centralny ośrodek informatyki

INSTRUKCJA DLA INTEGRATORA

WERYFIKACJA PODPISANEGO DOKUMENTU

Spis treści

COI - Informacja publiczna

1 Historia zmian

Wersja	Data	Opis
1.0	14.05.2026	Utworzenie dokumentu na podstawie rozdziału 3.6 z dokumentu Instrukcja dla integratora Profilu Zaufanego v 3.08.
1.1	15.05.2026	Aktualizacja dokumentu w zakresie rozdziału „Dostęp do usług sieciowych weryfikacji podpisanych dokumentów”.
1.2	18.05.2026	Aktualizacja dokumentu w zakresie usługi weryfikacji dokumentów PDF.
1.3	3.07.2026	Usunięcie z dokumentu treści związanych z usługą TpPdfSigning.verifySignedDocument

2 Cel i zakres dokumentu

Niniejszy dokument opisuje usługi sieciowe umożliwiające weryfikację podpisanych dokumentów na poziomie technicznym. Jest przeznaczony jest dla twórców systemów integrujących się z systemem **Węzeł Podpisu**. Usługi weryfikacji podpisu służą uzyskaniu informacji o poprawności złożonego podpisu.

Uwaga: aktualnie zintegrowane systemy nie wymagają zmian.

2.1 Słownik pojęć i skrótów

Pojęcie/skrót	Znaczenie
System Węzeł Podpisu	System umożliwiający złożenie podpisu zaufanego, podpisu kwalifikowanego przy użyciu certyfikatu kwalifikowanego lub podpisu osobistego z wykorzystaniem certyfikatu z e-dowodu.
Profil Zaufany	Środek identyfikacji elektronicznej zawierający zestaw danych identyfikujących i opisujących osobę fizyczną, która posiada pełną albo ograniczoną zdolność do czynności prawnych, który został wydany w sposób, o którym mowa w art. 20c albo art. 20cb ustawy o informatyzacji działalności podmiotów realizujących zadania publiczne (Dz.U.2005 nr 64, poz.565; t.j. Dz. U. 2021 poz. 670).
Podpis zaufany	Podpis elektroniczny, którego autentyczność i integralność są zapewniane przy użyciu pieczęci elektronicznej ministra właściwego do spraw informatyzacji, zawierający: 1. dane identyfikujące osobę, ustalone na podstawie środka identyfikacji elektronicznej wydanego w systemie, o którym mowa w art. 20aa pkt 1 ustawy o informatyzacji działalności podmiotów realizujących zadania publiczne, obejmujące: • imię (imiona), • nazwisko, • numer PESEL, 2. identyfikator środka identyfikacji elektronicznej, przy użyciu którego został złożony, 3. czas jego złożenia.
Podpis osobisty	Jest to zaawansowany podpis elektroniczny. Prawdziwość danych posiadacza podpisu potwierdza certyfikat podpisu osobistego, zawierający imię (imiona), nazwisko, obywatelstwo oraz numer PESEL. Podpis osobisty ma taki sam skutek prawny jak podpis własnoręczny. Podpis osobisty może być wykorzystywany również w kontaktach z podmiotami innymi niż publiczne, ale tylko, jeżeli obie strony wyrażą na to zgodę. Podpis osobisty może służyć m.in. do złożenia podpisu dokumentów elektronicznych wysyłanych do urzędu.

Pojęcie/skrót	Znaczenie
Podpis kwalifikowany	Jest to zaawansowany podpis elektroniczny, który jest składany za pomocą kwalifikowanego urządzenia do składania podpisu elektronicznego i który opiera się na kwalifikowanym certyfikacie podpisu elektronicznego (Dz.U.UE.910/2014). Podpis kwalifikowany to powszechnie używana nazwa bezpiecznego podpisu elektronicznego weryfikowanego za pomocą ważnego kwalifikowanego certyfikatu. Z uwagi na bezpieczeństwo obrotu prawnego dokumenty przedkładane usługodawcy przez usługobiorcę za pośrednictwem środków komunikacji elektronicznej muszą być zabezpieczone przed niekontrolowaną modyfikacją, a więc muszą być elektronicznie podpisane. Podpis kwalifikowany jest autoryzowanym sposobem potwierdzenia tożsamości. Centra Certyfikacji wydają certyfikaty kwalifikowane służące do złożenia podpisu kwalifikowanego. Dokument w ten sposób podpisany jest prawnie wiążącym.
e-dowód	Dowód osobisty (dokument tożsamości) posiadający warstwę elektroniczną, umożliwiającą jego posiadaczowi uwierzytelnienie w usługach online za pomocą profilu osobistego oraz potwierdzenie obecności w określonym czasie i miejscu.
System zewnętrzny	System używający usług sieciowych systemu **Węzeł Podpisu.**
System PZ	Systemu Profil Zaufany
Administrator systemu PZ	Użytkownik systemu Profil Zaufany posiadający m.in. uprawnienie do zarządzania słownikiem systemów zewnętrznych.
REST	Representational State Transfer, to styl architektoniczny do tworzenia usług sieciowych, który umożliwia komunikację między różnymi systemami komputerowymi poprzez Internet. Jest to zestaw zasad i wytycznych dla tworzenia skalowalnych i wydajnych usług, które wykorzystują protokół HTTP i operacje na zasobach (np. pobieranie, dodawanie, aktualizacja, usuwanie danych).
SOAP	Simple Object Access Protocol – protokół wymiany informacji ustrukturalizowanej w usłudze sieciowej. (http://www.w3.org/TR/soap).
WSDL	Web Services Description Language (http://www.w3.org/TR/wsdl).
JSON	JavaScript Object Notation, lekki format wymiany danych komputerowych. JSON jest formatem tekstowym, bazującym na podzbiorze języka JavaScript.
Operacja usługi sieciowej	Akcja SOAP w znaczeniu stosowanym w WSDL.
WS-Security	Web Services Security – rozszerzenie SOAP stosowane w celu zabezpieczenia usług sieciowych. (http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=wss)

COI - Informacja publiczna

Pojęcie/skrót	Znaczenie
XAdES	Format podpisu elektronicznego, który służy do podpisywania dokumentów w formacie XML. Specyfikacja techniczna wskazana w Decyzji wykonawczej Komisji (UE) 2015/1506 z dnia 8 września 2015 r. ustanawiającej specyfikacje dotyczące formatów zaawansowanych podpisów elektronicznych oraz zaawansowanych pieczęci elektronicznych, które mają być uznane przez podmioty sektora publicznego to ETSI TS 103171 v.2.1.1. W usłudze są dostępne dwa rodzaje podpisu XAdES: otoczony i otaczający. Różnią się one strukturą podpisanego pliku.
XAdES otoczony (enveloped)	Podpis znajduje się w strukturze dokumentu – jest „otoczony" treścią dokumentu. Jeśli taki dokument podpisuje więcej osób, ich podpisy zostaną umieszczone w strukturze dokumentu obok siebie i mogą zostać odczytane jednocześnie.
XAdES otaczający (enveloping)	Treść dokumentu znajduje się w strukturze podpisu – jest „otoczona" podpisem. Jeśli taki dokument podpisuje więcej osób, struktura każdego kolejnego podpisu będzie „otaczała" treść dokumentu z wcześniejszym podpisem. Oznacza to, że podczas odczytywania pliku nie zostaną wyświetlone wszystkie podpisy jednocześnie.
PAdES	Format podpisu elektronicznego, który służy do podpisywania dokumentów w formacie PDF. Specyfikacja techniczna wskazana w Decyzji wykonawczej Komisji (UE) 2015/1506 z dnia 8 września 2015 r. ustanawiającej specyfikacje dotyczące formatów zaawansowanych podpisów elektronicznych oraz zaawansowanych pieczęci elektronicznych, które mają być uznane przez podmioty sektora publicznego to ETSI TS103172 v.2.2.2.

3 Dostęp do usług sieciowych weryfikacji podpisanych dokumentów

Wszystkie usługi sieciowe systemu Profil Zaufany zabezpieczone są za pomocą protokołu WS-Security. Uzyskanie dostępu do usługi przez system zewnętrzny związane jest ze spełnieniem wszystkich poniższych warunków:

- Żądanie wysyłane do systemu musi być podpisane certyfikatem klienckim. Podpis musi być zgodny z protokołem WS-Security.
- System zewnętrzny musi być wpisany na listę znanych systemów zewnętrznych w systemie Profil Zaufany.
- Certyfikat kliencki użyty przez system zewnętrzny do podpisania żądania musi być dodany przez administratora systemu Profil Zaufany do listy certyfikatów systemu zewnętrznego.
- System zewnętrzny musi być oznaczony jako aktywny w systemie Profil Zaufany.
- System zewnętrzny musi mieć przyznane uprawnienie do wywoływania operacji usługi sieciowej w systemie Profil Zaufany.

Dla zwiększenia bezpieczeństwa, system Profil Zaufany przy konstruowaniu odpowiedzi nie ujawnia, który z powyższych warunków nie został spełniony przez system zewnętrzny. W każdym przypadku zwracana jest odpowiedź podobna do poniższej:

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
<soap:Header/>
  <soap:Body>
    <soap:Fault>
      <faultcode>soap:Client</faultcode>
      <faultstring>Brak uprawnień do wywołania operacji.</faultstring>
        <detail>
          <ns4:errorFault callId="2913616646816870912" responseTimestamp="2014-06-30T12:01:05.373+02:00"
xmlns:ns2="http://www.cpi.gov.pl/pz/CommonSchema" xmlns:ns4="http://www.cpi.gov.pl/pz/TpApplicationSubmissionServiceSchema">
            <ns2:code>401</ns2:code>
            <ns2:description>Brak uprawnień do wywołania operacji.</ns2:description>
          </ns4:errorFault>
        </detail>
    </soap:Fault>
  </soap:Body>
</soap:Envelope>
```

3.1 WS-Security

Każde żądanie wysyłane przez system zewnętrzny musi być podpisane zgodnie z rozszerzeniem SOAP: WS-Security. Szczegółowa specyfikacja tego rozszerzenia dostępna jest pod adresem http://www.oasis-open.org/committees/wss. System Profil Zaufany wymaga, aby w wiadomości SOAP podpisany był element <soap:Body>. System weryfikuje obecność w żądaniu binarnego tokenu bezpieczeństwa typu X509v3.

Przykładowe podpisane żądanie wygląda następująco (długie wartości elementów zakodowane w Base64 zostały skrócone dla przejrzystości):

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:tpus="http://www.cpi.gov.pl/pz/TpUserObjectsInfoServiceSchema">
  <soapenv:Header>
    <wsse:Security xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd"
xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd">
      <wsse:BinarySecurityToken EncodingType="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-soap-message-security-
1.0#Base64Binary" ValueType="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-x509-token-profile-1.0#X509v3" wsu:Id="X509-
8D58A61A83EB7640661398960091579265">MIIDZTCCAk2gAwI(...)</wsse:BinarySecurityToken>
      <ds:Signature Id="SIG-176" xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
        <ds:SignedInfo>
          <ds:CanonicalizationMethod Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#">
            <ec:InclusiveNamespaces PrefixList="soapenv tpus" xmlns:ec="http://www.w3.org/2001/10/xml-exc-c14n#"/>
          </ds:CanonicalizationMethod>
          <ds:SignatureMethod Algorithm="http://www.w3.org/2000/09/xmldsig#rsa-sha1"/>
          <ds:Reference URI="#id-175">
          <ds:Transforms>
            <ds:Transform Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#">
              <ec:InclusiveNamespaces PrefixList="tpus" xmlns:ec="http://www.w3.org/2001/10/xml-exc-c14n#"/>
            </ds:Transform>
```

```
            </ds:Transforms>
            <ds:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1"/>
              <ds:DigestValue>e3x8yAjaJoXfO058N0z05VjSP4Q=</ds:DigestValue>
            </ds:Reference>
          </ds:SignedInfo>
          <ds:SignatureValue>hN1LsCfXAgUfuMLji5Zk(…)</ds:SignatureValue>
          <ds:KeyInfo Id="KI-8D58A61A83EB7640661398960091579266">
           <wsse:SecurityTokenReference wsu:Id="STR-8D58A61A83EB7640661398960091579267">
           <wsse:Reference URI="#X509-8D58A61A83EB7640661398960091579265" ValueType="http://docs.oasis-
open.org/wss/2004/01/oasis-200401-wss-x509-token-profile-1.0#X509v3"/>
           </wsse:SecurityTokenReference>
          </ds:KeyInfo>
        </ds:Signature>
      </wsse:Security>
    </soapenv:Header>
  <soapenv:Body wsu:Id="id-175" xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd">
  <tpus:reqGetTpUserObjectsInfo callId="6347177294896046332" requestTimestamp="2014-06-30T12:01:30.128+02:00">
    <tpus:userId>user01</tpus:userId>
  </tpus:reqGetTpUserObjectsInfo>
 </soapenv:Body>
</soapenv:Envelope>
```

Odpowiedź serwera na powyższe przykładowe żądanie jest również podpisana zgodnie z protokołem WS-Security i wygląda następująco:

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
<SOAP-ENV:Header xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<wsse:Security soap:mustUnderstand="1" xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-
1.0.xsd" xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd">
<wsse:BinarySecurityToken EncodingType="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-soap-message-security-
1.0#Base64Binary" ValueType="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-x509-token-profile-1.0#X509v3" wsu:Id="X509-
8ACEA668E7B1B1E54F1398960101010217">MIIDbzCCAlegAwIBAgII(…)</wsse:BinarySecurityToken>
          <ds:Signature Id="SIG-73" xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
            <ds:SignedInfo>
            <ds:CanonicalizationMethod Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#">
              <ec:InclusiveNamespaces PrefixList="soap" xmlns:ec="http://www.w3.org/2001/10/xml-exc-c14n#"/>
            </ds:CanonicalizationMethod>
                  <ds:SignatureMethod Algorithm="http://www.w3.org/2000/09/xmldsig#rsa-sha1"/>
                    <ds:Reference URI="#Id-15963761">
                      <ds:Transforms>
                          <ds:Transform Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#">
                          <ec:InclusiveNamespaces PrefixList="" xmlns:ec="http://www.w3.org/2001/10/xml-exc-c14n#"/>
                        </ds:Transform>
                       </ds:Transforms>
                      <ds:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1"/>
                      <ds:DigestValue>azRca9fQn08cGOxF3jBi6pt2vFw=</ds:DigestValue>
                    </ds:Reference>
                  </ds:SignedInfo>
                <ds:SignatureValue>kbhy/woiLGiUrkp43Joh8Jm(…)</ds:SignatureValue>
                <ds:KeyInfo Id="KI-8ACEA668E7B1B1E54F1398960101010218">
                <wsse:SecurityTokenReference wsu:Id="STR-8ACEA668E7B1B1E54F1398960101010219">
                <wsse:Reference URI="#X509-8ACEA668E7B1B1E54F1398960101010217" ValueType="http://docs.oasis-
open.org/wss/2004/01/oasis-200401-wss-x509-token-profile-1.0#X509v3"/>
              </wsse:SecurityTokenReference>
            </ds:KeyInfo>
          </ds:Signature>
        </wsse:Security>
      </SOAP-ENV:Header>
    <soap:Body wsu:Id="Id-15963761" xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd">
  <ns3:respGetTpUserObjectsInfo callId="6347177294896046332" responseTimestamp="2014-06-30T12:01:41.004+02:00"
xmlns:ns2="http://www.cpi.gov.pl/pz/CommonSchema" xmlns:ns3="http://www.cpi.gov.pl/pz/TpUserObjectsInfoServiceSchema"/>
 </soap:Body>
</soap:Envelope>
```

W dalszej części dokumentu w przykładowych komunikatach usług sieciowych nagłówek SOAP wraz z podpisem WS-Security będzie pominięty dla przejrzystości.

3.2 Wspólny nagłówek żądania i odpowiedzi

Nowe usługi sieciowe systemu Profil Zaufany dodane w ramach wydzielenia systemu z ePUAP posiadają standardowe atrybuty dodawane do żądania i odpowiedzi, ułatwiające analizę poprawności i efektywności działania usług.

W żądaniu wymagane są przez system następujące atrybuty wymienione w tabeli

Pole	Typ	Wymagane	Opis
callId	long	Tak	Losowa liczba naturalna identyfikująca wywołanie usługi sieciowej; Dopuszczalne jest użycie przez klienta tego samego identyfikatora w różnych wywołaniach. Klient powinien jednak zadbać o możliwie największą unikalność tych identyfikatorów, np. przez losowanie identyfikatora z całego zakresu 0 – 2^63-1 aby prawdopodobieństwo powtórzenia się było jak najniższe.
requestTimestamp	dateTime	Tak	Znacznik czasu możliwie najbliższy momentowi wysłania żądania od klienta do systemu PZ. Możliwe odchylenie wartości podanej w polu requestTimestamp wynosi [maksymalne dopuszczalne przesunięcie w czasie (niedokładność) między zegarami systemów biorących udział w wymianie wiadomości protokołu SOAP] ± 3 minuty (wartość konfigurowalna).

W odpowiedzi system PZ dołącza następujące atrybuty:

Pole	Typ	Wymagane	Opis
callId	long	Tak	Identyfikator wywołania usługi sieciowej skopiowany z żądania
responseTimestamp	dateTime	Tak	Znacznik czasu możliwie najbliższy momentowi wysłania odpowiedzi od systemu PZ do klienta

Atrybuty te są dołączane przez system PZ również w przypadku zwrócenia odpowiedzi typu fault. Przykładowe atrybuty w żądaniu wyglądają następująco:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:tpus="http://www.cpi.gov.pl/pz/TpUserObjectsInfoServiceSchema">
  <soapenv:Header/>
  <soapenv:Body>
    <tpus:reqGetTpUserObjectsInfo callId="6347177294896046332" requestTimestamp="2014-06-30T12:01:30.048+02:00">
      <tpus:userId>user01</tpus:userId>
    </tpus:reqGetTpUserObjectsInfo>
  </soapenv:Body>
</soapenv:Envelope>
```

Odpowiedź serwera na powyższe żądanie wygląda następująco:

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Header/>
  <soap:Body>
    <ns3:respGetTpUserObjectsInfo callId="6347177294896046332" responseTimestamp="2014-06-30T12:01:30.868+02:00"
xmlns:ns2="http://www.cpi.gov.pl/pz/CommonSchema"
xmlns:ns3="http://www.cpi.gov.pl/pz/TpUserObjectsInfoServiceSchema"/>
  </soap:Body>
</soap:Envelope>
```

3.3 Odpowiedź informująca o błędzie

W przypadku, gdy system PZ nie jest w stanie poprawnie obsłużyć żądania, w odpowiedzi zwracany jest element typu SOAP Fault. Przykładowa odpowiedź wygląda następująco:

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Header/>
  <soap:Body>
    <soap:Fault>
      <faultcode>soap:Client</faultcode>
      <faultstring>Błąd walidacji: APPLICATION_PESEL: Wartość pola nie jest poprawnym numerem PESEL.</faultstring>
      <detail>
        <ns4:errorFault callId="5840781678820004861" responseTimestamp="2014-06-30T12:14:47.460+02:00"
xmlns:ns2="http://www.cpi.gov.pl/pz/CommonSchema"
xmlns:ns4="http://www.cpi.gov.pl/pz/TpApplicationSubmissionServiceSchema">
          <ns2:code>600</ns2:code>
          <ns2:description>Błąd walidacji: APPLICATION_PESEL: Wartość pola nie jest poprawnym numerem
PESEL.</ns2:description>
        </ns4:errorFault>
      </detail>
    </soap:Fault>
  </soap:Body>
</soap:Envelope>
```

Odpowiedź zawiera elementy wymienione w poniższej tabeli:

Element	Odbiorca	Przeznaczenie
faultcode	System zewnętrzny	Element przyjmuje następujące wartości, zgodne ze specyfikacją SOAP: • Client – oznacza że żądanie jest nieuprawnione, skonstruowane w sposób nieprawidłowy lub zawiera nieprawidłowe dane. Po otrzymaniu takiej odpowiedzi system zewnętrzny nie powinien ponawiać żądania w niezmienionej postaci, gdyż jego obsługa nigdy się nie powiedzie. • Server – oznacza że wystąpił błąd na serwerze uniemożliwiający obsługę żądania. Po otrzymaniu takiej odpowiedzi system zewnętrzny może (ale nie musi) ponowić żądanie w niezmienionej postaci natychmiast, lub po pewnym czasie, gdyż jest prawdopodobne, że jego obsługa w końcu się powiedzie.
faultstring	Administrator systemu zewnętrznego	Opis powodu nieobsłużenia żądania w postaci tekstu zrozumiałego dla człowieka; Jest przeznaczony dla administratora systemu zewnętrznego do diagnozowania błędów w komunikacji między systemami. Element nie powinien być używany do automatycznego podejmowania decyzji przez system zewnętrzny, gdyż komunikaty w nim zawarte mogą ulegać zmianie w wyniku aktualizacji oprogramowania systemu PZ.
code	System zewnętrzny	Element przyjmuje wartości właściwe dla konkretnej operacji usługi sieciowej, wymienione w opisie tej usługi. Może być użyty do automatycznego podejmowania decyzji przez system zewnętrzny.

4 Dostęp do usługi `TpSigning`

Usługa służy do weryfikacji podpisu pod dokumentami.

4.1 Operacja `verifySignedDocument`

Operacja służy do weryfikowania podpisu lub podpisów pod dokumentem XML. W odpowiedzi system PZ zwraca strukturę XML zawierającą szczegółowe informacje na temat podpisu. Żądanie składa się z następujących pól:

Pole	Typ	Wymagane	Uwagi
document	String	Tak	Podpisany dokument w formacie XML, zakodowany w Base64; Maksymalna dopuszczalna wielkość dokumentu to 25 MB.

Jeśli weryfikacja podpisu lub podpisów się powiedzie, zwracana jest struktura XML składająca się z następujących elementów:

Element	Typ	Uwagi
ValidDocumentSignature	Boolean	Informacja czy dokument jest poprawnie podpisany; Atrybut znaczenie opisuje zawartość tego pola i przyjmuje wartości „Prawidłowy” oraz „Nieprawidłowy”.
SignatureType	String	Zawsze ciąg znaków „XAdES”
GenerationTime	Date	Data i godzina wygenerowania tego dokumentu XML z informacjami na temat podpisu.
StatusInfo	Element XML	Szczegółowe informacje na temat podpisu; Element ten jest tworzony dla każdego podpisu pod dokumentem. W przypadku braku podpisu tworzony jest jeden taki element zawierający informację o braku podpisu.

Element StatusInfo składa się z następujących elementów:

Element	Typ	Uwagi
ValidSignature	Boolean	Informacja czy podpis jest prawidłowy; Atrybut znaczenie opisuje zawartość tego pola i przyjmuje wartości „Prawidłowy” oraz „Nieprawidłowy”.
VerifyStatus	Int	Status weryfikacji podpisu; Atrybut `znaczenie` opisuje zawartość tego pola. Zwracane są następujące wartości: 0 – Zgodny z dokumentem 1 – Niezgodny z dokumentem 2 – Brak załączników 3 – Niepoprawna struktura podpisu 5 – Brak podpisu
VerifySignerCert	Int	Certyfikat użyty w podpisie; Atrybut `znaczenie` opisuje zawartość tego pola. Zwracane są następujące wartości: -1 – [Brak] – brak informacji lub podpisu 0 – Ważny – certyfikat ważny 1 – Nieważny – certyfikat nieważny 2 – Unieważniony – certyfikat podpisujący unieważniony 3 – Nieznany wystawca – nie znaleziono certyfikatu wystawcy w bazie 4 – Brak OCSP lub CRL – brak odpowiedzi OCSP lub CRL 5 – Błędny – ogólny błąd certyfikatu
VerifySignerCertUsage	Int	Sposób użycia certyfikatu wykorzystanego w podpisie; Atrybut `znaczenie` opisuje zawartość tego pola. Zwracane są następujące wartości: 1 – kwalifikowany 2 – niewykorzystany 3 – logowanie 4 – Zaufana odpowiedź OCSP 5 – Generacja odpowiedzi OCSP na podstawie CRL

		6 – Przekierowanie na OCSP danego CA 7 – UPO 8 – EPO 9 – TSA Element może wskazywać na więcej niż jedno zastosowanie certyfikatu. Podane w liście wartości traktowane są jako pozycje bitu wartości w reprezentacji binarnej. Bit na dziesiątej pozycji pełni rolę pomocniczą i oznacza, że przynajmniej jedna pozycja z listy jest prawdziwa. Przykładowo wartość elementu 924 – binarnie 1100011100 oznacza, że prawdziwe są pozycje logowanie, Zaufana odpowiedź OCSP, Generacja odpowiedzi OCSP na podstawie CRL, TSA.
CommitmentType	String	Wartość pola „Commitment type” z podpisu
GracePeriod	Int	Wartość pola „Grace period” z podpisu
ParentSignatureId	String	Identyfikator podpisu nadrzędnego w przypadku kontrasygnaty
SignatureCertIssuer	String	Dane wystawcy certyfikatu
SignatureCertSerial	String	Numer seryjny certyfikatu użytego w podpisie
SignatureCertSubject	String	Dane posiadacza certyfikatu
SignatureId	String	Identyfikator podpisu
SigningTime	Date	Data i godzina podpisania dokumentu
UriID	String	Wskaźniki do podpisanych elementów
SignatureTimeStamp	Element XML	Informacje o oznaczeniu podpisu czasem; W przypadku braku oznaczenia czasem atrybut `znaczenie` przyjmuje wartość „Brak oznaczenia czasem”. W przeciwnym razie element ten tworzony jest dla każdego oznaczenia czasem. Struktura elementu opisana jest w tabeli poniżej.
ArchiveTimeStamp	Element XML	Informacje o postaci archiwalnej podpisu; W przypadku braku oznaczenia czasem atrybut `znaczenie` przyjmuje wartość „Brak postaci archiwalnej”. W przeciwnym razie tworzony jest element o strukturze opisanej w tabeli poniżej.
ZP	Element XML	Informacja o podpisie profilem zaufanym; Atrybut `czy_obecny` informuje czy dokument jest podpisany profilem zaufanym. Atrybut może przyjmować następujące wartości: true – dokument jest podpisany profilem zaufanym; Element ZP zawiera podpis profilem zaufanym false – dokument nie jest podpisany profilem zaufanym

Elementy SignatureTimeStamp i ArchiveTimeStamp mają następującą strukturę:

Element	**Typ**	**Uwagi**
TimeStampTime	Date	Czas oznaczenia podpisu
VerifyStatus	Int	Status weryfikacji oznaczenia podpisu; Atrybut `znaczenie` opisuje zawartość tego pola. Zwracane są następujące wartości: -1 – [brak] 0 – Brak znacznika 1 – Znacznik prawidłowy 2 – Znacznik nieprawidłowy 3 – Nieważny certyfikat OCSP 4 – Niezaufany certyfikat OCSP 5 – Nieważny certyfikat TSA 6 – Niezaufany certyfikat TSA

Jeśli weryfikacja podpisu lub podpisów się nie powiedzie, zwracany jest komunikat typu fault, a w nim jeden z poniższych kodów błędów:

Kod	Znaczenie	Przyczyna
401	brak uprawnień	system zewnętrzny nie jest uprawniony do wywołania operacji
500	błąd wewnętrzny	wystąpił nieoczekiwany błąd w systemie PZ
600	nieprawidłowy parametr wywołania	• pole document jest puste • dokument nie jest w formacie XML • wystąpił błąd w trakcie parsowania dokumentu, dla którego ma być przeprowadzona weryfikacja podpisu PZ • Wartość pola document zawiera nieprawidłowe kodowanie Base64.
602	przesyłany dokument jest zbyt duży	dokument w polu document przekracza dopuszczalny rozmiar

Przykładowe żądanie operacji wygląda następująco:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:sig="http://signing.ws.comarch.gov">
  <soapenv:Header/>
  <soapenv:Body>
    <sig:verifySignedDocument>
      <document>PD94bWwgdmVyc2lvbj0iMS4wIiBlbm(…)</document>
    </sig:verifySignedDocument>
  </soapenv:Body>
</soapenv:Envelope>
```

Jeśli powyższe żądanie jest prawidłowe, to odpowiedź serwera wygląda następująco:

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Header/>
  <soap:Body>
    <ns1:verifySignedDocumentResponse xmlns:ns1="http://signing.ws.comarch.gov">
      <verifySignedDocumentReturn xmlns:ns2="http://exception.ws.comarch.gov"><![CDATA[<?xml version="1.0" encoding="UTF-8"
standalone="yes"?><VerifyResult><ValidDocumentSignature
znaczenie="Prawidłowy">true</ValidDocumentSignature><SignatureType>XAdES</SignatureType><GenerationTime>2014-06-30 13:14:48
CEST</GenerationTime><StatusInfo><ValidSignature znaczenie="Prawidłowy">true</ValidSignature><VerifyStatus znaczenie="Zgodny z
dokumentem">0</VerifyStatus><VerifySignerCert znaczenie="Nieznany wystawca">3</VerifySignerCert><VerifySignerCertUsage
znaczenie="">0</VerifySignerCertUsage><CommitmentType></CommitmentType><GracePeriod>3600</GracePeriod><ParentSignatureId></
ParentSignatureId><SignatureCertIssuer C="PL" OU="Sigillum Polskie Centrum Certyfikacji Elektronicznej" O="Polska Wytwórnia Papierów
Wartościowych S.A." CN="Sigillum PCCE - CALevel3">C=PL,O=Polska Wytwórnia Papierów Wartościowych S.A.,OU=Sigillum Polskie Centrum
Certyfikacji Elektronicznej,CN=Sigillum PCCE -
CALevel3</SignatureCertIssuer><SignatureCertSerial>2930305822001951294</SignatureCertSerial><SignatureCertSubject C="PL"
O="MSWiA" CN="ePUAP ZP 1">C=PL,O=MSWiA,CN=ePUAP ZP 1</SignatureCertSubject><SignatureId>Signature-f72dc9dd-8fcf-48f3-85f5-
61a24b55ff93</SignatureId><SigningTime>2014-05-27 01:32:36 CEST</SigningTime><UriID lp="1"></UriID><UriID lp="2">#SignedProps-
f72dc9dd-8fcf-48f3-85f5-61a24b55ff93</UriID><SignatureTimeStamp znaczenie="Brak oznaczenia czasem"/><ArchiveTimeStamp
znaczenie="Brak postaci archiwalnej"/><ZP czy_obecny="true"><ppZP:PodpisZP
xmlns:os="http://crd.gov.pl/xml/schematy/osoba/2009/03/06/"
xmlns:ppZP="http://crd.gov.pl/xml/schematy/ppzp/"><ppZP:DaneZP><ppZP:DaneZPOsobyFizycznej><os:Nazwisko
rodzajCzlonu="pierwszy">Kowalski</os:Nazwisko><os:Imie>Jan</os:Imie><os:PESEL>10101010103</os:PESEL><ppZP:IdZaufanegoProfilu>12
</ppZP:IdZaufanegoProfilu><ppZP:IdKontaUzytkownikaEpuap>user01</ppZP:IdKontaUzytkownikaEpuap></ppZP:DaneZPOsobyFizycznej></p
pZP:DaneZP><ppZP:DanePodpisu><ppZP:IdKontaUzytkownikaEpuap>user01</ppZP:IdKontaUzytkownikaEpuap><ppZP:IdPolitykiAutoryzacji>
5</ppZP:IdPolitykiAutoryzacji></ppZP:DanePodpisu></ppZP:PodpisZP></ZP></StatusInfo></VerifyResult>]]></verifySignedDocumentReturn
>
    </ns1:verifySignedDocumentResponse>
  </soap:Body>
</soap:Envelope>
```

Odpowiedź serwera na powyższe żądanie w przypadku pustego elementu `document` jest następująca:

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Header/>
  <soap:Body>
    <soap:Fault>
      <faultcode>soap:Client</faultcode>
      <faultstring>Wartość pola document nie może być pusta.</faultstring>
```

```
      <detail>
        <ns2:WSSigningException xmlns:ns2="http://exception.ws.comarch.gov">
          <code>600</code>
          <errMessage>Wartość pola document nie może być pusta.</errMessage>
        </ns2:WSSigningException>
      </detail>
    </soap:Fault>
    </soap:Body>
</soap:Envelope>
```

5 Dostęp do usługi `SignatureVerification`

Usługa umożliwia weryfikację podpisanych dokumentów; usługa zachowuje pełną zgodność z usługą SignatureVerification z poprzedniej wersji Profilu Zaufanego, opisanej w Powykonawczym Projekcie Technicznym Profilu Zaufanego (wersja 01.05).

Usługa dostępna jest pod adresem https://pz.gov.pl/pz-services/SignatureVerification.

Definicja usługi znajduje się w pliku SignatureVerification.wsdl.

Uwaga: definicja usługi SignatureVerification nie przewiduje dedykowanego elementu dla kodu błędu w komunikacie typu fault. Z tego względu kody błędów umieszczane są na początku komunikatów błędu.

5.1 Operacja `verifySignature`

Operacja służy do weryfikowania podpisu lub podpisów pod dokumentem XML wraz z załącznikami. W odpowiedzi system PZ zwraca strukturę XML zawierającą szczegółowe informacje na temat podpisu. Żądanie składa się z następujących pól:

Pole	Typ	Wymagane	Uwagi
doc	String	Tak	Podpisany dokument w formacie XML, zakodowany w Base64; Maksymalna dopuszczalna wielkość dokumentu 25 MB.
attachments	Element XML	Nie	Lista elementów `Attachment`, może mieć 0 elementów

Element Attachment składa się z następujących elementów:

Pole	Typ	Wymagane	Uwagi
content	string	Tak	Załącznik uwzględniony w podpisie zakodowany w Base64; Maksymalna dopuszczalna wielkość załącznika to 25 MB.
name	string	Tak	URI załącznika podane podczas operacji podpisu

Jeśli weryfikacja podpisu lub podpisów się powiedzie, zwracana jest struktura XML taka sama jak w operacji verifySignedDocument w usłudze TpSigning. Jeśli weryfikacja podpisu lub podpisów się nie powiedzie, zwracany jest komunikat typu fault, a w nim jeden z poniższych kodów błędów:

Pole	Typ	Wymagane
401	brak uprawnień	system zewnętrzny nie jest uprawniony do wywołania operacji
500	błąd wewnętrzny	wystąpił nieoczekiwany błąd w systemie PZ
600	nieprawidłowy parametr wywołania	• parametr w polu `<doc>` lub `<content>` jest puste • plik w polu `<doc>` lub załącznik w polu `<content>` przekracza dopuszczalną wielkość • dokument nie jest w formacie XML • parametr w polu `<doc>` lub `<content>` zawiera nieprawidłowe kodowanie Base64 • wystąpił błąd w trakcie parsowania dokumentu, dla którego ma być przeprowadzona weryfikacja podpisu PZ
602	dokument jest zbyt duży	dokument lub jego załączniki przekraczają dopuszczalną wielkość

COI - Informacja publiczna

Przykładowe żądanie operacji wygląda następująco:

```
<soapenv:Envelope xmlns:soapenv=http://schemas.xmlsoap.org/soap/envelope/ xmlns:ver="http://verification.zp.epuap.gov.pl">
  <soapenv:Header/>
    <soapenv:Body>
      <ver:verifySignature>
       <doc>PD94bWwgdmVyc2lvbj0iMS4wIiBlbmNvZGluZz0iVVRGLTgiPz48YT5hPGRzOlNpZ25h(...)</doc>
       <attachments>
         <Attachment>
           <content>QUJDREVGR0hJSktMTU5PUFFSU1RVVldYWVo=</content>
          <name>załącznik.txt</name>
        </Attachment>
      </attachments>
    </ver:verifySignature>
  </soapenv:Body>
</soapenv:Envelope>
```

Jeśli powyższe żądanie jest prawidłowe to odpowiedź serwera wygląda następująco:

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Header/>
    <soap:Body>
      <ns2:verifySignatureResponse xmlns:ns2="http://verification.zp.epuap.gov.pl">
        <verifySignatureReturn><![CDATA[<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
          <VerifyResult>
            <ValidDocumentSignature znaczenie="Prawidłowy">true</ValidDocumentSignature>
            <SignatureType>XAdES</SignatureType>
            <GenerationTime>2014-06-30 12:23:37 CEST</GenerationTime>
            <StatusInfo>
              <ValidSignature znaczenie="Prawidłowy">true</ValidSignature>
              <VerifyStatus znaczenie="Zgodny z dokumentem">0</VerifyStatus>
              <VerifySignerCert znaczenie="Nieznany wystawca">3</VerifySignerCert>
              <VerifySignerCertUsage znaczenie=""
kwalifikowany="false">0</VerifySignerCertUsage><CommitmentType></CommitmentType>
               <GracePeriod>3600</GracePeriod>
               <ParentSignatureId></ParentSignatureId>
               <SignatureCertIssuer C="PL" O="Polska Wytwórnia Papierów Wartościowych S.A." OU="Sigillum Polskie Centrum Certyfikacji
Elektronicznej" CN="Sigillum PCCE - CALevel3">CN=Sigillum PCCE - CALevel3,OU=Sigillum Polskie Centrum Certyfikacji
Elektronicznej,O=Polska Wytwórnia Papierów Wartościowych S.A.,C=PL</SignatureCertIssuer>
               <SignatureCertSerial>2930305822001951294</SignatureCertSerial>
               <SignatureCertSubject C="PL" O="MSWiA" CN="ePUAP ZP 1">CN=ePUAP ZP 1,O=MSWiA,C=PL</SignatureCertSubject>
               <SignatureId>Signature-4e1bfa91-696b-4b12-ac5c-4c18e5f0703b</SignatureId>
               <SigningTime>2014-06-30 12:13:08 CEST</SigningTime>
               <UriID lp="1"></UriID>
               <UriID lp="2">załącznik.txt</UriID>
               <UriID lp="3">#SignedProps-4e1bfa91-696b-4b12-ac5c-4c18e5f0703b</UriID>
               <SignatureTimeStamp znaczenie="Brak oznaczenia czasem"/>
              <ArchiveTimeStamp znaczenie="Brak postaci archiwalnej"/>
              <ZP czy_obecny="true">
                <ppZP:PodpisZP xmlns:os=http://crd.gov.pl/xml/schematy/osoba/2009/03/06/
xmlns:ppZP="http://crd.gov.pl/xml/schematy/ppzp/">
               <ppZP:DaneZP>
                 <ppZP:DaneZPOsobyFizycznej>
                   <os:Nazwisko rodzajCzlonu="pierwszy">Kowalski</os:Nazwisko>
                   <os:Imie>Jan</os:Imie>
                   <os:PESEL>10101010103</os:PESEL>
                   <ppZP:IdZaufanegoProfilu>17</ppZP:IdZaufanegoProfilu>
                   <ppZP:IdKontaUzytkownikaEpuap>user01</ppZP:IdKontaUzytkownikaEpuap>
                 </ppZP:DaneZPOsobyFizycznej>
               </ppZP:DaneZP>
               <ppZP:DanePodpisu>
                 <ppZP:IdKontaUzytkownikaEpuap>user01</ppZP:IdKontaUzytkownikaEpuap>
                 <ppZP:IdPolitykiAutoryzacji>1</ppZP:IdPolitykiAutoryzacji>
               </ppZP:DanePodpisu>
               </ppZP:PodpisZP>
             </ZP>
           </StatusInfo>
         </VerifyResult>]]>
       </verifySignatureReturn>
     </ns2:verifySignatureResponse>
</soap:Body>
```

```
</soap:Envelope>
```

Odpowiedź serwera na powyższe żądanie w przypadku gdy dokument nie jest w formacie XML jest następująca:

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Header/>
    <soap:Body>
      <soap:Fault>
        <faultcode>soap:Client</faultcode>
        <faultstring>Błąd w trakcie parsowania dokumentu, dla którego ma być przeprowadzona weryfikacja podpisu PZ.</faultstring>
        <detail>
          <ns2:VerificationException xmlns:ns2="http://verification.zp.epuap.gov.pl">
            <message>600: Błąd w trakcie parsowania dokumentu, dla którego ma być przeprowadzona weryfikacja podpisu PZ.</message>
          </ns2:VerificationException>
        </detail>
      </soap:Fault>
    </soap:Body>
</soap:Envelope>
```