

# An Examination of Asserted PKI Issues and Proposed Alternatives

John Linn, RSA Laboratories, Bedford, MA, USA  
Marc Branchaud, RSA Security Inc., Vancouver, BC, Canada

15 March 2004

## 1 Introduction

Since the 1980s, public-key infrastructures (PKIs) have been widely anticipated as a primary means to make entities' keys available to others in a trusted fashion, thereby enabling a qualitative improvement in the protection and assurance of communications and transactions carried out over the Internet. Certificate-based authentication has become common practice in certain contexts, particularly in conjunction with SSL-protected web sites. In recent years, however, many commentators have lamented the fact that PKI has not achieved more pervasive adoption and deployment. Some, like [Clar01], [ElSc00], and [Gutt02], have concluded that PKI is a failure or does not address users' primary security needs. Opinions differ on the reasons for these results, but most can be distilled into a few general categories:

- A belief that demand for the services offered by PKI, in terms of PKI-integrated applications and/or security-oriented use cases for those applications, has not yet emerged to a degree sufficient to motivate deployment of a trust infrastructure.
- A belief that characteristics of current PKI architectures and implementations make them unnecessarily difficult to deploy, and/or that those characteristics render them incapable of delivering value which alternate approaches could achieve.
- A belief that deployment of PKI technology intrinsically implies and enforces a higher assurance environment than is appropriate or cost-effective in many operational contexts.

A 2003 survey undertaken by the OASIS PKI Technical Committee [Hann03] on obstacles to PKI deployment and usage suggests a mix of factors spanning each of these categories. If increased PKI adoption is taken as a goal, the first interpretation suggests a strategy of promoting applications and usage modes that would make use of certificates. Existing PKI technologies would stand ready to satisfy the demand if and as it emerges. While incremental changes might remain necessary to satisfy integration requirements, fundamental PKI architectures could safely remain intact. Questions of candidate applications and usages for PKI technology are interesting and important, but lie outside this paper's scope.

The second and third interpretations imply criticisms of elements within the PKI technology base, and motivations to revisit and modify those aspects of PKI that are considered to be contentious or problematic. Different commentators have expressed concerns about different elements of PKI technology, and have proposed different alternatives as a result; the goal of this paper is to examine a range of perceived issues and suggested approaches,

not to assert that all are equally valid or appropriate. Following this introduction, we characterize various perceived problem areas. Then, we examine several proposed approaches, seeking to characterize them in terms of the goals that they address, and the properties and value that they offer. We conclude by assessing asserted problems, and the contributions that suggested solutions make towards those problems.

This paper focuses on architectural and functional aspects of PKI. It is not primarily concerned with encoding alternatives, such as choices between ASN.1 and XML representations for protocol objects. For purposes of discussion, we assume the following elements as aspects of the contemporary PKI baseline, and therefore do not consider them under the category of candidate future variations:

- Support for hierarchic and non-hierarchic trust models
- Support for certificate revocation via Certificate Revocation Lists (CRLs) and via basic on-line status query mechanisms such as OCSP
- Syntactic support within certificates for a range of name forms, such as X.500 Distinguished Names, Internet-form names within AltName extensions, and pseudonyms.

While particular enhancements can be considered within many of these areas, their general premises have been widely presented and adopted, so do not constitute qualitative shifts from current accepted practice.

## **2 Contentious Aspects of PKI**

In this section, we discuss several aspects of PKI technology and its operation that have attracted criticism and controversy.

### ***2.1 Difficulty in Retrieving Keys and Certificates***

To perform operations using public keys, those public keys must be available at the point where the operations are to be performed. In a conventional certificate-based PKI, this implies that a sender cannot encrypt a message for a recipient unless the recipient has already obtained a certificate and has made the certificate available to the sender (whether by direct transfer or posting on an accessible repository). If off-line operation is required, the appropriate certificates must be obtained in advance, when connectivity is available. Since large-scale directories have not become widely available to serve as certificate publication vehicles, interest has grown in approaches that enable public-key encryption to be performed without first satisfying these preconditions.

### ***2.2 Questionable Value of Certified Key Representations***

Certificates' usage practice reflects characteristics of environments for which they were originally developed, where it was considered inappropriate or impractical to rely on on-line availability of trusted servers. A primary goal of certificates' design was to represent keys and their bindings to named principals in an integrity-protected form, whose content could be stored safely on unprotected repositories or transferred across unprotected channels. Retrieval of a certificate requires that a suitable repository be available, but use of signed representations abstracts away the need to depend on that repository for security

properties other than availability. If, instead, keys are stored and retrieved from trusted servers, some of the rationale for representing them within signed certificate objects becomes superfluous. Channel-level mechanisms can protect a key from attackers while in transit between a server and a client, and can assure the client that it is receiving a key from a securely identified source.

### **2.3 Certificate Processing Complexity**

PKI technologies have been criticized as being difficult to integrate with the applications that could make use of their services, requiring significant PKI-specific security expertise on the parts of application writers and maintainers. Today's X.509 certificates, e.g., have evolved into complex structures, with processing semantics that are far from trivial; this is primarily a matter of the information they carry, although it also involves its representation and encoding. Formalization and simplification of these semantics may represent a valuable area for investigation.

Some of the complexity in certification results from a desire for a certificate to include a comprehensive set of ancillary information so that it can be used for off-line processing, without consulting other trusted entities on an interactive basis. Increasingly, however, PKI models are evolving to include on-line components, which can offer alternative information sources to complement the certificates themselves.

Revocation mechanisms have long been recognized as a complex element in PKI, and path construction also introduces complexity [Elle01]. Despite the design attention that has been paid to revocation, it appears today that only a relatively small proportion of accepted certificates are actually checked for revocation status on an ongoing and timely basis.

### **2.4 Costly Certificates**

Many assumptions about certificate usage have been based on a premise that certificates are expensive, and therefore that they can only be issued sparingly and infrequently. Some enrollment methods strive to provide confidence commensurate with high-value transactions and high-assurance client implementations, entailing high monetary costs and/or cumbersome registration processes. While this practice is appropriate for some types of technology (e.g., one-time placement of a user's long-term certificate into a smart card), and may be necessary to provide high levels of accountability, it need not be an intrinsic characteristic associated with the use of PKI methods. Imagine, by comparison, how computing might have developed if it had become accepted practice that an independent organizational authority needed to be consulted (and, possibly, paid) whenever a file was to be created. Most likely, only a subset of information, perhaps associated with a subset of critical users, would be deemed to warrant file representation. Other data would be stored and shared using different objects without the constraints associated with files. For a PKI, even when high levels of administrative assurance are not required, certification paradigms can be retained and adapted rather than developing or applying separate types of infrastructures to bind principals, keys, and attributes.

Dynamic issuance of certificates, which may be short-lived to avoid the need for separate revocation infrastructures, may allow new and innovative PKI models to be constructed.

In the Security Assertion Markup Language (SAML) [Male03], e.g., assertions bearing the Holder-of-Key confirmation method can take the form of signed objects carrying public keys, used to enable the corresponding private keys' holders to gain access to resources. Servers are expected to issue such assertions frequently, as needed to support authentication or resource access operations; no laborious procedures are required when an assertion is coined. Further, a number of on-line PKI key registration protocols (e.g., CMP [AdFa99], XKMS's X-KRSS [W3C03]) have been defined, which can provide the basis for interactive certification. The form of the resulting object, whether X.509, XML, or another format, need not imply or dictate the scope of procedural processing that is appropriate before the object is issued.

## **2.5 Problematic Cross-Domain Trust Management**

The prospect of applying PKI technology to establish trust across heterogeneous domains can be daunting, both in administrative and technical terms. Some PKI architectures have sought to provide a sufficient basis to allow parties in different jurisdictions to engage in high-value transactions with one another, without prior shared knowledge beyond that manifested in the PKI. Few other technologies have attempted such ambitious goals, and it is debatable whether other approaches would necessarily achieve greater success in solving such a fundamentally challenging problem. In cases where the level of required assurance can be constrained, it may become easier to achieve (and benefit from) PKI-enabled interoperability.

PKI technologies can be applied to manifest trust relationships rooted at remote entities. Some (e.g., [DoE102]) have argued, however, that users' trust is primarily local, and should be based on direct personal knowledge of other individuals. If this premise is accepted, reliance on remote roots is not considered practical or useful, and the ability to represent such trust relationships offers only irrelevant complexity.

Meaningful algorithmic translation of policies across domain boundaries is a significant challenge; often, the mapping between different organizations' policy elements can be based on administrative practices and interpretations that are difficult to encode. Management of inter-domain validation and trust relationships within a relatively small set of entities (e.g., bridge CAs, domain-level Delegated Path Validation (DPV) servers interacting with their peers representing other domains) may help to contain and simplify some aspects of the problem.

## **2.6 Naming Semantics**

Naming plays an important role in PKIs, as public keys are typically bound to named entities. Conventional PKIs have been criticized for seeking to manifest a global naming structure that some view as fundamentally unrealistic. As with trust, some view naming as intrinsically local; further, given duplications among human individuals' names, ambiguities can arise in identifying a particular person based on his or her location in a distributed namespace. In some alternate approaches, e.g., SDSI [RiLa96], entities are named in a relative manner extending from one principal, and then can be linked to other principals through intermediary hops.

Another aspect of PKI entity names is the degree to which a name form matches or resembles names that people and software use on a regular basis. This has a direct bearing

on how useful the name is to the user or application that is trying to accomplish a security goal. Some PKIs – such as Pretty Good Privacy (PGP) [Call98] and the DNS security extensions (DNSSEC) [East99] – employ name forms that match their environments (or, rather, they adopt the name form of their environment). X.509 is an example of a PKI that started out adopting the name form of its environment (X.500 Distinguished Names), but then grew to accommodate application-specific names (through the Alternative Name extensions). A SDSI “well-defined” name – one that links a local name space to a particular principal, such as (using SDSI’s “syntactic sugar”) `jim's john's joe's jack` – is only meaningful to the SDSI PKI. However, each individual local name is an arbitrary string, and so can be meaningful to an application. For example, `10.1.1.1` might be a local SDSI name assigned to a VPN server whose IP address is, presumably, `10.1.1.1`. PKIs with PKI-specialized name forms require applications to translate between their native name form and the PKI's, a process that can be error-prone and introduce security risks.

PKI	Name Locality	Name Form Application	Utility PKI
PGP	Low	High	Low
DNSSEC	Low	High	High
X.509	Low	High	Low
SPKI/SDSI	High	Medium	High

Table 1 - PKI naming properties

A third property of PKI names is the degree of utility that the name has to the PKI itself. By "degree of utility" we mean the efficiency with which the PKI can use the name to obtain and validate a public key. PKIs provide keys by discovering and validating paths between entities, and so the PKI-efficiency of a name can be measured by the amount of path information that it encodes. Well-defined SDSI names are an extreme example of a name form that is almost entirely devoted to expressing path information, so much so that a (non-global) SDSI name is usually only meaningful to a single entity. DNSSEC names also encode a large amount of path information. In contrast, PGP names are email addresses, which are completely devoid of any PGP PKI path data. X.509's names – all of them – also contain no X.509 path information whatsoever.

Table 1 summarizes the naming properties for various PKIs. SDSI scores highly for name form PKI utility because of its well-defined names, but only moderately for application utility because although an individual local name can be an application-meaningful string, there are no conventions for an application to reliably extract a meaningful local name from a SDSI certificate. A VPN client, for example, has no way to tell that the `10.1.1.1` name in a SDSI certificate is supposed to be the IP address of a VPN server.

Recent PKI proposals have emphasized certificate processing and cryptographic methods rather than naming. A viable naming strategy seems to be a factor in a PKI's success, but it is not clear what combinations of properties (per Table 1) offer most value. Naming strategies do appear to require some consideration, and yet they remain relatively unexplored. Some of the questions that arise include:

- Are there any other useful naming properties?
- Is it necessary or desirable to rank highly in all of these properties?
- Have approaches to naming had an impact on PKI deployment? We note, for example, that an X.509 certificate in fact has two names – Issuer and Subject

- which together provide a small amount of path information. Would more (or less) path information in the certificate help or hinder widespread deployment of an X.509 PKI?

## **2.7 Use with Insecure Clients**

Some PKI architecture premises were developed in anticipation of widespread security features at user clients, e.g., smart cards encapsulating users' private keys and cryptographic processing capabilities so that the keys need never be exposed elsewhere. Such implementations are particularly desirable when the keys mediate access to particularly sensitive data or resources, or when strong accountability (i.e., a non-repudiation service) is tied to their use. While such environments are gradually becoming more common (as with use of SIMs and other cards), most candidate PKI user applications continue to reside on platforms that offer limited security. From an attacker's viewpoint, the strength of a cryptographic algorithm can become irrelevant if its keys can be obtained by attacking a weak platform. Where high assurance is required, these arguments motivate approaches that perform cryptographic processing in other entities, whether protected devices or shared services, and/or distribute the processing with such entities.

There are many cases, however, where the assurance level of commercial platforms is an adequate basis to support useful, interoperable security. Use of PKI need not also imply use of specialized, higher-security technologies by clients; higher assurance requirements may be warranted at CAs, as misuse of a single CA private key can compromise an entire community. Today, it is common practice to store user keys in a password-encrypted form. It is arguable that passwords used to unlock private keys may warrant higher quality or tighter protection than other passwords, as the keys they release can enable direct authentication to multiple entities rather than just to a single system, but user convenience may conflict with such measures.

## **2.8 Privacy Compromises**

It has been observed, e.g., in [Bran99], that conventional PKI is unfriendly to privacy, as its certificates provide persistent, widely visible linkages between keys and principal identifiers. This property is appropriate in contexts where authorizations or signatures depend on individuals' authenticated identities, but not all possible uses of public-key technology fit this model. Even if data messages are encrypted, patterns of certificate acquisition and usage can reveal identities of principals and their communicating peers; a certificate validation server could be particularly well placed to collect such information. Certified pseudonyms can provide a partial countermeasure, but do not satisfy all privacy goals; if a fixed pseudonym is used to represent a principal to multiple sites for an extended period, the sites can use it as the basis to collect an extensive behavior profile which may then be associated with an individual.

Use of X.509 certificates to hold principal attributes other than identifiers has been proposed and considered for some time, recently in [FaHo02], though has not yet achieved wide adoption. Attribute statements within SAML assertions are another form of attribute representation within a signed object corresponding to a principal. Both have the property of disclosing an aggregate set of attributes to their certifier and to the parties that rely

on the certified object, even if not all of these entities necessarily require the full set of information.

## **3 Proposed Approaches**

In this section, we examine approaches that have been proposed as extensions or alternatives to conventional PKI technologies, addressing one or more of the concerns identified in the preceding section.

### **3.1 IBE and Related Work**

The concept of Identity-Based Encryption (IBE) has been considered in the cryptographic community for some time, and recent work has yielded a variety of methods realizing variations on the concept. Some, but not all, approaches in this group allow a sender to prepare a protected message for a recipient without first obtaining a certificate for the recipient. This section considers some of their properties.

#### **3.1.1 Identity-Based Encryption**

IBE, surveyed in [Gagn03], enables senders to encrypt messages for recipients without requiring that a recipient's key first be established, certified, and published. The basic IBE paradigm allows a sender to determine the key to be used to encrypt for a particular recipient based on the recipient's identifier; the recipient derives the corresponding decryption key through interaction with a Private Key Generator (PKG) system. While the sender must determine the PKG corresponding to a particular recipient, and must obtain domain-level parameters associated with that PKG, it need not obtain information specific to an individual recipient before encrypting a message. The basic IBE approach implies intrinsic key escrow, as the PKG can decrypt on behalf of the user. Variant approaches ([AlPa03] [Gent03]) cited below apply some aspects of IBE, but seek to avoid the escrow characteristic.

#### **3.1.2 Certificateless Public Key Cryptography**

This approach, proposed in [AlPa03], incorporates IBE methods, using partial private keys so the PKG can't decrypt on behalf of the user. These are combined with secret information held by the recipient, yielding a public key that the recipient can publish and/or transfer directly, but for which no certification is required. Would-be senders must, however, first obtain that key through some means in order to encrypt a message for a recipient. Publication of a key for this method may not prove significantly easier than publishing a conventional PKI certificate. In fact, the publication problem could become significantly worse, since use of the approach might imply a need for frequent republication in lieu of a revocation mechanism.

#### **3.1.3 Certificate-Based Encryption**

This approach, proposed in [Gent03], incorporates IBE methods, but uses double encryption so that its CA can't decrypt on behalf of the user. A sender must obtain a recipient's certificate in order to encrypt a message for a recipient. In order for a recipient to decrypt successfully, he/she must have both a current CA-issued certificate and a personal secret key; use of IBE methods in certificate generation means that the same certificate used by

the sender to encrypt is also used by the recipient as part of the decryption process. Frequent certificate updates are performed, so that senders need not separately check revocation status of the certificates they obtain.

### **3.2 PKI Augmented with On-Line TTP**

Some properties similar to those of IBE can be achieved by augmenting conventional PKI with an on-line trusted third party (TTP) system. Two classes of TTP-based operations can be considered:

- Encryption using a TTP's public key rather than one associated with an individual recipient; in this case, a recipient could request that the TTP perform decryption services on his/her behalf, or a message could be routed to the TTP which would then decrypt it and forward the result to the recipient. This eliminates the need for recipients to register individual key pairs, and for senders to obtain per-recipient keys; it implies that the TTP can decrypt all recipients' traffic and requires involvement by the TTP in order to process each of their messages. [DeOt01] provides examples and discussion of this type of approach.
- Encryption using an individual recipient's public key, which the sender would request from the TTP. For already-registered recipients, a TTP (such as that suggested in [Dier03]) would provide their existing keys or certificates. Additionally, such a TTP could revoke keys or certificates by removing them from its store. If no public key or certificate existed for the recipient at the time of the request, the TTP would generate one dynamically, provide the public component to its requester, and make the corresponding private key available to the recipient. In this model, the TTP's possession of recipients' private keys need not be more than temporary in nature, pending their retrieval by the corresponding recipient.

The second type can be considered as an example of a general class which has previously been considered in various contexts but has not become part of the PKI mainstream, that of "on-the-fly PKI" approaches where certificates are signed dynamically as needed rather than being generated by a CA in advance as a prerequisite to secure operation. Such certificates and the keys they certify can be short-lived, enabling particular operations or use of a session while becoming disposable thereafter. Some other examples include the delegation certificates that represent login sessions within Digital Equipment Corporation's Distributed System Security Architecture (DSSA) as proposed ca. 1990 [GaMcD90], and recent IETF-PKIX contributions on proxy certificates [Tuec03].

### **3.3 Distributed Computation**

Methods have been developed (see, e.g., [Gold02]) that distribute cryptographic operations so that the cooperative contribution of a number of entities is required in order to perform an operation such as a signature or a decryption. Use of such measures could help to ameliorate the risks associated with insecure client platforms; even if such a client's keys were compromised, they would be insufficient to impersonate the client's associated user.

Analogous to the case with IBE, some similar properties can also be achieved without specialized cryptography by holding a user's keys at a server, which would perform operations on behalf of the user upon receipt of an authenticated request. This strategy can take advantage of tighter protection at servers vs. clients, but implies that the users must fully trust the servers to apply their keys appropriately.

### **3.4 Alternative Validation Strategies**

PKI's original Certificate Revocation List (CRL) mechanisms implied significant storage, communications bandwidth, and processing overhead, yet could only provide revocation with significant time latency. Newer on-line approaches, such as OCSP, SCVP, and XKMS, address many of these concerns, but introduce requirements for trusted on-line servers to process certificates and for connectivity between the servers and their relying parties. Their effective revocation latencies can vary, as a result of caching and when information updates are available only on a periodic basis. These approaches' capabilities, and the extent to which clients must trust the servers, increase as the scope of server-based processing extends from revocation checking on single certificates to acquisition and validation of full certification paths, and from independent, self-contained validation servers to distributed networks of cooperating validators. More broadly, however, the extent of trust required should correspond to the value of the information that the underlying certificates protect. Further discussion of validation alternatives and their prospects and implications can be found in [BrLi02].

Hash-tree approaches (e.g., [Mica02] [NaNi98]) have been proposed, offering compact, protected representations of the status of large numbers of certificates. Their value is most apparent for PKIs operating at extremely large scale; in smaller contexts, such as within typical enterprises, their benefits relative to CRLs appear less compelling. Like CRLs, they reflect certificate status information only at fixed intervals, rather than with the immediacy that on-line status queries can offer.

Levi and Caglayan [LeCa00] propose the concept of "nested certificates" in order to avoid some of the performance burdens associated with verification of long certification paths. Several variations are suggested, but a general premise is that a hierarchy's higher-level CAs certify not only their immediate descendants but also directly certify members of more distant generations. While this approach can indeed reduce the number of certificates in a validated path, it appears to suffer from a serious flaw. Among other reasons, CA hierarchies are constructed in order to distribute certification responsibilities, and to place them in hands close to the principals they certify. In a condensed hierarchy, higher-level CAs would need to be involved in enrollment of remote generations, and potentially to generate very large numbers of certificates. In the limit, a CA hierarchy could be flattened to a single CA, making any hierarchy below it moot, but such an approach is unlikely to be attractive from a technical or policy perspective.

### **3.5 Key Servers**

Given today's generally high level of connectivity, and widespread interest in simplifying client-side operations, an emerging approach is to use servers to perform some, or all, certificate processing. Clients would delegate certificate path discovery and/or validation to a trusted server (see [PiHo02]).

DPV, in particular, changes the basic PKI model. A DPV server assumes the primary responsibilities of a traditional CA, from the client's perspective. That is, the client relies upon the server to ensure correct correspondences between principals and their public keys.

This approach has implications for assurance and availability, especially when a DPV server relies on other DPV servers (see [BrLi02]). However, once the premise of trusting an on-line server for certificate retrieval and validation is accepted, it is only an incremental step to relying on the server to provide the bare key over a secure channel – eliminating the need for the client to process certificate formats entirely. Such an approach is one of the models supported within XKMS's X-KISS [W3C03].

The full impact of delegating key validation and acquisition to servers has yet to be investigated. The benefits to PKI client applications for smaller, simpler code are apparent, but it is not yet clear what effects delegated key servers will have on a PKI's policies and procedures, or what levels of assurance are enabled (or disabled).

### **3.6 Privacy Protection**

Some PKI privacy implications can be ameliorated by reducing the amount of principal-related information bound within a single certificate or other signed object. Certified pseudonyms can easily be supported, and are appropriate and sufficient in many operational contexts. Further, use of attribute certificates (ACs) can offer privacy advantages over placement of attributes within public-key identity certificates (PKCs). Even in the common case where an AC is bound to a PKC for use, implying a linkage to the PKC within the AC, the PKC's contents need not disclose all of the ACs that may be used with it. This modularity allows the attributes within ACs to be disclosed selectively, when needed in order to support a particular access request, and to remain confidential otherwise. To take advantage of this capability, it is desirable for accessors to present ACs selectively along with requests rather than posting them for general access within a directory or other repository.

Use of on-line certificate validation services introduces the prospect of user tracking, if the validation service can identify the set of locations from which a certificate's status is queried. Aggregation and/or anonymization of status requests can help to mitigate this concern.

Stefan Brands, in [Bran99], proposes cryptographic certification techniques which address privacy goals outside the scope of traditional PKI models, and which imply different assumptions and paradigms for PKI protocols and interactions. Brands' techniques seek to allow certificate holders to disclose certified attributes selectively in a general manner, and to limit the extent to which presentation of certified attributes can be proven to third parties by recipients. Cryptographic blinding is used for certificate issuance, so that not all of the attributes represented within a certificate need be visible to a particular issuing CA. These approaches can provide privacy assurance unavailable in conventional PKIs, particularly in terms of constraining the scope of trust that a certified user must place in a CA and of countering use of certificates as a means to aggregate data. Their operational models would require changes in certificate-based protocols, one factor which would likely complicate their deployment.

## 4 Conclusions

Any concrete system can suffer in comparison with a hypothetical, ideal alternative. PKI has been a particularly attractive target, perhaps partly because it has sometimes been perceived and promoted as a general panacea, intended to solve even organizational issues outside the realm of technology, rather than as a technical answer to clearly understood and practically achievable requirements. Variations to many aspects of PKI are possible and worthy of consideration, but an appropriate comparison between practice and proposal requires a specific alternative and an understanding of its impact on the system as a whole.

Certificates have been criticized for a variety of reasons, particularly:

- Processing complexity and overhead, including both the contents of certificates and the usage of signed representations to carry those contents; many of these characteristics derive from design assumptions which presumed off-line certificate processing without reliance on trusted servers, and use of such servers may allow significant simplifications.
- Association with operational models that imply high costs for certificate issuance; here, the use of a signed key-bearing object should properly be distinguished from a particular type of deployment. Public-key methods can be used to construct a wide variety of useful approaches with different assurance, semantics, and dynamics.

PKI has also been criticized on the basis that it fails to render the problems of securely interconnecting different entities and trust domains simple. These problems are fundamentally difficult, for organizational as well as technical reasons. Few proposals outside the realm of PKI have attempted to satisfy these concerns comprehensively, though trust management research activities [Blaz99] have proposed various supporting mechanisms. Generally, PKIs' trust management capabilities should be evaluated in terms of their supporting contributions to distributed security, rather than against an expectation that all such requirements should be satisfied solely by PKI or any other technology.

Much cryptographic research activity has concerned forms of IBE, applied to avoid the need for senders to retrieve certificates from repositories. Unfortunately, many proposed alternatives substitute different publication requirements, or introduce implicit key escrow properties. Other computational methods can distribute processing, mitigating some of the impact of key compromise at weakly protected clients. These cryptographic innovations provide elegant approaches, but many of their properties can also be achieved by using trusted third parties with conventional cryptographic algorithms.

Fundamentally, PKIs exist to provide public keys that correspond to principals, in a fashion enabling other parties to rely on their correspondence. This function is an essential basis on which to construct secure distributed computing environments, and necessarily implies some form of infrastructure. Many PKIs seek to provide high levels of technical and procedural assurance, particularly at CAs, but some of these measures may not be necessary for environments where the ability to communicate with at least some level of protection takes precedence over especially strong security guarantees. Naming is a cen-

tral element in PKI, and further research focused on aspects of alternate naming methods may warrant attention.

Certificates are a convenient, self-sufficient means of representing keys, but their use may become superfluous in server-centered environments. Further, new PKI models can evolve based on signed key-bearing assertions; these objects can provide the same functions as certificates, but are emerging unbounded by existing assumptions about how certificates must be created, processed, and managed. Generally, it seems that PKI suffers today from a perception that it can assume only a particular, monolithic form; to satisfy a broad range of applications and environments, it must be possible for its underlying methods to be composed and applied in a variety of ways.

## 5 Acknowledgment

The authors would like to acknowledge this paper's anonymous reviewers for comments helping to improve its final version.

## 6 References

- [AdFa99] C. Adams, S. Farrell, "Internet X.509 Public Key Infrastructure Certificate Management Protocols", Internet RFC-2510, March 1999.
- [AlPa03] S. S. Al-Riyami and K. G. Paterson, "Certificateless Public Key Cryptography", IACR Cryptology ePrint Archive paper 2003/126, 2 July 2003.
- [Blaz99] M. Blaze, J. Feigenbaum, J. Ioannidis, A. D. Keromytis, "The Role of Trust Management in Distributed System Security", in *Secure Internet Programming: Issues in Distributed and Mobile Object Systems*, Springer-Verlag Lecture Notes in Computer Science State-of-the-Art Series, pp. 185-210, Berlin, 1999.
- [Bran99] S. Brands, "Rethinking Public Key Infrastructures and Digital Certificates – Building in Privacy", PhD Dissertation, University of Utrecht, October 1999.
- [BrLi02] M. Branchaud, J. Linn, "Extended Validation Models in PKI: Alternatives and Implications", 1<sup>st</sup> PKI Research Workshop, Gaithersburg, MD, April 2002.
- [Call98] J. Callas, et al., "OpenPGP Message Format", Internet RFC-2440, November 1998.
- [Clar01] R. Clarke, "The Fundamental Inadequacies of Conventional Public Key Infrastructure", Proceedings, ECIS'2001, Bled, Slovenia, June 2001. Available at <http://www.anu.edu.au/people/Roger.Clarke/II/ECIS2001.html>. (Date of access: 26 November 2003.)
- [DeOt01] T. Dean, W. Ottaway, "Domain Security Services Using S/MIME", Internet RFC-3183, October 2001.
- [Dier03] T. Dierks, "Re: Fwd: [IP] A Simpler, More Personal Key to Protect Online Messages". Message posted to Cryptography electronic mailing list, ar-

chived at <http://www.mail-archive.com/cryptography@metzdowd.com/msg00409.html>. (Date of access: 4 December 2003.)

- [DoEl02] S. Dohrmann, C. Ellison, “Public-key Support for Collaborative Groups”, 1<sup>st</sup> PKI Research Workshop, Gaithersburg, MD, April 2002.
- [East99] D. Eastlake, “Domain Name System Security Extensions”, Internet RFC-2535, March 1999.
- [Elle01] Y. Elley, et al., “Building Certification Paths: Forward vs. Reverse”, NDSS-01, San Diego, 2001.
- [ElSc00] C. Ellison, B. Schneier, “Ten Risks of PKI: What You’re Not Being Told about Public Key Infrastructure”, Computer Security Journal, Vol. XVI, No. 1, 2000. Available at <http://www.schneier.com/paper-pki.html>. (Date of access: 4 March 2004.)
- [FaHo02] S. Farrell, R. Housley, “An Internet Attribute Certificate Profile for Authorization”, Internet RFC-3281, April 2002.
- [GaMcD90] M. Gasser, E. McDermott, “An Architecture for Practical Delegation in a Distributed System”, Proceedings, IEEE Computer Society Symposium on Research in Security and Privacy, Oakland, CA, May 1990.
- [Gagn03] M. Gagné, “Identity-Based Encryption: a Survey”, RSA Laboratories Cryptobytes, Vol. 6, No. 1, Spring 2003.
- [Gent03] C. Gentry, “Certificate-Based Encryption and the Certificate Revocation Problem”, EUROCRYPT 2003, LNCS 2656, pp. 272-293, 2003.
- [Gold02] O. Goldreich, “Secure Multi-Party Computation (Final (Incomplete) Draft Version 1.4)”, 27 October 2002. Available at <http://www.wisdom.weizmann.ac.il/~oded/pp.html>. (Date of access: 19 December 2003.)
- [Gutt02] P. Guttman, “PKI: It’s Not Dead, Just Resting”. Available at <http://www.cs.auckland.ac.nz/~pgut001/pubs/notdead.pdf>. (Date of access: 5 March 2004.)
- [Hann03] S. Hanna, ed., “Analysis of August 2003 Follow-up Survey on Obstacles to PKI Deployment and Usage”, OASIS PKI Technical Committee, 1 October 2003. Available at <http://www.oasis-open.org/committees/pki/pkiobstaclesaugust2003surveyreport.pdf>. (Date of access: 4 March 2004.)
- [LeCa00] A. Levi, M. Caglayan, “An Efficient, Dynamic, and Trust Preserving Public Key Infrastructure”, Proceedings, IEEE Computer Society Symposium on Research in Security and Privacy 2000. IEEE, Piscataway, NJ, USA, pp. 203-214.
- [Male03] E. Maler, P. Mishra, R. Philpott, eds. (2003), “Assertions and Protocol for the OASIS Security Assertion Markup Language (SAML) V1.1.” OASIS Standard.

- [Mica02] S. Micali, “Novomodo: Scalable Certificate Validation and Simplified PKI Management”, 1<sup>st</sup> PKI Research Workshop, Gaithersburg, MD, April 2002.
- [NaNi98] M. Naor, K. Nissim, “Certificate Revocation and Certificate Update”, 8<sup>th</sup> USENIX Security Symposium, San Antonio, January 1998.
- [PiHo02] D. Pinkas, R. Housley, “Delegated Path Validation and Delegated Path Discovery Protocol Requirements”, Internet RFC-3379, September 2002.
- [RiLa96] R. Rivest, B. Lampson, “SDSI – A Simple Distributed Security Infrastructure”, 30 April 1996.
- [Tuec03] S. Tuecke, V. Welch, D. Engert, L. Pearlman, M. Thompson, “Internet X.509 Public Key Infrastructure Proxy Certificate Profile”, work in progress, IETF PKIX working group, 2003.
- [W3C03] World Wide Web Consortium, “XML Key Management Specification (XKMS)”, Version 2.0, W3C Working Draft, 18 April 2003.