

When A Password Is Not A Password

Kenneth P. Weiss

Security Dynamics, Inc.
One Alewife Center
Cambridge, MA 02140-2312

Abstract

Single-factor password technology is not appropriate in today's networked environment. Powerful workstations, file servers, and remotely accessible mainframes require new high tech computer security solutions. Often, the technologies available and the reasons for them are not well understood.

This presentation focuses on the less obvious vulnerabilities of information systems and the philosophy and technology behind two-factor identification, authentication, and access control systems; and how they can be technically overlaid for implementation in existing environments. The theoretical issues associated with using ignition keys stored in tokens versus transmitting them in both clear text and encrypted form are discussed.

Introduction

Would you allow average white-collar employees to manipulate, privately, stacks of uncounted hundred-dollar bills? How much care would you expect them to take to protect these assets? Would those employees go out of their way to ensure the security of that money? Would you make them accountable? Would you hesitate to require them to keep the stacks locked up? Would you entertain a complaint that locking the vault or protecting the key was inconvenient? If you found the combination to the vault taped to it or in a desk drawer, would you tolerate that casual treatment?

Now consider the fact that, daily, many of your employees *are* entrusted with "stacks of *thousand-*

dollar bills"—the (very real, but difficult to quantify) inestimably valuable computerized resources on which the future of your company or organization depends.

Those traditionally charged with the responsibility for protecting and maintaining corporate resources—CEOs, CFOs, Boards of Directors, etc.—often have very little understanding as to the degree of exposure and vulnerability they may have with respect to computer security. They may incorrectly think that the threat is outsiders and hackers. If they really understood the potential liability and the potential risks to corporate assets and to their reputations, they might shut down all networks and computer centers.

Those charged with implementing computer technology, distributed systems, and networks—primarily middle management—are usually over-burdened, under-budgeted, and often insensitive to the real security issues. Their focus is often on through-put and keeping user clients satisfied. Very few M.I.S. professionals have had specific, formal training in information security, so they are poorly prepared when it comes to fundamental concepts and theoretical underpinnings of effective information resource security.

Those using the technology want user friendliness and do not understand the nature of the threats, potential methods of abuse, or the value of some of the information resources they are manipulating. They often resist anything like security procedures and back-up responsibilities that they perceive as impeding their goals.

100

Dr. Kenneth P. Weiss is an internationally recognized authority, author, and spokesperson on computer security. He is Chairman and Chief Technical Officer of Security Dynamics, headquartered in Cambridge, Massachusetts (USA).

CH2892-8/90/0000-0100 \$1.00 © 1990 IEEE

It should be noted, here, that requiring reasonable computer security procedures of end users is not an area with which they have to agree. While there is significant value in winning over the hearts and minds of end users with respect to computer security, this goal should not be a necessary condition at the expense of effective, responsible, and relevant information security.

Today, all executives are being held to new and higher standards of accountability, both because society is growing increasingly litigious and because, in a ferociously competitive marketplace, jobs are on the line, companies and their reputations are at risk, and our wealth of future technological advantage is threatened. Executives are responsible for protecting the integrity, availability, confidentiality, and privacy of corporate information resources.

The Seven Threats

In most environments, computer security is often too narrowly perceived. In banking the primary threat is perceived to be embezzlement, while in government it is often espionage. A hospital would be concerned about patient privacy, and a pharmaceutical company, formulas, test data, and trade secrets. Public utilities are often concerned with terrorists and sabotage. In actuality, the vulnerability of most corporations subsumes the entire range of possible computer abuse.

The reader may find it useful to conceptualize and categorize the range of threats to computer systems and information integrity by what I call the **7 E's: Error, Embezzlement, Ego, Eavesdropping, Espionage, Enmity, and Extortion.** Responsible management must implement a comprehensive computer strategy which is sensitive to **all seven** threat areas of computer abuse.

Because the potential scenarios for computer abuse are so pervasive (7 E's), the financial damage inflicted is difficult to quantify in dollars. Responsible estimates of the *annual* loss have ranged from five hundred million dollars (provided by Ernst & Young) to fifteen billion dollars, world

wide (provided by Coopers & Lybrand). The FBI estimates the average computer crime at \$400,000.

If one were to extrapolate from the most credible sources and currently available hard facts, an estimate of annual direct dollar loss of at least two billion dollars in the U. S. seems reasonable. However, this loss estimate (mostly Embezzlement and industrial Espionage) is, at best, a misleading tip of the iceberg.

How do you calculate the cost of the estimated sixty thousand hours of debugging time needed after Robert Morris' Virus/Worm attack. Although Morris was caught, convicted, fined \$10,000, and sentenced to "three years probation and community service," this is poor consolation to the users of the six thousand networked computers that were harmed (Ego & Error).

How do you determine a value for the loss to businesses and the public from this year's AT&T outage on Martin Luther King Day? At this writing AT&T has not stated definitively if the problem was a program bug or sabotage (Error or Enmity). We may never know.

How do we put a value on the losses to National Security when, for nearly two years, the Hanover Hacker, Marcus Hess, roamed through over 400 Defense Department, government, and government contractors' computers (Espionage)? Or the loss to the United States as a result of almost twenty years of espionage by John Walker, whose crimes were made possible, in part, by inappropriate policies, technology, and reliance on centrally-(NSA)-controlled encryption keys used too widely throughout government and the DoD?

How about Shadowhawk, the juvenile delinquent who, using his home computer, gained unauthorized remote access to AT&T computers in Georgia, Illinois, New Jersey, and North Carolina, stealing, among other things, copyrighted AT&T source code valued at over \$1.1 million (Ego & Embezzlement)? The hacker's sentence, nine months imprisonment, thirty months probation, and \$10,000 restitution, doesn't even come close to balancing the scales, dollar for dollar, and is

pitifully insufficient when compared to the incalculable loss of prestige, market potential, and competitive advantage suffered by his victims.

Consider, also, that Shadowhawk published his methods on electronic bulletin boards (Ego & Enmity), giving untold numbers of others the necessary means to duplicate his crimes.

Certainly there are people in corporate America who would have a difficult time placing a value on the public relations and other losses caused by "whistle blowers" or "leakers" who illicitly examine or copy documents from E-mail systems and corporate files and make them public (Eavesdropping). Managers in several companies have reported incidents where money was demanded (Extortion) to decrypt files, return stolen data, or disable a time or logic bomb, but such crimes, when successful, may never be documented, because the victims would not want the general public to be aware of their exploitation or even of their vulnerability.

Industrial Espionage can vary in severity and impact from the theft of mailing lists to threatening our national base of innovative, proprietary technology and future technological advantage. The *acquisition* of IBM proprietary trade secrets and copyrighted materials by Hitachi and Fujitsu are far from isolated examples. Bob Wade, of Motorola's Secure Telecommunications Strategic Business Unit, recently wrote that the annual loss through industrial espionage alone is estimated at one billion dollars.

While it is instructive to provide examples of the vulnerability of both our corporate and national infrastructure, it is more useful to examine why the vulnerability exists and what responsible professionals can do to protect resources.

Computer Security in the Information Age

It is easy to understand how we evolved into our current predicament. In the corporate environment, the rate of acceptance and proliferation of computer

systems, desk top computers, work stations, and networks has outstripped the rate for any other technology, even photocopiers. (Does anyone still use carbon paper?)

The only phenomenon that has grown faster than the proliferation of information systems is corporate exposure and vulnerability. The range of sensitivity and value of information stored and manipulated on terminals and desk top computers in a commercial environment is broader than what was traditionally available with hard-copy files and filing cabinets. The centralization of information resources, including personnel, financial, electronic and voice mail, marketing, and technical information, such as engineering drawings, proprietary formulae, and patent information, combined with broad network and host access, file servers, and high density storage available within work stations, has produced, in less than a decade, corporate vulnerability that is disproportionate to any phenomenon previously experienced by the corporate community.

Corporate office policies and procedures, which have evolved over the last two centuries for effective protection of resources, have been reversed, obviated, or obliterated in less than a half dozen years. For example, no responsible employee would allow a sensitive document to remain in an unattended typewriter or leave a sensitive file, engineering drawing, or critical formula exposed on a desk. But these same employees think little about leaving more sensitive material, *in more available and exploitable form*, stored in a centralized computer, PC, or work station.

Part of the problem is the fact that computerized data and information is not as visible or obvious as an open file folder or a document in a typewriter. The vulnerabilities or potential for abuse is not understood by most managers. Out of sight, out of mind! A false perception of security is far worse than no security at all!

Today's corporate desk top computers and work stations have almost as much power and memory as

many corporate computer centers had only a decade ago. But whereas the "old" computer centers' batch environment was once likely to be the most secure environment in the corporate structure, ministered to by specialists in white lab coats in air-conditioned "inner sanctums," today's networked desk top computer or terminal by contrast, is often seen as just another fixture in a typical office.

In such an environment, the interaction of human nature, reasonable security, and corporate policies has to be reconceptualized. The type of information asset manipulated or created at a terminal or desk top computer—and the data resources that may be available to it through local area networks or other network facilities—have created a corporate exposure and vulnerability that is unprecedented.

Guidelines for Securing Computer Resources

The weakest link in most computer security systems, *limiting the value of subsequent security technologies*, is reliance on inappropriate methods for identifying and authenticating authorized users and excluding all others. By implementing new "token" technologies, the threat of outsider abuse can be virtually eliminated and a substantial reduction of the threats posed by insiders can be accomplished while retaining user convenience, the power of networks, remote access, and centralized file serving.

The biggest impediment to providing appropriate security in today's environment is the lack of an enlightened and motivated management, willing to initiate user education programs, invest in appropriate technology, establish effective policies, and monitor and *enforce* compliance.

Classification Scheme

In the corporate environment, in addition to appropriate policies and procedures, computer resource security approaches can be classified as belonging to five fundamental groups:

1. Physical security — isolation and protection of certain computing equipment and data media from users;
2. Authentication of the identity of authorized users — access permitted to the authorized and access denied to the unauthorized;
3. Fine-grained definition of privileges — limits on what authorized users can do;
4. Encryption of sensitive transmitted and stored information; and
5. Audit trails, with meaningful accountability.

This classification scheme demonstrates a basic and essential requirement—the **prerequisite**, hierarchical relationship among components of the fundamental groups. For example, it is meaningless to implement privilege definition and audit trails without reliably identifying and providing access only to authorized users. Similarly, unrestricted access to the physical computing hardware and software media render ID authentication, privilege definition, and audit trails all meaningless.

Enhancing Identification Security on Computer Systems

The two major challenges to increasing security in host processors, file servers, and networks have been *cost* and *convenience*. What has been needed is a cost-effective increase in the level of security in authenticating the identity of authorized individuals without burdening either the user or the system manager. We need to maintain the convenience, portability, and flexibility of a simple password, while, at the same time, exponentially increasing system security. The way to accomplish these goals is to supplement the traditional password with another authentication mechanism.

Passwords have been used for thousands of years to authenticate the identity (ID) of an individual. They were the first security system

implemented on computers thirty years ago and represent the most common, if inappropriate (when used alone), security technology used in today's computer environment.

A password is representative of one of the only three acceptable basic methods of authenticating the identity of an individual. These three methods include:

- (1) **Something known:** a memorized secret word, phrase, number, code, or fact known only to an individual user...and his or her computer.
- (2) **Something possessed:** a discrete "token" which strongly resists counterfeiting, such as a signet ring, key, credit card, badge, etc.
- (3) **Something one is:** a measurable personal characteristic (a "biometric"), such as a fingerprint, signature, retinal pattern, voice print, etc.

Virtually all authorities today agree that an effective computer security system requires a combination of **at least two independent** ID authenticators. However, many security administrators inappropriately rely on popular existing software to protect their resources.

For example, excellent programs have been developed for IBM systems such as RACF, ACF2, and TOP SECRET, for control of access to specific applications (privileges) and to generate audit trails. Unfortunately, these systems alone are less than worthless if you can not depend on the authenticated identity of the user. An unsophisticated abuser, co-opting a colleague's password can masquerade as someone else, often with high privilege, and have the audit trail point to another.

A sophisticated abuser, taking advantage of weaknesses in the typically implemented twenty-year-old IBM operating system (MVS) computer architecture, can often get directly into an application, circumventing the audit trail.

IBM's MVS, and other, even more secure operating systems, are solely dependent on passwords— thirty-year-old, *one factor* technology—to reliably identify the individual for computer access. Passwords, alone, are simply not appropriate for today's resource protection.

Two Factor Identification

An example of *two-factor* ID authentication can be found in the ATM (automatic teller machine) cards in wide use today. To use the card a depositor is first required to insert the card into a "card reader" at the ATM terminal, which reads data stored in the card's "magnetic tape" and then demands a second identifier, the user's memorized password or PIN (personal identification number). In other words, the ATM terminal requires the customer to reveal "something known," the PIN, in order to verify authorized access to "something possessed," the ATM card, thereby demonstrating that this was the particular card (token) assigned to the particular customer.

It is ironic that a child with \$50 in the bank gets the protection of a two-factor ATM ID authentication, while many corporate officers and managers—with millions of dollars worth of assets under their direct control—can issue orders, and can move, manipulate, or erase major corporate assets, through a computer system protected by a mere password.

Corporate America is still a generation behind the kid at the ATM when it comes to sophisticated security—the heart of which is always user identification and authentication. Even the credit cards used to pay for gas or lunch use two ID authenticators—the presence of the card, perhaps validated by a mag-tape reader, and a biometric, your signature. With millions of installed PCs and terminals, however, the economics involved in adding card reading devices at everyone's keyboards have been, and will most certainly continue to be, prohibitive.

Any strategy to develop a creative solution should emphasize three overriding criteria: greatly

increased security; end user convenience; and flexibility. These lead to the following design goals:

1. Any new scheme must be at least several orders of magnitude more secure than existing technologies.
2. The convenience, portability, and ease of use associated with passwords should be maintained.
3. No additional equipment should be required at the physical terminal.
4. If a token were used, it should be as convenient to carry as a credit card, (hopefully very similar in size).
5. A token should strongly resist counterfeiting.
6. The computer access token should support added value by: being adaptable as an ID badge; providing over-the-phone identity authentication; allowing for additional uses such as physical access control and encryption key generation.

Token-based ID Authentication

Combining a password with an uncounterfeitable device or "token" is one way of providing reliable authentication of the identity of computer users, making privilege definition and audit trails meaningful. But how can a *two-factor* identification system be economically overlaid on the complex systems which already exist?

One solution is to take full advantage of recent advances in microprocessors, LCDs, and miniature power supplies and to design a credit card-sized device which can electronically "display" a pseudo-random code, unique to an individual. If such a card could automatically change the code in its visual display, say, every 30 or 60 seconds, or with every use—and a synchronized host computer was programmed to accept that card's displayed code only while it was being displayed, and then but

once—then the risk of electronic eavesdropping or casual observation evaporates. If the card would self-destruct upon tampering, then the risk of counterfeiting disappears.

Password That Are Not Passwords

Several token based products have been developed to provide two-factor computer access security. The use of these devices requires what appears to be two passwords: one classic and conventional, the PIN; the other, only superficially similar, the pseudo-random code displayed on the token device or card (i.e., the cardcode).

They are, of course, very different. They share only convenience, portability, and ease of use. The second ID validator, the automatically-changing and unpredictable cardcode displayed on the card's LCD screen, becomes concrete evidence (without a card reader) that the card (token) is at that point in time available to the user. It is a coded representation of the possession of an uncounterfeitable token. A user's eyes are the card reader, the existing terminal keyboard is the data entry device, the code entered is *a password that is not a conventional "password."*

It definitely isn't something secret or memorized (the classical definition of a "password.") There is no secret in the displayed numbers.

Time-Synchronized Dynamic Tokens

One of the strengths of such a time-synchronized technology is that the codes generated and displayed by the card can not be known ahead of time, memorized, guessed, or even loaned by anyone. There is no pattern; prior codes become irrelevant and *future codes are unavailable*.

The user simply reads the alphanumeric characters displayed by the card, types them in at the terminal or keyboard, in combination with his or her memorized secret PIN, and, now, any PC or dumb terminal captures two ID authenticators. Corporate America, finally, can afford the *two*

factor class of security that the kid at the ATM has had for years, but with a level of security, portability, and convenience that far exceeds that of the ATM machine.

The use of the word "password" in this second context (so different in form and function from the traditional memorized password) has led to some confusion. IBM uses the phrase "static password" to differentiate the memorized password from a constantly changing "one-time password" generated by a token device. Unfortunately, that's a distinction rarely heard outside MIS (Management Information Systems).

The memorized or "static" password is, and will continue to be, the hoary work horse of end-user ID authentication in computer and communications security. The many problems associated with its use—the necessity of frequent password changes, the irrepressible habits of users who write them down, the vulnerability to statistical attacks and guessing, the ease of casual interception in networks, and the hassles of password administration—are industry lore, but the memorized password remains the bedrock. It is the first and basic level of user authentication.

The "one-time password"—in this example, produced by the token—can replace the memorized password in identifying the user to computer system, file server, or network. It can also, as we at Security Dynamics require of our products, be combined with a second authenticator to supplement the first. In installations, the system at the host demands, first, an assigned, secret, and memorized password, and only second, the cardcode, which, at that moment, is being displayed on the ID card's LCD screen. Together, they form a PASSCODE.

The use of this second *independent* token-based identifier vastly increases the certainty of end-user authentication—a claim even IBM now endorses. Managers need no longer worry about one co-worker learning another's "password," the headache of password administration is greatly simplified, privilege definition becomes meaningful, outsider

and hacker-related threats virtually disappear, and internal threats—always more prevalent and serious—are controlled because audit trails finally offer solid accountability.

Threats posed by outsiders and privileged insiders can only be reduced when proper attention is directed to reliably authenticating user identity, providing access only to authorized users with limited privileges, implementing audit trails with true accountability, and establishing meaningful social and legal sanctions. A meaningful audit trail for authorized users is almost as important as keeping unauthorized users out.

The PASSCODE requires two *independent* elements—something known (the memorized password) and something possessed (the ID card and its displayed cardcode). A lost card becomes useless without its complement, the memorized password. The memorized password is worthless without the uncounterfeitable card token to generate the one-time cardcode.

It is in the nature of security, from castles to CPU's, that walls can always be higher and thicker. In a networked universe, token-based identifiers (with their "one-time passwords") will become a minimal standard. Also, their security can be painlessly and dramatically reinforced if they are used to complement rather than displace a traditional memorized password. And when cost-effective, useful, and reliable biometric options become available, they will probably be used to further buttress the gate by incorporating that technology into ID authentication systems as well.

Security Tokens or "Token" Security

Tokens should be used to *augment, not replace* either a password or a biometric. When this is accomplished, then an individual identity is verified by *true, two-factor* technology. Some products store the memorized PIN within the token. The PIN is then used as an ignition key. The device will not function without the PIN, but if the token is subjected to a sophisticated attack, the PIN can be exposed. This "token" technology is probably

appropriate in some limited environments, but must be considered one-factor security. When the only PIN used is physically stored (semi-permanently) in the token device, the technology is reduced to the lowest common denominator—something held, the token and only the token. Should such a device be technically compromised, all is lost!

Another class of security token device, called "semi-synchronous" by some vendors, was developed in order to eliminate the inconvenience of challenge-response technology (described below). Every time a button is pushed, the device provides a one-time code. In some environments, this may also be a useful technology; however, there are several problem procedures and security weaknesses associated with the technology. For example, there is the ever-present problem of synchronization between the portable token and the host, if the button is pressed independent of a log on. These devices can also be easily defeated or compromised if a user writes down several codes ahead, either for himself or to loan to others, or if someone surreptitiously demands a series of future codes without the knowledge of the device's owner.

A computer security product, which can be easily misused, provides only "token" security.

Open Networks and Secure Passwords

The proliferation of networks has created great and valid concern about the security of passwords transmitted in cleartext. A new technology is available which protects the secrecy and integrity of passwords, without the expense and complications of full network encryption.

The great strength of the one-time password generated by a hand-held ID card lies in its computational unpredictability. The displayed cardcode is, in part, the result of a mathematical process known as a one-way function (with data loss). There is no known way to reverse the calculation, or predict or compute a fraudulent cardcode, even with the card in hand for study. Absolutely nothing transmitted over the network

will ever allow an intruder to later, or sequentially, gain illicit access to a protected system.

To give full protection to the memorized password, the first of the two independent ID authenticators suggested, a product has been developed (i.e., the SecurID "P Card," with the "P" for "PINPAD"). The P card, still the size of a credit card, has a pad of pressure-sensitive keys built into it. A user enters his memorized PIN into this card, and the displayed sum of the two separate authentication codes can then be transmitted over an open line with full assurance that an eavesdropper gains absolutely nothing if the resulting PASSCODE is intercepted.

A constant added to a random number produces only a random sum. No cleartext password is ever transmitted; nor is the user's memorized password ever stored within the card. The card does not compare or validate the entered PIN; it simply adds it to the next random number generated by the card, within the card. The card then automatically zeroes the PIN.

It is imperative that the PIN not be stored within the card. To do so would reduce security to the lowest common denominator—something possessed, the token and *only* the token. Should such a device be technically compromised, all is lost. That can't happen when the PIN is combined with the random number.

Challenge-Response

Another kind of security token requires the user to turn on the device and enter a correct PIN into it. Then a public ID is entered into the terminal. The host computer then requests a password. The user must then type the memorized password into the terminal. If the password is correct, the terminal displays a challenge code. The code is then entered into the device. An encryption button of the device is pressed by the user, and the device then encrypts and displays a response. The user then types the response into the terminal. If the encryption is correct, the user has proved that he or she knows something secret and possesses a discrete token,

uniquely able to encrypt the data. Identity is authenticated and access is allowed.

Sidewalks and Architects

I once suggested to an architect that he wait six months before installing paved walkways on a new college campus and then create the paved walkways where the students and faculty had made paths (and killed the grass). The plan was successful! When the paved walkways were installed, everyone used them. Some were in less obvious places. Some connected to each other. They were all used—willingly used. There was no need for *Keep Off The Grass* signs, because no one walked on the grass. Corners were not cut. The pavement went where they wanted to go.

The process employed was called an unobtrusive, non-reactive measurement. The technology is borrowed from human factors engineering, which, in the computer industry today, is the art and science of what is commonly referred to as "user friendliness." "User friendly" can mean that doing it right is the easy way.

Nowhere is this more important than in computer security. Cumbersome, time-consuming security procedures will motivate end users to find ways around the security technology. When end users are forced into inappropriate compliance ("pre-paved walkways"), their lack of compliance and its effects are not as obvious as the downtrodden grass pattern that deviates from the paved walkway. In the information security environment, it is the corporation (not the grass) that is vulnerable. User "short cuts" and resistance to security procedures will not be as obvious as downtrodden grass, but they can be devastating to corporate resources, assets, and reputations.

Keep in mind that "cumbersome" and "time-consuming" are subject to individual perception. After all, we all know that we save only a few steps and very little time when we "cut" the corner of a paved walkway, but this knowledge does not save the grass at the corner!

Does a few more seconds logging on, or three or four more interactive steps make such a big difference if the new log on procedure properly identifies the user and locks out the unauthorized? You bet it does—a BIG difference! Many users will either delay making full use of resources by logging on less often, or they will find ways of defeating the purpose of security procedures, such as leaving sessions active when stepping away from their desks.

When procedures are perceived as cumbersome, authorized users see themselves as being unjustly penalized. Security ought to be a frustration to enemies, not friends. Computer users are pretty smart people and there are always lots more of them than security personnel, so security management really does not need to turn all those friends into adversaries. Ignore user friendliness in your security plans and that is exactly what will happen.

This perceived inconvenience is why acceptance of security tokens took so long, in spite of their obvious advantages. Early tokens depended on lengthy challenge-response procedures which were difficult to implement, even more difficult to use and were, in some instances, very inconveniently packaged. Users were unwilling to use technology that required four to six interactive steps and 30 to 45 seconds to log on. They found ways to "cut corners." Challenge-response technology is still around. Packaging has improved, and some now use "abbreviated" challenges, but the basic "user UNfriendliness" is still there. User resistance has not changed much either.

If a security procedure is perceived to be cumbersome or time-consuming, users will cut corners. Too often, the result is "token security," rather than properly used security tokens.