

BRIEF DESCRIPTION OF DRAWINGS

Each of these drawings represents an example embodiment of the systems they are depicting.

FIG 1. Illustrates the main architectural components.

5 FIG 2. Illustrates an example login screen.

FIG 3. Illustrates the communication between the user (both browser and mobile phone), the authentication server, and the network provider.

FIG 4. Illustrates the primary subsystems on the mobile user device.

FIG 5. Illustrates the primary subsystems on the authentication server.

10 FIG 6. Illustrates the split key system and the high-level data flow.

FIG 7. Illustrates the sample interactions between the use, the authentication server, and the provider.

FIG 8. Illustrates the data control areas for various pieces of data within the system.

15 FIG 9. Illustrates the communication between the user (both browser and mobile phone), the authentication server, and the network provider.

FIG 10. Illustrates a sample paper based enrollment process.

FIG 11. Illustrates a sample communication flow for the notification system.

DETAILED DESCRIPTION

OVERVIEW

First Authentication Framework

According to aspects of the invention provides an authentication framework, see FIG 1, that includes a mobile device of the user which has a network connection, e.g. a smart phone 400 or tablet, and a software application capable of executing various cryptographic and authentication operations101. A network service 103, like a web site or terminal server, is also included. Access software is used to reach the network service, like a web browser 102, that may reside on the mobile device or on a secondary device like a desktop or laptop. The access software may have a trusted component which can reliably send security data to the user's mobile device including information to validate that credentials being provided by the user are for the current network service. The trusted component of the access software may be a browser plug-in. Additional security information, provided

Fourth Authentication Framework

According to aspects of the invention, an authentication framework includes a mobile device of the user which includes a network connection and a software application capable of executing various cryptographic and authentication operations. A network service, like a web site or terminal server, is also included. Access software is used to reach the network service, like a web browser, that may reside on the mobile device or on a secondary device like a desktop or laptop. An authentication server attached to a network and reachable by the network service, the access software and the mobile device. If desired, the authentication server can include a fraud and anomaly detection system. See FIG. 9. An identifier 902 is supplied to the browser and uniquely identifies the mobile device's software application. The identifier could be a unique value per user like a phone number or user name.

A first connection between the mobile device software application and the authentication server performs the transfer of security checks for liveness of the session, and credentials of the user and the network service. The credentials of the user may be based on asymmetric public key cryptography using a public/private key system, and the credentials could be specific to the network service being accessed. The security checks may further include additional authentication factors including: biometric data, token based authentication data, or additional knowledge based factors. The biometric data could be collected from devices attached to the user mobile device

A second connection to the access software allows the access software to know when the authentication has been completed. The second communication channel may be between the access software and the authentication service, or between the access software and the network service.

Fifth Authentication Framework

This authentication framework includes a mobile device of the user which has a network connection and a software application capable of executing various cryptographic and authentication operations. A network service, like a web site or terminal server, is also included. Access software is

a QR code, processing the communication may include reading the QR code with a camera and decoding the QR code.

A first connection between the mobile device software application and the authentication server that performs the transfer of security checks and authentication of the user. A second connection to the network service allows the network service to know when the enrollment has been completed.

DESCRIPTION OF EMBODIMENTS

Multiparty Authentication System

A multi-party authentication system which uses an essentially un-trusted authentication provider 100 to validate users to network service providers 103. The system involves a trusted user device, like a mobile tablet or smart phone 400, that secures and holds the various user credentials (for example public/private keys and certificates); and can collect a variable number and type of authentication credentials including biometric, knowledge based (i.e. passwords), and token based. The authentication service 100 provides user driven provisioning and controls allowing real-time scalable authentication across a WAN and allows lost key recovery and simplified enrollment or re-enrollment.

Reference is now made in detail to the description of the embodiments as illustrated in the drawings. While embodiments are described in connection with the drawings and related descriptions, there is no intent to limit the scope to the embodiments disclosed herein. On the contrary, the intent is to cover all alternatives, modifications and equivalents. Those of ordinary skill in the art will appreciate that other embodiments, including additional devices, or combinations of illustrated devices, may be added to, or combined, without limiting the scope to the embodiments disclosed herein.

The present invention is based on a multi-party system (the user, the authentication service provider 100, and the network service provider 103) and includes the following main architectural components (see FIGS 1 through 5 for views of the architectural components):

- A user mobile device 400 which includes a network connection 402, data input system 401, and a software authentication application or app (Qapp) 101 which allows various authentication functions to be done.

browser storage location, or iii) from a communication means with the user's mobile device (attached USB, nearfield, etc.) where the user's mobile device then provides the unique user ID.

- An authentication server 100 providing a variety of services (Qserver).

5 The Qserver acts as an intermediary PKI management portal ensuring that various user and Provider credentials are correctly setup and used including 501 and 505; it handles additional forms of authentication 502 including biometric and token based options; it allows users to manage their own authentication including reporting lost phones or upgrading authentication through a policy mediation system 503; optionally, it
10 manages user identities; and it does fraud detection 504 on authentication requests and authentication data used. In one embodiment the authentication server is a single Internet reachable host 500 or cluster of servers allowing network connected devices to connect, via communications channels 104, 105, and 107 and
15 managed by 506, that requests or sends a variety of information. In a second embodiment the authentication server may consist of a hierarchy of servers allowing decision on authentication to flow between servers. In this embodiment the lower level server may provide authentication services to an organizational or corporate
20 network behind firewalls or other network and security boundaries. In one embodiment the Qserver is housed on a server separate from the Provider to limit the security risks of losing authentication data.

Because the roles of each party spread the trust throughout the
25 system, if one party is compromised risks to the entire system can be limited. See FIG 7. For example one strength of the present invention is that if the user loses their mobile device the credentials can easily be revoked 705, credentials may also require the user to enter secret data (like a password) to access – which can be audited, and secure sites will require biometric data to
30 access. In addition, none of these controls requires Provider intervention or modifications. Another security benefit is that during phishing attacks the user authentication data is never transferred to the Provider (or pretend Provider) so no credentials can be stolen. Finally, if the Qserver is compromised;

areas 811, and 812. For example for the first user/provider pair information flows from the user area 714 through channel 807 to the first provider area 811 over channel 809. The second user/provider pair sends different data over a similar set of channels 808 and 810. The providers each maintain their own set of account data 802, and 804 including the authentication configuration options 803 and 805.

In addition to the architectural pieces there are a set of communications that happen between the different parties. One common interaction with the system is when a user logs into a web site. The following is a high-level view of one embodiment for login.

1. The user goes to a web site that implements the present invention's login system. FIG 2. shows a screen shot of what the login might look like. In addition to the normal account login selection a QR code 200 with the Qcode is displayed. The Qcode includes a header block (described below) and a Session Id (Qsid) which is guaranteed to be unique across all users for as long as the code is valid and acts as a simple identifier for the authentication session.

2. Referring now to Figure 3, the user, on their smart phone, starts up the Qapp and scans the Qcode 302. This starts a communication 303 with the Authentication Server (Qserver) 100 and, depending on the policies of the user and the Provider which are provided in 304 the user will enter additional authentication information 305 that may include a pin, token data, and/or biometric data. For the purposes of a higher security site, for example PC Banking, the user might go through the following. FIG 3 shows an example login.

- i. They will see a message saying "Would you like to login to Provider X". This will allow the user to validate that who they are trying to login to is the same location that they are connecting to in 102.
 - ii. The user will approve the login and then enter their secret data and if the site 103 requests three-factor authentication they may also need to submit some form of biometric data (such as a picture of their face, hand image, or voice sample).

iii. The session information, session validation, approval, and the biometric data is then submitted in 306 to the Qserver server 100 for validation. Behind the scenes if the Qserver approves the authentication in 307 and then it will notify the Provider (the site can then do an independent validation of the user's challenge/response) and the users browser gets automatically refreshed and logged in.

- 5
3. The user is now logged in – besides scanning the Qcode in 302 they do not need to enter any information or even click on any links on the web site. This makes a virtually transparent, completely automated, and yet secure authentication system.
- 10

The following is a high-level view of another embodiment for login. See FIG 9.

1. The user goes 901 to a web site that implements the present invention's login system. Instead of the traditional username and password fields the web site has a single field for the user to enter their unique user ID 902. The unique ID could be assigned like a username or be something like a phone number. The user then clicks submit and the web site with collaboration from the Qserver assigns a unique session id (Qsid) 900 and starts authentication. In a second embodiment the unique ID might be saved in a cookie or retrieved from the mobile device during access. It is also possible to tie the browser and mobile device together by transmitting from the browser the Qsid or other identifier specific to the session. The transmission could be in the form of a QR code displayed to the user and scanned via the mobile device.
- 15
2. The provider communicates the unique user ID and the Qsid to the Authentication Server 903.
- 20
3. Based on the unique user ID the authentication server contacts the user's mobile device 904. This communication with the Authentication Server (Qserver) which will include the requirements for authentication 905. Depending on the policies of the user and the Provider the user will enter additional authentication information that may include a pin, token data, and/or biometric data 906. FIG 9., which shows an example
- 25
- 30

login as in FIG. 3 described above, also shows the completion steps 907 and 908 to finish the login.

4. The user is now logged in – besides providing their unique user ID, they do not need to enter any information or even click on any links on the web site. This makes a virtually transparent, completely automated, and yet secure authentication system.

Specific Security and Architectural Issues

The Qcode includes the following information: a header tag which specifies that this is an authentication token for the present invention, and a Qsid which is long enough to make guessing virtually impossible, guarantee uniqueness within the environment, and long enough to make reuse of the Qsid happen infrequently. In one embodiment, the Qsid is a random number at least 128- bits long. Optionally, the Qcode may include session specific information like the type of login/enrollment requested, or alternate authentication server.

In addition to the Qsid, there is a liveness ID (Qliveness) which is another piece of information passed from the Provider, through the Qserver to the Qapp. In one embodiment, the Qliveness is a random number generated by the Provider. The Qliveness provides two main security functions 1) makes it harder to reuse Qsid as an attacker would need to know both the Qsid and the Qliveness numbers, and 2) makes it impossible for the Qserver to replay attacks since the Provider can choose the Qliveness. In one embodiment, it is a configuration option if the Provider wishes to create the Qliveness random number or trust the Qserver.

In a second embodiment the user's access software (i.e. browser for web services) also has a plug-in or other technique for validating the user is at the Provider (i.e. web-site) that registered the Qsid. This plug-in could be used to exchange information securely between the Qapp and the user's browser. This might include validation of the URL, session keys, or other information for signing and/or securing communication. For a man-in-the-middle attack where a user has gone to the wrong site (from a spam link, mis-typed the URL, etc) and believe they are at the Provider's web site but instead are at a site run by an attacker, the only assured method to alert the user to their mistake is to

validate at the browser the User's intention with the Qserver. This can be done as a browser plug-in or stand alone program on the desktop.

In different embodiments, some or all of the user credentials stored on the smart phone are stored in an encrypted key-store which includes the private keys and certificates used to communicate with Providers or potentially other users. User credentials could include any information or token to validate a user including public/private key pairs, passwords, biometric templates, one time password seeds, etc.

To decrypt the keystore one embodiment uses a unique split key system to prevent brute force guessing of the decryption key 601 and 606 if the keystore is lost, and prevent the Qserver from having access to the keystore. See FIG 6. The present invention includes the following split key system: where a user enters secret data is split into multiple portions, where one embodiment is to use 2 portions: A and B. The secret data can be any type of user data including: password, pin, biometric template, an image drawn by the user, etc. The A portion of the password is used locally to create the A portion of the decryption key 605. In various embodiments the A portion might be used to decrypt another block on the local system, it might be combined with system information like the smart card's unique serial number, or it might be hashed or otherwise modified to make the A portion longer or more obscure to decode. Since the A portion never leaves the phone the user maintains control over the decryption key. The B portion of the secret data 602 is sent to the Qserver over an encrypted and authenticated channel (using the user's communication key), and the Qserver sends back the B portion of the decryption key 604. In different embodiments the B portion of the key might be looked up from a database, created as a hash from the B portion of the secret data, or combined other user/hardware specific information. Since the Qserver can monitor the use of the B portion it is possible to lock the account after a set number of failed attempts, alert the user, or otherwise respond to suspicious behavior 603.

In one embodiment, the user credentials are based on a public/private key and certificate system (PKI). The benefits of this system are that the user's credentials, the private key, never leave the smart-phone and the

certificates can be signed and validated into the system at enrollment. These certificates can be handed out and controlled individually, allowing users to authenticate with separate credentials to each Provider. In addition other pairs of credentials for example user-to-user credentials can also be created and maintained separately. One embodiment uses three different types of certificates which are signed by three different signing authorities: 1) Provider Certificate Authority (CA) – signs provider certificates for use in communicating between the Provider and the Qserver, 2) User CA – signs the user communication certificates and the user-to-provider certificates created by the user for use in validating their credentials with a Provider. And 3) Qserver CA – signs the certificates used on the Qservers to validate to the user and provider that they are talking to a legitimate Qserver. In a second embodiment, the user might have a single user-to-all-providers certificate, and the signing authorities could be consolidated into a single CA or hierarchical CA configuration.

In the present invention it is possible for the user to use multiple forms of authentication data including biometrics. The current state-of-the-art in authentication systems combine some form of knowledge, biometric, and token factors; provisioning for just one of these factors is complex without the present invention. Because smart phones and other types of mobile devices have multiple sensors and other data input methods it is possible to collect data from all traditional authentication types:

- Something you are: includes biometric options like face recognition taken from a front facing camera, hand recognition taken from a rear facing camera, speaker recognition taking from the microphone, etc. Some phones have specialized inputs build in like fingerprint readers, and devices can often be attached to the smart phone allowing additional capabilities.
- Something you know: includes passwords, security questions, the phrase spoken during speaker recognition, an image drawn on the screen, a phone shaking pattern, etc.
- Something you have: the most obvious is the phone or mobile device itself. But this category can also include tokens associated with the

phone for example: attached usb devices, tokens discovered via near-field radio options, card data read off of card readers, data from the SIM card of the device, secure data stored on the device itself in a secure co-processor or data lockbox, etc.

5 The present invention allows the Provider and user to use one or more types of authentication giving flexibility and additional security where needed on a individualized basis. The Qserver manages additional factor security including: enrollment of various biometric or token options, storage of security data like biometric templates, fraud detection targeted at the different forms of authentication, and comparing login authentication to the enrolled data. The Qapp manages collecting the authentication data and may do a set of pre-processing steps on the data before submission to the Qserver. The pre-processing is primarily directed at ensuring the quality of the submission to give the user feedback quickly if the image is out-of-focus, not received etc.

10 but the pre-processing is also used to limit the size of data sent and potentially clean up submitted data (align the head to the center of the image, etc.). The Qapp may also have a set of fraud detection functions which are done in addition to the main fraud detection done on the Qserver. Even though the Qserver maintains the additional authentication factors, since the Qserver

15 does not have access to the keystore 406 the Qserver can not pretend to be a user. This separation of trust allows centralized control and management without compromising the user's control over their own authentication. Another benefit of this architecture is new forms of authentication can be rolled out to users without the need for the Provider to change anything.

25 The architecture of the present invention also allows simplified provisioning when Providers want to add new users. There is no need for the Provider to pre-configure the authentication data, for example there is no need to assign temporary passwords. This also allows the Provider to have greater flexibility in the types of authentication policies they want to support. For

30 example if the Provider wants to upgrade the authentication to require 2-factor, they simply change the global policy on the server and users will start being required to authenticate with 2-factors. The Provider does not need to go back to each user trying to collect enrollment data for the new factor that is

all handled by the Authentication Server. When combined with simplified identity and enrollment, there is no need for Providers to pre-configure any account on their systems.

5 In one embodiment, it would be possible to replace a traditional user with an automated process, like batch system that need to authenticated to a variety of services across a network. Although the system would work the same certain optional components like biometric authentication or knowledge based authentication factors would not be possible.

10 The present invention also allows a secure form of communication between the Provider and the User via the secure channel through the Authentication Server and the Qapp on the user's mobile device. This secure communication channel would allow Provider's to send critical notifications in a trusted manner allowing information like "you have made a purchase" or "you have transferred money" to be transmitted in a manner that can not be
15 spoofed or modified. The encryption keys for the data could be based on any standard including the public certificates previously exchanged between the parties or previously shared session keys. This type of communication would also allow secure communication between any two users on the network using a pre-established user-to-user credential.

20 Unidirectional requests like "you have made a purchase" would not require a response from the user, but bidirectional requests like "are you sure you would like to purchase X" would allows the Provider to wait for a response from the user. Thus allowing the user to validate transactions through the secure communication channel before they are finalized. The verification
25 process could occur even if the user is not currently logged into the Provider and could involve the user "approving" the transaction while simultaneously providing additional authentication data.

Detailed Sample Embodiment of the Login Process-See FIG 3

- 30
- User goes to the web site via a desktop or mobile browser 301.
 - Web Site 103 (Provider) gets, over channel 104, a Session Id (Qsid) 300 from the Authentication Server 100 and locally creates a session-specific random number (Qliveness) 300. The Provider could pre-cache for performance reasons a set of Qsids from the Qserver. The

CLAIMS

1. A method of operating an authentication server capable of authenticating any of multiple different users to multiple different service providers via a network, comprising:

- storing validation information for the first user;
- storing authentication policy requirements of a first of the multiple service providers, in association with an identifier of the first provider;
- transmitting a first random number to the first provider via the network;
- receiving, from the first provider via the network, first other information;
- receiving, from a requesting user via the network, the first random number and a request of that user to be authenticated;
- transmitting, after receipt of the first random number and authentication request, the stored first provider identifier, the received first other information, and the stored first provider authentication policy requirements, to the requesting user via the network;
- receiving, in response to the transmitted first provider authentication policy requirements, a user identifier and validation information, from the requesting user via the network;
- matching the received requesting user identifier to the stored first user identifier;
- determining that the validation information received from the requesting user corresponds to the stored first provider authentication policy requirements;
- comparing the validation information received from the requesting user with the stored first user validation information to authenticate that the requesting user is the first user;
- receiving, from the requesting user via the network, a message, including the first random number and the first other information, signed with a credential; and
- transmitting, to the first provider via the network, notice of authentication of the first user and the received signed message.

2. The method of claim 1, wherein:

BRIEF DESCRIPTION OF DRAWINGS

Each of these drawings represents an example embodiment of the systems they are depicting.

FIG 1. Illustrates the main architectural components.

5 FIG 2. Illustrates an example login screen.

FIG 3. Illustrates the communication between the user (both browser and mobile phone), the authentication server, and the network provider.

FIG 4. Illustrates the primary subsystems on the mobile user device.

FIG 5. Illustrates the primary subsystems on the authentication server.

10 FIG 6. Illustrates the split key system and the high-level data flow.

FIG 7. Illustrates the sample interactions between the use, the authentication server, and the provider.

FIG 8. Illustrates the data control areas for various pieces of data within the system.

15 FIG 9. Illustrates the communication between the user (both browser and mobile phone), the authentication server, and the network provider.

FIG 10. Illustrates a sample paper based enrollment process.

FIG 11. Illustrates a sample communication flow for the notification system.

DETAILED DESCRIPTION

OVERVIEW

First Authentication Framework

An exemplary authentication framework, see FIG 1, may include a mobile device of the user which has a network connection, e.g. a smart phone 400 or tablet, and a software application capable of executing various
 25 cryptographic and authentication operations¹⁰¹. A network service 103, like a web site or terminal server, is also included. Access software is used to reach the network service, like a web browser 102 that may reside on the mobile device or on a secondary device like a desktop or laptop. The access software may have a trusted component which can reliably send security data
 30 to the user's mobile device including information to validate that credentials being provided by the user are for the current network service. The trusted component of the access software may be a browser plug-in. Additional security information, provided

Fourth Authentication Framework

An exemplary authentication framework may include a mobile device of the user which includes a network connection and a software application capable of executing various cryptographic and authentication operations. A
5 network service, like a web site or terminal server, is also included. Access software is used to reach the network service, like a web browser, that may reside on the mobile device or on a secondary device like a desktop or laptop. An authentication server attached to a network and reachable by the network service, the access software and the mobile device. If desired, the
10 authentication server can include a fraud and anomaly detection system. See FIG. 9. An identifier 902 is supplied to the browser and uniquely identifies the mobile device's software application. The identifier could be a unique value per user like a phone number or user name.

A first connection between the mobile device software application and
15 the authentication server performs the transfer of security checks for liveness of the session, and credentials of the user and the network service. The credentials of the user may be based on asymmetric public key cryptography using a public/private key system, and the credentials could be specific to the network service being accessed. The security checks may further include
20 additional authentication factors including: biometric data, token based authentication data, or additional knowledge based factors. The biometric data could be collected from devices attached to the user mobile device

A second connection to the access software allows the access software to know when the authentication has been completed. The second
25 communication channel may be between the access software and the authentication service, or between the access software and the network service.

Fifth Authentication Framework

This authentication framework includes a mobile device of the user
30 which has a network connection and a software application capable of executing various cryptographic and authentication operations. A network service, like a web site or terminal server, is also included. Access software is

a QR code, processing the communication may include reading the QR code with a camera and decoding the QR code.

A first connection between the mobile device software application and the authentication server that performs the transfer of security checks and authentication of the user. A second connection to the network service allows the network service to know when the enrollment has been completed.

DESCRIPTION OF EMBODIMENTS

Multiparty Authentication System

A multi-party authentication system which uses an essentially un-trusted authentication provider 100 to validate users to network service providers 103. The system involves a trusted user device, like a mobile tablet or smart phone 400, that secures and holds the various user credentials (for example public/private keys and certificates); and can collect a variable number and type of authentication credentials including biometric, knowledge based (i.e. passwords), and token based. The authentication service 100 provides user driven provisioning and controls allowing real-time scalable authentication across a WAN and allows lost key recovery and simplified enrollment or re-enrollment.

Reference is now made in detail to the description of the embodiments as illustrated in the drawings. While embodiments are described in connection with the drawings and related descriptions, there is no intent to limit the scope to the embodiments disclosed herein. On the contrary, the intent is to cover all alternatives, modifications and equivalents. Those of ordinary skill in the art will appreciate that other embodiments, including additional devices, or combinations of illustrated devices, may be added to, or combined, without limiting the scope to the embodiments disclosed herein.

The techniques disclosed herein may be based on a multi-party system (the user, the authentication service provider 100, and the network service provider 103) and includes the following main architectural components (see FIGS 1 through 5 for views of the architectural components):

- A user mobile device 400 which includes a network connection 402, data input system 401, and a software authentication application or app (Qapp) 101 which allows various authentication functions to be done.

- browser storage location, or iii) from a communication means with the user's mobile device (attached USB, nearfield, etc.) where the user's mobile device then provides the unique user ID.
- An authentication server 100 providing a variety of services (Qserver).
 5 The Qserver acts as an intermediary PKI management portal ensuring that various user and Provider credentials are correctly setup and used including 501 and 505; it handles additional forms of authentication 502 including biometric and token based options; it allows users to manage their own authentication including reporting lost phones or upgrading
 10 authentication through a policy mediation system 503; optionally, it manages user identities; and it does fraud detection 504 on authentication requests and authentication data used. In one embodiment the authentication server is a single Internet reachable host 500 or cluster of servers allowing network connected devices to
 15 connect, via communications channels 104, 105, and 107 and managed by 506, that requests or sends a variety of information. In a second embodiment the authentication server may consist of a hierarchy of servers allowing decision on authentication to flow between servers. In this embodiment the lower level server may
 20 provide authentication services to an organizational or corporate network behind firewalls or other network and security boundaries. In one embodiment the Qserver is housed on a server separate from the Provider to limit the security risks of losing authentication data.

Because the roles of each party spread the trust throughout the
 25 system, if one party is compromised risks to the entire system can be limited. See FIG 7. For example one strength of the techniques disclosed herein is that if the user losses their mobile device the credentials can easily be revoked 705, credentials may also require the user to enter secret data (like a password) to access – which can be audited, and secure sites will require
 30 biometric data to access. In addition, none of these controls requires Provider intervention or modifications. Another security benefit is that during phishing attacks the user authentication data is never transferred to the Provider (or pretend Provider) so no credentials can be stolen. Finally, if the Qserver is compromised;

areas 811, and 812. For example for the first user/provider pair information flows from the user area 714 through channel 807 to the first provider area 811 over channel 809. The second user/provider pair sends different data over a similar set of channels 808 and 810. The providers each maintain their own set of account data 802, and 804 including the authentication configuration options 803 and 805.

In addition to the architectural pieces there are a set of communications that happen between the different parties. One common interaction with the system is when a user logs into a web site. The following is a high-level view of one embodiment for login.

1. The user goes to a web site that implements the login system. FIG 2. shows a screen shot of what the login might look like. In addition to the normal account login selection a QR code 200 with the Qcode is displayed. The Qcode includes a header block (described below) and a Session Id (Qsid) which is guaranteed to be unique across all users for as long as the code is valid and acts as a simple identifier for the authentication session.
2. Referring now to Figure 3, the user, on their smart phone, starts up the Qapp and scans the Qcode 302. This starts a communication 303 with the Authentication Server (Qserver) 100 and, depending on the policies of the user and the Provider which are provided in 304 the user will enter additional authentication information 305 that may include a pin, token data, and/or biometric data. For the purposes of a higher security site, for example PC Banking, the user might go through the following. FIG 3 shows an example login.
 - i. They will see a message saying "Would you like to login to Provider X". This will allow the user to validate that who they are trying to login to is the same location that they are connecting to in 102.
 - ii. The user will approve the login and then enter their secret data and if the site 103 requests three-factor authentication they may also need to submit some form of biometric data (such as a picture of their face, hand image, or voice sample).

- iii. The session information, session validation, approval, and the biometric data is then submitted in 306 to the Qserver server 100 for validation. Behind the scenes if the Qserver approves the authentication in 307 and then it will notify the Provider (the site can then do an independent validation of the user's challenge/response) and the users browser gets automatically refreshed and logged in.
3. The user is now logged in – besides scanning the Qcode in 302 they do not need to enter any information or even click on any links on the web site. This makes a virtually transparent, completely automated, and yet secure authentication system.

The following is a high-level view of another embodiment for login. See FIG 9.

1. The user goes 901 to a web site that implements the login system. Instead of the traditional username and password fields the web site has a single field for the user to enter their unique user ID 902. The unique ID could be assigned like a username or be something like a phone number. The user then clicks submit and the web site with collaboration from the Qserver assigns a unique session id (Qsid) 900 and starts authentication. In a second embodiment the unique ID might be saved in a cookie or retrieved from the mobile device during access. It is also possible to tie the browser and mobile device together by transmitting from the browser the Qsid or other identifier specific to the session. The transmission could be in the form of a QR code displayed to the user and scanned via the mobile device.
2. The provider communicates the unique user ID and the Qsid to the Authentication Server 903.
3. Based on the unique user ID the authentication server contacts the user's mobile device 904. This communication with the Authentication Server (Qserver) which will include the requirements for authentication 905. Depending on the policies of the user and the Provider the user will enter additional authentication information that may include a pin, token data, and/or biometric data 906. FIG 9., which shows an example

login as in FIG. 3 described above, also shows the completion steps 907 and 908 to finish the login.

4. The user is now logged in – besides providing their unique user ID, they do not need to enter any information or even click on any links on the web site. This makes a virtually transparent, completely automated, and yet secure authentication system.

Specific Security and Architectural Issues

The Qcode includes the following information: a header tag which specifies that this is an authentication token, and a Qsid which is long enough to make guessing virtually impossible, guarantee uniqueness within the environment, and long enough to make reuse of the Qsid happen infrequently. In one embodiment, the Qsid is a random number at least 128- bits long. Optionally, the Qcode may include session specific information like the type of login/enrollment requested, or alternate authentication server.

In addition to the Qsid, there is a liveness ID (Qliveness) which is another piece of information passed from the Provider, through the Qserver to the Qapp. In one embodiment, the Qliveness is a random number generated by the Provider. The Qliveness provides two main security functions 1) makes it harder to reuse Qsid as an attacker would need to know both the Qsid and the Qliveness numbers, and 2) makes it impossible for the Qserver to replay attacks since the Provider can choose the Qliveness. In one embodiment, it is a configuration option if the Provider wishes to create the Qliveness random number or trust the Qserver.

In a second embodiment the user's access software (i.e. browser for web services) also has a plug-in or other technique for validating the user is at the Provider (i.e. web-site) that registered the Qsid. This plug-in could be used to exchange information securely between the Qapp and the user's browser. This might include validation of the URL, session keys, or other information for signing and/or securing communication. For a man-in-the-middle attack where a user has gone to the wrong site (from a spam link, mis-typed the URL, etc) and believe they are at the Provider's web site but instead are at a site run by an attacker, the only assured method to alert the user to their mistake is to

validate at the browser the User's intention with the Qserver. This can be done as a browser plug-in or stand alone program on the desktop.

In different embodiments, some or all of the user credentials stored on the smart phone are stored in an encrypted key-store which includes the
5 private keys and certificates used to communicate with Providers or potentially other users. User credentials could include any information or token to validate a user including public/private key pairs, passwords, biometric templates, one time password seeds, etc.

To decrypt the keystore one embodiment uses a unique split key
10 system to prevent brute force guessing of the decryption key 601 and 606 if the keystore is lost, and prevent the Qserver from having access to the keystore. See FIG 6. The exemplary methods disclosed herein may implement the following split key system: where a user enters secret data is split into multiple portions, where one embodiment is to use 2 portions: A and
15 B. The secret data can be any type of user data including: password, pin, biometric template, an image drawn by the user, etc. The A portion of the password is used locally to create the A portion of the decryption key 605. In various embodiments the A portion might be used to decrypt another block on the local system, it might be combined with system information like the smart
20 card's unique serial number, or it might be hashed or otherwise modified to make the A portion longer or more obscure to decode. Since the A portion never leaves the phone the user maintains control over the decryption key. The B portion of the secret data 602 is sent to the Qserver over an encrypted and authenticated channel (using the user's communication key), and the
25 Qserver sends back the B portion of the decryption key 604. In different embodiments the B portion of the key might be looked up from a database, created as a hash from the B portion of the secret data, or combined other user/hardware specific information. Since the Qserver can monitor the use of the B portion it is possible to lock the account after a set number of failed
30 attempts, alert the user, or otherwise respond to suspicious behavior 603.

In one embodiment, the user credentials are based on a public/private key and certificate system (PKI). The benefits of this system are that the user's credentials, the private key, never leave the smart-phone and the

certificates can be signed and validated into the system at enrollment. These certificates can be handed out and controlled individually, allowing users to authenticate with separate credentials to each Provider. In addition other pairs of credentials for example user-to-user credentials can also be created and

5 maintained separately. One embodiment uses three different types of certificates which are signed by three different signing authorities: 1) Provider Certificate Authority (CA) – signs provider certificates for use in communicating between the Provider and the Qserver, 2) User CA – signs the user communication certificates and the user-to-provider certificates created

10 by the user for use in validating their credentials with a Provider. And 3) Qserver CA – signs the certificates used on the Qservers to validate to the user and provider that they are talking to a legitimate Qserver. In a second embodiment, the user might have a single user-to-all-providers certificate, and the signing authorities could be consolidated into a single CA or hierarchical

15 CA configuration.

In the exemplary techniques disclosed herein it is possible for the user to use multiple forms of authentication data including biometrics. The current state-of-the-art in authentication systems combine some form of knowledge, biometric, and token factors; provisioning for just one of these factors is

20 complex without the techniques disclosed herein. Because smart phones and other types of mobile devices have multiple sensors and other data input methods it is possible to collect data from all traditional authentication types:

- Something you are: includes biometric options like face recognition taken from a front facing camera, hand recognition taken from a rear facing camera, speaker recognition taking from the microphone, etc.
- 25 Some phones have specialized inputs build in like fingerprint readers, and devices can often be attached to the smart phone allowing additional capabilities.
- Something you know: includes passwords, security questions, the phrase spoken during speaker recognition, an image drawn on the
- 30 screen, a phone shaking pattern, etc.
- Something you have: the most obvious is the phone or mobile device itself. But this category can also include tokens associated with the

phone for example: attached usb devices, tokens discovered via near-field radio options, card data read off of card readers, data from the SIM card of the device, secure data stored on the device itself in a secure co-processor or data lockbox, etc.

5 The exemplary techniques disclosed herein may allow the Provider and user to use one or more types of authentication giving flexibility and additional security where needed on a individualized basis. The Qserver manages additional factor security including: enrollment of various biometric or token options, storage of security data like biometric templates, fraud detection
10 targeted at the different forms of authentication, and comparing login authentication to the enrolled data. The Qapp manages collecting the authentication data and may do a set of pre-processing steps on the data before submission to the Qserver. The pre-processing is primarily directed at ensuring the quality of the submission to give the user feedback quickly if the
15 image is out-of-focus, not received etc. but the pre-processing is also used to limit the size of data sent and potentially clean up submitted data (align the head to the center of the image, etc.). The Qapp may also have a set of fraud detection functions which are done in addition to the main fraud detection done on the Qserver. Even though the Qserver maintains the additional
20 authentication factors, since the Qserver does not have access to the keystore 406 the Qserver can not pretend to be a user. This separation of trust allows centralized control and management without compromising the user's control over their own authentication. Another benefit of this architecture is new forms of authentication can be rolled out to users without
25 the need for the Provider to change anything.

 The architecture for the exemplary techniques disclosed herein may also allow simplified provisioning when Providers want to add new users. There is no need for the Provider to pre-configure the authentication data, for example there is no need to assign temporary passwords. This also allows the
30 Provider to have greater flexibility in the types of authentication policies they want to support. For example if the Provider wants to upgrade the authentication to require 2-factor, they simply change the global policy on the server and users will start being required to authenticate with 2-factors. The Provider does not need to go back to each user trying to collect enrollment
35 data for the new factor that is

all handled by the Authentication Server. When combined with simplified identity and enrollment, there is no need for Providers to pre-configure any account on their systems.

5 In one embodiment, it would be possible to replace a traditional user with an automated process, like batch system that need to authenticated to a variety of services across a network. Although the system would work the same certain optional components like biometric authentication or knowledge based authentication factors would not be possible.

10 The exemplary techniques disclosed herein may also allow a secure form of communication between the Provider and the User via the secure channel through the Authentication Server and the Qapp on the user's mobile device. This secure communication channel would allow Provider's to send critical notifications in a trusted manner allowing information like "you have
15 made a purchase" or "you have transferred money" to be transmitted in a manner that can not be spoofed or modified. The encryption keys for the data could be based on any standard including the public certificates previously exchanged between the parties or previously shared session keys. This type of communication would also allow secure communication between any two
20 users on the network using a pre-established user-to-user credential.

 Unidirectional requests like "you have made a purchase" would not require a response from the user, but bidirectional requests like "are you sure you would like to purchase X" would allows the Provider to wait for a response from the user. Thus allowing the user to validate transactions through the
25 secure communication channel before they are finalized. The verification process could occur even if the user is not currently logged into the Provider and could involve the user "approving" the transaction while simultaneously providing additional authentication data.

Detailed Sample Embodiment of the Login Process-See FIG 3

- 30 • User goes to the web site via a desktop or mobile browser 301.
- Web Site 103 (Provider) gets, over channel 104, a Session Id (Qsid) 300 from the Authentication Server 100 and locally creates a session-specific random number (Qliveness) 300. The Provider could pre-cache for performance reasons a set of Qsids from the Qserver. The

CLAIMS

1. A method of operating an authentication server capable of authenticating any of multiple different users to multiple different service providers via a network, comprising:

- storing validation information for a first user;
- storing authentication policy requirements of a first of the multiple service providers, in association with an identifier of the first provider;
- transmitting a first random number to the first provider via the network;
- receiving, from the first provider via the network, first other information;
- receiving, from a requesting user via the network, the first random number and a request of that user to be authenticated;
- transmitting, after receipt of the first random number and authentication request, the stored first provider identifier, the received first other information, and the stored first provider authentication policy requirements, to the requesting user via the network;
- receiving, in response to the transmitted first provider authentication policy requirements, a user identifier and validation information, from the requesting user via the network;
- matching the received requesting user identifier to the stored first user identifier;
- determining that the validation information received from the requesting user corresponds to the stored first provider authentication policy requirements;
- comparing the validation information received from the requesting user with the stored first user validation information to authenticate that the requesting user is the first user;
- receiving, from the requesting user via the network, a message, including the first random number and the first other information, signed with a credential; and
- transmitting, to the first provider via the network, notice of authentication of the first user and the received signed message.

2. The method of claim 1, wherein: