operational certificates on a regular basis. Yet, this would result in companies that have hundreds of servers only needing one certificate, ruining the revenue model.

In summary, it is difficult to reconcile the business practice of the Certificate Authorities with security needs. It is far easier to reconcile with commercial needs. Modelling the CA as a seller of certificates is compelling; modelling it as a security provider raises far too many questions.

Process Capture

In contrast to the above, much of the infrastructure of SSL and PKI comes from open source projects such as the OpenSSL and Mozilla projects. In these projects, volunteers do the bulk of the coding.

Although one might expect that this would result in work done to the public benefit, in an open fashion, the reverse is more true. Most of the work done on PKI is financed directly by companies that have a large vested interest in the process. Often, these are direct players in the PKI business, and in this way, the open source projects have their security areas ’captured’ by these elements.

This shows up in security groups that claim to be open but reject ’foreign influences’ and in mission statements that contradict security practices.

Reliance

> *A certificate is like a bullet-proof T-shirt; both can be sold as "satisfaction guaranteed, or your money back."*

What are we relying on anyway?

Robert Hettinga makes the point that the only value in a certificate is the value that you get when it goes wrong; yet it is not clear that a certificate authority would or ever has paid out in a fraud, and even the CAs agree when pressed that the warranty programs are more emotional reassurance than substantial backing. Consider this session in which Gervase Markham grills goDaddy, a well known CA [6]:

> ***Gerv:*** *"So under what circumstances might you pay out?"*
>
> ***goDaddy:*** *"Well... you are covered if it's through our negligence. So, for example, if the encryption failed for some reason."*
>
> ***Gerv:*** *"The encryption failed?"*
>
> ***goDaddy:*** *"Yeah."*
>
> ***Gerv:*** *"But if that happened, then everyone's encryption would fail, the entire Internet would be insecure, and you've got a massive world crisis. Are there any less apocalyptic scenarios where you might pay out?"*
>
> ***goDaddy:*** *"Well, not really, no."*
>
> ***Gerv:*** *"Have you ever paid out under the warranty program?"*
>
> ***goDaddy:*** *"No. It's really there just to reassure you that it's a true 128-bit certificate, and to make you feel better about purchasing it."*

The CA is correct of course, and is obviously aware of Adi Shamir's 3rd law: *Cryptography is typically bypassed, not penetrated* [7]. From which we can suggest that the system is better modelled as having no warranty at all, as there is no intention to actually pay out and no plausible scenario where a payout is likely. Indeed, the amounts on offer in a certificate's guarantee are not high enough to make a difference to modern day fraud figures, and often inappropriate to reliance by users.

ArticSoft states it more forcefully [8]:

> ***Relying party liability.*** Get real. If we are stupid enough to put this one in front of our finance people they will shred us faster than you can say Sarbanes-Oxley. The only people we can rely upon is us, unless we have it in writing. So what we need is a system where we can switch on and off who we are willing to do business with whenever we choose. We certainly don't want to be left letting them tell us if they can do business, which is what the whole relying party/revocation approach is all about;

See below for the fallacy of assumption in the One CA model implies One Risk Model, and the discussion on outsourcing Outsourcing Trust, *versus* Outsourcing Risk in Section 3.

Costs

The United States General Accounting Office (GAO) provides some basic costs from Federal PKI experience [9]:

> As of October 1999, GSA made awards to three prime contractors to provide a range of services to any agency wishing to implement PKI technology. At the most basic level, the contractors can provide digital signature certificates to agencies without their having to develop their own PKIs. For each certificate, agencies will be charged an issuance fee - which varies depending on which ACES contractor is issuing the certificate and that currently could be as high as $18.00- and a transaction fee ranging from 40c to $1.20 each time the certificate is used. Agencies will have to determine which applications are best suited to use ACES certificates. For example, GSA officials have stated that it would probably not be cost-effective to use ACES for less sensitive, high-volume applications, such as electronic mail.

A New Zealand government report also warns that [10]:

> "Based upon overseas and New Zealand experiences, it is obvious that a PKI implementation project must be approached with caution. Implementers should ensure their risk analysis truly shows PKI is the most appropriate security mechanism and wherever possible consider alternative methods."

Problems in Engineering

Key Validation

Key validation - done properly - is too inefficient to work [11]. Don Davis views the complexity of validation as a "compliance defect," whereby the rules for managing own keys and validating other’s keys are so complex, that they are unlikely to be met sufficiently [12]. This criticism was borne out in the infamous Microsoft Internet Explorer bug where the full certificate chain was not being validated.

The Trusted Third Party is a Security Weakness

The introduction of the *trusted third party* (TTP) is an *a priori* weakness [13]. His part in this protocol is an additional complication, and an additional party to attack [Trent]. Indeed, as he has total power over issuance of certificates, he is now a major source of weakness, reflecting great stress on the word ’trust’.

The expansion of complexity is not just in the numerical sense of from two parties to three. As well as being a source of technical weakness, the existence of the TTP requires sophisticated techniques of governance - standards, best practices, auditing - to be brought into the security model. There are very few observers or critics capable of isolating dependencies between the governance side and the technical side, and vice versa, and then closing the loop on those requirements. That is, there are very few chartered accountants that also double as security gurus and cryptographers. The result is that the PKI system is strained beyond the plausible limits of comprehension.

These two factors will result in lowered security and need to be balanced against any benefit in security gained by the presence of the TTP.

Expiry and Key Revocation

Expiry and key revocation is an unsolved / unsolvable problem. For example, ArticSoft points at the difficulty of trusting an external party to revoke internal users, and to be verified in doing so [14].

And in the technical domain, Professor Ron Rivest points out the difficulties in staleness [15]. If the certificate has an attribute of staleness, and a relying party really does need to rely on it, then it would need to be brought up to date before the cert could be considered safe. (Imagine here, a long polemic on just how fresh a cert needs to be, which is an unanswerable question.)

Ron Rivest's closing remarks in a panel at FC99 were to the effect of, given all these problems, you may as well do online verification, and dispose of certificates altogether [16]. Bohm, et al, seem to concur [17]:

> Their widespread use would depend ... on achieving practical solutions to many unsolved problems connected with expiry and revocation of certificates.

At this point, we need to examine the assumptions that lead to the engineering need for revocation.

Historical Design Assumptions

PKI has an extraordinarily long history, as computer systems engineering goes. It traces its roots back through the PhD thesis of (*need ref*) back to the seminal article of Diffie and Hellman that introducted public key cryptography (*need ref*). PKI could be said to be the brain child of those early developments, which raises the question of what assumptions then evident are no longer relevant to today's Internet?

Online *versus* Offline

PKI was formally designed around a *store and forward* model of electronic mail. In this model the user connects to her mail server, downloads all her incoming mail, sends out her outgoing mail, and then disconnects. While reading the mail, she has problems authenticating the source of the mail, and must use an offline method unless she wishes to incur the expense of connecting again.

This whole assumption of offine mail was derived by various telecoms and postal committees examining the potential to offer store-and-forward electronic mail systems. In other words, UUCP.

Yet the assumption of offline mail disappeared with the success of the internetworking protocol family known as TCP/IP, and more generally as the Internet. These protocols assumed an always online mode. The offline model tried during the mid 1990s, but the challenge was defeated by various DSL and cable innovations.

OCSP and the Offline Assumption

Online Certificate Status Protocol (OCSP) may be well be the answer to the deprecation of the assumption of offline authentication [18]. In OCSP (*check*) the user contacts the server to check the certificate when she needs to rely upon it, so it can be done on an as-needed basis.

What remains unclear, then, is why continue with use of a certificate at all, and why not simply use a key or a self-signed certificate? Presumably the certificate is considered a gate-keeping operation to enter a user into the system, but the actual cryptographic effect of the certificate's signature from a certificate authority now seems lost.

Everything has to be Authenticated

UUCP and its copies are dead, as is the assumption that the user is always offline. Although the client server approach remains, and offline mail itself continues to be supported, another assumption also disappeared: that the user needed to authenticate the source of the email.

Indeed authentication as an absolute assumption of ecommerce is rather difficult to support. For most of the Internet’s latter history, unauthenticated email has been and continues to be the norm. *Spam* is both evidence of a need for authenticated email, and evidence that the need is not great. The response to the invasion of the user’s email box has been the development of filters, and efforts to use authentication have failed. It appears that authenticating oneself has little to do with the content of an email, and commercial spammers adopted to the notion of adding special features far more readily than did the consumers that were meant to be protected.

With email in particular, it seems that the crying need for protection against phishing is a dead ringer for a need to authenticate email. Yet that is only superficial, as the systems that have been built based on PKI - S/MIME for email and HTTPS for browsing - are not strong enough to stop a clever attacker from tricking the user. Breaching the authentication in the human-computer interface (HCI) is so powerful that PKI is easily bypassed, at least as far as secure browsing goes.

Yet commerce goes on. The vast majority of protective and authentication methods are based around the password and user or account name. To some extent, PKI serves a minor role in protecting passwords from eavesdropping, yet that role is equally served by passive cryptographic techniques such as anonymous diffie-hellman key exchanges (ADH) or challenge-response techniques.

The Trusted Third Party as a Single Point of Failure

Within the PKI standards, there is an inbuilt and deep assumption that the root or *trust anchor* does not operate on itself. In practice this means that even though a root might be delivered as a self-signed certificate, that signature is not checked by the signature validations on a subsidiary cert.

From a logical point of view this makes sense, as a signature on ones own key has little merit in cryptography or security at a higher level. However, there are dangers here; the self-signature has many *engineering* ramifications. For example in the PGP world, this was required to eliminate a potential attack. In the PKI world, using self-signed certificates is indicated any time the infrastructure and software is to be used where trust should not be outsourced.

One of these engineering ramifications is the single point of failure. As the root cannot operate on itself, it cannot revoke itself. There is then one simple attack that cannot be dealt with which is to compromise the root. Rather than deal with this by simply permitting an engineering solution of revoking the root and thus addressing the single point of failure, PKI takes the logical path and states that this is not possible.

This has lead on the face of it to a very strong claim that the root must be protected at all costs. Offline protection, secure hardware, trusted parties and the full weight of governance designs are found in the makeup of the Certificate Authority, reflecting the need to deal with the full ramifications of the single point of failure.

This raises costs significantly; stating that the root must never be compromised immediately creates a very expensive requirement, and feeds directly into barriers to entry (c.f., Porter), which practically guaruntee that the market for certificates becomes a stagnant protected market, even before the first player has got off the ground and becomes profitable enough to think defensively.

There is one more effect that is significant. A single point of failure has important ramifications in finance, military and government sectors. Large, slow sectors that face intensive external scrutiny do not in general accept single points of failure. In a sense, any sector that thinks about disaster recovery would be a poor fit for PKI. This unfortunately results in a clash of revenue models, because such sectors are often the only ones that can afford the highly expensive protections needed by the single point of failure.

Threat Models

Ian Grigg suggests that SSL was designed to use PKI based on the wrong threat model [19].

The "Internet Threat Model" as described by Eric Rescorla is one of the wire being unsafe and the end-nodes being safe [20]. Grigg sees this as the reverse of the reality of the Internet, with miniscule or non-existent reports of threats and losses on the wire, and massive threats and losses on both end-nodes (e.g., phishing, trojans, insider attacks and compromised servers a la Choicepoint).

The "Internet Threat Model" may trace back to military traditions where aggressive radio operations of listening and interfering are routine, thus suggesting the preeminence of the MITM threat [21]. More formally, Peter Williams suggests the model derives from three influences [22]:

1. Voydock and Kent’s influentual 1983 paper on secure protocols [23]:

    > *The model assumes that both ends of the association terminate in secure areas, but that the remainder of the association may be subject to physical attack. A terminal that forms one end of the association may, at different times, be used by various individuals at different authorization levels. The hosts on which the communicating protocol entities reside may provide services to a diverse user community, not all of whose members employ communication security measures. An intruder is represented by a computer under hostile control, situated in the communication path between the ends of the association. Thus all PDUs transmitted on the association must pass through the intruder, The association model is depicted in Figure 3.*

2. SP4, a classified standard from the NSA.

3. NCSC Red Book, Part II’s per-layer-threat analysis.

Perhaps disagreeing with this, Clark and Wilson trace the "Historical Threat Model" back to the US intelligence community where disclosure of confidential material was of preeminent importance, and more specifically the security of nuclear arming modules [24].

Whatever the historical pedigree of the model, Grigg concludes that the choice of the MITM as *the primary threat* to guard against on the Internet lacks any foundation. The lack of validity in the threat model is becoming more and more of an issue as browser manufacturers refuse to compromise on their unfounded model of protocol MITM protection, in the face of rising spoofing attacks from phishing, in itself a variant of MITM.

X.509

PKI, in the general forms that have been touted, has normally been based on X.509. Unfortunately, X.509 turns out to be more of a *glue and string* approach than a real solution. There is good reason for this: PKI was invented before it was needed, and in fact its genesis was in the 1970s work of Whitfield Diffie and Hellman, and a follow on Masters thesis by (*need ref*) [25]. As it was invented before the Internet, and as no demanding application set its requirements, the design of PKI drifted until picked up by telecoms and OSI committees.

This section is a bottom-up analysis that looks at how the x.509 structure came to be used. See also Ian Grigg's top-down analysis - looking at the client's needs, elsewhere. The two analysies are in accord.

String

On X.509's capability to support the notions of a PKI, Peter Gutmann states that X.509 was [26]:

> Anything can be a PKI (PGP keys hand-carried on floppies are a PKI), what X.509 lacks is a match to any pressing real-world problem. For example when I make an online purchase, I don't need a PKI, I need an online credit/debit authorisation mechanism. PKI just gets in the way (the closest anyone ever got was SET, which wasn't a PKI but an online CC system dressed up to look like a PKI - I'm sure Anne/Lynn Wheeler would have much more to say about this).

Gutmann also states that [27]:

> .... [x.509 was] originally designed solely for use for user authentication to the worldwide X.500 directory (something which is very obvious in the structure of an X.509v1 cert), a problem that never eventuated. It is quite literally a solution in search of a problem. The difficulty in applying it to any pressing real-world problem arises directly from its X.500 origins

Glue

Peter Gutmann goes on to outline how it can't be changed [28]:

> Unfortunately any attempts to fix this by switching to practical, widely-used technology (e.g. dump X.500 DNs as identifiers, use online whitelist checking instead of offline blacklists, move them around using HTTP instead of X.500/LDAP, etc etc) so you can actually do something useful with the things, is met with extraordinary resistance by the people writing the standards. As the quote on my home page says: "[PKI designs are based on] digital ancestor- worship of closed-door-generated standards going back to at least the mid 80's. [...] The result seems to be protocols so convoluted and obtuse that vendor implementation is difficult/impossible and costly".

Reengineering

Efforts at reengineering PKI have not succeeded in gaining traction. SPKI is one such effort, and the reasons for it having failed to oust the incumbent are worth studying. Another effort is the OpenPGP *web of trust* model which likewise has achieved some localised successes (notably in email) but has not done more than irritate the PKI school.

Patterns of Commerce

In many commerce spaces, PKI does not mirror actual commerce patterns, so cannot help [29]. That is, if you take any existing commerce pattern, and model it (for example, by drawing out a graph of the interrelationships), it looks completely different to the PKI model.

In fact, it is relatively rare to find any pattern of commerce that maps easily to the PKI model. This practically means that there is little chance of it being used, as to switch from one pattern to another is an expensive exercise, and is only done over time, and for great savings in costs or increases in benefits.

Some observers have commented on the apparent nexus with military needs, and the similarity with military models of control. Yet even there the comparison is only superficial; although the military works to a theoretical hierarchical control model, in actuality modern armies strive to push decision making as far down as possible. Specifically, there are many use cases where commands are overridden at a local level, something that could not be emulated in PKI.

This section presents top-down analysis - looking at the client's needs. See also Peter Gutmann's bottom-up analysis. that looks at how the x.509 structure came to be used. The two analysies are in accord.

Trust is not Outsourced

Trust is one particular aspect of the patterns of commerce stands out as being totally at odds with PKI. Trust is something that business simply does not outsource. PKI's notion of outsourcing trust places an inescapable conflict in front of the managers.

This observation came from the finance markets [30]. There, one would think, that trust derives from ones presence within a regulated system, based on central banks or securities regulators as the ultimate authorities.

Not so! The authorities are only trusted, and transitively trusted, at the level of lip-service. In practice, trust in financial markets derives from a network of personal contacts.

This can be seen when a new player arrives on the scene. She can present all her *creds* or *quals*, but will still not enter in, until she has established her personal trustworthiness with the first other player. Then, that other player can *authenticate* her trustworthiness to others.

In general, there is no single point of trust in the finance markets. Trust is distributed, and is not outsourced. Indeed, it is embedded at the level of individuals, more than it is held at corporate levels.

It is relatively easy to then map this arrangement to other businesses, and see that the general rule applies: trust is not outsourced. The particular trust networks of the finance markets do not so easily map, and trust is often abrogated from the individual level to the corporate level in other businesses.

Applied to PKI, we can see that there is no real business case for an external certificate authority, and if a PKI is needed, it is almost a given that an internal certificate authority is called for - it would depend on how well the internal trust and other commerce patterns map within the company as to whether divisions could share a CA across internal borders.

What then happens when two companies wish to use each other’s certs? They will simply exchange root certs, and for the most part still authenticate relationships along local trust lines, not along PKI lines.

Mental transaction costs when the user does "outsource Trust"

What then happens when Trust is "outsourced" and a PKI is used to intermediate this trust? In practice, what has happened is a shifting of the burden pattern, where the user has simply replaced her trust in the end second party with trust in the TTP. Szabo writes that:

> Trust, like, taste, is a subjective judgment. Making such judgement requires mental effort. A third party with a good reputation, and that is actually trustworthy, can save its customers from having to do so much research or bear other costs associated with making these judgments. However, entities that claim to be trusted but end up not being trustworthy impose costs not only of a direct nature, when they breach the trust, but increase the general cost of trying to choose between trustworthy and treacherous trusted third parties [31].

Are the costs of choosing between good and bad TTPs commensurate with the benefits of not having to choose over second parties? That’s an open question which Szabo does not address. However, it is important to point out that even if it were addressed, for many PKIs such as the public Internet ones, the option simply isn’t there. See the sections below on One or Many CAs?> <p>

Risk *is* Outsourced!

Risk, which Lynn Wheeler describes as the inverse of trust, is indeed outsourced and on a massive scale. The entire derivatives, securitization and insurance markets are simply outsourcing of risks. Yet to outsource risk, the relying party has to be able to measure the risk, measure the cost of that risk, and make an economic decision to buy coverage.

Quite the reverse of outsourcing trust! The SPKI theory RFC describes it thusly [32]:

> 5.7.2 Rivest’s Reversal of the CRL Logic Ron Rivest has written a paper [R98] suggesting that the whole validity condition model is flawed because it assumes that the issuer (or some entity to which it delegates this responsibility) decides the conditions under which a certificate is valid. That traditional model is consistent with a military key management model, in which there is some central authority responsible for key release and for determining key validity.
>
> However, in the commercial space, it is the verifier and not the issuer who is taking a risk by accepting a certificate. It should therefore be the verifier who decides what level of assurance he needs before accepting a credential. That verifier needs information from the issuer, and the more recent that information the better, but at the end of the day, the decision remains with the verifier.

Lynn Wheeler suggests that the [W05.6]:

> One of the issues in the CRL push model is that it is the relying party which is judging the risk (sort of the inverse of trust), and they alone know the basis of their dynamic risk parameters. One issue that they are aware of is that as the value of the transaction goes up, the risk goes up. Another is that the longer the time interval, the bigger the risk.

> The problem then was that since it is the relying party that is taking the risk, and understands their own situation, it should be they that decide the parameters of their risk operation. I.e., as the value of the transaction goes up they may want to reduce risk in other ways, which might include things like trust time windows.
>
> In normal traditional business scenario, the relying party is the one deciding how often they might contact 3rd party trust agencies (i.e. credit bureaus).
>
> PKI/certificate operations have frequently totally inverted standard business trust processes. Instead of the relying party being able to make contractual agreements and make business decisions supporting their risk & trust decisions, the key owner has the contractual agreement with any 3rd party trust operation (i.e., the key owner buys a certificate from the CA).

Decision Outsourcing is a Systemic Flaw

Rather than being a bug in a validity model to be rectified, this flaw is one of systemic proportions.

As described above, in classical business, a party or user conducts the needed due diligence on the other parties. In the PKI view, the audit of a CA generally terminates at the point of showing that a CA does what the Certificate Practice Statement ("CPS") says. That is, the audit cannot say much about the fitness for a given purpose or user, because it does not know in advance what that purpose or user might be. Hence, the PKI process of necessity requires the user to read the CPS and judge for herself, and in this sense is aligned with the general business process if *caveat emptor*.

Consider Prof. Kaufmann’s reading of the Verisign CPS from 1998 [33]:

> *The VeriSign CPS defines the procedures Versign will follow before issuing a Digital ID. Individual Digital IDs are currently offered in three classes.(212) Class 1 Digital IDs "are issued to individuals only," and are issued after VeriSign determines that there are no existing entries in VeriSign’s database of subscribers with the same name and e-mail address.(213) The CPS notes that these certificates are* ***not suitable for commercial use where proof of identity is required****.(214) Class 2 Digital IDs are "currently issued to individuals only" after VeriSign checks not only its own database of subscribers, but also performs an automated check of the applicant’s information against "well-recognized consumer databases."(215) The CPS emphasizes that Class 2 certificates, "[A]lthough . . . an advanced automated method of authenticating a certificate applicant’s signature," are issued without requiring the applicant’s personal appearance before a trusted party such as a notary; therefore,* ***relying parties should take this into account*** *before accepting a Class 2 certificate as identification of the subscriber.(216)*

(My emphasis added.) Yet if we look at popular implementations of PKI, we often find the very reverse of this clear requirement to be familiar with the policies of each CA. Consider the policies of popular distributors such as browser manufacturers: the inclusion of a CA has (at least historically) been justified solely on the basis of the audit of the CA, even though the audit process would generally stress that the user herself should not accept an audit on face value. Not only do manufacturers add the CA’s root keys *without a policy of reading and accepting the CPS themselves*, they offer no access to the CPS to users, and even go to some significant extent to *hide the CA’s name and brand* (offering various motives for this such as screen real estate costs and user confusion).

Browser manufacturers especially are potentially exposed to litigation by their negligence, if there is any SSL-based value breaches such as may arise from phishing. Audit practices and CPSs expressely limit their efficacy by stating that relying parties need to judge for themselves whether the results are suitable; browsers actively seek to remove that information, judgement and choice from users, and don’t themselves take on the role in any serious sense.

As a further quirk or twist of fate in user protection, the CAs are generally complicit in this reversal of PKI practices and general business. Their practices and statements of same are deliberately crafted to transfer risk to the user, while their marketing promises to reduce risk. CPSs are written to be unintelligible as well as so restrictive of benefits to the relying party as to be practically useless. Professor Wynn continues [34]:

> *As a risk allocation system, the VeriSign CPS is moving in the opposite direction of most other electronic commerce systems, and resembles the system established by credit card issuers prior to federal consumer regulations protection.(226) No significant pooling of risks exists for consumer subscribers because, although insurance is now offered, the insurance mimics the standard of care the subscriber is required to maintain by the CPS and, thus, is unlikely to offer any relief to a consumer who cannot prove how a copy of his or her private key was obtained. The CPS allocates fraud or error losses to the consumer who is likely to be much less sophisticated than VeriSign, and is completely incapable of deploying the kind of technology used by credit card companies to reduce fraud.(227) The problem of information asymmetry is acute in consumer dealings with VeriSign because no plain language disclosure of the risk allocation system exists outside the CPS, which is over 100 pages of single spaced text written in dense legal prose.*

Browser manufacturers might argue that their policies are justified on this basis, but this then exposes them to anti-trust considerations - why are they requiring audits of CAs if they know the CPS to be worthless or neutered practice statements?

Granovetter’s Theory of Weak Ties

The above assumes that commerce is the context, as does the PKI industry. It is pretty much accepted that the purpose of PKI is ecommerce, and issues like privacy or trust are not addressed except where they help commerce, or, more cynically, where they relate to sales of certificates and PKIs.

PKI has it backwards. Commerce is simply an example of interaction, and the patterns of behaviour applied to commerce are taken from general patterns of behaviour. Specifically, where trust is distributed, it is delivered transitively, from person to person. That is, Alice gives Bob some information on Carol that allows Bob to trust Carol, to an extent derived from his trust in Alice.

This relates to Mark Granovetter’s theory on the strength of weak ties [35]. The theory predicts that many weak ties are more efficacious than few strong ones. Casting this across the assumptions in PKI, we can see that PKI attempts to create strong ties from CAs to user, and the emphasis on CAs being totally trusted results necessarily in there being few of them. In contrast, competing approaches such as OpenPGP’s web of trust lend themselves more to many weak ties, where each tie is often an interaction of signing between casual strangers. Granovetter’s theory thus predicts that web of trust is stronger than PKI.

Identity is not the Application

One assumption that continues to dog the financial world is that all problems are solved if you can establish the identity of the counterparty. This is not, and has never been the case for most applications.

For payments, especially, identity is irrelevent and what is instead required is a statement concerning the value presented. That is, what is the colour of your counterparty's money? This attention to value not identity is a core result of the psuedonymous designs of SOX and x9.59 [36] [W9.59] [37] [38]. Yet, PKI as embodied in x.509 insists on identity as being the core unit of issuance of certificates. This leads many applications and users down blind alleys, as they attempt to map the notion of identity to servers, browsers, accounts, and protocol end points.

PKI's experiences with Identity may have been a cautionary tale, but not cautionary enough for some: Microsoft's Passport and .NET systems grasped at Identity in a big way. Liberty Alliance set up a counter proposal, to block yet another Microsoft plan to conquer the world, but they also went heavily into the Identity domain. Now, as Stefan Brands reports, these systems are failing to draw support [39]. Why? Nobody has the ability to be able to sit down and design a complete Identity system and expect people to accept it. The application drives how Identity is done, not the other way around, and the application was not identified in these proposals.

See also below, The One True Name, a discussion on whether x.509 meets the requirement of mapping Identities, when it is determined that this is required.

Identity *is* the Myth

Scratch any PKI supporter and they will go to great lengths to support the claim that the Identity expressed in a certificate is of the utmost importance. But this is yet another marketing myth. Lynn Wheeler identifies that when early certificate issuers came to investigate seriously what it meant to issue *certificates with Identity*, they discovered more than just the obvious implementation issues [40].

During the early 1990s, it was realised that the authoritative agencies were not certifying one true identity, nor were they issuing certificates representing such one true identity. This was in part because there were some liability issues if somebody depended on the information and it turned out to be wrong.

Original design discussions in the early 90s by and of independent 3rd party trust organizations centered around claims that they would check with the official bodies as to the validity of the information. They would then certify, so the model suggested, that they had done that checking, and issue a public key certificate indicating that they had done such checking. Even at that point, the independent 3rd party trust organizations were not actually certifying the validity of the information; they were more simply certifying that they had checked with somebody else regarding the validity of the information.

The original business model of these independent 3rd party trust organizations was that they wanted to make money off of certifying that they had checked with the real organizations as to the validity of the information, and the way they were going to make this money was by selling public key digital certificates indicating that they had done such checking.

The issue that then came up was *what sort of information would be of value to relying parties*, and consequently should be checked and included in a digital certificate as having been checked. It started to appear that the more personal information that was included, the more value it would be to relying parties. Not just ones name, but address, age, marital status, and many other characteristics such as ancestory were mooted. Indeed, the very type of detail that relying parties might get if they did a real-time check with a credit agency.

Another of the characteristics of the public key side of these digital certificates was that they could be freely distributed and published all over the world. But by the mid-90s, institutions were starting to realize that such public key digital certificates, if freely published and distributed all over the world with enormous amounts of personal information, represented significant privacy and liability issues [41]. It was also considered that if there were such enormous amounts of personal information, the certificate was no longer being used for just authenticating the person, but was, in fact, *identifying the person* (which is another way of viewing the significant privacy and liability issues).

In response to this uncertainty, some institutions started issuing *relying-party-only certificates* which contained just a public key and some sort of database or account lookup value, which latter directed to where all the real information of interest to the institution was kept [42]. The public key technology in the form of digital signature verification, would be used to authenticate the entity identified in the certificate and the account lookup would establish association with all the necessary real-time information of interest to the institution. This had the beneficial side-effect of reverting public key operations to purely authentication operations, as opposed to straying into the horrible privacy and liability issues related to constantly identifying the entity. Name ==> Individual which is flaw-ridden at each step [62]:

> "All-purpose digital name certificates are of very doubtful utility, among other reasons because names do not adequately distinguish people in large populations."

A better concept is to consider the asset as the basis for the retail transaction: financial cryptography ==> Asset [63]:

> "They are also irrelevant to many transactions (what the merchant needs to know is that a card number is given by the person authorised to give it, whatever their name may be), where they needlessly reduce legitimate privacy."

Sure to make a difference ...

One other factor to bear in mind is that the field is confused with *silver bullets*. Digital signatures are touted as a solution to many things, and because there is no field experience in the veracity of these claims, the story gets bigger each time it is told.

Here are some of the areas where digital signatures are "sure to make the difference:"

- Political Voting
- Payment Systems
- Contracts [64] [65]
- Identity
- Retail Shopping

Conclusion

Wherein, I justify the aggressive title: PKI considered harmful. *To be written...*

In How Many Ways?

Wherein, I summarise the chief claims as disputed.

The Harm That is Wrought

In order to justify the title of this review, I must identify the harms and lay claim to them as costly; theory alone will be insufficient as critical theory will merely beget its companion anti-critical theory. I do not however go further than identification of the harms; it is left as an open topic for research to document and cost these harms.

The harm wrought by PKI is threefold.

In the first instance PKI simply costs money and effort to put in place. This is the least of the damages, as there is only the loss to the implementor (opportunity costs, the economists would say), and indeed for this very reasons PKI escaped the rancour of serious researchers for a long time. We live in a society of capitalism which preaches that all are permitted to spend their own money as they see fit; there is no global guardian that designates the one true way to be secure. In such a society, if a company chooses to spend its hard earnt revenues on schlock, that is its choice and we may do no more than *tut, tut* in mildness.

It is in the second instance, its failure to deliver on the promise, that changes the above situation. Failure to deliver results in losses from fraud. Phishing is the broadest and best documented attack in recent times, but hacking, viruses and the like have all been impacted by PKI in a positive or negative fashion (e.g., the complexity of PKI-based systems has resulted in weaknesses as measured against other systems that might have been employed more economically).

These losses have often been incredibly difficult to deal with and herein lies the third harm: As PKI fills the security spot in the user public's collective mind, there has been inordinate cost in discussing fixing the security woes [66]. This arises primarily out of an unwillingness of security practitioners to face the enemy within; no professional wants to admit to themselves, let alone the world, that the last decade's work was fatally flawed. No professional wants to be told that his or her work participated in the arisal of billion dollar losses.

We face a large and persistent loss of time and energy in each contemporary discussion of security. Each practitioner has to walk through the tortured path of PKI flaws until they reach the inevitable conclusion. Each will reject at every step of the way, reflecting their training and faith in their mentors and peers, and each will need to be tutored through the flaws and failings individually in order to break through the faith and get back to the science. This deconstruction of the religion of PKI is extremely costly; but if we are to make headway in returning science to security, it is a cost that each of us must bear.

In the meantime, the losses continue. PKI is the ultimate expression in the *false sense of security*, and no amount of discussion, paper authoring, and other forms of hand wringing can ever cater or redeem the losses that users have incurred. It can only be stated that this is a task of some urgency, even if we as a body scientific have dispatched the responsibility elsewhere.

Why So Long?

If PKI is an amphigory, it is one of elegance.

It has persisted since the 1970s, which makes it substantially older than the Internet. It continues to be the default security statement for many organisations and people. Only the deepest analysis and the most persistent of criticisms can unearth the flawed assumptions within, and even when laid bare, the structure still impresses with its apparent solidity.

Why so long? remains an open question; for further research in security; why was such an artiface so persistent in the face of continual lack of field evidence and reasonable doubts cast by notable practitioners as listed in the credits above?

Appendix 1: Resources

- CACert, an open CA of members, has a PKI Philosophy Page with collected links. They have the unenviable task of writing a proper CPS with a security goal and not a profits goal.
- Civics.com’s Daniel Greenwood has a Compilation of Resources Regarding Difficulties with PKI.
- Simson Garfunkel *Security and usability can be made synergistic*, thesis dissertation on HCI and the security of users. Especially, Chapter 1: Introduction, Chapter 4: A Survey of Attitudes Towards Digitally-Signed and Sealed Email, Chapter 5: Secure Email and PKI.
- Nicholas Bohm, Ian Brown, Brian Gladman, Who Carries the Risk of Fraud?
- Dr. Stefan Brands’ home page. Dr. Brands’ book, *Rethinking Public Key Infrastructures and Digital Certificates; Building in Privacy* . includes an important summary of the pitfalls of PKI (Chapter 1 .
- Roger Clarke, PKI Misfit
- Don Davis, Compliance Defects in Public-Key Cryptography.
- Carl Ellison and Bruce Schneier , "Ten Risks of PKI", an article for Computer Security magazine, but also see Ben Laurie’s Seven and a Half Non-risks of PKI as a rejoinder.
- Jane Kaufman Winn, The Emperor’s New Clothes: The Shocking Truth about Digital Signatures and Internet Commerce and Couriers without Luggage. See also her full publications page for much good stuff on the legal perspective to electronic commerce.
- PKI resource page.
- Peter Gutmann has done a lot of work on X.509, and has regretted it. His page carries testimony of many of those regrets.
- Anne & Lynn Wheeler’s site at www.garlic.com/~lynn/.
- ArticSoft, Ten things I wish they warned me about PKI. 08 October 2004.

References

[1] PKI Page

[2] Originally, this list derived from a post by Ian Grigg, 04 Dec 2000. Over time, it has expanded into a survey of criticisms of PKI.

[3] Jane Kaufman Winn, Couriers without Luggage.

[4] Anne & Lynn Wheeler's Assorted Writings include many references to the search for the revenue model.

[5] ArticSoft, Ten things I wish they warned me about PKI. *eBCVG*.

[6] Gervase Markham, GoDaddy's $1000 "Warranty" blog entry 17th May 2005.

[7] Adi Shamir Turing Lecture on Cryptology: A Status Report, and also summarised here: blog entry 26th May 2004.

[8] ArticSoft, *op cit*.

[9] US Government Accounting Office, Report on PKI.

[10] E-government Unit, New Zealand State Services Commission, International and New Zealand PKI experiences across government, S.E.E. PKI Paper 14.

[11] Dan Geer made this point to me in a private conversation in 1997.

[12] Don Davis, "Compliance Defects in Public-Key Cryptography," *Proc. 6th Usenix Security Symp* 1996. See also the slides on that page.

[13] Nick Szabo, Trusted Third Parties Are Security Holes, 2001 - 2004

[Trent] In the cryptographic literature, a trusted third party is known as Trent, and is masculine.

[14] ArticSoft, *op cit*.

[15] Ron Rivest, et al. See 4 papers in FC99, and response in FC00, as well as panel. Also R98, op cit.

[16] Ron Rivest, FC99 panel on revocation.

[17] Nicholas Bohm, Ian Brown and Brian Gladman, "Electronic commerce: who carries the risk of fraud?" *Journal of Information, Law and Technology,* October 2000. http://elj.warwick.ac.uk/jilt/00-3/bohm.html

[18] One pretty good explanation is here: OpenValidation.org. *RFC2560 Online Certificate Status Protocol - OCSP* June 1999.

[19] Ian Grigg, " WYTM?" (What's your threat model?), *Rants on SSL and Security practice*. ",

[20] Eric Rescorla, " 1.2 The Internet Threat Model ," *SSL & TLS*, Addison Wesley, 2001. ",

[21] Anecdotal evidence suggest that where authorities bump up against protest elements, the military and para-military parties use MITM and active attacks routinely. This suggests that active and MITM attacks may be employed only when there is no cost of discovery to the attacker (as opposed to risk).

[22] Peter Williams, private email, 2005.01.04.

[23] Voydock and Kent, "Security Mechanisms in High-Level Network Protocols," Computing Surveys, Vol. 15, No. 2, June 1983.

[24] Simson Garfinkel, *Security and usability can be made synergistic,* Thesis dissertation, Work in progress, 2005 (Implied Title). Peter Williams, private email 2005.01.04. ",

[25] A presentation on the history of PKI: http://66.102.7.104/search?q=cache:DYPK1sH7nIYJ:devgroup.mephist.ru/

[26] Peter Gutmann, private email 20 May 2003. 'my approach to PKI work could best be described as "gleeful masochism", sort of like playing golf. That is, I know it's crap, but it's fun playing with the technology, the same attitude take by people who rebuild Trabi's for fun.'

[27] Peter Gutmann, post 18 May 2003 to cryptography list *at* metzdowd *dot* com.

[28] Peter Gutmann, *op cit* 18 May 2003

[29] Ian Grigg.

[30] Ian Grigg.

[31] Nick Szabo, op cit.

[32] Ellison, Frantz, Ylonen, ... SPKI Theory, RFC2693

[R98] R. Rivest, "Can We Eliminate Revocation Lists?", Proceedings of Financial Cryptography 1998, .

[W05.6] Lynn Wheeler, Re: More Phishing scams, still no SSL being used... Post, 14th June 2005, Mozilla-crypto.

[33] Professor Jane Kaufman Winn, Couriers without Luggage.

[34] JKW, ibid.

[35] Mark Granovetter, "The Strength of Weak Ties." *American Journal of Sociology,*, 1973, Vol 78 (May): 1360-1380. PDF of paper and an Overview on Granovetter’s Theory. Granovetter’s diagram is much loved by the capabilities school.

[36] Gary Howland Development of an Open and Flexible Payment System , 1996.

[W9.59] Anne & Lynn Wheeler Writings on X9.59 standard .

[37] Roger Clarke The Fundamental Inadequacies of Conventional Public Key Infrastructure Proc. Conf. ECIS’2001, Bled, Slovenia, 27-29 June 2001

[38] RFC2693, op cit.

[39] Stefan Brands The Identity Corner

[40] This section is fundamentally the writings of Lynn Wheeler, albeit heavily paraphrased by the paper editor.

[41] Note that this was also the era of the EU Data Privacy Directive. That Directive pushed for names be removed from various payment card instruments for doing online electronic fund transactions. If the payment card is purely a *"something you have"* piece of authentication, then it should be possible to perform a transactions without also requiring identification.

[42] Lynn Wheeler, Assorted posts.

[43] Bryce ’Zooko’ Wilcox, Names: Decentralized, Secure, Human-Meaningful: Choose Two

[44] Stefan Brands, A Primer on user identification - Part 4 of 4. This makes more sense if the reader starts at Part 1.

[45] Bohm, *op cit.*

[46] CESG, *Cloud Cover Trusted Third Party Protection Profile,*

[47] I was first made aware of this bug by the writings of Peter Gutmann but thought it a curiousity (no reference as yet). Discussions on the mozilla-crypto list have highlighted it, and I now see it as a fatal security flaw. It permits unprotected MITMs within the boundary of SSL and the core security implementation.

[48] CESG, *op cit*

[49] Ian Grigg, " Who are you?", and " ", Selected rants on SSL.

[50] Ian Grigg, " WYTM?" (What’s your threat model?), Ibid. ",

[51] Bohm, op cit

[52] Nick Szabo, op cit.

[53] Jane Kaufman Winn, Couriers without Luggage.

[54] Ian Grigg, " WYTM?" Ibid. ",

[55] Ellison, RFC2693, Ibid,

[56] Nick Szabo, op cit.

[57] Carl Ellison and Bruce Schneier, "Ten Risks of PKI", an article for Computer Security Journal, v 16, n 1, 2000, pp. 1-7.

[58] Bruce Schneier, "Why Digital Signatures Are Not Signatures", an essay in Cryptogram, 15th November 2000.

[59] Marchesini, Smith, Zhao, " Keyjacking: The Surprising Insecurity of Client-side SSL" Dartmouth Technical Report TR2004-489, *and* forthcoming in *Computers and Security*, 2004

[W05.07] Anne & Lynn Wheeler, Post to Cryptography, 11th July 2005.

[60] Anne & Lynn Wheeler, Post Ibid. Edited for clarity.

[61] Ian Grigg " The Ricardian Contract," *First IEEE International Workshop on Electronic Contracting,* (WEC) 6th July 2004.

[62] Bohm, Brown and Gladman, "Electronic commerce: who carries the risk of fraud?" *op cit*.

[63] Bohm, op cit

[64] Jane Kaufman Winn, The Emperor’s New Clothes: The Shocking Truth about Digital Signatures and Internet Commerce

[65] Ian Grigg " The Ricardian Contract," Op cit.

[66] Here, I refer to the programmer and purchasing body public outside the direct world of security research and primary design. As an example of how large this public collective mind is, it includes today all browser manufacturers, all CAs, all banks, and all large security vendors, none of whom in today’s world have (taken on) primary responsibility for repairing the current models or designing new ones. Arguably, ten years ago, three companies were not in that user public: Netscape, RSADSI and Verisign.