Here, we extend our prior work to consider solutions to the key management problem, which surfaces in various contexts. Three of many potential examples include the following.

Below we will describe various examples of how key management can be beneficially layered on top of a QOOBA architecture. Our first example relates to digital signing. In applications that require digital signing, a user needs to be provisioned a private key and a digital certificate, i.e. a binding of the user's identity and public key as certified by a Certificate Authority. The use of such a private key, which is not known to any 3rd party, including the security server, provides for strong non-repudiation which is necessary for some applications. We follow the industry convention of referring to signatures created with public key cryptography as "digital signatures". As will be understood by those skilled in the art, signatures based on underlying symmetric cryptography with shared secrets, like that which the QOOBA system as described above already provides, are usually referred to as "electronic signatures".

Our second example relates to encrypted document delivery. When an encrypted file is sent to a user, for example a PDF of a brokerage statement, the user needs to be provided with the key with which the file was encrypted.

Our third example relates token authenticators. When users are provisioned a token authenticator, either for a one time password generator or a transaction authenticator, the user's token needs to be provided with a shared secret key. Those skilled in the art will recognize that in this context, the shared secret key is often characterized as a "seed").

In all these examples key management adds directly to the cost of the system, and indirectly effects the security. Keys need to be generated, distributed and maintained in sync. As keys can get lost, corrupted or stolen, key management is usually a significant source of costs, and a point of vulnerability in the system.

OBJECTIVES OF THE INVENTION

Accordingly, it is an objective of the present invention to provide an innovative way to leverage the QOOBA system to perform key management for digital signing, encryption and token seed management.

Additional objects, advantages, novel features of the present invention will become apparent to those skilled in the art from this disclosure, including the following detailed description, as well as by practice of the invention. While the invention is described below with reference to preferred embodiment(s), it should be understood that the invention is not limited thereto. Those of ordinary skill in the art having access to the teachings herein will recognize additional implementations, modifications, and embodiments, as well as other fields of use, which are within the scope of the invention as disclosed and claimed herein and with respect to which the invention could be of significant utility.

SUMMARY DISCLOSURE OF THE INVENTION

In accordance with certain preferred aspects of the invention, a security server can be operated so as to provide key management layered on a quasi out of band authentication system. A request for activation of a user interface window for a particular user at a network device, such as a desktop computer, associated with that user is received from the network device via a communication channel. At this point, the communication channel is non-secure or what is sometimes referred to as "clear". It will be recognized by those skilled in the art that having an non-secure channel at this point in a communication protocol is not unusual. For example, when a user initiates communications with an SSL protected website, an insecure or clear TCP/IP connection is initially set up, and only later is SSL security established on top of the non-secure or clear TCP/IP connection.

Next, to authenticate the user to the security server, an activation PIN (personal identification number) is transmitted to an out of band authentication (OOBA) system for forwarding to the user's telephone via a voice or text message. For example, the OOBA system may place a call to the user's home or cell phone, and convey the PIN to the user verbally, or may send a text message to the user's smart phone, e.g. the user's IPhone™ or Blackberry™, and convey the PIN to the user in written form. In either case, the user must enter, i.e. copy, the conveyed PIN into the user interface window, so that it will be conveyed back to the security server. In this way, the PIN previously transmitted by the security server to the OOBA system is received back by the security server from the user's network device via the

BRIEF DESCRIPTION OF DRAWINGS

Figure 1 depicts the main components of the flexible quasi out of band authentication architecture, in accordance with the present invention.

Figure 2 shows the flexible quasi out of band authentication architecture with the key management functionality layered on top, in accordance with the present invention.

PREFERRED EMBODIMENT(S) OF THE INVENTION

General Overview Of Prior Work Related To QOOBA

In prior work we have described how the introduction of a network based security server, which has an independent channel to a window displayed on a user device, can be used in conjunction with a user's browser, computer software or smart phone or other mobile communications device application, and the website they are visiting to provide user authentication for login or transaction authorization via one or more user network device.

QOOBA is an innovative approach to create a solution that can be used to authenticate every transaction in a manner that will feel natural to users. The central idea is to create a small secure window, the QOOBA Window, that has an independent encrypted channel to a secure server (the QOOBA Server). The QOOBA Window can be implemented as a zero-download browser pop-up (the QOOBA Pop-up Window), as a small desktop application (the QOOBA Software Window) or as an app on a smart phone (the QOOBA Phone Window). An important aspect to the innovation is that, unlike as with soft tokens, the QOOBA Window does not require secure storage of long term secrets. Rather, it is "activated" during login by using out of band authentication. As a user transacts at a web site, (e.g. a merchant or bank website, that is part of the QOOBA Network, the transactions that the web site believes the user intends are sent securely via the user's browser to the QOOBA Server which displays the transaction in the QOOBA Window. The user is also optionally shown a transaction signature that the user can cut and paste from the QOOBA Window into their browser to authenticate to the website for login or transaction authorization purposes.

Overview Of Prior Work Relating To A Flexible QOOBA Architecture

The QOOBA solution has the following benefits in terms of ease of use, total cost of ownership and, of particular interest here, security.

Overlaying Key Management On The QOOBA Architecture

We now describe how we overlay components for key management on the QOOBA architecture.

The QOOBA System

Referring now to Figure 1, in accordance with the present invention, the QOOBA system consists of a desktop personal computing device 100 having the QOOBA Window 110 and a Browser Window 112 executing and displayed thereon, a QOOBA Server 125 and Web Service 150, which has the QOOBA Application Programming Interface (API) 155 operable thereon. It should be understood that in a practical implementation there would typically be multiple websites. Also included in the system as shown is an OOBA Service 165, which is utilized by the QOOBA Server 125 to bootstrap authentication of the user using the user's phone 175, which may be a landline, cellphone or smartphone.

As described in more detail in the related applications referenced above, the user activates the QOOBA Window 110, typically by using out of band authentication via OOBA Service 165, and establishes a session with the QOOBA Server 125. Web Service 150 participates in the QOOBA Network and goes through a onetime set up process to establish a shared secret with the QOOBA Server 125, which is not shared with or known by the user. When the user has an active session with the QOOBA Server 125 via communication channel 450 and is also at the Website 150 via communication channel 400, the Website can use the QOOBA API 155 to request, via back end communication channel 500, transaction authentication by sending the transaction directly to the QOOBA Server 125. The QOOBA Server 125 then displays the transaction to the user in the applicable QOOBA Window, which is shown in Figure 1 to be Window 110.

The QOOBA Windows

The QOOBA Server 125 can present various information to the user in the displayed QOOBA Window 110. For example, as we have described in our prior work, the QOOBA Server 125 can display a transaction to the user in the QOOBA Window 110, and if requested, also display in the QOOBA Window 110 a transaction, i.e. electronic, signature derived from the transaction, the secret shared between the QOOBA Server 125 and the

Here, we extend our prior work to consider solutions to the key management problem, which surfaces in various contexts. Three of many potential examples include the following.

Below we will describe various examples of how key management can be beneficially layered on top of a QOOBA architecture. Our first example relates to digital signing. In applications that require digital signing, a user needs to be provisioned a private key and a digital certificate, i.e. a binding of the user's identity and public key as certified by a Certificate Authority. The use of such a private key, which is not known to any 3rd party, including the security server, provides for strong non-repudiation which is necessary for some applications. We follow the industry convention of referring to signatures created with public key cryptography as "digital signatures". As will be understood by those skilled in the art, signatures based on underlying symmetric cryptography with shared secrets, like that which the QOOBA system as described above already provides, are usually referred to as "electronic signatures".

Our second example relates to encrypted document delivery. When an encrypted file is sent to a user, for example a PDF of a brokerage statement, the user needs to be provided with the key with which the file was encrypted.

Our third example relates token authenticators. When users are provisioned a token authenticator, either for a one time password generator or a transaction authenticator, the user's token needs to be provided with a shared secret key. Those skilled in the art will recognize that in this context, the shared secret key is often characterized as a "seed").

In all these examples key management adds directly to the cost of the system, and indirectly effects the security. Keys need to be generated, distributed and maintained in sync. As keys can get lost, corrupted or stolen, key management is usually a significant source of costs, and a point of vulnerability in the system.

OBJECTIVES OF THE INVENTION

Implementation of the techniques disclosed herein may provide an innovative way to leverage the QOOBA system to perform key management for digital signing, encryption and token seed management.

Additional objects, advantages, novel features of the present invention will become apparent to those skilled in the art from this disclosure, including the following detailed description, as well as by practice of the invention. While the invention is described below with reference to preferred embodiment(s), it should be understood that the invention is not limited thereto. Those of ordinary skill in the art having access to the teachings herein will recognize additional implementations, modifications, and embodiments, as well as other fields of use, which are within the scope of the invention as disclosed and claimed herein and with respect to which the invention could be of significant utility.

SUMMARY DISCLOSURE OF THE INVENTION

The invention is defined in the independent claims. Some optional features of the invention are defined in the dependent claims.

In accordance with the techniques disclosed herein, a security server can be operated so as to provide key management layered on a quasi out of band authentication system. A request for activation of a user interface window for a particular user at a network device, such as a desktop computer, associated with that user is received from the network device via a communication channel. At this point, the communication channel is non-secure or what is sometimes referred to as "clear". It will be recognized by those skilled in the art that having an non-secure channel at this point in a communication protocol is not unusual. For example, when a user initiates communications with an SSL protected website, an insecure or clear TCP/IP connection is initially set up, and only later is SSL security established on top of the non-secure or clear TCP/IP connection.

Next, to authenticate the user to the security server, an activation PIN (personal identification number) is transmitted to an out of band authentication (OOBA) system for forwarding to the user's telephone via a voice or text message. For example, the OOBA system may place a call to the user's home or cell phone, and convey the PIN to the user verbally, or may send a text message to the user's smart phone, e.g. the user's IPhone ™ or Blackberry ™, and convey the PIN to the user in written form. In either case, the user may enter, i.e. copy, the conveyed PIN into the user interface window, so that it will be conveyed back to the security server. In this way, the PIN previously transmitted by the security server to the OOBA system is received back by the security server from the user's network device via the

BRIEF DESCRIPTION OF DRAWINGS

Figure 1 depicts the main components of an exemplary flexible quasi out of band authentication architecture.

Figure 2 shows the flexible quasi out of band authentication architecture with the key management functionality layered on top.

PREFERRED EMBODIMENT(S) OF THE INVENTION

General Overview Of Prior Work Related To QOOBA

In prior work we have described how the introduction of a network based security server, which has an independent channel to a window displayed on a user device, can be used in conjunction with a user's browser, computer software or smart phone or other mobile communications device application, and the website they are visiting to provide user authentication for login or transaction authorization via one or more user network device.

QOOBA is an innovative approach to create a solution that can be used to authenticate every transaction in a manner that will feel natural to users. The central idea is to create a small secure window, the QOOBA Window, that has an independent encrypted channel to a secure server (the QOOBA Server). The QOOBA Window can be implemented as a zero-download browser pop-up (the QOOBA Pop-up Window), as a small desktop application (the QOOBA Software Window) or as an app on a smart phone (the QOOBA Phone Window). An important aspect to the innovation is that, unlike as with soft tokens, the QOOBA Window does not require secure storage of long term secrets. Rather, it is "activated" during login by using out of band authentication. As a user transacts at a web site, (e.g. a merchant or bank website, that is part of the QOOBA Network, the transactions that the web site believes the user intends are sent securely via the user's browser to the QOOBA Server which displays the transaction in the QOOBA Window. The user is also optionally shown a transaction signature that the user can cut and paste from the QOOBA Window into their browser to authenticate to the website for login or transaction authorization purposes.

Overview Of Prior Work Relating To A Flexible QOOBA Architecture

The QOOBA solution has the following benefits in terms of ease of use, total cost of ownership and, of particular interest here, security.

Overlaying Key Management On The QOOBA Architecture

We now describe how we overlay components for key management on the QOOBA architecture.

The QOOBA System

Referring now to Figure 1, in accordance with the present example, the QOOBA system consists of a desktop personal computing device 100 having the QOOBA Window 110 and a Browser Window 112 executing and displayed thereon, a QOOBA Server 125 and Web Service 150, which has the QOOBA Application Programming Interface (API) 155 operable thereon. It should be understood that in a practical implementation there would typically be multiple websites. Also included in the system as shown is an OOBA Service 165, which is utilized by the QOOBA Server 125 to bootstrap authentication of the user using the user's phone 175, which may be a landline, cellphone or smartphone.

As described in more detail in the related applications referenced above, the user activates the QOOBA Window 110, typically by using out of band authentication via OOBA Service 165, and establishes a session with the QOOBA Server 125. Web Service 150 participates in the QOOBA Network and goes through a onetime set up process to establish a shared secret with the QOOBA Server 125, which is not shared with or known by the user. When the user has an active session with the QOOBA Server 125 via communication channel 450 and is also at the Website 150 via communication channel 400, the Website can use the QOOBA API 155 to request, via back end communication channel 500, transaction authentication by sending the transaction directly to the QOOBA Server 125. The QOOBA Server 125 then displays the transaction to the user in the applicable QOOBA Window, which is shown in Figure 1 to be Window 110.

The QOOBA Windows

The QOOBA Server 125 can present various information to the user in the displayed QOOBA Window 110. For example, as we have described in our prior work, the QOOBA Server 125 can display a transaction to the user in the QOOBA Window 110, and if requested, also display in the QOOBA Window 110 a transaction, i.e. electronic, signature derived from the transaction, the secret shared between the QOOBA Server 125 and the