
C_12133_Anlage

Änderung in gemSpec_Krypt

Die folgenden 3 Anforderungen werden so angepasst, dass speziell die in der OpenApi I_Authorization_Service realisierte Rückgabe des vau-np korrekt als Bestandteil der Response einer Operation erwartet wird (und nicht im Header der Response).

Es wird dabei auch die Schreibweise von "VAU-NP" auf "vau-np" geändert, da das Element in der OpenApi als json Element case sensitive ist und dort klein geschrieben wird.

Die Verwendung des NP im HTTP-Header der Nachricht 1 (A-24757) wird auch aus Konsistenzgründen auf die kleine Schreibweise geändert. Eine Anpassung seitens des Clients/des AS dürfte nicht erforderlich sein, da HTTP-Header nicht case sensitive sind (es kann also wie bisher "VAU-NP2 verwendet werden).*

alt:

A_24757 - VAU-Protokoll: Nutzerpseudonym

Ein VAU-Client MUSS die HTTP-Variable "VAU-NP" aus vorherigen VAU-Protokoll-Verbindungen speichern und bei den folgenden VAU-Protokoll-Handshakes bei der Nachricht 1 (vgl. A_24428-*) im HTTP-Request-Header aufführen (Hinweis: zu dem Zeitpunkt kann es noch keine inneren HTTP-Request geben / Verbindungsaufbau). Initial hat ein Client diesen Wert noch nicht, dann führt er die Variable nicht im Request-Header auf.

Ein VAU-Client MUSS nach einer erfolgreichen Client-Authentisierung (vgl. [gemSpec_Krypt#7.3 und 7.4]) den inneren HTTP-Response-Header prüfen, ob die Variable "VAU-NP" aufgeführt ist, falls ja MUSS er einen lokal gespeicherten Wert ggf. aktualisieren (i. S. v. ein Aktensystem (VAU-Instanz) wird dieses Nutzerpseudonym auch wechseln).

[<=, CS_ePA_KTR, PS_ePA_Apotheke, CS_ePA_DiGA, PS_ePA, eRp_FD, CS_ePA_Ombudsstelle, NCPeH_FD, Frontend_Vers_ePA, funkt. Eignung: Herstellererklärung, funkt. Eignung: Konformitätsbestätigung, Sich.techn. Eignung: Herstellererklärung]

neu:

Korrektur "HTTP-Variable" -> "Variable" und "innerer HTTP-Response-Header" -> "innere HTTP-Response"

A_24757-01 - VAU-Protokoll: Nutzerpseudonym

Ein VAU-Client MUSS die Variable "vau-np" aus vorherigen VAU-Protokoll-Verbindungen speichern und bei den folgenden VAU-Protokoll-Handshakes bei der Nachricht 1 (vgl. A_24428-*) im HTTP-Request-Header aufführen (Hinweis: zu dem Zeitpunkt kann es noch keine inneren HTTP-Request geben / Verbindungsaufbau). Initial hat ein Client diesen Wert noch nicht, dann führt er die Variable nicht im Request-Header auf.

Ein VAU-Client MUSS nach einer erfolgreichen Client-Authentisierung (vgl. [gemSpec_Krypt#7.3 und 7.4]) die innere HTTP-Response prüfen, ob die Variable "VAU-NP" (bzw. vau-np) aufgeführt ist, falls ja MUSS er einen lokal gespeicherten Wert ggf. aktualisieren (i. S. v. ein Aktensystem (VAU-Instanz) wird dieses Nutzerpseudonym auch wechseln).

[<=, CS_ePA_KTR, PS_ePA_Apotheke, CS_ePA_DiGA, PS_ePA, eRp_FD, CS_ePA_Ombudsstelle, NCPeH_FD, Frontend_Vers_ePA, funkt. Eignung: Herstellererklärung, funkt. Eignung: Konformitätsbestätigung, Sich.techn. Eignung: Herstellererklärung]

alt:

A_25055-01 - VAU-Protokoll: OIDC Authentisierung oberhalb der VAU-Protokoll-Schicht

Eine VAU-Instanz MUSS für eine OIDC/OAuth2/PCKE notwendige HTTP Endpunkte per inneren HTTP-Request/Responses einem Client verfügbar machen (Codechallenge per GET zur Verfügung stellen, Authentication Code per POST vom Client empfangen) machen.

Nach erfolgreicher Authentisierung über OIDC MUSS die VAU-Instanz in den Metadaten der Verbindungsschlüssel (KeyID, K2_*_app_data) den neuen Authentisierungszustand vermerken (Rückwirkung auf A_24634-*).

Nach erfolgreicher Authentisierung über OIDC, genauer innerhalb der Response (innerer Request) nach dem POST-Request für den Authentication-Codes durch den Client, MUSS sie ein Nutzerpseudonym dem Client in der HTTP-Variable "VAU-NP" verfügbar machen, vgl. A_24770-*.

[<=, Aktensystem_ePA, funkt. Eignung: Herstellererklärung, Sich.techn. Eignung: Produktgutachten]

neu:

Korrektur "HTTP-Variable" -> "Variable"

A_25055-02 - VAU-Protokoll: OIDC Authentisierung oberhalb der VAU-Protokoll-Schicht

Eine VAU-Instanz MUSS für eine OIDC/OAuth2/PCKE notwendige HTTP Endpunkte per inneren HTTP-Request/Responses einem Client verfügbar machen (Codechallenge per GET zur Verfügung stellen, Authentication Code per POST vom Client empfangen) machen.

Nach erfolgreicher Authentisierung über OIDC MUSS die VAU-Instanz in den Metadaten der Verbindungsschlüssel (KeyID, K2_*_app_data) den neuen Authentisierungszustand vermerken (Rückwirkung auf A_24634-*).

Nach erfolgreicher Authentisierung über OIDC MUSS die VAU-Instanz ein Nutzerpseudonym (NP) in der Response des inneren POST-Requests für den Authentication -Code als "vau-np" eingetragen und so an den Client übertragen, vgl. A_24770-*.

[<=, Aktensystem_ePA, funkt. Eignung: Herstellererklärung, Sich.techn. Eignung: Produktgutachten]

alt:

A_24770 - VAU-Protokoll: VAU-Instanz, Nutzerpseudonym erzeugen

Nach erfolgreicher Nutzer-Authentisierung des Clients MUSS die VAU-Instanz ein Nutzerpseudonym (NP) erzeugen. Dieses NP MUSS, im HTTP-Response-Header des inneren HTTP-Request unter dem Variablennamen "VAU-NP" eingetragen werden und so an den Client übertragen werden. [<=, Aktensystem_ePA, funkt. Eignung: Herstellererklärung, Sich.techn. Eignung: Produktgutachten]

neu:

Korrektur "HTTP-Response-Header" -> "Response" und "VAU-NP" -> "vau-np"

A_24770-01 - VAU-Protokoll: VAU-Instanz, Nutzerpseudonym erzeugen

Nach erfolgreicher Nutzer-Authentisierung des Clients MUSS die VAU-Instanz ein Nutzerpseudonym (NP) erzeugen. Dieses NP MUSS, in der Response des inneren HTTP-

Requests als "vau-np" eingetragen und so an den Client übertragen werden.
[<=, Aktensystem_ePA, funkt. Eignung: Herstellererklärung, Sich.techn. Eignung:
Produktgutachten]