white paper

Positive Networks
Enabling work beyond the office

Two-Factor Authentication Without Tokens: PhoneFactor ™

Contents

For most companies, information security is a top priority. Demand for protecting data and employee confidentiality is only continuing to grow, especially in industries that require a regulatory-compliant environment. However, applying usernames and passwords for authentication is insufficient. While two-factor authentication is an effective security solution, traditional token-based systems have been difficult to implement and administer, leading to limited adoption.

PhoneFactor, a new product from Positive Networks, uses any mobile phone (or traditional phone) as the second authentication factor. Users do not need to carry an additional device, and there are no expensive tokens to manage. During login, PhoneFactor makes a call to the user's phone, confirming the authentication. This second factor - the possession of the phone itself - adds a significant additional layer of security. PhoneFactor can be set up in hours without the purchase of any hardware.

This paper discusses the technical architecture of PhoneFactor, along with related security, deployment and integration issues. For additional background on PhoneFactor, please visit www.phonefactor.net.

May 2007

Steve Dispensa
Chief Technology Officer
Positive Networks, Inc.

white paper

Positive Networks
Enabling work beyond the office

Two-Factor Authentication Without Tokens: PhoneFactor™

Introduction

Authentication, which is the process by which a computer system positively identifies a user, is commonly considered to be one of the weakest links in modern computer security systems. Every day a new story emerges about an identity theft or a computer break-in due to stolen credentials. With the proliferation of network-based applications, the trend is only going to continue. Unfortunately, the dominant authentication system in production today is based on usernames and passwords. This relatively weak system is subject to a number of flaws, including notoriously poor user password choices, password harvesting via keylogging software, phishing attacks, and others.

The most common solution to these authentication problems is to use a two-factor authentication system. Two-factor authentication works by requiring both **something the user has** *and* **something the user knows**, as opposed to just something known (typically a password). The "something you have" is usually a piece of hardware that is impossible (or at least very difficult) to duplicate, and the "something you know" is typically a password or PIN.

Two-factor authentication systems are secure because it is very difficult to obtain possession of both factors. Even if an attacker manages to learn the user's password, it is useless without also having physical possession of the token. Conversely, if the user happens to lose the physical token, the finder of that token won't be able to use it unless he or she can also guess the user's password.

Network administrators have deployed various two-factor solutions, but no solution has widely displaced traditional username and password authentication. The industry has seen deployments of token-based systems from vendors such as RSA and Verisign, smartcard-based solutions, and various forms of biometric authentication. Each solution has significant drawbacks, historically leading to limited adoption by users.

PhoneFactor provides IT departments a strong, two-factor authentication service without the drawbacks. It uses the public telephone network for the second authentication factor, which allows PhoneFactor to be deployed rapidly and inexpensively while maintaining the advantages of a two-factor authentication system.

white paper

Positive Networks
Enabling work beyond the office

PhoneFactor Overview

PhoneFactor implements two-factor authentication using the user's telephone as the second factor of authentication. The authentication process begins as a normal login to the system, in which the user supplies a username and password. If the supplied credentials are valid, the system initiates a phone call to the user's registered phone number. The user then answers the phone and indicates whether or not the authentication should succeed by entering an optional PIN.

Once the user acknowledges the authentication attempt via the phone call, the system completes the pending authentication and the login proceeds as normal. If the user did not request authentication, action can be taken to lock the user account and prevent an attack.

PhoneFactor's security rests on its use of the user's telephone as the second authentication factor. Telephones are extremely difficult to duplicate and phone numbers are extremely difficult to intercept. The combination of their phone, a physical possession, and a memorized password yields strong, two-factor authentication that provides minimal hassle to the user.

This architecture has a number of advantages over traditional two-factor systems. Most importantly, it doesn't require the user to carry an extra physical device. This is a huge improvement because users hate having to carry "yet another device." In addition, IT departments aren't excited about coordinating the logistics of issuing, mailing, RMAing, and servicing tokens. Most users already have mobile phones (there are about 1 billion active mobile subscriptions today), and they're going to carry them regardless. And most importantly, each device has a unique identifier that's almost impossible to copy: the phone number.

In addition to the advantage of using an existing device, PhoneFactor is also the *only* two-factor authentication system in the world that allows for instant attack detection. Every authentication attempt in which the attacker knows the user's username and password will generate a phone call to the (true) user. That user can immediately contact Positive Networks support by pressing a key combination during the authentication phone call. Support and the company IT department can instantly take appropriate action.

The phone-based system also improves resilience to phishing attacks. Phishing attacks generally work by fooling the user into entering credentials into a fake authentication form. This is possible because users generally have no way to authenticate the form itself. While most two-factor solutions can't solve this problem, PhoneFactor can leverage its unique, out-of-band authentication call to prove to the user that it really is PhoneFactor calling. Other solutions must rely

on application modification using technologies like SiteKey to work around the phishing problem.

This is not a theoretical problem. A recent phishing attack used this technique against a bank whose customers used RSA's SecurID™ product for two-factor authentication.

PhoneFactor Implementation Details

PhoneFactor is implemented as a web service and a suite of network and application agents. The web service is hosted by Positive Networks, and the agent software runs on the customer's premises as a part of the site's authentication infrastructure. PhoneFactor integrates with a wide variety of off-the-shelf applications and IT technologies.

PhoneFactor doesn't require changes to existing applications, so integration is typically fast and easy. PhoneFactor adds a second, confirmatory authentication to existing one-factor authentication architectures. Integration with existing applications and infrastructures is done using one of several protocols.

Most routers, RAS servers, VPN appliances, and other related hardware use the RADIUS protocol for authentication. RADIUS allows a central server to maintain a username and password database, allowing shared authentication databases and centralized account management. PhoneFactor plugs into the RADIUS protocol exchange by acting as a standard RADIUS proxy server. When a RADIUS request is made by an element to the PhoneFactor agent, it is first forwarded to the target RADIUS server. Once the authentication is successful, the agent triggers the secondary authentication phone call. If the user answers and approves the authentication, the user's authentication message is returned back to the original requesting device with an access-accept message. If the secondary authentication call fails, the requestor is notified of this fact and the entire authentication attempt fails.

Web-based applications running on Microsoft Internet Information Server integrate with PhoneFactor via a standard Microsoft IIS plug-in called an ISAPI filter. The filter implements both HTTP authentication support (basic and integrated) and form-based authentication (FBA) support. In the case of HTTP authentication, the browser supplies credentials with every request, and for requests that the web server authenticates successfully, the PhoneFactor filter triggers secondary phone-based authentication, in a similar manner. Authentication state is maintained by means of a session cookie.

In the case of form-based authentication (FBA) applications, PhoneFactor integrates using a combination of pre- and post-authentication. PhoneFactor never places a secondary authentication call to a user without first verifying that the primary authentication (using username and password) is successful. However, because there is no standard way to learn the status of FBA authentication after it takes place, and in order to avoid creating unnecessary state in web applications, the ISAPI filter performs a pre-authentication step before submitting the form's contents to the server for processing. During pre-authentication, the user's primary credentials (username and password) are tested against the authentication database, which in most cases is either Microsoft Active Directory or the local Windows user account database. If this pre-authentication step indicates that the credentials are valid, the secondary authentication call is placed to the user, and if that is also successful, the form is then submitted to the application. Of course, by that time it is known that the login will be successful. This mechanism ensures the broadest possible compatibility with web applications.

Finally, PhoneFactor supports integration with the Windows authentication process at either the workstation level or at the domain level. This allows users to receive secondary authentication calls for Windows domain login, drive mapping, Exchange access, and so on.

PhoneFactor agents are installed on various servers at the customer's site. The basic PhoneFactor package treats each server individually, and the available Enterprise Deployment Pack adds the ability to build sophisticated networks of PhoneFactor agents that share load and provide for fault tolerance.

Exactly which servers are used depends upon the types of authentication that are to be subjected to secondary confirmation by PhoneFactor. In general, the software is installed on each web server with a PhoneFactor-protected resource, and on each domain controller on a protected Windows domain. The software can be installed on any Windows computer when providing secondary authentication for RADIUS. In general, it's a good idea to plan appropriate levels of load balancing and redundancy into the deployment.

PhoneFactor runs as a Windows service and starts automatically on boot. Agents communicate with each other and with Positive Networks data centers using strong mutual authentication based on X.509 certificates and secure Windows RPC. All per-user data, including the actual list of PhoneFactor users, is stored on the agents at the customer site, and is never sent to Positive Networks other than on a one-at-a-time basis to trigger a secondary authentication call.

PhoneFactor Implementation Security Considerations

The PhoneFactor service was developed from the ground up, using the latest in secure software design methodologies. The entire system includes strong, mutual authentication, and all network communications are encrypted using high-strength cryptography algorithms. Industry-accepted cryptographic standards are used at every point in the design: authentication is based on X.509 certificates (both client and server), data transport is done using SSL or secure RPC (the same protocol used among domain controllers in Windows networks), and secure resources such as certificates and keys are stored using secure storage providers built into the operating system. Cryptographic-quality random numbers are used whenever randomness is needed.

The data architecture of PhoneFactor is designed to put administrators in total control of their authentication information. All per-user data is stored on the customer site, including the list of users enabled for PhoneFactor. The only information that is passed to Positive Networks during an authentication is the minimum necessary for appropriate auditing and for the placement of the secondary authentication call.

Administration of the system can be delegated to others via the PhoneFactor website, allowing clean sharing of management responsibilities while retaining a complete audit trail.

PhoneFactor As a Two-Factor Authentication System

The security of PhoneFactor lies in its ability to strongly authenticate users based on proof that they know a secret (their password) and are in physical possession of a unique physical device (their phone). While it may be possible (or even easy) for an attacker to gain access to a user's login credentials, it is usually a much more difficult problem to obtain that same user's phone.

Password theft is rampant on the Internet, and the proliferation of botnets and phishing sites means that stolen credentials will probably be a fact of life for years to come. PhoneFactor neutralizes the threat of password theft, since phones are not vulnerable to being stolen in the same ways that passwords are. While mobile phones can theoretically be diverted to third parties by phone companies, individual attackers have a much harder time gaining access to a phone number. Modern mobile phone networks make use of some form or other of encryption or other link-level security. Mobile phone "cloning" attacks have been described in academic papers, but the techniques they describe are aimed more

at unauthorized access to the network rather than taking over a particular phone number, and in any case, it is generally accepted that phone number hijacking is beyond the capabilities of most attackers.

But beyond securing a given single user from password theft attacks, PhoneFactor dramatically increases the difficulty of mass password theft attacks that are becoming so common with phishing sites and keyloggers. There's virtually no chance that an attacker will be able to use or sell a list of authentication credentials that are protected by PhoneFactor.

When the High Strength Authentication Pack is added, PhoneFactor can be configured in either Standard Mode or PIN Mode. Standard Mode requires only that the user answer the phone and press a button to prove that the authentication is expected and approved. In PIN mode, users are required to enter a PIN that functions essentially as a third authentication factor. This makes it even more difficult for an attacker that comes into possession of the user's phone to get logged in.

Existing single-step authentication solutions are often vulnerable to phishing attacks, where the user is tricked into supplying credentials to an attacker impersonating a legitimate application. Most existing two-factor solutions are vulnerable to this attack as well, because users have no way to know if the system they're communicating with is the genuine system or a man-in-the-middle.

PhoneFactor solves this problem as well, using either a user-supplied voice message (similar to recording a voice mail greeting) that only the legitimate PhoneFactor server could have, or by having the end user select a random authentication word for the PhoneFactor server to repeat to the user at every authentication call. The latter system is similar to the SiteKey™ system in use by many financial websites.

Comparisons With Other Systems

Two-factor systems have been on the market for years, and the basic concepts have been refined over that time such that the basic security of two-factor has been well established. However, every other system produced to date has had serious drawbacks that have prevented widespread deployment. Due to its unique phone-based architecture, PhoneFactor overcomes these drawbacks and sets the stage for easy,widespread deployment of two-factor technology for the first time.

One of the most common existing two-factor systems is based on hardware tokens that generate a pseudo-random sequence of digits. The most popular of

these systems is RSA Data Security's SecurID system. While these systems provide an additional level of security over legacy one-factor authentication systems, they bring with them several drawbacks. Nobody likes token management, and users don't enjoy waiting to receive tokens (or replacement tokens). IT departments don't like inventorying, mailing, and accounting for tokens. Furthermore, the logistics of token distribution become painful quickly as the number of users grows.

Tokens bring other operational issues as well. Most systems require some form of token synchronization, and not all applications support the token synchronization protocol. Synchronization often turns into a helpdesk call, and regardless, it represents yet another manual process that IT departments must manage. Because token-based systems require users to change their behavior substantially, significant training is needed. Users sometimes have a hard time remembering which order the PIN and the token digits are entered in, and training users to "wait for the bars" is difficult. Some systems even require administrators to modify applications before they will work, invoking all of the change control difficulties associated with non-standard vendor software.

Finally, tokens of this kind are typically expensive and time-limited. Add to that the expense of buying server hardware and software, training IT personnel to manage the system, the various costs associated with rollout/training, and you have a very expensive solution.

Another common two-factor solution involves the use of smartcards. Smartcards are credit card-sized tokens that have an embedded private key that is protected by a PIN or password. This private key positively identifies the user to the system. Like tokens, users are required to carry around a new object that they didn't have before. Cards must be mailed, handled; RMA'd, and so on, creating similar logistical problems. And, since very few modern computers (and virtually no older systems) have built-in smartcard readers, an additional piece of hardware, together with drivers (and associated platform dependencies), must be distributed to users. Smartcards present lockout risks – most cards deactivate themselves after a certain number of failed attempts, and require physical replacement – leading to increased IT management time, and in most cases require regular updating to stay current. Finally, few applications have native support for smartcard technology, and those that do often have narrow support for operating system versions, card reader models, and so on. Adding smartcard support to applications is often difficult.

Biometric authentication is another form of two-factor authentication that is occasionally deployed, although currently its use is typically only seen in ultra-high-security applications (e.g. the defense industry). Biometric devices are a

mixed bag when it comes to availability, compatibility, price, and security. For example, one well-known vendor distributes a fingerprint reader that uses an insecure data path to transmit biometric data back to the computer for analysis, making it vulnerable to the very keylogging attacks it tries to avoid.

In any case, biometric data cannot easily be changed in case it is compromised (e.g. fingerprint checksums are stolen, etc.), and integration is even more challenging and less available than with smartcards. Training is a significant issue, and logistics are perhaps more difficult than any other solution. The authenticator has to obtain the biometric data from the user the first time. Finally, one of the biggest reasons few companies employ biometric authentication is the cost of the hardware.

PhoneFactor generally does not suffer from any of the above challenges. Because users already have phones, there is no hardware distribution. If a user's phone is lost, it that user has the responsibility of replacing it, not the IT department. All that is required for provisioning is getting users' phone numbers, and often, that data is already available in a company directory. Because PhoneFactor adds a secondary authentication call to an existing authentication system, it doesn't require application-specific changes. It is compatible with a broad range of applications and uses standard, documented interfaces to add the secondary authentication step, thereby helping to ensure future compatibility. User training is simple; as long as a user is told to expect a phone call during login, the rest is self-explanatory. At worst, the training is explained during the authentication call itself. There are never synchronization or lock-out issues, and the user's computer needs no extra hardware or drivers. Because of how it integrates, PhoneFactor natively supports clients running on Macintosh, Linux and PDAs. As long as the server uses an industry-standard authentication method like RADIUS, HTTP authentication, FBA, or Windows authentication, PhoneFactor can support clients running on virtually any platform.

Conclusion

PhoneFactor enables new classes of users to experience the benefits of strong authentication without placing undue burdens on the IT department or on the budget. Rollout is fast and maintenance is easy. And, PhoneFactor is backed by Positive Networks, a company with extensive experience in managed security.

Usernames and passwords no longer provide enough security in many environments. For the first time, PhoneFactor will allow IT departments to roll out two-factor authentication to anyone, using any application, and provide companies the many benefits of strong authentication.