

Wykorzystanie SAML 2.0 w systemie ePUAP

Spis treści

1	Wstęp.....	2
2	Co to jest SAML 2.0.....	3
3	Podstawowe cechy SAML	4
4	Znane biblioteki dla realizacji SAML 2.0.....	5
5	Inne specyfikacje określające SSO	6
6	Wykorzystanie standardów technologicznych w produkcie DRACO	7
7	Diagram sekwencji w DRACO.....	8
8	Szczegółowe przypadki użycia.....	9
8.1	Autoryzacja użytkownika	9
8.2	Autoryzacja systemu w imieniu użytkownika.....	13
8.3	Usunięcie kontekstu bezpieczeństwa.....	15

1 Wstęp

System ePUAP będący platformą dostępu do usług publicznych zarówno w pierwszej fazie jak i dalszym rozwoju ma za zadanie integrować systemy wykorzystywane przez administrację publiczną. Jedną z podstawowych cech projektu ePUAP jest możliwość realizacji jednokrotnego logowania (Single Sign On) użytkowników do heterogenicznych systemów podmiotów publicznych. Warunkiem realizacji rozwiązania jest oparcie się o standard spełniający wymagania otwartości, powszechnie wykorzystywany, testowany i rozwijany przez niezależną międzynarodową organizację.

2 Co to jest SAML 2.0

Specyfikacja Security Assertion Markup Language (SAML) została opracowana przez OASIS¹ dla realizacji uwierzytelnienia i przekazywania tożsamości oraz atrybutów tożsamości pomiędzy systemami komputerowymi. Ma zastosowanie w środowiskach i systemach, w których proces uwierzytelnienia jest niezależny od pozostałych komponentów – stanowi usługę dla systemu.

Dotychczas opracowano i zatwierdzono następujące wersje specyfikacji SAML: 1.0 5 (listopad 2002), 1.1 (wrzesień 2003) i 2.0 (marzec 2005). Standardy SAML 1.0 i 1.1 definiują mechanizmy żądania i przesyłania zwrótnego kilku rodzajów komunikatów (asercji): asercji dotyczących uwierzytelnienia, asercji dotyczących atrybutów i asercji dotyczących decyzji. Standard SAML 2.0 został rozbudowany o dodatkowe atrybuty bezpieczeństwa oraz możliwość przekazywania tożsamości użytkownika pomiędzy systemami korzystającymi z różnych technologii bezpieczeństwa.

¹ OASIS, Organization for the Advancement of Structured Information Standards, to międzynarodowe konsorcjum o charakterze non-profit, zajmujące się rozwojem standardów e-biznesu, w tym także standardów sieciowych. Konsorcjum powstało w 1993 roku pod nazwą SGML Open, w celu promocji języka SGML, zmiana nazwy nastąpiła w 1998 roku, aby odzwierciedlić poszerzenie obszaru działań organizacji. W skład OASIS wchodzi ponad 5 tysięcy podmiotów z ponad 100 krajów, w tym ponad 600 organizacji. Proces podejmowania decyzji przez członków konsorcjum jest otwarty i demokratyczny. Prace odbywają się obecnie między innymi w takich działach jak usługi sieciowe, e-handel, bezpieczeństwo, prawo i administracja, aplikacje, dokumenty, przetwarzanie XML czy zgodność i współpraca.

3 Podstawowe cechy SAML

- Specyfikacja SAML jest standardem otwartym, bezpłatnie udostępnianym oraz utrzymywanym i rozwijanym przez niezależną międzynarodową organizację OASiS. Utrzymywanie tego standardu przez niezależną instytucję gwarantuje otwartość standardu w przyszłości.
- Specyfikacja SAML jest rozwijana i testowana przez kilkadziesiąt ośrodków technologicznych (firm i instytucji naukowych od 7 lat). W ramach tych prac została zaimplementowana w wielu rozwiązaniach informatycznych i językach programowania.
- Procesy dostępu (uwierzytelnienia, autoryzacji i przekazania tożsamości między systemami) są podstawowym elementem podlegającym obserwacji hackerów i częstym atakiem. SAML ze względu na długą historię, nowe wersje i wiele implementacji był poddany testom i wielokrotnym próbom ataku.
- Specyfikacja SAML została stworzona dla systemów o różnej budowie, różnych technologiach uwierzytelnienia. Wiele rozwiązań aktualnie dostępnych na rynku ma zaimplementowane scenariusze i możliwości integracji z innymi systemami w oparciu o standard SAML, występujące biblioteki dla popularnych języków oprogramowania i rozwiązań tj. serwery WWW i serwery aplikacji.
- Każde rozwiązanie informatyczne podlega naturalnej ewolucji. Specyfikacja SAML będzie się rozwijała i będzie dostosowywana do nowych potrzeb. Jej rozwój będzie ściśle kontrolowany oraz będzie zakładał wykorzystanie wcześniejszych doświadczeń i standardów.

4 Znane biblioteki dla realizacji SAML 2.0

Specyfikacja SAML 2.0 została zaimplementowana w kilkudziesięciu znanych i dostępnych bibliotekach. Poniżej zostały wymienione biblioteki, które zostały poddane specjalistycznym testom zgodności ze specyfikacją. Większość z tych bibliotek jest udostępniona na bazie licencji otwartej.

- OpenSAML – biblioteka dla języków C++ i Java – dostępna na licencji otwartej i dystrybuowana wraz z oprogramowaniem Apache 2.0
- Shibboleth – biblioteka dla języków Java i C++ – dostępna na licencji otwartej
- OpenSSO – rozwiązanie J2EE dla aplikacji i serwerów Web
- PingFederate
- Sun Federated Access Manager
- Symlabs Federated Identity Suite realizuje wsparcie dla wielu rozwiązań, realizuje wszystkie profile SAML 2.0
- ZXID.org Identity Management – implementuje SAML 2.0. w C. Umożliwia wykorzystanie mechanizmów PHP, Perl i Java

5 Inne specyfikacje określające SSO

Poza specyfikacją SAML 2.0 są dostępne następujące specyfikacje realizujące funkcje jednokrotnego logowania oraz przekazania tożsamości:

- Specyfikacja WS-Trust została opracowana przez IBM, Microsoft. Nie została ona jednak jeszcze zatwierdzona jako standard ani przez żadną niezależną organizację. Zakres tej specyfikacji pokrywa się częściowo ze specyfikacją SAML. Podobnie jak w SAML, w WS-Trust zdefiniowano sposoby uwierzytelniania dostępu do usług przy pomocy trzeciej strony pełniącej funkcję ośrodka autoryzacji.
- Specyfikacja WS-Federation została opracowana przez IBM, Microsoft. Jej najnowsza wersja ukazała się lipcu 2003. Umożliwia ona stworzenie mechanizmów tłumaczenia np. żetonów bezpieczeństwa stosowanych przez różne systemy.
- Standard ID-WSF został opracowany przez Liberty Alliance w celu rozwiązywania podobnych problemów co WS-Federation i częściowo się z nim pokrywa. W przeciwieństwie do WS-Federation, który opiera się na innych standardach IBMa i Microsoftu (np. WS-Trust), ID-WSF jest oparty o standardy SAML, WS-Security i SSL.

6 Wykorzystanie standardów technologicznych w produkcji DRACO

Zadania systemu DRACO jako komponentu bezpieczeństwa w systemie ePUAP są następujące:

- uwierzytelnianie użytkowników końcowych korzystających z przeglądarek internetowych
- uwierzytelnianie systemów zewnętrznych łączących się z systemem ePUAP
- autoryzacja systemów oraz użytkowników do zasobów

Standardy wykorzystywane w produkcji bazują na wytycznych pochodzących z rozporządzenia w sprawie minimalnych wymagań dla systemów teleinformatycznych podmiotów publicznych – standard te wykorzystywane są w następujący sposób:

- Komunikacja użytkowników → portal WWW – wykorzystanie protokołu SSL/TLS
- Uwierzytelnienie systemów zewnętrznych do system ePUAP – wykorzystanie SSL
- Komunikacja pomiędzy systemem ePUAP, a systemami zewnętrznymi – wykorzystanie IPSec

Cześć funkcjonalności wymaganej w SIWZ dotyczącej funkcjonalności SSO oraz wymaganego rozszerzonego modelu uwierzytelniania oraz autoryzacji wymusza konieczność skorzystania z dodatkowych standardów technologicznych nie objętych rozporządzeniem w sprawie minimalnych wymagań dla systemów teleinformatycznych. Jednakże stanowiących otwarte i powszechnie używane standardy.

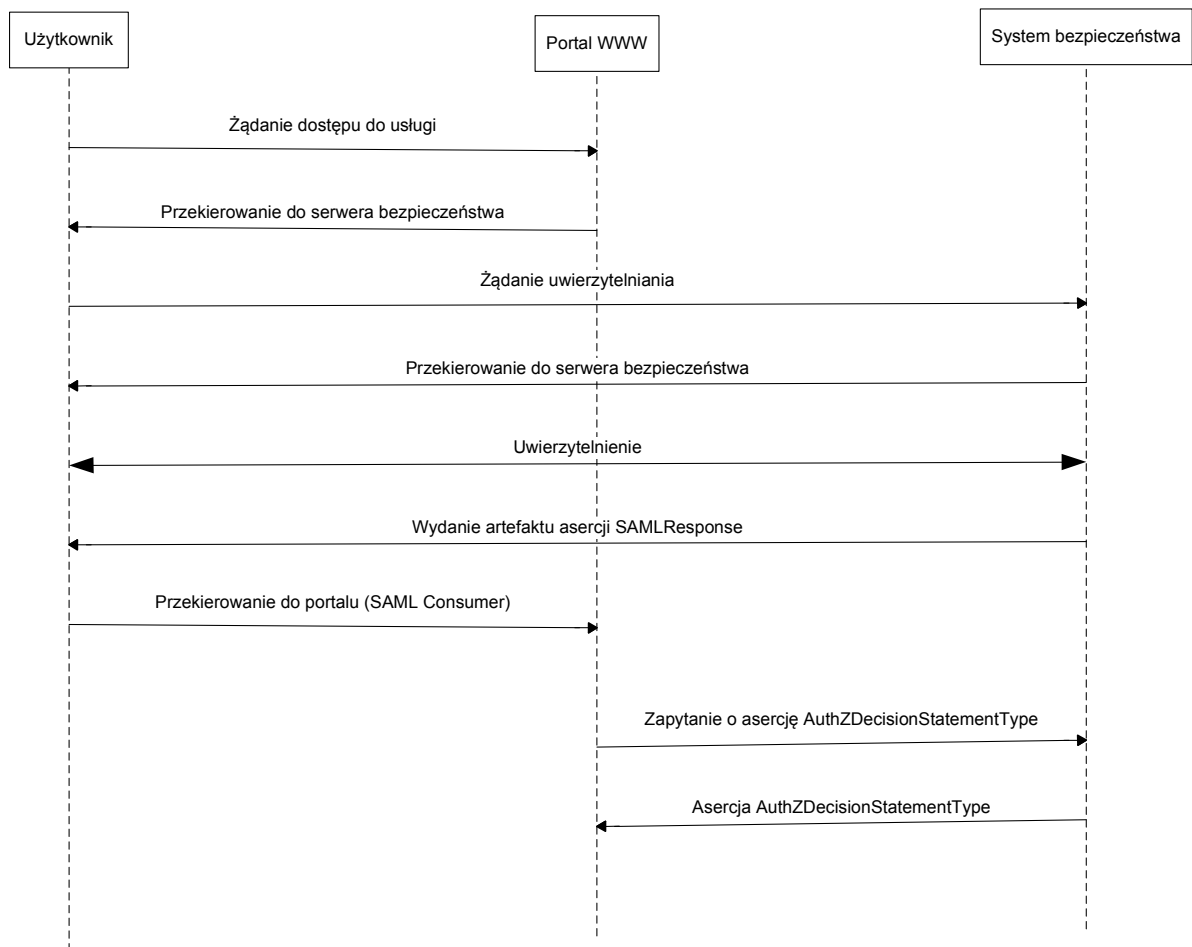
Dodatkowo system DRACO korzysta z następujących standardów:

- WS-Security - w kontekście uwierzytelniania oraz autoryzacji systemów zewnętrznych względem systemu ePUAP .
- SAML 2.0 - w kontekście uwierzytelnienia użytkowników oraz autoryzacji użytkowników portali WWW – zastosowanie asercji SAML 2.0.

System DRACO umożliwia wykorzystanie SAML 2.0 również w kontekście uwierzytelnienia oraz autoryzacji systemu zewnętrznego wobec ePUAP, co wiąże się z implementacją standardu SAML po stronie systemu uwierzytelniającego się (usługodawcy) – zgodnie ze scenariuszami opisanymi w rozdziałach poniżej.

7 Diagram sekwencji w DRACO

Poniższy rysunek przedstawia diagram sekwencji związany z protokołem uwierzytelniania użytkownika:



8 Szczegółowe przypadki użycia

8.1 Autoryzacja użytkownika

Nazwa przypadku użycia: PD_UC_21_autoryzacja_użytkownika IIB, III

PD_UC_21_autoryzacja_użytkownika IIB, III

Warunki początkowe:

Użytkownik próbuje odwołać się do określonego URLa, pod którym działająca aplikacja stwierdza, iż dany użytkownik nie posiada identyfikatora TGSID w tej sesji aplikacyjnej dla tego URL.

Użytkownik zostaje przekierowany przez aplikację pod danym URL na Usługę Autoryzacyjną z załączonym parametrem określającym pierwotny URL oraz identyfikatorami aplikacji, do których wymagany jest dostęp

SAML 2

Alternatywnie użytkownik zostaje przekierowany na Usługę Autoryzacyjną za pomocą HTTP Redirect Binding zgodnie z Web Browser SSO Profile.

Wynik końcowy:

Decyzja autoryzacyjna dla uwierzytelnionego użytkownika

Usługa bezpieczeństwa zwraca informacje o przyczynie odmowy autoryzacji (wyczerpanie limitów, aplikacja spoza ePUAP do uwierzytelnienia za pomocą WS-Security użyła innego certyfikatu niż ten powiązany z aplikacją).

SAML 2

Odpowiedź na AuthnRequest jako HTTP Artifact Binding.

Reguły:

etap II

Uprawnienie może mieć ustawioną flagę, która oznacza, że jest ono dostępne dla wszystkich uwierzytelnionych użytkowników. Uwzględniane to jest przy pytaniu autoryzacyjnym.

etap III

Od aplikacji zależy czy będzie wykorzystywała mechanizmy SAML albo natywne Draco.

Usługa ustawia początkową wartość licznika po zakończeniu zadanego interwału.

Zliczanie globalnej ilości wywołań uprawnień w zadanym interwale czasu (zgodnie z konfiguracją uprawnień) jest odpowiedzialnością aplikacji

udostępniającej chroniony zasób, a nie podsystemu bezpieczeństwa.

W przypadku, gdy dany podmiot ma przypisane uprawnienie ze zdefiniowanymi limitami wynikające z kilku ról globalnych, stosowana jest największa wartość limitu.

Limity wykorzystania uprawnień są zliczane per organizacja.

W przypadku, gdy aplikacja wywołuje zapytanie autoryzacyjne dla nieuwierzytelnionego użytkownika, do zasobu wymagającego Visual Challenge będzie ona dwukrotnie przekierowana (raz na Usługę Uwierzytelniającą, raz na VC).

<modyfikacja>

Upewnienie może mieć flagę, że jest dostępne dla użytkowników posiadających zaufany profil konta lub działających w organizacji posiadającej zaufany profil podmiotu. Uwzględniane to jest przy pytaniu autoryzacyjnym.

Agent wskazuje jaki certyfikat jest powiązany z daną aplikacją.

<modyfikacja>

SAML 2

Przy HTTP Redirect Binding będzie obsługiwany jedynie encoding urn:oasis:names:tc:SAML:2.0:bindings:URL-Encoding:DEFLATE.

Pierwotny URL będzie przekazywany jako AssertionConsumerServiceURL z AuthnRequest, a identyfikatory aplikacji jako Issuer (<saml:Issuer><saml:NameID NameQualifier="hetman.epuap.gov.pl" format="urn:oasis:names:tc:SAML:1.1:nameid-format:unspecified">...)

.

Odpowiedź na AuthnRequest jako HTTP Artifact Binding (przekazanie TGSID).

Odpytanie o asercję przy pomocy Artifact Resolution Protocol – komunikat ArtifactResolve zawierający Artifact mający wartość TGSID. Odpowiedź – komunikat ArtifactResponse nie jest podpisany i zawiera asercję AuthnStatement o

identyfikatorze ID z wartością TGSID.

AuthnStatement zawiera:

element Subject zawierający login użytkownika (<saml:Subject><saml:NameID NameQualifier="hetman.epuap.gov.pl" format="urn:oasis:names:tc:SAML:1.1:nameid-format:unspecified">...)

- element SubjectConfirmation typu „urn:oasis:names:tc:SAML:2.0:cm:bearer” zawierający SubjectConfirmationData z elementami InResponseTo zawierającym identyfikator requestu, Recipient będący wartością AssertionConsumerServiceURL z AuthnRequest. oraz NotOnOrAfter zawierający czas ważności odpowiedzi.

- element AuthnContext zawierający saml:AuthnContextClassRef> ustawiony na „URI: urn:oasis:names:tc:SAML:2.0:ac:classes>Password” lub „URI: urn:oasis:names:tc:SAML:2.0:ac:classes:X509”

- <saml:Issuer>https://hetman.euap.gov.pl</saml:Issuer> - nazwa issuera może ulec zmianie

Przebieg podstawowy - dla użytkownika bez kontekstu bezpieczeństwa (użytkownika):

1. Wywołanie - wywołanie PD_UC_20_Uwierzytelnij

2. Użytkownik jest przekierowany na pierwotny URL z załączonym parametrem określającym wartość identyfikatora TGSID.

SAML 2

Alternatywnie użytkownik jest przekierowany na pierwotny URL jako HTTP Artifact Binding będący częścią Web Browser SSO Profile. Aplikacja wykorzystuje Artifact Resolution Profile do pobrania odpowiedzi uwierzytelniającej (pytanie przy użyciu elementu AssertionIDRequest z wartością TGSID) z asercją AuthnStatement przy użyciu soap binding.

Krok 3,4 i 5 są wykonywane dla każdej pary aplikacja-uprawnienie.

3. Aplikacja odczytuje identyfikator TGSID i na jego podstawie zadaje pytanie autoryzacyjne do Usługi Bezpieczeństwa jako parametry podając identyfikator TGSID oraz wymaganą aplikację i uprawnienie.

SAML 2

Aplikacja zadaje pytanie autoryzacyjne za pomocą AuthzDecisionQuery w SOAP Binding. Komunikat AuthzDecisionQuery zawiera element Resource będący identyfikatorem aplikacji, Action zawierający nazwę uprawnienia (<action namespace="hetman.epuap.gov.pl" ...) oraz Evidence zawierający TGSID w elemencie AssertionIDRef.

<modyfikacja>

4. W przypadku, gdy usługa jest spoza ePUAP, usługa bezpieczeństwa sprawdza czy certyfikat usługi S2 użyty do uwierzytelnienia się do usługi bezpieczeństwa (za pomocą WS-Security) w celu zadania pytania, jest powiązany z aplikacją, do której odnosi się zapytanie.

</modyfikacja>

5. Usługa bezpieczeństwa zwraca decyzję autoryzacyjną, czas ważności decyzji, ilość możliwych wykorzystania uprawnień, informację czy uprawnienie wymaga interakcji z człowiekiem. Usługa bezpieczeństwa zmniejsza aktualny licznik dla organizacji o wartość zwróconą przez zapytanie autoryzacyjne. Wartość ta jest równa ilości możliwych wykorzystania uprawnień.

6. W przypadku, gdy uprawnienie wymaga interakcji z człowiekiem następuje wywołanie PD_UC_62 weryfikacja Visual Challenge.

SAML 2

Alternatywnie usługa bezpieczeństwa zwraca decyzję autoryzacyjną jako Assertion z AuthzDecisionStatement zawierającą tylko Element Resource o wartości identyfikatora aplikacji, Action o wartości uprawnienia oraz Decision . Ilość możliwych użyć

uprawnienia oraz VC określa element Condition z Assertion (będzie to rozszerzenie typu podstawowego).

7. Aplikacja cache'uje odpowiedź na czas określony w kroku 4, aby w przypadku ponownej próby dostępu użytkownika do danej aplikacji i uprawnienia nie było konieczności ponownej interakcji z systemem bezpieczeństwa. Ponadto usługa otrzymany identyfikator TGSID powinna przepisać do Cookie lub użyć metody hidden input.

Przebieg alternatywny: - dla użytkownika z kontekstem bezpieczeństwa (użytkownika):

1. Użytkownik jest przekierowany na pierwotny URL z załączonym parametrem określającym wartość identyfikatora TGSID.

Krok 2,3 i 4 są wykonywane dla każdej pary aplikacja-uprawnienie.

2. Aplikacja odczytuje identyfikator TGSID i na jego podstawie zadaje pytanie autoryzacyjne do Usługi Bezpieczeństwa jako parametry podając identyfikator TGSID oraz wymaganą aplikację i uprawnienie.

3. Usługa bezpieczeństwa zwraca decyzję autoryzacyjną, czas ważności decyzji, ilość możliwych wykorzystania uprawnień, informację czy uprawnienia wymaga interakcji z człowiekiem. Usługa bezpieczeństwa zmniejsza aktualny licznik dla organizacji o wartość zwróconą przez zapytanie autoryzacyjne. Wartość ta jest równa ilości możliwych wykorzystania uprawnień.

4. W przypadku, gdy uprawnienie wymaga interakcji z człowiekiem następuje wywołanie PD_UC_62 weryfikacja Visual Challenge.

5. Aplikacja cache'uje odpowiedź na czas określony w kroku 3, aby w przypadku ponownej próby dostępu użytkownika do danej aplikacji i uprawnienia nie było konieczności ponownej interakcji z systemem bezpieczeństwa. Ponadto usługa otrzymany identyfikator TGSID powinna przepisać do Cookie lub użyć metody hidden input.

Przebieg alternatywny - odmowna odpowiedź autoryzacyjna dla użytkownika bez kontekstu bezpieczeństwa (użytkownika):

1. Wywołanie - wywołanie PD_UC_20_Uwierzytelnij

2. Użytkownik jest przekierowany na pierwotny URL z załączonym parametrem określającym wartość identyfikatora TGSID.

Krok 3,4 i 5 są wykonywane dla każdej pary aplikacja-uprawnienie.

3. Aplikacja odczytuje identyfikator TGSID i na jego podstawie zadaje pytanie autoryzacyjne do Usługi Bezpieczeństwa jako parametry podając identyfikator TGSID oraz wymaganą aplikację i uprawnienie.

4. Usługa bezpieczeństwa zwraca odmowną decyzję autoryzacyjną

5. Wywołanie PD_UC_41 Obsługa odmowy autoryzacji

Przebieg alternatywny - odmowna odpowiedź autoryzacyjna w wyniku wyczerpania kwoty:

1. Użytkownik jest przekierowany na pierwotny URL z załączonym parametrem określającym wartość identyfikatora TGSID.

Krok 2,3 i 4 są wykonywane dla każdej pary aplikacja-uprawnienie.

2. Aplikacja odczytuje identyfikator TGSID i na jego podstawie zadaje pytanie autoryzacyjne do Usługi Bezpieczeństwa jako parametry podając identyfikator TGSID oraz wymaganą aplikację i uprawnienie.

3. Usługa bezpieczeństwa zwraca odmowną decyzję autoryzacyjną, czas ważności odpowiedzi oraz informację, że odmowa wynika z wyczerpania limitu

4. Aplikacja wyświetla odpowiedni komunikat użytkownikowi

8.2 Autoryzacja systemu w imieniu użytkownika

Nazwa przypadku użycia: **PD_UC_25_ autoryzacja systemu w imieniu użytkownika III**

PD_UC_25_ autoryzacja systemu w imieniu użytkownika III

PD_UC_25_ autoryzacja systemu w imieniu użytkownika III

Warunki początkowe:

Aplikacja WWW wykorzystująca usługę S1.

W przypadku, gdy jako S2 występuje ePuap, komunikacja z usługi S1 do S2 odbywa się w nawiązanym przez S1 tunelu SSL (co umożliwia potwierdzenie tożsamości ePUAPu); wszystkie komunikaty z S1 do S2 są podpisane WS-Security, co potwierdza tożsamość systemu S1.

Wymagana odpowiednia konfiguracja usługi S2 – umieszczenie w jej konfiguracji certyfikatów akceptowanych przez nią CA (zaufanych przez ePUAP).

W przypadku, gdy ePuap występuje jako S1 (nawiązuje komunikację do systemu S2), uwierzytelnienie ePUAPu przez S2 odbywa się w oparciu o WS-Security. Protokół nie zapewnia uwierzytelnienia systemu S2 (można to zapewnić za pomocą VPN).

Wynik końcowy:

Decyzja autoryzacyjna

Reguły:

Upewnienie może mieć ustawiona flagę, która oznacza, że jest ono dostępne dla wszystkich uwierzytelnionych użytkowników. Uwzględniane to jest przy pytaniu autoryzacyjnym.

Usługa S2 sama zlicza ilość wywołań poszczególnych uprawnień (kwoty globalne).

Usługa ustawia początkową wartość licznika po zakończeniu zadanego interwału.

SAML 2

Pytanie autoryzacyjne realizowane jako AuthzDecisionQuery po SOAP Binding.

TGSID przekazywane jako referencja do asercji uwierzytelniającej (tej która zawiera AuthnStatement) w elemencie Evidence.

Aplikacja przekazywana jako atrybut Resource, a uprawnienie jako element Action.

Odpowiedź zwracana w Assertion z AuthzDecisionStatement. Decyzja zwracana w atrybucie Decision, a dodatkowe informacje w Action.

Ustawiane będą tylko wymagane elementy i atrybuty oprócz wymienionych.

Pytanie autoryzacyjne o uprawnienia aplikacji A dostępne tylko dla aplikacji A. Realizacja przez podpis WS-Security

certyfikatem odpowiadającym aplikacji A.

Uprawnienie może mieć flagę, że jest dostępne dla użytkowników posiadających zaufany profil konta lub działających w organizacji posiadającej zaufany profil podmiotu. Uwzględniane to jest przy pytaniu autoryzacyjnym.

Agent wskazuje jaki certyfikat jest powiązany z daną aplikacją.

Przebieg podstawowy:

1. Usługa S2 odczytuje wartość TGSID przekazaną przez usługę S1 i na jej podstawie zadaje pytanie autoryzacyjne do Usługi Bezpieczeństwa jako parametry podając TGSID oraz żadaną przez S1 aplikację i uprawnienie.

2. W przypadku, gdy usługą S2 nie jest ePUAP, usługa bezpieczeństwa sprawdza czy certyfikat usługi S2 użyty do uwierzytelnienia się do usługi bezpieczeństwa (za pomocą WS-Security) w celu zadania pytania, jest powiązany z aplikacją, do której odnosi się zapytanie.

3. Usługa bezpieczeństwa zwraca S2 decyzję autoryzacyjną, czas ważności decyzji, ilość możliwych wykorzystania uprawnień. Usługa bezpieczeństwa zmniejsza aktualną wartość licznika dla organizacji.

4. Usługa S2 cache'uje odpowiedź na czas z kroku 2, aby w przypadku ponownej próby dostępu systemu S1 do danego zasobu nie musieć ponownie przeprowadzać interakcji z systemem bezpieczeństwa.

SAML 2

Przebieg alternatywny:

1. Usługa S2 odczytuje wartość asercji odpowiadającej TGID przekazaną przez usługę S1 (<soap:header><TGSID>wartośćTGSID</TGSID> ... </soap:header>) i na jej podstawie zadaje pytanie autoryzacyjne do Usługi Bezpieczeństwa zgodnie z AuthzDecisionQuery zawierającym referencję asercji TGSID oraz żadaną przez S1 aplikację i uprawnienie.

2,3. Analogicznie do przebiegu podstawowego, tylko zwrócenie informacji autoryzacyjnej jako odpowiedzi na AuthzDecisionQuery.

8.3 Usunięcie kontekstu bezpieczeństwa

Nazwa przypadku użycia: PD_UC_42_Usuń_kontekst_bezpieczeństwa

Unieważnienie aktualnego kontekstu bezpieczeństwa (użytkownika)

Warunki początkowe:

Kontekst bezpieczeństwa

Wynik końcowy:

Unieważniony kontekst bezpieczeństwa

Reguły:

SAML 2

Wykorzystanie Single Logout Protocol po HTTP Redirect Binding.

Przy HTTP Redirect Binding będzie obsługiwany jedynie encoding `urn:oasis:names:tc:SAML:2.0:bindings:URL-Encoding:DEFLATE`.

Ustawiane są tylko wymagane elementy i atrybuty SAML.

Przez inne podsystemy usunięcie kontekstu będzie widziane tylko jako odmowne odpowiedzi na pytania autoryzacyjne.

Przebieg podstawowy:

1. Usunięcie kontekstu bezpieczeństwa (użytkownika).

SAML 2

Przebieg alternatywny:

1. Usunięcie kontekstu realizowane jako Single Logout Protocol.