System PEM

Sylwia Miśkiewicz

Privacy Enhanced Mail

- Standard **PEM** został opracowany w celu zwiększenia ochrony przesyłanych wiadomości tekstowych. Prace rozpoczęto w roku 1985, a zakończono w 1993 r. Raport RFC
- Zwiększa prywatność korespondencji w sieci Internet
- W PEM korzysta się zarówno z szyfrowania z kluczem jawnym, jak i tajnym.

PEM

- Obejmuje rozszerzenia istniejącego oprogramowania do przetwarzania komunikatów oraz infrastrukturę zarządzania kluczami.
- Łączą się one w celu zapewnienia użytkownikom możliwości poufności, autentyczności i integralności wiadomości.
- Chociaż RFC zezwalają na użycie kryptografii symetrycznej lub asymetrycznej (np. RSA) w celu dystrybucji kluczy symetrycznych, RFC zdecydowanie zalecają stosowanie w tym celu kryptografii asymetrycznej, generowanie i weryfikowanie podpisów cyfrowych dla wiadomości oraz certyfikaty.

PEM

- Zarządzanie kluczami publicznymi w PEM opiera się na wykorzystaniu certyfikatów zdefiniowanych przez CCITT Directory Authentication Framework [CCIT88c].
- Hierarchia certyfikacji klucza publicznego dla PEM jest tworzona przez Internet Society. Ta hierarchia obsługuje uniwersalne uwierzytelnianie użytkowników PEM, w ramach różnych polityk, bez potrzeby wcześniejszych umów dwustronnych między użytkownikami lub organizacjami, z którymi użytkownicy mogą być powiązani.
- **Internet Society** (ISOC) - międzynarodowe stowarzyszenie mające na celu dbanie o harmonijny rozwój Internetu oraz w pewnym sensie reprezentujące użytkowników Internetu przed rządowymi agencjami odpowiedzialnymi za nadzór nad Internetem w poszczególnych krajach.

PEM zapewnia

- **poufność** - uzyskano przez szyfrowanie (DES),
- **uwierzytelnienie źródła** - uzyskano przez zastosowanie podpisu cyfrowego (RSA-MD2, RSA-MD5),
- **integralność (spójność) wiadomości -** uzyskano przez zastosowanie podpisu cyfrowego (RSA-MD2, RSA-MD5),
- **niezaprzeczalność nadania**,
- **mechanizm zarządzania kluczami** (DES, RSA).

PEM jest opisany w raportach **RFC**:

- RFC 1421 (procedury szyfrowania i uwierzytelnienia),
- RFC 1422 (zarządzanie kluczami),
- RFC 1423 (algorytmy szyfrowania i sprawdzania spójności wiadomości),
- RFC 1424 (trzy typy usług wspierających PEM - poświadczanie kluczy, przechowywanie ***list unieważnień certyfikatów*** *CRL* (*certificate revocation list*), wyszukiwanie i odzyskiwanie list CRL.

Jak działa

- Użytkownicy poczty otrzymują parę: klucz jawny i prywatny od lokalnego programu PEM i publikują klucz jawny za pomocą własnych adresów pocztowych.
- Wysyłając komunikat, program PEM generuje jednorazowy klucz tajny i za jego pomocą szyfruje komunikat według algorytmu DES (symetryczny).
- Klucz tajny jest szyfrowany za pomocą jawnego klucza odbiorcy i dodawany do zaszyfrowanego komunikatu.
- PEM jest dostępny w postaci programu RIPEM z biblioteka RSAREF szyfrowania z kluczem jawnym produkcji RSA Data Security Inc. na licencji PKP.
- Licencje dotyczące oprogramowania RIPEM i RSAREF zezwalają na bezpłatne jego używanie w celach niekomercyjnych.

Klucz publiczny i klucz prywatny

- Klucz jawny (*public key*), jeden z dwu kluczy stosowanych w systemie szyfrowania z kluczem jawnym; rozpowszechniany w powiązaniu z kluczem tajnym.
- Klucz prywatny (*private key*) jest kluczem tajnym, powinien go znać tylko jego właściciel.

Etapy pracy PEM

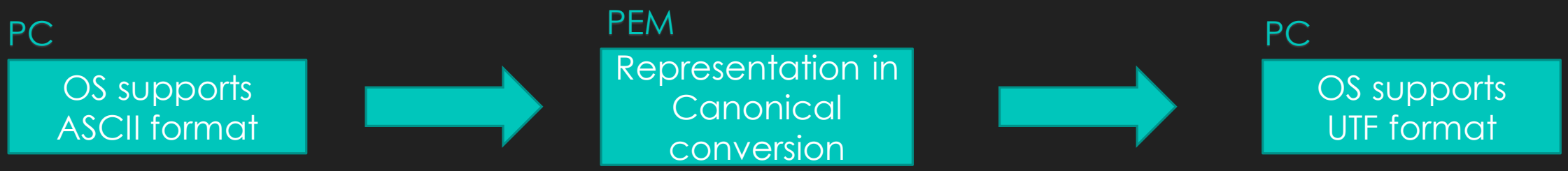
- Canonical conversion
- Digital signature
- Encryption
- Base-64 encoding

PEM pozwala na trzy opcje podczas wysyłania wiadomości

- Tylko podpis (etap 1 i 2)
- Podpis i kodowanie base-64 (etap 1, 2 i 4)
- Podpis, szyfrowanie i kodowanie base-64 (wszystkie etapy)

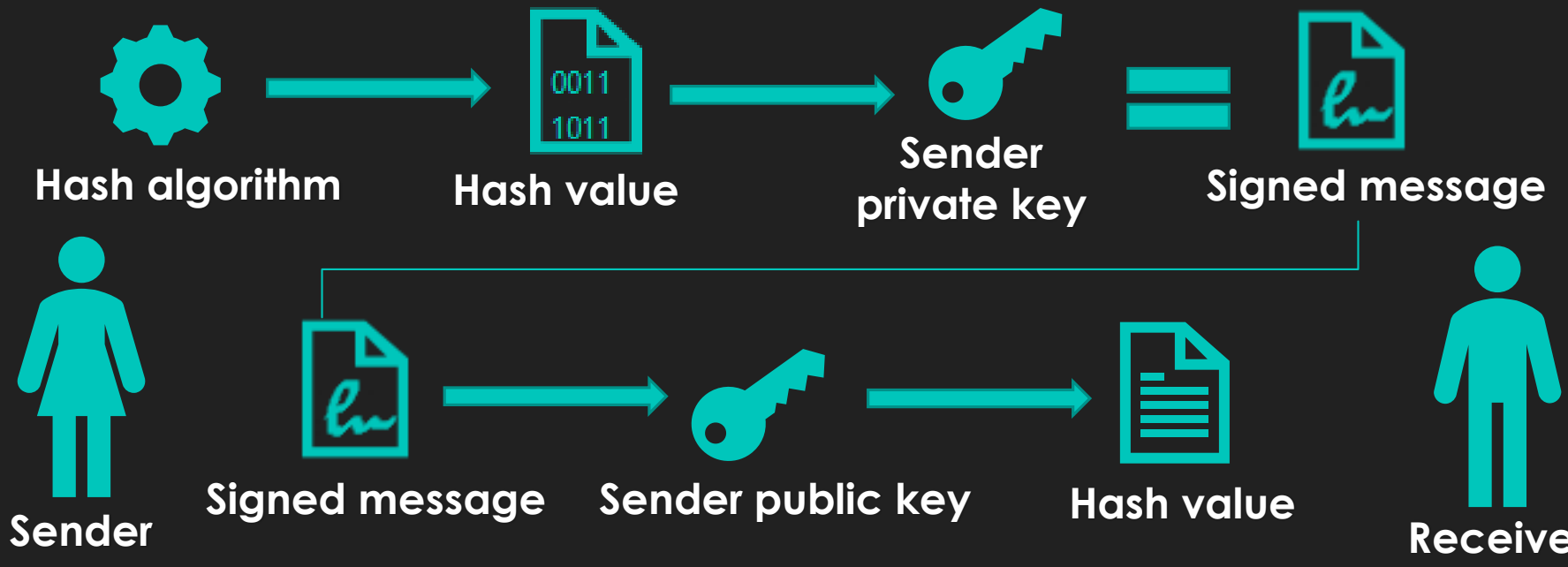
1. Canonical conversion

Istnieje możliwość, że nadawca i odbiorca wiadomości e-mail korzystają z komputerów, które mają różne architektury i systemy operacyjne. Dzieje się tak dlatego, że Internet działa na dowolnych komputerach ze stosem TCP / IP. Dlatego jest całkiem możliwe, że to samo jest reprezentowane inaczej na różnych komputerach. Może to powodować problemy podczas tworzenia treści wiadomości, a tym samym podpisu cyfrowego. Tak więc PEM przekształca każdą wiadomość e-mail w abstrakcyjną reprezentację kanoniczną. Oznacza to, że niezależnie od architektury i systemu operacyjnego komputerów wysyłających i otrzymujących, wiadomość e-mail zawsze przesyłana jest w otoczce, niezależnego formatu.



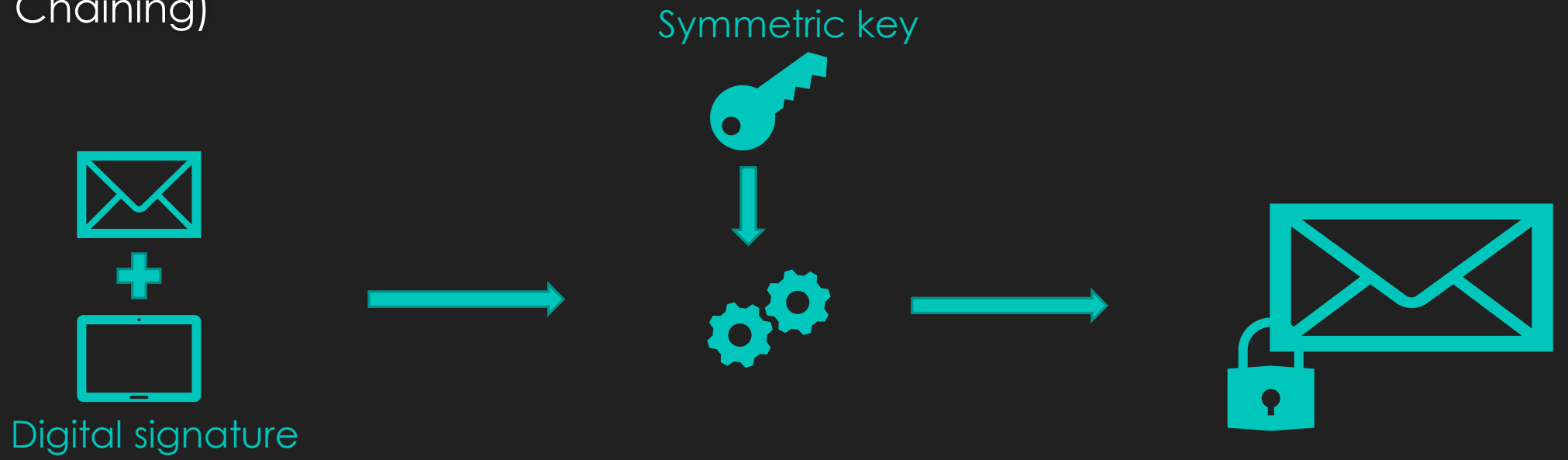
2. Digital signature

Podpis cyfrowy wykorzystuje znane algorytmy asymetryczne oraz funkcje skrótu. Opiera się na tajności klucza prywatnego.



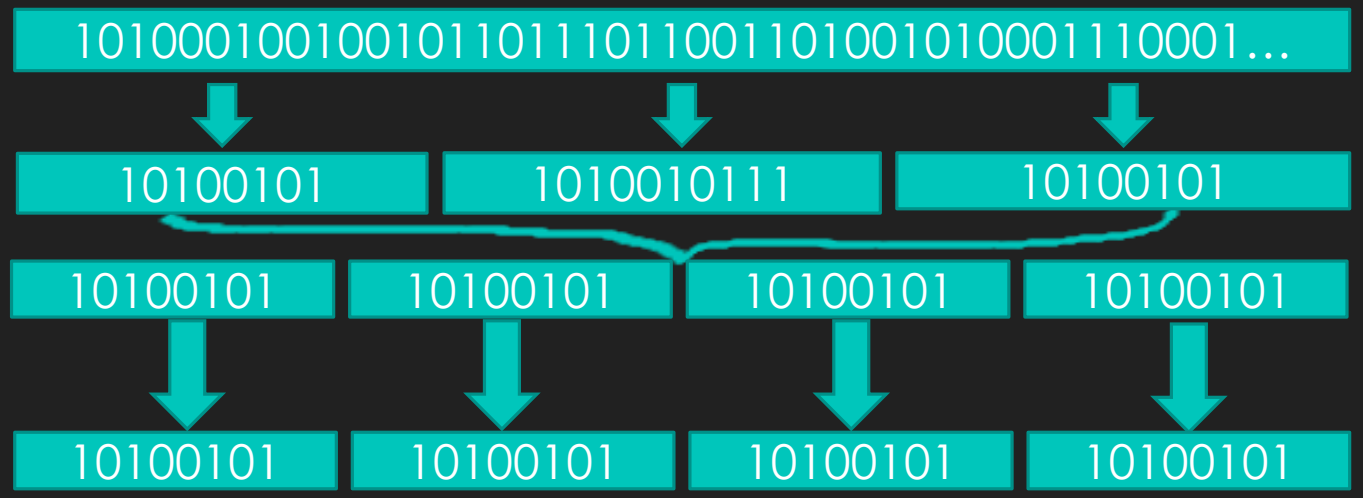
3. Encryption

Oryginalny e-mail i podpis cyfrowy są szyfrowane razem z kluczem symetrycznym. W tym celu wykorzystywane są algorytmy DES lub DES-3 w trybie CBC (Cipher Block Chaining)



4. Base-64 encoding

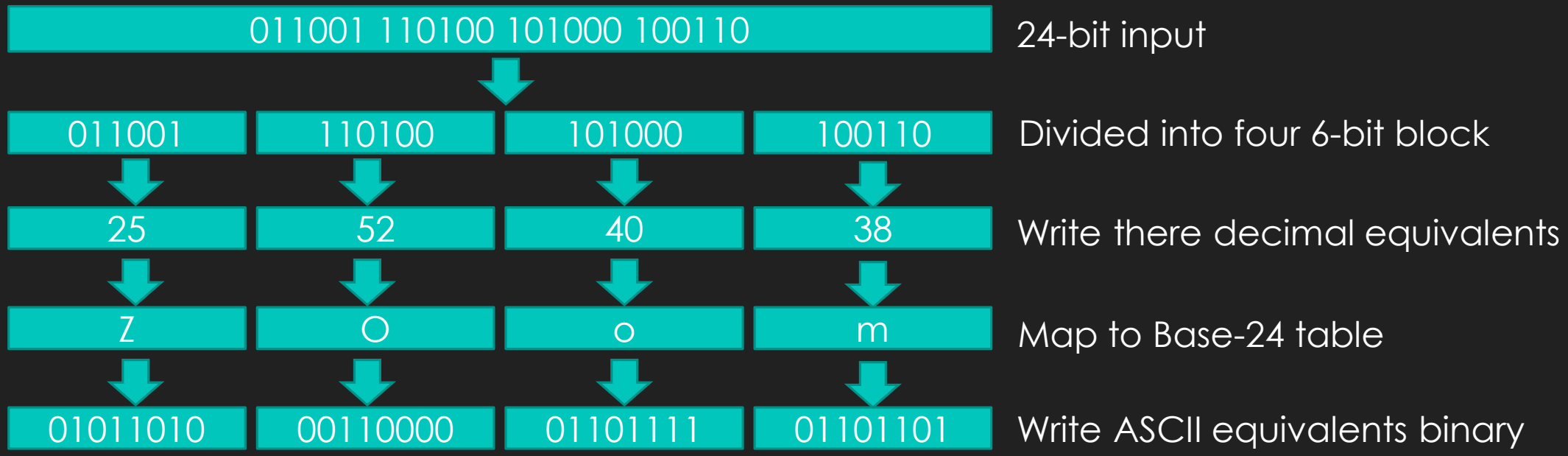
Kodowanie base-64 (zwane także kodowaniem Radix-64 lub ASCII armour) przetwarza dowolne dane wejściowe w drukowalny znak wyjściowe. W tej technice wejściowe dane binarne są w blokach o 3 oktetach lub 24 bitach. Te 24 bity są przygotowane do złożonia w 4 zestawy, każdy z 6 bitów. Każdy taki zestaw 6 bitów jest odwzorowany w 8-bitowy znak wyjściowy.



4. Base-64 encoding

Wydaje się to być dość prostym procesem. Jednak jedno kluczowe pytanie brzmi: jaka jest logika używana do mapowania 6-bitowego bloku wejściowego na wyjściowy 8-bitowy? W tym celu wykorzystuje się tablicę odwzorowania.

W tym przykładzie kodowania Base-64, rozważana jest 24-bitowa nieprzetworzoną strumień binarny 011001 110100 101000 100110



Binary	ASCII
000000	A
000001	B
000010	C
000011	D
000100	E
000101	F
000110	G
000111	H
001000	I
001001	J
001010	K
001011	L
001100	M
001101	N
001110	O
001111	P

Binary	ASCII
010000	Q
010001	R
010010	S
010011	T
010100	U
010101	V
010110	W
010111	X
011000	Y
011001	Z
011010	a
011011	b
011100	c
011101	d
011110	e
011111	f

Binary	ASCII
100000	g
100001	h
100010	i
100011	j
100100	k
100101	l
100110	m
100111	n
101000	o
101001	p
101010	q
101011	r
101100	s
101101	t
101110	u
101111	v

Binary	ASCII
110000	w
110001	x
110010	y
110011	z
110100	0
110101	1
110110	2
110111	3
111000	4
111001	5
111010	6
111011	7
111100	8
111101	9
111110	+
111111	/

Pretty Good Privacy (PGP)

Program ten został napisany w celu zapewnienia kompleksowej ochrony przesyłek pocztowych, i jest to de facto standard w tej dziedzinie. Zasługuje na szczególną uwagę dzięki swojej ogromnej funkcjonalności i popularności. Jest to program autorstwa Phila Zimmermanna dostępny zarówno na systemy operacyjne serii Windows jak i na systemy typu Unix. Zadaniem jego jest dostarczenie użytkownikowi jak najwięcej użytecznych usług związanych z szyfrowaniem. Część z tych usług jest związana bezpośrednio z ochroną poczty elektronicznej.

Ponieważ program ten z czasem podlegał komercjalizacji, więc opracowany został program, o nazwie ***GNU Privacy Guard*** (GPG), oferujący podstawową ochronę poczty

Źródła

Kurs autorstwa Zbigniewa Suski: link

Słownik Encyklopedyczny – Informatyka: link

Semantic Scholar: link

Prohackers.in: link

Kurs Devendra Ahirwar: link