White Paper

incentivated
building mobile

# Two Factor Authentication

## Using mobile for additional IT security

9th June 2011

# Introduction

Many companies have already identified the need for additional security for their computer networks, websites and other IT assets.

This additional security typically takes the form of 'multi-factor authentication' where a 'factor' is one of:

- Something you have
- Something you know
- Something you are

The most common variant of this is 'something you have' – i.e. a physical device which provides a unique and one-use code for authentication.

However, these devices rely on the user carrying them at all times, and although they are small, they have a minimum size due to the requirement for a display and sufficient battery capacity to make them reliable throughout their life expectancy.

The obvious extension to this concept is to use something which is always accessible to the user. Biometric scanners ('something you are') which scan fingerprints or unique features of user's eyes are very expensive to produce, and are required at each location that authentication is required so are out of the scope of most projects.

Given that 95% or all adults over 15 in the UK own a mobile phone[1], and most people ensure that their phone is with them at all times, mobile becomes an ideal candidate for the 'something you have' element of multi-factor authentication.

This document outlines some of the issues involved, and how Incentivated can help your business leverage our technology to achieve your security goals without purchasing additional expensive or cumbersome hardware.

[1] OFCOM Consumer Market Report August 2010

# 1. Existing Authentication Solutions

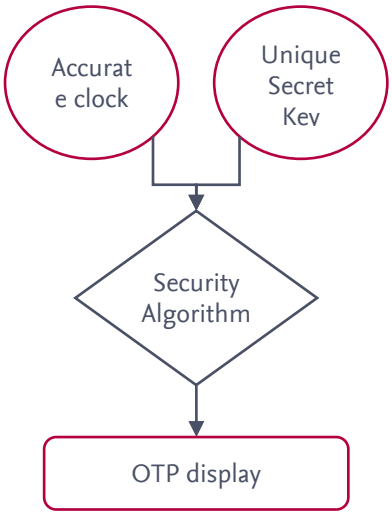


Figure 2 - Time-based token methodology

1.1. Probably the most ubiquitous solution on the market at the moment is the RSA SecurID tag [Figure 1]. This, and most other simple code display products on the market tag works by means of a 'time-synchronised' One Time Password (OTP) as outlined in [Figure 2].

1.2. Another solution along similar lines is a secure USB 'dongle' or 'key' which needs to be plugged in by the user. This dongle has the encryption key encoded within the hardware, which can then be read by software on the users' machine. The primary issue with this solution is that it requires software and drivers to be installed on the client machine.

1.3. A way of getting around the need for software is for the dongle to present itself as a keyboard to the computer. This requires the user to physically press a single button on the device, but means that it can be used anywhere. Yubico have produced an increasingly popular open source solution along these lines called a 'Yubikey[2]'.

1.4. An alternative to time-synchronised devices is to keep a sequence number within the device instead of a timestamp. For USB dongles such as the Yubikey, this makes sense as the device can 'know' how many times it has been used. This number then becomes the seed key for the encryption algorithm, and has the advantage that multiple logins with the same code are not possible; reducing the efficacy of 'man in the middle' attacks.


Figure 1 - RSA SecureID token

1.5. All of these solutions however, require the use of a physical device, which needs to be carried by the user at all times they need access to the network or machine(s) being secured. One solution to this is to use biometric solutions (fingerprint, eye or other physical scanners), which come under the 'something you are' factor, rather than 'something you know', but deserve a mention here.

1.6. The main drawback with biometric scanners is the cost and availability of the equipment required. Some laptops and keyboards have fingerprint scanners built into them, but these also require special software installed on the device, so do not suit external or remote access from a range of devices.

# 2. How can mobile help?

2.1. The primary requirement of a 'something you have' multi-factor authentication solution is that the user must have that item with them. Given that one thing that the vast majority of people own, and keep with them at all times is their mobile phone, this provides an ideal platform on which to deliver this second authentication factor.

2.2. People consider mobile phones to be very personal items, and so they are much more likely to be properly looked after than simple key fob- or USB-based devices. Combined with the security features available as standard on most 'smartphone' devices (remote wipe, 'locate me' and encrypted file systems), the mobile phone is an ideal platform on which to provide secondary authentication factors within an enhanced security solution.

[2] See www.yubico.com for more details

2.3. There are a number of ways in which mobile can be used for two-factor authentication, depending on the level of security required, the range of users devices, and the systems being connected to. This document sets out to discuss many of the various alternatives available.

# 3. Considerations

3.1. Any security solution should be considered carefully to suit the required application, taking into account the following aspects of the system:

- Systems – are you trying to secure a distributed network of devices? Single machines? Physical locations?
- Data – what are you trying to secure? How critical is the data and what other security plans are in place for that data?
- Access – what access is already available to the data? Is this security effort in order to enable remote access? Or to improve existing access provisions? Will the system only be accessible from onsite? Or from public locations? Users' homes?
- Users – who are they? What mobile devices do they own? What devices are they using to access the secured systems? Will their devices have online access when they need to authenticate?

3.2. There are five steps involved in any two-factor authentication 'transaction' – the request, where the user 'asks' to receive an authentication token, OTP generation, where the OTP is created for the user, the response, where the system provides the OTP to the user, entry, where the user enters the OTP in order to access the system, and finally validation, where the server checks the OTP entered is valid.

3.3. Figure 3 shows a generalised 'user experience' for designing a suitable mobile two-factor authentication solution. Some of the product options can be described below and are provided in detail in the following sections:

- SMS / MMS
  - User requests an OTP via SMS or on the machine to which they need access
  - User receives a code via SMS or MMS reply (a picture is a more secure way of transmitting an alphanumeric code)
- Mobile website
  - User logs in to an HTTPS secured mobile site and is recognised by the site
  - Site displays an alphanumeric code or image of that code to use for auth
- Dedicated application (app)
  - Works in a similar way to the site, user runs the application and receives the OTP
  - Application can take advantage of existing device-specific encryption features
- Cross-platform application
  - Appears like the dedicated application but is built for all mobile operating systems
  - Cannot necessarily take full advantage of device-specific features

3.4. Figure 4 on the following page gives an at-a-glance view of the pros and cons of various 'response methods' – i.e. what the user interacts with to get the OTP.

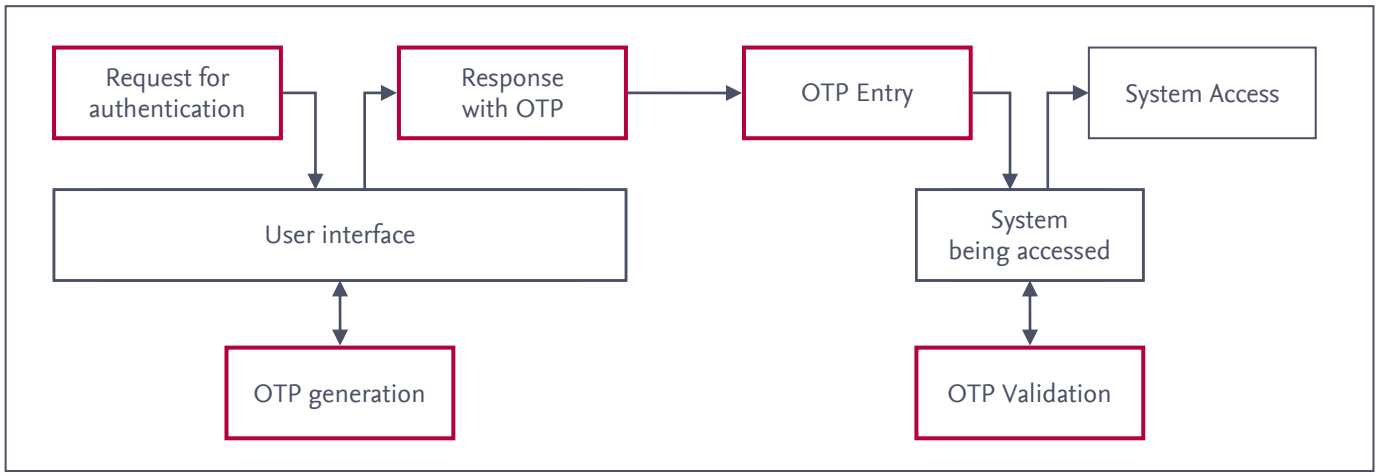


Figure 3 - Two Factor Authentication user experience. Boxes in red are phases involving the mobile solution, defined below

| L=Low, M=Medium H=High | SMS / MMS | IVR | Mobile Site | X-Platfrm App | Dedicated App |
|---|---|---|---|---|---|
| **Security level** | M | H | M | M | H |
| **Relative setup cost** | L | L | M | H | H |
| **Relative running costs** | H | H | L | L | L |
| **Extra user authentication** | PIN | PIN/Voice | Spec. URL<br>Device ID<br>Password<br>Location | Device ID<br>Password<br>Location | Password<br>Location<br>Device ID |
| **Latency of request** | M | L | L | L | L |
| **Picture / Captcha** | Yes | No | Yes | Yes | Yes |
| **Rel customer cost** | M* | H* | L† | L† | L |
| **Online access required?** | No | No | Yes | Yes | (If server generated) |
| **OTP generated where?** | Server | Server | Server | Server | Server /Device |

* depends on user's mobile contract
† depends if user has data bundle (although v. low usage)

Figure 4 – Summary of response methods possible with a mobile device

# 4. Request Phase

4.1. The user needs the ability to request an authentication token. In the case of a physical token such as the RSA SecurID product, this 'request' simply requires looking at the device, but for a multifunction device such as a mobile phone clearly an alternative is required.

## i: Periodic push

4.2. For some circumstances it may be sufficient for the user to receive a 'password for the day' or similar on a predetermined schedule. This really is not a request at all, but ensures that the user receives their OTP and can keep it until they need it.

4.3. In the event of the device being compromised, there has to be a means of cancelling this periodic request as soon as possible, and the 'current' OTP should also be invalidated.

4.4. Due to the potential for compromise, this solution is arguably the least secure of the alternatives listed here, but it may well suit some solutions with simplistic security requirements.

## ii: SMS

4.5. If the user needs to request an OTP as and when they need it, then the most universal way of doing this is to require the user to send an SMS message to a shortcode (5 digits) or longcode (standard 07x number).

4.6. This mobile-originated message could include a user-specific PIN code to avoid the potential compromise in the event of a device being stolen, but if there is a history of SMS messages on the device then this PIN code will be visible, so shouldn't be relied upon as a security measure.

4.7. One concern about an incoming SMS request for an OTP is the possible delay, although the vast majority of SMS messages are received within minutes of leaving the system, they may take considerably longer than this (typical retry times are up to 3 days for SMS messages).

## iii: Voice dial-in

4.8. Due to this potential for delay / failure with an asynchronous solution like SMS, an alternative entry such as a voice dial-in 'emergency' number should always be considered.

4.9. The user could dial a phone number to request that the OTP be read back to them by an automated system – if necessary after the entry of a PIN code or by saying a recognisable phrase.

4.10. The automated phone system will initially recognise the user and validate the PIN code or phrase, by means of the network identifier (MSISDN or caller ID) number.

4.11. Without some additional form of identification this solution suffers from the same potential compromise solution as SMS, but if a PIN code is used this at least will not be stored within the phone's memory after the call is complete.

## iv: Mobile Website

4.12. Another way of avoiding the potential time delays associated with SMS request methods is to use a mobile website. A unique-to-user URL could initially be 'pushed' to the user in an SMS message, and the user can bookmark this URL on their device.

4.13. When they click on the link the device uses a unique identifier for the browsing device (Incentivated produce an identifier via a number of properties of the browsing session) to check the legitimacy of the request before providing the OTP to the user.

4.14. Just like the other request methods, the user can be required to enter a PIN code or passphrase to ensure that they are who they claim to be and the device hasn't been compromised.

4.15. For obvious reasons, this solution requires that the user's device be connected to a mobile data network (WiFi will not provide the same unique identifier so cannot necessarily be used) in order to communicate with the server and provide an OTP.

4.16. One added benefit of this approach is that the device's built in location aware functionality is available, so the system can log the users' location and use it to further authenticate the request (e.g. they have to be within 500 yards of the system they are trying to access). One drawback of this is that many devices will be indoors with poor satellite coverage, so alternative procedures must be considered.

4.17. The site can encourage the user to 'add to home screen' allowing quick access to the site for frequent visitors.

## v: Cross-platform App

4.18. A cross-platform app is one built using Incentivated's device specific wrapper, which enables us to produce fully featured applications rapidly and at a low cost, which still take advantage of a number of phone features.

4.19. Such an application can provide the user with an intuitive interface to request an OTP along similar lines to the mobile website solution outlined above.

4.20. Entry to the application will be via an icon on the users' application or home screens depending on the device.

4.21. Despite this appearing a very cost effective and user friendly approach, it in fact offers little improvements for an authentication solution over a standard mobile web site with 'add to home screen' functionality as outlined above, so for the purposes of this whitepaper will not be discussed further.

## vi: Dedicated App

4.22. A dedicated application has the massive advantage that it can make full use of the features available on the device, and with modern smartphones this often includes advanced encryption facilities for data storage on the device itself.

4.23. Such facilities could be used to provide offline authentication of the user, so that their device is not required to be online to receive their OTP – as is the case with all of the above options.

4.24. The request phase however, would appear identical to the user to the Mobile Website or Cross-platform app solutions, appearing in the application or home screens. Once the user clicks on the icon, they receive a welcome screen and can touch a button to receive their one time password.

4.25. As before, the user can be required to enter a PIN code before they see the OTP.

## vii: Dedicated App with NFC token

4.26. A new mobile technology being adopted by phone manufacturers is the addition of short-range Radio Frequency Identifier (RFID) chips to devices.

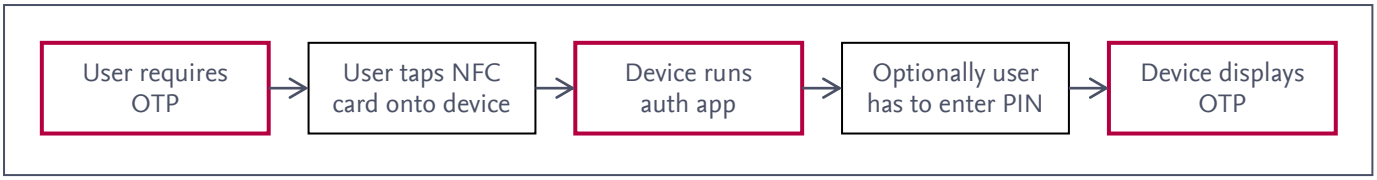


Figure 5 - NFC token used to generate an OTP

4.27. The use of these chips within mobile is known as Near Field Communications or NFC, and is slowly becoming more popular as the latest devices adopt the technology.

4.28. For the purposes of this discussion, the technology essentially consists of 'tags' which can be stuck to or embedded within business card sized items. These tags store text, or an action which can be interpreted and carried out by the phone.

4.29. Due to the compact size, and obscure nature of the tags (they don't have to obviously be labelled 'security key' for example), they can be used as an authentication token for a dedicated app. For added security, the app could even re-programme the tag on a per-use basis and 'blacklist' the previous code, to avoid unauthorised copying of the tag.

4.30. Figure 5 shows the user experience for NFC-based security, when used without a PIN it is a very quick and efficient way to retrieve an OTP, requiring very little user effort.

# 5. OTP Generation Phase

5.1. Some means of producing the one time password or authentication key is required. In general this falls into two categories – either generated directly on the mobile device, or pulled from an internet-based server or site.

5.2. In addition, the server providing the OTP could be one hosted by Incentivated, or hosted within the clients' existing IT infrastructure.

## i: Internet based OTP generation service

5.3. In the case of the voice- or SMS- based request methods, the server providing the SMS / Voice service must communicate to the OTP generation service – this communication may need to be over the public internet in which case an appropriate security protocol should be employed to avoid 'man in the middle' attacks.

5.4. Similarly, for the mobile internet solution, the web server host must be able to securely communicate with the OTP generation service.

5.5. For the cross-platform or dedicated app solutions, the application itself needs to communicate securely with the OTP generation service. This is best done with an HTTPS-secured web service.

5.6. Incentivated can provide the OTP generation service with appropriate encrypted channels designed for use on a mobile device if required. Please contact your account manager for more details.

## ii: Direct OTP generation on the mobile device

5.7. Although at first glance, generating the one time password on the users' device directly removes the risk of communicating keys over the public internet. However, the encryption functionality available on a number of devices may not be sufficient to generate a suitably secure password, so a compromise needs to be drawn between convenience, feature requirements, multiple device support and security.

5.8. For obvious reasons the only solution which will provide direct OTP generation is an application. Cross-platform applications have no access to device security features, so a dedicated application is recommended for this OTP generation method.

5.9. Paragraph 5.7 notwithstanding, later versions[3] of the Apple iPhone, Android, Windows Mobile 7 and RIM (Blackberry) all provide built in encryption facilities for data, so a private key can safely be stored on these devices and used to generate a one-time password. Featurephone and older devices however may not universally have this flexibility.

[3] iPhone versions 3GS and up with the latest OS updates support encrypted files within device-specific applications.

5.10. As well as the source of the key, the information to be encrypted to generate the OTP should also be decided, as mentioned above this can either be time-synchronised or sequential, depending on the requirements. Typically for user-initiated solutions such as this a sequential 'seed' provides the best security and maximum reliability, and will be assumed for the remainder of this study.

5.11. The format of the one time password is important; whether it is numeric or alphanumeric, and the required length of the resultant OTP. As it is likely that the user then will have to type the OTP into a computer, this should be as short as possible whilst maintaining the required level of security.

## iii: OTP algorithm

5.12. There are a number of considerations to be taken into account in generating the OTP itself – Incentivated's OTP generation service uses industry standard public/private key encryption techniques to generate the password effectively as described below.

5.13. The constituents used to generate the OTP are as follows:

- A private key, which is specific to and only accessible by the device (or in the case of server-side generation, the user session). This private key is the most critical security requirement of the system, as if compromised; an attacker can recreate OTP strings at will, without the knowledge of the user.
- A sequence number or timestamp
- (Optionally) a device or additional shared secret, which is added to the sequence number or timestamp for additional verification.

5.14. Recalling the diagram in Figure 2, the service uses the private key to encrypt the sequence number and shared secret and then generate the OTP in the required format. The encryption algorithm itself is outside the scope of this document but can be provided if required, or custom built to client requirements as appropriate to the solution.

# 6. Response Phase

6.1. Once the OTP has been generated, it needs to be provided to the user. The simplest way is to display it within the application or mobile website, or within an SMS response to the user.

6.2. In the case of a voice dial-in solution the OTP needs to be read out using a text-to-speech engine.

6.3. One consideration for the response to the user is 'man in the middle' attacks, both physical and technical. In the case of a web or SMS displayed OTP, an attacker may be able to read the OTP before it gets to the user. A potential solution to this man-in-the middle attack is to display the OTP as an image; especially an image which has been distorted so it is difficult to read using Optical Character Recognition (OCR) software – the technology to do this is well understood in the form of 'human detection' CAPTCHA images[4].

6.4. OTP images can be displayed to the user within a website or application, or sent directly via MMS to the users' handset.

6.5. Alternatively, with the addition of appropriate hardware and software on the target system, the phone could display the OTP as a 2-dimensional barcode to be 'read' by a camera on the target system, or transmit the OTP over Bluetooth to it. These solutions have their own security considerations, and require a potentially costly rollout but can be used with far longer (and hence harder to compromise) OTP formats. This solution can be provided by Incentivated but is out of scope for this document.

# 7. OTP Entry Phase

7.1. Once the user has the OTP, they need to enter it into the system. Obviously this stage depends on the system being authenticated. A number of common systems are outlined below, but integration with other third party systems can of course be investigated.

[4] See http://en.wikipedia.org/wiki/CAPTCHA for a description of the technology

## i: Windows Domains

7.2. Incentivated can develop a client-branded authentication plug-in for Windows Vista and above, which requires that the user type in their OTP before their regular Windows password. No change to the domain controller configuration is needed for this authentication solution; an installation package is simply rolled out to the workstations that require two factor authentication.

## ii: Web Sites / Apps

7.3. Incentivated can provide plug-ins for Apache (implemented as a module) and IIS web servers (implemented as a .NET filter) which provide authentication facilities for web sites via HTTP(S) BASIC and DIGEST authentication. The user is asked to type the OTP and password together in the browser displayed password box.

7.4. Alternatively, and preferred by most site authors, is a web form based solution. Incentivated can provide APIs to authenticate users based on username, OTP and password, and sample code in most common enterprise server programming languages.

## iii: Enterprise applications

7.5. The same APIs mentioned above can be built into enterprise applications for authentication if required.

7.6. However, as these applications may not be sufficiently modular to add authentication layers, Incentivated can investigate provision of various solutions based on industry standards such as RADIUS, CHAP / PAP and LDAP.

# 8. OTP Validation Phase

8.1. Once the user has provided their OTP to the server, it needs to be validated. If created using the method described in section 5 with private key encryption, then the corresponding public key is required to decrypt the OTP and validate the sequence number or timestamp and (if required) verify the additional shared key.

8.2. If a sequence number is used, then it is compared with a counter associated with the user's record and passed if the user's current counter is lower than the sequence number passed.

8.3. If timestamps are used, then the timestamp should be within a predetermined time interval from the current time. Here, care must be taken to ensure that the server and devices' times are in sync. An additional check to ensure that the timestamp hasn't previously been used should also be made.

8.4. Once validated, then the standard password is passed on to the original authentication method to ensure the user has access to the system.

8.5. The OTP validation system needs access to a database of user IDs, public keys and 'used' sequences / timestamps, as well as any other access controls (user disabled, access allowed times, etc) as required. Access to this database should have appropriate access and audit controls in place for obvious reasons.

## Conclusion

In summary, a mobile device is a very capable solution for providing one time authentication passwords to users, and adding a second authentication factor in a corporate security system.

As with any security system, care must be taken that that system is specified correctly, and the level of sophistication and functionality is appropriate for the level of protection required, and weighed up against user experience for that system.

Incentivated can help you evaluate the various available solutions and work through to rolling out a reliable security solution whilst not inconveniencing your users.

# Contact us for more information

Email: info@incentivated.com

Web: www.incentivated.com or scan the QR code below to see our optimised website on your phone.

Tel: +44 (0)20 7392 2323

A selection of other white papers we offer:

- The Mobile Web
- mCommerce
- Mobile Coupons

Incentivated is an independent technology company with 10 years' experience operating exclusively in the mobile marketing services sector.

We help our international client base engage with their customers by designing, developing and delivering integrated acquisition, retention (CRM) and transaction (mCommerce) campaigns and services for mobile.

Our proprietary technology and specialist staff are well positioned to help brands, the public sector and charities to develop everything from enterprise messaging (SMS & MMS) through to mobile internet sites, to server-side software or handset applications, including web-apps, for 'smartphones' and feature-phones.

We also provide strategic, creative and technical advice for the use of mobile by businesses to raise awareness, deliver marketing ROI and provide customer service.

Scan the QR code below to see our website optimised for your mobile phone, but accessed through our existing website

WP-2FA1.0/06-11