SPLAT! Requirements Bugs on the Information Superhighway

Cem Kaner, Brian Lawrence, and Bob Johnson

Software QA Magazine, in press, 1998.

> *Here is a system whose requirements are inadequately defined, ambiguous, and in conflict. Despite these problems with the requirements, the system is already far along in design. Smaller versions have already been implemented and are in service, ready to interact with the full system when it comes on line. We are convinced that, once the system comes under heavy use, it will fail miserably.*
>
> *This is a pure case of a requirements-based failure. Even if the implementation of every component in the system is bug free, we expect the full system to fail so seriously in the field that, in the best case, significant changes will have to be retrofitted to it, at significant expense. In the worst case, the system will be taken permanently out of service after costing innocent people unsuspected but significant losses.*
>
> *We are using this case to point out some ways that you can test the design of systems that you work on, to demonstrate flaws in those systems' requirements.*
>
> *Another interesting feature of this case is that it is a work still in progress. The train is on the tracks and gathering speed, but it will be a while before it reaches the gorge, falls off and crashes and burns. We're probably not going to get a bridge built in time to prevent the crash, nor do we think we can convince the conductor to stop the train or to take a different route. But we do get to watch: The design meetings are open—you can attend them. You can get the main design documents for free on the Net. Over the next few years, some people might learn a lot about requirements by watching this case unfold.*

Electronic commerce is getting a lot of attention in the press and in legal circles today. One of the common complaints about electronic commerce is that there isn't enough of it. Despite the obvious potential of the Net as a place to do business cheaply, people are responding to it very cautiously.

One of the key limiting factors on commerce on the Net is the lack of a well-defined legal infrastructure. For example, suppose that you use e-mail to negotiate a contract with Bob, who is an executive of Company X. Eventually you reach agreement and send confirming e-mails to each other. You will write a test plan. The company will pay you for it. A month later, you e-mail the test plan, and an invoice for your time. The company never pays you. Here are some interesting questions:

- Do you know who the company is? You negotiated with someone who said that his name is Bob and that he represents Company X. Now how do you get paid? Can you prove in court that the mail you received really came from Bob and that he really does represent Company X? Will the judge accept your e-mail messages as evidence? (This is the *identity* problem.)
- Suppose that Bob doesn't pay you because the project was cancelled. Your test plan was fine, but he no longer needs it. He ignores your invoices for a while, then writes you a letter saying "Contract? What contract? I never made this agreement with you." How do you stop him from canceling a contract by falsely denying that it was ever made. (This is the *repudiation* problem.)
- Suppose that Bob agrees to pay you, but only $20 per hour. He changed his copy of the e-mail that he received from you so that it says $20 per hour instead of $50. How do you prove that your copy of the message is the correct one? (This is the *message integrity* problem.)
- What country or state's law governs the contract that you made with Company X? That probably depends on where Bob and Company X are (and they might be in very different places). Suppose that you are in California and Bob lives in Israel and Company X is located in Russia, but Bob

> lied to you during negotiations and said that he and Company X were both located in Florida. What law governs your contract? It could be California's, Florida's, United States federal law, Russia's, Israel's, or an international law created by a treaty that was written by the United Nations Commission on International Trade Law (UNCITRAL). (This is the *choice of law* problem. A related, and equally complex problem, the *choice of forum* problem, asks where you can sue Bob. Even if American law applies, you might have to travel to Russia to sue him.)

Until the laws get straightened out, people should be cautious about making contracts on the Net. There is a tremendous amount of work being done, around the world, to write those new laws. We have some knowledge of this work because Kaner is involved in three of those efforts, serving as a participating observer in *the National Conference of Commissioners on Uniform State Laws'* (NCCUSL's) drafting committee *for Article 2B of the Uniform Commercial Code (Law of Licensing)*, NCCUSL's drafting committee for a *Uniform Electronic Transactions Act*, and as a member of the United States' *Department of State's Advisory Committee on Private International Law: Study Group on Electronic Commerce*, which is developing American policy for its participation at UNCITRAL (United Nations Commission on International Trade Law), which recently developed the *United Nations Model Law on Electronic Commerce.* Johnson has significant experience with software security. Lawrence's expertise is in the development and implementation of system requirements.

The Digital Signature System

The system that we're writing about is a legal system rather than software, but the problems that we raise are directly comparable to requirements-related problems that we see in the development of software systems.

Once the electronic commerce laws have been passed, they will be a stable source of requirements for some very large software systems. Once laws get into place, it is expensive and politically difficult to change them and therefore software systems designers will rely on them as fixed requirements, not likely to change. (In other words, once they pass we are stuck with them and we don't get to ignore them, and we probably can't easily fix them.) At the moment, however, these draft laws are products under development, just like programs under development. Like software products, draft laws are subject to the sometimes conflicting requirements of different stakeholders and to constraints that make it difficult or impossible for the system developers to meet all of the requirements.

The primary requirement for a commercial law is that it should facilitate commerce.

For a general discussion of electronic commerce, we recommend Wright (1997). We're going to look at a very narrow part of that field, the proposed law of digital signatures. For general discussions of digital signatures, we recommend Ford & Baum (1997) and the American Bar Association Science & Technology Section's *Digital Signature Guidelines* (1996), which is a very understandable discussion of the legal issues and is available free on the web. For technical explanations of digital signatures, see Schneier (1996) or the overview in Garfinkel & Spafford (1996). To learn how to encrypt your own messages and/or apply digital signatures to them, we recommend downloading PGP at http://www.pgp.com.

In this paper, we look at one narrow but important issue for digital signatures.

What is a Digital Signature?

The *point* of using a digital signature is that it is a mathematically sound method for solving two critical problems:

- It identifies the person who sent the message (solving the identity problem).
- It makes it easy to tell whether the message has been tampered with since it was sent (solving the message integrity problem).

Lawyers would also very much like to use digital signature technology to solve the problem of false repudiation. They'd like to say that if a message comes with your digital signature, it must have come from you and therefore you can't deny having sent it.

The legal definition of a "digital signature" is "a transformation of a message using an asymmetric cryptosystem such that a person having the initial message and the signer's public key can accurately determine whether: (a) the transformation was created using the private key that corresponds to the signer's public key; and (b) the message has been altered since the transformation was made." (Utah Digital Signature Act, Utah Code Ann. § 46-3-103(10)).

A digital signature is created using public key encryption. Encryption involves encoding (encrypting) messages so that people can't read them without decoding them. An asymmetric cryptosystem involves a pair of encryption keys, one private, the other public. Suppose that Alice possesses a key pair. She can reveal her public key to Bob, and to anyone else who has use for it. When she uses her private key to encrypt a message, Bob can use Alice's public key to decrypt it. When Bob uses Alice's public key to encrypt a message, she can use her private key to decrypt it. The keys are a matched pair. Bob's keys won't decrypt Alice's messages.

Alice can digitally sign a message instead of encrypting it. In this case, she designates the part of the message to be signed. The encryption program uses a hash function to create a fairly short, standard-length string (called the hash result) that is derived from the original message. Alice then encrypts the hash result and sends the message, with the hash result included, to Bob. The message itself is in plain text (unless Alice encrypts it, too). Bob will use Alice's public key to decrypt the hash result. He runs the same hash function on the message. If he gets the same result, then he knows that this message hasn't been tampered with (a different message would, with a probability indistinguishable from 100%, generate a different hash result) and he knows that it was encrypted using Alice's private key.

Alice might obtain her key pair from Carol, who is a certificate authority. Or Alice might register her public key with Carol. In either case, when a person who purports to be Alice sends a message to Bob saying "Here is my public key," Bob can ask Carol whether this key actually belongs to Alice and whether the key is still valid (has not been suspended or revoked by Alice). We aren't going to say much about certificate authorities here. There are lots of fun questions about how Carol finds out that Alice is Alice, and what rights Alice and Bob should have against Carol for Carol's mistakes. One of us (Johnson) feels strongly that the entire notion of a certificate authority is a bad idea, but these issues will have to be addressed in their own papers, not here. For more discussion of the role of the certificate authority, we recommend Froomkin (1996), Ford & Baum (1997) and Wright (1997) (who points out that business people can create perfectly binding contracts on the Net without needing a certificate authority.)

Private keys can be kept in various places, such as on the hard disk or a separate disk or on a smart card. Storage of the key on Alice's local computer is very common. The key itself is probably encrypted or otherwise hidden and not made available to Alice (and therefore not to anyone else who uses Alice's computer) unless she types a pass-phrase. To see an example of the creation and storage of an encryption key pair, download and install the latest version of PGP (Pretty Good Privacy) from http://www.pgp.com.

Some Business Transactions Using Digital Signatures

==================

Insert Figure 1 here

==================

Figure 1 illustrates a simple sales transaction using a credit card. This is a five step transaction:

1. Alice (the customer) gets a credit card from the credit card company. She probably won't get a new one for every transaction, but she has to get one to start.

2. Alice sends an e-mail to Bob (the seller), asking him to send her (to her shipping address) a copy of *Testing Computer Software* for $32. She includes her credit card number.

3. Bob checks with the credit card company, which confirms that card is still good by giving Bob an authorization number. The company charges Bob as much as 5% of the transaction for this.

4. Bob sends the book to Alice, along with a receipt.

5. The credit card company will bill Alice and she'll pay it.

==================

Insert Figure 2 here

==================

Figure 2 illustrates a simple sales transaction using a digital signature. This is a five step transaction:

1. Alice (the customer) gets an encryption key pair from Carol (the certificate authority). She probably won't get a new key for every transaction. Carol doesn't know Alice's private key, but she has Alice's public key on file.

2. Alice sends an e-mail to Bob (the seller), asking him to send her a copy of *Testing Computer Software* for $32. She digitally signs the message and sends a certificate (an associated message) that appears to be from Carol that says that Carol says that Alice is Alice.

3. Bob checks with Carol. Carol supplies Alice's public key to Bob (or confirms that the key sent to Bob is Alice's) and says that Alice has not yet suspended or revoked this key. (Telling Carol to revoke your key is like calling VISA to tell them you lost your credit card.) Carol doesn't pay Bob and she doesn't guarantee the transaction. However, Bob probably pays Carol a lot less for this than he pays the credit card company.

4. Bob sends the book to Alice, along with an invoice.

5. Alice sends Bob the money.

==================

Insert Figure 3 here

==================

Figure 3 illustrates a fraudulent sales transaction using a digital signature. This is a six step transaction, with lots of follow-up steps.

1. Alice (the customer) gets an encryption key pair from Carol (the certificate authority).

2. Fred gains access to Alice's computer and makes a copy of her key. Alice doesn't realize this, and so she doesn't revoke or suspend her key.

3. Fred impersonates Alice. Fred sends an e-mail to Bob asking him to send Alice a copy of *Black-Box Testing*, but gives a mailing address controlled by Fred. Fred digitally signs the message with Alice's key and sends a certificate (an associated message) that appears to be from Carol that says that Carol says that this key belongs to Alice.

4. Bob checks with Carol. Carol supplies Alice's public key to Bob (or confirms that the key sent to Bob is Alice's) and says that Alice has not yet suspended or revoked this key.

5. Bob sends the book to "Alice" (Fred), along with an invoice. Fred picks up the book and flees with his loot to the Bahamas, never to be found again. (Fred placed 10,000 other orders for merchandise using Alice's key. Even though no single order was very big, Fred got a lot of stuff.)

6. Bob (along with Fred’s 10,000 other victimized companies) sends Alice a letter demanding his money. She denies placing the order(s). Alice, the 10,001 Bobs, and all their lawyers get to know each other well and live unhappily ever after.

Interest Analysis

Testers are rarely involved in the initial requirements analysis of a system. However, if you are a tester, one of your key objectives is to prove that the system doesn’t meet the stakeholders’ requirements. (This is the *definition* of system testing. See Myers, 1979.) In doing this, testers often develop an impression or an implicit model of the requirements of these stakeholders. It’s very helpful to make that model explicit. In writing this, we will speak to you as though you were creating this explicit model, that is, as though you were creating a system requirements analysis.

Bob, Carol, Fred and Alice all have their own interests in the digital signature-based system. When analyzing the requirements of a system, it is extremely valuable to analyze the interests and requirements of each stakeholder (or class of stakeholders) in turn. (Gause & Lawrence, *in preparation*).

An interest analysis should yield a list for each stakeholder in the system. The list should include the things that the stakeholder wants or needs in the system. It should also point out things that the stakeholder would *not* want in the system, especially those favored by some other stakeholder. The lists should expose and highlight conflicts, not hide them.

- Alice is interested in a no-hassle method of shopping. She is not a security expert and doesn’t care to become one. She will probably agree to use a digital signature if it doesn’t cost anything and doesn’t expose her to significant risk. She is more likely to use one if Bob gives her a discount for using it. However, Alice benefits from shopping with her credit card instead of a digital signature. First, she understands it. Second, she can pay her bills over time. Third, the credit card company will help her get a refund from Bob if Bob delivers an inferior product, double bills, or treats her unfairly in some other way. And finally, the credit card company protects Alice if she loses her card or if someone else gets her number and commits fraud with it. Alice will only have to pay $50, even if she was very sloppy in her handling of the card.

  NOTE: We don’t know whether consumers will use digital signatures in favor of credit cards. It will depend on the incentives offered by the merchant. We analyze this system carefully, however, because it is being deliberately created within a highly visible new body of commercial law.

  (Another note: Alice might like digital signatures because she has no credit cards to use instead. A few million people have declared bankruptcy recently enough that they probably don’t have credit cards.)

- Bob wants a hassle-free way of selling stuff. The credit card system has served him well. It pays him right away. But this system is expensive and it sometimes forces Bob to give refunds that he thinks are totally unjustified. Bob is willing to sell merchandise on credit, but he needs to be sure of who he is selling to, he wants some assurance that they are credit worthy, and he wants a guarantee that if he ships the goods, someone will pay for them. Bob would like an assurance of credit-worthiness from Carol, but Carol’s fees are very low because she’s not in that business. Bob will probably be willing to extend a little bit of credit (perhaps $50 or $100) to Alice (who is affluent enough to have a computer) as long as he can get a positive ID from Carol and a legal guarantee that Alice will have to pay him. (Even if he can’t collect from Alice, this will let him file a credit report and hurt her credit rating. That’s a powerful threat to use against many customers.) Bob knows that he won’t always be repaid in this system, but losses from fraud must be so low that it is cheaper for him to do electronic commerce using digital signatures (after fraud losses are factored into the cost) than to do it using credit cards. Bob probably also wants his decision to ship or not ship the goods to be a no-brainer. If possible, he’d like to automate the entire process—his computer receives the message from Alice, forwards it to Carol, gets a positive ID from Carol, and then arranges for shipment of the goods. For his purposes, then, the message from Carol should free him from having to think about whether to extend credit to Alice or not. He should have a simple, programmable way of deciding yes or no.

- Carol wants to make money in the identification business. She is free of federal regulations that force credit card companies to deal with customer complaints and that limit customer losses from lost cards. She doesn't want to provide credit or insurance and she doesn't want to mediate buyer/seller disputes. She wants to make her money by charging a small amount per transaction, for many very simple transactions. Carol could make a fortune if she can turn this into a clear cut, automated process. However, she charges so little per transaction that she could lose a fortune if she accepts liability risks for things that require judgment on her part or that involve misbehavior by a third party (such as Bob or Fred). Carol is perfectly willing to accept responsibility for checking Alice's ID (within limits that are revealed to Bob) when Carol initially sells Alice her encryption keys, and for accurately telling Bob that Alice has suspended or revoked her key. If Carol fails in either duty, she will pay. Bob and Alice will be free of liability for any results of these mistakes.
- Lawyers who work with banks have expressed some disgust with consumers' handling of credit cards. They express anger that there is so much fraud in the industry and they say that they feel that much of this fraud would have been avoided if the consumers followed modest security precautions. They also feel that much of the fraud is committed by the holder of the credit card who falsely reports that it was lost or stolen or that a series of purchases were made by an unknown third party. Kaner has heard, under circumstances that don't allow attribution, at least one very senior banking lawyer and one more junior one say that consumers need to be taught a lesson about security and that if a few people lose money because third parties fraudulently use their digital signatures, it will serve those consumers right. Many other lawyers whose stated positions are less extreme would still feel strongly that the system must encourage Alice to take reasonable care to keep her encryption key private, and that the cost to Alice, if her key is compromised through her negligence, should be enough to hurt. Otherwise, they feel, the Freds of the world will flourish.
- Finally, there is Fred. Fred is a *disfavored* stakeholder. We want to design the system in a way that either doesn't take into account Fred's interests or is specifically designed to exclude Fred. Fred wants easy access to Alice's encryption keys. He wants Bob to trust him when he says he's Alice. He would like to be able to buy small, valuable things or to transfer Alice's money with this key. He wants the system to have built-in delays so that he can have some safe time between the time he starts using Alice's key and the time that she realizes that someone is committing fraud, and he doesn't want anybody to fix the system once he's found out how to steal from it.

With Fred active in the system, Alice's, Bob's and Carol's interests are in conflict. Alice wants to do safe shopping, Bob wants to be paid, and Carol wants Bob and Alice to do lots of business (they pay Carol a fee whenever they do business) but doesn't want to pay anything when Fred impersonates Alice by using her encryption key.

Fred poses a difficult problem for this system, because there is no fair allocation of risk here. Alice, Bob, and Carol are all victims of Fred. There is no argument in principle that makes Alice or Bob or Carol the fairer target to hit.

Finally, look at how comparable systems deal with the different stakeholders' interests. These set expectations. For example, in a paper based system, if Fred forged Alice's signature, then Bob would lose the money, not Alice, if Bob accepted Fred's forgery. However, in a credit card system, if Carol authorizes a transaction (giving Bob an authorization number for a $100 credit card purchase by Alice), then Carol loses the money if the person who had Alice's card was Fred. However, if Alice uses a signature stamp (a rubber stamp with an impression of Alice's signature on it), then whoever possesses the stamp can make binding-on-Alice cheques in her name. Woe unto the person who scans her handwritten signature into clip art and then starts signing electronic checks and contracts with this clip art signature. This might be taken as equivalent to a signature stamp. In sum, there are good reasons for Bob, Carol, and Alice to each think that the other two should pay for Fred’s mischief. Each of them is sometimes subject to liability for Fred’s frauds.

The Resolution of the Conflict

One typical problem in a study of system requirements is that some stakeholders are better represented in the process than others. For example, whenever someone asks you to test code that hasn’t been designed for testability, you are probably looking at a result of a system in which the testers’ interests were not well represented. It’s not just the testers who suffer from that breakdown of the process. The company will now have to pay more for testing, get less efficient work from the testers, and deal with more customer complaints about undiscovered-during-testing but serious bugs. The cost to the company and the customers might be enormous.

In the digital signature legislation design meetings, Alice has not been as heavily represented as Carol (e.g VeriSign) and the bankers. Encryption technology and its application to purchasing is a technically complex problem that has been somewhat intimidating to the consumer protection community.

We show the results of these meetings in the Appendix, with sample statutory language from draft Article 2B of the Uniform Commercial Code (NCCUSL, 1997a), which will probably be introduced to state legislatures in the USA this fall. We quote Article 2B because its wording is easier for a non-lawyer to understand than some other documents, but the same results fall out of Utah’s (already enacted into law) Digital Signature Act, the American Bar Association’s Digital Signature Guidelines, the United Nations Model Law on Electronic Commerce and several others. See Smedinghoff (1997) and Gidari & Morgan (1997) for pointers to many other statutes and draft statutes.

Under Article 2B (section 116(a)(3)), if Fred impersonates Alice and places orders using her private key, Alice will pay for every order that Fred placed unless she can show that her key was compromised (read by Fred) under circumstances that did not constitute a failure to exercise reasonable care. Under Article 2B-116(b)(1) the burden of establishing her non-negligence lies with Alice, not with Bob or Carol. The underlying assumption is that a fraudulent sender would have gained access to Alice's key through her negligence. In effect, she is assumed guilty of negligence unless she can prove herself innocent. But if Alice doesn't know who Fred is or how or when he gained access to her key (it’s often difficult for the police to find out who Fred is, and Alice is probably not as effective at criminal investigation as the police), she will probably find it impossible to prove her non-negligence, even if she was in fact not negligent at all.

If Alice cannot prove her non-negligence, she is responsible for every transaction done in her name by Fred, up to the point when she revokes her key. Because Fred can send hundreds or thousands of messages per day with his computer, by the time Alice discovers that Fred has been using her key, she may have lost a fortune. Her liability is unlimited.

This satisfies Bob’s requirements because he can force Alice to pay whether she placed the order or not. And it satisfies Carol’s requirements because she’s not liable for anything. But they get their requirements met at Alice’s expense. Why would any rational consumer (Alice) take on the risk of having to pay for all of the losses that would result if an organized crime syndicate snatches a copy of her key (and many others)? Why should she risk bankruptcy in order to shield retailers and certificate authorities (and their insurance companies) from white collar crime that she is no less a victim of than they?

This Resolution Won’t Work

Remember that the core requirement of a commercial law is to facilitate commerce. If we design a digital signature system that no one will use, we have failed.

We contend that once the system is in widespread use, some sophisticated criminals will figure out how to gain access to consumers’ encryption keys, they will use those keys to commit fraud, and the result will be dead puppy stories in the mass media. (A dead puppy story features a cute, lovable, defenseless victim who has been injured, killed, cheated or terrorized by the forces of darkness.) Poor Alice will appear on

20/20, on 60 Minutes and on various other news shows, crying about losing her house, her job, everything she owned, all because some unknown Fred somehow gained access to her encryption keys. Alice is a senior nurse, or an English professor, or some other professional that no one would expect to be a security expert. She says between bouts of sobbing that she doesn't fully understand this encryption stuff but she knows that she handled her encryption key carefully. But she can't *prove* that she was never, ever, just a little bit sloppy about who had access to her computer for every hour of the past three years that she's possessed her encryption key, and therefore she is going to lose everything.

It won't take many news shows featuring a few different Alices before most or all consumers in the US realize that no one in her right mind would accept this liability risk.

We aren't the only people to talk about this issue. For example, UNCITRAL (1996, paragraph 65a) says it in legalese. Article 13 ("Attribution of data messages) of the United Nations Model Law (discussed in this quote) is similar to Article 2B-116.

> *"The Working Group may wish to discuss whether the issue of attribution of digitally signed messages might be dealt with simply by reference to article 13 of the Model Law. Draft article E, which is modelled on Article 13 of the Model Law, is intended to provide an illustration of the principles contained in article 13 in the context of digital signatures. It is based on the need to provide certainty as to the legal effect of digital signatures, which are currently regarded as a highly secure authentication procedure. The draft provision places a heavy burden on the originator of a message bearing that originator's digital signature. It may be recalled that, under article 2(c) of the Model Law, "originator" means any person by whom, or on whose behalf, the data message purports to have been sent. The draft provision illustrates the need for any user of a digital signature to protect its private key which, if applied to encrypt a message, will create an irrebuttable presumption that the message was that of the purported originator.*

People are aware of this "heavy burden" but despite this recognition, laws are being passed (such as Utah's) and are being recommended by extremely influential legislative drafting groups (American Bar Association Section of Science & Technology, 1996, and the work in progress at NCCUSL, 1997a, 1997b).

The Six Denials

Why adopt a system that bears a significant risk of creating dead puppy stories and thereby failing?

This is not just a question for legislative analysts. We see this in the design of software systems all the time. Look at the general situation.

- The system has many stakeholders.
- The stakeholders have conflicting interests.
- One group of stakeholders has convinced the requirements analyst to ignore or discount the interests of another stakeholder. (That ignored stakeholder was probably absent or under-represented at the meetings in which the system requirements were defined. Very often, the main end user of the system is unrepresented or underrepresented at these meetings.)
- At some later point, you file a bug report that says that some aspect of this system will cause a bad result (harm the ignored stakeholder). (You may not even realize that this stakeholder's needs were discussed earlier in the project.)

Your bug report will get a response, but it will often be that you are filing a non-bug. The report will come back saying that the system "works as designed." If you push back, you are likely to be met with one or more of the six denials:

1. This isn't a problem because we talked about it when designing the system and we decided it wasn't a problem. You should have raised the issue earlier.
2. This isn't a problem because it won't happen very often and it won't happen to anyone important.

3. This isn’t a problem because it’s the user’s fault.
4. This isn’t a problem because new technology will solve it.
5. This isn’t a problem because it would cost too much to fix it if it was a problem.
6. This isn’t a problem because people won’t really see it as a problem. We have to ship this product sometime. We should put it out there and let the market decide.

Sound familiar?

We’ll work through the six denials as they apply to digital signatures, but you should keep them in mind as you work with software. On a controversial system, you’ll probably have to think through a convincing response to each class of denial.

Denial 1: We talked about it already

The system has to stabilize sometime, after all. It’s very frustrating when new people come in and challenge decisions that have been discussed time and again already.

Unfortunately, many stakeholders aren’t represented in early discussions or they are significantly underrepresented or they just don’t visualize future problems well. In the digital signature situation, relatively few consumer activists have been involved (few are involved today) and even fewer (perhaps none—Kaner certainly doesn’t claim to be one) are experts in the technology.

The problem with this argument is that it doesn’t address the underlying issue. If the problem is serious, the consequences will be serious. In this case, innocent Alices will lose lots of money, and other consumers will walk away from the technology. The system will thereby fail.

Serious requirements errors cause serious losses whether you face them during development or not.

Denial 2: It won’t affect anyone

On one of the digital signature discussion lists, some people are already saying that this risk is way overblown. Utah’s law has been in effect for a couple of years and no one has reported this type of problem. Therefore, the system will not have this problem, or not very often.

This is an interesting argument because Utah has yet to register a single Certificate Authority. The system isn’t yet fully up and running. Not many people (if any people) are using digital signatures to guarantee purchases in Utah yet. Therefore, even though the law has been in place for 2 years, there is very little test data that is relevant to this issue.

We shouldn’t expect a significant fraud problem until large numbers of merchants are willing to accept digital signatures on purchase orders instead of paid-in-advance-by-credit-card orders. If only two merchants accept the signatures, Alice can’t lose much money until Fred can impersonate her in many different places. Until then, field experience hasn’t tested for this problem.

Another variation of this argument is a claim (which we haven’t heard made for digital signatures) that customer reactions were studied by using focus groups or other predictive techniques. Or just a claim (which we have heard) that if people cared about it, they’d speak up more loudly.

It’s important to ask yourself why user representatives to a system don’t protest an obvious design failure. We think that this often occurs because user representatives don’t understand the potential failure or haven’t digested it to the point that they understand how serious it is. Requirements document review is often not done in a way that encourages readers to use their imagination to visualize practical results.

In the digital signature case, we think that people are so used to having an extensive safety net in credit-related transactions that they find it hard to believe that the consequences in the United States would be that drastic. (Of course, that expectation will make the dead puppy stories all the more stunning five years from now.)

Denial 3: It’s the user’s fault

This is the main denial in the digital signature system. The claim is that it is virtually impossible for Fred to impersonate Alice because it is virtually impossible for Fred to crack Alice’s key. Therefore, if Fred uses Alice’s key, he must have gotten it from Alice, and Alice should learn how to take care of her valuable possessions.

This is a load of baloney.

If you are interested in seeing encryption and digital signatures in detail, we recommend that you visit the PGP web site and download and install PGP (http://www.pgp.com). (We have no relationship with PGP.) This will give you a concrete example of an encryption system that lives on your computer.

In PGP's case, the private and public key pair will both be stored on your hard disk. PGP will ask for a "pass phrase" that you will retype in the future to gain access to your private key.

To obtain your private key, a criminal must gain access to your computer and then guess your pass phrase. This can be very time consuming. A wise criminal will probably copy your hard drive, or sections of it, and then crack your pass phrase at his leisure.

Most people are not good at creating unguessable pass phrases and passwords. Hackers crack into systems all the time by guessing passwords. We don’t see any reason to expect Alice to get better at this than she is, on average, today. Alice will probably also store her pass phrase, along with several other passwords, in a file that she keeps on her computer. If Fred can find this file, he can get Alice’s key. You might say that these are examples of poor security practices by Alice, but we have to be cautious about passing a new law that requires Alice to do new things or do things more carefully than she has ever done them before. As we’ve learned in software time and time again, user errors don’t go away just because we say “Don’t do that.” Usually, we have to design a system that is robust against the error, rather than trying to retrain all the users to work around the system.

We can predict a strategy for criminals. Fred will want to gain access to large numbers of computers, copying a large number of hard drives. If the user of that computer used a good pass phrase, or doesn’t have a digital signature, then Fred will move on to his next copy of the next hard disk. If he can crack 10% of the pass phrases (40% might be realistically achievable) then he can develop a good inventory of cracked keys. With this, he can use a lot of keys to place a lot of orders for lots and lots of merchandise. He takes delivery for a few days, sells the stolen merchandise to someone who will resell it, and then hides out for a while.

Here are a few ways that Fred can gain access to Alice’s hard disk. Kaner (1998) describes a few others and explores most of these in more detail.

First, if Alice electronically registers her software, she might send more data than she expects. For example, some electronic registration systems upload information about your computer’s file structure and the applications loaded on your system. Alice isn’t told about this, the software just does it. This is not an urban legend. Joe Jacoboni, of Software Support, Inc., was honored ("The Year's Best Support Ideas") at the 1996 OpCon West (Soft-Letter operations conference) for selling system software that does this. Fred can write electronic registration software too. His would send more data from Alice’s hard disk, to him, including her encryption key and files that might contain the pass phrase to it.

Second, Fred might set up a web site to make legitimate sales. His trick will be to have Alice fill out a very large (e.g. several high resolution pieces of clip art) form, encrypt the message with her private key, and send him back the whole thing. If he can get a very long message or two from Alice that he knows the contents of, he’ll find it much easier to crack her key.

Third, if Alice computer breaks down, she will take it to the shop. If Fred works as a technician at the repair shop, he gets to read her hard disk.

In each of these three cases, Fred will get Alice’s encryption key without her realizing it. She won’t know who Fred was or how he got access to her key. How does she prove that her handling of her key was non-negligent?

And if you react to these three examples by saying that they illustrate negligent use of the key, we say that these are common-looking situations that would not trigger the suspicions of an average computer user. Remember that we are not designing this system just for security experts. We are designing a system that will be marketed to mass-market consumers who don't know much at all about security. We see magazine articles pushing digital signature technology as "digital drivers licenses" that you use to ride the information superhighway. They sure don't warn people about these risks. If a system is designed for use by ordinary consumers, it has to be safe in the hands of ordinary consumers. Otherwise it is seriously defective.

Denial 4: New technology will solve it

In the digital signature case, the magic technology is the smart card. We have been reassured that your private key will live on the smart card, and it will be uncrackable there because a smart card is truly secure. Smart card readers will be built into every computer soon, we have been told. And smart card readers will be available for purchase as upgrades to old computers. We are told that consumers won't mind spending $125 plus installation costs to put one of these readers on their machine because it makes their use of encryption completely safe.

Anyone who keeps their key on their hard disk instead of using a smart card will be said to be handling their key negligently. If Alice loses all her money because her key was hacked off of her hard disk, it will serve her right. (The user is at fault.)

There are a few problems here. First, it is usually a mistake for a law to specify or assume technology that is not yet in common use. We have no idea whether smart cards will catch on with American customers or not. We have no idea whether people will happily upgrade their systems. And if the law is going to create a trap here, it should be fixed to eliminate the trap. If the law declares that Alice is negligent if she creates a digital signature from a key stored on her hard disk, rather than from a key on her smart card, then the law should ban consumer (digital signature) use of encryption keys except from smart cards.

Apart from the questions of consumer acceptance of this new technology, we want to remind you of all of the other silver bullets that have been promoted as absolutely secure. The smart card is a computer that sits on a small chip. We've already heard stories of different ways to hack the chip (gain access to its information). Like so many other silver bullets before, we predict that the smart card will turn out to be just another speed bump on the fraud superhighway.

Many discussions of weak requirements get sidetracked by promises of a rescue by silver bullet technology. The participants get lulled into believing that disaster isn't possible after all, as long as the customer invests in the silver bullet. But experience doesn't bear out this new sense of safety. As Weinberg (1992, p. 241) puts it, "The thought that disaster is impossible often leads to an unthinkable disaster."

Denial 5: It can't be a problem because the solution is too hard

The claim is that this can't be a problem because there is no economical solution. There is some plausibility in such claims because no system can meet all of the needs of a group of stakeholders whose needs are in conflict. But "no other way" is often trotted out when the speaker really means "it would be inconvenient" or "I haven't thought about this before but the system works well for me as it stands, so let's not change it."

The fact is that this system could easily be improved to provide more security for Alice. We provide a few examples here. Kaner (1998) provides a few more and a bit more detail for these. In both papers, the examples are illustrations and not, in themselves, an alternative design.

Carol could let Alice specify a few things with her key. Whenever Carol sent a message about Alice's key to Bob, Carol's message would also include these extra facts, such as:

- Alice says that for any transaction greater than $50, you must call her to confirm it at 408-555-1212.

- Alice says that this key is used only for signing announcements and it is not valid for purchases of any amount.
- Alice says that this key is used only in conjunction with her VISA card and is not for purchases made without a credit card.
- Alice says that this key is used only for purchases that result in delivery of goods to 123 Main Street, Frogsquat, Minnesota. She will not pay for merchandise shipped to any other address.

This makes life a little harder for Bob because he might have to have a human read these messages rather than just making a simple, automated sell/don't-sell decision based on Carol's response. But maybe Bob, Carol and representatives of Alice can standardize the most common of these messages so that a computer can interpret most of them. The point to note here is that this is less convenient for Bob and Carol, but it dramatically reduces the probability of a catastrophe for Alice.

Here are some other ways the system could be changed that might be achievable.

- When Bob gets an order from Alice, he includes the amount of the order in his message to Carol. If she has "approved" more than $1000 (an Alice-specified number) worth of orders over the past thirty days, then Carol tells Bob that the encryption key is suspended.
- When Carol receives a query about Alice, she adds one to her Alice-counter. If she has received more than 10 (an Alice-specified number) orders today, Carol suspends the purchasing power of the encryption key for three days and sends an e-mail to Alice.

This leaves Bob's decision simple (the message from Carol simply says that the key is either good or it is not) but requires Carol to do more work.

By the way, we aren't saying that Alice should be required to add these security restrictions to her key. We're just saying she should have a choice. Imagine what will happen if Carol asks Alice, "Do you want to restrict this key so that purchases made with your digital signature go only to your billing address?" and Alice says "No." If Alice does suffer fraud later, the public will be less inclined to sympathize with Alice because she could have protected herself very simply, and she chose not to. This situation will not kill public confidence in the fairness of the system.

Sometimes people deny that there are alternatives because they benefit too much from the current design and they know that no fairer solution would be as good for them. Sometimes designers fall in love with their designs and can't step far enough back to see that another approach would solve problems that their system leaves open. And sometimes some people just don't see alternatives that other people can imagine. Rather than accepting the argument that there is no economic solution to a problem, we suggest that you brainstorm for a while, sometimes with other people, sometimes on your own, looking for plausible alternatives. The alternatives might not be perfect, but you might be able to come up with a demonstration that there is some cost-effective other way, and challenge the designers to work with it in order to develop a better cost-effective other way.

Denial 6: Let the market decide

The claim here is that this isn't so bad, that other businesses (such as insurance companies and fuller-service certificate authorities) will spring up and take care of customer needs. The law should stay out of the way of this. Let the market decide.

Have you ever noticed that your PIN (personal ID number) is not printed on your ATM (automated teller) card? In the United States, your bank is liable for losses if your card is stolen and used in their machine. Therefore, your bank requires you to supply security information, or it won't give out money.

No federal law requires the telephone company to absorb losses caused by fraudulent use of your telephone calling card. These cards come with the PIN printed right on them. One of us has asked for cards that don't have the PIN on them, and has been told that they're not available. ***When the risk of loss is legally allocated to the customer, companies that intensely compete with each other still aren't helping the customer limit his losses by providing an obvious security feature.***

What guarantee is there that the market will be any kinder to digital signature users?

Anderson (1994) discusses the difference in security systems of ATMs between the United States and the United Kingdom. When there is a fraudulent withdrawal in the United States, the bank has to prove a customer withdrew the money, or absorb the loss. In the United Kingdom, the customer has to prove that he didn't withdraw the money, or he absorbs the loss. According to Anderson, security is much better in the US. The market isn't being very kind to ATM cardholders in the UK. Why should we assume its kindness to digital signature users in the USA?

Closing Remarks

In some ways, public key encryption is remarkably secure. In other ways, we can identify clear risks. We have and use encryption key pairs. We have not registered them with any certificate authorities because we think that only an insane person, an ignorant person, or a fool would choose to accept the current risk allocation of owning an encryption key that Carol could certify to Bob as valid. As an attorney, Kaner advises his clients not to obtain key pairs from Certificate Authorities nor to register their key pairs with CA's.

We don't think that we're being overcautious. We think that the law is being designed recklessly with respect to the risks being imposed on a large number of unsuspecting Alices.

At the point that consumers do suffer significant fraud, we expect to see wide publicity and widespread abandonment of the system. And not just by consumers. Businesses have their security systems cracked disappointingly often. How many horror stories will it take before businesses walk away from this technology too. If you were a nontechnological business (a paper manufacturer or a pizza maker, etc.), would you ban your employees from using digital signatures to order merchandise in your company's name? Would you ban or severely restrict the use of encryption in your company in order to avoid the possibility of Fred getting your encryption keys and then ordering tons of merchandise in your name?

As digital signature laws are passed, software systems are being built to implement systems that are, in some ways, specified by the laws. Those software systems will probably treat the laws as stable requirements. Changing the assumptions in those systems (such as assumptions about what information Carol should be able to pass back to Bob on behalf of Alice) will probably require significantly expensive revisions.

Can you spell "Y-E-A-R-2-0-0-0?" This is what happens when we base complex systems on assumptions that we know are false. No matter how carefully crafted the implementation, the result will be failure.

We see this system as a case study of requirements difficulties. There's plenty more to study in this case (probably a Ph.D. dissertation or two), but there are some lessons that we'd like to suggest here:

1. Stakeholders in a system often have conflicting interests. There is great value in studying the interests of each stakeholder, rather than trying to come up with a "good-for-everyone" description of stakeholder requirements.
2. These conflicts are often swept under the carpet during system design. As a reviewer of the system's design or of implemented software, you will often see problems that disadvantage an important stakeholder (especially the often underrepresented-during-design end users, but don't forget that testing groups are often-ignored stakeholders, too).
3. Reports of problems with the design, that are really reports that one stakeholder's interests are being ignored or undervalued, will often be quickly dismissed by the development manger or the design team.
4. On seeing the dismissal, ask yourself whether you are being brushed off with one of the six denials. If so, can you build a strong case that the denial is inappropriate? If you can, you will probably have to escalate your argument to the full design team or to a supervising executive. This won't be an easy discussion and you will have to prepare your argument well, and with a lot of data. However, the long term savings for your company might be enormous.

Appendix: Risk Allocation Law

Here is an illustration of risk allocation from the language of Article 2B of the draft Uniform Commercial Code (NCCUSL, 1997a). As we mentioned in the main text, several other statutes and model statutes follow the same structure. We chose this one because we think it is more clearly written and better laid out than some others.

Article 2B-102(a)(3) defines authentication as follows:

> "Authenticate means to sign, or to execute or adopt a symbol or sound, or encrypt a record in whole or in part, with intent to
>
> > (i) identify the party;
> >
> > (ii) adopt or accept a record or term; or
>
> establish the authenticity of a record or term that contains the authentication or to which a record containing the authentication refers.

Article 2B-116 lays out the liability rule:

> SECTION 2B-116. ATTRIBUTION TO A PARTY OF ELECTRONIC MESSAGE, RECORD, OR PERFORMANCE.
>
> (a) As between the parties, an electronic authentication, message, record, or performance is attributable to a party if:
>
> > (1) it was in fact the action of that party, a person authorized by the party, or the party's electronic agent;
> >
> > (2) the other party, in good faith and in compliance with an attribution procedure for identifying a party concluded that it was the action of the other party, a person authorized that party, or the party's electronic agent; or
> >
> > (3) the authentication, message, record, or performance:
> >
> > > (A) resulted from acts of a person that obtained access numbers, codes, computer programs, or the like from a source under the control of the alleged actor creating the appearance that it came from that party;
> > >
> > > (B) the access occurred under circumstances constituting a failure to exercise reasonable care by the alleged actor; and
> > >
> > > (C) the other party reasonably relied to its detriment on the apparent source of the message or performance.
>
> (b) In a case governed by subsection (a)(3), the following rules apply:
>
> > (1) The relying party has the burden of proving reasonable reliance, and the alleged actor has the burden of proving reasonable care.
> >
> > (2) Reliance on an electronic record or performance that does not comply with an agreed attribution procedure is not reasonable unless authorized by an individual representing the other party.
>
> (c) Attribution under subsection (a)(2) creates a presumption that the authentication, message, record or performance was that of the party to which it is attributed. However, except as otherwise provided in this section, if a loss occurs because a party relied on an electronic message or record as being attributable to the other party, as between the two parties, the party who relied bears the loss.

References

American Bar Association Section of Science and Technology, Information Security Committee of the Electronic Commerce Division (1996), *Digital Signature Guidelines,* available at http://www.abanet.org/scitech/ec/isc/home.html.

Anderson, R. (1994) “Why Cryptosystems Fail,” *Communications of the Association for Computing Machinery*, volume 37, November, p. 32.

Ford, W. & Baum, M.S. (1997), *Secure Electronic Commerce* Prentice-Hall.

Froomkin, A.M. (1996) “The Essential Role of Trusted Third Parties in Electronic Commerce,” *Oregon Law Review,* volume 75, p. 49.

Garfinkel, S. & Spafford, G. (1996) *Practical UNIX & Internet Security*, 2nd ed., O’Reilly.

Gause, D. & Lawrence, B. (*in preparation*) *Managing Requirements*, to be published by Dorset House.

Gidari, A. & Morgan, J. (1997) *Survey of Electronic and Digital Signature Legislative Initiatives in the United States* at http://www.ilpf.org/digsig/digrep.htm.

Hughes, L.J. (1995) *Actually Useful Internet Security Techniques*, New Riders Publishing.

Kaner, C. (1998). “Speed Bump on the Fraud Superhighway: The Insecurity of the Digital Signature.” *UCC Bulletin*, in press, February 1998.

National Conference of Commissioners on Uniform State Laws (NCCUSL, 1997a) *Article 2B of the Uniform Commercial Code (Law of Licensing)* (November 1, 1997 draft) available at http://www.law.upenn.edu/bll/ulc/ulc.htm.

National Conference of Commissioners on Uniform State Laws (NCCUSL, 1997b) *Uniform Electronic Transactions Act* (August 15, 1997 draft)) available at http://www.law.upenn.edu/bll/ulc/ulc.htm.

Schneier, B. (1996) *Applied Cryptography : Protocols, Algorithms, and Source Code in C*, 2nd Ed., Wiley.

Smedinghoff, T.J. (1997), *Summary of Electronic Commerce and Digital Signature Legislation* available at http://www.mbc.com/ds_sum.html

United Nations Commission on International Trade Law, Working Group on Electronic Commerce (UNCITRAL, 1996) *Planning of Future Work on Electronic Commerce: Digital Signatures, Certification Authorities and Related Legal Issues*, 31st Sess., U.N. Doc. A/CN.9/WG.IV/WP.71, http://www.un.or.at/uncitral/sessions/wg_ec/wp-71.htm..

United Nations Model Law on Electronic Commerce (available at http://www.un.or.at/uncitral/texts/electcom/ml-ec.htm)

Utah Digital Signature Act, Utah Code Annotated, § 46-3-101 (1996).

Weinberg, G. (1992) *Quality Software Management, Volume 1: Systems Thinking*, Dorset House.

Wright, B. (1997) *The Law of Electronic Commerce: EDI, E-mail, and Internet—Technology, Proof, and Liability*. 2nd Ed., Aspen Law & Business.

About the Authors

Cem Kaner serves as a participating observer in ***the National Conference of Commissioners on Uniform State Laws'*** (NCCUSL's) drafting committee ***for Article 2B of the Uniform Commercial Code (Law of Licensing)***, NCCUSL's drafting committee for a ***Uniform Electronic Transactions Act***, and as a member of the United States' ***Department of State's Advisory Committee on Private International Law: Study Group on Electronic Commerce***. He practices law, usually representing individual developers and small development services companies. He also consults on technical and management issues and teaches within the software development community. His book, ***Testing Computer Software***, received the ***Award of Excellence*** in the Society for Technical Communication's Northern California Technical Publications Competition. It is currently the best selling book in its area. He has managed every aspect of software development, including software development projects, software testing groups and user documentation groups. He has also worked as a programmer, a human factors analyst / UI designer, a salesperson, a technical writer, and an associate in an organization development consulting firm. He has also ***served pro bono*** as a Deputy District Attorney, as an investigator/mediator for Santa Clara County's Consumer Affairs Department, and as an Examiner for the California Quality Awards. He holds a B.A. (Math, Philosophy), a J.D. (law degree), and a Ph.D. (Experimental Psychology) and is Certified in Quality Engineering by the American Society for Quality. He teaches at UC Berkeley Extension and at UC Santa Cruz Extension, and by private arrangement, on software testing and on the law of software quality.

Brian Lawrence is an author and consultant with extensive experience in the software industry. He teaches and facilitates requirements analysis, peer reviews, project planning, risk management, life cycles, and design specification techniques. Brian has worked in the industry as a Software Process Engineer, a Software Designer in Quality Assurance, as well as a Development Manager, and Developer. He is formally trained in software process assessment by the Software Engineering Institute and in ISO-9000 registration by the National Standards Authority of Ireland. Brian served as program chair for paper selection for the 1997 Software Engineering Process Group Conference and is the industrial program chair for the IEEE Computer Society's 1998 International Conference on Requirements Engineering. Brian is a participant in Jerry Weinberg's 1996 Software Engineering Management Group and is a member of the ACM and the IEEE Computer Society. Brian is an instructor at the University of California Santa Cruz Extension program in software engineering.

Bob Johnson has over 18 years experience in software engineering with key strengths in building applications on a variety of hardware and software platforms. He has implemented, maintained, and tested security systems for the banking industry. Current assignments have focused on deploying Web-based applications requiring scalable, secure transactions. Bob has planned and implemented processes for the continuous improvement of software quality and on-time delivery through the introduction of standards, inspections, metrics, and technical improvement programs.

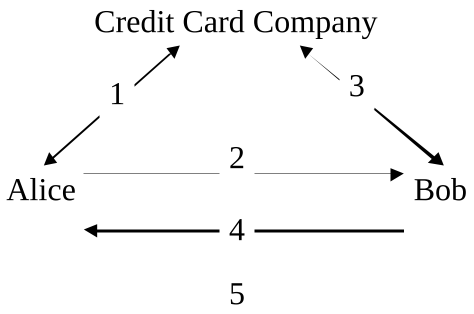

Figure 1. Alice buys stuff from Bob using her credit card.

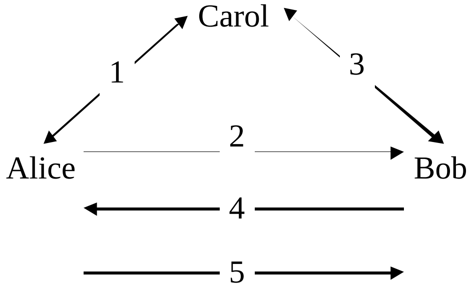

Figure 2. Alice buys stuff from Bob using her digital signature.

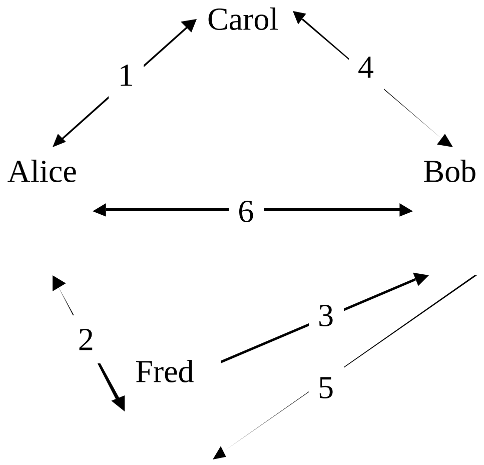

Figure 3. Fred impersonates Alice.