

A security analysis of MitID

TALENT-TRACK, FALL 2021

Thomas Kingo Thunbo Mogensen, studienummer: 201704877

Institut for Datalogi, Aarhus Universitet

Contact: Thomaskingo.tm@gmail.com

11.03.2022

1 What is MitID?

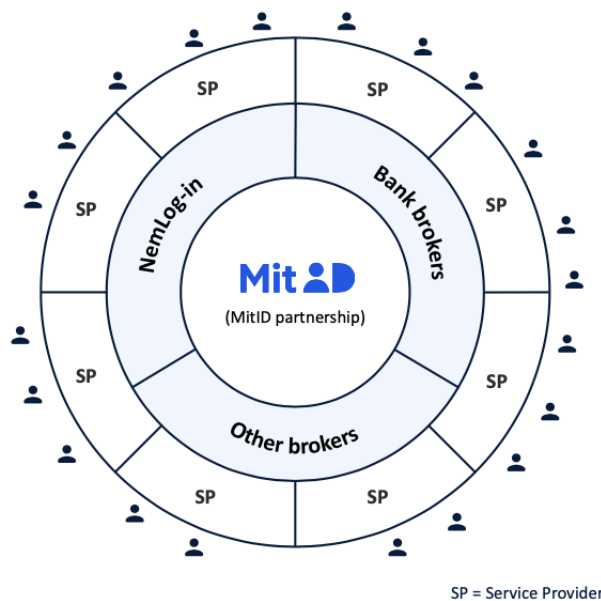
As part of the Talent-track on the Computer Science Bachelor in Aarhus I've chosen to dive into the new eID-system of Denmark, called MitID, which will replace the 10-year old system NemID. MitID is a collaboration between the Digitisation Agency, an agency under the danish state, and *Finans Danmark*, the interest organization for the banks. This report contains our findings as of December 2021. We disclosed this information to MitID exactly three months ago, and the system might have changed in the months that have passed since then.

In many ways MitID looks like the old system, but the underlying structure is more cohesive and composable, and the underlying cryptography can quickly be swapped out, if necessary. NemID has been the electronic identification system used in Denmark for the past 10 years, and the purpose of it is to give assurance to online services that the correct person is performing the action or transaction. In practice, NemID has given access to services like online-banking, skat.dk and borger.dk. Borger.dk is a big collection of services such as issuing marriage certificates, ordering a health-insurance card, applying for different kinds of financial aid, or even giving power of attorney to someone else. All of these services are locked behind authentication using NemID, and from now on they will instead be locked behind authentication using MitID.

1.1 MitID Structure

MitID is structured with a core and surrounding layers. There are 3 layers as shown below:

- ✓ Future-proof – flexible and modular
- ✓ Easier to develop
- ✓ One unified system



Every layer in the structure can only be accessed by its immediate surrounding layer. End-users will access a service from a Service Provider by being redirected to log in through a Broker. The Broker then handles the actual access to the core and simply communicates the result of the authentication to the Service Provider, in order for the Service Provider to allow access to the needed service.

1.2 MitID core

"MitID will be a common, national solution for identities and authentication, that consists of an identity-core to support authentication and life cycle management of personal digital identities." [12] Everything necessary to do authentication lies inside the core: Knowledge of the specific identification means associated to each user, and the Digital Identity of each user, where a Digital Identity refers to a collection of data that uniquely identifies an individual or professional user. [4, chapter 2] Also, Risk-data and Authentication-answers are handled between the MitID core and the brokers. Risk-data refers to the login-location, the device used, network-info, identity-info and information on the last authentication, whereas an Authentication-answer refers to things such as name, session-ID and identification measures - alongside other information on the user and the specific authentication being handled.

1.3 Brokers

Brokers - or identity-brokers¹ - are the entities that mediate access to the core and communicate identities from MitID to service providers. They are used by public actors, banks and other private service providers. More specifically, brokers are certified businesses/entities that serve as the layer around the core, providing access to authentication and handling these authentications. This entails handling user's

¹For a formal definition in danish see 12.7

inputs and checking against the core to ensure that the authentication is valid. In practice a user who wants to authenticate, will go to a Service Provider and try to access a resource or service. The Service Provider will redirect the user to authenticate into MitID through their associated broker. The entire authentication process is handled by the broker, and the resulting assurance-level of the user's identity along with other information needed by the Service Provider, called a "security-token", will be put into a SAML-assertion² and sent to the Service Provider.

The brokers are divided into three categories:

1. Bank Brokers: Access points for banks
2. NemLog-in: Access point for the public sector
3. Private Brokers: Access points for the rest of the private sector

As of this point there are five private brokers who will cover every service provider that wants access to MitID and is neither a Bank nor a public entity like Skat.dk or Borger.dk.

1.4 Service providers

The Service Providers are banks, public entities with options for self-service, businesses³ or even local organizations. For end users these will usually be in the shape of web pages or apps that offer some service in which MitID works as the entry point. The role of Service Providers is threefold:

1. To assess what level of assurance is needed from the authentication in order to give users access to the service.
2. To verify the SAML-assertion (security-token) issued by the Identity-broker.
3. To ensure that the required Assurance Level (low, substantial or high) has been reached and give appropriate access to the user depending on the reached assurance-level.

Service Providers will not handle any part of the authentication, as this is dealt with by the brokers. However, they are required to make a Risk-assessment, which is to assess what Assurance Level users will need to access their resource.⁴

2 Two-factor authentication and identification means

Two-factor authentication means that the user has to verify by using two out of three identification categories: [6, page 7]

- Something you know - passwords, private knowledge

²See the bibliography for a reference to the OIOSAML profile used.

³Businesses like dba.dk and danskespil.dk

⁴Section 12.5 contains the source of this information.

- Something you have - phone, chip or one-time password device
- Something you are - biometrics such as fingerprint and face recognition

MitID has implemented both one-factor and two-factor authentication, based on the Assurance Level needed to access the resource. [5] In practice, almost all users will do two-factor authentication, since this is required by public entities [4], so a typical login-flow would look something like the following, assuming the app is used:

- Press "log on" and get redirected to the MitID login page
- Type in UserID and press continue
- Open MitID-app and type pin-code or use biometrics to access the app
- Swipe right on the app (assuming the name of the correct webpage/service is shown in the app)
- The user will now have access to the wanted resource

2.1 No password for MitID-login:

A big point of contention is the fact that MitID doesn't require you to type in a password alongside your UserID when starting a login-session. This allows for some confusion as to what a UserID is. Usually one would expect the name "password" to mean something secret, while in MitID the UserID is now the secret that all users must keep to themselves and never share with anyone else. This is even more problematic since the UserID is visibly shown when a user is logged in - and the UserID is also visibly shown while typing it during login.

2.2 Different devices for two-factor authentication

There are several different factors used to authenticate into MitID. The last three refer to physical devices that users can order:

1. MitID app
2. MitID Code-viewer
3. MitID Code-reader
4. MitID Chip

MitID app:

The MitID app is the solution that most users are expected to use. In many ways it works like the NemID app, and the flow is as follows:

- Open the app
- Type in 6-digit pin-code or use biometrics to open the app
- Swipe right to accept the login-request

This counts as two-factor since opening the app requires a password, which is centrally validated such that brute-force attacks won't be effective. When using biometrics to open the app, this simply releases the 6-digit pin-code that lives in the app [5, page 11] and has been locked behind biometrics, and sends it to be centrally validated.⁵ Having access to the app and knowing the password shows that the user has access to certain knowledge and also has possession of the phone, in which the MitID app is connected to their account. In spite of this, the solution has been under critique because of relying heavily on an app. [2] Although this is a valid argument, certain measures have been taken to keep the app secure, such as checking for Hardware Configurations, overlays, changes to used packages and tampering, making for an app that provides a considerable amount of security.

MitID Code-viewer and Code-reader:

These are physical devices that either show a 6-digit pin-code on the display when the user presses the button, or it reads it aloud in headphones (for visually impaired). These devices are non-personalized when shipped or when given to the user in person. The setup of these devices is done through authenticating into MitID with another identification mean or authenticating into NemID. Then the serial number on the physical device is used along with the first code generated by the device. When generating a 6-digit one-time code, they take into account the current time, and the one-time codes last roughly 30 seconds.^{6 7}

3 Differences between MitID and NemID

The change that becomes apparent immediately when using MitID, is that instead of doing the authentication in a small window on the same web page that you are trying to access, MitID will instead do a full redirect to another page, where "MitID.dk" will be in the URL of the page.

Furthermore, when logging into NemID, a user enters username and password, the app prompts for an accept, so the user opens the NemID app with a 4-digit pin-code to be able to accept requests. In contrast, MitID has renamed the username to UserID, removed the password⁸ and changed the pin-code from a 4-digit pin-code to a 6-digit pin-code. Concerning the structure of the two systems, NemID was two different systems: One for banks, and one for the public.⁹ [13] MitID is one structured system with many entities, each with specified roles to play. It's supposedly easier to develop and maintain, since it is flexible and modular.

Pertaining to security, the main difference between NemID and MitID is that all of the Service Providers had direct access to the information in NemID. In MitID this access is handled through a broker, meaning that only the brokers have direct access, and only the brokers have to update their systems because of changes in the

⁵Section 12.1

⁶After turning it on and off every 2 seconds for a minute or so, it became apparent.

⁷These devices may end up slightly desynchronized, which can be fixed by calling MitID support.

⁸Users no longer provide a password on the website - only the pin-code in the app

⁹See Section 12.2 for a visual representation of NemID

MitID core. It limits the amount of entry-points to the sensitive information by a large margin.

There is also a difference between the respective apps. Where the NemID would prompt the user to approve a request, using push-notifications, the MitID app does not. This is to prevent users from approving requests, which they did not initialize.

Transferring from NemID to MitID:

"When it's your turn to get MitID, you'll receive a message in your online-bank, and then you have 30 days." [15] Transferring requires the user to have NemID, or in special cases the user can have other types of identification, such as a passport or a driver's license, but this means the user has to go to the local citizen service-center to get MitID. Users with NemID will simply authenticate into NemID, receive a few one-time passwords on their phone and then pick their preferred two-factor solution and activate it. This will be the flow for most users.¹⁰

4 Assurance Levels

NSIS is a document that specifies the "National Standard for Identity Assurance Levels". The purpose of the document is to "create a framework for trust in digital identities and digital ID services". [6, page 4] It outlines what is necessary to obtain the different Assurance Levels. It does not specifically state how these things are supposed to be implemented, but it does contain normative requirements concerning electronic identification. NSIS outlines three Assurance Levels: Low, Substantial and High. These three levels will give different amounts of access, depending on the resource, such as read-only for Assurance Level Low. In MitID they distinguish between these three layers depending on the factors used to authenticate. All of them need the UserID to be entered into the MitID form, but they differ quite heavily on the requirements for authentication. For reference, authenticating into NemID equals Assurance Level Substantial. [11, page 9] For each Assurance Level there are two ways to do the authentication.¹¹

- Low: 1. Password or 2. MitID chip
- Substantial: 1. App or 2. Password + Code-viewer/Code-reader
- High: 1. Password + chip or 2. app + chip

Now, the term "password" is a little confusing. The term password in the case of MitID refers to a password that is chosen by the user, when the user activates one of the three physical devices: Code-viewer, Code-reader or Chip.¹² As is evident from the list, the only way to reach Assurance Level High is by having the Chip. In short, authentication gives as output an identity (who the user is), which is input for any succeeding access control (what the user is allowed to do). [6, page 13] Public

¹⁰See [16] and [17] for danish guides on how to get MitID.

¹¹For the original image, see Section 12.4.

¹²Since specific information regarding the password is hard to find, it might be only the Code-viewer and the Code-reader that force the user to create a password.

entities acting as Service Providers, are required to use at least Assurance Level Substantial, if their service provides the user with the authority to carry out any kind of self-service tasks. [4]

5 The MitID App

"MitID is primarily an app" [5, page 3], meaning that MitID users will mainly use the app to authenticate, and so the app will be the most obvious point of attack between the different authentication-factors. Luckily the app has been safeguarded heavily against attacks that would make it run on a machine that could compromise it's security. In terms of hardware, it's possible to have a constellation with a microchip that can "hide sensitive data and start applications protected from malware". [9, page 5] In that case, the app counts as Assurance Level High all by itself. To further protect the app, the 6-digit pin-code required to unlock the app, will be automatically suspended for an hour after three failed attempts, if six failed attempts happen, the pin-code will be blocked.¹³ Furthermore, the keyboard in the app has been specially developed and is controlled by the app itself. Finally, the pin-code is centrally validated [5, page 11] to protect against attacks such as brute force. It is also relevant to mention that whenever a user authenticates into MitID using the app, the service, that the user is trying to enter, has it's name along with the requested action, such as "login", displayed inside the app. Also, this information is sent to the app by the broker, so services cannot trick users into thinking they are legit by simply displaying a fake name.

6 Encryption schemes and standard protocols

For anyone interested in learning about the specific cryptographic solutions currently used in MitID, here's a publicly available list: [3, page 26]

- ECDSA signatures with ES256, ES384 and ES512.
- RSASSA-PKCS1-V1_5 signatures with: RS256, RS384 and RS512.
- RSASSA-PSS signatures (probabilistic signature scheme with appendix) with: PS256, PS384 and PS512.
- HMAC signing algorithms: HS256, HS384, or HS512
- RSASSA-PKCS1-V1_5 encryption with: RSA1_5
- RSAES OAEP encryption with: RSA-OAEP
- ECDH-ES encryption with: ECDH-ES

¹³See 12.3.

So the general cryptographic algorithms used are: DSA, RSA, HMAC and Diffie-Hellman.

In terms of web security, SAML is used for Single Sign-on and for issuing SAML-assertions to Service Providers. This is where information concerning the Assurance Level reached in an authentication is communicated. OIDC is used for redirects and the tokens issued and handled between Brokers and Service Providers in the system.¹⁴

7 Security analysis

7.1 The term "UserID"

Having been used to the terms Username and Password for online accounts, it is confusing to see the term UserID used to mean something private, since it contains the word "user", while usernames for other services are not sensitive information. This is even more problematic when taking into account that old usernames from NemID are proposed as UserIDs for MitID (which has a different threat-model entirely), and that the process of creating a UserID doesn't remove options such as just your first name or just your last name. Furthermore the UserID is visible when entered into the system - instead of *** like passwords - making for easy Over-the-shoulder attacks, and it is also visible when logged into MitID.dk. However, there are certain guidelines for choosing a good UserID,[18] although these are not enforced by the system.

7.2 The UserID is case-insensitive

This means that the string MyName is identical to MYNAME and myname. This drastically limits the amount of UserIDs an attacker has to check. For a UserID of length 5, which is the minimum, and 60 different input options, this would equate to 60^5 options, which is easy to brute force as long as it's done locally.

7.3 Man-in-the-middle

It seems possible for an attacker to create a web page that looks exactly like that of a specific bank. Then it's possible to lure users into trying to log into MitID (on a fake website), and the attacker would simply forward the request to the real MitID, and since the user would not know that they are on a fake page, since they would be prompted to swipe in the app, and the app would display the correct name of the bank, they might swipe, letting the attacker into their bank-accounts.

7.4 Over the shoulder

Logging into your MitID-account on MitID.dk shows a home-page where your UserID is shown. Since this is to be kept secret, it's an issue that anyone who can see your

¹⁴For more information on OIDC, an extension of OAuth 2.0, and SAML, see [1] and [10].

screen can simply get your UserID fast and easy. Even worse: any time you try to authenticate with MitID, your UserID is visible, giving anyone in your vicinity the option to see it. However, this does not grant any kind of access to their identity, since their phone and their pin-code to the app would be needed for that.

7.5 Denial of Service

Since starting a Login session in MitID only requires a UserID, if we know someone's UserID, we can start login sessions on their behalf. Doing this will end in one of five outcomes.

1. The login will hang for 5 minutes before it times out
2. The login is accepted in the app
3. Two active login sessions have been started simultaneously with the consequence that both have been canceled, and the user's app being temporarily disabled for 30 seconds
4. The user's app is already disabled (inside the 30 second window of "temporarily disabled")
5. "Something went wrong"

With these in mind, it is possible for an attacker to make a script that always keeps an active login session for a given user, meaning that when the user wants to log into MitID, the user will probably hit case 3, making the second login session and forcing both of them to be canceled. The attacker's goal is then to keep active login sessions for as long as possible. We have made such a script, and it works. We can effectively keep users out of MitID as long as we know their UserID.

Now, other identification measures exist, such as the code-viewer and the chip. These are even easier to do Denial of Service attacks against. Knowing the UserID, the attacker starts a login session, chooses the option to authenticate with the code-viewer and is immediately prompted for a password. After three wrong guesses of the password, the user is blocked from using their password and code-viewer for an hour. After having waited an hour, the attacker once again starts a login session, makes three wrong guesses with the consequence that the user's password has now been temporarily suspended and can only be reactivated by calling MitID or visiting the local *Borgerservice* in person, having either your passport or your driver's license, along with certain private knowledge, to prove your identity.

This makes for three possible cases:

1. User has only the app
2. User has only the code-viewer or code-reader
3. User has both

The chip, in this situation, does not make a difference, since it can only authenticate on Assurance Level Low without needing any of the other measures. Authenticating on this level shouldn't be enough to change a UserID, judging from how Assurance Level Low is described in [6] and [11], and especially the notion of Safe Authentication in [4].

The first two cases have been handled above by the script and the wrong guesses for the password, respectively. The third case simply does both. It first suspends the password for an hour then lets the script run. After an hour, the password can be suspended entirely, and the script can keep running, periodically checking for the option to type in the password, since the real user might have reactivated it by contacting MitID, in which case the process simply starts over. In conclusion: An attacker can keep any user out of MitID on one condition: Knowing their UserID. This goes against the rules in NSIS, which state that for Assurance Level Low: "Measures must be put in place to ensure that Electronic Identification Means are not unduly revoked or suspended in an attempt to deny access of a legitimate person." [6, page 18] Since each Assurance Level necessitates that all requirements of the lower Assurance Levels have been met, this can be understood to mean that at the moment, the current MitID solution does not even reach Assurance Level Low, no matter what Identification Means are used.¹⁵ It is important to note, however, that the main concern of this section of NSIS is The Revoking of Electronic Identification Means, so a Denial of Service might not fall into this specific category, meaning that the system as a whole still reaches the higher security levels, even though DoS attacks are a real threat.

7.6 User-enumeration and dictionary attacks

Making a login session in MitID with a wrong UserID actually tells you: "UserID doesn't exist". But when a correct UserID is entered, it starts a valid login session. This means that we can do User-enumeration attacks. We have successfully made a script that goes through all of the legal first names in Denmark, and we have found a success-rate of about 5-6 percent for uncommon names, and about 10 percent of common danish names are currently used as UserIDs in MitID. We have not tried adding one or two letters or even numbers before and after the name, and we haven't tried last names at all. This simply proves that finding UserIDs is possible, since we have complete access to an oracle that gives us the needed information. These kinds of attacks can either be done using the login-form on MitID.dk or any other entrance-point, or it can be done using the "change UserID" after having logged into MitID, or at enrollment. These all give information about available and used UserIDs.

As a side note: it might be the case the MitID has simply made template-accounts for several different UserIDs such as "aaaaa" and some of the legal first names, and that no actual user has these UserIDs.

¹⁵See 12.8 for the requirements that are not fulfilled in the current MitID solution

7.7 MitID as an oracle

When trying to start a login session for a given UserID, MitID will tell you one of a few things: The ones listed in the "Denial of Service" section, and one more option, namely the "No identification measure has been connected to this account". Knowing that a user creating a MitID account has to choose an identification measure when creating the account, this can only mean one thing: The user is waiting for a physical device to be sent to them, and they are also not using the app at all, since they would have then been forced to activate it during enrollment. This information can then be used to do Social Engineering attacks.

7.8 Social Engineering attacks

Having access to information about the user, such as their UserID (assuming the attacker has done a dictionary attack on uncommon names), or even that they are expecting to receive a physical device soon, it is easier to impersonate an employee of Nets or MitID and convince users to either let the attacker into their MitID or even set up other identification measures that the attacker owns, such that full access is given to the attacker. Dictionary attacks become quite easy when web pages such as *Krak.dk* and *Degulesider.dk* exist.¹⁶ Furthermore, the attacker is able to start two simultaneous login sessions for the user, resulting in an SMS being sent to their phone. This further strengthens the credibility of the attacker as an employee of Nets or MitID.

7.9 Code-viewer and Code-reader getting desynchronized

I have had my Code-viewer for two weeks, and I have already experienced that a one-time code was invalid and I have therefore had it re-synchronized by MitID support. These devices are supposed to last 10 years. Having to call MitID support often to re-synchronize them, should not be necessary.

8 Recommendations to improve the security and design in MitID

We have disclosed the above information to MitID through a contact and are hoping to see coming changes to the system in order to fix the issues mentioned.

First of all, the main point in our disclosure was to suggest bringing back passwords as we know them from NemID. This would be a real solution to most, if not all, of the issues discussed above. Secondly, it might be a good idea to implement a 1-second delay before the user can swipe in the app, since this would force the user to at least take a glance at the information displayed, before swiping. If, for some reason, passwords are not an option, the way to mitigate certain attacks would be to look through all incoming traffic, apply certain heuristics, and to treat requests from the same IP-address with ever-increasing response times. An attacker would

¹⁶These are databases of names and phone numbers.

eventually be able to get around these, but this would mitigate several simple attacks and demand more investment from potential attackers. It is important to mention, however, that all other solutions than bringing back the password, are, in fact, not real solutions to the problem. They would simply delay the attacks, not properly prevent them.

9 Access to information concerning MitID

Having contacted the Agency for Digitisation ("Digitaliseringsstyrelsen") and Nets yielded no information regarding the cryptography behind MitID, even though it was publicly available information that I later found in the Technical reference from one of the six brokers. In Denmark, asking for publicly available information should be done through *Aktindsigt* [14], which is a legal right to gain access to information on a specified topic. In stark contrast to this, a new law on MitID has been made, that prevents people from gaining knowledge about the technical and security-specific details in the MitID-solution. [4, chapter 14] This will prevent any research concerning the cryptography in MitID, unless NDAs are involved.

10 Future work

As of now the project has taken a turn towards the user-minded and front-end minded. This has led me to ask some researchers in the field of HCI if we could do a future project where we could make a study concerning how people actually interact with MitID and the MitID app. Questions like:

- Do users choose complex or simple UserIDs? (following the guidance on making UserIDs)
- Do users check their app to see if the login-request is for the correct website, or do they simply accept, expecting it to be correct?
- Do users know that their UserID is supposed to be secret?

These questions cannot be answered solely in the field of cryptography, so a joint project between Crypto and HCI would be fitting.

11 Acknowledgements

I thank Sophia Yakoubov, Diego F. Aranha, Mathias Hall-Andersen and Ivan Damgård at Aarhus University for useful discussions around the topic in preparation to this report. I also thank Morten Storm Petersen for his help in understanding the MitID terminology and for answering a lot of questions. Lastly, I thank Henrik Moltke for helping me find the MitID Whitepaper, which was pivotal in figuring out key elements of the system.

References

- [1] OktaDev (2018). OAuth 2.0 and OpenID Connect (in plain English), <https://www.youtube.com/watch?v=996OieXHze0>, accessed December 2021.
- [2] Version 2 (2019). NemID-papkortet på vej til at vige pladsen for en app. Published at <https://www.version2.dk/artikel/nemid-papkortet-paa-vej-at-vige-pladsen-app-utilgivelig-slaekkelse-sikkerheden-1089380>, accessed December 2021.
- [3] Signaturgruppen (2021). Technical reference for service providers, version 1.2.2. Published at https://broker.signaturgruppen.dk/application/files/7016/3852/2907/Nets_eID_Broker_Technical_Reference_v._1.2.2.pdf, accessed December 2021.
- [4] Lov om MitID og NemLog-in (May 2021). Published at <https://www.retsinformation.dk/eli/lta/2021/783>, accessed December 2021.
- [5] MitID Hvidpapir: Om MitID baggrund (2021). Published at https://digst.dk/media/24710/om-mitid-baggrund-whitepaper_webtilgaengeligt-version.pdf, accessed December 2021.
- [6] National Standard for Identity Assurance Levels (NSIS), version 2.0.1a. Published at <https://digst.dk/media/24697/nsis-engelsk-version-201a.pdf>, accessed December 2021.
- [7] Vejledning til National Standard for Identiteters Sikringsniveauer (NSIS), Version 2.2 (2021). Published at <https://digst.dk/media/24673/vejledning-til-national-standard-for-identiteters-sikringsniveauer-nsis-version-22.pdf>, accessed 2021.
- [8] Digst.dk NSIS-standarden. This contains all of the documents related to NSIS. <https://digst.dk/it-loesninger/nemlog-in/anvendelse/nsis-standarden/>
- [9] Vilkår og betingelser for MitID, v1.2 - Terms and conditions. Published at https://www.mitid.dk/media/yfeisft0/vilkaar_og_betingelser_for_mitid_og_fysiske_identifikationsmidler_v1_2.pdf, accessed December 2021.
- [10] SAML 2.0 Technical Overview - Youtube video. <https://www.youtube.com/watch?v=SvppXbpv-5k>, accessed December 2021.
- [11] Vejledning til valg af NSIS Sikringsniveau for tjenesteudbydere, version 2.0.2. Published at <https://digst.dk/media/21945/vejledning-til-valg-af-sikringsniveau-for-tjenesteudbydere-202.pdf>, accessed December 2021.
- [12] Nye fremtidsudsigter for digital identifikation Published at <https://digst.dk/it-loesninger/mitid/nyheder-om-mitid/tidligere-ny>

heder/fremtidens-digitale-infrastruktur/, accessed December 2021.

- [13] **MitID projektet** <https://digst.dk/it-loesninger/nemlog-in/om-loesningen/aendring-i-funktionaliteter/implementeringssite/infrastrukturbeskrivelse/mitid-projektet/>, accessed December 2021.
- [14] **Sådan anmoder du om aktindsigt.** <https://journalistforbundet.dk/sadan-anmoder-du-om-aktindsigt>, accessed December 2021.
- [15] **Spørgsmål og svar om MitID.** <https://digst.dk/it-loesninger/mitid/spoergsmaal-og-svar-om-mitid/>, accessed December 2021.
- [16] **Sådan får du MitID app - Youtube video.** <https://www.youtube.com/watch?v=mX-qIcysd48>, accessed December 2021.
- [17] **Sådan får du MitID kodeviser eller MitID kodeoplæser - Youtube video.** <https://www.youtube.com/watch?v=A9tAAgy19wo>, accessed December 2021.
- [18] **MitID bruger-ID.** <https://www.mitid.dk/hjaelp/hjaelpeunivers/mitid-bruger-id/?language=da-dk>, accessed December 2021.
- [19] **OIOSAML Web SSO profile 3.0.1. Published at** <https://digst.dk/media/21892/oiosaml-web-ss-profile-301.pdf>, accessed December 2021.

12 Appendix

12.1 No password

From a Questions and Answers section on MitID from DR.dk.

Har man ikke længere et kodeord tilknyttet sit brugernavn, som man havde det på nemid?

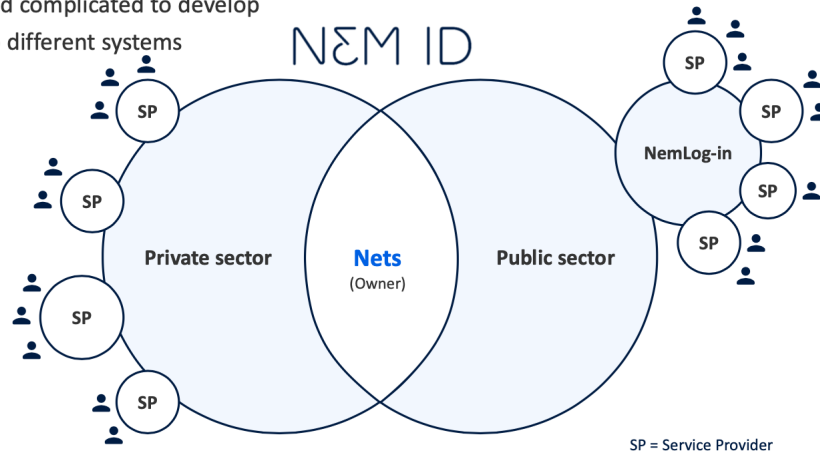


Adam Lebech

Man har stadig et centralt valideret kodeord, men det er flyttet ind i appen, hvor det er bedre beskyttet. Det består af 6 cifre, som man skal bruge i appen.

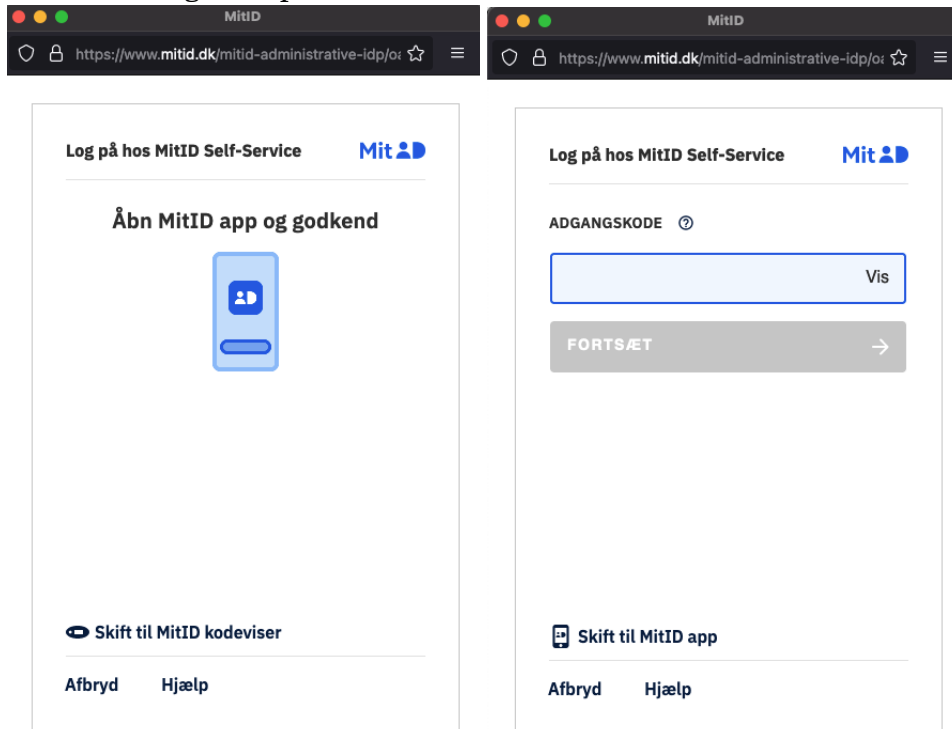
12.2 The structure of NemID

- ✓ Functions fine, but is not future-proof
- ✗ Is expensive and complicated to develop
- ✗ Consists of two different systems



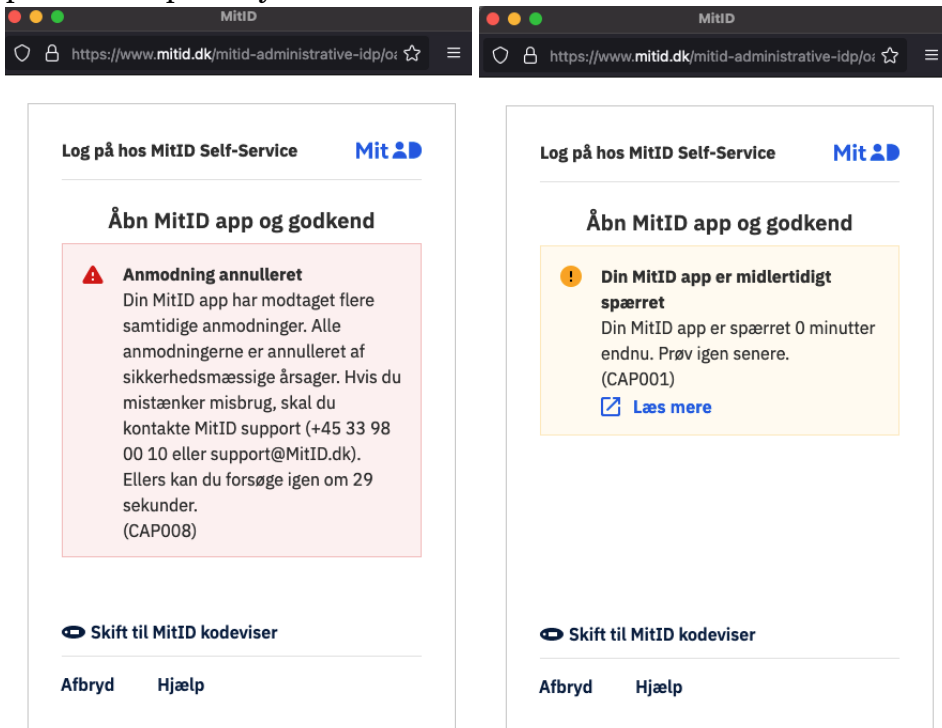
12.3 API response

The first screenshot below shows the login-session waiting for the app, and the second is waiting for a password in order to then use a Code-viewer.

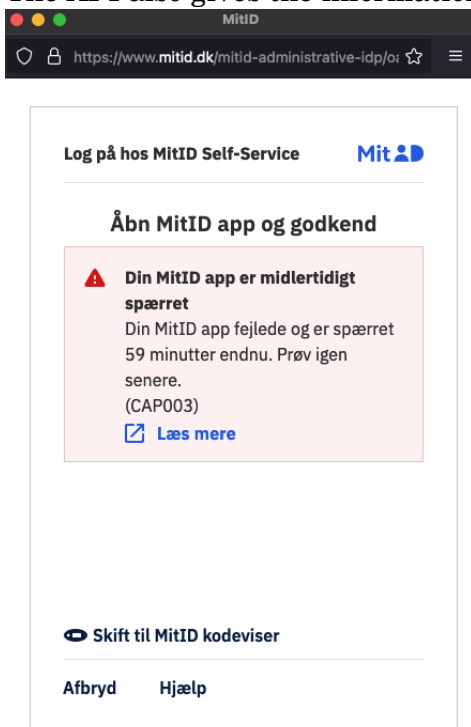


The first screenshot below shows the API's response to two active login sessions at the same time, ending in the app being suspended temporarily, and the second

is the response when trying to authenticate during the time when the app is suspended temporarily.

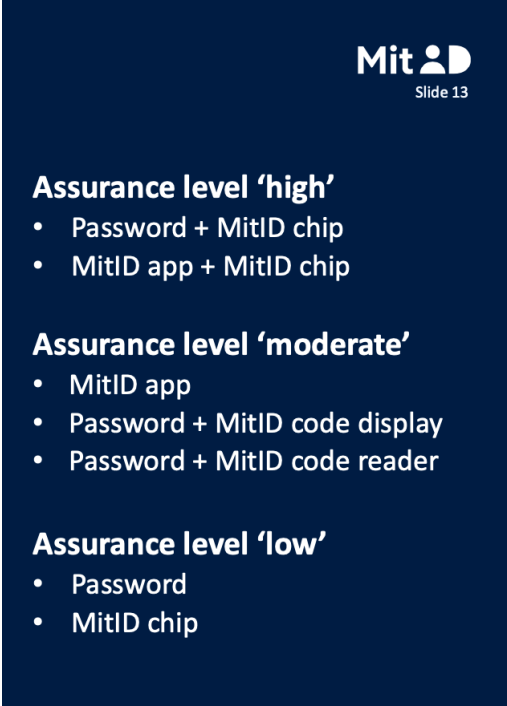


The API also gives the information that the app is suspended, if this is the case:



12.4 Assurance Levels

"Moderate" is interchangeable with "Substantial"



MitID
Slide 13

Assurance level 'high'

- Password + MitID chip
- MitID app + MitID chip

Assurance level 'moderate'

- MitID app
- Password + MitID code display
- Password + MitID code reader

Assurance level 'low'

- Password
- MitID chip

12.5 Responsibilities of Brokers and Service Providers

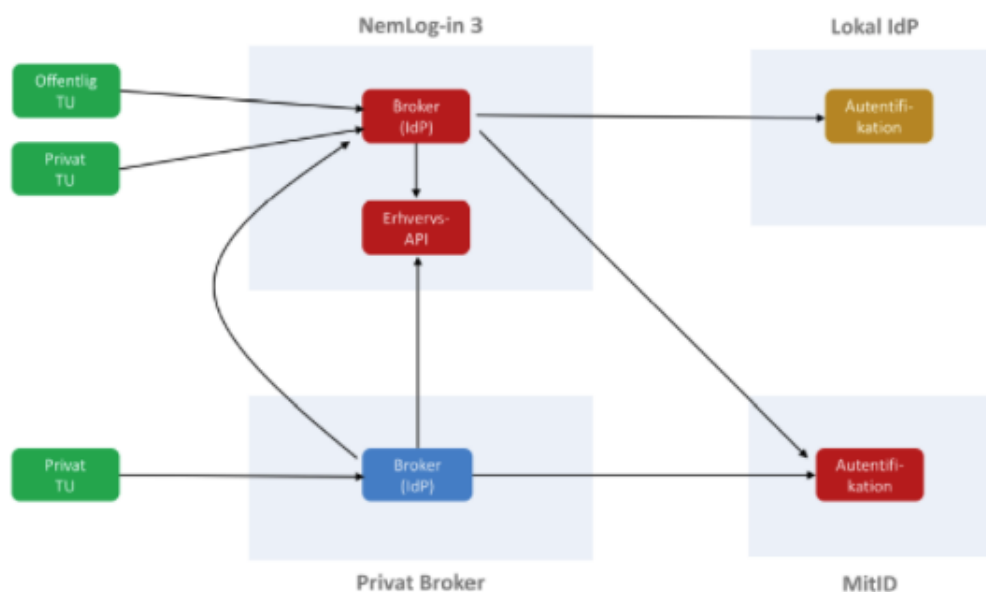
Ansvar for Identitetsbroker (fx NemLog-in):

- Autenticere bruger.
- Udstede adgangsbillet med opnået NSIS Sikringsniveau.

Ansvar for tjenesteudbyder:

- Vurdere krævet Sikringsniveau på baggrund af risikovurdering.
- Verificere SAML adgangsbillet for brugere udstedt af Identitetsbroker.
- Sammenligne opnået NSIS sikringsniveau i adgangsbillet med krævet sikringsniveau.
- Håndhæve adgangspolitik og give korrekt adgang til data og funktioner på baggrund af opnået Sikringsniveau og øvrige forhold (herunder autorisationer). Dette omfatter bl.a. at afvise brugere med for lavt Sikringsniveau.

12.6 Logical structure of MitID



Figur 1: Logisk arkitektur

12.7 Definition of Broker

The definition of Identity-brokers from *Guide to NSIS*, which is in danish: "Indledningsvis er det relevant at præcisere, hvad der menes med en Identitetsbroker. I kontekst af NSIS menes en tjeneste, som videreformidler en autentifikation til en tredjepart ved at udstede og signere et såkaldt Security Token (en 'billet') for en elektronisk Identitet. Disse benævnes i nogen sammenhænge for 'Identity Providers' eller 'Security Token Services'¹⁰, og der findes en række internationale standarder (visse med tilhørende danske profiler), som regulerer deres snitflader som fx SAML, WS-Trust og OpenID Connect. Et konkret eksempel er NemLog-in løsningen, der udsteder SAML Assertions til offentlige tjenesteudbydere, når borgere eller medarbejdere tilgår tjenesten. Det er med andre ord attributterne i SAML Assertion, der beskriver den elektroniske Identitet, og tjenesten ser herved ikke det bagvedliggende Elektronisk Identifikationsmiddel men kun attributter og et formidlet Sikringsniveau."

12.8 NSIS demand

From NSIS section 3.2.3 in english:

Assurance Level	Requirements
Low	<ol style="list-style-type: none"> 1. It is possible to suspend/or revoke an Electronic Identification Means in a timely and effective manner. 2. Measures must be put in place to ensure that Electronic Identification Means are not unduly revoked or suspended in an attempt to deny access of a legitimate person.

From the danish guide to NSIS:

Niveau: Lav	Krav: 2) Der skal etableres foranstaltninger, som sikrer mod, at Elektroniske Identifikationsmidler spærres eller suspenderes uretmæssigt i et forsøg på at lukke en legitim Persons adgang.
Vejledning: I implementeringen af spærrefunktionen er det relevant at tage højde for risikoen for <i>denial-of-service</i> angreb, hvor uvedkommende forsøger at spærre andres Elektroniske Identifikationsmidler – fx ved at etablere mekanismer, der gør det vanskeligt at massespærre Elektroniske Identifikationsmidler samt kontroller der sikrer, at det er rette vedkommende (eller anden autoriseret part), der spærres sit Elektroniske Identifikationsmiddel.	