AT LAST *a Two-factor authentication*
without the hassle factor

Evaluating Strong Authentication Systems

Introduction & Goals

The purpose of this document is to provide the information required for you to evaluate the WiKID Strong Authentication System on its financial, technical and operational merits.

There are a number of drivers for adopting strong authentication: risks are increasing, the number of connected systems is increasing, the number of users requiring remote access is increasing, and regulations such as HIPAA and PCI are affecting more companies. Authentication is a key pillar for secure systems and enterprises are more and more aware that passwords are not a strong enough form of authentication. They have looked at hardware-based tokens and other solutions, but have not significantly adopted strong authentication.

There are three reasons why hardware tokens have failed to significantly penetrate the authentication market: convenience, extensibility, and cost. Users tend to dislike having the carry the additional hardware. If it's lost or left at home, it is very inconvenient. Further, tokens are a micro-point solution: they are only good for one authentication system at one enterprise. They cannot be used across multiple enterprises, nor are they very extensible inside the enterprise.

The WiKID Strong Authentication System address these concerns with a new approach built for today's authentication needs. The WiKID Strong Authentication System is the first and only two-factor authentication system capable of 100% end-user self-service. In most cases, it generates immediate cost savings. It is easy to implement and maintain and it is highly extensible. It is the first authentication system to handle both strong authentication and password resets out of the box.

A more recent entry to the two-factor authentication market is hosted services or Authentication as a service. Theses services host the server for you and provide either hardware tokens, software tokens, dial-back services or send OTPs via text message. They tend to be more expensive than WiKID and often have no cap on the costs. There are extra fees for SMS messages and telephony charges. Some services provide 'credits' up to a certain number of authentications, but there is no cap thereafter.

Hosted authentication services may also be problematic for companies facing regulation. PCI-DSS, for example, has fairly precise logging requirements. PCI mandates limited access to logs. Does using a hosted service comply? Are you comfortable not having administrative access over your authentication server?

Two-factor Authentication

There are three factors of authentication: something you know, something you have and something you are. Weak authentication only requires one factor, strong authentication more than one. Passwords are the most dominant form of weak authentication. ATM cards, the most ubiquitous two-factor system, require both knowledge of the PIN and possession of the card. WiKID Systems has developed a unique, patent-pending architecture that dramatically simplifies two-factor authentication and creates an extensible platform for future needs.

Historically, there are three types of strong authentication architectures for reader-less single-purpose hardware devices ("tokens"):

- Challenge-response – A challenge is issued and the response must match the expected response.
- Time-synchronous – The passcode changes periodically and must match the expected passcode. The system must deal with clock drift, usually by allowing multiple codes to be valid at any given time, reducing security. Time-synchronous soft-tokens are vulnerable to the generation of future valid-codes by moving the device clock forward.
- Event-synchronous – A counter on the device generates the next valid passcode, which must match the expected passcode on the server. The server must be able to "hunt" for a future valid passcode in case the counter on the device is moved ahead of the counter on the server.

The WiKID Strong Authentication System is based on a "***request-response"*** architecture. When the user wants to login to a service, they enter a PIN into the device; it is encrypted and sent as part of a request to the WiKID Strong Authentication Server. The WAS checks the encryption, validates the PIN and if the account is active and enabled, responds with an encrypted passcode. The device decrypts the passcode and the user enters it into, say, their VPN service, which in turn validates it with WAS. If the code matches, the user is granted access. When the client doesn't have Internet access such as when a cell phone is out of coverage, the client falls back into a challenge-response mode.

Development & Security Philosophy

Our goals in developing the WiKID Strong Authentication System are as follows:

- The system should be as secure as existing two-factor authentication solutions
- The system should be easier to use, maintain and administer than existing two-factor authentication systems
- The system should be less expensive in TCO than existing authentication systems – either passwords or tokens, relying on self-service as much as securely possible
- The system should be highly extensible, addressing multiple authentication needs as they arrive

It is our belief that a company can increase security while simultaneously reducing costs. We have designed and developed our system with these goals in mind.

The WiKID Architecture

The WiKID Strong Authentication System consists of the WiKID Strong Authentication Server (WAS), the WiKID Client and various protocol modules that connect to network clients such as a RADIUS server, firewall, LDAP directory, AD Domain server, etc.

The WiKID Strong Authentication Server

The WiKID Strong Authentication Server handles requests from network clients (while a network client may be a server, it is a client to WiKID), authentication (passcode) requests, logging, reporting, user management, certificate management, WiKID Domain creation, protocol module management and administrative preferences.

The WAS is running a firewall and is not accessible via Ping.

WiKID Domains

Each instance of the WAS runs under a particular security domain. The security domain is intended to segregate users with respect to access and services. For example, Intranet access may be provided with one domain, partner Extranet access with another and public Internet (Website) access with a third. Separate security policies can be provided for each domain and access can be granted on a device/individual user basis. The security policy includes PIN length, max bad PIN attempts, max bad passcode attempts, passcode lifetime, and max number of consecutive offline challenge-response logins. Upon creation, each domain generates a key pair for encryption within the passcode request/passcode reception process.

Each domain is represented by a 12-digit code, which represents the zero-padded IP address of the server (or zeropaddedipaddress.wikidsystems.net). The domains have both a device name and a server name, so that the Client can have a domain that is "VPN" on the client but refer to "Executive VPN" on the server.

The WiKID Client

The WiKID Client runs on a PC or a smart-phone (Blackberries, iPads/iPhones, Android phones or Windows Mobile devices). The client generates the public/private key pair and maintains the domain connection information. It does not store the PIN (PINs are stored on the WAS). The client can add new domains, delete domains and select domains for passcode requests. Unlike other two-factor systems, the WiKID client for each domain is the same, whether the domain is on the same WAS or for a completely different company.   This capability makes WiKID perfect for cross-enterprise strong authentication, which is increasingly important for today's extended supply chains.

There are two types of PC tokens, the standard token and the "locked" token. The standard token is protected by a PKCS 12 store. (No passphrase is required, though. Using a blank passphrase is accepted.) It is portable to other devices by anyone with basic technical skills. The locked token is also in a PKCS 12. However, on domain creation, the token grabs certain information from the PC such as CPU identifier or MAC address, hashes it and send the hash to the server. This same hash must be sent for each OTP request. In addition, the locked token requires a passphrase and uses a variable pin pad.

Protocol Modules

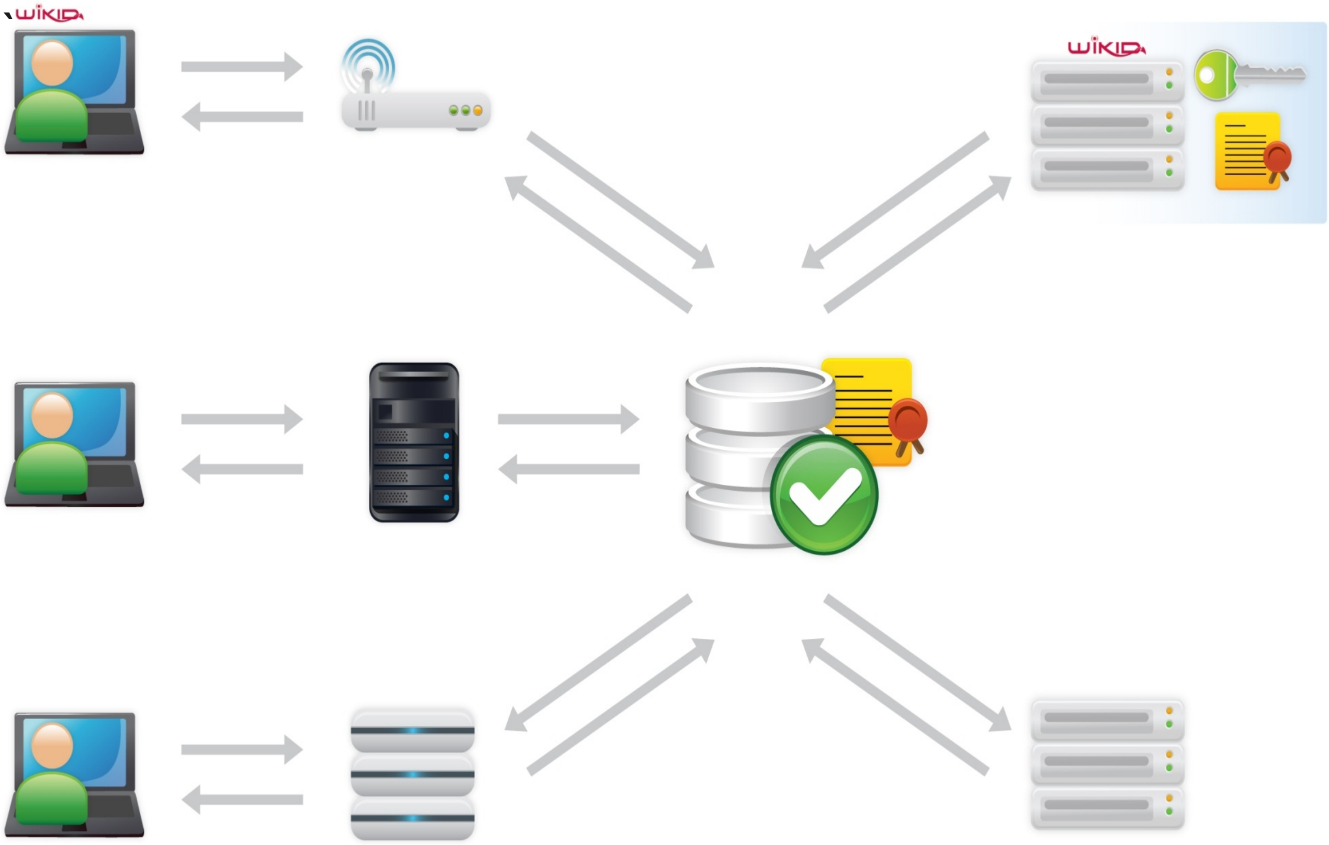
The WAS communicates to various network clients (VPN concentrators, firewalls, servers, etc) via protocol modules.   Currently, WiKID supports RADIUS., TACACS+, LDAP and Google SSP. The server also supports an API using our wAuth protocol. The API supports token registration, user management, authentications (of course!) and other functions so you can easily embed WiKID in your custom application. The server comes with example scripts you can edit to allow users to register their own tokens after logging in with their AD credentials.

Network Client

Network clients are simply the devices on the network responsible for granting access to the users: RADIUS servers, VPN concentrators, firewalls, routers, switches, applications, etc. They accept the username and passcode from the user, send the passcode to the WAS, and, if the WAS validates the code, they grant access. For a network client to be active on a security domain, it must be registered on the WAS. 11

Below is a sample diagram of a fully configured WIKID System:

Example Scenarios



This scenario is typical for our customers. They have deployed a Radius server such as NPS on AD, Freeradius or a Cisco ASA to manage authorization and authentication.

All requests from Radius clients pass through the Radius server to the directory and the WiKID Strong Authentication server. This configuration adds security because both authorization and authentication must pass. Disabling a user in one location locks them out entirely. AD administrators do not need to be WiKIDAdmin administrators.

Please see our free eGuide to adding two-factor authentication to your network for more examples: http://www.wikidsystems.com/webdemo/Two-factor_Authentication_in_your_Network_eGuide.pdf

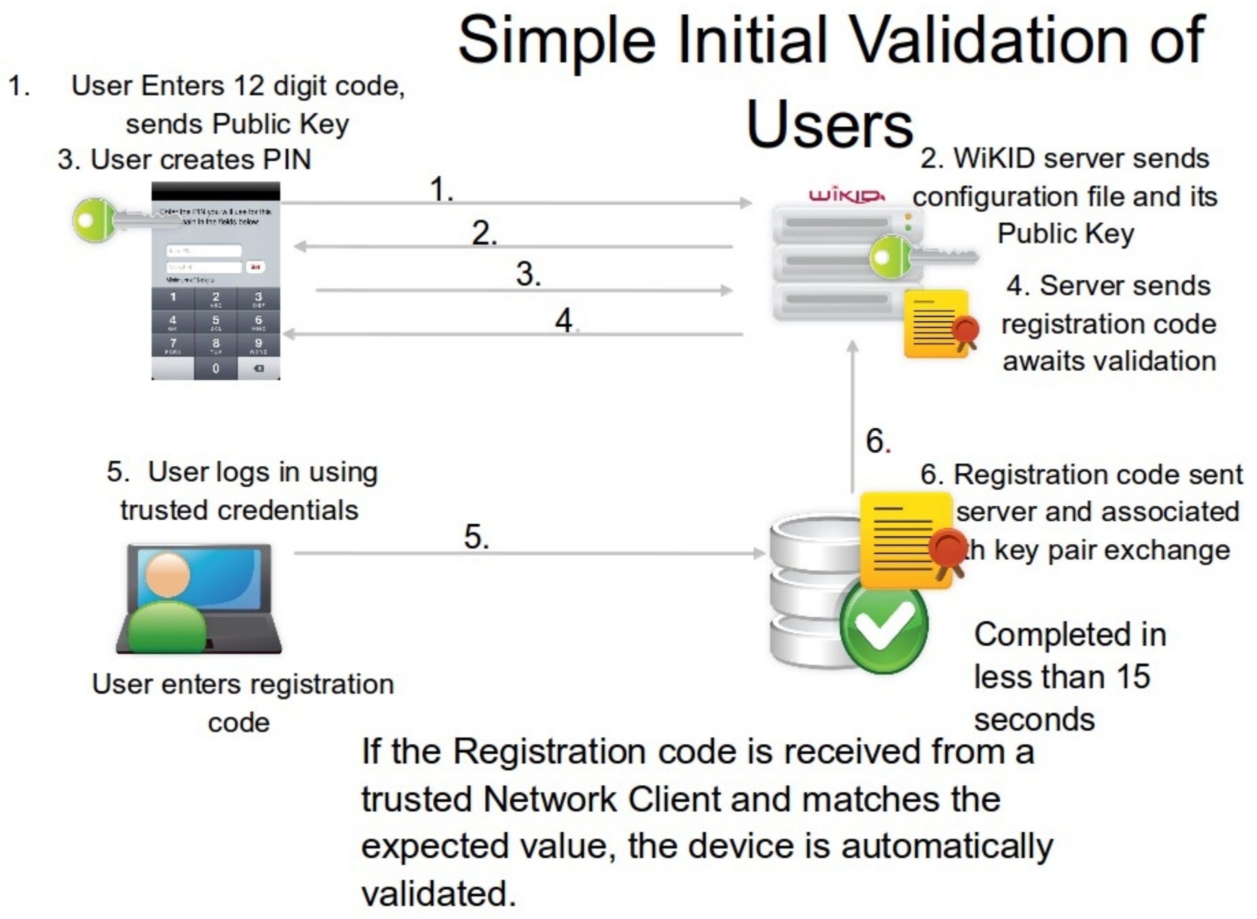
Operational Considerations

One of the reasons why two-factor authentication has failed to deeply penetrate the authentication market is due to the operational hassles of hardware tokens. WiKID Systems has eliminated the largest aggravations in deploying, managing and using two-factor authentication. By enabling employee self-service to the fullest extent possible, deploying a WiKID Strong Authentication System is easy, cost-effective and secure.

Deployment & Initial Validation

The WiKID Strong Authentication System is the only solution where initial validation can be completed automatically – 100% self-service by the end-user.   The WiKID Client as shipped contains no security information, only the ability to create the public/private key pair, so this client can be installed anywhere by anyone. (Many devices now support over-the-air download, making software deployment incredibly simple.) It is not until after the key pairs are exchanged between the WiKID Client and the WiKID Server *and* after that exchange has been validated through a second, trusted channel that a security relationship has been created.

Below is a diagram of how the initial validation can work in full self-service mode:



Simple Initial Validation of Users

1. User Enters 12 digit code, sends Public Key
3. User creates PIN

2. WiKID server sends configuration file and its Public Key

4. Server sends registration code awaits validation

5. User logs in using trusted credentials

User enters registration code

6. Registration code sent server and associated th key pair exchange

Completed in less than 15 seconds

If the Registration code is received from a trusted Network Client and matches the expected value, the device is automatically validated.

- ➢ The user enters the 12-digit WiKID Domain identifier. The WiKID Client creates the public/private key pair and sends its public key to the server requesting a configuration file for the domain.
- ➢ The WiKID Server responds with the configuration file and its public key, encrypted by the Client’s public key.
- ➢ The User is prompted for a PIN and told the minimum PIN length for that domain. The PIN is encrypted and sent to the Server.
- ➢ The Server decrypts the PIN and stores it. It returns a one-time registration code. The account has now been created, but is not validated. In order to validate the account, the WiKID Server must receive the same registration code from a trusted Network Client over a second channel.
- ➢ The registration code is sent to the WiKID Strong Authentication Server and the account is activated.
- ➢ The end-user logs into a small application using their AD credentials (this application is provided with the WiKID server and only requires slight editing). The are prompted to enter their registration code.
- ➢ The application can then optionally prompt them to add another token client.

Of course, the administrator can manually enable the user. Please note that the end-user can re-initialize the same way should they forget their PIN.

The PC tokens support pre-registration. In pre-registration, a list of usernames and registration codes are uploaded to the WiKID server. On the token, pre-registration is configured in the jw.properties file. When the user wants to add a pre-registration domain, they are prompted to double-enter their PIN and the pre-registration code. Once the server receives the PIN and code, the user name is validated.

With hard-tokens, the device must be associated with the user on the server and physically delivered to the user, a process that costs $15 or more. As you can see, there are any number of ways to securely register users with WiKID.

Management

The web-based interface of the WiKID Strong Authentication server is intuitive and easy to use. Security professionals will appreciate working on security, rather than the logistics of token management.

The WiKID Client is easily replaceable. There is no box of hardware tokens with limited lifetimes locked up in the administrator's office. If a device is lost, it is much easier to replace software. Seat licensing is much more convenient if you have a lot of contractors.

Convenience & Ease of use

WiKID Systems is committed to ease-of-use for the end-users and the administrators. Employees dislike having to carry additional hardware, such as hardware tokens. Most employees tend to like self-service applications and to dislike calling helpdesks. WiKID automated initial validation system is simple for employees to use. Our ability to reset LAN passwords is a perfect self-service application.

Portability – works anywhere

A key requirement for mobile workers is to be able to log in from any location, even from a kiosk or other non-company owned PC. WiKID provides the online request-response mode and falls back to the challenge-response mode when out of wireless coverage. It requires no reader and works everywhere.

Additionally, users can have more than one token – and all on the same seat license. So, a user can have a PC token on their corporate laptop and a token on their iPhone. The administrator can limit the types of tokens on a per-domain basis.

Extensibility

Counter-based and time-synchronous tokens have a one-to-one relationship with their authentication server. The WiKID Client, can have relationships with multiple WiKID Servers. This extensibility can save money internally and increase security externally.

The best example of the extensibility of WiKID is our support for **mutual https authentication**. Mutual https authentication is enabled on a domain by entering a Registered URL in the domain page. The server goes to the https URL, grabs the cert, hashes it and stores the hash. When the token requests an OTP, the server sends the OTP and the hash. Before presenting the user with the OTP, the token goes out through the user's Internet connection to the URL, grabs the cert, hashes it and compares the hash to the one from the server. If they match, the OTP is presented, the default browser is launched to the URL and the OTP is pasted to the clipboard. If they do not match, an error is shown indicating a potential man-in-the-middle attack.

Increasingly, companies are opening up their networks and applications to suppliers, vendors, consultants and other 3rd parties. While these tight ties have increased productivity and smoothed supply chains, they have also increased security risks. Do your vendors use strong authentication to log into your network? Do your employees use strong authentication when they log into your vendor's networks? The ease of deployment and low price-point of WiKID makes it practical to deploy strong authentication through the whole supply chain. Moreover, end-users can use the same client for both companies.   Your vendors will see the same financial benefits as you do, increase the value of the entire chain.

Scalability

Each request and response on the system is a mere 251 bytes. This small transaction size allows the server to handle a huge number of users. Multiple servers are more for fault-tolerance than scalability.

Future capabilities

Already WiKID Systems is the first authentication system to combine two-factor authentication with a password reset capability out of the box. The WiKID architecture provides for robust future flexibility through new clients, new protocol modules, new network clients and new technologies. As an example, we can add location as a 3rd factor once the wireless carriers publish location-based information.

Security/Technology

There are many flavors of two-factor authentication, some more secure than others. We believe that relative security is a very central factor in choosing an authentication solution. While cost savings, extensibility and manageability are important, without security, you don't want to create a false sense of security.

Relative Security Analysis Chart

Item	Notes
Two-factor	The combination of PIN and device is very strong
Passcodes random	There is no way to predict the passcodes or to brute-force attack the server
Passcode length random	Randomizing the passcode length protects against a race attack/man-in-the-middle attack on a fixed length response system (future release).
Only one passcode valid at any moment	Passcode lifetime can be set per domain by the administrator, which can't be done with a time-synchronous system.
Eliminates shoulder surfing, keyboard sniffers, Trojans	Passcode is only used once.
PINs and passcodes never sent over network together	In some systems, the PIN is appended with the passcode, which increases the risk of PIN compromise. With WiKID, the PIN and passcode are never transmitted together and are always asymmetrically encrypted.
Published algorithm	WiKID uses only published algorithms, increasing the security of the system through peer-review process. 2048-bit RSA or equivalent encryption is utilized.
Mutual HTTPS Authentication	PC tokens can validate the SSL certificate of a server for your users, thwarting man-in-the-middle attacks and greatly increasing the security of your web-applications or SSL-VPNs.
Risk from loss	Users more likely to keep wireless device separate from laptop, decreasing risk of combined loss. Tokens are often kept with laptop. A lost or stolen token is a nuisance. A lost cell phone is a financial risk for the user, aligning incentives.
No password file for attackers to target	Password files are the gold mine for attackers. WiKID removes that target.
PIN stored on server	There is no way to brute force attack the PIN as it is stored safely on the WiKID Strong Authentication Server. Certificates protected by passwords are subject to cloning and brute-force attacks on the password.
Network clients require a WiKID Server Certificate	Prevents a Denial-of-service attack from un-approved Network Clients.
Domain Security Options	Maximum bad PIN attempts Maximum bad passcode attempts Maximum consecutive challenge-response logins PIN length configurable Passcode lifetime
Cross-enterprise security	There is no reduction in security when multiple domains are created making cross-enterprise strong authentication viable for the first time. This capability fits well with Single Sign-On efforts such as Liberty Alliance.
Logging	Complete logging and reporting. Integration via Syslog is available.

Financial Considerations

Potential Cost Savings

The Request-Response architecture provides a flexible, extensible platform that significantly increases the cost savings potential from implementing the WiKID Strong Authentication System. Wherever possible, WiKID has enabled self-service and automation and we have attacked the costs associated with authentication.

The Costs of Passwords

Companies are aware that passwords are neither secure nor inexpensive. Employees seek to choose passwords that our easily remembered, which makes them easy to guess. The IT department wants a strong password policy – 8 to 12 characters, no words, alphanumeric and changing every 30-60 days. The result is calls to the helpdesk to reset forgotten passwords that cost $15-30 each. The average employee will call 4-5 times per year! For a 30,000 employee Fortune 500 company, that means $1-3,000,000 per year! For smaller firms the cost per call is higher. Moreover, most small and medium sized businesses (SMBs) don’t have 24x7 helpdesks. So if you’re an attorney billing $300 per hour and you’re locked out of the network after 5:00 PM, you’ve lost some serious money.

The cost of hardware tokens

The vast majority of companies still rely on usernames and passwords for remote authentication. Some companies have deployed one-time passwords systems, especially time-synchronous, hardware based tokens, but the majority of deployments are very small and the total penetration is minimal. Each token costs money; it is expensive (and an administrative hassle) to distribute tokens; the tokens have a limited life; the authentication servers are expensive both upfront and in ongoing costs.

The key costs for any two-factor authentication system are:

- Token client license fees
- Token distribution costs
- Server hardware costs
- Server hardware (although increasingly virtual instances are used)
- Server maintenance

Here is a comparison between WiKID and a leading token vendor's software tokens:

1 Year Software tokens

Number of users	100	500
Number of Years	1	1
Cost per Software Token	$22.63	$22.63
Total Token Cost	$2,263.00	$11,315.00
Server Cost	$4,232.00	$21,160.00
Server Maintenance @ 20% per year	$846.40	$4,232.00
Server Hardware	$1,100.00	$1,100.00
	$6,178.40	$26,492.00
Token Distribution	$2,000.00	$2,000.00
Token Deployment Software	$8,280.00	$8,280.00
Total Upfront Distribution	$10,380.00	$10,780.00
Total Costs competitors tokens	**$18,821.40**	**$48,587.00**

WiKID Systems

Seat License and Maintenance Fees (with pre-pay discount)	$2,400.00	$10,000.00
WiKID Server Softare	$0.00	$0.00
Server Hardware	$1,100.00	$1,100.00
Token Deployment Software	Included	Included
Token Distribution	$2,000.00	$2,000.00
Total Costs for WiKID	**$5,500.00**	**$13,100.00**
Savings	***$13,321.40***	***$35,487.00***
Saving as a percentage	***242.21%***	***270.89%***

WiKID is different from many vendors in that we post our pricing online and do not discount off it for any unstated reason. We are building a company designed to deliver low-cost goods and complexity is not supportable.

As for hosted two-factor authentication services, they appear to be quite expensive, starting at $36 per user per year, compared to WIKID's $24. That is not including the unlimited additional charges for telecom and text messaging. While there might be some cost to running WiKID in a virtual machine, you maintain control of yours server and the key to your kingdom.

Conclusion

More than ever, companies are seeking to get more for their money. Security traditionally has been viewed as a cost center that mitigates risk primarily. WiKID Systems takes the approach that there are always ways to improve systems to eliminate costs that can also improve security. The WiKID Strong Authentication System embodies this belief by attacking authentication costs and risks simultaneously and providing automated self-service capabilities that eliminate the overhead of typical strong authentication systems.   The WiKID Strong Authentication System improves upon traditional authentication systems operationally, in security capabilities and in financial terms.