Trust is essential to most human transactions. A decision to trust is usually associated with an explicit or implicit assessment of risk. If risk is low, it’s easier to trust; if risk is high, trust is generally less willingly assumed. The use of credit cards for transactions illustrates some methods for risk mitigation—for example, credit-card issuers indemnify their users for anything over US$50 in personal liability, assuming the loss is reported promptly. This both reduces risk and promotes use. The merchant pays fees based on the credit-card issuer’s assessment of the merchant’s practices (face-to-face, over the phone, and over the Internet transactions have different degrees of validation and associated risk).

How do these notions apply to the Internet? We frequently trust email systems in the sense that we believe we’re receiving email from the asserted source. One reason we’re comfortable doing this is that the context and content of the received messages can often establish a basis for trusting that the source is accurate. Email from strangers should be treated with more suspicion unless, perhaps, introduced by a trusted source.

It’s worth observing that trust doesn’t always scale well. We can establish trust among a small group of people known to us, but it’s harder to achieve trusting relationships on a larger scale. Going back to the credit-card analogy, we’re able to trust merchants we don’t otherwise know because we know that the credit-card companies can withdraw payment for a fraudulent transaction. This is a clear example that risk mitigation enhances our willingness to trust an otherwise unknown party.

This preamble takes us to the question of trust in the use of the Internet. Its nearly 2 billion users probably trust the Internet’s applications more than they should but don’t engage in risk mitigation practices as much as they should. They use easily guessed passwords, frequently the same one for many accounts. They don’t change passwords often enough or at all. Service providers introduce risky methods for password recovery such as “secret” questions whose answers can sometimes be guessed or ascertained by searching the Web for information about the target user.

We make it even easier by putting personal information out on the Internet (through application profiles that contain our names, addresses, phone numbers, and other personally identifiable information). We certainly put a good deal of it up on social networking application sites! We know that there are compromises of these systems, and we know that personal computers can be compromised. But we assume this won’t happen to us, even though there’s no good basis for that belief. Even without all the various malware attacks and password-guessing incidents, computers and thumb drives are lost every day, most of them containing personal information or, worse, personal information about others.

This isn’t an argument that we should get off the Internet or stop using it. It’s too convenient and increasingly too necessary. Instead, it’s an argument for improving our security practices so as to mitigate risk and engender justifiable trust in the system. To start with, it would be a notable improvement to find affordable means to avoid reusable passwords. Rather, we want cryptographically supported, one-time passwords so that exposure doesn’t create a significant opportunity for reuse. Ideally, such a device could contain multiple identities and the capability to generate one-time passwords for each of them so that a single device can support distinct, strong authentication for different services. Plainly, the device itself needs to be protected so that its loss doesn’t compromise every identity it’s intended to protect. This might be done through biometrics or a password that activates it (something chosen not...
Trust and the Internet

to be easily guessed and presumably easily changed).

Alternatively, we might use public-key cryptographic methods to perform mutual authentication through the exchange of random, encrypted challenges. For this method to work, we must know and trust the other party's public key. A variety of methods can help us achieve that goal. As with all things Internet, we must think through various means of compromising the system (such as “man-in-the-middle” attacks) so as to reduce the risk of trusting the authentication exchanges and subsequent transactions.

In hierarchical organizations (the military, governments, typical enterprises), it's possible to mandate the use of security systems, such as smart cards and two-factor authentication. It's harder to achieve this goal for the general public because there is no special “forcing” function that induces the use of such technology. I've often wondered whether the system of automatic teller machines might yield an opportunity to create a smart card infrastructure at least in the domestic US.

A high percentage of all the cash dispensed in the US is taken from ATMs via conventional, magnetic stripe cards. If the banks concluded that smart cards that could produce non-reusable authenticators would make for a more secure system, it seems likely that most users would agree to use the smart cards for purposes of cash withdrawal. Once an infrastructure of smart cards and readers for use by the general public is in place, you could imagine a wide range of additional functions that could be enabled. Transactions requiring more than casual authentication might be supportable in such a system. If the cards were capable of holding more than one identity and strong authenticators for each, they would represent one possible implementation of the ideas discussed here.

These ideas aren’t new. It seems important, however, to begin a serious campaign to bring better technology and methods to the problem of authentication so as to reduce the risk inherent in trusting the Internet and its applications.

Vinton G. Cerf is vice president and chief Internet evangelist at Google. Contact him at vint@google.com.

Selected CS articles and columns are also available for free at http://ComputingNow.computer.org.