



Krajowy Depozyt Papierów Wartościowych

ESDI

WebService

Wersja 1.2

Spis treści

1.	Informacje ogólne	4
2.	Komunikacja.....	6
3.	Format komunikatu ESDK dla ESDI Webservice.....	7
4.	Podpis CADES	8
5.	Funkcje API - formaty komunikatów	10
6.	Przykłady wywołania usług za pomocą klienta .NET	20

Historia zmian

Wersja	Data modyfikacji	Autor zmian	Szczegóły
1.0	2014-05-05	KDPW	Utworzenie dokumentu
1.1	2014-08-21	KDPW	Aktualizacja dokumentu
1.2	2016-04-25	KDPW	Aktualizacja procesu tworzenia nagłówka ESDK

1. Informacje ogólne

Usługa ESDI WebService jest rozwiązaniem umożliwiającym komunikację pomiędzy systemami informatycznymi Grupy KDPW a systemami uczestników rynku za pomocą usług sieciowych (web-services). Dzięki ESDI WebServices możliwa jest efektywna wymiana komunikatów ESDK opatrzonych podpisem elektronicznym poprzez szyfrowany kanał HTTPS.

ESDI WebServices służy tylko do przesyłania/odbierania komunikatów systemowych z KDPW/KDPW_CCP. Za pomocą tego trybu (A2A) nie będą wysyłane/odbierane inne komunikaty np. uchwały zarządu, ONP i inna korespondencja.

Proces podpisywania i wymiany komunikatów ESDK może być zautomatyzowany poprzez stworzenie odpowiedniego oprogramowania działającego w środowisku użytkownika i łączącego się do usługi ESDI WebService.

Proces tworzenia komunikatu ESDK dla ESDI WebService wynika z dokumentu „DokumentacjaESDK” dostępnego pod adresem:

http://www.kdpw.pl/pl/kdpw_stream/system/Documents/Systemy/ESDK/DokumentacjaESDK.pdf

i został rozszerzony o wydzielenie z pola „Obszar zarezerwowany” pola „Środowisko” o długości 4 znaków (dla środowiska testowego jest to „TSTA” lub „TSTB”, dla środowiska produkcyjnego „PRD ”). Szczegółowy opis formatu komunikatu ESDK dla ESDI WebService znajduje się w rozdziale 3.

Aplikacja użytkownika uwierzytelnia się za pomocą atrybutów bezpieczeństwa wydanych w pliku p12 dostarczonego przez KDPW.

Wszystkie pliki przesyłane poprzez ESDI WebService są podpisywane elektronicznie. Użytkownicy mogą wykorzystać do generowania podpisów wysyłanych komunikatów ESDK mechanizm realizowany przez bibliotekę BouncyCastle.Crypto.dll lub wykorzystać tożsame rozwiązanie innych firm. Podpis generowany jest w formacie CAdES z wykorzystaniem certyfikatu wydanego przez KDPW w pliku p12.

Bibliotekę dla języka C# można pobrać ze strony:

<http://www.bouncycastle.org/csharp/>

Bibliotekę dla języka Java można pobrać ze strony:

http://www.bouncycastle.org/latest_releases.html

Licencja na używanie biblioteki BouncyCastle.Crypto.dll znajduje się na stronie

<http://www.bouncycastle.org/licence.html>

<http://www.bouncycastle.org/csharp/licence.html>

Szczegóły licencji MIT X11:

<http://opensource.org/licenses/mit-license.php>

Wszystkie działania użytkowników w ramach wywoływania komunikatów web-services są rejestrowane.

2. Komunikacja

Wymiana komunikatów pomiędzy KDPW S.A. a uczestnikami rynku w usłudze ESDI WebService odbywa się przez wymianę komunikatów XML przy wykorzystaniu technologii web-services. Całość komunikacji jest zabezpieczona protokołem TLS.

Usługa ESDI WebService dostępna jest pod adresem:

<https://esdi.kdpw.pl/API/EsdiService.asmx> - środowisko produkcyjne

<https://esditst.kdpw.pl/API/EsdiService.asmx> - środowisko testowe

Pierwszym krokiem do nawiązania połączenia jest zawsze inicjacja bezpiecznego tunelu komunikacji (TLS). Po zestawieniu kanału, wewnątrz niego przekazywane są żądania SOAP.

Do zestawienia połączenia niezbędny jest certyfikat w pliku p12 wydany przez KDPW.

3. Format komunikatu ESDK dla ESDI Webservice

Nazwa pola	Długość	Typ
Numer komunikatu	9	N
Data	10	A
Godzina	8	A
ID Adresata	10	A
ID Nadawcy	10	A
Typ komunikatu	24	A
Podtyp komunikatu	4	A
Środowisko	4	A
Obszar zarezerwowany	16	A
Długość danych	8	N
Dane	Długość danych	B
Długość podpisu elektronicznego	8	N
Podpis elektroniczny	Długość podpisu elektronicznego	B

Typy pól:

A – znakowe

B - binarne

N – numeryczne

4. Podpis CADES

Przesyłane komunikaty ESDK, w celu zapewnienia integralności ich treści, opatrzone są podpisem cyfrowym. Każdy przekazywany komunikat ESDK musi zawierać podpis treści komunikatu. Podpisy są wykonywane przy użyciu certyfikatów autoryzacyjnych wydanych przez KDPW w pliku p12.

Poniższy opis przedstawia proces tworzenia podpisu cyfrowego komunikatu ESDK. Proces może być wykonywany przy użyciu biblioteki BouncyCastle.Crypto.dll lub tożsamego rozwiązania innych firm.

Dla wygenerowania podpisu cyfrowego, konieczna jest możliwość odczytu klucza prywatnego z pliku p12 wydanego przez KDPW.

Przykład odczytu certyfikatu z pliku i przekazanie go do procedury generującej podpis w języku C#:

```
X509Certificate2 x509Cert = new X509Certificate2(certPath, pass, X509KeyStorageFlags.Exportable | X509KeyStorageFlags.PersistKeySet);
```

certPatch – ścieżka do pliku p12

pass – hasło do certyfikatu

Przykład w języku C# funkcji generującej podpis z wykorzystaniem pliku p12:

dataToSign – tablica bajtów, dla której ma zostać wygenerowany podpis

x509Cert – certyfikat użyty do wygenerowania podpisu

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Security.Cryptography.X509Certificates;
using System.Collections;

public static byte[] SignData(byte[] dataToSign, X509Certificate2 x509Cert)
{
    Org.BouncyCastle.X509.X509Certificate cert =
    Org.BouncyCastle.Security.DotNetUtilities.FromX509Certificate(x509Cert);

    Org.BouncyCastle.Crypto.AsymmetricKeyParameter keyParameter =
    Org.BouncyCastle.Security.DotNetUtilities.GetKeyPair(cert.PrivateKey).Private;

    Org.BouncyCastle.Cms.CmsSignedDataGenerator generator = new
    Org.BouncyCastle.Cms.CmsSignedDataGenerator();
    generator.AddSigner(keyParameter, cert, Org.BouncyCastle.Cms.CmsSignedDataGenerator.EncryptionRsa,
    Org.BouncyCastle.Cms.CmsSignedDataGenerator.DigestSha1);

    ArrayList certList = new ArrayList();
    certList.Add(cert);
}
```



```
    Org.BouncyCastle.X509.Store.X509CollectionStoreParameters storeParameters = new
Org.BouncyCastle.X509.Store.X509CollectionStoreParameters(certList);
    Org.BouncyCastle.X509.Store.IX509Store store =
Org.BouncyCastle.X509.Store.X509StoreFactory.Create("CERTIFICATE/COLLECTION", storeParameters);

    generator.AddCertificates(store);

    Org.BouncyCastle.Cms.CmsSignedData signedData = generator.Generate("1.2.840.113549.1.7.1", new
Org.BouncyCastle.Cms.CmsProcessableByteArray(dataToSign), false);

    return signedData.GetEncoded();
}
```

Zwracana tablica bajtów jest podpisem odłączonym (detached), który należy zamieścić w komunikacie ESDK.

5. Funkcje API - formaty komunikatów

5.1. DeleteAllMessages

Wiadomość DeleteAllMessages jest żądaniem usunięcia wszystkich komunikatów odbiorczych dla danej jednostki organizacyjnej.

Parametry wejściowe: brak

Odpowiedź:

DeleteAllMessagesResult:

true – operacja zakończyła się powodzeniem;

false – błąd.

Request

```
<?xml version="1.0" encoding="utf-8"?>
<soap12:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap12="http://www.w3.org/2003/05/soap-envelope">
  <soap12:Body>
    <DeleteAllMessages xmlns="https://esdi.kdpw.pl/API/EsdiService" />
  </soap12:Body>
</soap12:Envelope>
```

Response

```
<?xml version="1.0" encoding="utf-8"?>
<soap12:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap12="http://www.w3.org/2003/05/soap-envelope">
  <soap12:Body>
    <DeleteAllMessagesResponse xmlns="https://esdi.kdpw.pl/API/EsdiService">
      <DeleteAllMessagesResult>boolean</DeleteAllMessagesResult>
    </DeleteAllMessagesResponse>
  </soap12:Body>
</soap12:Envelope>
```

5.2. DeleteAllFirstMessages

Wiadomość DeleteAllFirstMessages jest żądaniem usunięcia komunikatów odbiorczych dla danej jednostki organizacyjnej do daty przekazanej w parametrze wejściowym.

Parametry wejściowe:

dateTime – data do której będą usunięte komunikaty;

Odpowiedź:

DeleteAllFirstMessagesResult:

true – operacja zakończyła się powodzeniem;
false – błąd.

Request

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <DeleteAllFirstMessages xmlns="https://esdi.kdpw.pl/API/EsdiService">
      <dateTime>dateTime</dateTime>
    </DeleteAllFirstMessages>
  </soap:Body>
</soap:Envelope>
```

Response

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <DeleteAllFirstMessagesResponse xmlns="https://esdi.kdpw.pl/API/EsdiService">
      <DeleteAllFirstMessagesResult>boolean</DeleteAllFirstMessagesResult>
    </DeleteAllFirstMessagesResponse>
  </soap:Body>
</soap:Envelope>
```

5.3. DeleteMessage

Wiadomość DeleteMessage jest żądaniem usunięcia komunikatu odbiorczego dla danej jednostki organizacyjnej o podanym ID.

Parametry wejściowe:

Id – numer identyfikacyjny komunikatu;

Odpowiedź:

DeleteMessageResult:

true – operacja zakończyła się powodzeniem;

false – błąd.

Request

```
<?xml version="1.0" encoding="utf-8"?>
<soap12:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap12="http://www.w3.org/2003/05/soap-envelope">
  <soap12:Body>
    <DeleteMessage xmlns="https://esdi.kdpw.pl/API/EsdiService">
      <id>int</id>
    </DeleteMessage>
```

```
</soap12:Body>
</soap12:Envelope>
```

Response

```
<?xml version="1.0" encoding="utf-8"?>
<soap12:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap12="http://www.w3.org/2003/05/soap-envelope">
  <soap12:Body>
    <DeleteMessageResponse xmlns="https://esdi.kdpw.pl/API/EsdiService">
      <DeleteMessageResult>boolean</DeleteMessageResult>
    </DeleteMessageResponse>
  </soap12:Body>
</soap12:Envelope>
```

5.4. GetAllFirstMessages

Wiadomość GetAllFirstMessages jest żądaniem pobrania komunikatów odbiorczych danej jednostki organizacyjnej. Parametr wejściowy określa maksymalną ilość komunikatów zwróconych w odpowiedzi (komunikaty będą zwracane od najstarszego).

Parametry wejściowe:

count – maksymalna ilość komunikatów zwróconych w odpowiedzi;

Odpowiedź:

MessageID – id komunikatu;

FileName – nazwa pliku;

FileData – dane pliku w formacie podpisanego komunikatu ESDK;

CreateDate – data i czas utworzenia komunikatu.

Request

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <GetAllFirstMessages xmlns="https://esdi.kdpw.pl/API/EsdiService">
      <count>int</count>
    </GetAllFirstMessages>
  </soap:Body>
</soap:Envelope>
```

Response

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <GetAllFirstMessagesResponse xmlns="https://esdi.kdpw.pl/API/EsdiService">
      <GetAllFirstMessagesResult>
        <Message>
```

```

    <MessageID>int</MessageID>
    <FileName>string</FileName>
    <FileData>base64Binary</FileData>
    <CreateDate>dateTime</CreateDate>
  </Message>
</Message>
  <Message>
    <MessageID>int</MessageID>
    <FileName>string</FileName>
    <FileData>base64Binary</FileData>
    <CreateDate>dateTime</CreateDate>
  </Message>
</GetAllFirstMessagesResult>
</GetAllFirstMessagesResponse>
</soap:Body>
</soap:Envelope>

```

5.5. GetAllMessages

Wiadomość GetAllMessages jest żądaniem pobrania wszystkich komunikatów odbiorczych danej jednostki organizacyjnej.

Parametry wejściowe: brak;

Odpowiedź:

MessageID – id komunikatu;

FileName – nazwa pliku;

FileData – dane pliku w formacie podpisanego komunikatu ESDK;

CreateDate – data i czas utworzenia komunikatu.

Request

```

<?xml version="1.0" encoding="utf-8"?>
<soap12:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap12="http://www.w3.org/2003/05/soap-envelope">
  <soap12:Body>
    <GetAllMessages xmlns="https://esdi.kdpw.pl/API/EsdiService" />
  </soap12:Body>
</soap12:Envelope>

```

Response

```

<?xml version="1.0" encoding="utf-8"?>
<soap12:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap12="http://www.w3.org/2003/05/soap-envelope">
  <soap12:Body>
    <GetAllMessagesResponse xmlns="https://esdi.kdpw.pl/API/EsdiService">
      <GetAllMessagesResult>
        <Message>
          <MessageID>int</MessageID>
          <FileName>string</FileName>
          <FileData>base64Binary</FileData>
          <CreateDate>dateTime</CreateDate>
        </Message>

```

```

    <Message>
      <MessageID>int</MessageID>
      <FileName>string</FileName>
      <FileData>base64Binary</FileData>
      <CreateDate>dateTime</CreateDate>
    </Message>
  </GetAllMessagesResult>
</GetAllMessagesResponse>
</soap12:Body>
</soap12:Envelope>

```

5.6. GetEnvironments

Wiadomość GetEnvironments jest żądaniem zwrócenia dostępnych środowisk dla danej jednostki organizacyjnej.

Parametry wejściowe: brak;

Odpowiedź:

string – kod środowiska.

Request

```

<?xml version="1.0" encoding="utf-8"?>
<soap12:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap12="http://www.w3.org/2003/05/soap-envelope">
  <soap12:Body>
    <GetEnvironments xmlns="https://esdi.kdpw.pl/API/EsdiService" />
  </soap12:Body>
</soap12:Envelope>

```

Response

```

<?xml version="1.0" encoding="utf-8"?>
<soap12:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap12="http://www.w3.org/2003/05/soap-envelope">
  <soap12:Body>
    <GetEnvironmentsResponse xmlns="https://esdi.kdpw.pl/API/EsdiService">
      <GetEnvironmentsResult>
        <string>string</string>
        <string>string</string>
      </GetEnvironmentsResult>
    </GetEnvironmentsResponse>
  </soap12:Body>
</soap12:Envelope>

```

5.7. GetFirstMessages

Wiadomość GetFirstMessages jest żądaniem pobrania komunikatów odbiorczych danej jednostki organizacyjnej, utworzonych po określonej dacie. Parametry wejściowe określają

maksymalną ilość komunikatów zwróconych w odpowiedzi (komunikaty będą zwracane od najstarszego) oraz datę od której mają być pobierane komunikaty

Parametry wejściowe:

count – maksymalna ilość komunikatów zwróconych w odpowiedzi;

dateTime – data od której będą pobrane komunikaty;

Odpowiedź:

MessageID – id komunikatu;

FileName – nazwa pliku;

FileData – dane pliku w formacie podpisanego komunikatu ESDK;

CreateDate – data i czas utworzenia komunikatu.

Request

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <GetFirstMessages xmlns="https://esdi.kdpw.pl/API/EsdiService">
      <count>int</count>
      <dateTime>dateTime</dateTime>
    </GetFirstMessages>
  </soap:Body>
</soap:Envelope>
```

Response

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <GetFirstMessagesResponse xmlns="https://esditst.kdpw.pl/API/EsdiService">
      <GetFirstMessagesResult>
        <Message>
          <MessageID>int</MessageID>
          <FileName>string</FileName>
          <FileData>base64Binary</FileData>
          <CreateDate>dateTime</CreateDate>
        </Message>
        <Message>
          <MessageID>int</MessageID>
          <FileName>string</FileName>
          <FileData>base64Binary</FileData>
          <CreateDate>dateTime</CreateDate>
        </Message>
      </GetFirstMessagesResult>
    </GetFirstMessagesResponse>
  </soap:Body>
</soap:Envelope>
```

5.8. GetMessage

Wiadomość `GetMessages` jest żądaniem pobrania wszystkich komunikatów odbiorczych danej jednostki organizacyjnej, utworzonych po określonej dacie.

Parametry wejściowe:

`dateTime` – data od której będą pobrane komunikaty;

Odpowiedź:

`MessageID` – id komunikatu;

`FileName` – nazwa pliku;

`FileData` – dane pliku w formacie podpisanego komunikatu ESDK;

`CreateDate` – data i czas utworzenia komunikatu.

Request

```
<?xml version="1.0" encoding="utf-8"?>
<soap12:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap12="http://www.w3.org/2003/05/soap-envelope">
  <soap12:Body>
    <GetMessages xmlns="https://esdi.kdpw.pl/API/EsdiService">
      <dateTime>dateTime</dateTime>
    </GetMessages>
  </soap12:Body>
</soap12:Envelope>
```

Response

```
<?xml version="1.0" encoding="utf-8"?>
<soap12:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap12="http://www.w3.org/2003/05/soap-envelope">
  <soap12:Body>
    <GetMessagesResponse xmlns="https://esdi.kdpw.pl/API/EsdiService">
      <GetMessagesResult>
        <Message>
          <MessageID>int</MessageID>
          <FileName>string</FileName>
          <FileData>base64Binary</FileData>
          <CreateDate>dateTime</CreateDate>
        </Message>
        <Message>
          <MessageID>int</MessageID>
          <FileName>string</FileName>
          <FileData>base64Binary</FileData>
          <CreateDate>dateTime</CreateDate>
        </Message>
      </GetMessagesResult>
    </GetMessagesResponse>
  </soap12:Body>
</soap12:Envelope>
```

5.9. SendMessage

Wiadomość `SendMessage` wysyła plik z podpisanym komunikatem ESDK.

Parametry wejściowe:

file – dane pliku w formacie podpisanego komunikatu ESDK;

fileName – nazwa pliku;

Odpowiedź:

SendMessageResult – ID przyjętego komunikatu nadane przez system ESDI WebService.

Request

```
<?xml version="1.0" encoding="utf-8"?>
<soap12:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap12="http://www.w3.org/2003/05/soap-envelope">
  <soap12:Body>
    <SendMessage xmlns="https://esdi.kdpw.pl/API/EsdiService">
      <file>base64Binary</file>
      <fileName>string</fileName>
    </SendMessage>
  </soap12:Body>
</soap12:Envelope>
```

Response

```
<?xml version="1.0" encoding="utf-8"?>
<soap12:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap12="http://www.w3.org/2003/05/soap-envelope">
  <soap12:Body>
    <SendMessageResponse xmlns="https://esdi.kdpw.pl/API/EsdiService">
      <SendMessageResult>int</SendMessageResult>
    </SendMessageResponse>
  </soap12:Body>
</soap12:Envelope>
```

5.10. VerifyReceiveMessage

Wiadomość VerifyReceiveMessage jest żądaniem weryfikacji podpisu odebranego komunikatu ESDK.

Parametry wejściowe:

id – numer identyfikacyjny komunikatu zwróconego metodą GetMessage, GetFirstMessages, GetAllFirstMessages lub GetAllMessages;

Odpowiedź:

MessageID – ID komunikatu;

FileName – nazwa pliku;

Status – status weryfikacji:

SGN_OK – weryfikacja pozytywna;

SGN_NOT_VER – weryfikacja negatywna;

VerifyFailCode – w przypadku weryfikacji negatywnej - kod błędu;

VerifyFailReason – w przypadku weryfikacji negatywnej - opis błędu;
SignedBy – w przypadku weryfikacji pozytywnej - informacje o podpisie.

Request

```
<?xml version="1.0" encoding="utf-8"?>
<soap12:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap12="http://www.w3.org/2003/05/soap-envelope">
  <soap12:Body>
    <VerifyReceiveMessage xmlns="https://esdi.kdpw.pl/API/EsdiService">
      <id>int</id>
    </VerifyReceiveMessage>
  </soap12:Body>
</soap12:Envelope>
```

Response

```
<?xml version="1.0" encoding="utf-8"?>
<soap12:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap12="http://www.w3.org/2003/05/soap-envelope">
  <soap12:Body>
    <VerifyReceiveMessageResponse xmlns="https://esdi.kdpw.pl/API/EsdiService">
      <VerifyReceiveMessageResult>
        <MessageID>int</MessageID>
        <FileName>string</FileName>
        <Status>string</Status>
        <VerifyFailCode>string</VerifyFailCode>
        <VerifyFailReason>string</VerifyFailReason>
        <SignedBy>string</SignedBy>
      </VerifyReceiveMessageResult>
    </VerifyReceiveMessageResponse>
  </soap12:Body>
</soap12:Envelope>
```

5.11. VerifySendMessage

Wiadomość VerifySendMessage jest żądaniem weryfikacji podpisu dostarczonego komunikatu ESDK.

Parametry wejściowe:

id – numer identyfikacyjny komunikatu zwróconego metodą SendMessage;

Odpowiedź:

MessageID – ID komunikatu;

FileName – nazwa pliku;

Status – status weryfikacji:

SGN_OK – weryfikacja pozytywna;

SGN_NOT_VER – weryfikacja negatywna;

VerifyFailCode – w przypadku weryfikacji negatywnej - kod błędu;

VerifyFailReason – w przypadku weryfikacji negatywnej - opis błędu;

SignedBy – w przypadku weryfikacji pozytywnej - informacje o podpisie.

Request

```
<?xml version="1.0" encoding="utf-8"?>
<soap12:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap12="http://www.w3.org/2003/05/soap-envelope">
  <soap12:Body>
    <VerifySendMessage xmlns="https://esdi.kdpw.pl/API/EsdiService">
      <id>int</id>
    </VerifySendMessage>
  </soap12:Body>
</soap12:Envelope>
```

Response

```
<?xml version="1.0" encoding="utf-8"?>
<soap12:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap12="http://www.w3.org/2003/05/soap-envelope">
  <soap12:Body>
    <VerifySendMessageResponse xmlns="https://esdi.kdpw.pl/API/EsdiService">
      <VerifySendMessageResult>
        <MessageID>int</MessageID>
        <FileName>string</FileName>
        <Status>string</Status>
        <VerifyFailCode>string</VerifyFailCode>
        <VerifyFailReason>string</VerifyFailReason>
        <SignedBy>string</SignedBy>
      </VerifySendMessageResult>
    </VerifySendMessageResponse>
  </soap12:Body>
</soap12:Envelope>
```

6. Przykłady wywołania usług za pomocą klienta .NET

Środowisko Visual Studio dysponuje zaawansowanym kreatorem aplikacji, który pozwala zaimportować asmx i wygenerować na jego podstawie klasy, niezbędne do przygotowania komunikatów i wywołania usługi ESDI Webservice. Kreator dostępny jest w menu Project → Add Service Reference. Następnie wybieramy Advanced → Add Web Reference.

W wyświetlonym oknie dialogowym w polu URL należy wpisać adres usługi ESDI Webservice.

Dla środowiska produkcyjnego

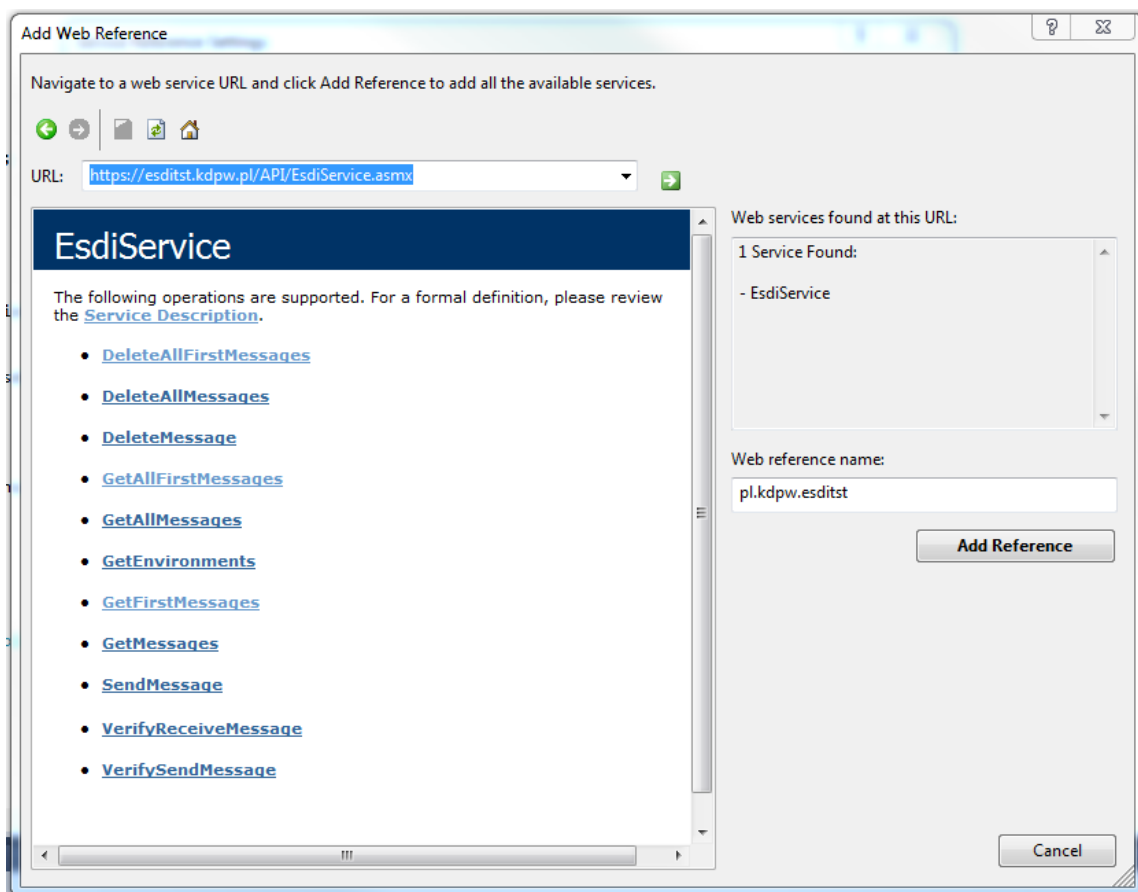
`https://esdi.kdpw.pl/API/EsdiService.asmx`

dla środowiska testowego

`https://esditst.kdpw.pl/API/EsdiService.asmx`

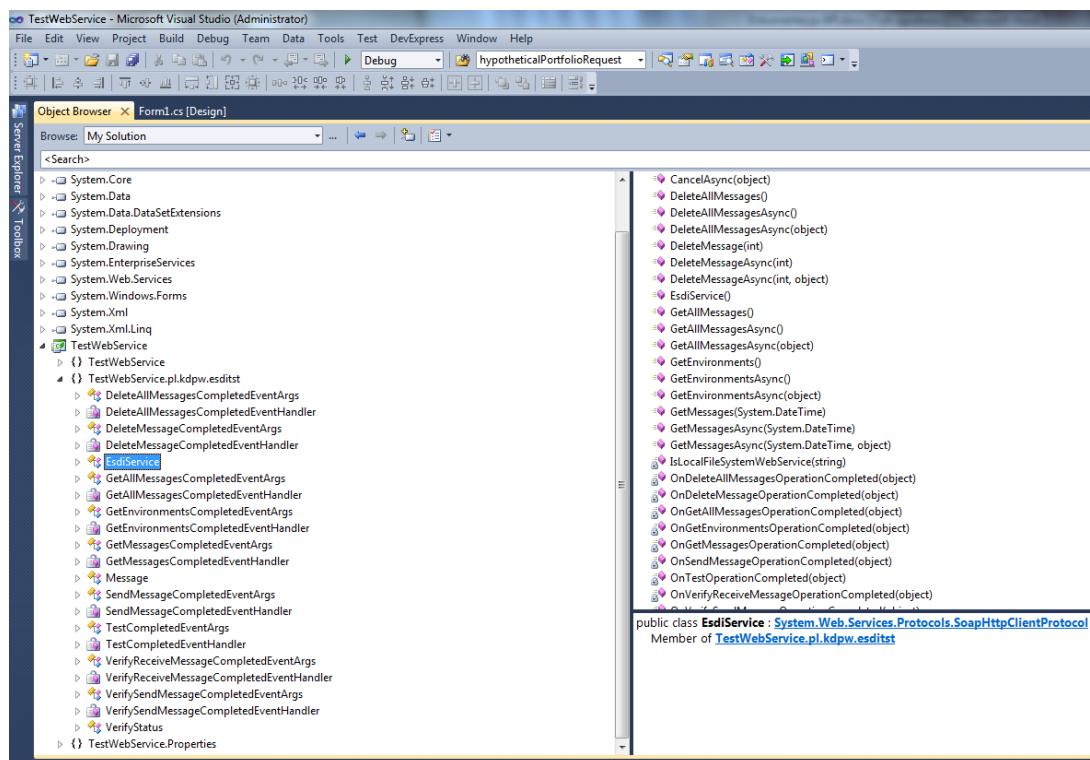
i wcisnąć przycisk GO.

Po wybraniu certyfikatu, którym połączymy się do usługi, możemy dodać referencje.



Rys. 1

Po zatwierdzeniu okna dialogowego przyciskiem Add Reference, kreator Visual Studio wygeneruje szereg klas, które odpowiadają m. in. poszczególnym komunikatom usługi ESDI WebService. Strukturę klas można przejrzeć za pomocą przeglądarki Object Browser:



Rys. 2

Konfiguracja połączenia do usługi ESDI WebService została zapisana w pliku konfiguracyjnym app.config wewnątrz znacznika applicationSettings:

```
<applicationSettings>
  <TestWebService.Properties.Settings>
    <setting name="TestWebService_pl_kdpw_esditst_EsdiService" serializeAs="String">
      <value>https://esditst.kdpw.pl/API/EsdiService.asmx</value>
    </setting>
  </TestWebService.Properties.Settings>
</applicationSettings>
```

Wywołanie przykładowej usługi:

poniższy przykład demonstruje wywołanie metody GetMessage(DateTime) zwracającej wiadomości utworzone po wskazanej dacie.

W celu ustanowienia połączenia z usługą ESDI WebService należy do obiektu utworzonej usługi dodać certyfikat klienta. Certyfikat z pliku p12 można zainstalować w magazynie certyfikatów Windows w celu jego późniejszego przekazania.

Przykład przeszukania magazynu certyfikatów użytkownika systemu Windows, sekcji certyfikatów osobistych i pobrania certyfikatu o numerze seryjnym 13B7:

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Security.Cryptography.X509Certificates;

namespace TestWebService
{
    static class Cert
    {
        public static X509Certificate2 GetCertificate()
        {
            X509Certificate2 cert = null;
            X509Store store = new X509Store(StoreLocation.CurrentUser);
            try
            {
                store.Open(OpenFlags.OpenExistingOnly);
                X509Certificate2Collection collection = store.Certificates;
                X509Certificate2Collection findCollection =
collection.Find(X509FindType.FindBySerialNumber, "13B7", true);
                if (findCollection.Count == 1)
                    cert = findCollection[0];
                return cert;
            }
            finally
            {
                store.Close();
            }
        }
    }
}

```

Tak pobrany certyfikat można przekazać do klasy `EsdiService` reprezentującej referencje do usługi ESDI WebService:

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using TestWebService.pl.kdpw.esditst;
using System.Security.Cryptography.X509Certificates;

namespace TestWebService
{
    class TestService
    {
        static X509Certificate2 cert = Cert.GetCertificate();
        static EsdiService esdiService = new EsdiService();

        public static Message[] GetMessages(DateTime dateTime)
        {
            esdiService.ClientCertificates.Add(cert);
            Message[] messages = esdiService.GetMessages(dateTime);
            return messages;
        }
    }
}

```

Metoda `GetMessages` klasy `TestService` zwraca tablicę typów `Message` z przestrzeni nazw `TestWebService.pl.kdpw.esditst`. Tak zwróconą tablicę można przeszukiwać i odczytywać pola poszczególnych obiektów:

```

for(int i = 0; i < messages.Length; i++)
{
    Message message = messages[i];
    DateTime createDate = message.CreateDate;
}

```

```
byte[] fileData = message.FileData;  
string fileName = message.FileName;  
int messageID = message.MessageID;  
}
```