Podpis cyfrowy a systemy rozproszone

Michał Lewowski

26 marca 2009r.

Spis treści

Czym jest podpis ... dla purystów

Podpis cyfrowy

Podpis cyfrowy (ang. digital signature) to dane dołączone do danych lub ich przekształcenie kryptograficzne, które pozwala odbiorcy danych udowodnić pochodzenie danych i zabezpieczyć je przed fałszerstwem.

Podpis elektroniczny

Podpis elektroniczny (ang. electronic signature) określony przez unijną dyrektywę to operacja podpisywania konkretnych danych (dokumentu) przez osobę fizyczną.

To z dyrektyw Unii Europejskiej i jak widać to nie do końca to samo.

Czym jest podpis ... dla purystów

W Polsce powyższe terminy używane raczej wymiennie, a w polskim prawie:

Podpis elektroniczny

Podpis elektroniczny to dane w postaci elektronicznej, które wraz z innymi danymi, do których zostały dołączone lub z którymi są logicznie powiązane, służą do identyfikacji osoby składającej podpis elektroniczny.

Bezpieczny podpis elektroniczny

Bezpieczny podpis elektroniczny

- Jest przyporządkowany wyłącznie do osoby składającej ten podpis
- Jest sporządzany za pomocą podlegających wyłącznej kontroli osoby składającej podpis elektroniczny bezpiecznych urządzeń służących do składania podpisu elektronicznego i danych służących do składania podpisu elektronicznego
- Jest powiązany z danymi, do których został dołączony, w taki sposób, że jakakolwiek późniejsza zmiana tych danych jest rozpoznawalna

Inne ważne wymagania

Osoba składająca podpis elektroniczny

Osoba fizyczna posiadająca urządzenie służące do składania podpisu elektronicznego, która działa w imieniu własnym albo w imieniu innej osoby fizycznej, prawnej albo jednostki organizacyjnej nie posiadającej osobowości prawnej.

Inne ważne wymagania

Dane służące do składania podpisu elektronicznego

Niepowtarzalne i przyporządkowane osobie fizycznej dane, które są wykorzystywane przez tę osobę do składania podpisu elektronicznego,

Certyfikat

Certyfikat

Certyfikat w kryptografii to dane podpisane cyfrowo przez stronę, której ufamy - Certificate Authority, chociaż tak naprawdę każdy może oczywiście wygenerować sobie certyfikat i podpisywać nim dane lub kolejne certyfikaty przyjmując rolę CA.

Certyfikaty mogą służyć do podpisywania danych, a zakres ich stosowania (np. czy można podpisać nim kolejne certyfikaty i jaką moc można im przydzielić) ustalają odpowiednie wpisy w samym certyfikacie i polityka certyfikacji CA.

Certyfikat a ustawa

Certyfikat

Certyfikat w rozumieniu Ustawy o podpisie elektronicznym jest elektronicznym zaświadczeniem, za pomocą którego dane służące do weryfikacji podpisu elektronicznego są przyporządkowane do osoby składającej podpis elektroniczny i które umożliwiają identyfikację tej osoby.

Patrząc od strony informatycznej certyfikat jest zwykle plikiem w którym zostały m.in. zgromadzone informacje o podpisującym, jego klucz publiczny oraz o urzędzie certyfikacji wydającym dany certyfikat.

Certyfikat kwalifikowany

Certyfikat kwalifikowany

Certyfikat kwalifikowany to certyfikat wystawiony przez uprawnione Centrum Certyfikacji. W stosunku do zwykłych certyfikatów musi on spełnić dodatkowe wymagania związane z bezpieczeństwem oraz weryfikacją tożsamości osoby tym certyfikatem się posługującej.

Certyfikat kwalifikowany

Certyfikat kwalifikowany

Ważną cechą certyfikatu kwalifikowanego jest jego ściśle określone (ograniczone) przeznaczenie - certyfikat ten służy do potwierdzenia wiedzy i woli osoby podpisującej.

Certyfikat kwalifikowany nie może służyć np. do szyfrowania, ani podpisywania poczty elektronicznej, czy logowania do systemu. Te dodatkowe funkcjonalności zapewniają dopiero tzw. certyfikaty komercyjne (ale one mają z kolei słabsze umocowania prawne jeżeli chodzi o podpisywanie plików).

Polityka certyfikacji

Polityka certyfikacji

- Dokument określający zasady według których CA wydaje certyfikaty swoim klientom
- Określa np. dla jakiego typu certyfikatów jakie dane są weryfikowane
- Określa też np. zakres odpowiedzialności CA, jak często mają być publikowane listy CRL itp.

Co zawiera certyfikat?

Co zawiera certyfikat?

- Klucz publiczny

Co zawiera certyfikat?

Co zawiera certyfikat?

- Klucz publiczny
- Nazwa zwyczajowa (np. imię i nazwisko, pseudonim, nazwa firmy)

Co zawiera certyfikat?

Co zawiera certyfikat?

- Klucz publiczny
- Nazwa zwyczajowa (np. imię i nazwisko, pseudonim, nazwa firmy)
- Nazwa organizacji

Co zawiera certyfikat?

Co zawiera certyfikat?

- Klucz publiczny
- Nazwa zwyczajowa (np. imię i nazwisko, pseudonim, nazwa firmy)
- Nazwa organizacji
- Jednostka organizacyjna

Co zawiera certyfikat?

Co zawiera certyfikat?

- Klucz publiczny
- Nazwa zwyczajowa (np. imię i nazwisko, pseudonim, nazwa firmy)
- Nazwa organizacji
- Jednostka organizacyjna
- Przeznaczenie certyfikatu - np. niezaprzeczalność, szyfrowanie poczty, podpisywanie poczty, logowanie

Co zawiera certyfikat?

Co zawiera certyfikat?

- Czas do kiedy certyfikat jest ważny

Co zawiera certyfikat?

Co zawiera certyfikat?

- Czas do kiedy certyfikat jest ważny
- Informacje o wystawcy certyfikatów

Co zawiera certyfikat?

Co zawiera certyfikat?

- Czas do kiedy certyfikat jest ważny
- Informacje o wystawcy certyfikatów
- Sposób weryfikacji certyfikatu (adres listy CRL)

Co zawiera certyfikat?

Co zawiera certyfikat?

- Czas do kiedy certyfikat jest ważny
- Informacje o wystawcy certyfikatów
- Sposób weryfikacji certyfikatu (adres listy CRL)
- Adres polityki certyfikacji

Co zawiera certyfikat?

Co zawiera certyfikat?

- Czas do kiedy certyfikat jest ważny
- Informacje o wystawcy certyfikatów
- Sposób weryfikacji certyfikatu (adres listy CRL)
- Adres polityki certyfikacji
- Opcjonalnie można przechowywać właściwie wszystko: próbkę głosu właściciela, informacje biometryczne, zdjęcie itp.

Co jest potrzebne by mieć certyfikat kwalifikowany?

Co jest potrzebne by mieć certyfikat kwalifikowany?

- Dowód osobisty
- Czasami drugi dokument tożsamości - np. paszport lub prawo jazdy
- Pieniądze (kilkaset złotych)

Bezpieczne urządzenie

Czyli czym możemy składać podpis?

W Polsce podpis kwalifikowany może być składany wyłącznie przy pomocy oprogramowania posiadającego "deklarację zgodności z ustawą". Takie oprogramowanie wraz z komponentem technicznym nosi nazwę "bezpiecznego urządzenia". A w sumie to ani bezpieczne ani urządzenie.

W Polsce dodatkowo nieścisłość z Unią Europejską. Tam SSCD (bezpieczne urządzenie) to tylko komponent techniczny, u nas bezpieczne urządzenie to komponent i oprogramowanie (czyli może też np. Windows?).

Jakie ograniczenia rodzi bezpieczne urządzenie?

Żeby w Polsce podpis miał moc prawną musi być

- Złożony certyfikatem kwalifikowanym
- Złożony przez bezpieczne urządzenie (bezpieczne w sensie ustawy; posiadające deklarację zgodności)

Problem

Czyli podpis złożony bezpośrednio w Wordzie (przez mechanizmy Worda) nie jest w świetle polskiego prawa bezpieczny (a w prawie istnieje tylko pojęcie podpisu bezpiecznego) nawet jeżeli złożymy go certyfikatem kwalifikowanym!

Cytat

O bezpieczeństwie w Internecie

"Using encryption on the Internet is the equivalent of arranging an armored car to deliver credit-card information from someone living in a cardboard box to someone living on a park bench."

Professor Gene Spafford

Nośnik certyfikatu i klucza prywatnego

Kupując certyfikat kwalifikowany dostaniemy go na karcie kryptograficznej (wraz z urządzeniem i oprogramowaniem). Tam też będzie znajdował się nasz klucz prywatny - dodatkowo zabezpieczony 4-cyfrowym PIN-em.

Główny wymóg jest taki, że klucz prywatny nie może opuścić karty (inaczej urządzenie nie jest bezpieczne). Cała logika szyfrowania musie wykonać się na karcie i musi wykonać ją karta. Składać wiadomość w odpowiednim formacie można już poza kartą (ale jeżeli chcemy mieć podpis kwalifikowany to możemy i tak podpisywać tylko oprogramowaniem, które ma certyfikat zgodności z ustawą).

Co umie karta?

Typowe cechy kart kryptograficznych to m. in.

- Generowanie liczb losowych w oparciu o zjawisko fizyczne
- Generowanie kluczy RSA 1024 bity (lub 2048)
- Wykonywanie operacji podpisu z użyciem algorytmu RSA 1024 lub 2048 - **klucz prywatny NIGDY nie opuszcza karty**
- Kontrolowanie dostępu do obiektów prywatnych (klucze prywatne, dowolne dane poufne) z mechanizmami blokowania
- Otwarte interfejsy programistyczne: PKCS#11, MS CSP, JCE lub interfejsy niskiego poziomu

X.509

Urząd certyfikacji

Urząd certyfikacji (Centrum certyfikacji, ang. Certificate Authority) to w wielkim skrócie instytucja, która wydaje elektroniczne certyfikaty tożsamości dla innych osób czy instytucji. Tak zwana zaufana trzecia strona.

X.509

X.509 to standard opisujący sposób użycia asymetrycznych algorytmów kryptograficznych w celu składania podpisu elektronicznego oraz jego weryfikacji.

Trochę propagandy

Trochę propagandy ze strony Sigillum o zaletach podpisu elektronicznego

- Możliwość podpisywania danych w formie komputerowej - plików, poczty elektronicznej, urządzeń informatycznych (np. serwera)
- O wiele większa trudność podrobienia w porównaniu do podpisu odręcznego
- Większa precyzja stosowania
- Łatwiejsza weryfikacja zmian w podpisywanych dokumentach

To może coś podpiszemy

Mała prezentacja podpisywania dokumentu.

Standardy podpisów

W Polsce dopuszczalne są 3 standardowe formaty podpisów

1. CMS (Cryptographic Message Syntax)
2. Format oparty o XML
3. PKCS#7 - poprzednik CMS

Problem

One nie są między sobą kompatybilne. (Tylko plik w CMS jest poprawnym plikiem PKCS#7).

Co tak naprawdę podpisujemy?

Co podpisujemy na przykładzie PKCS#7 i CMS

Można myśleć o tym mniej więcej tak

1. Liczymy hash na pliku (np. MD5, SHA-1 itp.)

Co tak naprawdę podpisujemy?

Co podpisujemy na przykładzie PKCS#7 i CMS

Można myśleć o tym mniej więcej tak

1. Liczymy hash na pliku (np. MD5, SHA-1 itp.)
2. Stosujemy algorytm szyfrujący (np. RSA) na tym hashu

Co tak naprawdę podpisujemy?

Co podpisujemy na przykładzie PKCS#7 i CMS

Można myśleć o tym mniej więcej tak

1. Liczymy hash na pliku (np. MD5, SHA-1 itp.)
2. Stosujemy algorytm szyfrujący (np. RSA) na tym hashu
3. Doklejamy wymagane informacje o podpisującym i zaszyfrowany hash

Co tak naprawdę podpisujemy?

Co podpisujemy na przykładzie PKCS#7 i CMS

Można myśleć o tym mniej więcej tak

1. Liczymy hash na pliku (np. MD5, SHA-1 itp.)
2. Stosujemy algorytm szyfrujący (np. RSA) na tym hashu
3. Doklejamy wymagane informacje o podpisującym i zaszyfrowany hash

To nie jest do końca prawda.

Co naprawdę podpisujemy?

Nie będziemy wchodzić bardzo w szczegóły, ale idea jest taka

- Tak naprawdę standardy CMS i PKCS#7 definiują tabelkę "signed attributes", czyli atrybuty, które możemy sami definiować i które trzeba podpisać

Co naprawdę podpisujemy?

Nie będziemy wchodzić bardzo w szczegóły, ale idea jest taka

- Tak naprawdę standardy CMS i PKCS#7 definiują tabelkę "signed attributes", czyli atrybuty, które możemy sami definiować i które trzeba podpisać
- Oczywiście podpisów w pliku może być kilka i te tabelki są różne dla różnych podpisujących

Co naprawdę podpisujemy?

Nie będziemy wchodzić bardzo w szczegóły, ale idea jest taka

- Tak naprawdę standardy CMS i PKCS#7 definiują tabelkę "signed attributes", czyli atrybuty, które możemy sami definiować i które trzeba podpisać
- Oczywiście podpisów w pliku może być kilka i te tabelki są różne dla różnych podpisujących
- W PKCS#7 ona może być pusta i wtedy rzeczywiście podpisujemy sam hash
- W CMS tam muszą być dodatkowo np. znaczniki czasowe

Co naprawdę podpisujemy?

Nie będziemy wchodzić bardzo w szczegóły, ale idea jest taka

- Tak naprawdę standardy CMS i PKCS#7 definiują tabelkę "signed attributes", czyli atrybuty, które możemy sami definiować i które trzeba podpisać
- Oczywiście podpisów w pliku może być kilka i te tabelki są różne dla różnych podpisujących
- W PKCS#7 ona może być pusta i wtedy rzeczywiście podpisujemy sam hash
- W CMS tam muszą być dodatkowo np. znaczniki czasowe
- Jeżeli tabelka występuje (czyli w CMS zawsze, w PKCS#7 niekoniecznie), to musi w niej pojawić się obowiązkowo hash z pliku

Co naprawdę podpisujemy?

- Mogą pojawiać się dodatkowe inne atrybuty

Co naprawdę podpisujemy?

- Mogą pojawiać się dodatkowe inne atrybuty
- Standard mówi jak zakodować taką tabelkę

Co naprawdę podpisujemy?

- Mogą pojawiać się dodatkowe inne atrybuty
- Standard mówi jak zakodować taką tabelkę
- Liczymy hash na zakodowanej tabelce i dopiero ten hash podpisujemy (czyli hash pliku jest tu jeszcze raz hashowany i to może być nawet różnymi algorytmami!)

Co naprawdę podpisujemy?

- Mogą pojawiać się dodatkowe inne atrybuty
- Standard mówi jak zakodować taką tabelkę
- Liczymy hash na zakodowanej tabelce i dopiero ten hash podpisujemy (czyli hash pliku jest tu jeszcze raz hashowany i to może być nawet różnymi algorytmami!)
- Następnie szyfrujemy hash całej tabelki i to doklejamy wraz z wymaganymi informacjami do pliku (np. jakie algorytmy zostały użyte do hashowania i szyfrowania, certyfikaty itp.)

Różnice w standardach

- Inne kodowanie
  - CMS i PKCS#7 są plikami binarnymi, a XML jest plikiem tekstowym
- Przechowywane informacje
  - Formaty XML i CMS są mniej więcej równoważne, PKCS#7 jest uboższy (np. nie musi przechowywać informacji o znacznikach czasowych)

Kto w Polsce wydaje certyfikaty?

W Polsce uprawnione do wydawania podpisów są 3 Kwalifikowane Centra Certyfikacji (ang. QCA)

1. Sigillum
2. KIR
3. Unizeto

Były 4 ale Signet (uruchomione przez TP Internet) zawiesił działalność.

Jak być powinno w idealnym świecie?

W świecie idealnym (który na pewno gdzieś istnieje) powinno być mniej więcej tak

1. Księgowa firmy A kupuje certyfikat kwalifikowany w jednej z czterech (już trzech) uprawnionych do tego firm

Jak być powinno w idealnym świecie?

W świecie idealnym (który na pewno gdzieś istnieje) powinno być mniej więcej tak

1. Księgowa firmy A kupuje certyfikat kwalifikowany w jednej z czterech (już trzech) uprawnionych do tego firm
2. Instaluje aplikację

Jak być powinno w idealnym świecie?

W świecie idealnym (który na pewno gdzieś istnieje) powinno być mniej więcej tak

1. Księgowa firmy A kupuje certyfikat kwalifikowany w jednej z czterech (już trzech) uprawnionych do tego firm
2. Instaluje aplikację
3. Podpisuje i wysyła e-faktury z bezpiecznym podpisem cyfrowym

Ciąg dalszy świata idealnego

1. Kontrahent z firmy B nie musi kupować certyfikatu w tej samej firmie (nie powinien musieć kupować go wcale!)

Ciąg dalszy świata idealnego

1. Kontrahent z firmy B nie musi kupować certyfikatu w tej samej firmie (nie powinien musieć kupować go wcale!)
2. Może jedną aplikacją sprawdzać dowolne podpisy

Ciąg dalszy świata idealnego

1. Kontrahent z firmy B nie musi kupować certyfikatu w tej samej firmie (nie powinien musieć kupować go wcale!)
2. Może jedną aplikacją sprawdzać dowolne podpisy
3. Gdy chce sam podpisywać, może kupić sobie certyfikat w innej firmie i przy jego pomocy podpisywać faktury i wysyłać je do A

Wszędzie niby są standardy, więc gdzie jest problem? O tym za chwilę.

A tu zaczyna się piekło :)

- Każde polskie centrum dostarcza własne oprogramowanie do składania podpisu (niedarmowe, dostaje się je przy zakupie podpisu) i darmową aplikację do jego weryfikacji
- Okazuje się że każda firm co prawda formalnie zastosowała się do rozporządzenia o warunkach technicznych ale... każda wybrała sobie inny format...

I co ciekawsze do niedawna było tak

Firmy były 4 a standardy 3, więc (z Dirichleta :)) chociaż któreś teoretycznie powinny się zgadzać, ale ...

- Certum - CMS
- KIR - PKCS#7
- Signet - XML

A Sigillum do niedawna np. przy podpisywaniu plików Worda integrowało się z aplikacją i składało podpis w prywatnym formacie SDOC, który w praktyce był archiwum CAB z podpisanymi kolejnymi stronami dokumentu.

A weryfikacja?

Do niedawna było tak (początek 2008)

Każda aplikacja weryfikująca weryfikowała poprawnie tylko "swoje" podpisy, a podpisów konkurencji już nie.

I powstają absurdalne układy

Rozważmy więc taki scenariusz

- Firma A kupuje certyfikat w Certum (i używa ich programu proCertum Signer)

I powstają absurdalne układy

Rozważmy więc taki scenariusz

- Firma A kupuje certyfikat w Certum (i używa ich programu proCertum Signer)
- Wysyła faktury do firmy B

I powstają absurdalne układy

Rozważmy więc taki scenariusz

- Firma A kupuje certyfikat w Certum (i używa ich programu proCertum Signer)
- Wysyła faktury do firmy B
- Firma B musi zainstalować proCertum CombiLite żeby je weryfikować

I powstają absurdalne układy

Rozważmy więc taki scenariusz

- Firma A kupuje certyfikat w Certum (i używa ich programu proCertum Signer)
- Wysyła faktury do firmy B
- Firma B musi zainstalować proCertum CombiLite żeby je weryfikować
- Ale firma B kupiła certyfikat w KIR

I powstają absurdalne układy

Rozważmy więc taki scenariusz

- Firma A kupuje certyfikat w Certum (i używa ich programu proCertum Signer)
- Wysyła faktury do firmy B
- Firma B musi zainstalować proCertum CombiLite żeby je weryfikować
- Ale firma B kupiła certyfikat w KIR
- Firma A musi więc instalować inne oprogramowanie do weryfikacji

I powstają absurdalne układy

Rozważmy więc taki scenariusz

- Firma A kupuje certyfikat w Certum (i używa ich programu proCertum Signer)
- Wysyła faktury do firmy B
- Firma B musi zainstalować proCertum CombiLite żeby je weryfikować
- Ale firma B kupiła certyfikat w KIR
- Firma A musi więc instalować inne oprogramowanie do weryfikacji

I tak powstaje absurdalny, asymetryczny układ w którym każda z firm będzie podpisy wystawiać jednym programem, a weryfikować drugim.

I powstają absurdalne układy

- I żeby było jeszcze ciekawiej i wygodniej obie aplikacje zawłaszczają sobie pliki z rozszerzeniem .sig.

Teraz jest już trochę lepiej

- Sigillum podpisuje "normalnie" w formacie PKCS#7 (a przynajmniej umie już tak podpisywać)
- Aplikacje weryfikujące są między sobą coraz bardziej kompatybilne i dodawana jest obsługa standardów z innych centrów
- Sigillum jest obecnie w stanie weryfikować wszystko

Ale jeszcze nie jest idealnie

Takie osobiste uwagi, co by się jeszcze przydało

- Póki co wszystko działa tylko pod Windows - z Linuxem problem jest już nawet na poziomie bibliotek (i ich braku)

Ale jeszcze nie jest idealnie

Takie osobiste uwagi, co by się jeszcze przydało

- Póki co wszystko działa tylko pod Windows - z Linuxem problem jest już nawet na poziomie bibliotek (i ich braku)
- Aplikacja Sigillum jest w stanie zweryfikować wszystko, ale weryfikuje poprawnie tylko ścieżki kończące się w systemowym magazynie certyfikatów Zaufanych Centrów Certyfikacji, a tam nie można dodawać czegokolwiek (np. własnych certyfikatów self-signed)

Ale jeszcze nie jest idealnie

Takie osobiste uwagi, co by się jeszcze przydało

- Póki co wszystko działa tylko pod Windows - z Linuxem problem jest już nawet na poziomie bibliotek (i ich braku)
- Aplikacja Sigillum jest w stanie zweryfikować wszystko, ale weryfikuje poprawnie tylko ścieżki kończące się w systemowym magazynie certyfikatów Zaufanych Centrów Certyfikacji, a tam nie można dodawać czegokolwiek (np. własnych certyfikatów self-signed)
- Aplikacja Certum pozwala wybrać magazyn certyfikatów, ale za to nie weryfikuje wszystkiego, a tylko swój format

Co np. robią banki?

- Podpis to nie jest bardzo tania sprawa (koszt to kilkaset złotych, co prawda w tym jest już wszystko: oprogramowanie, czytnik, certyfikat itp.)
- Koszt samego nośnika to co najwyżej kilkanaście złotych, czytnika kilkadziesiąt
- Od strony technicznej można więc oszczędzać

Co robi np. PEKAO SA?

Co robi PEKAO SA?

- Generuje własny certyfikat

Co robi np. PEKAO SA?

Co robi PEKAO SA?

- Generuje własny certyfikat
- Kupuje karty, czyli nośniki certyfikatów dla klientów

Co robi np. PEKAO SA?

Co robi PEKAO SA?

- Generuje własny certyfikat
- Kupuje karty, czyli nośniki certyfikatów dla klientów
- Generuje dla klienta certyfikat podpisany swoim certyfikatem

Co robi np. PEKAO SA?

Co robi PEKAO SA?

- Generuje własny certyfikat
- Kupuje karty, czyli nośniki certyfikatów dla klientów
- Generuje dla klienta certyfikat podpisany swoim certyfikatem
- Wrzuca certyfikat na kartę i daje ją klientowi

Co robi np. PEKAO SA?

Co robi PEKAO SA?

- Generuje własny certyfikat
- Kupuje karty, czyli nośniki certyfikatów dla klientów
- Generuje dla klienta certyfikat podpisany swoim certyfikatem
- Wrzuca certyfikat na kartę i daje ją klientowi

Uwaga

Oczywiście w świetle prawa dokument podpisany takim certyfikatem nie ma jeszcze żadnej mocy! To nie jest certyfikat kwalifikowany, bo nie wydała go właściwa instytucja.

Co dalej?

- Bank podpisuje wtedy oddzielną umowę z klientem, że obie strony honorują te konkretne podpisy i uważają, że pliki podpisane tym certyfikatem mają dokładnie taką moc jak podpisane odręcznie

Co dalej?

- Bank podpisuje wtedy oddzielną umowę z klientem, że obie strony honorują te konkretne podpisy i uważają, że pliki podpisane tym certyfikatem mają dokładnie taką moc jak podpisane odręcznie
- Klient nie musi więc wydawać kilkuset złotych, a tylko kilkanaście i dostaje tę samą funkcjonalność

Co dalej?

- Bank podpisuje wtedy oddzielną umowę z klientem, że obie strony honorują te konkretne podpisy i uważają, że pliki podpisane tym certyfikatem mają dokładnie taką moc jak podpisane odręcznie
- Klient nie musi więc wydawać kilkuset złotych, a tylko kilkanaście i dostaje tę samą funkcjonalność
- Dodatkowo jest bardziej przywiązany takim "gadżetem" do danego banku

Co dalej?

- Bank podpisuje wtedy oddzielną umowę z klientem, że obie strony honorują te konkretne podpisy i uważają, że pliki podpisane tym certyfikatem mają dokładnie taką moc jak podpisane odręcznie
- Klient nie musi więc wydawać kilkuset złotych, a tylko kilkanaście i dostaje tę samą funkcjonalność
- Dodatkowo jest bardziej przywiązany takim "gadżetem" do danego banku
- Bank może dalej wykorzystywać też taki certyfikat np. do autoryzacji na stronie (np. zamiast loginu i hasła)

To można też zrobić na Uniwersytecie

- Nowe legitymacje studenckie mogą być nośnikiem certyfikatu i działać dokładnie jak karta kryptograficzna
- Wystarczyłoby więc wygenerować cerytfikat root dla UW i wrzucać studentom na ich legitymacje podpisane nim certyfikaty
- Dodać odpowiedni wpis w regulaminie studiów
- I student może zapomnieć o sekcji a podpisane dokumenty wysyłać mailem lub przez stronę www

W tym rozwiązaniu problemem jest jeszcze np. programowe wrzucanie certyfikatu na kartę. Nie ma standardowego API do tej operacji - trzeba przez aplikację producenta kart.

A jak się programuje takie karty? Co ze zgodnością?

Istnieje standard do komunikacji z inteligentnymi kartami (ang. smart cards), czyli PKCS#11.

PKCS#11

- Przyjęty standard do komunikacji z kartą
- Zdefiniowany w oficjalnym dokumencie RSA Laboratories
- Każdy szanujący się obecnie producent dostarcza biblioteki w tym standardzie

PKCS#11

- Dostępne oficjalne pliki nagłówkowe do języka C / C++ z deklaracjami
- Wszystkie polskie centra certyfikacji dostarczają takie biblioteki w postaci .dll (dramatyczny brak .so dla Linuxa)
- Można więc pisać "ogólne" aplikacje do podpisywania czy weryfikacji

Przykłady z API PKCS#11

Żeby dać lepsze wyobrażenie, to parę przykładów (i to byłyby pewnie potrzebne metody przy podpisywaniu czegoś w naszej "ogólnej" aplikacji)

- C_OpenSession
- C_CloseSession
- C_GetSessionInfo
- C_Login
- C_SignInit
- C_Sign
- C_VerifyInit
- C_Verify

Subtelności

- Wciąż są pewne subtelności, które pojawiają się przy pisaniu takich "ogólnych" aplikacji
- Pojawia się np. problem z PIN-em, gdy chcemy podpisać więcej niż jeden plik - teoretycznie w PKCS#11 zdefiniowano możliwość podpięcia własnego okienka do pytania o PIN a nawet podawania go "programowo", ale część bibliotek to ignoruje i i tak wyświetla własne okienko
- Co więcej czasami za każdym razem, gdy chcemy coś podpisać to okienko się pojawia, czyli 10 plików = 10 okienek do wpisania PIN-u (w sumie w świetle prawa to chyba nawet tak powinno być)

W czym pisać?

Biblioteki to .dll, ale Java jest bardzo dobrze przygotowana do ich obsługi. To jest dobra wiadomość, gdy chcemy np. pisać aplet.

- Jest już wiele gotowych wrapperów i providerów: istnieją biblioteki IAIK, Sun, BouncyCastle...
- Wtedy po prostu wywołujemy metody a cała magia dzieje się gdzieś pod spodem
- Ale to też początek drogi do napisania takiej aplikacji
- Sam wygenerowany podpis trzeba potem jeszcze odpowiednio poskładać - tutaj generujemy tylko jego cząstkę - pewnie najważniejszą, ale jednak cząstkę :)

Mały przerywnik i szansa na wafelka

Ustawa o podpisie cyfrowym

- Kiedy (wystarczy podać rok) weszła w życie w Polsce ustawa o podpisie elektronicznym?

Mały przerywnik i szansa na wafelka

Ustawa o podpisie cyfrowym

- Kiedy (wystarczy podać rok) weszła w życie w Polsce ustawa o podpisie elektronicznym?
- Odpowiedź: 18 września 2001r.

Jest raczej źle - 2007

Rok 2007

- W Polsce sprzedano dotychczas od 20 do 30 tysięcy podpisów

Jest raczej źle - 2007

Rok 2007

- W Polsce sprzedano dotychczas od 20 do 30 tysięcy podpisów
- Ich wykorzystanie w praktyce jest nikłe, a głównie dlatego że nie ma obowiązku ich stosowania i większość urzędów wciąż jest nieprzygotowana

Jest trochę lepiej - 2008

Rok 2008

- Ożywienie w 2008 roku, gdy Zakład Ubezpieczeń Społecznych zaczął akceptować kwalifikowany e-podpis jako formę uwierzytelnienia deklaracji rozliczeniowych przedsiębiorstw

Jest trochę lepiej - 2008

Rok 2008

- Ożywienie w 2008 roku, gdy Zakład Ubezpieczeń Społecznych zaczął akceptować kwalifikowany e-podpis jako formę uwierzytelnienia deklaracji rozliczeniowych przedsiębiorstw
- Od maja 2008 r. organy administracji są zobowiązane do przyjmowania wniosków i podań z kwalifikowanym e-podpisem (na równi z papierowymi), ale w praktyce różnie z tym bywa (często jest tak zwana "awaria komputera", która to uniemożliwia)

Jest trochę lepiej - 2008

Rok 2008

- Ożywienie w 2008 roku, gdy Zakład Ubezpieczeń Społecznych zaczął akceptować kwalifikowany e-podpis jako formę uwierzytelnienia deklaracji rozliczeniowych przedsiębiorstw
- Od maja 2008 r. organy administracji są zobowiązane do przyjmowania wniosków i podań z kwalifikowanym e-podpisem (na równi z papierowymi), ale w praktyce różnie z tym bywa (często jest tak zwana "awaria komputera", która to uniemożliwia)
- Jednak w 2008 nastąpił znaczny wzrost sprzedaży certyfikatów - w 2008 roku sprzedano ich podobno ok. 150 tysięcy

Jest trochę lepiej - 2008

Rok 2008

- Ożywienie w 2008 roku, gdy Zakład Ubezpieczeń Społecznych zaczął akceptować kwalifikowany e-podpis jako formę uwierzytelnienia deklaracji rozliczeniowych przedsiębiorstw
- Od maja 2008 r. organy administracji są zobowiązane do przyjmowania wniosków i podań z kwalifikowanym e-podpisem (na równi z papierowymi), ale w praktyce różnie z tym bywa (często jest tak zwana "awaria komputera", która to uniemożliwia)
- Jednak w 2008 nastąpił znaczny wzrost sprzedaży certyfikatów - w 2008 roku sprzedano ich podobno ok. 150 tysięcy
- Wciąż zwykły podatnik nie może złożyć np. PIT37 przez internet (ale niektóre już można)

Powinno być lepiej?

Szykowana jest nowelizacja ustawy o podpisie

- W planach jest już rozszerzenie definicji "podpisującego" o podmioty inne niż osoby fizyczne i wprowadzenie tzw. elektronicznej pieczęci, czyli podpisu generowanego automatycznie przez systemy informatyczne (aby np. potwierdzać wpłynięcie dokumentów elektronicznych)

Powinno być lepiej?

Szykowana jest nowelizacja ustawy o podpisie

- W planach jest już rozszerzenie definicji "podpisującego" o podmioty inne niż osoby fizyczne i wprowadzenie tzw. elektronicznej pieczęci, czyli podpisu generowanego automatycznie przez systemy informatyczne (aby np. potwierdzać wpłynięcie dokumentów elektronicznych)
- W planach jest "podpis urzędowy" zaszyty w dowodzie osobistym, ale tutaj jest dużo kontrowersji co do treści proponowanej ustawy i urzędów, które mogłyby wydawać takie podpisy

Powinno być lepiej?

Szykowana jest nowelizacja ustawy o podpisie

- W planach jest już rozszerzenie definicji "podpisującego" o podmioty inne niż osoby fizyczne i wprowadzenie tzw. elektronicznej pieczęci, czyli podpisu generowanego automatycznie przez systemy informatyczne (aby np. potwierdzać wpłynięcie dokumentów elektronicznych)
- W planach jest "podpis urzędowy" zaszyty w dowodzie osobistym, ale tutaj jest dużo kontrowersji co do treści proponowanej ustawy i urzędów, które mogłyby wydawać takie podpisy
- Mają zostać wprowadzone nowe typy podpisów (obecnie jest zwykły i kwalifikowany, przy czym tylko kwalifikowany ma skutki prawne) - tutaj jeszcze więcej kontrowersji

Weryfikacja

Można wyróżnić 2 główne fazy weryfikacji:

- Weryfikacja czy podpis zgadza się z certyfikatem, który dała nam dana osoba (czy dane się nie zmieniły)
- Weryfikacja czy certyfikat jest ważny - trzeba sprawdzić całą ścieżkę certyfikacji

Pierwsza faza jest dość techniczna (oczywiście musimy mieć pewność, że mamy właściwy certyfikat danej osoby) z dokładnością do różnych standardów podpisu, które być może trzeba obsłużyć.

Ścieżka certyfikacji

- W architekturze PKI (Public Key Infrastructure) certyfikaty są podpisywane przez inne certyfikaty (i każdy ma odpowiednią politykę, w której np. określa się, czy dany certyfikat może służyć do wystawiania innych certyfikatów)
- Trzeba zweryfikować całą ścieżkę - każdy certyfikat ma "wskazanie" na certyfikat "ojca" - każdy certyfikat po drodze musi być ważny
- Ostatni certyfikat tzw. root musi być certyfikatem, któremu ufamy

To nie byłoby jeszcze takie złe, ale przecież centra mogą unieważniać certyfikaty.

Listy CRL

- Lista CRL to lista odwołanych certyfikatów wydanych przez dane centrum (np. wskutek ujawnienia, a fachowo skompromitowania klucza prywatnego)
- W przypadku centrów kwalifikowanych publikowanie list CRL jest obowiązkowe

Problemy z listami CRL

- Adres listy CRL danego certyfikatu powinien być zapisany w certyfikacie
- Listy CRL stanowią jednak dużą przeszkodę w weryfikacji - trzeba mieć dostęp on-line do list CRL centrów i zawsze jest szansa na problemy ze "współbieżnością"
- I dochodzi kwestia upewnienia się, że dana lista jest rzeczywiście właściwą listą CRL
- Dochodzi też kwestia potencjalnego ataku typu DoS, jeżeli ktoś sprawi, że lista CRL jest niedostępna - niczego nie da się wtedy zweryfikować!
- Polskie programy sprawdzają po prostu wszystkie polskie listy (nie jest ich tak dużo, bo były tylko 4 centra)

Alternatywa to OCSP

OCSP = Online Certificate Status Protocol

- Internetowy protokół do sprawdzania ważności certyfikatu
- To nie jest poprawa jakościowa, a ułatwienie - zamiast listy CRL możemy dostać adres, pod którym podając numer certyfikatu otrzymamy odpowiedź: TAK / NIE / NIE WIADOMO (oczywiście odpowiednio podpisaną) w zależności czy jest on ważny czy nie czy nie wiadomo
- Zyskujemy znaczną oszczędność pasma - pytamy o konkretny certyfikat, a nie ściągamy całej listy
- Ujawniamy jednak informację, z kim "robimy interesy"
- Problemy pozostają podobne

Coś dla ludzi o mocnych nerwach :)

1. Czy w świetle polskiego prawa podpis złożony certyfikatem, który był ważny w momencie podpisywania, ale wygasł jest ważny?
2. Jak to można weryfikować skoro nie ma konieczności umieszczania znacznika czasu?
3. Czy lokalny znacznik czasu jest wystarczający?

Jak to jest w Polsce?

Praktyka jest taka, że jeśli podpis zawiera tylko atrybut lokalny (czyli łatwy do zmanipulowania) lub go nie ma, to weryfikuje się ważność certyfikatu według czasu bieżącego weryfikacji, tak jakby podpis został złożony przed chwilą.
Nie ma problemu, jeśli weryfikujemy jeszcze w okresie ważności certyfikatu. Jest problem, jeśli weryfikujemy po jego wygaśnięciu.

Można jednak podpis znakować czasem i polskie centra udostępniają takie serwery znaczników czasu, które gwarantują, że można zweryfikować czas podpisu, ale wtedy trzeba mieć dostęp do Internetu przy podpisywaniu.
Przy weryfikacji zawsze trzeba mieć dostęp do Internetu (listy CRL).

Jak to jest w Polsce?

Polska realizuje tutaj model zagnieżdżony, tzw. "shell model"

Jeśli np. jeden z certyfikatów CA wygaśnie przed czasem, to unieważnia to automatycznie wszystkie certyfikaty leżące "pod nim" i przez to wszystkie pliki nimi podpisane chyba że... udowodnione zostanie że zostały podpisane przed unieważnieniem.

W Niemczech jest trochę inaczej, tzw "chain model"

Do poprawnej weryfikacji ścieżki wystarczy, by poszczególne certyfikaty były ważne w momencie podpisywania następnego certyfikatu.

Praca magisterska

Jakie są obecnie problemy?

- Na Uniwersytecie Warszawskim około 400 pracowników ma certyfikat
- Nie jesteśmy niezależni od dostawcy certyfikatów (a nawet bardziej od jego oprogramowania) - certyfikaty są ważne max. 2 lata, ale potem koszt przejścia na innego dostawcę może być na tyle wysoki, że to się nigdy nie będzie opłacać
- Brak centralizacji - trudna sytuacja w momencie, gdy pani z sekcji idzie na urlop lub odchodzi z pracy

Co jest rozwiązaniem?

Centralizacja i serwer podpisów

Serwer podpisów

Kilka podstawowych założeń

1. Administrator może rejestrować certyfikaty użytkownikom
2. Do użytkownika z zarejestrowanym certyfikatem można wysłać prośbę o podpisanie danego pliku (takie prośby są kolejkowane)
3. Osoba z certyfikatem widzi swoją kolejkę plików, może ją przeglądać i podpisać lub nie wybrane pliki - tutaj musi być aplet (lub Java Web Start), bo karta musi być podłączona do komputera pani z sekcji
4. Dodatkowy zysk: do kolejki danej pani dokumenty może kierować np. USOS, elementu ludzkiego przy podpisywaniu nie da się wyeliminować (ktoś musi chociażby podawać PIN)

Jest dobrze, ale nie beznadziejnie :)

- Podpis cyfrowy daje ogromne możliwości jest wygodny w stosowaniu i może ułatwić wiele spraw

Jest dobrze, ale nie beznadziejnie :)

- Podpis cyfrowy daje ogromne możliwości jest wygodny w stosowaniu i może ułatwić wiele spraw
- W Polsce być może wciąż nie jest zbyt dobrze, ale temat przynajmniej nie jest martwy

Jest dobrze, ale nie beznadziejnie :)

- Podpis cyfrowy daje ogromne możliwości jest wygodny w stosowaniu i może ułatwić wiele spraw
- W Polsce być może wciąż nie jest zbyt dobrze, ale temat przynajmniej nie jest martwy
- I z obserwacji wydaje się, że zmierza ku lepszemu

Podziękowanie

Dziękuję za uwagę i chętnie wysłucham pytań.