UIC John Marshall Journal of Information Technology & Privacy Law

Volume 15
Issue 2 *Journal of Computer & Information Law - Winter 1997*

Article 1

Winter 1997

Eggs in Baskets: Distributing the Risks of Electronic Signatures, 15 J. Marshall J. Computer & Info. L. 189 (1997)

Benjamin Wright

ARTICLES

EGGS IN BASKETS: DISTRIBUTING THE RISKS OF ELECTRONIC SIGNATURES

by BENJAMIN WRIGHT†

I. INTRODUCTION

Electronic commerce unveils questions about how to sign or legally prove electronic documents.[1] Evidence that a person approved a particular electronic document might be gathered many different ways. This article evaluates public-key cryptography and pen biometrics, which are two emerging methods for proving electronic signatures.[2]

The signing of a document is a social event, not a scientific event.[3] It

---

† Benjamin Wright, a Dallas-based attorney, is special counsel to PenOp, Inc. (www.penop.com; tel: (800) 286-4137). He is also author of THE LAW OF ELECTRONIC COMMERCE: EDI, E-MAIL, AND INTERNET (Little, Brown & Company 2d ed. 1995). Mr. Wright is not an engineer, a computer scientist, or a forensics expert. The transaction of commerce is inherently risky, and nothing published by the author advises which level of risk is appropriate for you. For more articles on the legality of electronic commerce, see <http://infohaus.com/access/by-seller/Benjamin_Wright>.

1. *See* Christy Tauhert, *Electronic Signatures Software Emerges to Authenticate Electronic Commerce Transactions*, 33 BANK SYSTEMS & TECH. 7, July 1996, at 16 (noting that a challenge to electronic commerce is authenticating that the persons involved in a given transaction are actually who they purport to be).

2. *See, e.g.*, Larry Donovan, *Secret Identities—Two Systems to Combat Signature Fraud in Cyberspace Are Outlined by Larry Donovan*, FIN. TIMES, Feb. 6, 1996, at Technology 11 (discussing possible mechanisms for companies to secure business transactions in electronic commerce); *see also* Christy Tauhert, *Signatures in Cyberspace: Closer to Reality*, 7 INSURANCE & TECH. 7, July 1996, at 26 (stating that questions remain as to the security and legality of electronic signatures). This article does not consider whether any given method will be considered in law as a signature. For more on that topic, see BENJAMIN WRIGHT, THE LAW OF ELECTRONIC COMMERCE (1991).

3. *See, e.g.*, Wendy R. Leibowitz, *Technology and the Law Meet Online Commerce*, BUS. WATCH, Aug. 5, 1996, at B1-2 (stating that a digital signature authenticates the identity of the person who sent the electronic message, and verifies that the message was not altered).

is an act in which an individual—Alex—evinces approval of the document so that someone else—Bob—can perceive that approval, understand it, and later prove it to other people.[4] However, the legal binding of Alex to the document is never a perfectly reliable process. Whatever evidence exists to support the bond is subject to challenge. In other words, signing documents involves risk.

II. TRADITIONAL INK AND PAPER STRATEGY

Many risks afflict the traditional signing of a paper document. Because no standard method exists for signing in ink,[5] Alex is not taught or required to sign documents in a forensically reliable way. Alex is free to sign in any way he chooses,[6] and to change his signature from minute to minute. For any given signing, Alex can select any strange and indecipherable scribble as his signature. Whether any given document signed by Alex does or does not contain Alex's usual, verifiable signature is, for all practical purposes, a secret. Rarely is Alex's signature compared against specimens to confirm authenticity.

One risk that afflicts the traditional signing of paper documents is forgery. Forensic science cannot guarantee that any given ink signature can be verified. Scientists can only offer an educated opinion as to whether the signature is authentic, and they can do so only under the right circumstances. This includes, for example, the availability of several good specimen signatures.

Other risks exist that impede the linking of Alex to a given paper document. For example, in a multiple page document, one or two of the pages might be switched after the document is signed. Furthermore, the document may be organized in an ambiguous or confusing way, so that an observer cannot discern for certain which parts of the document Alex approved and which parts he did not.

These risks mean that in the event of a dispute,[7] it is not always easy to tie Alex to specific words in a paper document. When Alex signs a document and gives it to Bob, Bob is not guaranteed that he will later be

4. *See id.*

5. *See, e.g.*, RESTATEMENT (SECOND) OF CONTRACTS § 134 (1982). "The signature to a memorandum may be any symbol made or adopted with an intention, actual or apparent, to authenticate the writing as that of the signer." *Id.*

6. *See, e.g., Id.* cmt. a (noting that a signature can be the signer's name, initials, thumbprint, or arbitrary code sign).

7. In practice, disputes over the authenticity of commercial documents are rare. Of the many billions of commercial documents created every year, the authenticity of only a tiny fraction of the total is seriously contested in court. Among the reasons for this are that most people are happy with their commercial transactions most of the time, and the facts and circumstances surrounding the documents (including the signatures, but also including the context and content of the documents) tend to show their origin and authenticity.

able to prove Alex's signature. Alex, however, might raise any number of objections to repudiate the document. Conversely, Alex has no guarantee that he can repudiate a document that he in fact did not sign.

Under American law, the burden of proving that Alex did sign the document is normally on Bob.[8] This burden motivates Bob, at the outset of the transaction, to seek evidence of Alex's responsibility from things other than simply Alex's signature. This may entail Bob asking Alex to acknowledge his signature before a notary. More commonly, this burden means that Bob establishes a relationship with Alex in which they exchange feedback—for example, Bob might ask for partial advance payment, or he might send acknowledgments to Alex's independently verified address. Feedback reduces the risks to Bob.[9]

The myriad risks associated with a paper and ink signing are distributed across a number of features of the signing ritual—Alex's secret choice whether to use his usual signature, content of the signed document, facts external to the document (such as interaction between Alex and Bob) that place the document in historical context, the competence of the person who opines on the signature's authenticity, and so on. In other words, the eggs are spread into many baskets. No single egg is highly reliable or highly important.

In a dispute over the authenticity of a document, a court does not look at the signature in a vacuum. Rather, it considers all the relevant facts and circumstances—the historical context of the document and all ambient clues, including corroborating records and testimony that might bear on the authenticity question.

Just as risks plague the authentication of paper documents, so too will they plague the authentication of electronic documents. To expect perfect binding of an individual like Alex to the words of an electronic document is not realistic.

To bind Alex to his electronic words, inventors might craft any number of strategies. One such strategy that has attracted attention is embodied in the Digital Signature Act adopted by the Utah Legislature in 1995 and amended in 1996 ("Utah Act").[10] The Utah Act contem-

8. *See* Douglass v. Hustler Mag., Inc., 769 F.2d 1128, 1132 (7th Cir. 1985) (demonstrating the burden on the defendant-company to prove non-forged documents); *see also* McCormick on Evidence § 221, at 691 n.13 (3d. ed. 1984).

9. *See "Legal Identity and Signatures on the Information Highway,"* a component to Benjamin Wright, The Law of Electronic Commerce (2d ed. 1996).

10. Utah Code Ann. § 46-3-101 (1996). Other states have also proposed or drafted legislation dealing with electronic signatures in one way or another. *See also, e.g.*, Ariz. Legis. Serv. Ch. 213 (H.B. 2444) (West 1996); Cal. Govt. Code § 16.5 (West 1996); 29 Del. Laws Ch. 509 (S.B. 458) (1996); 1996 Fla. Sess. Law Serv. 96-224 (S.B. No. 942) (West); 1996 Haw. Sess. Laws 203 (S.B. No. 2401); 1996 Wash. Legis. Serv. Ch. 250 (S.B. 6423) (West).

plates the use of public-key cryptography.[11]

III. PUBLIC-KEY CRYPTOGRAPHY DESCRIBED

Public-key cryptography provides a mathematical scheme for arranging computer data[12] so that the data's integrity and origin can be proven.[13] Public-key cryptography involves the use of two keys,[14] a private key and a public key, which are assigned to a user (Alex).[15] Each key bears a complex mathematical relationship to the other.[16] As the name suggests, the private key is intended to be kept secret, so that only Alex can access that key.[17] The public key, however, is not intended to be kept secret.[18] It is published so that outsiders may obtain it and use it.[19]

Suppose, for example, that Alex wants to electronically sign a document for Bob in a way that confirms the document was really signed by Alex. After viewing the document's content, Alex would use his private key and a cryptographic program to attach a "digital signature" to the document. The digital signature is (generally) a short unit of data that bears a mathematical relationship to the data in the document's content.[20]

Bob can then confirm the document's authenticity by using Alex's public key and a cryptographic program. If the document was not altered between the time Alex signed it and the time Bob confirmed, then the program informs Bob that the document was signed by someone possessing Alex's private key.[21] Bob may then infer that Alex did sign the

---

11. UTAH CODE ANN. §§ 46-3-102, 46-3-201 (1996).

12. Computer data examples include electronic expense vouchers or medical records.

13. Public-key cryptography is but one of many tools in the world of cryptography. Even within the field of public-key cryptography, many specific technologies exist. For understanding the issues discussed here, one may simply consider public key as a single, generic technology. Depending on how they are implemented, various public key technologies can perform some or all of the functions described here. Cryptography is a complex topic. Any reader who seeks to achieve certain results from cryptography should consult a competent professional. For more on cryptography, see BRUCE SCHNEIER, APPLIED CRYPTOGRAPHY (John Wiley & Sons 1994).

14. These two keys are special strings of data.

15. *See, e.g.*, M.A. Stapleton, *Panel: Law Needed on 'Digital Signatures,'* CHI. DAILY L. BULL., Sept. 10, 1996, at 1, 18 (discussing private and public keys).

16. *See* Charles R. Merrill, *What Lawyers Need to Know About the Internet, A Cryptography Primer*, 443 PLI\PAT 187, 191-92 (1996) (discussing extensively the cryptography system).

17. *Id.*

18. *Id.*

19. *Id.*

20. *See* Donovan, *supra* note 2, at 11 (stating that a digital signature is formed by an algorithm or mathematical procedure).

21. *See* Merrill, *supra* note 16, at 192.

document.[22] If the cryptographic program cannot verify the document's signature with Alex's private key, then Bob infers either that the document was not signed with Alex's private key or that the document was altered after signature.[23]

Public-key cryptography can be used in endlessly creative ways. The Utah Act contemplates using public-key cryptography in one particular way, which this article refers to as the "Utah Strategy."[24]

IV. THE UTAH STRATEGY

One of the risks in signing with public-key cryptography is that the person using the public/private key pair might not be the right person. A person claiming to be Alex may in fact not be Alex. To alleviate this risk, the Utah Strategy imposes an elaborate scheme for binding Alex to a particular public/private key pair.

First, a CA would be licensed by state government to ascertain the identities of people like Alex and link them to their key pair.[25] When the CA confirms Alex's identity and his control of the requisite key pair, the CA must ask him to use a new "distinguishing name," a computer code that labels Alex, so that the nominal link between Alex and his key pair is unmistakable.[26] After the CA confirms Alex's identity, distinguishing name and key pair, the authority then issues a certificate confirming that Alex is associated with the public half of the key pair.[27] The Utah Act then obligates Alex not to allow his private key to fall into the hands of someone else.[28] If Alex neglects his obligation, he will find it difficult to avoid responsibility for documents signed with the key, even if he did not approve them.[29]

The Utah Strategy entails the keeping of secrets. Alex must keep his private key secret.[30] However, because Alex will not be able to remember his key, he will have to store it on a computer device such as a

22. *See* Merrill, *supra* note 16, at 192.

23. Merrill, *supra* note 16, at 189, 192.

24. UTAH CODE ANN. § 46-3-101 (1996). The Utah Act is an admirable intellectual work; within limits, the Act may support other strategies.

25. *See* UTAH CODE ANN. § 46-3-201 (1996).

26. *See id.* § 46-3-301.

27. The requirement for a distinguishing name appeared in the 1995 version of the Utah Act. Although it was removed in the 1996 amendment, it re-emerged in the Administrative Rules, Utah Administrative Code R154-10 published as of Oct. 1, 1996 at URL: <http://www.state.ut.us/ccjj/digsig/>.

28. UTAH CODE ANN. § 46-3-305 (1996).

29. Such documents could include divorce agreements, demands for the withdrawal of money from Alex's bank account, or any other legal instrument.

30. *See* UTAH CODE ANN. § 46-3-305.

"smart card"[31] and then keep the device in a safe place. In turn, the vendors of smart cards will need to keep secrets. If the vendors fully disclose methods used to control the private key, then it becomes easier for an attacker to steal the key.

Thus, strictly speaking, public-key cryptography does not reduce risk in the signing of an electronic document. Rather, public-key cryptography transfers risk. Public-key cryptography can be very effective in showing whether a particular document was signed with a certain private key. However, this transfer of risk does not result in the elimination or even in the reduction of risk. Risk simply shifts onto the private key. That key becomes the object of any criminals who want to cheat Alex or Bob. They will try tricking Alex into revealing or temporarily surrendering control of the key. They will endeavor to compromise the software that controls the key and its functions. Or they will steal and unlock the device in which Alex stores his key, such as a smart card. If under the Utah Strategy millions of people are assigned smart cards containing valuable private keys, then society can expect an underground industry of criminals devoted to corrupting those cards and the infrastructure that underpins them.[32]

Not only does the Utah Strategy shift risk to the private key holder, the strategy concentrates the risk there. The Utah Act gives recipients, like Bob, strong reason to expect that if a document is signed with Alex's private key then Alex is legally responsible for the document. Utah Act § 46-3-406 provides that a document signed with a digital signature is normally presumed to be signed by the person owning the relevant private key.[33] This presumption reduces Bob's incentive to gather or consider any evidence other than the digital signature when he evaluates whether he can prove that Alex is responsible for a document. This pre-

31. Harry A. Shamir, *New Technologies for Records Management*, RECORDS MGMT. QUARTERLY, July 1996, at 9. Developed for transaction processing, most card technologies contain a measure of information. *Id.* Early card technologies typically had embossed numbers and alphanumerics only. *Id.* Today, most cards contain the magnetic stripe containing less than 200 alphanumerics. *Id.* These magnetic stripes can be used on both paper and plastic. There is, however, the risk of losing information if the strip is scratched or the card is bent too much. *Id.*

The standard for the Smart Card is at 3 3/8 inches x 2 1/8 inches and made of thick plastic. *Id.* Smart cards have an embedded Central Processing Unit ("CPU") integrated circuit chip as well as a small amount of memory. *Id.* Most smart cards can be written to as well as read using a special dedicated card read/write unit. *Id.* In Europe, over 250 million smart cards were sold in 1992 alone. *Id.* The sale of these cards was mainly for the telecommunications market. *Id.* The price of these cards ranges from $3 - $10 per card, wholesale. *Id.* The retail price may increase to 20 times that of the wholesale price. *Id.*

32. Markoff, *Two Israelis Outline New Risks to Electronic Data Security*, N.Y. TIMES, Oct. 19, 1996, at 20.

33. UTAH CODE ANN. § 46-3-406 (1996). This assumes his public key is certified by a licensed CA.

sumption allows Bob to forego the trouble of establishing a relationship with Alex or exchanging acknowledgements or other feedback with him to confirm his responsibility. This in turn gives Alex powerful incentive to protect the key. The incentive is much greater than the incentive consumers presently have to protect their automated teller machine cards.[34]

Under the Utah Strategy, control of Alex's private key becomes all-important. In other words, all the eggs are placed in one basket—the private key. This allocation of risk may make sense for some transactions, particularly high-end financial deals initiated by sophisticated people. However, a common person like Alex may not feel comfortable with the system. He may not like dealing with a bureaucratic CA, associating himself with a computerized distinguishing name, or having responsibility for an extremely powerful private key.

Fortunately, the Utah Strategy is not the only way to allocate risk in the signing of electronic documents. Alternative strategies exist in electronic commerce that will spread the eggs among many baskets. PenOp™ is one such strategy.[35]

V. PENOP

A. PenOp Described

PenOp employs a pen biometrics technology.[36] The biometrics technology employs a computer software component that augments the function of other computer applications—such as applications that control electronic documents. PenOp has two primary features: the signature capture service and the signature verification service.

1. *The Signature Capture Service*

The Signature Capture Service ("SCS") captures and permits the storage of certain data associated with the manual inscription of a signature on the screen of a pen-based computer or a digitizer pad on a PC. The SCS works with "Client Application" software that informs the pen computer user what he is doing and prompts him when and how to do it.

In coordination with the Client Application, the SCS receives information, such as a user ID showing who the user (Alex) claims to be. The SCS then prompts Alex to inscribe his signature, using a stylus, to a window on the computer's screen.[37] The SCS supplies the wording of the prompt in the window, known as the "Gravity Prompt," which indicates

34. Under the Electronic Funds Transfer Act, 15 U.S.C. §§ 1693-1693r, consumer liability for a stolen ATM card is often limited to just $50.

35. Tauhert, *supra* note 2, at 16 (stating that PenOp, Inc., is a New York-based company).

36. *See also* Donovan, *supra* note 2, at 11 (describing in detail the PenOp strategy).

37. *See also* Donovan, *supra* note 2, at 11.

the purpose for which the signature is being captured.[38]

As Alex moves the stylus across the screen, an image appears that traces the movement of the stylus. Thus, he sees his autograph. At the same time, the SCS measures certain features of the inscription event. This measurement includes the size, shape, and relative positioning of the curves, loops, lines, dots, and crosses, as well as the relative speed at which each feature is imparted. The results of these measurements are known as "act-of-signing statistics." Alex then has the option, by tapping indicated buttons on the screen, of approving the inscription event, retrying, or aborting the signature.

If Alex taps the approval button, the SCS calculates a "checksum," or a brief string of data, that represents the content of the electronic document referred to by the Gravity Prompt. This checksum is not a complete statement of the original document, and the original document cannot be derived from the checksum. Rather, the checksum bears a mathematical relationship to the document. If the document is changed, then that document is no longer mathematically matched with the checksum.[39]

The SCS then compiles the "itemized data" and computes a second checksum. Included in the itemized data are: the first checksum;[40] the act-of-signing statistics; the date and time of signing (as represented by the computer operating system under which the SCS is operating); the identity of the particular machine on which the signing occurred (based on identity information programmed earlier in the SCS); the claimed identification of the user (Alex); the words that appeared in the Gravity Prompt;[41] and, data reflecting the graphic image of the user's signature.[42]

The SCS then completes two steps of encryption to create a "biometric token." The "first level of encryption" begins when the SCS retrieves a secret key previously programmed into the Client Application and uses that key to encrypt the itemized data. The SCS then calculates from that encrypted data a second checksum. This second checksum establishes a link between the itemized data and the Client Application.

---

38. The "Gravity Prompt" normally refers to an electronic document that is accessible to Alex through the pen computer.

39. PenOp creates "checksums" using the MD5 digest algorithm by RSA Data Security, Inc.

40. Donovan, *supra* note 2, at 11 (defining a "checksum" as computations used to show that two sets of data are identical).

41. Donovan, *supra* note 2, at 11. A biometrics token of a person's signature contains dynamics, speed of writing, and stroke order as the signature is written onto the screen. *Id.*

42. Donovan, *supra* note 2, at 11.

After the SCS creates the second checksum, the SCS then enters the "second level of encryption." The SCS encrypts the itemized data, along with the second checksum, using a different algorithm. This algorithm does not use a secret key from the Client Application. The resulting encrypted string of data—biometric token—is a tamper-resistant representation of the event in which Alex inscribed his autograph.

2. *The Signature Verification Service*

The second primary feature of PenOp is the Signature Verification Service ("SVS"). This service reports the probability that a particular signature is authentic based on prior authorized enrollment sessions. In these enrollment sessions, the SCS captures and the SVS holds act-of-signing statistics in a database for an identified user like Alex.

After the sessions, the SVS may be presented with a particular biometric token and directed to evaluate whether that token is a product of an authentic signature inscription belonging to the user identified in the token. The service decrypts the token and then compares the information therein with the signature statistics stored earlier in its database. Based on this comparison, SVS issues a "signature match score," for example, a score of fifty out of one hundred or a score of seventy-two out of one-hundred.[43]

B. PENOP STRATEGY

The PenOp Strategy might be used to "sign" electronic documents such as contracts, expense reports, or medical records. Under this strategy, Bob seeks merely to have Alex attach a biometric token to an electronic document for the purpose of "signing" that document.[44] Bob does not seek to verify the biometric token at the time of receipt, just as he would not verify Alex's signature at the time he receives a document from Alex.

The PenOp Strategy begins with the configuration of a Client Application within a pen computer, to display to Alex the data within the document in question (text, graphics, and so on, all in digital format). The Client Application then calls the SCS to write a Gravity Prompt, which invites Alex to "sign" the document by inscribing his signature in a window on the computer screen. The SCS also presents Alex with a button to approve the inscription. If he inscribes and approves, SCS captures the necessary data and creates a biometric token. The SCS delivers the token to the Client Application for storage, in a way that identifies the token as being related to the signed document.

43. The SVS applies scientific principles deemed relevant by PenOp's developers.

44. Donovan, *supra* note 2, at 11. With this software, a customer can read and approve documents such as a loan application all through a computer. *Id.*

If, at a later date, a third party, such as a court, wishes to verify that Alex did "sign" the document, that party could obtain the PenOp SVS, introduce Alex it, and use it with the help of an expert to verify (to the degree possible) that the biometric token represents an inscription by Alex.[45] A test could also be made (using the checksums in the biometric token) to establish whether the document to which the token is linked is the exact document used at the time of the token's creation. Another test could be made to confirm that the token was made with the identified Client Application.

The PenOp Strategy is similar to the traditional paper and ink strategy. Direct signature verification occurs only on rare occasions, and the full burden of proving that Alex signed a document rests with Bob. Bob is, therefore, motivated not to rely greatly on the signature; he will want to get evidence and security from other sources, such as advance payment or feedback from Alex and the use of reliable communications networks.

VI. RISK ALLOCATION WITH PENOP

Like the traditional paper and ink strategy, the PenOp Strategy allocates risk across multiple factors. In other words, PenOp spreads the eggs to many baskets. The creation of a biometrics token that falsely appears to come from Alex requires the attacker to defeat security features that are supplied by three different parties: the Client Application developer; the PenOp developer; and Alex.

A. Client Application Security

A biometric token must be made with the aid of an identifiable Client Application. The Client Application supplies the secret key that is used in the "first level of encryption." If an attacker stole many proprietary secrets from PenOp's developer, then he could learn in the abstract how to create false biometrics tokens.[46] However, in order to create a false token that convincingly links a specific transaction to Alex, the attacker would also need to steal the secret key and other information from the developer of the Client Application. The developer decides the de-

45. Alex's cooperation, although helpful, would not necessarily be required at the time his signature is verified. According to PenOp's developer, even if Alex is dead or refuses to cooperate, a limited forensic comparison could still be made between his signature, as captured earlier by PenOp, and one or more specimens of his signature, as written on sundry paper documents such as checks, letters, contracts, or credit card receipts.

46. As the PenOp Strategy is implemented, secret information will need to be divided and spread among segregated parties. Similarly, as the Utah Strategy is implemented, secret information about the private keys and their security (for example, information about the function and control of smart cards and supporting software) will have to be divided and spread among segregated parties.

gree and character of security that the Client Application employs. That developer may employ multiple secrets that are spread among multiple segregated parties, thus distributing more eggs into more baskets. The developer can employ audit trails, secure timestamps, physical access barriers, and even public-key cryptography as part of its security. The greater the security, the greater the forensic value of documents signed through that Client Application.

Before Bob relies very much on a biometric token, however, he will want to know about the reliability of the Client Application that helped create it. Bob would have a similar concern if he were relying on a document signed by Alex with private-key cryptography. The reliability of the link between a public key and a document signed by a private key depends on the reliability of the software that controls the key and exposes it to the document that the key signs.

B. PenOp Security

The PenOp software employs a complex array of secrets that are difficult for an attacker to obtain, understand, and use. (I) The methods PenOp uses to measure and record an act of signing are known only to PenOp's developer, and those methods change over time. (II) For the "second level of encryption" PenOp uses a novel encryption method that achieves five main goals. First, neither the SCS nor the SVS software possesses the key that encrypts a token at the second level of encryption. Second, neither the developer of PenOp nor the developer of the Client Application possesses the key that is used for the second level of encryption. (III) Someone who possesses the object code to the PenOp SVS software could decrypt a biometric token (at the second level of encryption) for purposes of a preliminary analysis of the checksums (linking the token to the original document) and the signature statistics (linking the token to Alex). That person, however, would not have the information to falsify tokens or inconspicuously corrupt them. A deeper analysis of the signature statistics would require expert advice and access to secrets kept by PenOp's developer. (IV) To understand and replicate the second level of encryption, an attacker would need to steal the source code to the SCS and the SVS. The developer of PenOp intends to keep the source code a secret. (V) A very sophisticated and determined attacker might be able break the second level of encryption. Yet the breaking of that level allows the attacker to overcome just one hurdle, or break just one egg, in his fraudulent effort to create a signed document. Furthermore, the second level of encryption is highly resistant to "known plaintext attacks."[47]

47. A "known plaintext attack" is one in which the attacker treats the encryption algorithm as a black box. He takes a piece of known plaintext (for example, a string of zeros), selects a key, and tries to discern how the algorithm works by running the text and key

C. Alex's Secrets

An attacker would need to obtain extensive information about Alex's signing behavior in order to create a convincing biometric token purporting to be from Alex. For the attacker, this would be a considerable burden.

Even if an attacker successfully tricks the developer of a responsible Client Application, the developer of PenOp, and Alex into disclosing the necessary information, the attacker would still have much work ahead of him. To perpetrate a fraud, he would have to fabricate a convincing transaction, one that makes sense under the prevailing facts and circumstances. The transaction would have to be consistent with the types of deals that Alex would have entered into at the time, including the records that Alex and other interested parties would keep. If, for example, Alex were a school teacher of modest means, a corporate debenture that appears to bear his signature would not be convincing.[48] Bob would have a hard time carrying his burden of proof that Alex signed that document.

VII. AUTHENTICATION IS AN ENDLESS JOURNEY

No particular application of paper and ink, the Utah Strategy, or the PenOp Strategy can provide a perfect bond between Alex and the words of a document. The development and use of authentication technology is a dynamic process, not a destination. It is an endless journey in which the good people hurry to stay a step or two ahead of the bad people. The paper and ink tools that provided adequate authentication in the year 1900 do not necessarily provide the same level of authentication in the year 1996. And the computer security tools that provided adequate authentication in 1970 do not provide the same level of authentication in 1996.

Similarly, the tools needed to provide adequate protection for a private cryptography key in 1996 will be different from those needed to provide equivalent protection in 2010. And, yes, the tools needed to protect

---

through the algorithm. He then analyzes the resulting encrypted text. For example, when a string of zeros is run with the key "KEY" through a very simple algorithm, the encrypted result might be "KEYKEYKEY."

However, PenOp's second level of encryption is highly resistant to a known plaintext attack because the second block never encrypts the same block of data twice in the same way. The reason is that the second level uses a random encryption key.

48. As the Information Age progresses, records about transactions become far more extensive and detailed than were possible before. The records become spread among many (and sometimes unexpected) parties, including sundry network service providers. The massive audit trail that will build up around commerce will make for an environment in which fraud is more difficult. More of the facts and circumstances that surround a transaction will be recorded by independent third parties and by other reliable means.

a PenOp application from abuse in 1996 will be different from those needed in 2010. As PenOp becomes more popular, PenOp's developer may need to divide and spread PenOp's secrets across a larger number of people. Under both the Utah Strategy and the PenOp Strategy, systems developers must work endlessly to keep their secrets out of the hands of criminals.

In principle, neither the Utah nor the PenOp Strategy is an inherently superior method for connecting Alex to his electronic words. Each strategy is an approach for staying one step ahead of the bad people. Whether any particular application of these strategies is adequate will depend on all the facts and circumstances of a particular transaction.

The chief difference between the strategies lies in the ways that people will use them. The Utah Strategy emphasizes the investment of many eggs in one basket—the private key; whereas the PenOp Strategy, like the old paper and ink strategy, emphasizes the spreading of eggs across many baskets.

VIII. A CONDUIT TO POPULAR ELECTRONIC COMMERCE

What does this difference in risk spreading mean? It means that the Utah Strategy may not appeal as much to members of the general public. The Utah Strategy stresses the responsibility of Alex to protect his private key, and places a light burden of proof on Bob. In contrast, the PenOp Strategy stresses that Alex and Bob act reasonably under the circumstances. Evidence of signing comes from all the relevant facts, with the full burden of proof being on Bob.

The Utah Strategy relies on the creation of a complex network of certification authorities, and requires Alex to register his identity with a certification authority and to take a new distinguishing name.[49] The PenOp Strategy, on the other hand, requires no advance planning or action on Alex's part. PenOp caters to Alex's work, thus making electronic commerce attractive to consumers and common people.

49. This is a socially and politically sensitive requirement.