EDI, XML i ochrona danych

Przemysław Kazienko

Zakład Systemów Informacyjnych, Wydział Informatyki i Zarządzania
Politechnika Wrocławska
kazienko@pwr.wroc.pl
http://www.pwr.wroc.pl/~kazienko

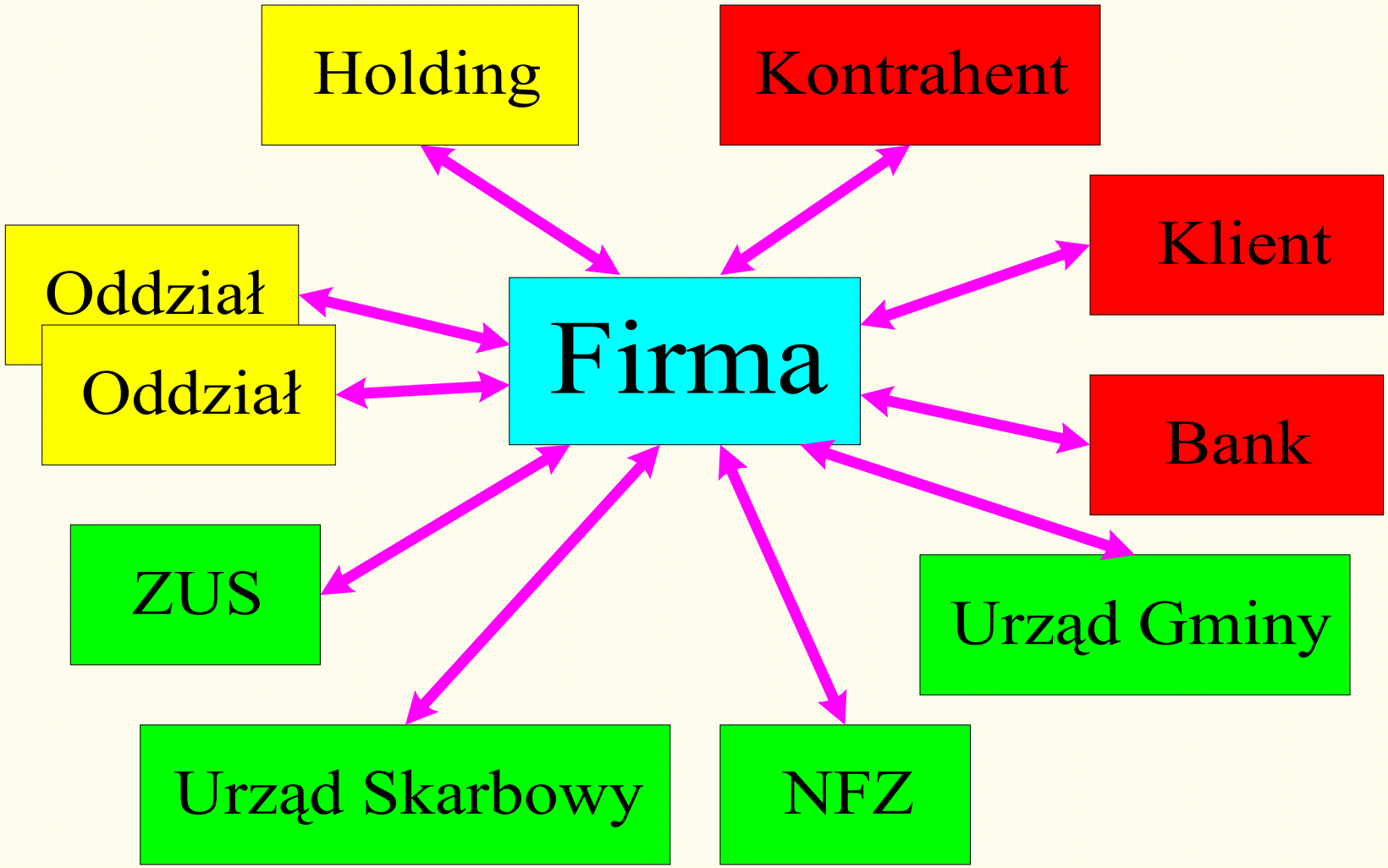
EDI

- **Elektroniczna wymiana danych** (*Electronic Data Interchange*)
- Klucz do rozwoju współczesnych przedsiębiorstw i społeczeństw
- Wymiana na poziomie **systemów**
- Wymiana na poziomie **organizacji**
- Wymiana danych to także **PROCESY BIZNESOWE**
- **Standardy EDI**, np. UN/EDIFACT (*Electronic Data Interchange For Administration, Commerce and Transport*), SWIFT

EDI na poziomie systemów

- Wymiana pomiędzy **modułami** jednej aplikacji
- Pomiędzy **aplikacjami**
- Usługi **web services**
- Z systemów transakcyjnych **do hurtowni** (ładowanie)
- **EAI** (*Enterprise Application Integration*), **systemy middleware** – integracja systemów wewnątrz instytucji

EDI na poziomie organizacji



XML

- Rodzina standardów
- Format tekstowy (niezbędny dla EDI)
- Znany i powszechny
- Integracja z WWW (przeglądarki, serwery)
- Współpraca z bazami danych (Oracle, SQL Server)
- Możliwość walidacji (sprawdzenie poprawności strukturalnej) – schematy XML Schema, DTD
- Wyszukiwanie – XPath, XQuery
- Transformacje - XSLT
- Łączenie i integracja – XLink, relacje *key-keyref*, XInclude, encje

EDI a XML

- Dokumenty XML zawierają wymieniane dane
- XML w EDI to aktualnie konieczność
- Rozwiązuje problem formatu, narzędzi, dostępności, ale nie problem identyfikacji
- Stare standardy EDI są przenoszone do języka XML, np. *SWIFTStandards XML*
- Nowe standardy EDI XML, np. *ebXML*
- Usługi *web services*

Dotychczasowe mechanizmy bezpieczeństwa

- Dedykowane łącza (EDIFACT, SWIFT)
- **Zabezpieczanie dokumentów**:
  - PGP
  - S/MIME, PEM
  - Systemy dedykowane
- Wady: brak uznanych standardów (poza S/MIME), nie specjalizowane do wymiany danych, implementacje zwykle ograniczone do konkretnych protokołów, np. pocztowych
- Zalety: tanie (oprócz dedykowanych łącz), niezależne od kanałów transmisji, bezpieczne po przesłaniu

Zabezpieczanie transmisji

- **Zabezpieczanie transmisji** w sieciach ogólnych:
  - SSL/TLS
  - VPN, IPSec
  - WEP
  - WTLS
  - VLAN
- Wady: brak dowodów po jej zakończeniu, mechanizmy bezpieczeństwa zwykle nie najsilniejsze, nie specjalizowane do wymiany dokumentów
- Zalety: tanie, powszechne

Alternatywa: standardy bezpieczeństwa XML

XML Encryption i XML Signature

- Języki wywodzące się z XML (zintegrowane z XML)
  - ... a więc mające postać tekstową
  - Obsługa przestrzeni nazw (*namespaces*)
  - Obsługa schematów XML Schema
  - Obsługa XPath (do wskazywania, także adresy URI)
  - Możliwość integracji z dowolnym językiem wywodzącym się z XML (np. XHTML, XSLT, SVG, WSDL, ...) także XML Encryption z XML Signature
  - Możliwość rozbudowy, np. dodawanie własnych parametrów, stempli czasowych, itd.

XML Encryption i XML Signature

- Zabezpieczanie **fragmentów dokumentu XML** (np. tylko treść elementu `NrKartyPłatniczej` lub `Cena`)
- Zabezpieczanie **kilku fragmentów a nawet dokumentów** „za jednym zamachem”
- **Jednoczesne zastosowanie różnych zabezpieczeń** (szyfrowania i podpisów) dla różnych fragmentów tego samego dokumentu
- Możliwość zastosowania różnych (także własnych) algorytmów i ich parametrów
- Możliwość **zabezpieczania danych binarnych**, także multimediów

XML Signature

- Podpis elektroniczny dla dokumentów XML
- Składa się z jednego elementu `Signature`
- W podpisie wskazanie na podpisywane dane (URI)

XML Signature – przykład

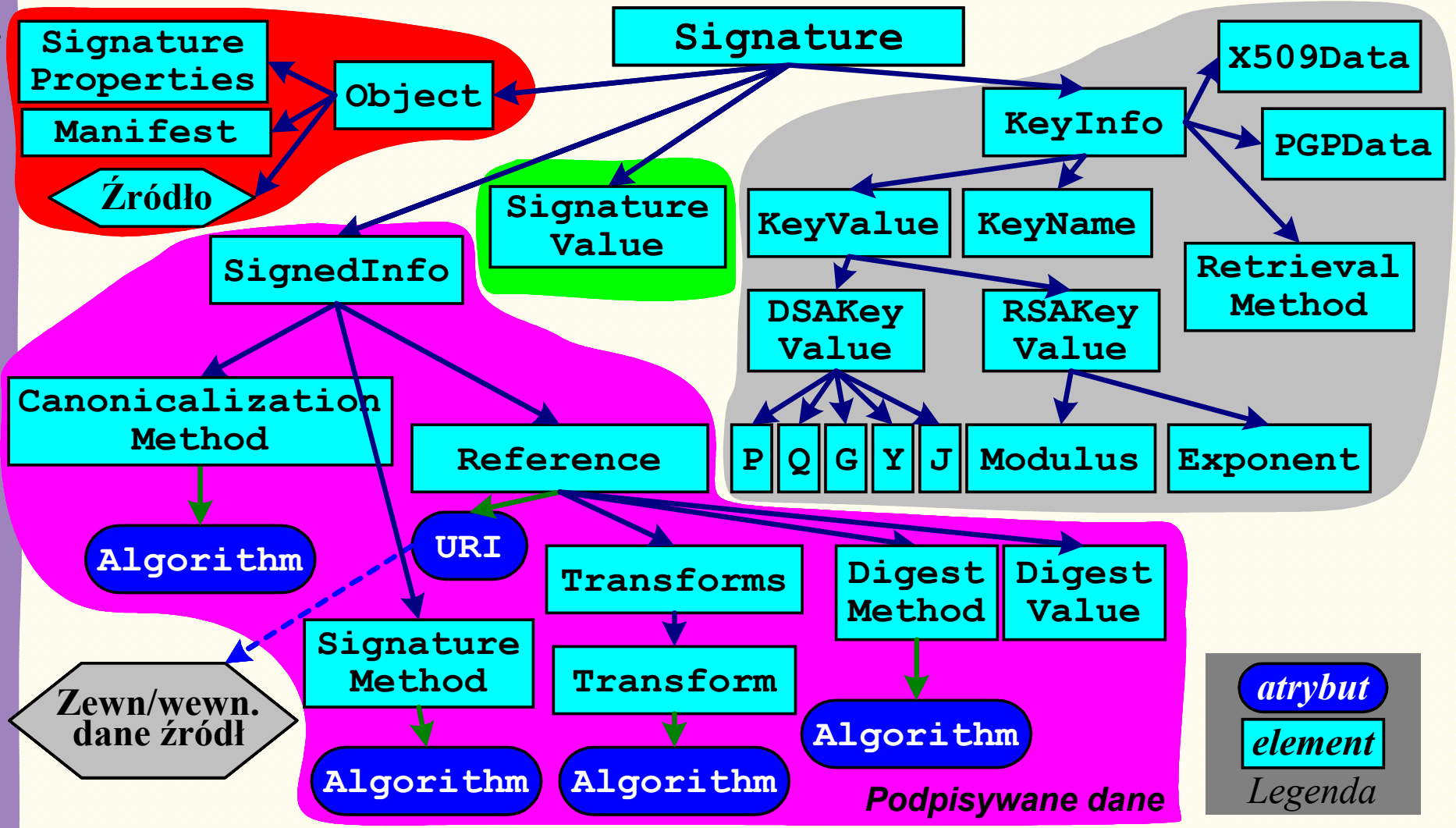
```
<?xml version="1.0" encoding="UTF-8"?>
<Signature xmlns="http://www.w3.org/2000/09/xmldsig#">
 <SignedInfo>
  <CanonicalizationMethod
    Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315" />
  <SignatureMethod
    Algorithm="http://www.w3.org/2000/09/xmldsig#dsa-sha1" />
  <Reference URI="http://przyklad.pl/xml/do-podpisu.b64">
   <Transforms>
    <Transform Algorithm="http://www.w3.org/2000/09/xmldsig#base64"/>
   </Transforms>
   <DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1"/>
   <DigestValue>60NvZvtdTB+7UnlLp/H24p7h4bs=</DigestValue>
  </Reference>
 </SignedInfo>
 <SignatureValue>OsH9A1jTNLmEldLmsPLlog6Gdw4YV8SiqD96GwYLAfMBqbk5
o3waOg==</SignatureValue>
 <KeyInfo>...</KeyInfo>
</Signature>
```

dalej

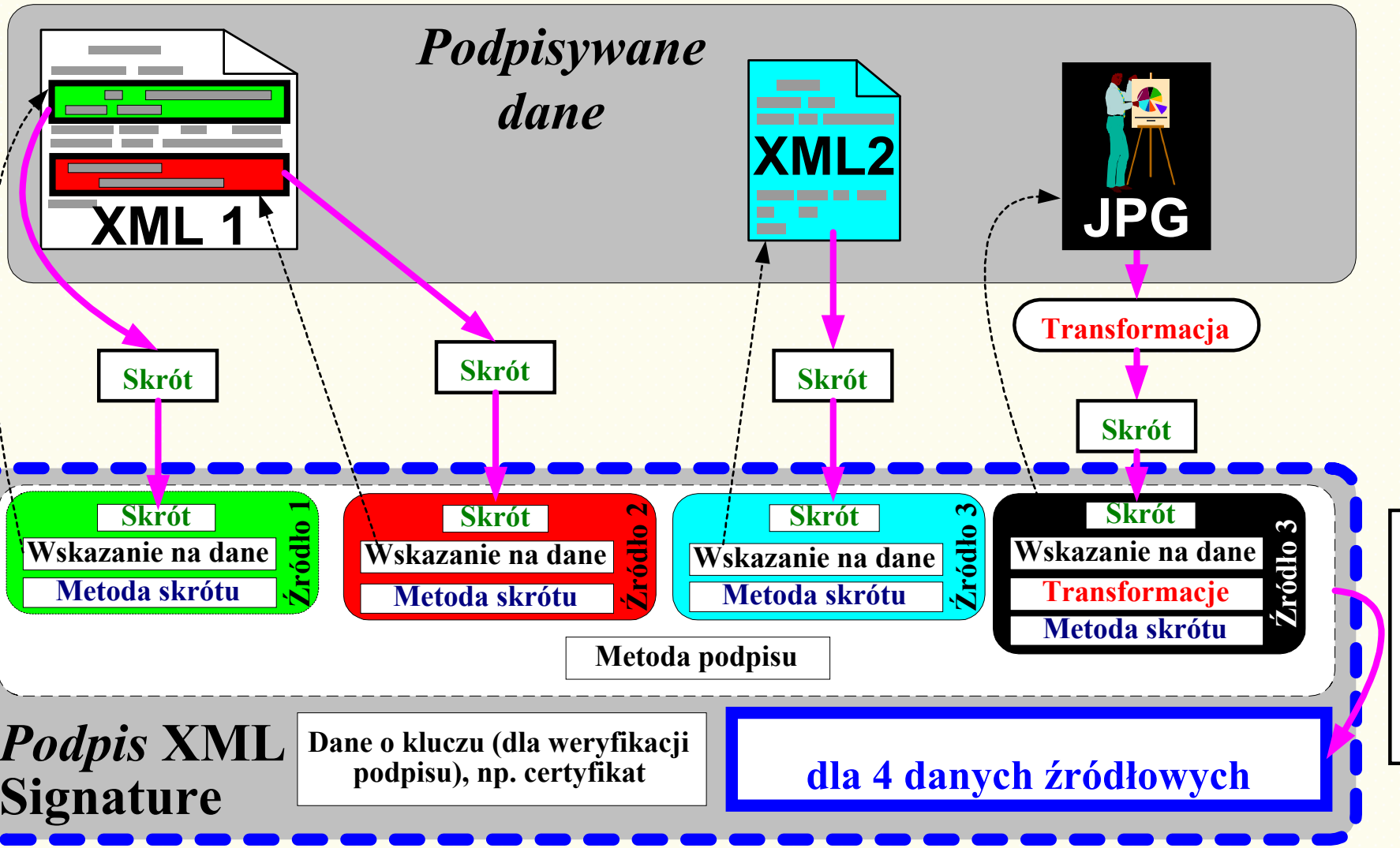
XML Signature – klucz DSA

```
<?xml version="1.0" encoding="UTF-8"?>
<Signature xmlns="... ">
 <SignedInfo>...</SignedInfo>
 <SignatureValue>OsH9A1j ... 3waOg==</SignatureValue>
 <KeyInfo>
  <KeyValue>
   <DSAKeyValue>
    <P>
imup6lmki4rAmUstKb/xdBRMWNtQ+pDN97ZnLA9X3lKbkEHtYFyjQ3uActgVSJ75iVRuKx
z4Cb5RzVm25EaKmKq8rif1MtBIi6jjDJxmIdNaEKG9zVTf9giJx1N9I0t3oh1fAVZDSrzK
zJGQ2WvDfIfFHdJMtB3C0VKGmLZR7Xk=
    </P>
    <Q>xDve3j7sEnh4rIzM5gK+5/gxxFU=</Q>
    <G>
NLugAf6IZJxo3BCOi5yrGEVwtlEzXcnndXhd0Tz38CnQKc4SEupm4PyP5TmLvK64TDfOD7
sno/W5oI1KZdimfW2c4r/6waNzZSvicMOWhLYY621Nn6njBc8VNwoxWpzCXhKm70b8+D4Y
ZMn/eU5DN8dvhTv/bNK21FfJqjp033U=
    </G>
    <Y>
W7dOmH/vWqocVCiqaxj6soxVXfR8XpMdY2Zv4Amjr3n81geyOLb6IZ+l7MUbdp8529DQzu
oVTthVpB9X4JKCprZIzifOTM1PFflTBzjx7egJwJWAIVdWyiIPjke6Va+wuV2n4Rl/cgCv
rXK5cTov5C/Bpaf6o+qrrDGFBLLZTF4=
    </Y>
   </DSAKeyValue>
  </KeyValue>
 </KeyInfo>
</Signature>
```

XML Signature - elementy

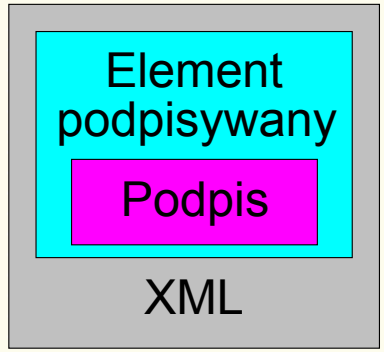


Podpisywanie wielu dokumentów

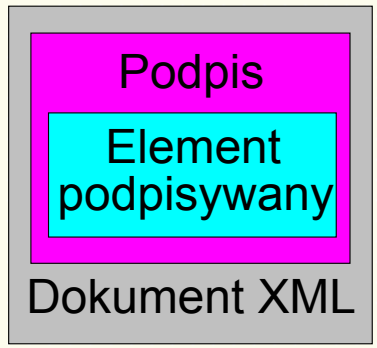


XML Signature – umiejscowienie

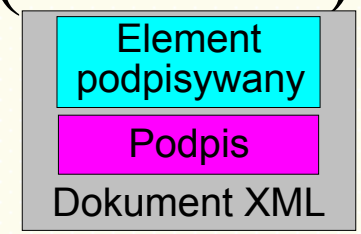
- Wbudowany (enveloped)

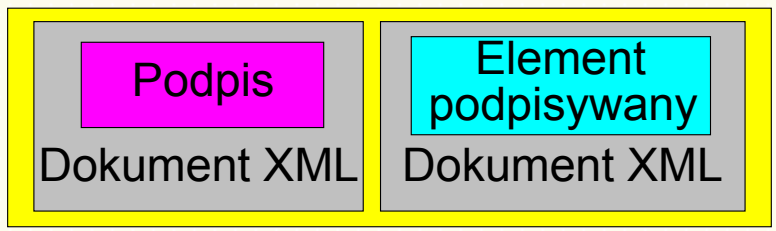


- Obejmujący (enveloping)

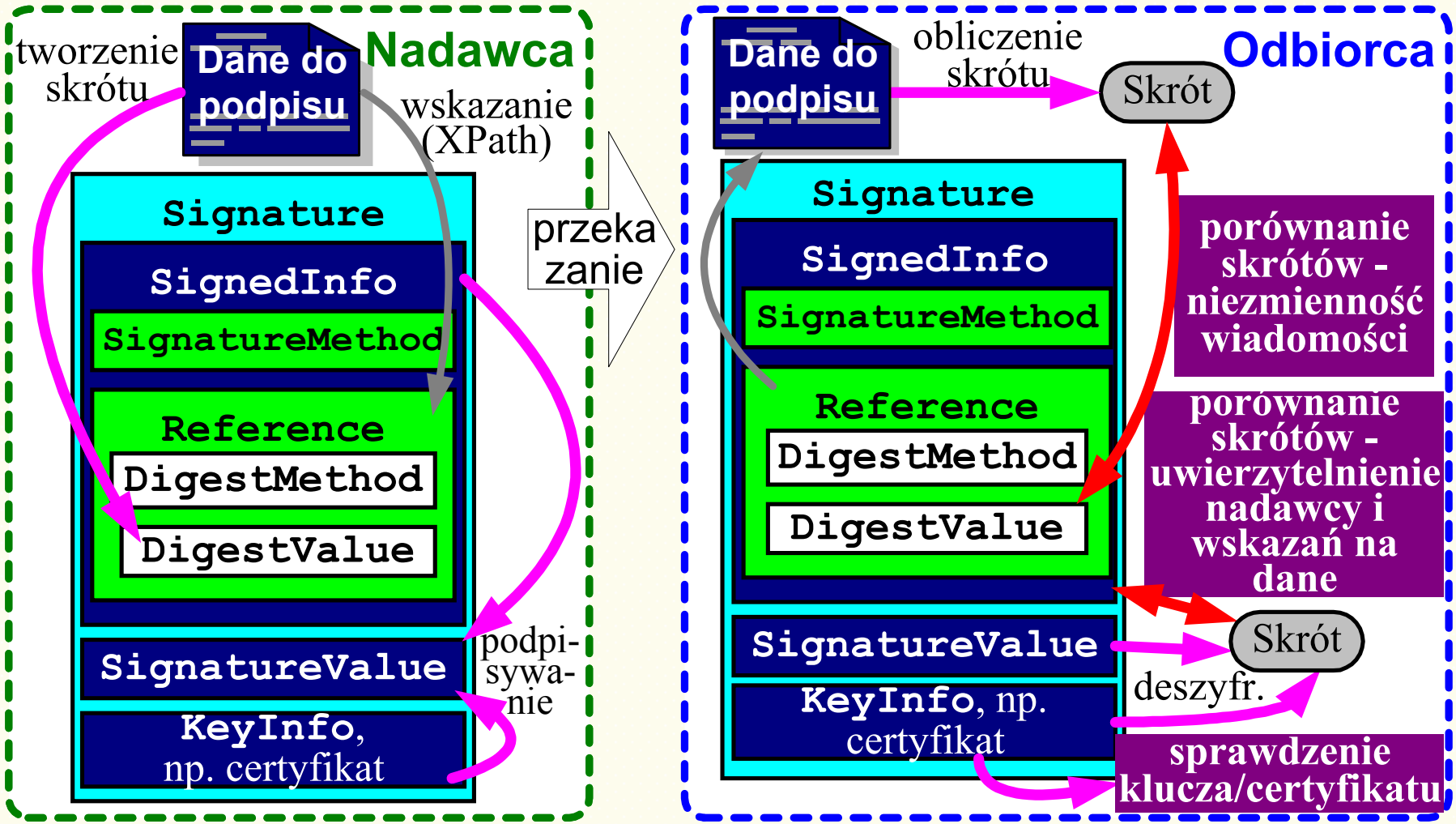


- Załączony (detached)





XML Signature – proces

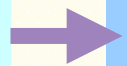


XML Encryption

- Szyfrowanie i deszyfrowanie danych (dokument, element, zawartość elementu)
- Efektem szyfrowania jest element `EncryptedData`, który zawiera albo wskazuje na zaszyfrowane dane
- Może on
  - Zastąpić szyfrowany element
  - Być podelementem szyfrowanego lub innego elementu
  - Być elementem innego dokumentu

XML Encryption – źródło przykładu

```
<?xml version='1.0'?>
<InfoPlatnicza xmlns='http://przyklad.pl/platnosc1'>
  <Nazwa>Józef Nowak</Nazwa>
  <KartaKredytowa Limit='5,000' Wal='PLN'>
    <NrKarty>4019 2445 0277 5567</NrKarty>
    <Wystawca>Nasz Bank S.A.</Wystawca>
    <DataWaznosci>10/03</DataWaznosci>
  </KartaKredytowa>
</InfoPlatnicza>
```

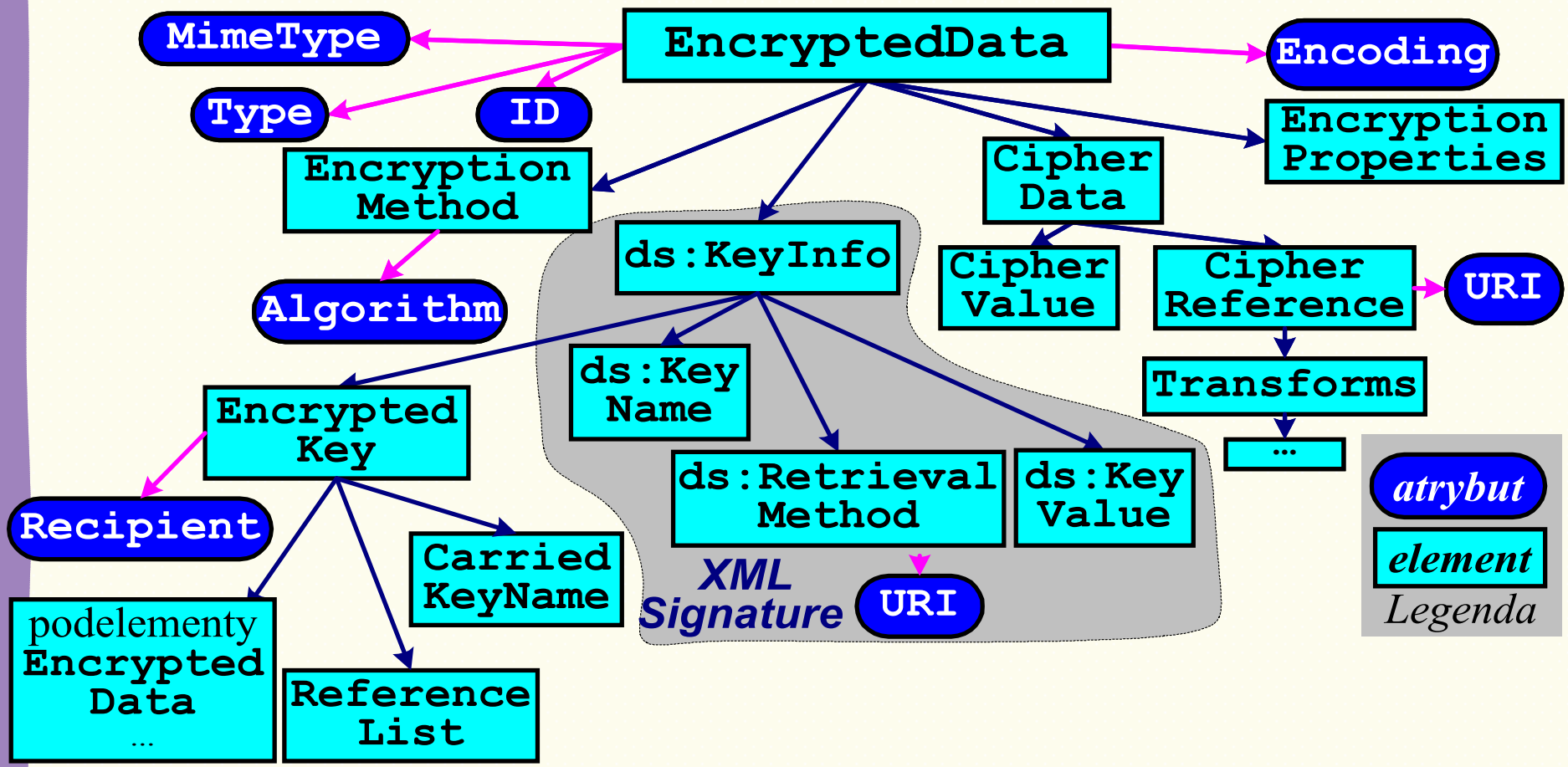


Zaszy-frować

XML Encryption – przykład

```
<?xml version='1.0'?>

<InfoPlatnicza xmlns='http://przyklad.pl/platnosc1'>

  <Nazwa>Józef Nowak</Nazwa>

  <EncryptedData
       Type='http://www.w3.org/2001/04/xmlenc#Element'
       xmlns='http://www.w3.org/2001/04/xmlenc#'>

    <CipherData>

      <CipherValue>A2s3B4f5TWc5cx6weQ3g5tesV=</CipherValue>

    </CipherData>

  </EncryptedData>

</InfoPlatnicza>
```

XML Encryption - elementy



Zastosowania - przykłady

- Szyfrowanie symetrycznym kluczem sesyjnym. Klucz sesyjny – asymetrycznym
- Szyfrowanie kluczy symetrycznych wieloma asymetrycznymi – dla wielu odbiorców
- Różne części szyfrowane różnie dla różnych odbiorców
- Podpisywanie jednym podpisem wielu dokumentów, np. XSLT
- Wydzielenie podpisów – baza podpisów dla kolekcji dokum.
- Zebranie podpisów różnych osób (np. członków zarządu)
- Podpisy wielokrotne: klient – sklep internetowy – bank
- Tekstowe podpisy / szyfrowanie danych binarnych

Problemy

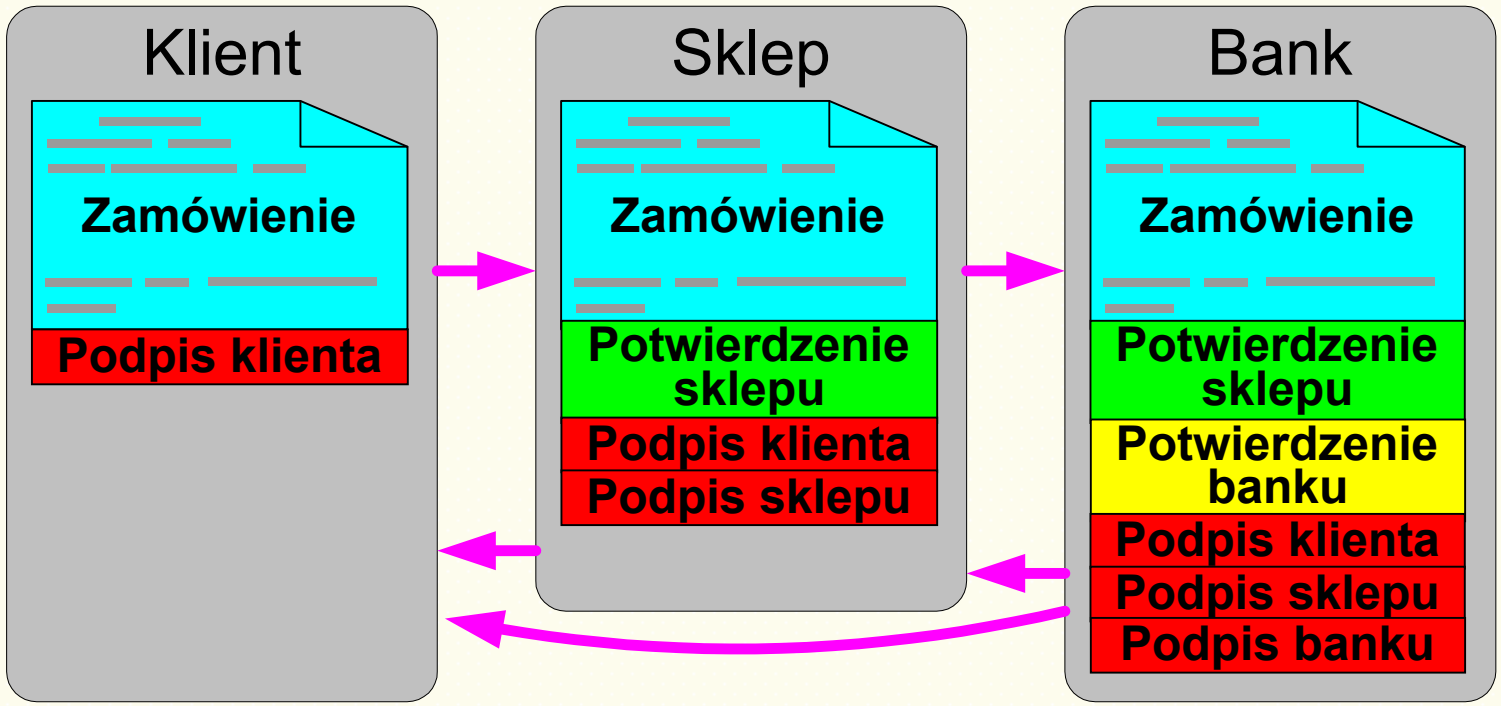
- Obsługa dokumentów modułowych
  - Jednostki (encje) zewnętrzne - obsługa
  - Dostęp do zdalnych zasobów
- Kodowanie polskich znaków (Unikod, windows-1250, ISO-8859-2)
- Atak Denial of Service:
  - „zapętlenie” przy deszyfrowaniu kluczy (XML Encryption)
  - Niepotrzebne sprowadzanie bardzo dużych plików z sieci (oba)

Problemy - prezentacja

- Inna zawartość dokumentu XML – inna prezentowana
- Ogólny problem prezentacji (programy, parametry programów, urządzenia)
- Dwa elementy:
  - Dokument XML (dane źródłowe)
  - Program prezentujący – zwykle osobny plik XSLT – „rodzaj programu źródłowego” dla procesora XSLT („interpretera”)
- Pliki XSLT najlepiej:
  - Stosować własne pliki XSLT, specjalizowane dla danego formatu dokumentów (np. FakturaNaszaXML) – najczęstszy i najłatwiejszy sposób
  - Zabezpieczać tak jak pliki XML (podpis / szyfrowanie łącznie z dokumentem XML z danymi). W przypadku XML Signature mamy dowód procesowy ewentualnego oszustwa
  - Stosować własne aplikacje (nie wykorzystujące XSLT)
- W elektronicznej wymianie danych (typowe zastosowanie dla XML) prezentacja nie musi być potrzebna

Podpisy wielokrotne

- Trzeci standard
- Kolejność weryfikacji wielokrotnych podpisów
- Zabezpieczenia procesów biznesowych – zmiany w dokumencie dokonywane przez wielu



Rozszerzenia

- **XAdES** (*XML Advanced Electronic Signature*) - European Telecommunications Standards Institute (ETSI)
- **Stemple czasowe**
- **Listy CRL** lub wskazania na nie
- **Podpisy długoterminowe** – dodawanie co jakiś czas nowych, bezpiecznych elementów podpisanych przez urząd certyfikujący

XML Encryption i XML Signature - podsumowanie

- Rożne rodzaje dozwolonych algorytmów kryptograficznych
  - Standardowe (potrójny DES, AES, RSA, SHA1, ...)
  - Niestandardowe - własne
- Zabezpieczanie
  - Fragmentów dokumentów XML
  - Wielu fragmentów jednym `Signature` lub `EncryptedData`
  - Wiele zabezpieczeń na raz
  - Tekstowe zabezpieczenia danych binarnych
- Integracja z językami XML, także XML Signature i XML Encryption
- Różne umiejscowienie zabezpieczeń względem chronionych danych
- Możliwość dodawania własnych rozszerzeń (np. stemple)

Dziękuję za uwagę...