This is a draft
Your feedback will help us improve it.

Using authenticators to protect an online service

1.0 You might need to know if a user has already created an account with your service before you give them access to it. This is called 'authentication' and can be useful if users need to sign in to your service more than once.

1.1 This guidance will help you choose the 'authenticator' that will give you the level of protection that's right for your service.

1.2 An authenticator could be some information (like a password), a piece of software or a device.

> You might also need to prove and verify a user's identity if your service gives them something valuable or lets them access personal information.

2.0 Different types of authenticator

2.1 There are different types of authenticators. An authenticator will usually be one of the following:

- something the user knows
- something the user has
- something the user is

2.2 Sometimes an authenticator can fit into more than one of these categories.

> **Example**
> A password is something the user knows. It's secure because no one apart from the user will know or be able to access it. If the user writes it down on a piece of paper, it will also become something the user has. The piece of paper is now the authenticator because it contains information that previously could not be known by anyone apart from the user.

3.0 Something the user knows

3.1 The most common way for users to sign in to a service is by entering a piece of information that only they know. This is called a 'secret'.

3.2 A secret could be something like:

- a PIN
- a password
- an answer to a question that only the user knows the answer to - also called knowledge-based verification (KBV)

3.3 A secret is one of the easiest ways for someone to sign in to a service, as a user does not need any special equipment or software to use it. But secrets can be easily:

- stolen, for example from a phishing attack
- guessed, for example if the password or PIN is low quality (like '1234')
- found out, for example if the answer to a KBV challenge is publicly available
- shared

> Read the National Cyber Security Centre's (NCSC's) guidance to find out how your service should use passwords.

3.4 You might need to use a different type of authenticator if you need to be more sure the user who's trying to sign in is the same person who created the account.

3.5 A secret is usually used with either:

- another piece of information, such as a username or email address
- a token, such as a chip and PIN card

4.0 Something the user has

4.1 A user might be able to sign in to a service using something called a 'token'. A token can be something physical, like a chip and PIN card or a mobile phone.

> **Example**
> A user can sign in to a service with a physical security key by inserting it into their computer or tapping it against their phone. This proves that someone is present when they're trying to sign in to a service.

4.2 A token can also be something digital, like a single use authentication code or a digital certificate.

> **Example**

When a user adds an electronic wallet app to their mobile phone, a digital certificate is created and stored securely within their phone. This digital certificate is the token. When a user pays for something using their phone, the bank checks the digital certificate. If the bank is sure the phone is the same device the user installed the app on, it will approve the transaction.

4.3 Using a token by itself might not be appropriate if your service needs a high level of protection. This is because tokens can be easily lost, stolen or shared.

A token can also be copied or tampered with if:

- it does not have any security features
- the security features it has have been badly designed

4.3 A token can only confirm that someone is there, which can help protect your service from being attacked by remote hackers. Unless you combine it with biometric information, you will not be sure that a token is being used by the same person that created the account.

4.4 But some tokens can contain information about:

- the person that's using it to sign in to the service
- the organisation that issued the token (for example a chip and PIN bank card will include the name of the bank that issued it)

5.0 Something the user is

5.1 A user might be able to sign in to a service using their biometric information. Biometric information is a measurement of someone's:

- biological characteristics, such as their fingerprint
- behavioural characteristics, such as their signature

Example
A user can open an app or unlock their phone by looking at it.

The app or device uses facial recognition software to check the user looks like the person who created the account or registered the phone. If there's a match, the user can access the service.

5.2 Using biometric information means your service can easily tell if the user who's trying to sign in is the same person who created the account. This is because:

- each person's biometric information is unique to them
- it's difficult for biometric information to be forgotten, lost, stolen or guessed

  Read the NCSC's guidance to find out more about how biometric information can be used to access a service.

5.3 There are some risks to using biometric information as an authenticator.

5.4 There's a small chance someone could try to impersonate another user by recreating their biometric information. For example, they could:

- hold up a photo of the user
- wear prosthetics or a mask to make themselves look like the user
- play a recording of the user's voice
- use a copy of the user's fingerprint

5.5 Some types of biometric information will be easier to recreate than others.

5.6 These are called 'presentation' or 'spoofing' attacks. Although attacks can be detected by the system that's used to capture biometric information, there's always a risk a fraudster could successfully sign in to a service this way.

5.7 It's also possible that the system can make a mistake when it's matching someone's biometric information. It could either:

- wrongly match a user to another person (called a 'false match')
- not be able to match a user to anyone, even though a record of their biometric information exists (called a 'false non-match')

5.8 It's easier to make these mistakes when matching some types of biometric information than others. You can lower these risks by asking users to use another authenticator as well as their biometric information.

6.0 2 factor authentication

6.1 You can protect your service using a combination of 2 authenticators. This is called '2 factor authentication' (2FA). It helps protect your service against some of the risks that come from using just one type of authenticator.

> **Example**
> A user can sign in to a social media account using a username and password (something they know) and an authentication code sent to their mobile phone (something they have).

If a user could sign in without the code, there's a risk that someone else could guess or steal their password to access their account. Using an authentication code as another authenticator means that, even with the password, a fraudster would still not be able to access the account.

6.2 It's important that you use 2 different types of authenticator. A fraudster who's able to compromise one authenticator is likely to be able to compromise another of the same type using a similar method.

Example
A fraudster can guess both a user's password and their PIN. This is because they're both the same type of authenticator (a secret). Fraudsters know users will often use personal information, such as their date of birth, to create them.

6.3 You can choose to protect your service using more than 2 authenticators. This is called 'multi factor authentication'. If you do this, you should be aware that it can have an impact on the usability and cost of your service.

7.0 The quality of an authenticator

7.1 A authenticator can be low, medium or high quality. The quality of an authenticator will depend on how secure it is.

7.2 The most secure authenticators have a strong link to the user. This means it's difficult for someone other than the user who created the account to guess, copy or make changes to the authenticator.

7.3 High quality authenticators are also protected against 'large scale' attacks. These are automated attacks that use large databases of stolen or weak authenticators to try to break into users' accounts.

7.4 The quality of a authenticator will depend on how it was:

- created by a user (or a manufacturer if it's something like a physical token)
- managed (including how the authenticator is issued and updated, and what happens when it's no longer being used)
- captured (if it's biometric information)

7.4 Some types (or 'modalities') of biometric information are higher quality than others. Read the NCSC's guidance to find out how to pick the right biometric modality for your service.

8.0 Low quality authenticators

8.1 A secret is low quality if it's one of the following:

- a password
- a PIN
- a KBV challenge based on information that does not change over time (known as ‘static’ information)

8.2 A token is low quality if you do not know how it was created or issued. This is because there could be a risk it was tampered with or issued to someone who should not have it.

8.3 Biometric information is low quality if you do not know how it was captured or processed. Using low quality biometric information might mean your service is at risk of being attacked.

8.4 For example, if your service is checking a facial biometric, you might be at risk of attack if you let a user upload a photo that they previously took of themselves. This is because a fraudster could use a photo of the user that they found on social media. It would be more secure to use 'live capture', which involves asking the user to take a photo on their webcam or phone and upload it as they use your service.

8.5 Using a low quality biometric information could also stop you from doing an accurate biometric comparison, which means you might not be able to match the right person to a record.

9.0 Medium quality authenticators

9.1 A secret is medium quality if it's either:

- automatically created and securely stored (for example in a password manager)
- a KBV challenge based on information that changes over time (known as 'dynamic' information)

9.2 A token is medium quality if you know tokens are created in a way that stops them from being:

- tampered with
- issued to the wrong person

You can find this out from the manufacturer or supplier of the token. For example, a manufacturer might have published a report or white paper that documents how the token was created.

9.3 Biometric information is medium quality if you know the system that captured it when the user created an account can detect spoofing or presentation attacks. You can find this out from the manufacturer. For example, they might have published a report or white paper that describes how the system detects presentation attacks.

10.0 High quality authenticators

10.1 A authenticator is high quality if it could not belong to anyone other than the user who created the account. A secret cannot be a high quality authenticator because it's easy for someone to steal, guess or copy.

10.2 A token is high quality if it has been independently tested to prove it meets industry standards, such as the Common Criteria guidelines, FIDO or NIST FIPS 140-2.

10.3 Biometric information is high quality if the system or process used to capture it has been independently tested to check if it meets industry standards, such as ISO/IEC 19795-1:2006 or ISO/IEC 30107-1:2016.

10.4 This will prove that the technology can protect your service from spoofing or presentation attacks.

11.0 Choosing an authenticator

11.1 An authenticator can protect your service from being accessed by someone who should not be able to use it. How much protection your service needs depends on:

- what information the user needs to use the service
- what information the service gives the user access to
- what the service or user can do with that information

When choosing an authenticator, you should also think about how much risk your service can accept. You should do a risk assessment if you do not already know this.

> Read the guidance about making sure your online service safe and secure to find out how to do a risk assessment.

11.2 You get different levels of protection by using different authenticators or combinations of authenticators.

11.3 Some authenticators might be easier to use than others. You should make it easy for users to access your service, but also make sure you choose an authenticator that gives you the level of protection you need.

11.4 Some users might struggle to use an authenticator that others find easy to use. For example, accessing an app using a fingerprint can be easy for a lot of users, but it would be very difficult for someone who has lost the use of their hands.

11.5 You should think about the types of people your users are when you choose an authenticator. Make sure there's more than one way they can sign in to your service. All options should give your service the same level of protection.

11.6 Some users might not be able to access your service online. If there are other ways to access your service, you should make sure they have a similar level of protection.

12.0 Low protection

12.1 You might need low protection if the information that's shared between the user and your service cannot be used to:

- get anything valuable
- do any harm to the user if it's seen by someone else

12.2 Low protection gives you some confidence your service cannot be accessed by users who have stolen or copied authenticators.

12.3 You'll have low protection if you protect your service with any low quality authenticator.

> **Example**
> A wifi network in a public place can usually be accessed with an email address and a password (something the user knows). The service does not need to be protected by a more secure authenticator than this because it only gives users access to the internet. If a user had unauthorised access to the network, it's unlikely they'd be able to use any information they get from the service to do any harm.

13.0 Medium protection

13.1 Your service might need medium protection if it gives users access to sensitive information.

13.2 Medium protection means your service knows:

- the authenticator is being used by the person who created the account

- any information (such as bank data in a chip and PIN card) is securely stored in the authenticator and has not been tampered with

13.3 You'll have medium protection if you protect your service with 2FA. Use 2 different authenticators from the following list:

- a low or medium quality secret
- a low or medium quality token
- low or medium quality biometric information

> **Example**
> A user will usually need to use 2FA to sign in to their email account. They might need to enter:
>
> - their email address and password (something the user knows)
> - an authentication code that has been sent to their mobile phone (something the user has)
>
> The service is protected by 2FA because it gives users access to emails. These could include sensitive information from their doctor, bank or family.

14.0 High protection

14.1 Your service might need high protection if it lets users access information that could be used to cause harm. This could be to another user, service or organisation.

14.2 If your service has high protection, you can be sure that the authenticators will not be used by anyone apart from the user who created the account.

14.3 You'll have high protection if you protect your service with 2FA that uses a medium quality authenticator and a high quality authenticator.

> **Example**
> A user can access their online bank account using a fingerprint (something they are) to sign in to an app on their phone (something they have). They will only be able to sign in if the app can confirm the fingerprint and phone belong to the same user that created the account.
>
> The fingerprint is a high quality authenticator. This is because it has been captured by a system with a documented process for capturing and checking biometric information.

15.0 Very high protection

15.1 Your service might need very high protection if it lets users access information that could be used to cause significant harm. This could be to another person, service or organisation. You should use it if the information your service keeps is more sensitive than information kept by a service with high protection.

15.2 You'll have very high protection if you protect your service with 2FA that uses a high quality token and high quality biometric information.

15.3 Very high protection gives you more confidence that the authenticators will not be used by anyone apart from the user who created the account.

15.4 This is because you're using 2 high quality authenticators, which will have been protected from being attacked when they were created or issued. You'll be sure that any information stored in the authenticator cannot be changed by someone other than the user or the organisation that issued it.

> **Example**
> Some organisations might need access to sensitive or personal information about their customers. To do this, employees could use a combination of:
>
> - a certificate-based smart card (something the user has)
> - their fingerprint (something the user is)
>
> These are 2 high quality authenticators. The smart card is protected by cryptographic security features which means it's difficult for someone to copy or tamper with it. A fingerprint is unique to that user and will be stored in the chip in the smart card.
>
> The service needs high protection because it gives employees access to sensitive information. If someone other than an employee managed to sign in to this service, they could use the information to do harm to a customer. This could also cause damage to the organisation's reputation by showing that their process is not secure.

16.0 If an authenticator has been forgotten, lost or stolen

16.1 You should give users a way to access your service if their authenticator has been forgotten, lost or stolen. You must make sure that the person you give the replacement authenticator to is the same user who first created the account.

16.2 You can do this by using either:

- information that you know was given to the user when they set up the account (such as a backup or recovery code)

- contact details that you know belong to the person who set up the account (such as their email address or phone number)

  Example
  A user has forgotten their password but still has access to the email address they used to set up their account. A service can use this email address to send them a link to a page that tells them how to reset their password.

16.3 You might need to be more confident that the user is the same person who created the account if all of their authenticators have been forgotten, lost or stolen. You should do the same checks you would do to make sure an identity belongs to the person who's claiming it.

> **Example**
> Some social media websites ask a user to upload an image of a piece of evidence if that user has lost access to their account.
>
> They check the photo and information on the evidence match the details from the user's account. If there's a match, they'll know that the user is the same person who set up the account and can let them sign back in.

17.0 If an authenticator has been revoked

17.1 You might need to temporarily stop a user accessing your service if you think their authenticator has been compromised.

17.2 You must:

- ask the user to use a new or different authenticator, if it's a secret
- cancel (or 'revoke') the authenticator and issue a new one, if it's a token

17.3 As biometric information is a measurement of someone's biological or behavioural characteristics, you will not be able to revoke it or issue a replacement authenticator if it's compromised.

18.0 Monitoring how users use your service

18.1 As well as making sure you give the right users access to your service, you should also look out for any unusual activity once they've signed in. This is called 'transaction monitoring' and will help keep your service and your users' data secure.

18.2 For example, it can protect your service from 'authenticator stuffing'. This is when a large number of usernames and passwords taken from compromised websites are automatically used to try to get access to a service.

18.3 You should look at who's signing in to your service and look for information like:

- their name or another identifier (such as their username or email address)
- their IP address, to find out where they are
- what device they're using