Jerzy PEJAŚ, Imed EL FRAY, Tomasz HYLA

Zachodniopomorski Uniwersytet Technologiczny w Szczecinie, Wydział Informatyki, Katedra Inżynierii Oprogramowania

doi:10.15199/48.2015.11.46

Wiarygodna prezentacji danych do podpisania i weryfikacji

Streszczenie. *Wiarygodna prezentacja danych przed złożeniem podpisu elektronicznego oraz podczas jego weryfikacji jest jednym z kluczowych czynności, które musi być wykonana za pomocą „bezpiecznego urządzenia służącego do składania i weryfikacji podpisu elektronicznego". Ze złożeniem podpisu elektronicznego pod dokumentem lub jego weryfikacją może być związane określone zobowiązanie prawne podmiotu podpisującego lub weryfikującego dokument. Artykuł zawiera przegląd różnych technicznych sposobów prezentacji danych i ich ocenę pod kątem rzeczywistych zagrożeń ze strony celowych lub niezamierzonych modyfikacji prezentowanych treści. W artykule zaprezentowano także koncepcję wiarygodnego prezentowania dokumentów. Koncepcja ta polega na oddzieleniu treści podpisywanego dokumentu od formy jego prezentacji oraz zastosowaniu szeregu atrybutów związanych z podpisem dokumentów oraz modułów do ich prezentacji, które minimalizują ryzyko zmanipulowania prezentowanej zawartości podpisywanego lub weryfikowanego dokumentu.*

Abstract*. The trusted presentation of the signed or being signed data is one of the key problem, which should be solved in so called secure signature creation and verification devices. Electronic signature on the document or its verification results in a legal commitment of a signer or a verifier. The paper provides an overview of the various technical ways to present data and their assessment in terms of real risks from intentional or unintentional modification of the presented content. The paper also presents the concept of a trusted presentation of documents. This concept is based on the separation of the electronic document contents from its presentation and using a number of attributes associated with the document signature and module to its presentation, minimizing the risk to manipulate the contents of a document to be signed or verified. (**Trusted data presentation for signature and verification**).*

Słowa kluczowe: system do składania podpisów elektronicznych, zaufana przeglądarka, podpis elektroniczny, protokoły kryptograficzne.
Keywords: trusted viewer, e-signature, cryptographic protocols.

Wstęp

Rozporządzenie Parlamentu Europejskiego i Rady (UE) *w sprawie identyfikacji elektronicznej i usług zaufania w dniesieniu do transakcji elektronicznych na rynku wewnętrznym* [1] zrównuje kwalifikowany podpis elektroniczny z podpisem własnoręcznym. Sytuacja ta stworzyła warunki do dalszej rozbudowy bezpiecznych systemów do składania podpisów elektronicznych.

System do składania podpisu (ang. Signature Creation System, SCS, patrz [8]), pracujący w określonym środowisku (niepublicznym lub publicznym) powinien gwarantować podmiotowi podpisującemu bezpieczne złożenie podpisu elektronicznego. W praktyce systemy do składania podpisów elektronicznych narażone są na mniej lub bardziej wyszukane ataki (patrz także A. Jøsang [4]), których wynikiem może być sfałszowanie podpisu elektronicznego lub podpisywanego/weryfikowanego dokumentu elektronicznego.

Na potrzeby niniejszego artykułu przyjęto, że podpisu elektronicznego nie można sfałszować na skutek przełamania schematu podpisu elektronicznego lub zabezpieczeń bezpiecznego urządzenia do składania podpisu (SSCD). Współczesne schematy podpisu elektronicznego bazują na silnych algorytmach kryptograficznych opartych na tzw. trudnych problemach obliczeniowych, np. problemie faktoryzacji liczby całkowitej lub problemie logarytmu dyskretnego. Jednocześnie powszechnie akceptowane podejście do realizacji schematów podpisu zakłada wykorzystanie w roli SSCD identyfikacyjnej karty elektronicznej z kryptoprocesorem [8]. W karcie przechowywany jest klucz prywatny i realizowane są funkcje kryptograficzne z jego udziałem – klucz prywatny nigdy nie opuszcza karty.

Nie oznacza to jednak, że fałszerstwo jest niemożliwe w ogóle. Proces składania podpisu elektronicznego składa się przynajmniej z trzech kroków: (a) przygotowania dokumentu źródłowego przeznaczonego do podpisania, (b) podglądu dokumentu i przesłania go (całości, jego części lub tylko skrótu) przez aplikację podpisującą do SSCD, (c) wykonania przez komponent techniczny podpisu i odesłania go z powrotem do aplikacji. Z kolei proces weryfikacji składa się z dwóch etapów: (a) formalnej weryfikacji ważności podpisu elektronicznego i (b) prezentacji weryfikowanego dokumentu. Jest oczywiste, że przy słabej ochronie informacji przesyłanej pomiędzy elementami systemu (oprogramowanie podpisujące, SSCD, moduł prezentacji) każdy z tych elementów może otrzymywać dane wejściowe inne niż te, które w efekcie końcowym powinny być podpisane lub zweryfikowane przez użytkownika. Z tego powodu w praktyce przyjmowane są dwa skrajne założenia: (1) komputer lub system użytkownika musi być zaufany, (2) komputer lub system użytkownika nie musi być w ogóle zaufany.

W dalszej części artykułu, przy założeniu, że komputer lub system użytkownika jest niewiarygodny, zaproponowano rozproszony system do składania bezpiecznego podpisu wraz z funkcjonalnym modelem rozproszonej zaufanej przeglądarki dokumentów elektronicznych i atrybutów podpisu. System ten pozwala użytkownikowi na złożenie podpisu elektronicznego tylko wtedy, gdy posiadane przez niego bezpieczne urządzenia do składania podpisu *uzna* oferowane środowisko do składania podpisu za bezpieczne, tj. takie, które oferuje właściwą ochronę przed złośliwym oprogramowaniem, próbującym sfałszować treść dokumentu i/lub podpis elektroniczny.

Dokument elektroniczny – szanse i zagrożenia

Dokumentacja w postaci papierowej generuje wiele problemów [12]. *Największym z nich jest bezpowrotnie tracony czas na czynności związane z wyszukiwaniem dokumentów. Według najnowszych badań PricewaterhouseCoopers wyszukanie jednego dokumentu zajmuje pracownikowi średnio od 9 do 12 minut i kosztuje pracodawcę 120$, a odtworzenie utraconego dokumentu przynajmniej 250$. Z kolei przekazywanie dokumentów z rąk do rąk bardzo często prowadzi do ich kradzieże, zagubienia lub zniszczenia* (1 na 20 dokumentów jest gubiony oraz 1 na 100 kradziony). Przechowywanie dokumentów i zarządzanie nimi generuje kolejne koszty. Zamiana dokumentacji papierowej na odpowiadającą jej postać elektroniczną (tzw. dematerializacja dokumentu) przyczynia się do jej szybszego i sprawniejszego obiegu, szybszego dostępu do jej zawartości oraz jej efektywniejszego wykorzystania i łatwiejszego

monitorowania podczas obiegu w organizacji. Korzyści związane są również z obniżeniem wydatków organizacji na bieżącą obsługę administracyjną (papier, tusz do drukarki, itp.). Jednak oprócz wymienionych zalet dokumenty elektroniczne mają wady, m.in. łatwo je zmanipulować (np. w celu fałszowania zawartości dokumentu). Szerszy opis zagrożeń mających wpływ na bezpieczeństwo dokumentu elektronicznego przedstawiono w rozdziale „Taksonomia zagrożeń".

Dokument elektroniczny jest wirtualnym odpowiednikiem dokumentu papierowego. Przyjęcie jednak założenia, że elektroniczne dane (odpowiadające dokumentowi) są semantycznie równoważne dokumentowi wirtualnemu za każdym razem, gdy następuje odwołanie do nich, jest złożeniem wątpliwym. Jednym rozwiązaniem problemu wirtualizacji dokumentów a przed wszystkim zawartych w nich danych jest ich podpisywania oraz weryfikacji.

Podpisywane lub weryfikowane dane można podzielić na dwie podstawowe grupy: dokument elektroniczny oraz atrybuty podpisu. Z każdym dokumentem można związać jego treść (zawartość) i format zawartości, odpowiadający temu z atrybutów podpisu, który wyraża sposób kodowania podpisywanego lub weryfikowanego dokumentu. Z kolei atrybuty podpisu są zbiorem dodatkowych informacji, które są podpisywane lub weryfikowane razem z dokumentem.

Dokument elektroniczny i atrybuty podpisane

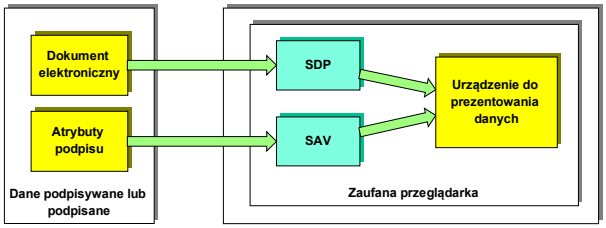
Na treść i format dokumentu ma wpływ szereg czynników, które mogą zaburzać ostateczną postać i status dokumentu przed jego podpisaniem/zweryfikowaniem lub w trakcie podpisywania/weryfikowania. Wśród najważniejszych należy wymienić następujące przypadki (ETSI EN 319 102-1 [8]):

- dokument może być w formacie procesora lub edytora tekstu i podlegać edycji, zaś prezentacja dokumentu może zależeć od aktualnej konfiguracji urządzenia umożliwiającego podgląd dokumentu; w praktyce podmiot podpisujący może więc widzieć dokument w innej postaci (nie tylko innej treści) niż „widziany|" przez weryfikatora,
- format dokumentu może być jednoznaczny i niemodyfikowalny (np. PDF, Postscript, ODA) zapewniający, że sposób jego prezentowania jest zawsze taki sam,
- w dokumencie może wystąpić ukryta informacja (np. makra, niewidoczny tekst, aktywne komponenty, wirusy), której obecności mogą być nieświadomi zarówno podmiot podpisujący, jak też weryfikator,
- dokument może być w postaci, z której, zanim zostanie zaprezentowany pomiotowi podpisującemu lub weryfikatorowi, musi zostać odwzorowany w inny format, lub też za każdym razem może być prezentowany w inny sposób, przy jednoczesnym zachowaniu jego semantyki (przykładami są formaty EDI, HTML, XML, SGML),
- dokument może zawierać obiekty danych podpisane przez inne osoby niż podmiot podpisujący.

Treść i format musi jednoznacznie wynikać z atrybutów podpisu, nierozłącznie związanych z podpisywanym lub weryfikowanym dokumentem. Atrybuty podpisu umożliwiają (ETSI EN 319 102-1 [8]):

- zidentyfikowanie podmiotu podpisującego oraz właściwego certyfikatu, który powinien być użyty do zweryfikowania podpisu oraz oceny rodzaju oświadczenia woli złożonego przez podmiot podpisujący,
- określenie przez podmiot intencji i ograniczeń interpretacyjnych w momencie składania podpisu elektronicznego; intencje i ograniczenia interpretacyjne są wyrażane w postaci tzw. polityki podpisu, która może być określona samodzielnie przez podpisującego lub przez inny podmiot i musi zostać zaakceptowana przez podpisującego,
- określenie formatu zawartości dokumentu i sposobu jego prezentowania weryfikatorowi zgodnie z intencją podmiotu podpisującego,
- określenie typu zobowiązania, precyzującego znaczenie podpisu w kontekście polityki zaakceptowanej przez podmiot podpisujący.

Z powyższego opisu wynika, że zaufanie do przeglądarki podpisanych danych nie może ograniczać się jedynie do wiarygodnego prezentowania treści dokumentu zgodnie z jego formatem (tzw. moduł prezentowania dokumentu, SDP), ale obejmować także wiarygodne prezentowanie atrybutów podpisu (tzw. moduł podglądu atrybutów podpisu, SAV). Oba moduły (patrz rys.1) są równie istotne i muszą być równie godne zaufania.



Rys.1. Zaufana przeglądarka podpisywanych lub podpisanych danych

Taksonomia zagrożeń

W procesie podpisywania lub weryfikowania podmiot podpisujący lub weryfikator mogą korzystać z dwóch typowych rozwiązań. W pierwszym rozwiązaniu funkcje podpisywania i weryfikowania dokumentu są zintegrowane z aplikacjami programowymi (zwykle w postaci tzw. wtyczek, ang. *plug-in*), które pozwalają na jego tworzenie, modyfikowanie i oczywiście podgląd. W drugim z rozwiązań stosuje się oddzielne aplikacje do składania i weryfikacji podpisu, wyposażone w samodzielny moduł prezentowania danych lub odwołujące się do aplikacji, za pomocą której dokument został utworzony lub zmodyfikowany.

W obu przypadkach możliwe jest takie przygotowanie złośliwego oprogramowania (np. Konia Trojańskiego), które będzie aktywnie działać [2]:

- wewnątrz aplikacji programowej, umożliwiającej przygotowanie lub zweryfikowanie danych, pośród których mogą wystąpić elementy aktywnie wpływające na ich semantykę,
- pomiędzy aplikacją programową a wtyczką lub aplikacja podpisującą/weryfikująca,
- wewnątrz wtyczki lub aplikacji podpisującej/weryfikującej,
- jako sterownik urządzenia pomiędzy aplikacją podpisującą/weryfikującą lub wtyczką, a SSCD.

Przygotowanie tego typu złośliwego oprogramowania nie wymaga zbyt dużego wysiłku, ani nadzwyczajnej wiedzy. Większość z powszechnie stosowanych typów dokumentów elektronicznych przygotowanych za pomocą procesorów tekstu i arkuszy kalkulacyjnych jest przechowywana w plikach, które zawierają dynamiczne treści – makra, formuły, pola dynamiczne, kod napisany np. w Visual Basic, dowiązania do innych dokumentów. Można w bardzo prosty sposób napisać złośliwy kod, który zmodyfikuje zawartość dokumentu, nie pozostawiając żadnego śladu swej aktywności. Trochę trudniejsze, ale także możliwe, jest fałszowanie atrybutów podpisanych, np.

pokazując inną politykę podpisu weryfikatorowi niż podmiotowi podpisującemu.
W tabelach 1 i 2 zebrano podstawowe zagrożenia i sposoby ataków na dokument elektroniczny i atrybuty podpisane (patrz także ETSI TS 119 101 [7] i K. Kain, i in. [2]). Zestawienie to nie pretenduje do miana wyczerpującego, ale zawiera naszym zdaniem najbardziej istotne elementy, które powinny być podstawą oceny bezpieczeństwa każdego systemu do składania bezpiecznego podpisu.

Tabela 1. Zagrożenia i wymagania bezpieczeństwa nakładane na moduł prezentacji dokumentu elektronicznego (SDP)

Kod zagrożenia	Typ zagrożenia	Przykład zagrożenia	Wymaganie bezpieczeństwa
Zag_1	**Niewidoczne parametry** – zagrożenie może wystąpić wtedy, gdy podgląd dokumentu nie jest w pełni określony przez sam dokument i zależy dodatkowo od innych parametrów lub aktywnego kodu	Zawartość i wygląd dokumentu manipulowane w zależności od: • czasu jego prezentowania, • parametrów przeglądarki, tj. jej identyfikatora, urządzenia prezentującego, systemu operacyjnego lub kontekstu danych, • aktualnie wykonywanej operacji przez przeglądarkę, • istnienia lub zwartości pliku zdalnie kontrolowanego przez adwersarza, zwłaszcza wtedy, gdy plik ten jest wskazany za pomocą adresu URL.	Moduł prezentacji dokumentu elektronicznego (SDP) musi ostrzegać podmiot podpisujący i weryfikatora o obecności niewidocznych parametrów w dokumencie
Zag_2	**Sfałszowana zawartość** – adwersarz próbuje skutecznie zmienić zawartość dokumentu	Prezentowana zawartość dokumentu może zostać zmieniona: • przed momentem złożenia podpisu, • po złożeniu podpisu.	Moduł prezentacji SDP dokumentu musi zapewnić, aby w dokumencie po jego zaprezentowaniu nie zostały umieszczone żadne zmiany (statyczne lub dynamiczne)
Zag_3	**Natura zmian** – określa w jaki sposób podglądana zmieniona zawartość dokumentu wpływa na podgląd całego dokument	Podgląd zmienionej części dokumentu: • nie ma wpływu na podgląd całego dokumentu, jeśli zmieniona część ma naturę statyczną, • w przypadku zawartości dynamicznej zmieniona część modyfikuje prezentowane dane w momencie otwierania dokumentu do podglądu; atak jest efektywny jedynie w przypadku weryfikowania podpisu złożonego pod oryginalnym dokumentem. • wpływa na cały dokument włącznie z podpisem, jeśli w momencie dynamicznej modyfikacji zawartości dokumentu zmodyfikowany został jednocześnie podpis elektroniczny	Moduł prezentacji SDP musi uniemożliwiać dodawanie statycznych lub dynamicznych fragmentów, zwłaszcza w momencie korzystania z zewnętrznych przeglądarek, np. przeglądarek plików Microsoft Word. Opcjonalnie moduł musi umożliwić dołączenie do dokumentu przed jego podpisaniem atrybutu określającego format zawartości.
Zag_4	**Format zawartości –** brak lub niewłaściwy format zawartości dokumentu.	Weryfikator może błędnie zinterpretować zawartość dokumentu oraz intencje podmiotu podpisującego lub przeglądarka nie będzie w stanie w ogóle zaprezentować wskazanego dokumentu.	Moduł SDP musi: • umożliwić dołączenie formatu zawartości bezpośrednio w treści dokumentu lub pośrednio jako jeden z elementów zbioru atrybutów podpisu, • wyraźnie wskazywać obsługiwane formaty zawartości i ostrzegać w przypadku formatów nieobsługiwanych oraz o wynikających z tego konsekwencji.
Zag_5	Sfałszowany element **- podpisanie sfałszowanego elementu umieszczonego w podpisywanym dokumencie**	Podmiot podpisujący może nieświadomie podpisać (kontrasygnować) wbudowany w dokument podpisany obiekt danych, którego treść została sfałszowana, lub sfałszowany został podpis elektroniczny.	Moduł prezentacji dokumentu musi informować podmiot podpisujący o obecności w jego treści innych podpisanych obiektów danych.

Techniczne metody prezentowania danych do podpisania lub weryfikacji

Techniczne metody prezentowania podpisywanych, podpisanych lub weryfikowanych danych są odzwierciedleniem dwóch podstawowych podejść do budowania aplikacji do składnia i weryfikowania podpisów (patrz także rozdz.2), polegających na zastosowaniu wtyczek w standardowych aplikacjach oraz aplikacji dedykowanych. Wtyczki oraz aplikacje dedykowane mogą korzystać z przeglądarek wewnętrznych (wbudowanych w ich funkcje) lub z przeglądarek zewnętrznych (zwykle rolę taką odgrywają aplikacje, które posłużyły wcześniej do utworzenia podpisywanego lub weryfikowanego dokumentu).

Niezależnie od przyjętego rozwiązania zaufane przeglądarki muszą eliminować zagrożenia wymienione w Tab.1 i Tab.2. Sposób i zakres ich eliminowania zależy jednak od zastosowanego podejścia. Przykłady kilku takich podejść przedstawiono poniżej (patrz także A. Alsaid, i in. [5], J. Pejaś, i in. [13]).

Blokowanie lub ograniczanie dynamicznie zmiennej zawartości

Jedno z rozwiązań problemu dynamicznych treści występujących w dokumencie lub atrybutach podpisanych polega na ich blokowaniu i uniemożliwieniu ich umieszczania w treści dokumentu lub pośród atrybutów podpisanych. Takie podejście zostało zaproponowane przez Spalkę i in. [6], może jednak czynić niektóre dokumenty bezużytecznymi. Ci sami autorzy w innym swoim artykule [3] zaproponowali dwa inne rozwiązania. Jedno z nich polega na ograniczeniu aktywnej zawartości dokumentu zamiast jej blokowania. Ograniczanie aktywnej zawartości dokumentu powinno być tak zaimplementowane, aby gwarantowało, iż dynamiczna zawartość nie będzie miała żadnego wpływu na semantykę dokumentu. Rozwiązanie takie wymagałoby wprowadzenia wielu zmian do istniejącego oprogramowania. Drugie z proponowanych rozwiązań polega na zastosowaniu tzw. „piaskownicy” (ang. *sandbox*), związanej z aplikacją, za pomocą której przygotowany został dokument. „Piaskownica” jest zbiorem

wszystkich parametrów definiujących zdeterminowany kontekst (m.in. nazwę użytkownika, nazwę komputera, adres sieciowy, parametry aplikacji), w którym był tworzony i podpisywany dokument. Kontekst zapisywany jest w atrybutach podpisu i po przekazaniu do weryfikatora podpisu umożliwia odtworzenie parametrów tego samego środowiska, w którym dokument był prezentowany i podpisywany przez podmiot podpisujący.

Tabela 2. Zagrożenia i wymagania bezpieczeństwa nakładane na moduł prezentacji atrybutów podpisanych

Kod zagrożenia	Typ zagrożenia	Przykład zagrożenia	Wymaganie bezpieczeństwa
Zag_6	**Niewłaściwy certyfikat**	Podpisujący dokument może niezgodnie z intencjami związać podpisywany dokument z innym certyfikatem, wywołując w ten sposób inne zobowiązania od zamierzonych	Moduł prezentacji atrybutów podpisanych (SAV) powinien umożliwić podmiotowi podpisującemu podgląd podstawowych elementów certyfikatu, którego atrybuty zostaną dołączone do atrybutów podpisu.
Zag_7	**Podpisanie niewłaściwego atrybutu podpisu**	Podmiot podpisujący podpisał omyłkowo niewłaściwe atrybuty podpisu	Moduł SAV musi umożliwić podpisującemu podgląd atrybutów podpisu. Jednocześnie moduł SAV musi zagwarantować, że prezentowane atrybuty podpisu są tymi samymi, które zostały wcześniej wskazane przez podmiot podpisujący, a następnie przez niego podpisane.
Zag_8	**Przypadkowa lub złośliwa modyfikacja dokumentu przez aplikację podpisującą**	Treść wskazanego dokumentu została zmieniona na skutek omyłkowej lub złośliwej zmiany atrybutów podpisu.	Atrybutom podpisanym należy zapewnić ochronę ich integralności i autentyczności. Moduł SAV musi ostrzegać podpisującego przed obecnością niewidocznego tekstu, makr lub aktywnego kodu w atrybucie.
Zag_9	**Podpisane atrybuty podpisu mogą automatycznie zmieniać się przed ich podglądem**	Kod atrybutów podpisu zawiera aktywne komponenty, które mogą zmienić sposób ich prezentowania lub semantykę.	Moduł SAV musi ostrzegać podpisującego lub weryfikatora przed obecnością w atrybutach podpisu jakiegokolwiek aktywnego komponentu (np. makr).

Format XML

Inne stosowane podejście to formatu XML (W3C 2003 [9]): dowolny dokument elektroniczny konwertuje się do formatu XML, a następnie do tak uzyskanego dokumentu stosuje się standard tworzenia podpisu cyfrowego XML-DSig (Bartel i in. [10]). Podejście to rozwiązuje wiele problemów, ale dynamiczna zawartość może nadal występować w plikach XML. Cecha ta jest groźna przede wszystkim dla weryfikatora w momencie, gdy przed obejrzeniem dokumentu zachodzi konieczność przekonwertowania pliku XML do oryginalnej postaci dokumentu: jeśli w treści pliku XML występują dynamiczne wstawki, to mogą zostać uaktywnione i zmienić pierwotną semantykę dokumentu. Twórcy normy XML-DSig są świadomi tej wady formatu XML i zalecają, aby w przypadku podpisywania dokumentu XML aplikacja podpisująca podpisywała wszystkie zewnętrzne dokumenty, a sam dokument XML jedynie na nie wskazywał.

Oczywiście można, podobnie jak w przypadku podejścia opisanego w rozdz.3.1, zabronić lub ograniczyć zakres stosowania w dokumentach XML dynamicznej zawartości występującej w dokumentach oryginalnych (przed konwersją), ale w wielu przypadkach sprawi to, że oryginalne dokumenty stracą swoje pierwotne znaczenie, nie mówiąc już o pozbawieniu użytkownika możliwości wprowadzania zmian do jego treści.

Rozbiór składni (ang. parser) dokumentu

Innym sposobem wyeliminowania niektórych zagrożeń wymienionych w Tab.1 i Tab.2 jest zaimplementowanie aplikacji podpisującej z własnym *parserem* dokumentu. Przy takim podejściu za każdym razem, gdy podmiot podpisujący zamierza złożyć podpis, aplikacja dokonuje rozbioru składni dokumentu i usuwa z jego treści wszystkie aktywne elementy. Rozwiązanie to wymaga jednak, aby aplikacja podpisująca obsługiwała wszystkie lub przynajmniej najbardziej popularne formaty dokumentów. Jednak nawet w tym ostatnim przypadku zadanie to może być trudne do zrealizowania, ponieważ nie zawsze dostępne są dokładne specyfikacje tych formatów.

Graficzna postać dokumentu

Koncepcje „*to co widzisz jest tym co podpisujesz*" i „*to co widzisz jest tym co zostało podpisane*" zostały wprowadzone po to, aby rozwiązać problem niejednoznaczności semantycznej dokumentu, wynikający z obecności w jego treści elementów dynamicznych. Scheibelhofer [11] zaproponował najprostsze z możliwych rozwiązań: najpierw tworzona jest graficzna postać dokumentu elektronicznego, która jest następnie podpisywana. Rozwiązanie ma wiele zalet (patrz także rozdz.4), ale też jedną wadę – pliki graficzne mają zwykle o wiele większą objętość niż dokumenty oryginalne.

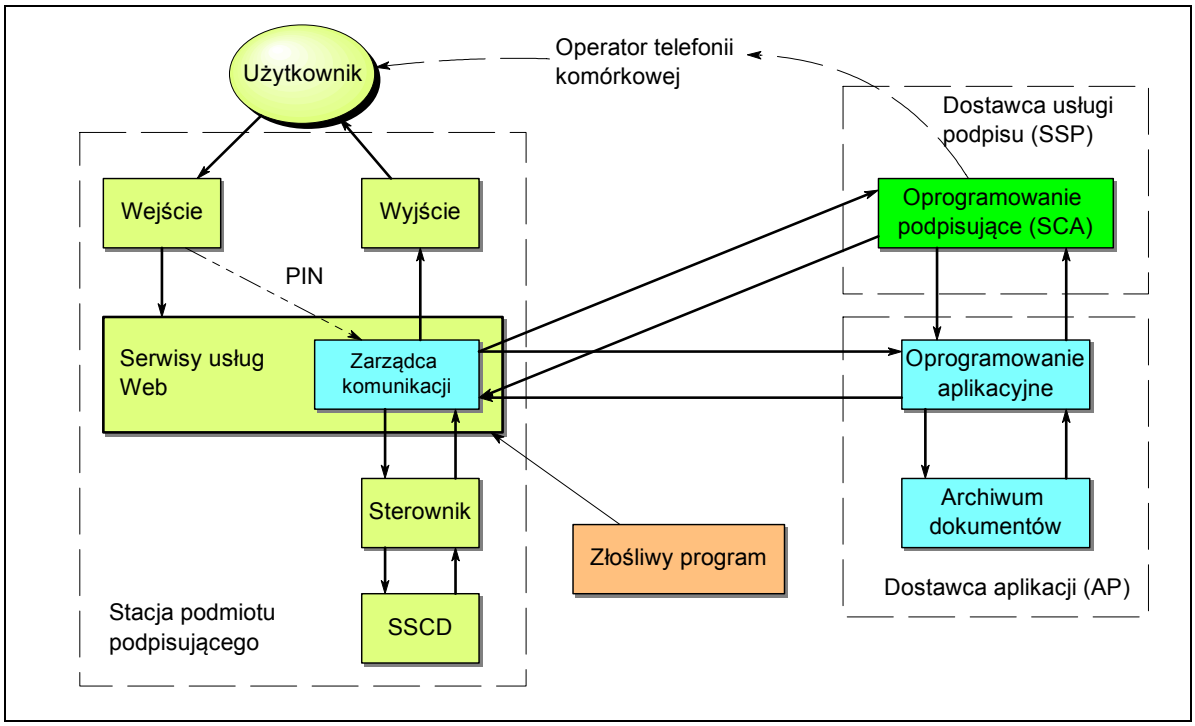
Architektura urządzenia do składania podpisu w środowisku rozproszonym i propozycja prezentowania danych do podpisania

W prezentowanej na rys.2. propozycji architektury systemu do składania podpisu w środowisku rozproszonym zakłada się, że oprogramowanie podpisujące (publiczne) do składania podpisu rozdzielone jest w sensie fizycznym i przestrzennym od bezpiecznego urządzenia do składania podpisu. Jest ono zlokalizowane na serwerze zaufanej strony trzeciej (tzw. Signature Service Provider – SSP), współpracującej z jednym lub wieloma dostawcami wzorców dokumentów (dostawcami aplikacji internetowych, tzw. Application Providers – AP), zarejestrowanymi przez SSP. Fakt rejestracji tożsamy jest z udostępnieniem przez AP odpowiednich mechanizmów umożliwiających SSP sprawdzenie autentyczności i integralności niezależnych od podmiotu podpisującego wzorców dokumentów

(w szczególności dzięki zastosowaniu podpisanych przez AP komponentów).

Wzorce dokumentów pobierane są z witryny usługowej AP przez podmiot podpisujący i wypełniane treścią semantyczną, obliczana jest wartość funkcji skrótu dla części dokumentu XML zredagowanej przez podmiot podpisujący, lecz nie jest ona odsyłana do AP, lecz przechowywana dla potrzeb porównania w końcowej fazie składania podpisu z odpowiednią wartością przesłaną bezpiecznym kanałem przez SSP. Do archiwum dokumentów przygotowanych do podpisania, a zlokalizowanym na serwerze AP, odsyłany jest tylko wypełniony treścią wzorzec dokumentu, po czym sesja komunikacyjna podmiotu podpisującego z dostawcą usługi (AP) jest zamykana.

W kolejnej fazie podpisywania podmiot podpisujący nawiązuje sesję komunikacyjną z wybranym przez siebie SSP. Oba podmioty muszą mieć uzgodnione wcześniej mechanizmy ustanawiania bezpiecznej ścieżki i bezpiecznego kanału, co więcej, wymagane jest także zapewnienie realizacji usługi poufności podczas przesyłania hasła (numeru PIN) podmiotu podpisującego do oprogramowania podpisującego zainstalowanego na serwerze SSP. Podmiot podpisujący wskazuje (podając co najmniej identyfikator AP oraz identyfikator wypełnionego przez siebie treścią dokumentu) dokument do podpisania „zdeponowany" poprzednio w archiwum dokumentów wskazanego AP. SSP pobiera ten dokument z archiwum, sprawdza jego integralność (w szczególności elementów wzorcowych dokumentu podpisanych przez AP) i oblicza wartość funkcji skrótu treści semantycznej zredagowanej przez podmiot podpisujący (wraz z ewentualnymi atrybutami podpisu). Dokument jest konwertowany przez SSP do formatu graficznego i w takiej formie, opatrzonej podpisem SSP, prezentowany podmiotowi podpisującemu (jako opcję umożliwiającą przeciwdziałanie atakom powtórzeniowym stosuje się wbudowanie w graficzną postać danych do podpisania niepowtarzalnych komponentów losowych, np. pewnej formy tzw. znaków wodnych).



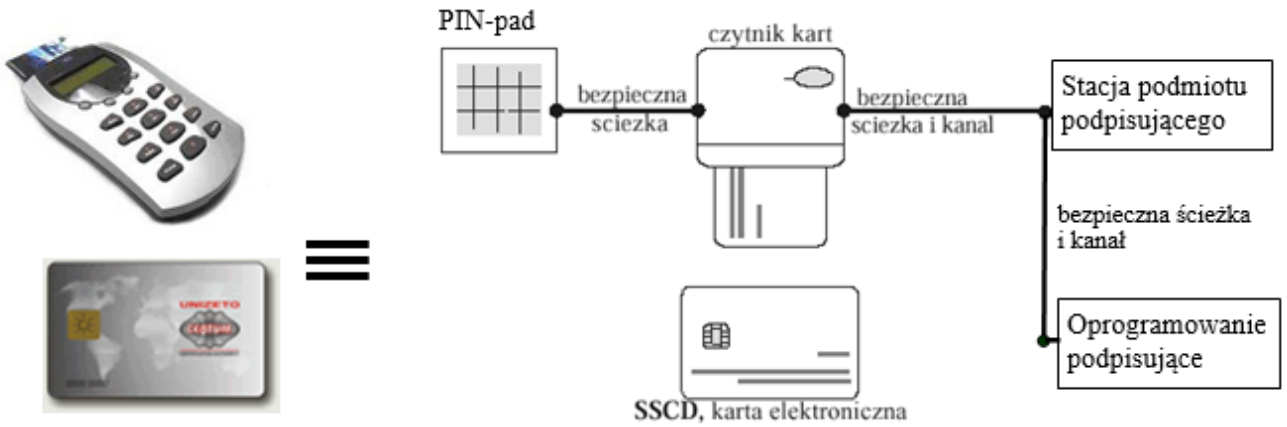
Rys. 2 Szczegółowa architektura urządzenia do bezpiecznego składania podpisu w środowisku rozproszonym (publicznym)

Akceptacja prezentowanego dokumentu do podpisania przez podmiot podpisujący inicjuje proces ustanawiania bezpiecznej ścieżki i bezpiecznego kanału między aplikacją podpisującą i bezpiecznym urządzeniem do składania podpisu. Niezależnie od tego, czy proces składania podpisu jest realizowany na stanowisku indywidualnym użytkownika (podmiotu podpisującego), czy też w „klasycznym" środowisku publicznym (infomat, kiosk internetowy, itp.) ze względu na konieczność zachowania poufności danych uwierzytelniających podmiot podpisujący (PIN, hasło) należy zastosować jako urządzenie wejściowe odpowiednik „PIN pad". W tym drugim przypadku niezbędne jest ustanowienie niezależnej bezpiecznej ścieżki między SSP a stanowiskiem, na którym będzie składany podpis, odmiennej od bezpiecznej ścieżki między SSCD a oprogramowaniem podpisującym w wersji wykorzystującej dodatkowe cechy funkcjonalne czytnika (patrz rys.3), lub tworzonej bezpośrednio między PIN-pad'em a serwerem SSP.

Dane uwierzytelniające wraz ze skrótem danych do podpisania są przesyłane bezpieczną ścieżką (pełniącą jednocześnie rolę bezpiecznego kanału, ale z zaimplementowaną usługą poufności, np. z tzw. „secure messaging'iem" dla identyfikacyjnych kart elektronicznych stosowanych jako SSCD) do bezpiecznego urządzenia do składania podpisu. W tym momencie dane są podpisywane i odsyłane do SSP.

Zanim SSP odeśle uzyskany podpis do archiwum dokumentów AP, w celu skojarzenia go z właściwym dokumentem zredagowanym przez podmiot podpisujący i zdeponowanym w zasobach AP, dokument może być ponownie (na żądanie podmiotu podpisującego) prezentowany w formacie graficznym (podpisanym przez SSP, z opcjonalnym niepowtarzalnym wzorcem odróżniającym go od formy graficznej prezentowanej przed podpisaniem) podmiotowi podpisującemu w celu uzyskania ostatecznej akceptacji treści podpisanego dokumentu. Brak takiej akceptacji przerywa proces podpisywania i SSP jest

zobowiązany uzyskany podpis zniszczyć. W przeciwnym przypadku proces podpisywania jest kończony, zaś podpisany dokument zajmuje miejsce swojego niepodpisanego pierwowzoru w archiwum dokumentów AP.



Rys.3. Koncepcja wykorzystania urządzenia typu PIN-pad do ustanowienia bezpiecznej ścieżki

Wnioski i podsumowanie

W artykule przedstawiono propozycje architektury rozproszonego systemu do składania podpisu elektronicznego, który spełnia wymagania obowiązujące Ustawy z dnia 18 września 2001 r o podpisie elektronicznym, a także wymagania Rozporządzenia Parlamentu Europejskiego i Rady dnia 23 lipca 2014 r. [1]. W szczególności dotyczy to wiarygodnego podglądu danych przed ich podpisaniem oraz w trakcie weryfikacji.

Analiza stosowanych aktualnie rozwiązań pokazuje, że trudno jest zastosować jedno uniwersalne podejście do podpisywania i weryfikacji danych dowolnego typu i o dowolnej semantyce. Dlatego, w przeciwieństwie do dotychczasowych propozycji, w artykule zaproponowano nałożenie określonych ograniczeń na podpisywane i weryfikowane dane oraz zastosowanie podejścia polegającego na rozproszeniu i kumulowaniu zaufania. Uzyskany w efekcie rozproszony moduł wiarygodnej prezentacji podpisywanych i weryfikowanych danych wydaje się być odporny na wszystkie zagrożenia, wymienione w Tab.1 i Tab.2. Praktyczny przykład tego typu systemu do składania podpisu elektronicznego przedstawiono w [14].

LITERATURA

[1] Rozporządzenie Parlamentu Europejskiego i Rady (UE) nr 910/2014 dnia 23.07.2014r. *w sprawie identyfikacji elektronicznej i usług zaufania w odniesieniu do transakcji elektronicznych na rynku wewnętrznym*, Dz. U. UE, L 257/73, 28 sierpnia 2014 r.

[2] Kain K., Smith S.W., Asokan R., *Digital signatures and electronic documents: A cautionary tale*, w B. Jerman-Blazic & T. Klobucar, eds, *Advanced Communications and Multimedia Security*, IFIP TC6/TC11 Sixth Joint Working Conference on Communications and Multimedia Security, September 26-27, 2002, Portoroz, Slovenia, Vol. 228 of IFIP Conference Proceedings, Kluwer Academic, Boston, MA, pp. 293–308

[3] Spalka A., Cremers A.B., Langweg H., *Trojan Horse Attacks on Software for Electronic Signatures*, Informatica 26 (2002) 191-203 pp.191-204

[4] Jøsang A., Povey D., Ho *What You See is Not Always What You Sign*, AUUG2002, Melbourne, 4-6 September 2002

[5] Alsaid A., Mitchell Ch.J., *Digitally Signed Documents – Ambiguities and Solutions*, Information Security Group, Royal Holloway, University of London, Egham, Surrey TW20 0EX, UK

[6] Spalka A., Cremers A.B., Langweg H., *Protecting the creation of digital signatures with trusted computing platform technology against attacks by trojan horse programs*, w M. Dupuy & P. Paradinas, eds, *Proceedings of the IFIP SEC 2001*, Kluwer Academic, Boston, MA, pp. 403–420

[7] ETSI TS 119 101 *Electronic Signatures and Infrastructures (ESI); Policy and Security Requirements for Electronic Signature Creation and Validation*, Draft, v0.0.3, January 2014

[8] ETSI EN 319 102-1 *Electronic Signatures and Infrastructures (ESI); Procedures for Creation and Validation of AdES Digital Signatures; Part 1: Creation and Validation*, Draft v1.0.0, July 2015

[9] W3C Recommendation (2003) *Extensible markup language (XML)*, http://www.w3.org/XML

[10] M. Bartel, J. Boyer, B. Fox, B. LaMacchia, E. Simon (2002) *XML-Signature Syntax and Processing*, http://www.w3.org/TR/xmldsig-core

[11] Scheibelhofer K., *Signing XML Documents and the Concept of 'What You See Is What You Sign'*, Master's thesis, Institute for Applied Information Processing and Communications, Graz University of Technology, 2001.

[12] *Zalety elektronicznego obiegu dokumentów a wady papierowego*, http://www.progmate.pl/pl/articles/differences (informacja dostępna 10 września 2015 r.)

[13] Pejaś J., Zawalich M. *Visual Cryptography Methods as a Source of Trustworthiness for the Signature Creation and Verification Systems*, w Advances in information processing and protection, Springer, 2007, str. 225-239

[14] Pejaś J., El Fray I., Ruciński A., *Authentication protocol for software and hardware components in distributed electronic signature creation system*, Przegląd Elektrotechniczny, R. 88, Nr 10b, 2012, str. 192-197

***Autorzy**: dr hab. inż. Jerzy Pejaś, Zachodniopomorski Uniwersytet Technologiczny w Szczecinie, Katedra Inżynierii Oprogramowania, ul. Żołnierska 52, 71-210 Szczecin E-mail: jpejas@zut.edu.pl; dr hab. inż. Imed El Fray, Zachodniopomorski Uniwersytet Technologiczny w Szczecinie, Katedra Inżynierii Oprogramowania, ul. Żołnierska 52, 71-210 Szczecin E-mail: ielfray@zut.edu.pl; dr inż. Tomasz Hyla, Zachodniopomorski Uniwersytet Technologiczny w Szczecinie, Katedra Inżynierii Oprogramowania, ul. Żołnierska 52, 71-210 Szczecin E-mail: thyla@zut.edu.pl.*