Installationshandbuch Authenticator

Das Installationshandbuch erläutert Schritt für Schritt, wie der Authenticator als Anwendung auf einem Computer installiert werden kann. Es werden die Voraussetzungen für die Installation genannt und die Konfigurationsmöglichkeiten für die Inbetriebnahme erklärt.

Inhaltsverzeichnis

Voraussetzungen

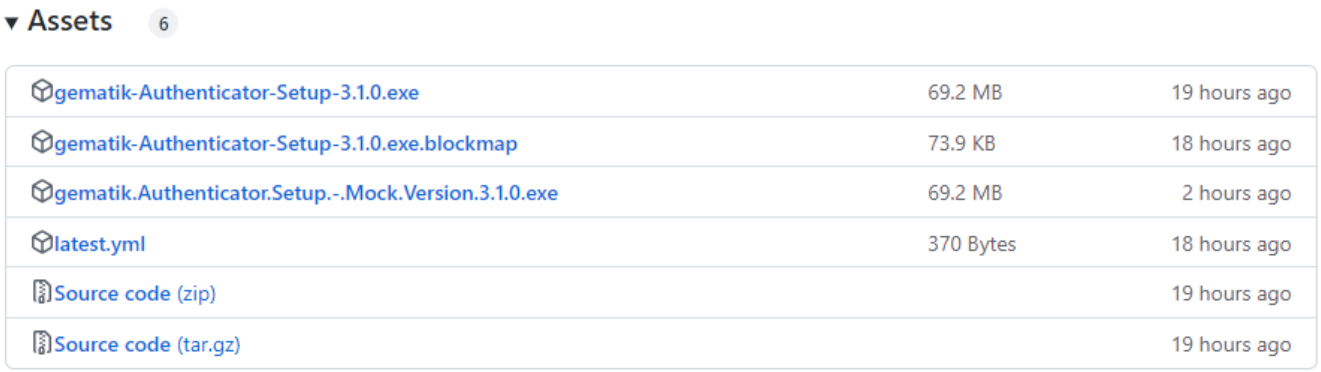
- mind. Microsoft Windows 10
  (Seit Authenticator Version 4.0.0 wird Electron in der Version 23 verwendet. In dieser Version wird seither Windows 7, 8/8.1 nicht mehr unterstützt).
- ab Windows Server 2016
- Freier Speicherplatz (ca. 150MB)
- Netzwerk- bzw. Internetverbindung
- Konnektor - RISE, Secunet, Koco
- Kartenterminals - Cherry, Ingenico
- Internet-Browser
  - Chrome (aktuelle Version)
  - Firefox (aktuelle Version)
  - Opera (aktuelle Version)
  - Edge (aktuelle Version)
  - Es wird **kein** **Internet Explorer** unterstützt
- Hinweis: Für eine bessere Benutzerfreundlichkeit sollte der vom Nutzer verwendete Browser dem im System hinterlegten Default-Browser entsprechen.
- Administratorrechte für die Installation sowie erstmaligem Start (aufgrund Setzen von Firewallregeln)

Beschaffung der Installationsdatei

Die Installationsdatei wird auf der gematik Cloud und auf GitHub veröffentlicht (Den Download-Ordner für den Authenticator finden Sie unter gematik SharePoint). Dort können Sie sowohl die aktuelle Version als auch frühere Versionen der Installationsdatei herunterladen.

Innerhalb des folgenden Links können Sie die jeweiligen Assets der Release-Version auffinden und nutzen: https://github.com/gematik/app-Authenticator/releases

Bsp. anhand Version 3.1.0:



Installation

1. Laden Sie die Installationsdatei herunter. (siehe oben, „Beschaffung der Installationsdatei“)
2. Führen Sie die Installationsdatei aus. Um das Programm für alle Nutzerkonten des Computers zu installieren, benötigen Sie die Administratorberechtigung
3. Folgen Sie den Aufforderungen des Installationsprogramms
4. Nachdem das Installationsprogramm erfolgreich durchgelaufen ist, wird eine entsprechende Meldung im Dialogfenster angezeigt
5. Konfigurieren Sie den Authenticator. (siehe unten, „Konfiguration“)

Die Installation und Ersteinrichtung sind nun vollständig abgeschlossen und der Authenticator Client ist auf Ihrem Computerarbeitsplatz einsatzbereit.

Der Authenticator Client befindet sich nach der Installation standardmäßig im Ordner "C:\\Program Files\gematik Authenticator".

Mit Doppelklick auf der Datei "gematik Authenticator.exe" wird der Authenticator Client gestartet.

Die notwendigen Resource-Dateien sind unter dem Ordner "C:\\Program Files\gematik Authenticator\resources" abgelegt. Unter certs-konnektor befinden sich die CA-Zertifikate für die sichere Verbindung mit dem Konnektor und unter certs-idp die des IDP. Ggf. müssen Sie diese um z. B. notwendige Proxy Zertifikate **ergänzen**.

Funktionsweise

Die genaue Funktionsweise bzw. der korrekte Ablauf zur Nutzung des Authenticators können Sie innerhalb unseres Fachportals einsehen.

Korrekte Nutzung der Karten

Der Authenticator unterstützt grundsätzlich die Authentifizierung mit HBA- sowie SMC-B-Karten. Allerdings entscheidet nicht der Authenticator, welche Karte gesteckt werden soll, sondern die genutzte Fachanwendung selbst.

Die Fachanwendung gibt grundsätzlich immer vor, welche Karte für den Authentifizierungsprozess mittels Authenticator gesteckt werden soll.

Bei Fragen hierzu wenden Sie sich bitte an den Support der Fachanwendung.

Multi-SMC-B Auswahl:

Sollten sich in den Konnektor-Terminals mehrere SMC-Bs befinden, erfolgt ab Version 4.0.0 des Authenticators keine Fehlermeldung mehr, sondern es erscheint ein Auswahldialog mit den vorhandenen SMC-Bs und detaillierten Informationen wie z. B. Kartenhalter und iccsn. Der Benutzer hat nun die Möglichkeit, eine SMC-B Karte für den weiteren Auth.-Flow auszuwählen oder diesen durch „Abbrechen“ komplett zu beenden.

Konfiguration

Es gibt mehrere Möglichkeiten das Programm zu konfigurieren. Die Konfigurationen unterscheiden sich für drei Szenarien:

1. Szenario: Einzelplatz/Standalone-Umgebung mit einer lokalen Konfigurationsdatei. Siehe auch: Konfigurationsvideos zum Authenticator
2. Szenario: Client in einer Remote-Umgebung (Citrix oder Windows-Server ab 2016)
3. Szenario: Standalone-Umgebung mit einem zentralen Konfigurationspfad

Hinweis: Eine ausführliche grafische Darstellung der drei Szenarien finden Sie unter: Installations-Szenarien

Szenario 1a: Konfiguration Einzelplatz/Standalone-Umgebung mittels Benutzeroberfläche

Siehe hierzu auch: Konfigurationsvideos zum Authenticator

1. Öffnen Sie das Programm "gematik Authenticator" und wählen Sie den Menüpunkt „Einstellungen“
2. Füllen Sie die erforderlichen Felder aus. Für weitere Informationen bezüglich der Felder werfen Sie einen Blick auf den Menüpunkt „Erforderliche Einstellungen“.
3. Nachdem Sie die erforderlichen Felder ausgefüllt haben, speichern Sie die Änderungen mit Klick auf den Button „Speichern“
4. Mit dem Punkt "Funktionstest (inkl. Speichern)" können Sie die Korrektheit der Einstellungen überprüfen

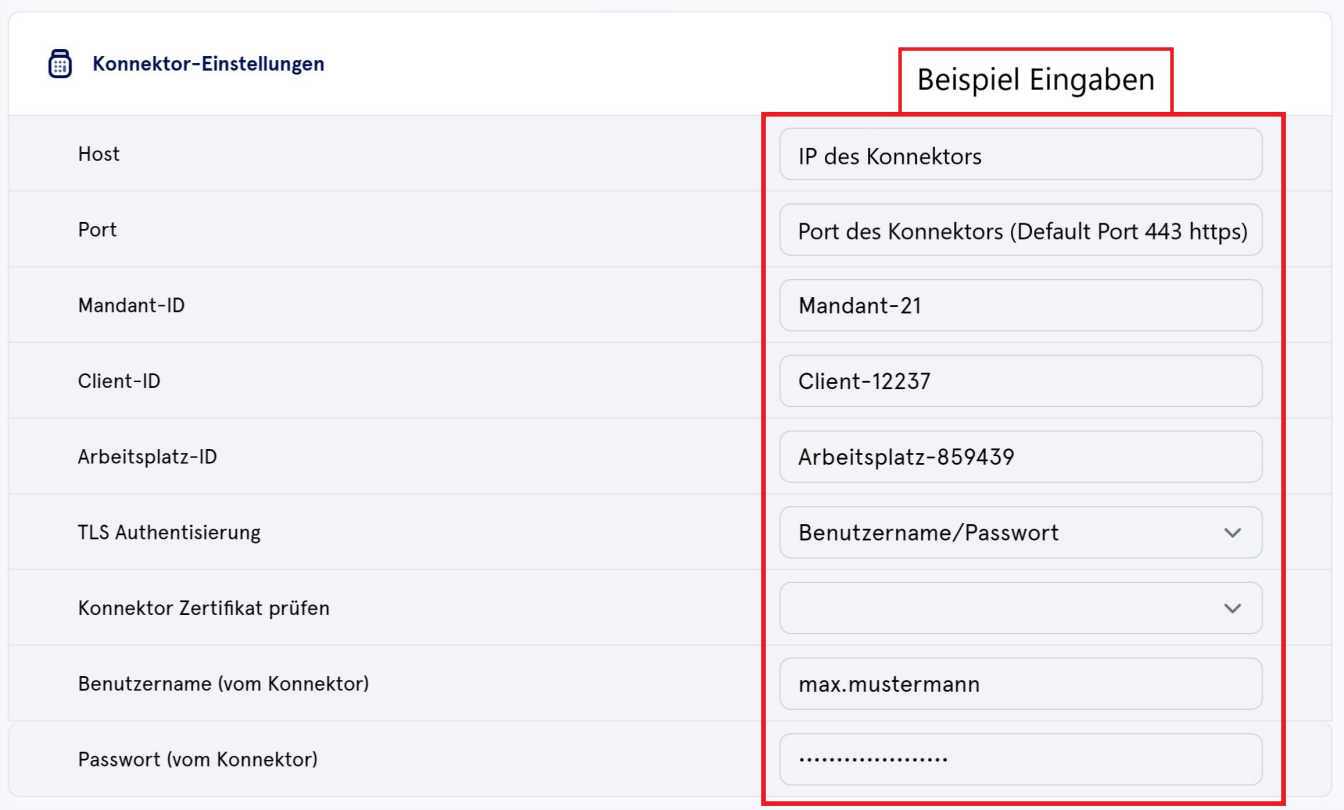


Abbildung 1: Die Konfigurationsoberfläche des gematik Authenticator

Hinweis: Wenn Sie die TLS-Authentisierung-**Zertifikat** nutzen möchten, achten Sie bitte darauf, dass diese im PEM-Format oder als P12-Datei vorliegen.

Möchten Sie eine P12-Datei nutzen, muss diese ein gültiges RSA Zertifikat enthalten, da es ansonsten zu einem Fehler kommt. Halten Sie hierfür das dazugehörige Passwort bereit.

Sie können vorhandene P12-Zertifikate auch in ein PEM-Format umwandeln - Eine Möglichkeit, wie die Datei in das PEM-Format exportiert werden kann, finden Sie hier: Umwandlung einer p12-Datei in PEM-Format.

Szenario 1b: Konfiguration Einzelplatz/Standalone-Umgebung mittels Konfigurationsdatei

1. Erstellen Sie eine Konfigurationsdatei im Nutzerverzeichnis (C:/Users/{Nutzerverzeichnis}/AppData/Local/gematik Authenticator/config.json)

Beispiel einer Konfigurationsdatei **(ohne Nutzung des Credential Managers):**

```
{
  "checkUpdatesAutomatically": true,
  "connector": {
    "contextParameter": {
      "mandantId": "Mandant-x",
      "clientId": "Client-System-x",
      "workplaceId": "Arbeitsplatz-x"
    },
    "entryOption": {
      "hostname": "127.0.0.1",
      "port": 443,
      "keyFile": "c:/",
      "certFile": "c:/",
      "pfxFile": "c:/",
      "pfxPassword": "pfx password",
      "username": "authenticator",
      "password": "password"
    },
    "tlsAuthType": "BasicAuth"
  },

   "proxy": {
    "proxySettingsType": "BasicAuth",
    "useOsSettings": true,
    "proxySettingsPassword": "pass123",
    "proxySettingsUsername": "user123"
  }
}
```

Beispiel einer Konfigurationsdatei **(mit genutztem Credential Manager (empfohlen)):**

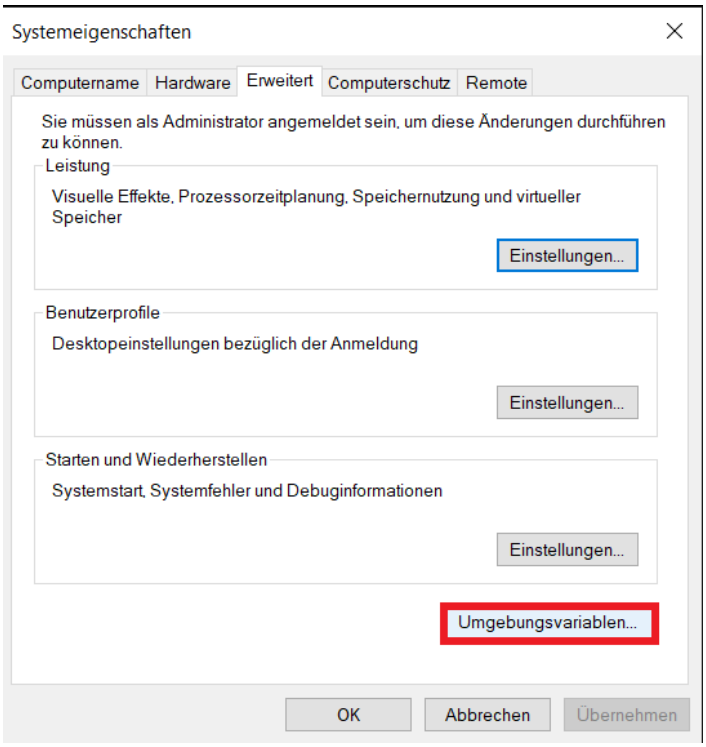
```
{
  "checkUpdatesAutomatically": true,
  "connector": {
    "contextParameter": {
      "mandantId": "Mandant-x",
      "clientId": "Client-System-x",
      "workplaceId": "Arbeitsplatz-x"
    },
    "entryOption": {
      "hostname": "127.0.0.1",
      "port": 443,
      "keyFile": "c:/",
      "certFile": "c:/",
      "pfxFile": "c:/"
    },
    "tlsAuthType": "BasicAuth"
  },

 "proxy": {
    "proxySettingsType": "BasicAuth",
     "useOsSettings": true
  }
}
```

Wie im obigen Konfigurationsbeispiel ersichtlich wurde, werden in der Anwendung des Credential Managers sensible Inhalte aus der Datei config.json herausgefiltert und in den Credential Manager übertragen. Dies ist ausschließlich für Szenarien gültig, die ab der Version 4.8.0 stattfinden, bei denen die Daten auf diese Weise gesichert werden.

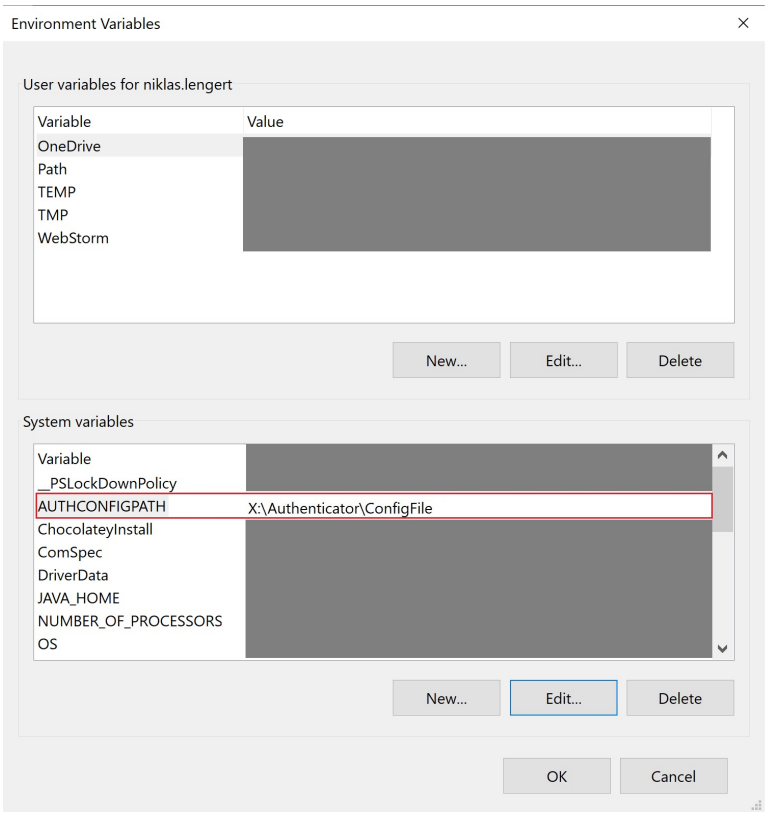
Szenario 2a: Konfiguration des Clients in einer Remote-Umgebung (Citrix und Windows Server (ab 2016))

Für dieses Szenario muss neben der in Szenario 1 erwähnten config.json in den Umgebungsvariablen **AUTHCONFIGPATH** angelegt werden (**CLIENTNAME** wird vom Remote-System (Citrix bzw. Windows Server) gesetzt).

- Öffnen Sie dafür die Systemeinstellungen des Clients und anschließend die Umgebungsvariablen:



- Fügen Sie anschließend die Variable hinzu, wie es im Beispiel zu sehen ist (***Hinweis:*** ab der Version 3.1.0 werden AUTHCONFIGPATH und CLIENTNAME nach dem Start des Authenticators, sofern sich zur Laufzeit Änderungen an diesen ergeben, aus den zugehörigen Einträgen in der Registry aktualisiert)
- Für den Variablen-Wert des AUTHCONFIGPATH wählen Sie bitte den Pfad, an dem sich der Ordner befindet, in dem die verschiedenen Ordner je Citrix- bzw. Windows-Server-Client (die eine passende config.json für diesen Client enthalten) abgelegt werden soll (im Beispiel ist es X:\Authenticator\ConfigFile)
- ***Hinweis:*** *die Namensgebung des Ablageordners ist hierbei Ihnen überlassen; wichtig ist, dass dieser Name je nach Umgebung* ***case sensitiv*** *ist und auf Groß- und Kleinschreibung geachtet werden muss und keine Variablen wie %PATH% enthalten darf.*
- Klicken Sie anschließend auf **OK**



- Nachdem die Umgebungsvariable gesetzt worden sind, müssen in dem unter **AUTHCONFIGPATH** hinterlegtem Pfad ein oder mehrere Ordner erstellt werden, deren Name den des möglichen Clients entsprechen und sich im Client in der von Citrix bzw. Windows Server gesetzten Umgebungsvariable **CLIENTNAME** wiederfindet.
- ***Hinweis:*** *für jeden CLIENTNAME muss ein separater Ordner in dem unter AUTHCONFIGPATH hinterlegtem Pfad angelegt werden, da sich in diesen die jeweiligen config.json Dateien befinden*
- In diesen **CLIENTNAME**-Ordnern werden die unterschiedlichen config.json Dateien abgespeichert. Hierfür kann die, wie in Szenario 1 beschrieben, durch den Authenticator erstellte config.json kopiert, ggf. angepasst und genutzt werden

Um zu testen, ob das Auslesen erfolgreich funktioniert hat, öffnen Sie die Authenticator-Log-Datei (`C:/Users/{Nutzerverzeichnis}/AppData/Local/Temp/authenticator-logging/authenticator-${datum}.log`) und prüfen Sie den dort aufgeführten AUTHCONFIGPATH und den CLIENTNAME.

Alternativ können Sie auch unter ihrem Explorer folgenden Pfad öffnen: %temp%

Somit gelangen Sie direkt in das locale tmp-Verzeichnis, in welchem das Authenticator-Log abgelegt wird.

Hinweis: *Ab Version 4.8.0 wird es keinen Fallback mehr auf den lokalen User-Pfad geben, sofern der Parameter AUTHCONFIGPATH in den Umgebungsvariablen richtig gesetzt wurde. Das bedeutet, dass wenn der Authenticator am angegebenen Speicherort keine gültige config.json Datei findet, die Einstellungsseite beim Öffnen des Authenticators leer ist und nicht mehr versucht wird, im lokalen User-Verzeichnis eine gültige config.json zu finden. Werden nun Einstellungen vorgenommen, werden diese auch in dem von Ihnen angegebenen Pfad (AUTHCONFIGPATH + CLIENTNAME) abgespeichert.*

Szenario 2b: Konfiguration des Clients in einer VM-Ware-Remote-Umgebung

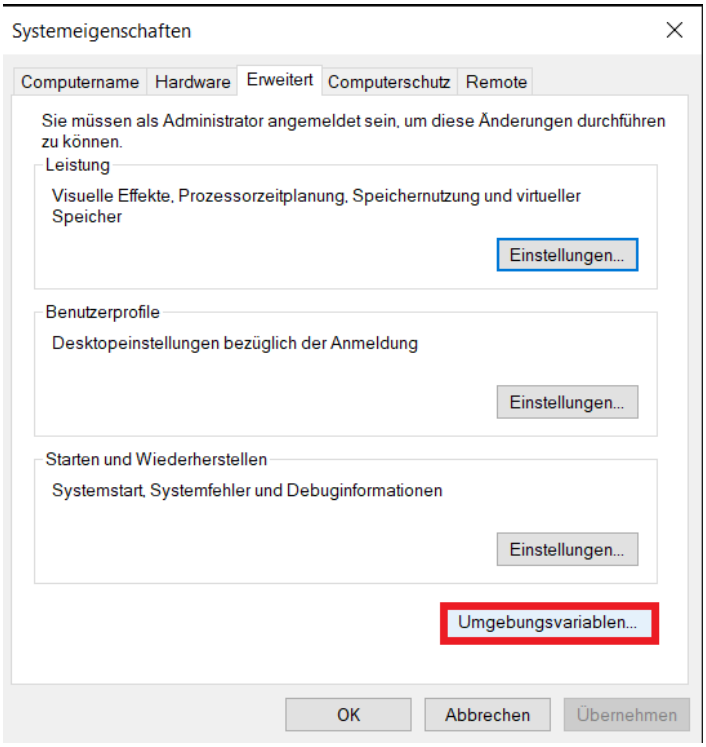
Ähnlich wie beim Szenario 2a (Citrix/Windows Server) wird für die remote VM-Ware-Konfiguration der Pfad für die Konfigurationsdatei zusammengesetzt. Anstatt des CLIENTNAME wird bei VM-Ware allerdings die Umgebungsvariable **VIEWCLIENT_MACHINE_NAME** verwendet.

Hinweis: *Ab Version 4.8.0 wird es keinen Fallback mehr auf den lokalen User-Pfad geben, sofern der Parameter AUTHCONFIGPATH in den Umgebungsvariablen richtig gesetzt wurde. Das bedeutet, dass wenn der Authenticator am angegebenen Speicherort keine gültige config.json Datei findet, die Einstellungsseite beim Öffnen des Authenticators leer ist und nicht mehr versucht wird, im lokalen User-Verzeichnis eine gültige config.json zu finden. Werden nun Einstellungen vorgenommen, werden diese auch in dem von Ihnen angegebenen Pfad (AUTHCONFIGPATH + VIEWCLIENT_MACHINE_NAME) abgespeichert.*

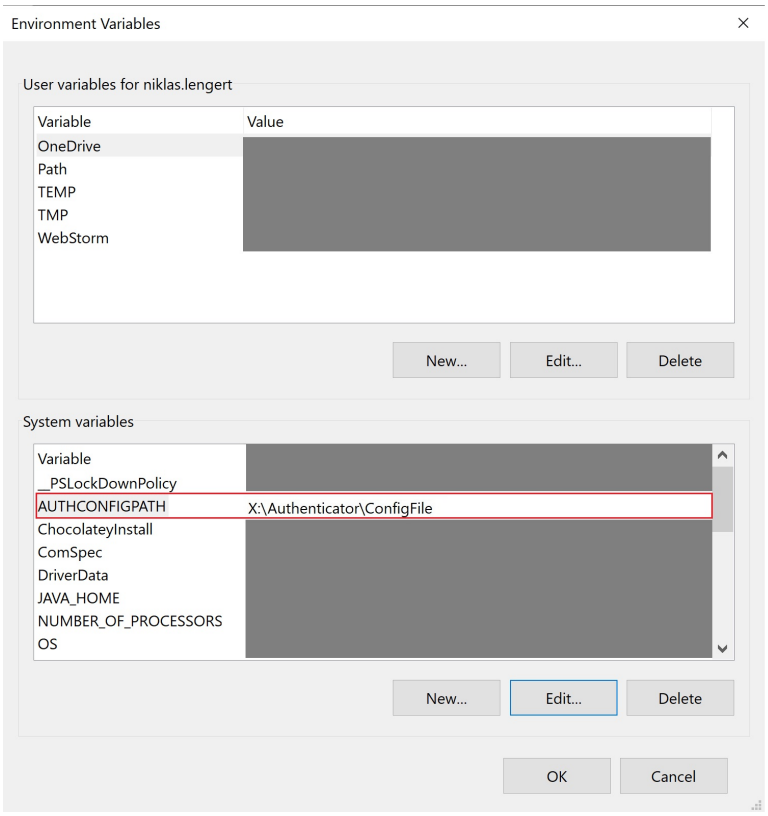
Szenario 3: Konfiguration einer oder mehrerer Standalone-Umgebungen mit einem zentralen Konfigurationspfad

Für dieses Szenario muss neben der in Szenario 1 erwähnten config.json in den Umgebungsvariablen die Variable **AUTHCONFIGPATH** je Computer angelegt werden.

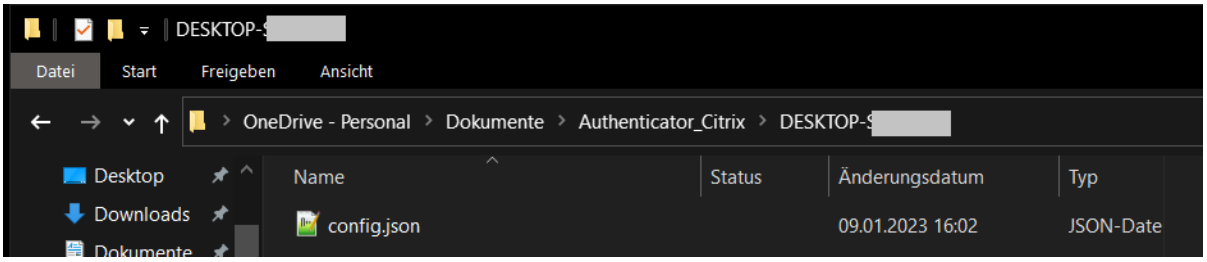
- Öffnen Sie dafür die Systemeinstellungen des Clients und anschließend die Umgebungsvariablen:



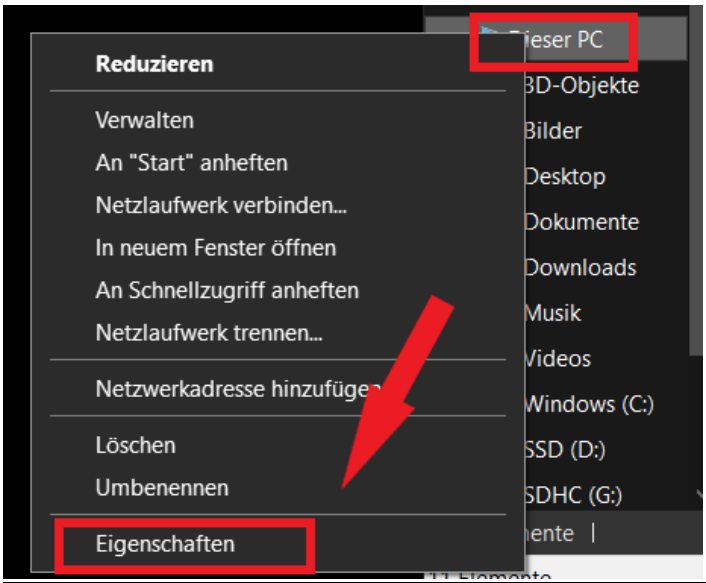
- Fügen Sie anschließend die Variable hinzu, wie es im Beispiel zu sehen ist
- Für den Variablen-Wert des AUTHCONFIGPATH nehmen Sie bitte den Pfad, an dem sich der bzw. die Ordner mit Namen COMPUTERNAME der möglichen Computer befinden, in dem die config.json je Computer abgelegt werden soll (im Beispiel ist es X:\Authenticator\ConfigFile)
- ***Hinweis:*** *die Namensgebung des Ablageordners ist hierbei Ihnen überlassen, wichtig ist, dass dieser Name je nach Umgebung* ***case sensitiv*** *ist und auf Groß- und Kleinschreibung geachtet werden muss und keine Variablen wie z.B. %PATH% enthalten darf.*
- Klicken Sie anschließend auf **OK**



- Nachdem die Umgebungsvariable gesetzt wurde, muss in dem unter **AUTHCONFIGPATH** hinterlegtem Pfad ein Ordner erstellt werden, deren Name dem des **COMPUTERNAME** entspricht
- Erstellen Sie nun in dem unter **AUTHCONFIGPATH** hinterlegtem Pfad einen Odner mit dem COMPUTERNAMEN (*wie im Beispiel zu sehen*)
- In diesem **COMPUTERNAME**-Ordner wird die jeweilige config.json Datei abgespeichert. Hierfür kann die, wie in Szenario 1 beschrieben, durch den Authenticator erstellte config.json kopiert, ggf. angepasst und genutzt werden
- ***Hinweis:*** *Ab Version 4.8.0 wird es keinen Fallback mehr auf den lokalen User-Pfad geben, sofern der Parameter AUTHCONFIGPATH in den Umgebungsvariablen richtig gesetzt wurde. Das bedeutet, dass wenn der Authenticator am angegebenen Speicherort keine gültige config.json Datei findet, die Einstellungsseite beim Öffnen des Authenticators leer ist und nicht mehr versucht wird, im lokalen User-Verzeichnis eine gültige config.json zu finden. Werden nun Einstellungen vorgenommen, werden diese auch in dem von Ihnen angegebenen Pfad (AUTHCONFIGPATH + COMPUTERNAME) abgespeichert.*



- ***Hinweis:*** den **COMPUTERNAME** (oder auch Gerätename) können Sie sich einfach über *Dieser PC - Rechtsklick - Eigenschaften* anzeigen lassen



Um zu testen, ob das Auslesen erfolgreich funktioniert hat, öffnen Sie die Authenticator-Log-Datei (`C:/Users/{Nutzerverzeichnis}/AppData/Local/Temp/authenticator-logging/authenticator-${datum}.log`) und prüfen Sie den dort aufgeführten AUTHCONFIGPATH und den COMPUTERNAME:

Alternativ können Sie auch unter ihrem Explorer folgenden Pfad öffnen: %temp%

Somit gelangen Sie direkt in das lokale tmp-Verzeichnis, in welchem das Authenticator-Log abgelegt wird.

```
0.475Z [info]: Listening localhost:39000
0.500Z [warn]: configPath:
0.500Z [warn]:   -  AUTHCONFIGPATH:C:\Users\[redacted]\OneDrive\Dokumente\Authenticator_Citrix
0.501Z [warn]:   -  CLIENTNAME:undefined
0.501Z [warn]:   -  COMPUTERNAME:DESKTOP-[redacted]
0.501Z [warn]:  - config über COMPUTERNAME
0.501Z [warn]:  - config.json path:C:\Users\[redacted]\OneDrive\Dokumente\Authenticator_Citrix\DESKTOP-[redacted]
```

- ***Wichtig für alle Szenarien:*** Die Umgebungsvariablen werden vom Authenticator beim Anwendungsstart und in Intervallen aus der **Registry** ausg elesen. Es werden hier **keine** via Skript gesetzten Umgebungsvariablen verwendet. Über den oben beschriebenen Weg werden die Umgebungsvariablen auch in der Registry gespiegelt.

Szenario 4: Default-Konfiguration (nur für Fachanwendungen mit SMC-B geeignet z. B. DEMIS)

In diesem Video zeigen wir einen Ansatz zur Umsetzung einer sogenannten Default-Config. Diese ist in erster Linie für Institute gedacht, die Fachanwendungen wie das DEMIS-Meldeportal und somit eine SMC-B Karte nutzen. Für Fachanwendungen, **die eine HBA benötigen, ist diese Art der Konfiguration ungeeignet.**

Der Gedanke bei dieser Umsetzung besteht darin, dass beispielsweise ein Institut, welches nicht nur mehrere Standorte vertritt, sondern ebenfalls über sehr viele Arbeitsplätze verfügt, die Möglichkeit erhält über "Standort-spezifische Pfade" eine Standardkonfiguration für den jeweiligen Standort zu hinterlegen. Für Arbeitsplätze, die eine gesonderte Konfiguration benötigen (beispielsweise wenn eine Fachanwendung zusätzlich einen HBA benötigt), bleibt der bereits bekannte Konfigurationsweg erhalten.

Was bedeutet das im Detail:

Jeder Standort sollte seine eigenen GPOs besitzen oder eine Unterscheidung in gemeinsamen GPOs durch Filter unterstützen.

In diese hinterlegen Sie dann wie gewohnt den AUTHCONFIGPATH je Standort, in dem der zentrale Netzwerkpfad zu den Konfigurations-Dateien des Authenticators hinterlegt ist.

Unter diesem Pfad erstellen Sie dann, wie ebenfalls bereits bekannt, arbeitsplatz-spezifische Ordner für die Arbeitsplätze, die eine individuelle Konfiguration benötigen.

Die Standardkonfiguration (Default-Config) legen Sie lediglich in dem im AUTHCONFIGPATH genannten Pfad ab.

Wird der Authenticator gestartet, dann überprüft dieser, ob ein passender arbeitsplatz-spezifischer Ordner vorhanden ist und sich eine config.json - Datei darin befindet. Findet er keinen, nutzt er automatisch die abgelegte Standardkonfiguration.

Kann der User diese einfach überschreiben? Nein, das ist nicht möglich. Ist ein arbeitsplatz-spezifischer Ordner angelegt, aber keine config.json-Datei hinterlegt und ein User nimmt Änderungen im Authenticator über die Einstellungsseite vor und möchte diese speichern, wird eine neue config.json-Datei in seinem arbeitsplatz-spezifischen Ordner abgelegt. Über ein Pop-Fenster wird der Nutzer zusätzlich darüber informiert. Der Authenticator nutzt dann diese soeben abgelegte config.json-Datei.

Was passiert, wenn kein arbeitsplatz-spezifischer Ordner vorhanden ist? Kann der User dann die Default-Konfig überschreiben? Nein, kann er nicht. Er kann aber auch nicht in seinem arbeitsplatz-spezifischen Ordner speichern, weil der Authenticator keine Ordner selbstständig erstellt.

Wie kann ich wieder die Standardkonfiguration nutzen? Löschen Sie dazu einfach die in Ihrem arbeitsplatz-spezifischen abgelegte conig.json-Datei.

Hinweis: *Dieses Feature ist erst ab Version 4.11.0 des Authenticators vorhanden*

Hinweis: *Eine detaillierte Grafik des Konzeptes der Funktionsweise kann zusätzlich hier eingesehen werden.*

Konfigurationsvideo für die Einrichtung einer Default-Config

Verwendung des Credential Managers

Ab Version 4.8.0 unterstützt der Authenticator auch die Verwendung des Credential Managers. Diese Funktion wurde implementiert, um zu verhindern, dass Passwörter und sensible Informationen im Klartext in der Konfigurationsdatei gespeichert werden. Anstatt die Konfigurationsdatei zu ersetzen, bietet dies einen hybriden Ansatz, in dem nur Passwörter im Credential Manager gespeichert werden.

***Achtung**: Der Credential Manager enthält Werte, die je User zugreifbar sind. D.h. es müssen die Credentials im Userkontext eingebracht werden, z.B. über ein Group Policy Object, das während der Anmeldung des Users ausgeführt wird.*

Im Credential Manager zu speichernde Werte

- Connector Basic Auth Benutzername
- Connector Basic Auth Passwort
- Proxy Basic Auth Benutzername
- Proxy Basic Auth Passwort
- P12-Zertifikat Passwort (der Benutzername wird standardmäßig als p12Password gespeichert)

Struktur

Die oben genannten Werte sollten im Credential Manager in folgender Struktur vorhanden sein:

- Gematik_Authenticator/Connector_BasicAuth ($Benutzername, $Passwort)
- Gematik_Authenticator/Proxy_BasicAuth ($Benutzername, $Passwort)
- Gematik_Authenticator/Connector_ClientCert_Password ('p12Password', $Passwort)
    - Da im P12-Zertifikat keine Benutzerinformationen enthalten sind, wird der Benutzername als p12Password gespeichert.

Hinweis: Für gewöhnlich benötigen Sie entweder "Gematik_Authenticator/Connector_BasicAuth" oder "Gematik_Authenticator /Connector_ClientCert_Password" für die erfolgreiche TLS Authentisierung gegenüber dem Konnektor, da nur eines aktiv im Konnektor konfiguriert ist. Nutzen Sie hingegen die TLS Authentifizierung mittels PEM-Dateien, benötigen Sie keines der beiden.

Anwendung

In der Standalone-Version des Authenticators wird dieser Prozess automatisch über die Konfiguration über die GUI durchgeführt, sobald die Einstellungen gespeichert sind. Sensible Daten werden im Credential Manager und nicht sensible Daten in der Datei config.json gespeichert.

Bei Verwendung einer zentralen Konfiguration kann es einfacher sein, Passwörter von einem zentralen Punkt aus im Credential Manager zu speichern.

Credentials Manager - Beispiel-Script zur Verteilung der Credentials

Wenn Sie sich für die Nutzung des Credential Managers entschieden haben (empfohlen), dann können Sie zur Verteilung der Credentials das von uns zur Verfügung gestellte Beispiel-Script nutzen.

Sicherheitshinweise

Wir empfehlen, ausschließliche Anmeldeinformationen für den Authenticator im Konnektor zu konfigurieren, um die Auswirkungen auf andere Systeme im Falle einer Preisgabe von Anmeldeinformationen zu minimieren.

Voraussetzungen

Stellen Sie sicher, dass das Ihr Client das Skript ausführen kann - hierfür müssen möglicherweise temporäre Änderungen an den Ausführungsrichtlinien erfolgen.

Laden Sie sich anschließend das von uns zur Verfügung gestellte Beispiel-Script hier herunter.

Konfiguration des Beispiel-Scriptes

Öffnen Sie das Script mit einem Texteditor Ihrer Wahl und passen Sie die dort aufgeführten Parameter Ihren an.

Hierbei sind zwei Schritte zu beachten:

Schritt 1: Tragen Sie einen der drei Targetnames (Punkt 1 im Screenshot) unter **$targetName=""** ein, um dem Script mitzuteilen, welche für den Authenticator notwendigen Credentials Sie nun übergeben möchten. Nutzen Sie hierfür gerne die Kopierfunktionen, da eine korrekte Schreibweise der Targetnames essentiell ist.

Schritt 2: Tragen Sie die dem Targetname zugehörigen Credentials ein (Punkt 2 im Screenshot)

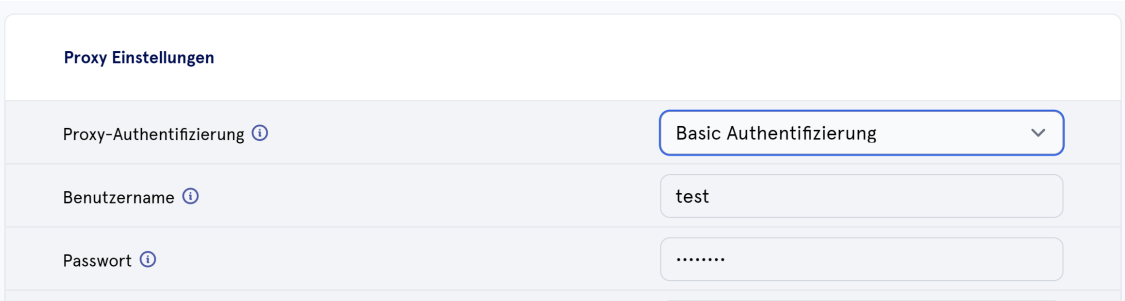
Hinweis: Für gewöhnlich benötigen Sie entweder "Gematik_Authenticator/Connector_BasicAuth" oder "Gematik_Authenticator /Connector_ClientCert_Password" für die erfolgreiche TLS Authentisierung gegenüber dem Konnektor, da nur eines aktiv im Konnektor konfiguriert ist. Nutzen Sie hingegen die TLS Authentifizierung mittels PEM-Dateien, benötigen Sie keines der beiden.

```powershell
#
# Copyright 2023 gematik GmbH
#
# The Authenticator App is licensed under the European Union Public Licence (EUPL); every use of the Authenticator App
# Sourcecode must be in compliance with the EUPL.
#
# You will find more details about the EUPL here: https://joinup.ec.europa.eu/collection/eupl
#
# Unless required by applicable law or agreed to in writing, software distributed under the EUPL is distributed on an
# IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the EUPL for the specific
# language governing permissions and limitations under the License.ee the Licence for the specific language governing
# permissions and limitations under the Licence.
#
# DISCLAIMER:
# This piece of software is an example to deploy credentials into the local Windows Credential Manager. It is not an
# official part of the Authenticator application. It is rather a helpful tool for administrators to assist the
# credentials distribution in a scaled environment. Hence, the gematik neither does offer support nor will the gematik
# be obliged to update this software.
# However, the gematik is free to do this voluntarily.

# Store credentials in Windows Credential Manager using cmdkey
# https://learn.microsoft.com/de-de/windows-server/administration/windows-commands/cmdkey

################################################################################
#       Choosing the Correct Credential Type
################################################################################

# Target name: 'Gematik_Authenticator/Connector_BasicAuth'
# Username and Password is required

# Target name: 'Gematik_Authenticator/Connector_ClientCert_Password'
# a generic (unused) username is required as well as the password of the pfx file. This is due to the fact, that the
# credential manager interface does not take password-only entries.

# Target name: 'Gematik_Authenticator/Proxy_BasicAuth'
# Username and Password is required

$validTargetNames = @(
"Gematik_Authenticator/Connector_BasicAuth",
"Gematik_Authenticator/Connector_ClientCert_Password",
"Gematik_Authenticator/Proxy_BasicAuth"
)

$targetName = ""

if ($targetName -notin $validTargetNames) {
    write-host "targetName only supports the following values:`n" ($validTargetNames -join "`n")
    return
}

################################################################################
#   Setting Up Credentials According to the Requirements for the Target
################################################################################
$userName = ""
$password = ""


# The following commented-out param command enables the parametrization of this script with a prompt.
# However, this only makes sense, if the script is run locally because the parameters are only usable on runtime.

#param ($userName, $password)


if ([string]::IsNullOrEmpty($userName)) {
    write-host "userName must be defined"
    return
}

if ([string]::IsNullOrEmpty($password)) {
    write-host "password must be defined"
    return
}

# This is the relevant command which is invoked.
$cmdkeyCommand = "cmdkey /generic:$targetName /user:$userName /pass:$password"
Invoke-Expression -Command $cmdkeyCommand;

```

1

2

Hinweis: Die Hinterlegung der Credentials für "Gematik_Authenticator/Proxy_BasicAuth" sind nur dann notwendig, wenn Sie im Authenticator in der Einstellungsseite die Proxy-Authentifizierung aktiviert sowie die Basic Authentifizierung ausgewählt haben. (Siehe nachfolgenden Screenshot)



Risikoverminderung und Ausführung des Beispiel-Scriptes

Im Verteilungsprozess könnte die Anwendung, die die Anmeldeinformationen enthält, auf einem Client-Gerät gefährdet sein. Um das Risiko einer Preisgabe von Anmeldeinformationen zu reduzieren, kompilieren wir in diesem Abschnitt das zuvor von Ihnen heruntergeladene Beispiel-Script .

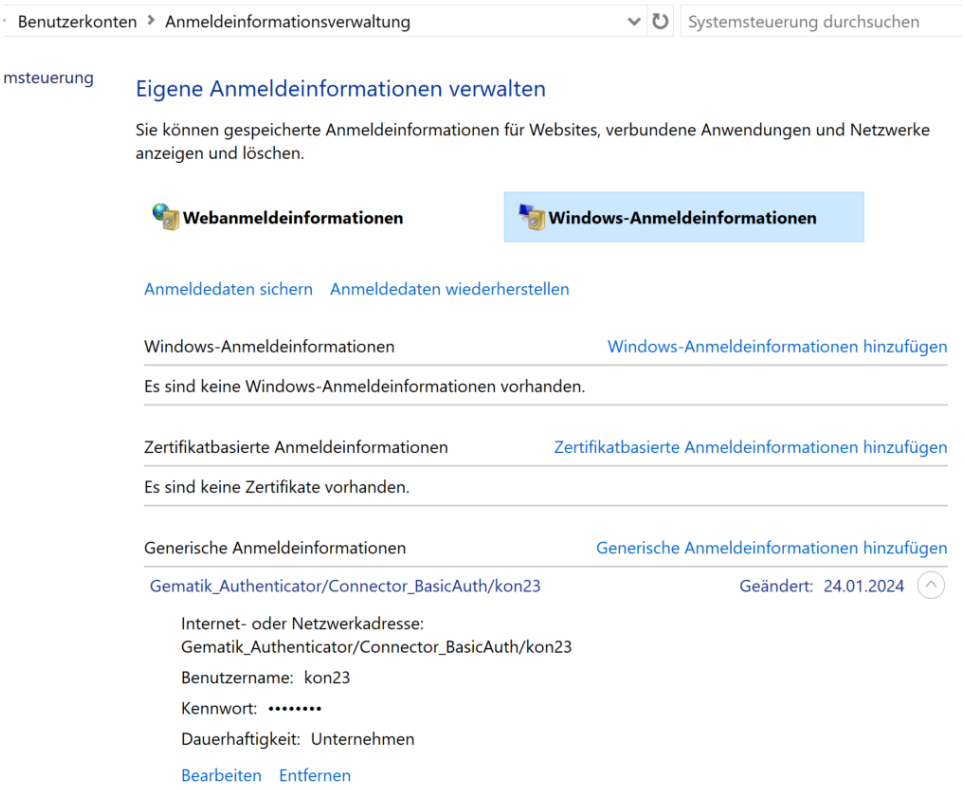
Beachten Sie, dass diese Methode die Anmeldeinformationen im Skript nicht sicher verschlüsselt. Es verschleiert sie vielmehr, da uns kein kryptografisches Material oder Geheimnis vorliegt, das wir verwenden können.

Schritt 1: Installieren Sie "ps2exe", um die Kompilierung von PowerShell zu exe zu ermöglichen. Der Compiler kann aus der PowerShell Gallery heruntergeladen werden oder einfach via PowerShell Command: "Install-Modul ps2exe"

Schritt 2: Führen Sie "ps2exe .\credential-distribution\store-credential.ps1 authenticator-configure-environment.exe" in Ihrer PowerShell aus

Hinweis: Der Name authenticator-configure-environment.exe ist hierbei ein Beispiel - Sie können die .exe auch anders benennen.

Schritt 3: Führen Sie die nun erstellte authenticator-configure-environment.exe aus und überprüfen Sie im Windows Credentials Manager (Anmeldeinformationsverwaltung), ob Ihre Daten korrekt übertragen wurden. (Siehe Screenshot am Beispiel "Gematik_Authenticator /Connector_BasicAuth")



Schritt 4: Löschen Sie nach erfolgreicher Verteilung Ihrer Credentials dieses Script (.\credential-distribution\store-credential.ps1) sowie die neu erzeugte kompilierte Anwendung (authenticator-configure-environment.exe), um die Präsenz des Skripts zu minimieren. Die Annahme ist, dass das Skript auf einem Client-Gerät weniger sicher ist als auf dem Administrator-Gerät.

Vereinfachte Konfiguration für Pflegeeinrichtungen (nur anwendbar für SMC-B Karten)

Wenn Ihre Institution für die Authentisierung gegenüber einer Fachanwendung nur SMC-B Karten benötigt, ist eine vereinfachte Konfiguration des Authenticators mittels eines *DummyArbeitsplatzes* möglich.

Einstellungen für Standalone-, Remote- oder mehrfacher Standalone-Umgebungen

Diese vereinfachte Konfiguration unterscheidet sich zu den zuvor erklärten Konfigurationsszenarien (Szenario 1.1, 1.2, 2a, 2b und 3) nur in der Definition der *ArbeitsplatzID.* Das bedeutet, wenn Sie beispielsweise eine Citrix-Umgebung verwenden, muss dennoch alles - *bis auf die ArbeitsplatzID* - wie in Szenario 2a konfiguriert werden.

Um eine *DummyArbeitsplatzID* zu konfigurieren, gehen Sie bitte die nachfolgend erklärten Schritte durch:

Einstellungen im Konnektor

Für dieses Szenario muss im Konnektor ein *DummyArbeitsplatz* angelegt und auch im dortigen Infomodell/Aufrufkontext hinterlegt werden.

Gehen Sie dafür auf die Managementoberfläche Ihres Konnektors und legen Sie eine neue *ArbeitsplatzID* an und nennen Sie sie zum Beispiel *DummyArbe itsplatz*.

Hinweis: *Wie Sie die ArbeitsplatzID benennen ist Ihnen überlassen, der Name hat keinerlei Auswirkungen auf die Anwendungen - wichtig ist, dass Sie ihn leicht zuordnen können.*

Wenn Sie die neue *DummyArbeitsplatzID* im Konnektor angelegt haben, müssen Sie diese nun auch im Aufrufkontext bzw. Infomodell des Konnektors hinterlegen. Das bedeutet, dass die *DummyArbeitsplatzID* mit der ClienId, MandantenID und den Kartenterminals verknüpft sein muss, damit anschließend alle IDs korrekt zugeordnet und die Fachanwendung ordnungsgemäß aufgerufen werden kann.

Einstellungen im Authenticator

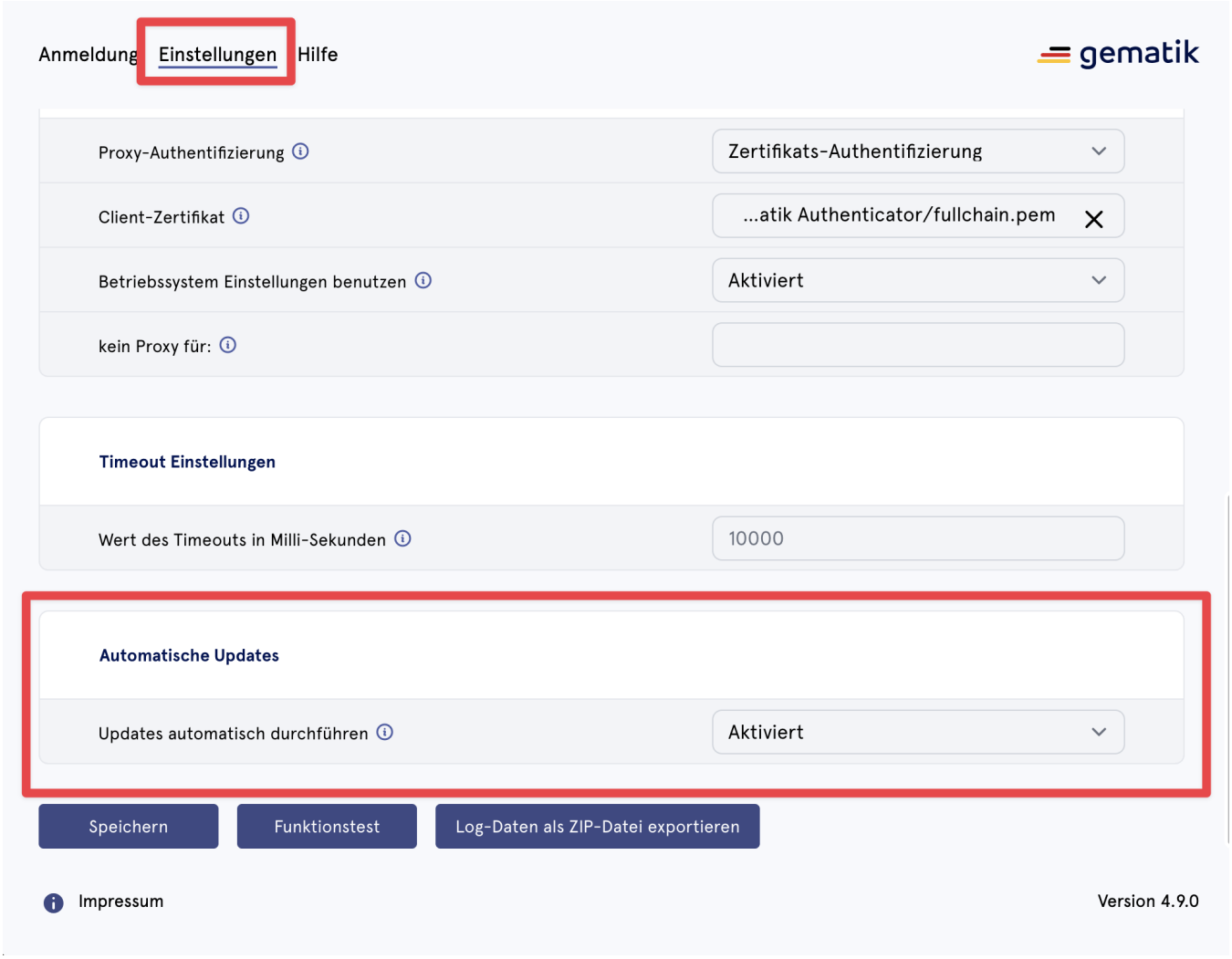
Der von Ihnen angelegte Aufrufkontext kann nun an allen Arbeitsplätzen, die eine vereinfachte Konfiguration benötigen (Fachanwendungen, die nur eine SMC-B erfordern) auf der Einstellungsseite des Authenticators eingetragen und abgespeichert werden.

Wichtiger Hinweis: Entscheiden Sie sich für diese Art der Konfiguration und verwenden doch eine Fachanwendung, die einen HBA erfordert, kann es zu unerwünschten, unschönen oder sogar sicherheitskritischen Nebeneffekten kommen. Der Grund hierfür liegt darin, dass alle Kartenterminals kontaktiert werden, die dem angelegten DummyArbeitsplatz zugeordnet sind. Sind also mehr als ein Kartenterminal zugeordnet und verbunden, kann es zu erheblichen Performanceproblemen kommen. Daher können wir diese Art der Konfiguration nicht offiziell empfehlen, da es nicht der Spezifikation entspricht! Unsere Empfehlung ist daher weiterhin eine Arbeitsplatz-spezifische Konfiguration vorzunehmen, um auf der sicheren Seite zu sein.

Automatische Updates

Innerhalb der Konfigurationseinstellungen haben Sie die Möglichkeit, die automatische Updatefunktion für den Authenticator zu aktivieren bzw. auch zu deaktivieren. Die automatische Updatefunktion ermöglicht es, Updates direkt aus der Application heraus installieren zu können, vorausgesetzt ist, dass auch eine neue Version veröffentlicht wurde.

Bei einer Neuinstallation ist das Auto-Update defaultmäßig aktiviert:



Auszug aus der config.json

```
"checkUpdatesAutomatically": true,
```

Hinweis:

- Bei jeglichem Update auf die Version 3.0.0 ist die Auto-Update-Funktion zunächst noch nicht gesetzt und somit leer.
  Hier müsste der Status entsprechend manuell auf "aktiviert" oder "deaktiviert" gesetzt werden. Defaultmäßig ist die Funktion in diesem Fall somit deaktiviert.
- Die Updates werden über die Seite https://github.com/gematik/app-Authenticator bezogen. Bitte achten Sie darauf, dass wenn Sie das Feature nutzen möchten, die Firewall evtl. angepasst werden muss.
- Das Update wird als signierte exe-Datei ausgeliefert. Wenn dies in Ihrer Umgebung nicht erlaubt ist, kann das Update nicht automatisiert durchgeführt werden.
- **Bis Version 4.0.0:** Nach einem Update über die Update-Funktion werden die aktuell eingespielten CA-Zertifikate **nicht** mit in die neue Version übertragen. Vergewissern Sie sich, dass die Zertifikate zwischengespeichert werden:
  - Pfade der Certs:
    - C:\Program Files\gematik Authenticator\resources\certs-idp
    - C:\Program Files\gematik Authenticator\resources\certs-konnektor
- **Ab Version 4.1.0:** Nach einem Update über die Update-Funktion werden die aktuell eingespielten CA-Zertifkate in die neue Version übertragen. Es werden nur die vom Authenticator mitgelieferten Zertifikate automatisch erneuert, sofern notwendig. Ihre eigens hinterlegten Zertifikate, die unter den folgend aufgeführten Pfaden abgelegt wurden, bleiben erhalten:
  - Pfade der Certs:
    - C:\Program Files\gematik Authenticator\resources\certs-idp
    - C:\Program Files\gematik Authenticator\resources\certs-konnektor

Proxyunterstützung

Der Authenticator unterstützt per Default HTTP und HTTPS Proxies. Dafür liest und nutzt der Authenticator den im Betriebssystem hinterlegten Proxy (u.a. auch PAC-Files). Für eine Proxyunterstützung mit Authentifizierung (Basic-Auth oder Client-Zertifizierung) müssen zusätzliche Einstellungen in der Einstellungsübersicht des Authenticators vorgenommen werden. Wie Sie diese vornehmen, ist nachfolgend erklärt.

Hinweis: Wenn Sie einen HTTPS-Proxy betreiben, müssen Sie die notwendigen Zertifikate unter

IDP C:\Users\${Username}\AppData\Local\gematik Authenticator\resources\certs-idp

Konnektor C:\Users\${Username}\AppData\Local\gematik Authenticator\resources\certs-konnektor

hinterlegen, da es sonst zu einem Fehler bei der Erreichbarkeit kommt.

Proxy Whitelisting von IP-Adressen/Ranges:

Die App unterstützt das Whitelisting von IP-Adessbereichen. Es können 1:n Adressbereiche auf der Einstellungsseite des Authenticators unter "*kein Proxy für"* hinterlegt werden.

Die Adressbereiche müssen mit einem Trenner ";" getrennt werden.

Bsp.: IP mit Subnetzbereich
125.19.23.0/24; XXX,XX,XX,X/16

Unterstütze Eingaben mit Beispielen (ab 4.8.0)

IP	Beispiele
IPv4 oder IPv6	• Regular IPv4: `10.0.0.0` • Wildcard IPv4: `10.0.0.*` or even `10.*.0.*` • Regular IPv6: `2001:0db8:85a3:0000:0000:8a2e:0370:7334` • Shortened IPv6: `2001:db8:85a3::8a2e:0370:7334` or `::` or `::1` or `a::` • Wildcard IPv6: `2001::*` or even `2001::*:abc:*` • Mixed IPv6 (mapped IPv4): `::ffff:127.0.0.1` *(no wildcards allowed in IPv4 part)* • **Nicht unterstützt**: `10.0.1*.0` or `2001::a*c`
IP Range	• IPv4: `10.0.0.0-10.1.2.3` • IPv6: `2001::abc-2001::1:ffff` • **Note**: Left side has to be "lower" than the right side
IP Subnetwork	• IPv4: `10.0.0.0/16` • IPv6: `2001::/123`
IP Mask	• IPv4: `10.0.0.0/255.0.64.0` • IPv6: `2001:abcd::/ffff:ff8::`
FQDN (a 4.9.0)	• 172.*.*.* • *.gematik.de • *.int.gematik.de

Proxy-Authentifizierung
Die App unterstützt zwei Arten der Proxy-Authentifizierung: Standardauthentifizierung und Client-Zertifikat-Authentifizierung. Standardmäßig ist diese Funktion deaktiviert und kann bei Bedarf entsprechend aktiviert werden.

Die App erkennt und verwendet automatisch die Proxyserveradresse und die Porteinstellungen, die auf Ihrem Betriebssystem konfiguriert sind. Das bedeutet, dass Sie diese Informationen nicht manuell in den App-Einstellungen hinterlegen müssen. Es ist jedoch wichtig sicherzustellen, dass Ihr Betriebssystem für die Verwendung eines Proxy-Servers konfiguriert ist, wenn Sie die Proxy-Authentifizierungsfunktion in der App verwenden möchten.

Proxy-Authentifizierung einrichten
Die Proxy-Einstellungen finden Sie unter dem Reiter „Einstellungen“ in der App. Von dort aus können Sie die Art der Authentifizierung konfigurieren, die Sie verwenden möchten.

Grundlegende Authentifizierung (Basic Auth)
Bei der Basisauthentifizierung werden bei jeder Anfrage an den Proxy-Server ein Benutzername und ein Passwort gesendet. Der Proxy-Server validiert dann die Anmeldeinformationen und gewährt Zugriff auf die angeforderten Ressourcen, wenn sie gültig sind.

Um die Standardauthentifizierung zu verwenden, müssen Sie Ihre Proxy-Anmeldeinformationen (Benutzername und Passwort) in den App-Einstellungen angeben. Sobald diese Informationen eingegeben wurden, sendet die App die Anmeldeinformationen automatisch mit jeder Anfrage an den Proxy-Server.

Client-Zertifikatsauthentifizierung
Die Client-Zertifikatsauthentifizierung ist eine weitere Methode der Proxy-Authentifizierung, bei der ein digitales Zertifikat zur Identifizierung des Clients verwendet wird. Diese Methode ist sicherer als die Standardauthentifizierung.

Um die Client-Zertifikatsauthentifizierung zu verwenden, müssen Sie den Pfad Ihres Proxy-Servers zur Clientzertifikatdatei in den App-Einstellungen angeben. Sie müssen außerdem sicherstellen, dass die Zertifikatsdatei im PEM-Format oder als P12-Datei vorliegt.

Feldname	Beschreibung	Beispiel
Proxy-Authentifizierung	Proxy Typ	Basic Authentifizierung oder Zertifikats-Authentifizierung
Benutzername	Benutzername (Basic Authentifizierung)	p_user
Passwort	Passwort (Basic Authentifizierung)	p_password
Client-Zertifikat	Passwort (Zertifikats-Authentifizierung)	PEM Zertifikat für Proxy Server

UNC-Pfade

In der Konfiguration config.json können alle Pfadangaben auch über UNC Pfade erfolgen. Bei der Eingabe ist darauf zu achten, dass "\" escaped werden muss.

Beispiel:

\\\\UNC-PFAD\\UNTERORDNER\\client.pem

C:\\Users\\VORNAME.NACHNAME\\Documents\\Mock-Modus Zertifikate\\SMC-B Universitätsklinik certificate.pem

Sicherheitshinweise zur Konfiguration

Bitte beachten Sie, dass bei einer Client-Authentisierung gegenüber dem Konnektor oder Proxy das Schlüsselmaterial oder Passwort unverschlüsselt lokal (Szenario 1.1, Szenario 1.2) oder zentral (Szenario 2, Szenario 3) gespeichert wird. Dies kann potenzielle Sicherheitsrisiken mit sich bringen, daher empfehlen wir Ihnen, angemessene Maßnahmen zum Schutz dieser sensiblen Informationen zu ergreifen, wie z.B. Festplattenverschlüsselung, Rechte-Management, etc., oder setzen Sie die Verzeichnisse auf "read-only" und "hidden".

Erforderliche Einstellungen

Einstellungen im Authenticator zum Konnektor

Host	Hostname / IP vom Konnektor	IP-Adresse des Konnektors	Speicherort
Port	Port des Konnektors	Port des Konnektors. In der Regel 443 da der Authenticator nur gesicherte Verbindungen (https) zu dem Konnektor aufbaut.	Config File
Mandant-ID	ID vom Mandant	Mandant-x	Config File
Client-ID	ID vom Client	Client-System-x	Config File
Arbeitsplatz-ID	ID vom Arbeitsplatz	Arbeitsplatz-x	Config File
TLS-Verbindung	Gesicherte Verbindung erforderlich	Auswählbare Möglichkeiten: • BasicAuth • ServerCertAuth • ServerClientCertAuth • ServerClientCertAuth_Pfx	Config File
Privater Schlüssel	Eine .pem-Datei mit private-key	Kann vom Konnektor generiert werden	Config File
Client-Zertifikat	Eine .pem-Datei mit Client-Zertifikat	Kann vom Konnektor generiert werden	Config File
Benutzername	Wird im Konnektor für BasicAuth angelegt	"user007"	**Credential Manager***
Passwort	Wird im Konnektor für BasicAuth angelegt	"se2ret-ser3ice"	**Credential Manager***

P12-Datei (Pfx-Datei)	Eine p12-Datei, die ein RSA Zertifikat enthält	Kann vom Konnektor generiert werden	Config File
Passwort der P12-Datei (pfx-Datei)	Passwort der p12-Datei (pfx-Datei)	Wird beim Generieren der P12-Datei festgelegt	**Credential Manager***

***Hinweis*: Damit die sensiblen Daten im Credentials Manager bei einer zentralen Konfiguration ordnungsgemäß gespeichert werden können, berücksichtigen Sie bitte die Hinweise unter Kapitel Verwendung des Credential Managers.

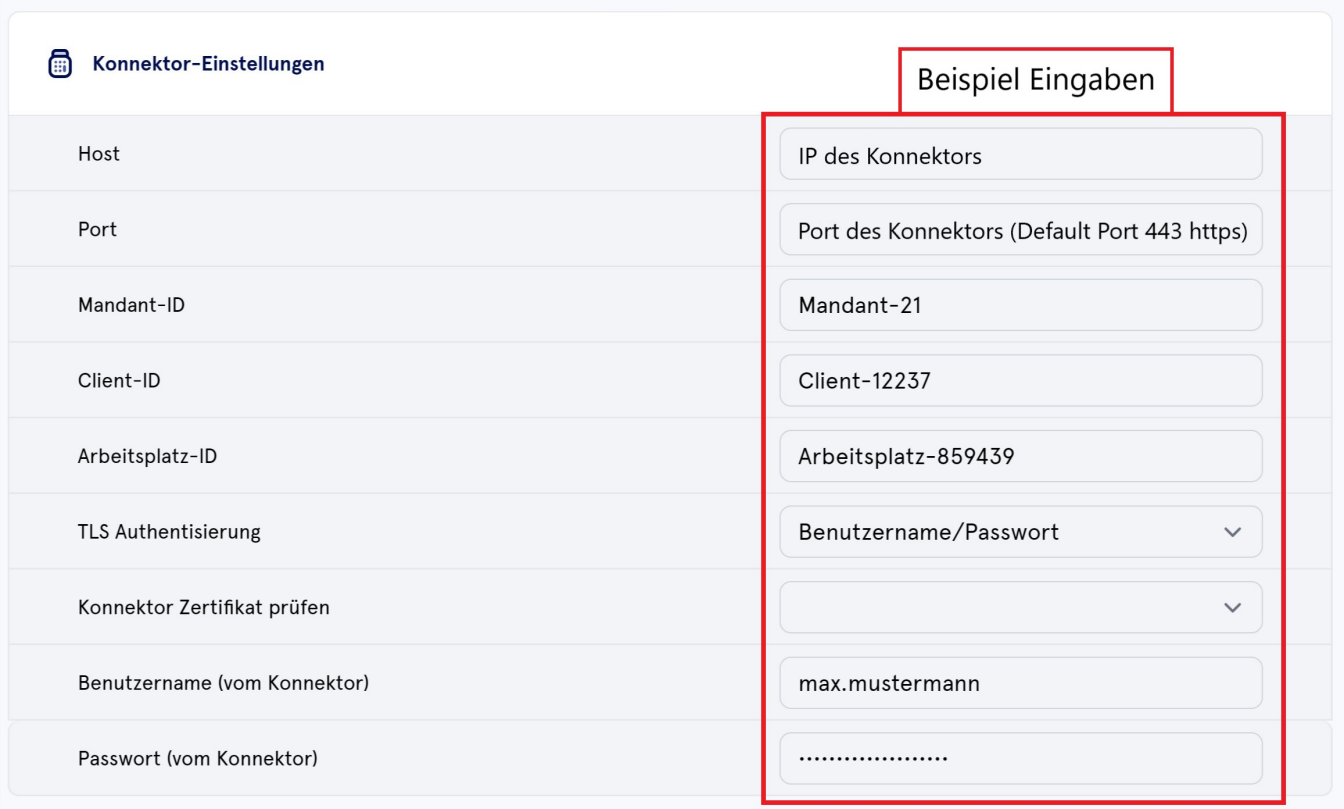
Woher bekomme ich die entsprechenden Konnektor-Einstellungen?

Für jeden Konnektor gibt es entsprechende Admin-Handbücher von dem jeweiligen Hersteller, in welchen beschrieben sind, was an welcher Stelle konfiguriert werden kann und sollte.

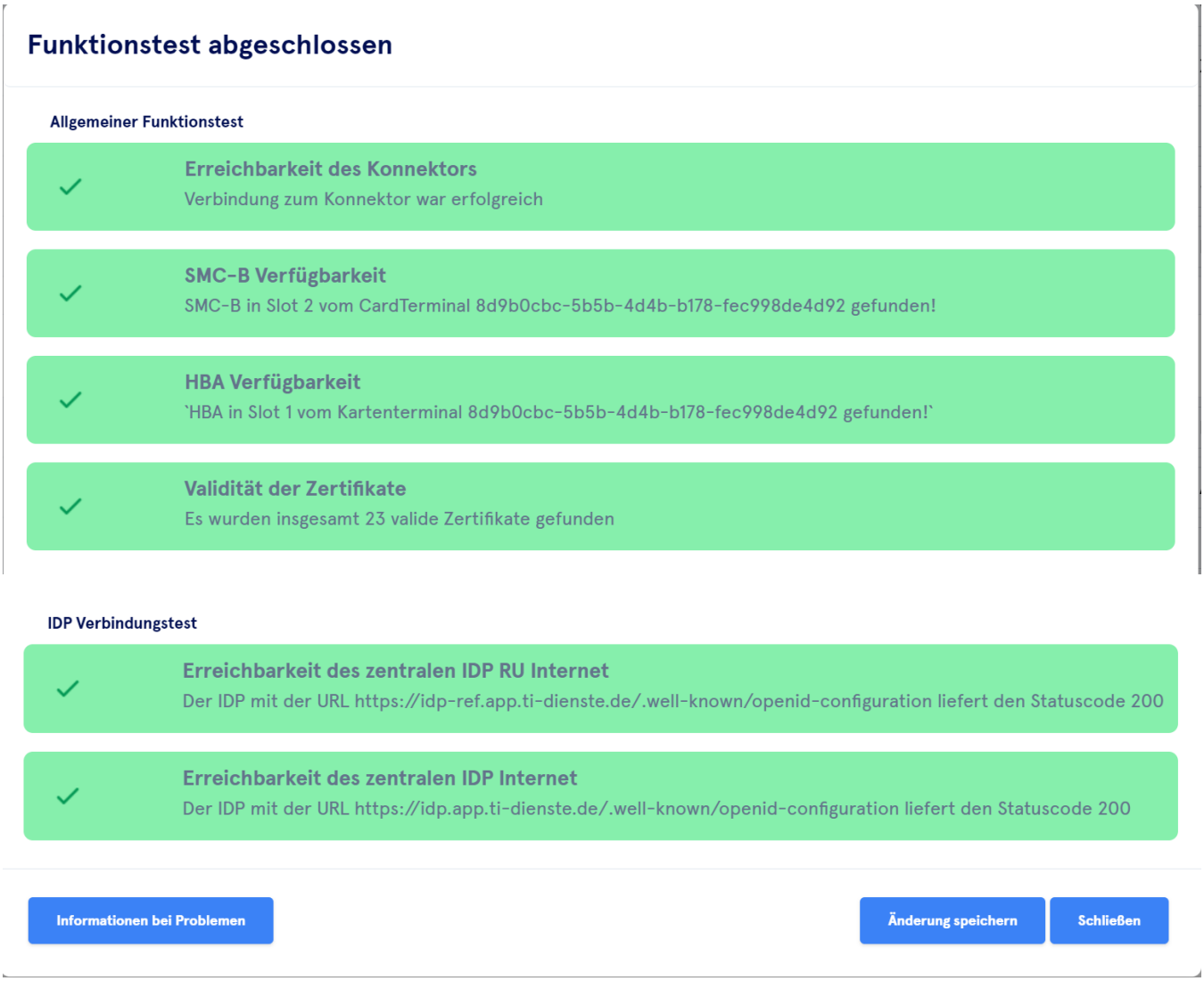
Was am aktuellen Konnektor konfiguriert wurde, weiß i.d.R. der Nutzer/Admin des Konnektors. Die jeweiligen Werte können in der Konnektor-Admin-Oberfläche aufgefunden werden und müssen dementsprechend beim Admin oder Betreiber des Konnektors angefragt werden.

Nach Hinterlegung aller Einstellungen besteht die Möglichkeit, einen Funktionstest durchzuführen.

Hierbei wird getestet, ob die hinterlegten Einstellungen eine Verbindung zum IDP-Dienst und zum Konnektor aufbauen können.



War der Funktionstest erfolgreich, wurden alle Einstellungen **innerhalb** des Authenticators korrekt hinterlegt.



Hinweis: Unter C:\Program Files\gematik Authenticator\resources\test-cases-config.json können die zu testenden Endpunkte definiert werden.

Bereitstellen der PEM-Dateien bei zertifikatsbasierter TLS-Authentifizierung gegenüber dem Konnektor

Der private Schlüssel sowie das Client-Zertifikat sollten im PEM-Format vorliegen und werden beim Auswählen automatisch in den Ordner C:/Users/{Nutzerverzeichnis}/AppData/Local/gematik Authenticator kopiert.

Hinweis: *Wenn Sie die TLS-Authentisierung-**Zertifikat** nutzen möchten, achten Sie bitte darauf, dass diese im PEM-Format oder als P12-Datei vorliegen.*

Möchten Sie eine P12-Datei nutzen, muss diese ein gültiges RSA-Zertifikat enthalten, da es ansonsten zu einem Fehler kommt. Halten Sie hierfür das dazugehörige Passwort bereit.

Sie können vorhandene P12-Zertifikate auch in ein PEM-Format umwandeln - Eine Möglichkeit, wie die Datei in das PEM-Format exportiert werden kann, finden Sie hier: Umwandlung einer p12-Datei in PEM-Format.

Zugriff auf Fachanwendung ermöglichen

IP-Routing konfigurieren

Der gematik Authenticator wird im Zusammenspiel mit weiteren Anwendungen der TI (WANDA) eingesetzt, die für die Nutzerinteraktion einen Web-Browser verwenden (eine sogenannte Web-Anwendung).

Je nach Anwendung und Konnektor-Konfiguration kann es hierbei erforderlich sein, ein IP Routing für diese Anwendung zu konfigurieren, damit die Anwendung über die TI erreichbar ist. Die Einrichtung des IP-Routings kann hierbei am Arbeitsplatz selbst oder zentral am Gateway konfiguriert werden.

Eine Konfiguration am Arbeitsplatz ist für Windows z.B. mit folgendem Kommando möglich (Kommandozeile CMD mit Admin-Rechten):

route add Kommando

```
route add <Netzwerk> MASK <Mask> <IP-Konnektor > –p
```

Für WANDA Basic können die Parameter in der Konnektor-Admin-Oberfläche über die Liste der Bestandsnetze eingesehen werden.

Für einige Anwendungen sind im Folgenden die notwendigen Parameter aufgeführt:

Anwendung	**Netzwerk**	**Mask**
Alle WANDA Smart Anwendungen	100.102.0.0	255.254.0.0

Hinweise:

- Das Netzwerk 100.102.0.0/15 fasst alle TI Adressen (offenen Fachdienst der TI und WANDA Smart) zusammen. I.d.R. sollte an TI Arbeitsplätzen bereits eine entsprechende IP-Route vorhanden sein (siehe oben, "IP-Routing konfigurieren").
- WANDA Basic kann öffentliche IP-Adressbereiche und in Zukunft auch TI Adressen aus dem Bereich 100.102.0.0/15 verwenden. Insbesondere wenn öffentliche IP-Adressbereiche für die Anwendung verwendet werden, muss für die Anwendung eine spezifische IP-Route konfiguriert werden.

Nach Einrichtung des Routings können Sie mit folgenden Befehlen innerhalb der Kommandozeile testen, ob die Fachwendung erreicht werden kann:

traceroute <Netzwerk>

ping <DNS Fachanwendung>

- Bsp. ZVR: ping zvr-ae.bnotk.de

Firewall-Freischaltungen

Für die Nutzung des Authenticators sollten folgende Firewall-Freischaltungen hinterlegt sein:

Verbindung zum IDP via Internet	Lokaler Client/Workstation (localhost)	idp.app.ti-dienste.de	443	https
Verbindung zum IDP via TI Endpunkt	Lokaler Client/Workstation (localhost)	idp.zentral.idp.splitdns.ti-dienste.de	443	https
Verbindung zu WANDA Applikationen *Bsp.: Zentrales Vorsorgerister*	Lokaler Client/Workstation (localhost) *Bsp.: Lokaler Client/Workstation (localhost)*	100.102.0.0 / 15 *Bsp.: https://zvr-ae.bnotk.de/*	443 *Bsp.: 443*	https *Bsp.: https*
Auto-Updatefunktion	Lokaler Client/Workstation (localhost)	https://github.com/gematik/app-Authenticator	443	https
Konnektor	Lokaler Client/Workstation (localhost)	internes Netzwerk	443	https
Kartenterminal	Lokaler Client/Workstation (localhost)	internes Netzwerk	443	https

---

Protokollierung

Die Anwendung protokolliert ihre Verarbeitungsprozesse anhand von Nachrichten, die in die Logdatei geschrieben werden. Die Logdatei wird an folgendem Ort im Dateisystem abgelegt:

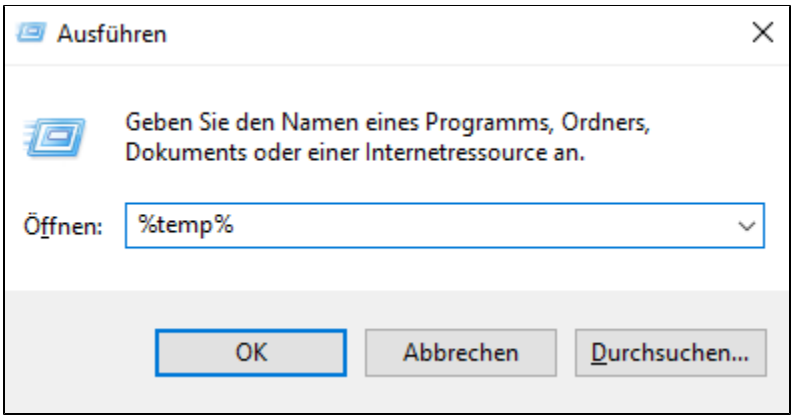
Pfad zum Logfile

```
C:/Users/{Nutzerverzeichnis}/AppData/Local/Temp/authenticator-logging/authenticator-${datum}.log
```

Das Logfile hat eine Obergrenze von 100MB. Dateien, die älter als 14 Tage sind, werden automatisch entfernt.

Bei Nutzung einer Virtualisierungs-Lösung (Citrix/VMware/RDP):

Die Log-Files werden im tmp-Verzeichnis der jeweiligen aktiven Session geschrieben. Um die Log-Files auffinden zu können, reicht es im Explorer oder unter Windows - Ausführen "%temp%" zu öffnen.

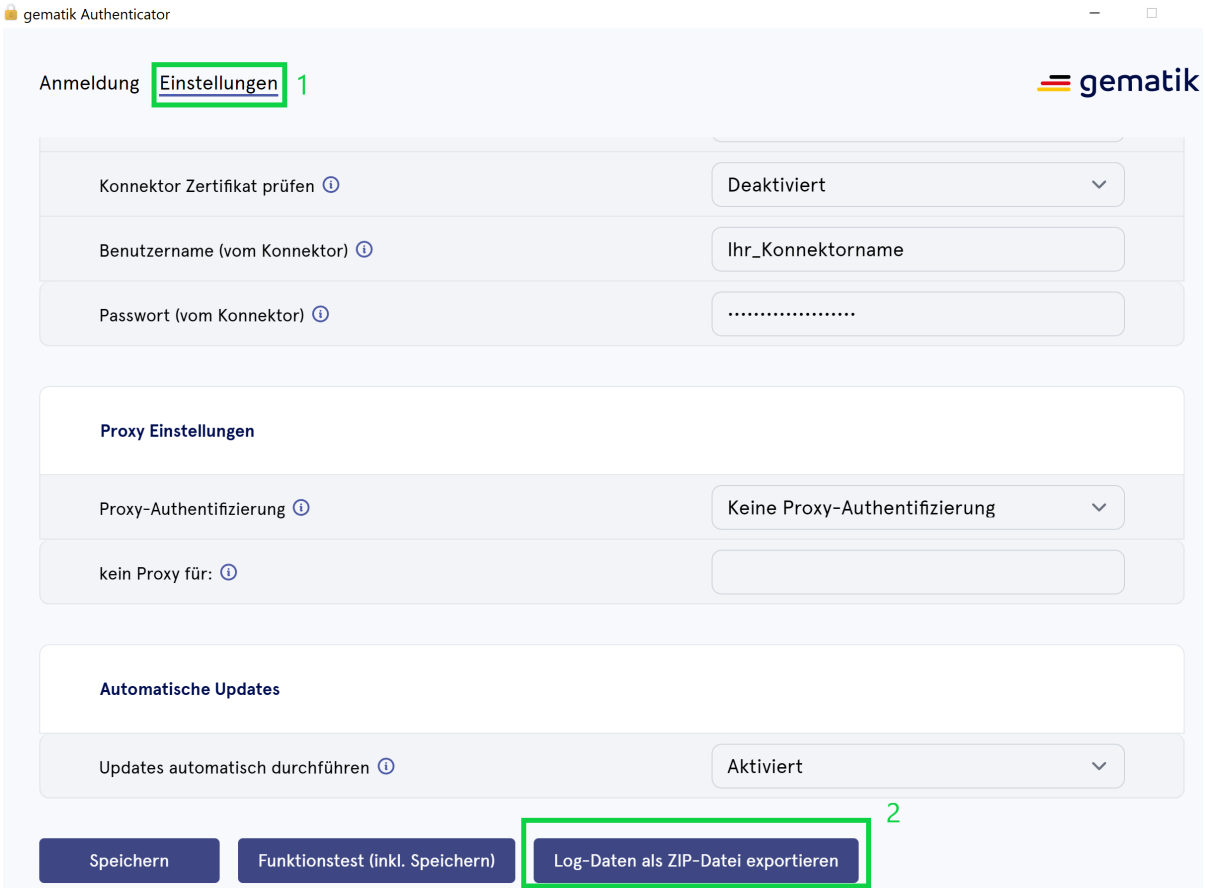


Somit gelangen Sie direkt in das lokale tmp-Verzeichnis der aktuellen aktiven Session und können das die Log-Files unter ./authenticator-logging/* auffinden.

Log-Daten als ZIP-File

Es gibt nun die Möglichkeit, sich die vom Authenticator erstellten Logfiles bequem via Knopfdruck, als Zip-File zu exportieren.

Den Button dazu finden Sie auf der Einstellungsseite (1) unten unter "*Log-Daten als ZIP-Datei exportieren*" (2) (siehe Screenshot):



Deinstallation

Um das Programm zu deinstallieren, verwenden Sie bitte den für Ihr Betriebssystem standardmäßigen Vorgang (über Windows/Add- bzw. Remove Files oder direktes Ausführen der Uninstall-Executable im Authenticator-Programm-Verzeichnis (z.B. C:\Program Files\gematik Authenticator)). Es werden beim Deinstallieren die installierten Dateien im Programm-Verzeichnis des Authenticators gelöscht und die 2 Unterordner ressources\certs-idp und ressources\certs-konnektor und deren Inhalt unter C:\Users\user.name\AppData\Local\Temp\gematik Authenticator\backup zwischengespeichert. Bei einer Neuinstallation oder einem Update wird der Inhalt dieser Verzeichnisse wieder in das Programm-Verzeichnis des Authenticators eingespielt, sodass manuell hinzugefügte Zertifikate erhalten bleiben.

Für die Installation und Deinstallation kann mit dem Parameter **/S eine "Silent"** Installation- bzw. Deinstallation durchgeführt werden.

Aktualisieren des Programms

Um das Programm auf den aktuellen Stand zu bringen, laden Sie bitte die aktuelle Version herunter (siehe Beschaffung der Installationsdatei). Nachfolgend führen Sie das Installationsprogramm aus (siehe Installation).