

MARCIN HARBOWSKI

PODSTAWY KRYPTOGRAFII

WYDANIE III

Przekonaj się, jak fascynująca jest kryptografia!

- _ Poznaj historię rozwoju kryptografii
- _ Opanuj jej matematyczne podstawy
- _ Rozpracuj najważniejsze algorytmy kryptograficzne
- _ Dowiedz się, jak zastosować je w praktyce

Helion

Wszelkie prawa zastrzeżone. Nieautoryzowane rozpowszechnianie całości lub fragmentu niniejszej publikacji w jakiegokolwiek postaci jest zabronione. Wykonywanie kopii metodą kserograficzną, fotograficzną, a także kopiowanie książki na nośniku filmowym, magnetycznym lub innym powoduje naruszenie praw autorskich niniejszej publikacji.

Wszystkie znaki występujące w tekście są zastrzeżonymi znakami firmowymi bądź towarowymi ich właścicieli.

Autor oraz wydawca dołożyli wszelkich starań, by zawarte w tej książce informacje były kompletne i rzetelne. Nie biorą jednak żadnej odpowiedzialności ani za ich wykorzystanie, ani za związane z tym ewentualne naruszenie praw patentowych lub autorskich. Autor oraz wydawca nie ponoszą również żadnej odpowiedzialności za ewentualne szkody wynikłe z wykorzystania informacji zawartych w książce.

Redaktor prowadzący: Michał Mrowiec

Projekt okładki: Jan Paluch

Fotografia na okładce została wykorzystana za zgodą Shutterstock.com

Helion S.A.

ul. Kościuszki 1c, 44-100 Gliwice

tel. 32 231 22 19, 32 230 98 63

e-mail: helion@helion.pl

WWW: <http://helion.pl> (księgarnia internetowa, katalog książek)

Drogi Czytelniku!

Jeżeli chcesz ocenić tę książkę, zajrzyj pod adres

<http://helion.pl/user/opinie/pokr3v>

Możesz tam wpisać swoje uwagi, spostrzeżenia, recenzję.

ISBN: 978-83-283-8569-6

Copyright © Helion S.A. 2014, 2021

Printed in Poland.

- [Kup książkę](#)
- [Poleć książkę](#)
- [Oceń książkę](#)

- [Księgarnia internetowa](#)
- [Lubię to! » Nasza społeczność](#)

Spis treści

Kilka słów wstępu	11
Rozdział 1. Historia kryptografii	15
1.1. Prolog — Painvin ratuje Francję	15
1.2. Początek... ..	19
1.2.1. Steganografia	19
1.2.2. Kryptografia	20
1.2.3. Narodziny kryptoanalizy	22
1.3. Rozwój kryptografii i kryptoanalizy	23
1.3.1. Szyfry homofoniczne	23
1.3.2. Szyfry polialfabetyczne	24
1.3.3. Szyfry digraficzne	29
1.3.4. Prawdziwy „szyfr nie do złamania”	30
1.3.5. Kamienie milowe kryptografii	32
1.4. Kryptografia II wojny światowej	33
1.4.1. Enigma i Colossus	33
1.5. Era komputerów	38
1.5.1. DES	39
1.5.2. Narodziny kryptografii asymetrycznej	40
1.5.3. RSA	41
1.5.4. PGP	42
1.5.5. Ujawniona tajemnica	43
1.5.6. Upowszechnienie kryptografii	44
Rozdział 2. Matematyczne podstawy kryptografii	47
2.1. Podstawowe pojęcia	48
2.1.1. Słownik tekstu jawnego	48
2.1.2. Przestrzeń tekstu	48
2.1.3. Iloczyn kartezjański	49
2.1.4. System kryptograficzny	50
2.1.5. Szyfrowanie monoalfabetyczne	51
2.1.6. Funkcje jednokierunkowe	51
2.1.7. Arytmetyka modulo	52
2.1.8. Dwójkowy system liczbowy	53
2.1.9. Liczby pierwsze	54
2.1.10. Logarytmy	59
2.1.11. Grupy, pierścienie i ciała	60
2.1.12. Izomorfizmy	61

2.2. Wzory w praktyce	63
2.2.1. Kryptosystem RSA	63
2.2.2. Problem faktoryzacji dużych liczb	65
2.2.3. Mocne liczby pierwsze	67
2.2.4. Generowanie liczb pierwszych	67
2.2.5. Chińskie twierdzenie o resztach	70
2.2.6. Logarytm dyskretny	70
2.2.7. XOR i AND	72
2.2.8. Testy zgodności	73
2.2.9. Złożoność algorytmów	82
2.2.10. Teoria informacji	83
Rozdział 3. Kryptografia w teorii	89
3.1. Ataki kryptoanalityczne i nie tylko	89
3.1.1. Metody kryptoanalityczne	89
3.1.2. Kryptoanaliza liniowa i różnicowa	91
3.1.3. Inne rodzaje ataków	92
3.2. Rodzaje i tryby szyfrowania	98
3.2.1. Szyfry blokowe	98
3.2.2. Szyfry strumieniowe	107
3.2.3. Szyfr blokowy czy strumieniowy?	112
3.3. Protokoły kryptograficzne	113
3.3.1. Protokoły wymiany kluczy	113
3.3.2. Podpis cyfrowy	117
3.3.3. Dzielenie sekretów	120
3.3.4. Inne protokoły	123
3.4. Infrastruktura klucza publicznego	126
3.4.1. PKI w teorii... ..	127
3.4.2. ...i w praktyce	127
3.5. Kryptografia alternatywna	130
3.5.1. Fizyka kwantowa w kryptografii	130
3.5.2. Kryptografia DNA	137
3.5.3. Kryptografia wizualna	142
3.6. Współczesna steganografia	144
3.6.1. Znaki wodne	144
3.6.2. Oprogramowanie steganograficzne	145
Rozdział 4. Kryptografia w praktyce	147
4.1. Konstrukcja bezpiecznego systemu kryptograficznego	147
4.1.1. Wybór i implementacja kryptosystemu	148
4.1.2. Bezpieczny system kryptograficzny	149
4.1.3. Najslabsze ogniwo	150
4.2. Zabezpieczanie połączeń internetowych	154
4.2.1. Protokół TLS	154
4.2.2. Protokół SSH	162
4.3. Symantec Encryption Desktop	169
4.3.1. PGP Keys	173
4.3.2. PGP Messaging	177
4.3.3. PGP Zip	181
4.3.4. PGP Disk	185
4.3.5. PGP Viewer	194
4.3.6. File Share Encryption	196
4.3.7. PGP Shredder	198
4.3.8. Web of Trust	199

4.4. GnuPG	201
4.4.1. Tworzenie certyfikatu	201
4.4.2. Obsługa certyfikatów	203
4.4.3. Szyfrowanie i podpisywanie	205
4.4.4. Obsługa serwerów	209
4.5. TrueCrypt	210
4.5.1. Tworzenie szyfrowanych dysków i partycji	210
4.5.2. Obsługa dysków wirtualnych	213
4.5.3. Ukryte dyski	213
4.5.4. Pozostałe opcje i polecenia	215
4.6. Składanie i weryfikacja podpisów elektronicznych	218
4.6.1. Wymagania techniczne	218
4.6.2. Jak zdobyć certyfikat cyfrowy?	219
4.6.3. O czym warto pamiętać?	221
4.6.4. Konfiguracja programu pocztowego	222
4.6.5. Struktura certyfikatu	226
4.7. Kryptografia w PHP i MySQL	229
4.7.1. Funkcje szyfrujące w PHP	229
4.7.2. Szyfrowanie danych w MySQL	234
4.7.3. Kolejne udoskonalenia	238
Podsumowanie	241
Dodatek A Jednokierunkowe funkcje skrótu	243
A.1. MD5	243
A.1.1. Przekształcenia początkowe	243
A.1.2. Pętla główna MD5	244
A.1.3. Obliczenia końcowe	246
A.2. SHA-1	246
A.2.1. Przekształcenia początkowe	246
A.2.2. Pętla główna algorytmu SHA-1	247
A.2.3. Operacje w cyklu SHA-1	247
A.2.4. Obliczenia końcowe	248
A.3. SHA-2	249
A.3.1. Dodatkowe pojęcia	249
A.3.2. Przekształcenia początkowe	250
A.3.3. Operacje w cyklu SHA-2	251
A.3.4. Dodatkowe różnice między algorytmami SHA-2	253
A.4. SHA-3	254
A.4.1. SHA-3 — ogólny opis	254
A.4.2. Funkcja rundy SHA-3	254
A.4.3. Funkcja mieszająca SHA-3	256
A.5. Inne funkcje skrótu	257
Dodatek B Algorytmy szyfrujące	259
B.1. IDEA	259
B.1.1. Przekształcenia początkowe	259
B.1.2. Operacje pojedynczego cyklu IDEA	259
B.1.3. Generowanie podkluczy	261
B.1.4. Przekształcenia MA	261
B.1.5. Deszyfrowanie IDEA	261
B.2. DES	263
B.2.1. Permutacja początkowa (IP)	263
B.2.2. Podział tekstu na bloki	263

B.2.3. Permutacja rozszerzona	265
B.2.4. S-bloki	266
B.2.5. P-bloki	267
B.2.6. Permutacja końcowa	268
B.2.7. Deszyfrowanie DES	268
B.2.8. Modyfikacje DES	269
B.3. AES	271
B.3.1. Opis algorytmu	271
B.3.2. Generowanie kluczy	271
B.3.3. Pojedyncza runda algorytmu	272
B.3.4. Podsumowanie	274
B.4. Twofish	275
B.4.1. Opis algorytmu	275
B.4.2. Pojedyncza runda algorytmu	275
B.4.3. Podsumowanie	280
B.5. CAST5	280
B.5.1. Opis algorytmu	280
B.5.2. Rundy CAST5	281
B.6. Blowfish	282
B.6.1. Opis algorytmu	282
B.6.2. Funkcja algorytmu Blowfish	283
B.7. DSA	284
B.7.1. Podpisywanie wiadomości	284
B.7.2. Weryfikacja podpisu	285
B.7.3. Inne warianty DSA	285
B.8. RSA	287
B.8.1. Generowanie pary kluczy	287
B.8.2. Szyfrowanie i deszyfrowanie	287
B.9. Inne algorytmy szyfrujące	288
Dodatek C Kryptografia w służbie historii	291
C.1. Święte rysunki	292
C.1.1. 1000 lat później...	293
C.1.2. Szyfr faraonów	294
C.1.3. Ziarno przeznaczenia	295
C.1.4. Je tiens l'affaire!	296
C.1.5. Tajemnica hieroglifów	297
C.2. Język mitów	298
C.2.1. Mit, który okazał się prawdziwy	298
C.2.2. Trojaczki Kober	301
C.2.3. Raport z półwecza	303
C.3. Inne języki	305
Bibliografia	307
Skorowidz	309

Rozdział 1.

Historia kryptografii

Dążenie do odkrywania tajemnic tkwi głęboko w naturze człowieka, a nadzieja dotarcia tam, dokąd inni nie dotarli, pociąga umysły najmniej nawet skłonne do dociekań. Niektórym udaje się znaleźć zajęcia polegające na rozwiązywaniu tajemnic... Ale większość z nas musi zadowolić się rozwiązywaniem zagadek ułożonych dla rozrywki: powieściami kryminalnymi i krzyżówkami. Odczytywaniem tajemniczych szyfrów pasjonują się nieliczne jednostki.

John Chadwick

Jeszcze nigdy tak wielu nie zawdzięczało tak wiele tak niewielu.

Winston Churchill

Szyfr Cezara wprowadzono w armii rosyjskiej w roku 1915, kiedy okazało się, że sztabowcom nie można powierzyć niczego bardziej skomplikowanego.

Friedrich L. Bauer

1.1. Prolog — Painvin ratuje Francję

21 marca 1918 roku o godzinie 4:30 rozpoczął się największy ostrzał artyleryjski I wojny światowej. Przez pięć godzin niemieckie działa pluły ogniem na pozycje połączonych sił brytyjskich i francuskich. Następnie 62 dywizje niemieckie zalały front na odcinku 60 kilometrów. Dzień po dniu alianci zmuszani byli do wycofywania się i dopiero tydzień później ofensywa została zatrzymana. Do tego czasu wojska niemieckie wbiły się 60 km poza linię frontu. Sukces ten wynikał w dużej mierze z przewagi liczebnej, jaką dysponowały — po kapitulacji Rosji przerzucono do Francji dywizje do tej pory związane walką na froncie wschodnim. Rozciągnięta linia frontu zmuszała obrońców do znacznego rozproszenia sił, co skwapliwie wykorzystywał generał Erich von Ludendorff. Jego taktyka opierała się na koncentrowaniu dużych sił w jednym punkcie i atakowaniu z zaskoczenia. Poznanie planów nieprzyjaciela było kluczowe dla skutecznej obrony. Dzięki temu możliwe stałoby się zgromadzenie większych sił na zagrożonym odcinku frontu. Prowadzono więc intensywny nasłuch radiowy i przechwytywano liczne meldunki przesyłane między niemieckimi centrami dowodzenia, problem polegał jednak na tym, iż w większości wyglądały one mniej więcej tak:

XAXXF AGXVF DXGGX FAFFA AGXFD XGAGX AVDFA GAXFX
GAXGX AGXVF FGAXA...

Był to nowy szyfr stosowany przez niemieckie wojska. Nazwano go ADFGX od stosowanych liter alfabetu tajnego. Ich wybór nie był przypadkowy. W alfabecie Morse'a różniły się one w istotny sposób, dzięki czemu ewentualne zniekształcenia komunikatów radiowych były minimalne.

Jedynym sukcesem francuskiego wydziału szyfrów na tym etapie było złamanie innego niemieckiego systemu, tzw. Schlüsselheft. Był to jednak szyfr stosowany głównie do komunikacji między oddziałami w okopach, natomiast naprawdę istotne informacje chronione były przy użyciu ADFGX. Wprowadzenie tego szyfru praktycznie oślepiło francuskie centrum dowodzenia. Najdobitniej świadczą o tym słowa ówczesnego szefa francuskiego wywiadu:

„Z racji mego stanowiska jestem najlepiej poinformowanym człowiekiem we Francji, a w tej chwili nie mam pojęcia, gdzie są Niemcy. Jak nas dopadną za godzinę, nawet się nie zdziwię”¹.

Oczywiście Bureau du Chiffre nie pozostawało bezczynne. Zadanie złamania niemieckiego szyfru powierzono najlepszemu z francuskich kryptoanalityków — Georges'owi Painvinowi. Jednak nawet on nie był w stanie przeniknąć spowijającej ów szyfr tajemnicy. Zdołał jedynie ustalić, iż system oparty jest na szachownicy szyfrującej i że klucze zmienia się codziennie. Te informacje mogłyby się na coś przydać, gdyby przechwycono większą liczbę zaszyfrowanych depeesz. Ta jednak była zbyt skromna i szyfr nadal pozostawał zagadką.

Sytuacja zmieniła się dopiero na początku kwietnia, kiedy Niemcy zwiększyli liczbę przekazów radiowych. W ręce Painvina wpadła większa ilość materiału do badań, co dało nadzieję na uczynienie pierwszych postępów w łamaniu szyfru. Po wstępnej analizie francuski kryptoanalityk zauważył, iż niektóre wiadomości pochodzące z tego samego dnia mają identyczne początki. Założył więc, że są to te same nagłówki meldunków zaszyfrowane kluczem dziennym. Pozwoliło mu to wydobyć pierwsze informacje na temat konstruowania tego klucza. Następnie posegregował wiadomości na segmenty o takich samych początkach i przesuwając je względem siebie, próbował znaleźć kolejne prawidłowości. Ogromnie pomocne okazało się przechwycenie 18 wiadomości tego samego dnia. Wszystkie były zaszyfrowane tym samym kluczem, dzięki czemu Painvin był w stanie porównać je ze sobą i wyodrębnić stosowane do szyfrowania pary liter (AA, AD, AF itd.). Następnie policzył częstotliwość występowania poszczególnych par. Najczęściej pojawiała się kombinacja DG. Nasunęło mu to podejrzenie, iż odpowiadała ona literze e, najczęściej pojawiającej się w języku niemieckim. Udało mu się również ustalić wygląd stosowanej tablicy (patrz rysunek 1.1).

¹ Kahn D., *Łamacze kodów — historia kryptologii*, Wydawnictwa Naukowo-Techniczne, Warszawa 2004.

Rysunek 1.1.

Tablica podstawień
szyfru ADFGX
ustalona przez
Painvina

	A	D	F	G	X
A					
D				e	
F					
G					
X					

Na niemieckim systemie szyfrowania pojawiła się pierwsza rysa. Był to jednak dopiero początek drogi. Teraz należało ustalić współrzędne pozostałych liter. Rozpoczęły się długie dni mozolnej analizy statystycznej przechwyconych kryptogramów. Painvin porównywał częstotliwość występowania pojedynczych liter w parach i na tej podstawie dzielił kryptogramy. Przypisał każdej literze dwie współrzędne — górną i boczną — a następnie próbował je ustalić. Opierał się na każdym, najmniejszym nawet strzępku informacji, jaki udało mu się zdobyć: na częstości występowania czy parzystości lub nieparzystości sumy współrzędnych. Mozolnie, litera po literze, zrekonstruował niemiecką tabelę podstawień i był teraz w stanie rekonstruować dzienne klucze niemieckich szyfrantów. Przed końcem maja doszedł do takiej wprawy, iż otrzymane wiadomości był w stanie odczytać już po dwóch dniach. I wtedy stało się to, czego najbardziej się obawiał. Niemcy zmienili szyfr.

Komunikaty niemieckie przechwycone 1 czerwca zawierały dodatkową literę — V. Oznaczało to zmianę wyglądu tabeli szyfrowania i być może całego systemu. Tymczasem niemiecka ofensywa trwała. Decydujący atak był kwestią czasu, a Francuzi stracili właśnie możliwość przewidzenia, w którym miejscu nastąpi. Po długiej, bezsennej nocy i kolejnym dniu pracy Painwinowi udało się jednak, poprzez porównywanie starych i nowych kryptogramów, odtworzyć szachownicę szyfrowania (patrz rysunek 1.2).

Rysunek 1.2.

Tablica szyfru
ADFGVX

	A	D	F	G	V	X
A	c	o	8	x	f	4
D	m	k	3	a	z	9
F	n	w	1	0	j	d
G	5	s	i	y	h	u
V	p	1	v	b	6	r
X	e	q	7	t	2	g

Czym prędzej zabrał się do łamania przechwyconych wiadomości i już tego samego dnia udało mu się wysłać pierwsze cenne informacje do sztabu dowodzenia. Mniej więcej w tym samym czasie pierwsze pociski z niemieckich dział dalekosiężnych spadły na Paryż...

Czasu było coraz mniej. Linia frontu była zbyt długa, by należycie zabezpieczyć wszelkie możliwe punkty ataku. Należało więc za wszelką cenę zdobyć informację, gdzie Ludendorff zamierza uderzyć. Francuzi wzmocnili nasłuch radiowy i czekali. Trzeciego czerwca udało się przechwycić depeszę z niewielkiego miasteczka Remaugies, opanowanego

przez wojska niemieckie. Po jej odczytaniu okazało się, iż zawiera ona rozkaz przysłania dużej ilości amunicji. To mogło być to! Ciężki ostrzał artyleryjski przed rozpoczęciem szturmów był powszechną praktyką. Zwiad lotniczy istotnie zaobserwował w ciągu kolejnych dni dużą liczbę ciężarówek na drogach prowadzących do Remaigies. Hipotezę o ataku potwierdzały również informacje od schwytanych jeńców i dezertorów. Prawdopodobną datę ataku wyznaczono na 7 czerwca.

Nie pozostawało już nic innego, jak tylko wzmocnić odpowiedni odcinek frontu i czekać. Wzmocniono obie linie obrony i poinformowano oficerów o zbliżającym się natarciu. Wreszcie nadszedł decydujący dzień. W nerwowym oczekiwaniu żołnierze spoglądali w kierunku niemieckich umocnień. Nic się jednak nie działo. Tak upłynął 7 czerwca, a po nim 8. Napięcie rosło. Oczywiście możliwe było pewne opóźnienie ataku, a informacje od jeńców mogły być nieścisłe, a jednak... w serca obrońców wkraść się niepokój. Wreszcie o północy 9 czerwca niemieckie działa otworzyły ogień. Francuskie linie były bombardowane przez 3 godziny z niespotykaną dotąd intensywnością. Chwilę później nastąpił atak.

Do przodu ruszyło 15 niemieckich dywizji. Kolejnych pięć dni wypełnionych było ciągłą walką o każde miasteczko i ulicę. Niemcy postępowali naprzód, by kolejnego dnia ustępować przed kontratakami Francuzów. Jeśli jednak ktokolwiek był zaskoczony przebiegiem bitwy, to jedynie generał von Ludendorf. Po raz pierwszy nie udało mu się skoncentrowanym atakiem przełamać linii oporu wroga. Co więcej, wróg odważnie kontratakował. W ciągu następnych tygodni próbował jeszcze kolejnych ataków, jednak wkrótce zabrakło mu sił. Paryż został ocalony. A wraz z nim Francja.

Wkrótce potem w Europie wylądowały siły amerykańskie. Dzięki ich wsparciu alianci byli w stanie przystąpić do kontrofensywy, zmuszając Niemców do odwrotu i ostatecznie do poddania się. Niemieccy generałowie podpisali akt kapitulacji 11 października w miejscowości Compiegne. I wojna światowa została zakończona. A Painvin? Cóż... Painvin pojechał na zasłużony urlop. Po latach, zapytany o historię złamania szyfru ADFGVX, odpowiedział:

„Osiągnięcie to pozostawiło niezmywalny ślad na mej duszy i pozostało jednym z najjaśniejszych i najwspanialszych wspomnień w całym moim życiu”².

I trudno mu się chyba dziwić. Nie każdemu dane jest ocalić własny kraj.

Przytoczona tu historia stanowi niewątpliwie znakomity materiał na film. Wiele osób może zadziwić to, jak wielki wpływ na losy wojny może mieć jeden człowiek. Oczywiście bez odpowiedniej reakcji ze strony dowództwa, odpowiedniego planowania i wykorzystywania zdobytej przewagi, a przede wszystkim bez odwagi i poświęcenia zwykłych żołnierzy, którzy oddali życie za swój kraj, informacje zdobyte przez Painvina zostałyby zmarnowane. Z drugiej jednak strony, gdyby nie on, szanse na ocalenie Paryża byłyby nikłe. Upadek stolicy wpłynąłby zaś nie tylko na losy Francji, ale i na wynik całej wojny.

² Kahn D., *Łamacze kodów — historia kryptologii*, op.cit.

Tymczasem z punktu widzenia historii kryptografii przypadek francuskiego kryptoanalityka nie jest niczym niezwykłym. Historia jest pełna opowieści o jemu podobnych, którzy łamiąc szyfr, decydowali o losach setek, tysięcy lub nawet milionów ludzi. Jednak ich osiągnięcia często wychodziły na jaw dopiero po latach, kiedy tajemnice rządowe mogły zostać bezpiecznie ujawnione. Byli więc szarymi eminencjami historii, wpływali na bieg politycznych negocjacji, gry wywiadów czy wreszcie wojen. Wszystko dzięki znakomitemu opanowaniu sztuki „sekretnego pisma” pozwalającej na odkrywanie cudzych tajemnic i zabezpieczanie swoich. Historia kryptografii to opowieść o tych właśnie ludziach. A zatem posłuchajcie...

1.2. Początek...

Na początku było pismo. Wykształcone niezależnie w wielu kulturach stanowiło niezbadaną tajemnicę dla tych, którzy nie potrafili czytać. Szybko jednak zrodziła się konieczność ukrycia informacji również przed tymi, którym umiejętność ta nie była obca. Najbardziej oczywistym rozwiązaniem było schowanie tajnej wiadomości przed ludźmi, którzy mogliby ją odczytać. Takie zabiegi wkrótce jednak przestały wystarczać. Wiadomość mogła zostać odnaleziona podczas wnikliwego przeszukania, a wtedy tajne informacje dostałyby się w ręce wroga. A gdyby udało się napisać list działający na zasadzie „drugiego dna”? Z pozoru zawierałby on błahę treść, jednak jeśli adresat wiedziałby, gdzie i jak szukać, mógłby dotrzeć do „mniej niewinnych” informacji. Tak narodziła się steganografia.

1.2.1. Steganografia

Steganografia to ogół metod ukrywania tajnych przekazów w wiadomościach, które nie są tajne. Jej nazwa wywodzi się od greckich słów: *steganos* (ukryty) oraz *graphein* (pisać). W przeszłości stosowano wiele wymyślnych sposobów osiągnięcia tego efektu. Popularny niewidzialny atrament to jeden z najbardziej znanych przykładów steganografii. Pierwsze zapiski na temat stosowania tej sztuki znaleźć można już w księgach z V w. p.n.e. Przykładem może być opisana przez Herodota historia Demaratos, Greka, który ostrzegł Spartan przed przygotowywaną przeciw nim ofensywą wojsk perskich. Nie mógł on wysłać oficjalnej wiadomości do króla, zeszkrobał więc wosk z tabliczki i wrył tekst w drewnie. Następnie ponownie pokrył tabliczkę woskiem i wręczył posłańcowi. Czysta tabliczka nie wzbudziła podejrzeń perskich patroli i bezpiecznie dotarła do celu. Tam co prawda długo głowiono się nad jej znaczeniem, wkrótce jednak żona spartańskiego wodza Leonidasa wpadła na pomysł zeszkrobania wosku, co pozwoliło odkryć tajną wiadomość.

W miarę postępu technicznego, a także rozwoju samej steganografii, powstawały coraz wymyślniejsze metody ukrywania wiadomości. Znana jest na przykład metoda ukrywania wiadomości w formie kropki w tekście drukowanym, stosowana podczas II wojny światowej. Wiadomość była fotografowana, a klisza pomniejszana do rozmiarów około milimetra kwadratowego i naklejana zamiast kropki na końcu jednego ze

zdań w liście. Obecnie bardzo popularne jest ukrywanie wiadomości w plikach graficznych. Kolejne przykłady można mnożyć, jednak nawet najbardziej wymyślne z nich nie gwarantują, iż wiadomość nie zostanie odkryta. Koniecznością stało się zatem wynalezienie takiego sposobu jej zapisywania, który gwarantowałby tajność nawet w przypadku przechycenia przez osoby trzecie.

1.2.2. Kryptografia

Nazwa *kryptografia* również wywodzi się z języka greckiego (od wyrazów *kryptos* — ukryty i *graphein* — pisać). Jej celem jest utajnienie znaczenia wiadomości, a nie samego faktu jej istnienia. Podobnie jak w przypadku steganografii data jej powstania jest trudna do określenia. Najstarsze znane przykłady przekształcenia pisma w formę trudniejszą do odczytania pochodzą ze starożytnego Egiptu, z okresu około 1900 roku p.n.e. Pierwsze tego typu zapisy nie służyły jednak ukrywaniu treści przed osobami postronnymi, a jedynie nadaniu napisom formy bardziej ozdobnej lub zagadkowej. Skrybowie zapisujący na ścianach grobowców historii swych zmarłych panów świadomie zmieniali niektóre hieroglify, nadając napisom bardziej wzniosłą formę. Często celowo zacierali ich sens, zachęcając czytającego do rozwiązania zagadki. Ten element tajemnicy był ważny z punktu widzenia religii. Skłaniał on ludzi do odczytywania epitafium i tym samym do przekazania błogosławieństwa zmarłemu. Nie była to kryptografia w ścisłym tego słowa znaczeniu, zawierała jednak dwa podstawowe dla tej nauki elementy — przekształcenie tekstu oraz tajemnicę.

Na przestrzeni kolejnych 3000 lat rozwój kryptografii był powolny i dosyć nierówny. Powstawała ona niezależnie w wielu kręgach kulturowych, przybierając różne formy i stopnie zaawansowania. Zapiski na temat stosowania szyfrów znaleziono na pochodzących z Mezopotamii tabliczkach z pismem klinowym. Ich powstanie datuje się na 1500 rok p.n.e. W II w. p.n.e. grecki historyk Polibiusz opracował system szyfrowania oparty na tablicy przyporządkowującej każdej literze parę cyfr (patrz rysunek 1.3).

Rysunek 1.3.
Tablica Polibiusza

	1	2	3	4	5
1	A	B	C	D	E
2	F	G	H	I/J	K
3	L	M	N	O	P
4	Q	R	S	T	U
5	V	W	X	Y	Z

W późniejszych czasach tablica ta stała się podstawą wielu systemów szyfrowania. Przekształcenie liter w liczby dawało możliwość wykonywania dalszych przekształceń za pomocą prostych obliczeń lub funkcji matematycznych. Metodę Polibiusza uzupełnioną kilkoma dodatkowymi utrudnieniami kryptoanalitycznymi zastosowała m.in. niemiecka armia przy opracowywaniu wspomnianego na wstępie systemu szyfrującego ADFGX oraz jego udoskonalonej wersji ADFGVX.

1.2.3. Narodziny kryptoanalizy

Kolebką kryptoanalizy były państwa arabskie, które najlepiej opanowały sztukę lingwistyki i statystyki, na nich bowiem opierała się technika łamania szyfrów monoalfabetycznych. Najwcześniejszy jej opis znajduje się w pracy Al-Kindiego, uczonego z IX wieku, znanego jako „filozof Arabów” (napisał on 29 prac z dziedziny medycyny, astronomii, matematyki, lingwistyki i muzyki). Jego największy traktat, *O odczytywaniu zaszyfrowanych listów*, został odnaleziony w 1987 roku w Archiwum Ottomańskim w Stambule. W pracy tej Al-Kindi zawarł szczegółowe rozważania na temat statystyki fonetyki i składni języka arabskiego oraz opis opracowanej przez siebie techniki poznawania tajnego pisma. To jeden z pierwszych udokumentowanych przypadków zastosowania ataku kryptoanalitycznego. Pomysł arabskiego uczonego był następujący:

„Jeden sposób na odczytanie zaszyfrowanej wiadomości, gdy wiemy, w jakim języku została napisana, polega na znalezieniu innego tekstu w tym języku, na tyle długiego, by zajął mniej więcej jedną stronę, i obliczeniu, ile razy występuje w nim każda litera. Literę, która występuje najczęściej, będziemy nazywać »pierwszą«, następną pod względem częstości występowania »drugą« i tak dalej, aż wyczerpiemy listę wszystkich liter w próbie jawnego tekstu.

Następnie bierzemy tekst zaszyfrowany i również klasyfikujemy użyte w nim symbole. Znajdujemy najczęściej występujący symbol i zastępujemy go wszędzie „pierwszą” literą z próbki jawnego tekstu. Drugi najczęściej występujący symbol zastępujemy „drugą” literą, następny „trzecią” i tak dalej, aż wreszcie zastąpimy wszystkie symbole w zaszyfrowanej wiadomości, którą chcemy odczytać³.

Opisana powyżej metoda znana jest jako analiza częstości i po dziś dzień stanowi podstawową technikę kryptoanalityczną. Każdy język posiada własną charakterystykę występowania poszczególnych liter w piśmie, zawsze jednak pewne znaki pojawiają się częściej niż inne. Na tej podstawie kryptoanalityk może zidentyfikować te litery w kryptogramie. To z kolei pozwala odgadnąć niektóre ze znajdujących się w tajnym piśmie wyrazów, dzięki czemu rozszyfrowuje się kolejne litery itd. Wszystko opiera się tutaj w dużej mierze na prawdopodobieństwie, gdyż najczęściej występujący w kryptogramie znak wcale nie musi być literą najczęściej występującą w danym języku. Niemniej jednak znajomość tej metody pozwalała znacznie zredukować liczbę możliwych podstawień i osiągnąć rozwiązanie metodą prób i błędów.

Należy tu podkreślić, że jeśli mamy do czynienia z jedną krótką wiadomością, analiza częstości występowania znaków może dać fałszywe wyniki (w tych kilku konkretnych zdaniach najczęściej pojawiającą się literą może być na przykład czternasta pod względem częstości występowania w danym języku) i utrudnić dekryptaż. **Stąd też im dłuższy jest zaszyfrowany tekst, tym większa szansa na złamanie szyfru.**

Dzięki wynalazkowi Al-Kindiego monoalfabetyczne systemy szyfrujące przestały być bezpieczne. Od tej chwili rozpoczął się trwający do dziś wyścig kryptografów z kryptoanalitykami.

³ Singh S., *Księga szyfrów*, Albatros, Warszawa 2001, s. 31.

1.3. Rozwój kryptografii i kryptoanalizy

Jeszcze wiele lat po odkryciu Al-Kindiego liczni uczeni negowali możliwość złamania szyfru podstawieniowego. Szybko jednak metody kryptoanalityczne rozprzestrzeniły się z Bliskiego Wschodu na Europę. W średniowieczu nie dokonał się większy postęp w europejskiej kryptologii. Szyfry znane były mnichom i skrybom, a i ci nie traktowali ich jako odrębnej nauki, a jedynie jako rodzaj intelektualnej rozrywki. Aż do początków XV wieku używano wyłącznie szyfrów podstawieniowych. Popularne były również tzw. *nomenklatory*. Było to połączenie szyfru podstawieniowego z kodem — oprócz klasycznego alfabetu tajnego nomenklator zawierał listę słów i ich odpowiedników kodowych. Prawdziwy rozkwit technik szyfrowania nastąpił równoległe z rozwojem i umacnianiem stosunków dyplomatycznych między europejskimi państwami. Ambasadorowie, pełniący jednocześnie rolę szpiegów na obcych dworach, potrzebowali sposobu na bezpieczne przekazywanie tajnych informacji. Z tych samych powodów wzrosło zainteresowanie kryptoanalizą. W związku z dokonanymi w tej dziedzinie postęпами szyfry monoalfabetyczne nie były już bezpieczne, zaczęto więc opracowywać nowe metody szyfrowania.

1.3.1. Szyfry homofoniczne

Jedną z najbardziej znanych metod jest szyfrowanie z użyciem homofonów. Miało ono zabezpieczyć szyfr przed atakiem z użyciem analizy częstości. Pierwszy znany przykład szyfru homofonicznego pochodzi z roku 1401. W szyfrach takich alfabet tekstu tajnego wzbogacano o pewne dodatkowe symbole, które następnie przypisywano literom najczęściej występującym w alfabecie tekstu jawnego. I tak, jeśli częstość występowania danej litery wynosiła 7%, przypisywano jej 7 różnych symboli. W ten sposób każdy znak tekstu tajnego pojawiał się w wiadomości z taką samą częstością. Mogłoby się wydawać, że od tej chwili tajne wiadomości pozostaną nieodczytane. Nic bardziej mylnego.

Częstość występowania liter nie jest jedyną charakterystyką języka. Istnieją również liczne powiązania między literami, takie jak częstość pojawiania się określonych par i trójek. Poszczególne wyrazy w języku również charakteryzują się określoną częstością występowania. Dzięki takim prawidłowościom możliwa jest kryptoanaliza szyfrów homofonicznych poprzez wyszukiwanie tzw. częściowych powtórzeń. Załóżmy dla przykładu, iż szyfrowanie opiera się na podstawianiu par cyfr zamiast liter. Literom o większej częstości występowania przypisana jest większa liczba kombinacji dwucyfrowych. Tak skonstruowany szyfr można złamać przy odpowiedniej ilości materiału do badań. Wystarczy wyszukać w tekście podobne kombinacje znaków, na przykład: 67 55 10 23 i 67 09 10 23. Z dużą dozą prawdopodobieństwa założyc można, iż odpowiadają one tym samym wyrazom. Dzięki temu łatwo zidentyfikować zestawy cyfr odpowiadające tej samej literze (w naszym przykładzie — 55 i 09). Po odtworzeniu odpowiedniej liczby takich powiązań szyfr złamać można tradycyjną metodą analizy częstości. Zaczęto więc udoskonalać szyfry homofoniczne, aby uodpornić je na tego typu kryptoanalizę.

Bardzo wiele usprawnień w szyfrowaniu wprowadziła włoska rodzina Argenticch. W XVI i XVII wieku jej członkowie pracowali dla kolejnych papieży, służąc im swoją bogatą wiedzą kryptologiczną. Na początku XVII wieku wprowadzili liczne udoskonalenia w stosowanych wówczas technikach szyfrowania.

Przed wszystkim stosowali symbole puste w każdym wierszu kryptogramu. Zlikwidowali również rozdzielanie wyrazów i zapisywanie znaków interpunkcyjnych. Nawet cyfry odpowiadające poszczególnym literom zapisywali razem, mieszając często liczby jedno- i dwucyfrowe. Dzięki tym zabiegom problem pojawiał się już na etapie podziału tekstu tajnego na pojedyncze znaki. Oczywiście złamanie szyfru nadal było możliwe, jednak zadanie to było znacznie trudniejsze niż w przypadku zwykłego szyfru homofonicznego.



Wskazówka

Symbol pusty — znak alfabetu tajnego nieposiadający odpowiednika w alfabecie jawnym. Adresat wiadomości podczas dekryptażu ignoruje takie znaki, natomiast dla kryptoanalityka są one dodatkowym utrudnieniem.

1.3.2. Szyfry polialfabetyczne

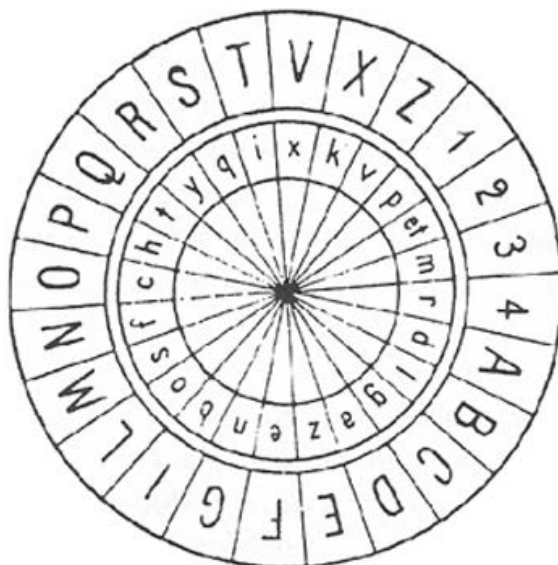
Szyfry polialfabetyczne opisać można jako połączenie wielu szyfrów monoalfabetycznych. Mają wiele alfabetów tajnych, z których każdy szyfruje jeden znak tekstu tajnego. Używane są cyklicznie, a więc po wyczerpaniu wszystkich powraca się do pierwszego i kontynuuje szyfrowanie. Prawdopodobnie pierwszym zastosowanym szyfrem polialfabetycznym był szyfr Albertiego, włoskiego architekta z XV wieku.

1.3.2.1. Tarcza Albertiego

Urodzony w roku 1404 Leone Battista Alberti był człowiekiem niezwykle wszechstronnym — komponował, malował, pisał, zajmował się aktorstwem, architekturą, prawem. Kryptografią zainteresował się dosyć późno, bo dopiero w roku 1466, za namową Leonardo Dato — ówczesnego papieskiego sekretarza.

Alberti napisał obszerną rozprawę o tematyce kryptologicznej. Obejmowała ona zarówno zagadnienia kryptoanalizy, jak i metodologii tworzenia nowych szyfrów. Architekt opisał w niej również swój własny szyfr i stwierdził, iż nikt nie będzie w stanie go złamać. Szyfr ten opierał się na urządzeniu zaprojektowanym przez niego samego. Składało się ono z dwóch okrągłych tarcz (patrz rysunek 1.5).

Jedna z nich zawierała się wewnątrz drugiej, na obu zaś, na osobnych polach, wypisane były litery alfabetu. Szyfrowanie polegało na zastępowaniu liter z małej tarczy literami znajdującymi się na odpowiadających im polach dużej. Wszystko to tworzyło by jedynie prosty szyfr monoalfabetyczny, gdyby nie fakt, iż wewnętrzna tarcza była ruchoma. Obracając ją, szyfrujący zmieniał przypisania wszystkich używanych liter, tym samym wybierając nowy alfabet szyfrowy. Oczywiście osoby prowadzące zaszyfrowaną korespondencję przy użyciu tarczy Albertiego muszą posiadać jej identyczne egzemplarze i ustalić początkową pozycję wewnętrznej tarczy względem zewnętrznej.

Rysunek 1.5.*Tarcza Albertiego**Źródło: Kahn D.,**Łamacze kodów**— historia kryptologii,**op.cit.*

Dodatkowo włoski architekt umieścił na zewnętrznej tarczy cyfry od 1 do 4, co umożliwiało wstawianie do wiadomości słów kodowych (na przykład nazwy własne mogły być zastępowane kombinacjami cyfr). W połączeniu z wynalezieniem szyfru polialfabetycznego i dokonaniem pierwszego na Zachodzie opisu kryptoanalizy stanowiło to niebywale osiągnięcie, zwłaszcza jak na człowieka, który kryptografią zajmował się raptem kilka lat. Osiągnięcia Albertiego zyskały mu miano *ojca kryptologii Zachodu*.

Szyfrowanie z użyciem wielu alfabetów stanowiło wielki przełom, jednak stosowanie w tym celu urządzenia szyfrującego powodowało pewne niedogodności. Pół wieku później zupełnie inny sposób wykorzystania techniki szyfrowania polialfabetycznego zaproponował niemiecki uczone Johannes Trithemius.

1.3.2.2. Tabula recta

Trithemius urodził się 2 lutego 1462 roku w Tritthenheim w Niemczech. W wieku 17 lat rozpoczął studia na uniwersytecie w Heidelbergu, gdzie szybko zdobył uznanie dzięki swemu niebywałemu intelektowi. Mając lat dwadzieścia, przez przypadek trafił do opactwa benedyktynów. Życie mnichów zafascynowało go do tego stopnia, iż postanowił rozpocząć nowicjat. Niecałe dwa lata później wybrany został opatem.

Oprócz sprawowania swego nowego stanowiska Trithemius zajmował się pisaniem książek. Pierwsza z nich została opublikowana, kiedy miał 24 lata. Pisał opowieści, słowniki, biografie, kroniki oraz kazania. Prowadził też bogatą korespondencję z innymi uczonymi. W roku 1499 rozpoczął pisanie książki pt. *Steganographia*. Opisowała ona znane metody szyfrowania. Tak naprawdę jednak w książce tej więcej było okultyzmu i czarnej magii niż kryptografii. Trithemius nie ukrywał swej fascynacji praktykami magicznymi i lubił uchodzić za cudotwórcę. Ze zrozumiałych względów kościelni zwierzchnicy zdecydowanie potępiali postępowanie opata i ostatecznie nie ukończył on swojej książki.

W roku 1508 Trithemius powrócił do tematyki kryptologicznej, tym razem traktując temat bardziej naukowo. Jego kolejna książka — *Poligraphia* — skupiała się wyłącznie na zagadnieniach czysto kryptograficznych. Ukazała się ona dopiero w roku 1518, dwa lata po śmierci uczonego. Była to pierwsza książka na temat kryptologii wydana drukiem. Jej tytuł brzmiał: *Sześć ksiąg o poligrafii przez Johannes Trithemiusa, opata w Wurzburgu, poprzednio w Spanheim, dla cesarza Maksymiliana*. Książka zawierała głównie kolumny słów używanych przez Trithemiusa w jego systemach kryptograficznych. W księdze piątej znajdował się jednak opis nowego systemu szyfrowania polialfabetycznego. Opierał się on na specjalnej tabeli nazwanej przez Trithemiusa *tabula recta*. Przedstawia ją rysunek 1.6.

Rysunek 1.6.

Tabela Trithemiusa

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
1.	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
2.	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A
3.	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B
4.	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C
5.	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D
6.	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E
7.	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F
8.	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G
9.	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H
10.	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I
11.	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J
12.	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K
13.	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L
14.	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M
15.	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N
16.	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
17.	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
18.	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
19.	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
20.	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
21.	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
22.	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
23.	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V
24.	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
25.	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X
26.	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y

Na samej górze tabeli umieszczono alfabet tekstu tajnego. Kolejne linijki to tajne alfabety utworzone przez przenoszenie kolejnych liter z początku alfabetu na jego koniec. W ten sposób Trithemius uzyskał 26 alfabetów szyfrowych.

Szyfrowanie tą metodą przebiega następująco: dla pierwszej litery tekstu jawnego używa się pierwszej linijki tabeli, dla drugiej litery — drugiej linijki itd. Pozwala to na zabezpieczenie tekstu przed atakiem przez analizę częstości. Jednak podobnie jak w przypadku szyfru Cezara nie chroni to przed odszyfrowaniem w przypadku, gdy kryptoanalityk zna stosowany algorytm. Próba rozwiązania tego problemu był opublikowany w 1586 roku szyfr Vigenere’a.

1.3.2.3. Le chiffre indechiffable

Blaise de Vigenere urodził się 5 kwietnia 1523 roku we Francji. W wieku 23 lat rozpoczął karierę dyplomatyczną na dworze w Wormancji. Podróżował po całej Europie i rok później został przyjęty na służbę u księcia de Nevers. W roku 1549 podczas misji dyplomatycznej w Rzymie Vigenere po raz pierwszy zetknął się z kryptografią. Ogromnie zafascynowany sztuką „tajnego pisma” oddał się studiowaniu książek największych kryptologów oraz własnym badaniom. Miał również możliwość współpracy z najwybitniejszymi ekspertami kurii papieskiej, co pozwoliło mu znacznie pogłębić wiedzę. Dzięki niej i bogatemu doświadczeniu został sekretarzem samego króla. W końcu w wieku 47 lat postanowił opuścić dwór i zająć się pisaniem książek.

W roku 1586 Vigenere opublikował *Traktat o szyfrach*. Podobnie jak w dziele Trithemiusa, tak i tutaj znajdują się liczne dygresje na tematy zupełnie niezwiązane z kryptografią, za to jak najbardziej związane z czarną magią. Autor zachował mimo to naukową solidność w tych fragmentach książki, które w ogóle miały coś z nauką wspólnego. Opisał również własny szyfr polialfabetyczny.

System opracowany przez Vigenere’a polegał na szyfrowaniu kolejnych liter wiadomości za pomocą różnych wierszy tablicy Trithemiusa. Różnica polegała na sposobie wyboru kolejnego wiersza szyfrującego. Dla pierwszej litery mógł to być wiersz 17., dla drugiej — 5., dla trzeciej — 13. itd. W ten sposób znajomość samego systemu przewidywała wystarczać do odszyfrowania wiadomości. Trzeba było jeszcze znać kombinację wierszy zastosowaną w danym przypadku. Nadawca i odbiorca mogli sobie ułatwić zapamiętanie tej kombinacji, ustalając specjalne słowo klucz. Jego litery stanowiły jednocześnie pierwsze litery kolejno stosowanych wierszy szyfrowania. Dla przykładu: słowo kluczowe *sekret* oznaczało, iż do zaszyfrowania pierwszej litery wiadomości zastosowano 19. wiersz tabeli, dla drugiej — 5., dla trzeciej — 11. itd. Znajomość słowa klucza wystarczała adresatowi do odszyfrowania wiadomości. Odszukiwał on kolejne litery szyfrogramu w odpowiadających im linijkach tabeli, po czym odczytywał literę tekstu jawnego z linijki znajdującej się na samej górze.

Vigenere stworzył również dwa systemy szyfrowania oparte na koncepcji autoklucza. W pierwszym przypadku kluczem stawał się odszyfrowywany tekst jawny. Konieczna była jedynie znajomość pojedynczej litery, stanowiącej tzw. *klucz pierwotny*. Dzięki niej adresat odczytywał pierwszą literę tekstu jawnego, którą wykorzystywał do odczytania drugiej itd.

Drugi system z autokluczem również wykorzystywał klucz pierwotny. Tutaj jednak po zaszyfrowaniu pierwszej litery tekstu jawnego jej odpowiednik w kryptogramie stawał się kolejną literą klucza. Obie metody były znacznie bardziej innowacyjne i błyskotliwe niż opracowany przez Vigenere’a szyfr polialfabetyczny, jednak z niewiadomych przyczyn uległy zapomnieniu, a z nazwiskiem francuskiego uczonego kojarzony jest głównie szyfr oparty na tabeli Trithemiusa. Warto również zaznaczyć, iż koncepcja autoklucza została pierwotnie opisana przez włoskiego matematyka Girolamo Cardano, jednak opracowany przez niego system był pełen niedoskonałości i dopiero udoskonalenia wprowadzone przez Vigenere’a pozwalały na wykorzystanie tej metody przy szyfrowaniu wiadomości.

Szyfr Vigenere’a przez bardzo długi czas uchodził za niemożliwy do złamania. Zyskał nawet przydomek *le chiffre indechiffirable* (pol. szyfr nieodszyfrowywalny). Został złamany dopiero w XIX wieku przez brytyjskiego uczonego Charlesa Babbage’a.

1.3.2.4. Złamanie szyfru „nie do złamania”

Charles Babbage urodził się w roku 1792. Pochodził z bogatej rodziny (jego ojciec był bankierem), co pozwoliło mu na rozwijanie różnorodnych zainteresowań, w tym tych dotyczących kryptografii. Już jako dziecko zdradzał wyjątkowy talent w tej dziedzinie, przez co nieraz wpadał w kłopoty — łamał szyfry swoich szkolnych kolegów, a ci w rewanżu spuszczali mu łanie. Wraz z upływem lat rozwijał swoje umiejętności, aż stał się znany w całej Anglii. Często pomagał w przygotowywaniu materiału dowodowego w prowadzonych sprawach sądowych poprzez odszyfrowywanie korespondencji z nimi związanej. W roku 1854 zainteresował się problemem kryptoanalizy szyfru Vigenere’a. Nie przejmując się opiniami, jakoby szyfr ten był nie do złamania, rozpoczął poszukiwanie punktu zaczepienia, który pozwoliłby na skuteczną kryptoanalizę. Jeszcze w tym samym roku dokonał przełomowego odkrycia.

Babbage zauważył mianowicie, że jeśli pozna się długość użytego słowa klucza, rozszyfrowanie tekstu będzie o wiele łatwiejsze, gdyż będzie wtedy wiadomo, które litery zaszyfrowane są przy użyciu takich samych podstawień. Na przykład: jeśli słowo kluczowe ma 5 liter, to co piąta litera tekstu jest szyfrowana przy użyciu identycznego alfabetu. Wystarczy zatem podzielić tekst na grupy liter szyfrowane tą samą literą klucza i dokonać kryptoanalizy opartej na analizie częstości. Grupy te nie są bowiem niczym innym jak prostym szyfrem podstawieniowym.

Oczywiście kryptoanalityk nie zna długości klucza, informację tę można jednak zdobyć podczas badania kryptogramu. Przy dłuższych tekstach często zdarzają się bowiem powtórzenia wyrazów lub ich fragmentów szyfrowane tym samym fragmentem klucza. W takiej sytuacji w kryptogramie wystąpią powtarzające się kombinacje liter. Analizując odległości między nimi, ustalić można najbardziej prawdopodobną długość klucza. Z reguły jest nią jeden ze wspólnych dzielników tych odległości. Jeśli zatem udało nam się wyodrębnić cztery takie przypadki, a odstępy wynoszą 8, 16, 20 i 23 litery, to możemy z dużą dozą prawdopodobieństwa przyjąć, iż długość klucza wynosi cztery. Czasem powtórzenie może być dziełem przypadku, a nie synchronizacji klucza i tekstu, dlatego też ostatnią wartość (23) można zignorować. Zawsze jednak warto odszukać jak najwięcej powtórzeń, gdyż dzięki temu uzyskujemy większą ilość materiału do analizy, a co za tym idzie — większą pewność co do wyznaczonej długości klucza.

Technika zastosowana przez Babbage’a została rozwinięta i usystematyzowana przez pruskiego wojskowego, Friedricha W. Kasickiego. W swojej książce *Die Geheimschriften und die Dechiffrier-kunst* (Tajne pisma i sztuka deszyfracji) szczegółowo opisał on metodykę łamania polialfabetów, począwszy od wyznaczania okresu klucza, a na analizie wyodrębnionych szyfrów monoalfabetycznych skończywszy. Książka stała się znana dopiero po jego śmierci w roku 1881 roku, a opracowaną metodę ochrzczono mianem *analizy Kasickiego*.

1.3.3. Szyfry digraficzne

Szyfr digraficzny opiera się na szyfrowaniu par znaków. Tekst jawny dzielony jest na pary znaków, a następnie przekształcany w kryptogram według ustalonego wzoru. Każdy symbol w kryptogramie jest więc zależny od dwóch liter tekstu jawnego, co utrudnia złamanie szyfru. Szyfry digraficzne zaliczyć można do szerszej grupy szyfrów wieloliterowych (operujących na grupach liter).

Pierwszy znany szyfr digraficzny pochodzi z dzieła *De Furtivis Literarum Notis* autorstwa Giovanniego Battisty Porty — włoskiego uczonego z XVI wieku. Zawierało ono opis znanych ówczesnie szyfrów, lingwistycznych aspektów kryptografii, technik kryptoanalitycznych oraz autorskie propozycje technik szyfrowania. Autor umieścił w nim również liczne cenne wskazówki dotyczące zarówno szyfrowania, jak i łamania szyfrów. To Porta jako pierwszy wpadł na pomysł kryptoanalizy opartej na prawdopodobieństwie występowania słów w tekście. Mówiąc najogólniej, kryptoanalityk znający przeznaczenie danej wiadomości może spróbować odszukać w tekście wyraz często występujący w tekstach o takim charakterze. Na przykład dla meldunku wojkowego mogą to być wyrazy *atak*, *wróg*, *dowódca* itp.

Co ciekawe, Porta nie podzielał powszechnej opinii, jakoby szyfry polialfabetyczne były nie do złamania. Przypuścił wiele ataków na znane wówczas polialfabety i był bardzo blisko sukcesu. W jednym przypadku udało mu się na podstawie występujących powtórzeń określić długość klucza, jednak nie zrobił z tej informacji żadnego użytku. W rezultacie szyfry polialfabetyczne uznawane były za bezpieczne przez kolejnych 300 lat.

Pierwszym w historii literowym szyfrem digraficznym był szyfr Playfaira, nazwany tak od nazwiska angielskiego uczonego epoki wiktoriańskiej. Nazwa ta przylgnęła do tego szyfru, mimo iż tak naprawdę jego autorem był inny uczonec, Charles Wheatstone. Obaj panowie byli jednak do siebie ludzaco podobni, przez co notorycznie ich ze sobą mylono.

Szyfr Playfaira opierał się na tablicy o wymiarach 5x5, w którą wpisywano kolejne litery alfabetu. Można też ją było wypełnić w oparciu o słowo klucz. W takim przypadku wpisywano je w tablicy (ignorując powtarzające się litery), a pozostałe litery wstawiano w puste miejsca w porządku alfabetycznym. Rysunek 1.7 przedstawia tablicę utworzoną w oparciu o słowo *Playfair*.

Rysunek 1.7.

Tablica szyfru
Playfaira

P	L	A	Y	F
I	J	R	B	C
E	G	H	K	M
N	O	Q	S	T
U	V	W	X	Z

Szyfrowanie rozpoczyna się od podzielenia tekstu jawnego na pary znaków (i oraz j traktowano jak ten sam znak, natomiast pary takich samych liter należało oddzielić literą x). Następnie przekształcano wiadomość w kryptogram w oparciu o następujące zasady:

- ◆ Jeśli obie litery znajdowały się w tym samym rzędzie, były zastępowane literami znajdującymi się bezpośrednio po ich prawej stronie. Obowiązywała tutaj zasada cykliczności, tzn. ostatnia litera w rzędzie była zastępowana pierwszą po lewej.
- ◆ Jeśli obie litery znajdowały się w tej samej kolumnie, zastępowano je literami znajdującymi się pod spodem. Tutaj również obowiązywała zasada cykliczności.
- ◆ Litery znajdujące się w innych kolumnach i wierszach były zastępowane literami z tego samego wiersza, ale znajdującymi się w kolumnie drugiej litery tekstu jawnego.

Być może brzmi to nieco zawile. Łatwiej będzie zrozumieć to na przykładzie. Zszyfrujemy wiadomość o treści: *tekst jawny* w oparciu o tablicę zamieszczoną na rysunku 1.7. Po podziale na pary znaków otrzymujemy: *TE KS TJ AW NY*. Litery T i E znajdują się w różnych kolumnach i wierszach, dokonujemy zatem podstawienia zgodnie z trzecią z wymienionych powyżej zasad — T zamienia się w N, a E w M. Kolejna para liter znajduje się w tej samej kolumnie, zastosowanie ma zatem zasada druga. W rezultacie otrzymujemy wynik *SX*. W przypadku trzeciej pary ponownie wykorzystujemy zasadę trzecią, przez co TJ zamienia się w ND. Czwarta para liter szyfrowana jest z użyciem zasady drugiej (*AW* przechodzi w *BA*), natomiast piąta — z użyciem zasady trzeciej (*NY* przechodzi w *SP*).

Szyfry digraficzne są trudniejsze do złamania za pomocą analizy częstości. Liczba digrafów jest zawsze o wiele większa niż liczba liter alfabetu jawnego (na przykład dla 26 liter mamy 676 digrafów) i mają one bardziej równomiernie rozłożoną częstość występowania. Z tego względu szyfr Playfaira przez wiele lat uważany był za wyjątkowo bezpieczny — wykorzystywano go zarówno podczas pierwszej, jak i drugiej wojny światowej.

1.3.4. Prawdziwy „szyfr nie do złamania”

Mimo iż zmagania na arenie kryptografii i kryptoanalizy trwają nieprzerwanie, tak naprawdę od niemal stu lat znany jest szyfr, który przy prawidłowym wykorzystaniu jest niemożliwy do złamania. W tym przypadku twierdzenie to nie jest jedynie pobożnym życzeniem twórców, ale udowodnionym matematycznie faktem. Mowa tu o opracowanym przez Gilberta Vernama i Josepha Mauborgne’a szyfrze z kluczem jednorazowym (ang. *one-time pad*).

Rozwiązanie opracowane przez Vernama i Mauborgne’a polega na szyfrowaniu każdego tekstu jawnego z wykorzystaniem generowanego losowo klucza o tej samej długości. Po wykorzystaniu klucz jest niszczone, a do zaszyfrowania kolejnego komunikatu należy wygenerować następny losowy klucz. W rezultacie kryptoanalityk nie ma żadnego punktu zaczepienia — nie ma tu prawidłowości związanych z cyklicznym

powtarzaniem się klucza, a jeśli nawet poszczególne znaki zaszyfrowane są z użyciem tego samego alfabetu szyfrowego, ich rozmieszczenie ma charakter losowy. Co za tym idzie, sam szyfrogram ma rozkład statystyczny ciągu losowego. Ponieważ klucz ma taką samą długość jak tekst jawny, a jego generowanie nie opiera się na żadnych sformalizowanych zasadach, po przechwyceniu kryptogramu możemy dopasować do niego wiele różnych kluczy, uzyskując różne sensowne teksty jawne, z których każdy jest tak samo prawdopodobny. Przykładowo: dla trzynastoliterowego tekstu tajnego możemy dobrać różne klucze, deszyfrując go jako „ATAKWPOŁUDNIE”, „WIĘZIENU-CIEKL” tudzież „VALARDOHAERIS” i tylko od kontekstu zależy, którą z tych wiadomości uznamy za najbardziej prawdopodobną. Gdybyśmy mieli więcej wiadomości zaszyfrowanych tym samym kluczem, mielibyśmy również potencjalny punkt zaczepienia, ale niepowtarzalność klucza jest podstawową zasadą omawianego systemu.

Być może zastanawiacie się teraz, dlaczego w ogóle przez ostatnie sto lat poszukiwano nowych szyfrów i metod ukrywania informacji. Przecież mamy już idealny system, odporny na wszelkie ataki! Niestety o ile w teorii mamy 100-procentowe bezpieczeństwo, o tyle w praktyce nie jest już tak różowo. Owszem, sam szyfr rzeczywiście jest nie do złamania, ale jego implementacja stanowi nie lada wyzwanie. Podstawowe problemy to:

- ♦ Losowość klucza — nawet obecnie w erze komputerów generowanie ciągów losowych stanowi niezwykle trudne i złożone zagadnienie. Więcej informacji na ten temat znaleźć można w kolejnych rozdziałach, tu wystarczy powiedzieć, że tworzenie naprawdę pozbawionych prawidłowości sekwencji znaków stanowi spore wyzwanie dla twórców oprogramowania i wciąż poszukuje się nowych sposobów na uzyskanie tego efektu.
- ♦ Jednorazowość klucza — dla każdej wiadomości musimy wygenerować co najmniej tak samo długi klucz, co podwaja ilość wymaganej pamięci i przesyłanych danych (oprócz szyfrogramów adresat musi też w jakiś sposób wejść w posiadanie kolejnych partii klucza). Przy zakrojonych na szeroką skalę operacjach, w których udział biorą dziesiątki lub setki tysięcy ludzi, byłby to często ogromny problem techniczny. Intensywnie wykorzystywany kryptosystem może wymagać milionów kluczy na sekundę, co niebagatelnie zwiększyłoby koszt całego przedsięwzięcia.
- ♦ Dystrybucja klucza — jak dostarczymy adresatowi jego kopie kluczy? Przy komunikacji pomiędzy dwoma osobami nie wydaje się to aż tak trudne, choć i tutaj trzeba ustalić jakiś bezpieczny kanał, który od tego momentu będzie wąskim gardłem całego przedsięwzięcia. Co jednak, jeśli komunikacja odbywa się między tysiącami osób na całym świecie? Parafrazując słynne pytanie z „Rejsu” — jaką metodą zaszyfrujemy jednorazowe klucze szyfrujące?

Oczywiście nie oznacza to, że system z kluczem jednorazowym jest nieprzydatny w praktyce. Z powodzeniem można go wykorzystywać w komunikacji o niskim natężeniu, której bezpieczeństwo ma krytyczne znaczenie (tajne informacje wymieniane na wysokim szczeblu).

1.3.5. Kamienie milowe kryptografii

Ogromny wpływ na rozwój kryptografii miało wynalezienie telegrafu. Umożliwiło ono komunikację na niespotykaną dotąd skalę i wywołało dyskusję na temat poufności przekazywanych informacji. W obawie przed nieuczciwymi telegrafistami wiele osób opracowywało własne szyfry „nie do złamania”. Powstawały też liczne książki kodowe spełniające podwójne funkcje — oprócz ochrony tajnych informacji pozwalały one zmniejszyć koszt wysyłanych wiadomości. W książkach takich pojedyncze słowa kodowe odpowiadały bowiem całym zdaniom w tekście jawnym, przez co telegram stał się krótszy.

Telegraf zmienił również oblicze wojny, która teraz mogła być prowadzona na znacznie większym obszarze. Dowódca mógł kontrolować wiele rozproszonych oddziałów i reagować znacznie szybciej na zachodzące na polu walki zmiany. Tutaj szyfrowanie było jeszcze istotniejsze, gdyż przechwycenie meldunków przez wroga mogło kosztować życie wielu ludzi. Powstawały zatem liczne szyfry polowe, nieraz oparte na pomysłach kryptologów amatorów. Wbrew pozorom opracowanie dobrego szyfru polowego nie było prostym zadaniem. Musiał on bowiem być nie tylko trudny do złamania, ale również prosty w implementacji. Podczas bitwy nie było czasu na przeprowadzanie wielu skomplikowanych obliczeń i przekształceń, a nieodłączny w takiej sytuacji stres mógł być przyczyną błędów w szyfrowaniu. Dobry szyfr polowy musiał zatem być prosty i skuteczny zarazem.

Kolejny rozkwit rozmaitych metod i technologii kryptograficznych przyniosła I wojna światowa. Oprócz telegrafu w powszechnym użyciu było już także radio, co zwiększało potencjał komunikacyjny, wymuszając jednocześnie większą dbałość o ochronę przekazywanych informacji. W tym ostatnim wypadku do podsłuchania przekazu nie trzeba już było uzyskiwać dostępu do linii telegraficznej — wystarczyło prowadzić nasłuch na odpowiedniej częstotliwości. Należało zatem liczyć się z faktem, iż każda wysłana w ten sposób informacja trafia w ręce wroga i może być odczytana, jeśli chroniący ją szyfr nie jest wystarczająco silny. Po obu stronach frontu pracowały więc całe sztaby ludzi prowadzących regularną kryptograficzną wojnę. Warto wspomnieć choćby brytyjski *Pokój 40*, którego członkowie, łamiąc niemieckie szyfry, otworzyli swoim wojskom drogę do wielu spektakularnych zwycięstw, czy przytoczoną we wstępie historię złamania szyfru ADFGX.

Był to również okres wprowadzania licznych książek kodowych w komunikacji między oddziałami na froncie. Taki sposób zabezpieczania łączności miał jednak tę wadę, iż przechwycenie jednej z nich kompromitowało cały system. W związku z tym w razie groźby pojmania w pierwszej kolejności niszczone posiadane egzemplarze książek kodowych. Czasem jednak któraś z nich wpadała w ręce wroga, co powodowało konieczność opracowania i wysłania do wszystkich oddziałów nowych egzemplarzy. Tymczasem w przypadku dobrego systemu szyfrowania jedynym ryzykiem była utrata klucza.

Powstawały zatem kolejne szyfry i kody, a kryptografia stawała się coraz bardziej popularna, jednak z naukowego punktu widzenia nie dokonano wówczas żadnego istotnego przełomu. Prawdziwie rewolucyjne zmiany przynieść miała dopiero kolejna wojna.

1.4. Kryptografia II wojny światowej

Niewiele osób zdaje sobie sprawę, że to właśnie potrzeby kryptoanalityków okresu II wojny światowej doprowadziły do zaprojektowania i skonstruowania pierwszego komputera. Przyczyna była dość pragmatyczna — łamanie szyfrów stało się bardzo skomplikowane obliczeniowo i konieczne stało się odciążenie kryptoanalityków z wykonywania żmudnych przeliczeń. Istnienie takiej maszyny przez długie lata objęte było tajemnicą wojskową, a oficjalnie za pierwszy komputer jeszcze do niedawna uznawano ENIAC. Duży wpływ na jej powstanie miał wkład polskich naukowców, ale — jak mawia pewien znany historyk — nie sprzedajmy faktów.

1.4.1. Enigma i Colossus

Wszystko zaczęło się od zastosowania przez niemiecką armię nowej wirnikowej maszyny szyfrującej — słynnej Enigmy (patrz rysunek 1.8).

Rysunek 1.8.

Enigma



Wywiad aliantów znał schemat zarówno cywilnej, jak i wojskowej wersji niemieckiej maszyny jeszcze przed wojną, jednak naukowcy uznali, że zastosowany w niej algorytm szyfrujący uniemożliwia złamanie szyfru. Istotnie, był on wyjątkowo trudny do kryptoanalizy, jednak głównym powodem niewielkiego zainteresowania Enigmą był panujący w krajach byłej koalicji po zakończeniu I wojny światowej brak poczucia zagrożenia ze strony Niemiec. Tymczasem Polska, która niedawno odzyskała niepodległość,

obawiała się znacznego pogorszenia stosunków z Niemcami, zwłaszcza po dojściu do władzy Adolfa Hitlera. Założono więc biuro szyfrów i podjęto kroki w celu poznania systemu szyfrowania zachodnich sąsiadów.

1.4.1.1. Jak działała Enigma?

Enigma była jedną z popularnych wówczas *maszyn wirnikowych*. Pierwszą taką maszynę skonstruował amerykański wynalazca Eduard Hugo Hebern. Jego wynalazek stanowiły dwie połączone elektryczne maszyny do pisania. Naciśnięcie klawisza w jednej z nich powodowało uruchomienie czcionki w drugiej. Połączenia były zmodyfikowane, a więc wstukiwane litery ulegały zamianie na inne, w rezultacie dając prosty szyfr monoalfabetyczny. Kable przebiegały przez wirniki, które można było obracać, zmieniając tym samym schemat połączeń. W swojej pierwszej maszynie Hebern zamontował pięć walców, każdy o 26 możliwych ustawieniach. Można je było obracać względem siebie, co dawało łącznie 26^5 możliwych schematów połączeń. Odpowiada to szyfrowi Vigenere'a z kluczem o długości około 12 000 000 znaków.

Równoległe do Heberna podobną maszynę wynalazł holenderski uczoney Hugo Aleksander Koch, a także niemiecki inżynier Artur Scherbius. Ten drugi zaproponował swój wynalazek armii niemieckiej już w 1918 roku, jednak wówczas nie spotkał się on z większym zainteresowaniem. Sytuacja zmieniła się po dojściu do władzy Hitlera. W ramach powszechnej modernizacji armii postanowiono wyposażyc niemieckie oddziały w maszyny szyfrujące. Wybór padł na maszynę Scherbiusa.

Enigma oprócz układu wirników wyposażona była w tzw. *walec odwracający*. Dzięki niemu możliwe było wykorzystanie maszyny zarówno do szyfrowania, jak i do deszyfrowania wiadomości. Co ciekawe, o ile z praktycznego punktu widzenia była to niewątpliwą zaletą, o tyle kryptograficznie stanowiło to poważną wadę. Taka konstrukcja powoduje bowiem powstanie *negatywnego wzorca*, czyli innymi słowy, zbioru zasad ograniczających liczbę możliwych kryptogramów. W tym przypadku żadna litera tekstu jawnego nie mogła zostać zaszyfrowana jako ona sama (czyli A w A , B w B itd.). Wiedza o tym okazała się bardzo cenna dla polskich, a później angielskich kryptoanalityków.

Wirniki Enigmy miały zdefiniowany układ połączeń, jednak można je było wkładać do urządzenia w różnej kolejności. Dodatkowo było ich więcej niż przeznaczonych na nie w maszynie gniazd (na początku wojny wirników było osiem). Każdy z nich można było ustawić na 26 sposobów. Podczas szyfrowania pierwszy z wirników obracał się o jedną pozycję z każdą szyfrowaną literą. Jego pełny obrót powodował przesunięcie o jedną pozycję drugiego wirnika, ten z kolei musiał wykonać pełny obrót, zanim o jedną pozycję przesunął się wirnik trzeci itd. Reasumując, o rodzaju zastosowanego przypisania decydowały następujące czynniki:

- ◆ wybór wirników szyfrujących,
- ◆ kolejność wirników w maszynie,
- ◆ początkowe pozycje wirników.

Na rysunku 1.8 widać Enigmę z czterema gniazdami wirników. Po naciśnięciu klawisza odpowiadającego literze tekstu jawnego na znajdującym się powyżej panelu podświetlana była litera tekstu tajnego. Szyfrowanie oparte było na systemie kluczy dziennych determinujących ustawienie wirników. Często już pierwsza litera wiadomości powodowała przesunięcie nie tylko pierwszego, ale również drugiego, a nawet trzeciego wirnika. Szyfrant zapisywał tekst tajny, po czym przekazywał go radiotelegraficznie. Dla uzyskania dodatkowego bezpieczeństwa korzystano również z osobnych kluczy dla poszczególnych depech. Klucz taki był szyfrowany kluczem dziennym na początku wiadomości. Dla pewności powtarzano go dwa razy. Odbiorca deszyfrował klucz depechy, po czym zmieniał zgodnie z nim ustawienia maszyny i odczytywał przekaz.

Wiedza na temat zasad stosowania kluczy dla poszczególnych wiadomości była kolejnym ułatwieniem dla polskich kryptoanalityków. Wiedzieli bowiem, iż na początku każdego kryptogramu znajduje się powtórzona dwukrotnie kombinacja liter, co pozwalało uzyskać cenne informacje na temat klucza dziennego oraz ustawienia wirników. Równie cenne okazało się lenistwo niemieckich szyfrantów, którzy wielokrotnie powtarzali ten sam klucz.

Nie bez znaczenia była również niemiecka pedantyczność i sformalizowany charakter nadawanych depech. Komunikaty zaczynały się i kończyły w identyczny sposób, zawierały również liczne powtórzenia samej treści. Innymi słowy, niemieccy szyfranci byli bardzo przewidywalni. Dawało to dodatkowe informacje na temat zawartych w depechy wyrazów i zwrotów.

1.4.1.2. Cyklometr i Bomby

W roku 1927 polskie służby celne przechwyciły jeden z egzemplarzy Enigmy wysłany do niemieckiej firmy w charakterze zaopatrzenia. Polacy zakupili później kolejne cywilne egzemplarze maszyny. Pomogły one w poznaniu zasad działania ich wojskowych odpowiedników. Rozpracowywaniem niemieckiego szyfru zajmowali się trzej naukowcy — Marian Rejewski, Henryk Zygałski i Jerzy Różycki. Dodatkową pomocą były dla nich dane udostępnione przez francuski wywiad. We Francji uznano Enigmę za niemożliwą do złamania, materiały te nie miały zatem dla francuskich naukowców większej wartości.

Niemcy ciągle doskonalili Enigmę (na przykład dodając kolejne wirniki), przez co łamanie szyfru stawało się coraz trudniejsze. Przede wszystkim rosła liczba koniecznych obliczeń. W końcu polscy matematycy postanowili zaprojektować specjalną maszynę, której zadanie polegałoby wyłącznie na wyszukiwaniu typowych permutacji występujących podczas szyfrowania za pomocą niemieckiej maszyny. Nie była to więc maszyna szyfrująca ani deszyfrująca, a jedynie narzędzie wspomagające obliczenia wykonywane podczas łamania szyfru. Urządzeniu nadano nazwę *Cyklometr*.

Szyfranci armii niemieckiej ustawicznie zwiększali złożoność algorytmu szyfrującego używanego w Enigmie i wkrótce Cyklometr nie był już w stanie wykonywać odpowiedniej liczby obliczeń. Dlatego skonstruowano nowe urządzenia obliczeniowe mające wspomagać kryptoanalizę szyfrów Enigmy. Urządzenia te nazwano *Bombami*.

Polski wywiad udostępnił Anglikom wyniki badań nad Enigmą w roku 1939. Jeszcze przed rozpoczęciem wojny polscy naukowcy (wraz z ich Bombami) zostali przewiezieni do Anglii. Tam badania były kontynuowane w słynnym Bletchley Park. Niestety, z niejasnych przyczyn polscy kryptoanalitycy nie zostali dopuszczeni do prac prowadzonych w tym miejscu. Powierzano im mniej istotne zadania, a z istnienia wielkiego ośrodka kryptoanalitycznego nawet nie zdawali sobie sprawy.

1.4.1.3. Bletchley Park

Centrum kryptoanalityczne w Bletchley Park powstało w wyniku poszerzenia personelu utworzonego w czasie I wojny światowej *Pokoju 40*. Początkowo zatrudniano tam głównie filologów i lingwistów, jednak po spektakularnym sukcesie trzech polskich matematyków postanowiono poszerzyć profil wykształcenia pracowników. Nowo zatrudnionych kierowano do Rządowej Szkoły Kodów i Szyfrów (GC&CS), a ta znajdowała się właśnie w ulokowanym w Buckinghamshire Bletchley Park. Znajdujący się tam niewielki pałacyk stał się brytyjskim centrum łamania szyfrów. W miarę przybywania nowego personelu w otaczających go ogrodach dobudowywano kolejne baraki i poszerzano specjalizację poszczególnych działów. Wkrótce podział ten w naturalny sposób wiązał się z przynależnością do określonego baraku. Na przykład barak 8 specjalizował się w kryptoanalizie depesz niemieckiej marynarki wojennej.

Po opanowaniu polskich metod kryptoanalitycznych specjaliści z Bletchley Park szybko zaczęli opracowywać własne techniki kryptoanalityczne. Jednym z najwybitniejszych pracowników centrum był Alan Turing. Opierając się na analizie archiwalnych kryptogramów, doszedł on do wniosku, iż często możliwe jest przewidzenie fragmentów depesz na podstawie ogólnych informacji na ich temat. Jeśli kryptoanalityk wie, iż w tekście musi się pojawić dany wyraz, może z dużym prawdopodobieństwem ustalić jego pozycję, korzystając z zasady negatywnego wzorca. Jak pamiętamy, żadna litera nie mogła zostać przekształcona w wyniku szyfrowania w nią samą, co eliminuje bardzo wiele potencjalnych pozycji wyrazu w tekście. Kryptoanalityk przesuwiał pasek z wyrazem lub zwrotem pod treścią kryptogramu, analizując powstające w pionie pary liter. Pozycję można było odrzucić, jeśli dawała się wyróżnić chociaż jedna para identycznych liter. Spójrzmy na rysunek 1.9.

Kryptoanalityk zakłada w tym przypadku, iż gdzieś w kryptogramie znajduje się słowo „angriff” (niem. atak). Przykłada zatem pasek z tym wyrazem pod kryptogramem. W pozycji początkowej pojawia się para liter F. Można ją zatem odrzucić, gdyż jak pamiętamy, żadna litera nie mogła zostać zaszyfrowana jako ona sama. Po pierwszym przesunięciu pojawia się z kolei para liter G. Oznacza to, iż również na tej pozycji nie może się znajdować szukane słowo. Kolejne przesunięcie daje aż dwie pary takich samych liter (A i I). Dopiero za czwartym razem udaje się znaleźć miejsce, gdzie (teoretycznie) mógłby się znajdować poszukiwany wyraz. Kolejne dwa przesunięcia również nie dadzą pozytywnego wyniku ze względu na znajdującą się na pozycji jedenastej w kryptogramie literę F, jednak przesunięcie szóste ujawni następną możliwą pozycję wyrazu w kryptogramie (nie pojawiają się żadne pary takich samych liter).

Rysunek 1.9.
Kryptoanaliza
Enigmy oparta
na negatywnym
wzorcu

Pozycja początkowa:

Kryptogram:	G	S	A	G	K	F	I	H	U	B	F	O	V	D
Badany wyraz:	A	N	G	R	I	F	F							

Pierwsze przesunięcie:

Kryptogram:	G	S	A	G	K	F	I	H	U	B	F	O	V	D
Badany wyraz:		A	N	G	R	I	F	F						

Drugie przesunięcie:

Kryptogram:	G	S	A	G	K	F	I	H	U	B	F	O	V	D
Badany wyraz:			A	N	G	R	I	F	F					

Trzecie przesunięcie:

Kryptogram:	G	S	A	G	K	F	I	H	U	B	F	O	V	D
Badany wyraz:				A	N	G	R	I	F	F				

Turing udoskonalił również Bomby, przystosowując je do zmieniającej się struktury niemieckiego szyfru i wprowadzając własne poprawki dotyczące zarówno efektywności działania, jak i zastosowanych algorytmów. Na dobrą sprawę skonstruował on więc zupełnie nowe urządzenia, choć oparte na pomysły polskiego kryptoanalityka. Maszyny wykorzystywano do poszukiwania ustawień wirników, które przekształcałyby podany wyraz w określony kryptogram. Ogólnie więc metodyka łamania Enigmy opierała się na wyszukiwaniu prawdopodobnych wyrazów w tekście, aby następnie ustalić wartości klucza na podstawie tak uzyskanej relacji „tekst jawny — kryptogram”.

Dzięki przeprowadzonej przez Turinga analizie niemieckiego szyfru oraz udoskonalonym przez niego Bombom możliwe było dalsze odczytywanie niemieckich przekazów radiowych mimo rosnącej złożoności stosowanych szyfrów. Warto wspomnieć, iż tak naprawdę w niemieckiej armii funkcjonowało kilka różnych kryptosystemów — inny szyfr miała na przykład marynarka, a nieco inny — siły lądowe. Stosowane były inne wirniki i modele Enigmy, a i sami szyfranci cechowali się różnym stopniem profesjonalizmu. Tym niemniej z większym lub mniejszym trudem pracownicy Bletchley Park dzień w dzień odkrywali przed alianckim dowództwem zamiary i sekrety niemieckiej armii.

1.4.1.4. Colossus

W Bletchley Park nie zajmowano się jedynie Enigmą. Był to co prawda najpopularniejszy, ale nie jedyny szyfr niemiecki. Do wymiany wiadomości między najwyższymi rangą wojskowymi Trzeciej Rzeszy używano tzw. *przystawki szyfrującej*. Było to urządzenie opracowane w firmie Lorenz. Wykorzystywało ono kod opracowany przez francuskiego wynalazcę J.M.E. Baudota. W kodzie tym każdy znak reprezentowany był w systemie dwójkowym z wykorzystaniem taśmy perforowanej. Jedyne odpowiadała dziura w taśmie, a zeru — jej brak. Przystawka odczytywała jednocześnie dwie

taśmy (jedna zawierała tekst jawny, a druga klucz), wykonując na odczytanych wartościach operację dodawania bez przenoszenia reszt (innymi słowy, dodawania modulo 2 — patrz rozdział 2.). Wynik zapisywany był na trzeciej taśmie.

Ten system szyfrowania był o wiele bardziej wyszukany niż stosowany w Enigmie, jednak i tutaj Anglicy odnieśli sukces. Po raz kolejny trzeba było wykorzystać maszyny do przeprowadzania niezbędnych obliczeń. W tym wypadku Bomby już nie wystarczały. Należało skonstruować nowe urządzenie, operujące na podobnej zasadzie jak niemiecka przystawka. Tak powstał Colossus.

Colossus opierał się na teoretycznym modelu opracowanym przez Alana Turinga. W odróżnieniu od Bomb, które były urządzeniami elektromechanicznymi, był urządzeniem elektronicznym. Zawierał półtora tysiąca lamp (dwa i pół tysiąca w późniejszych modelach) i potrafił zapamiętywać dane do dalszego przetwarzania. Czyniło to z niego pierwsze urządzenie, które można nazwać komputerem. Pierwszy model Colossusa oddano do użytku w roku 1943, a więc trzy lata przed słynnym komputerem ENIAC. Ponieważ jednak jego istnienie owiane było tajemnicą wojskową, świat dowiedział się o nim dopiero w roku 1975, po odtajnieniu dotyczących projektu akt.

Wkład alianckich kryptoanalityków w przebieg II wojny światowej był ogromny. Niemcy nie wierzyli, iż można złamać szyfr Enigmy, a tymczasem każdego dnia już po kilku godzinach od zmiany klucza pracownicy Bletchley Park odczytywali pierwsze kryptogramy i przesyłali je do dowództwa. Możliwość poznania zamiarów wroga była ogromnym atutem, o niczym jednak nie przesądzała. Podobnie jak w całej historii tajemnego pisma z odczytanego szyfru należało jeszcze zrobić odpowiedni użytek. Wiedzy tak zdobytej nie można było też nadużywać, by nie wzbudzić u Niemców podejrzeń, że ich system został skompromitowany.

Przesadą byłoby twierdzić, iż to kryptoanalitycy wygrali wojnę z Trzecią Rzeszą. Niemniej jednak gdyby nie ludzie tacy jak Rejewski czy Turing, z pewnością potwałaby ona kilka lat dłużej. Hitler zdążyłby użyć pocisków V1 i V2, zginęłyby również kolejne setki tysięcy ludzi. Bardzo możliwe, iż II wojna światowa zakończyłaby się dopiero po zrzuceniu bomb atomowych na Niemcy.

1.5. Era komputerów

Zastosowanie komputerów zasadniczo zmieniło dotychczasowe sposoby szyfrowania. Po pierwsze, proces szyfrowania przebiegał teraz szybciej i mógł się opierać na znacznie bardziej skomplikowanym algorytmie. Należy pamiętać, że mechaniczne maszyny szyfrujące ograniczały złożoność algorytmu poprzez samą swoją konstrukcję. W przypadku komputerów ograniczenie to zniknęło, ponieważ można było zasymulować dowolnie skomplikowane urządzenie. Innymi słowy, teraz można było szyfrować wiadomości przy użyciu „wirtualnych” szyfratorów, których fizyczna konstrukcja byłaby niemożliwa do wykonania.

Ostatnia, najważniejsza zmiana, jaka nastąpiła dzięki zastosowaniu komputerów, dotyczyła poziomu szyfrowania. Do tej pory odbywało się ono na poziomie liter. Oparte na elektronicznych przełącznikach maszyny operowały jedynie na liczbach dwójkowych. Spowodowało to przejście z szyfrowania liter i znaków na szyfrowanie ciągów zer i jedynek, które w systemie komputerowym służą do zapisu danych. Wcześniej należało ustalić reguły konwersji znanych nam znaków na system binarny. Stąd też w latach sześćdziesiątych opracowano kod ASCII.

Liczby w kodzie ASCII można z łatwością przedstawić w postaci binarnej, co umożliwia ich zapis w komputerze. Po zapisaniu wiadomości w postaci dwójkowej można przejść do szyfrowania, które zasadniczo nie różni się od procesu szyfrowania w erze przedkomputerowej. Nadal podstawową metodą jest przestawianie elementów zapisanej wiadomości według określonego klucza i algorytmu, tak by dla osoby postronnej nie miały one większego sensu — z tą różnicą, że tutaj podstawowym elementem, na którym dokonuje się operacji szyfrowania, jest pojedynczy bit, a nie znak, jak to miało miejsce wcześniej. Jak wiadomo, aby zapisać jeden znak, potrzeba jednego bajta, czyli ośmiu bitów.



ASCII (skrót od ang. *American Standard Code for Information Interchange*) jest zestawem kodów, standardowo z zakresu 0 – 127 (dziesiętnie), przyporządkowanych przez ANSI (Amerykański Instytut ds. Standardów) poszczególnym znakom alfanumerycznym (litery alfabetu angielskiego i cyfry), znakom pisarskim oraz sterującym (typu nowa linia). Na przykład litera „a” jest zakodowana przy użyciu liczby 97. Ponieważ ASCII jest standardem 7-bitowym, a większość komputerów operuje na wielkościach 8-bitowych (bajtach), pojawił się również rozszerzony kod ASCII. Dzięki niemu możliwe stało się wprowadzenie znaków narodowych do stosowanego na danym komputerze alfabetu. W związku z tym obecnie wykorzystywane w ramach kodu ASCII znaki mogą się różnić w zależności od komputera. Aby uniknąć tego typu różnicowania, opracowano standard UNICODE składający się z 65 536 znaków, dzięki czemu możliwe jest definiowanie znaków w wielu różnych językach.

1.5.1. DES

Kryptologia komputerowa najszybciej rozwijała się w Stanach Zjednoczonych. Powstało tam wiele systemów kryptograficznych, jednak ze względu na specyfikę amerykańskiego prawa wkrótce pojawiła się konieczność ustalenia powszechnie obowiązującego standardu szyfrowania. W 1973 roku z propozycją takiego uniwersalnego systemu o nazwie Demon wystąpił Horst Feistel, niemiecki emigrant, który przybył do USA w 1934 roku. Nazwa wywodziła się od słowa *Demonstration*, a jej skrócona forma spowodowana była ograniczoną długością nazw plików w używanym przez twórcę standardu systemie. Później Demon został „przechrzczony” na Lucyfera (ang. *Lucifer*), co stanowiło swoistą grę słów (angielskie słowo *cipher* oznacza szyfr). Lucyfer był szyfrem blokowym, a więc jako dane wejściowe przyjmował bloki danych o ustalonej długości, zaś na wyjściu podawał bloki kryptogramu o takiej samej długości. Innymi słowy, podstawową jednostką przetwarzania nie były tu pojedyncze bity czy bajty, a całe bloki danych (patrz podrozdział 3.2). Feistel utworzył kilka wersji tego szyfru; najbardziej znana opierała się na kluczu 128-bitowym, niezwykle odpornym na ataki metodą pełnego przeglądu (sprawdzania wszystkich kluczy po kolei — patrz podrozdział 3.1).

Aby Lucyfer został przyjęty jako standard, musiał najpierw zostać przedłożony Narodowej Agencji Bezpieczeństwa (NSA). Organizacja ta starała się cały czas kontrolować pojawiające się na rynku narzędzia kryptograficzne. Jej głównym celem było ograniczanie tych zabezpieczeń w taki sposób, aby mogły być złamane przez rządowych kryptoanalityków, gdyby zachodziło podejrzenie, że zabezpieczone nimi dane mogą stanowić zagrożenie dla bezpieczeństwa państwa. Podobnie było w przypadku kryptosystemu Feistela.

Przestrzeń kluczy systemu Feistela została przez NSA bardzo ograniczona — długość klucza zmniejszono ze 128 do 56 bitów. Każdy kolejny bit długości klucza powoduje podwojenie tej przestrzeni, a tym samym podwojenie bezpieczeństwa systemu, przynajmniej w odniesieniu do *ataku wyczerpującego* (patrz podrozdział 3.1). Wynika z tego, że skrócenie klucza o jeden bit implikuje 50-procentowy spadek bezpieczeństwa szyfru (w tym kontekście można sobie wyobrazić, jak drastyczny spadek bezpieczeństwa powoduje redukcja klucza ze 128 do 56 bitów). System ten nadal był wystarczająco bezpieczny dla sektora prywatnego, jednak ograniczenie wprowadzone przez NSA spotkało się ze zdecydowanym sprzeciwem pozarządowych środowisk kryptograficznych, które zdawały sobie sprawę z możliwości złamania tego szyfru przez agencję.

Ostatecznie jednak pomimo licznych protestów 23 listopada 1976 roku 56-bitowa wersja szyfru Lucyfer Feistela została oficjalnie przyjęta jako standard szyfrowania danych DES (skrót od ang. *Data Encryption Standard*).

1.5.2. Narodziny kryptografii asymetrycznej

Jedną z osób, które dążyły do przełamania kryptograficznego monopolu NSA, był Whit Diffie. Diffiemu marzył się system kryptograficzny wolny od problemu dystrybucji klucza. Jego koncepcja opierała się na założeniu dotąd uważanym za technicznie niemożliwe do zrealizowania — klucz szyfrujący miał być powszechnie dostępny. Jedną z niepodważalnych zasad kryptografii była zasada tajności tego klucza i używania go zarówno do szyfrowania, jak i deszyfrowania wiadomości. Diffie zaprojektował natomiast system oparty na dwóch kluczach szyfrujących. Para takich kluczy miała być tworzona dla każdego użytkownika jego systemu. Jeden z nich (klucz publiczny) miał służyć do szyfrowania wiadomości wysyłanych do użytkownika i miał być powszechnie dostępny. Drugi (klucz tajny) miał być wykorzystywany do dekryptaży wiadomości zaszyfrowanych przy użyciu pierwszego klucza. Za pomocą drugiego klucza można było również szyfrować wiadomości. W takim przypadku dekryptaż miał umożliwić klucz publiczny. Dzięki takiej kombinacji możliwe stawało się uwierzytelnianie nadawcy wiadomości elektronicznej, ponieważ tylko osoba posiadająca klucz tajny mogła zaszyfrować dokument tak, aby dało się go odczytać przy użyciu klucza publicznego. Zaszyfrowanie wiadomości kluczem prywatnym stanowiło więc jednocześnie formę elektronicznego podpisu.

Sama teoria kluczy nie wystarczała i konieczne stało się opracowanie odpowiednich podstaw matematycznych, możliwych do zaimplementowania w językach programowania. Problem ten rozwiązał współpracownik Whita Diffiego — Marty Hellman. Hellman i Diffie wspólnie opracowali system matematyczny oparty na funkcji jednokierunkowej bazującej na tzw. logarytmowaniu dyskretnym (patrz rozdział 2.). System ten znany jest obecnie jako algorytm Diffiego-Hellmana.

System Diffiego i Hellmana nie tylko rozwiązał problem dystrybucji klucza, ale również zapoczątkował technologię elektronicznego uwierzytelniania użytkowników. Jego metoda rodzi bowiem jeszcze jedną istotną implikację, którą jest niemożliwość wyparcia się swojego cyfrowego podpisu. Skoro wiadomo, że technicznie niemożliwe jest, aby osoba nieznająca tajnego klucza wygenerowała poprawny podpis, jego właściciel nie może się wyprzeć swojego udziału w poświadczanej nim transakcji. Ta właściwość stała się podstawą technologii podpisu elektronicznego.

1.5.3. RSA

Idea kryptosystemu z kluczem publicznym została rozwinięta przez trzech naukowców z uniwersytetu w Stanford — Rona Rivesta, Adi Shamira i Leonarda Adlemana. Podobnie jak Diffie i Hellman poświęcili oni dużo czasu na znalezienie matematycznego wzoru, który pozwalałby zrealizować ideę szyfrowania przy użyciu pary kluczy w praktyce. Przełomowego odkrycia podczas tych badań dokonał Rivest. Polegało ono na zastąpieniu algorytmu Diffiego-Hellmana jego własnym systemem obliczeń.

Koncepcja Rivesta opiera się na problemie rozkładu dużych liczb na czynniki pierwsze. Klucz publiczny generowany jest przez pomnożenie przez siebie dwóch dużych, losowo wybranych liczb pierwszych. Następnie wybierana jest kolejna duża liczba o określonych właściwościach — stanowi ona klucz szyfrujący. Klucz publiczny tworzony jest na podstawie klucza szyfrowania oraz wspomnianego iloczynu liczb pierwszych. Klucz prywatny można łatwo obliczyć, jeśli zna się liczby pierwsze tworzące iloczyn zastosowany przy tworzeniu klucza publicznego. Są one znane właścicielowi pary kluczy, natomiast kryptoanalityk może je uzyskać jedynie dzięki rozwiązaniu problemu faktoryzacji dużych liczb. Matematyczne szczegóły tej metody opisane zostały w rozdziale 2.

Algorytm opracowany przez Rivesta i jego współpracowników został wkrótce opatentowany pod nazwą RSA (od pierwszych liter nazwisk wynalazców). Agencja Bezpieczeństwa Narodowego próbowała zapobiec upowszechnieniu się tego standardu szyfrowania. Zaczęto wywierać naciski na NIST (skrót od ang. *National Institute of Standards and Technology*), aby przyjął jako obowiązujący w USA standard program DSA (skrót od ang. *Digital Signature Algorithm* — patrz dodatek B).

W wielu miejscach DSA powielał rozwiązania z RSA, jednak był systemem znacznie słabszym: „Pod względem czysto technicznym było jasne, że DSA był gorszy od RSA. Algorytm ten był, jak to wyłożył jeden z obserwatorów, »dziwacznym standardem«, o wiele wolniejszym od systemu RSA, jeśli chodzi o weryfikowanie podpisów (choć szybszym w podpisywaniu wiadomości), trudniejszym do wdrożenia i bardziej skomplikowanym. I nie umożliwiał szyfrowania. System opracowany przez rząd oferował jednak pewną korzyść w porównaniu z RSA [...]. Był bezpłatny”⁴.

Brak możliwości szyfrowania był poważną wadą techniczną systemu DSA, jednak poprzez rozprowadzanie go w charakterze darmowego oprogramowania NSA ustanowiła silną konkurencję dla standardu RSA (tym bardziej że DSA stał się ustawowo ustalonym standardem).

⁴ Levy S., *Rewolucja w kryptografii*, Wydawnictwa Naukowo-Techniczne, Warszawa 2002.

1.5.4. PGP

Działalność NSA zmierzająca do ograniczenia dostępu do kryptografii powodowała konflikt między rządem a kryptoanalitykami sektora publicznego. Ci drudzy szukali sposobu na wprowadzenie kryptografii do powszechnego użytku. Zdawali sobie sprawę, że upowszechnienie komputerów, a zwłaszcza internetu, znacznie ograniczy prywatność zwykłych użytkowników, jeśli nie zagwarantuje się im odpowiednich zabezpieczeń. Z tego powodu Phil Zimmerman, programista z Florydy, rozpoczął projektowanie własnego systemu kryptograficznego. Nadał mu nazwę PGP (skrót od ang. *Pretty Good Privacy* — całkiem niezła prywatność). Początkowo miał to być kolejny komercyjny pakiet szyfrujący. Zamiany Zimmermana zmienił projekt opracowanej w Komisji Prawnej Senatu ustawy nr 266 z 24 stycznia 1991 roku. Zgodnie z nią dostawcy usług komunikacji elektronicznej, a także wytwórcy sprzętu komunikacyjnego powinni zapewnić rządowi możliwość uzyskania jawnej treści komunikacji między użytkownikami, jeśli uzyska on do tego upoważnienie prawne.

Zimmerman zdał sobie sprawę, że po wejściu ustawy w życie nie będzie już mógł wprowadzić na rynek swojego programu. Postanowił więc udostępnić go jak największej liczbie osób, zanim posługiwanie się nim przestanie być prawnie dozwolone. W tym celu postanowił wykorzystać internet. Skontaktował się ze swoim przyjacielem, Kellym Goenem, powierzając mu zadanie umieszczenia PGP 1.0 na powszechnie dostępnych sieciowych witrynach znajdujących się w USA. Wyznaczonego dnia Goen przy użyciu telefonów publicznych, laptopa i telefonu komórkowego wprowadził PGP do internetowych serwerów plików. Mógł go stamtąd ściągnąć każdy, kto miał dostęp do internetu, nie tylko w USA.

Phil Zimmerman zagwarantował prywatność każdemu, kto miał dostęp do jego programu. Ustawa 266 ostatecznie nie została wprowadzona w życie ze względu na protesty środowisk walczących o swobody obywatelskie. Była ona także niemożliwa do wyegzekwowania ze względu na powszechną dostępność pakietu PGP. Sam Zimmerman natomiast został pozwany do sądu przez RSA Data Security z powodu naruszenia praw patentowych algorytmu RSA (Zimmerman użył go w swoim programie, nie wiedząc, że RSA jest objęty ochroną patentową). Firma domagała się od niego wycofania PGP. Ostatecznie doszło do ugody, w myśl której autor PGP 1.0 zaprzestał dalszej dystrybucji swojego programu, a firma nie wniosła oskarżenia.

W 1993 roku Zimmerman został oskarżony o złamanie prawa przez rząd USA. Zdaniem FBI, udostępniając publicznie silny system kryptograficzny, Zimmerman dostarczył wrogim państwom oraz terrorystom narzędzie do walki z USA. Dochodzenie umorzono po trzech latach. Zresztą wydanie wyroku skazującego i tak nic by już nie zmieniło. Wycofanie PGP z powszechnego obiegu było niemożliwe, tak jak niemożliwe jest pełne kontrolowanie przepływu informacji w internecie.

Prawne restrykcje wobec twórców kryptograficznego oprogramowania były dość częstym środkiem stosowanym przez rząd USA w celu ograniczenia dostępu do kryptografii. Eksport oprogramowania z zakresu tzw. silnej kryptografii (opartej na większej niż ustalona długości klucza) był prawnie zakazany i traktowany na równi z nielegalnym eksportem broni. Sytuacja ta zmieniła się dopiero na początku obecnego stulecia,

kiedy okazało się, że z uwagi na gwałtowny rozwój internetu niemożliwa jest dalsza kontrola nad rozprzestrzenianiem się oprogramowania, a tego typu ograniczenia osłabiają tylko gospodarkę USA. Firmy amerykańskie traciły zagranicznych klientów, gdyż nie były w stanie zagwarantować im takiego samego bezpieczeństwa danych jak firmy spoza USA, nieobjęte ograniczeniami co do długości klucza.

Należy zauważyć, że posunięcie Zimmermana oprócz niewątpliwych korzyści dla zwykłych użytkowników sieci komputerowych mogło również spowodować znaczne szkody. Teraz bowiem każdy mógł zaszyfrować swoje komunikaty w ten sposób, że nikt nie byłby w stanie ich odczytać. Innymi słowy, PGP umożliwia szyfrowanie korespondencji również terrorystom, miłośnikom dziecięcej pornografii i każdemu, kto wykorzystuje komputer do działalności przestępczej. Zimmerman zdawał sobie z tego sprawę, uznał jednak, że upowszechnienie pakietu będzie „mniejszym złem”.

Wkrótce po udostępnieniu PGP program został poddany szczegółowej analizie przez grono specjalistów od kryptografii. Okazało się, że posiada on pewne wady ograniczające jego bezpieczeństwo. Postanowiono więc stworzyć nową wersję programu — PGP 2.0. Zastosowane w niej rozwiązania były stosowane także w późniejszych wersjach. W roku 2010 PGP zostało przejęte za ponad 300 milionów dolarów przez firmę Symantec i zintegrowane z oprogramowaniem szyfrującym firmy.

1.5.5. Ujawniona tajemnica

Wart wspomnienia jest fakt, że technika szyfrowania asymetrycznego została opracowana już w 1969 roku przez Jamesa Ellisa, pracownika angielskiego GCHQ (skrót od ang. *Government Communication Headquarters* — głównej rządowej kwatery łączności), jednak nie mógł on jej upowszechnić ze względu na obowiązującą go tajemnicę państwową. Efekty jego pracy zostały ujawnione dopiero niedawno, dlatego też informacje o nich można znaleźć jedynie w najnowszych opracowaniach z dziedziny kryptografii.

Pomysł Ellisa był bardzo podobny do systemu Whita Diffiego. Również opierał się on na idei pary kluczy szyfrujących. Podobnie jak Diffie, również Ellis stworzył teoretyczne fundamenty systemu szyfrowania asymetrycznego i także nie był w stanie samodzielnie opracować wzoru matematycznego, na podstawie którego taki system mógłby funkcjonować. Przez kilka następnych lat wielu matematyków GCHQ zmagало się z tym problemem, jednak żaden nie znalazł rozwiązania. Sytuacja zmieniła się, kiedy zadanie to powierzono nowo zatrudnionemu matematykowi, Cliffordowi Cocksowi.

Klucz publiczny opracowany przez Cocksą opierał się, podobnie jak w systemie RSA, na iloczynie dwóch dużych liczb pierwszych. Iloczyn ten miał być dostępny dla każdego, kto chciał wysłać zaszyfrowaną wiadomość do odbiorcy. Cocks opracował następnie formułę matematyczną, dzięki której wiadomość zaszyfrowana przy użyciu klucza publicznego mogła być odszyfrowana jedynie pod warunkiem znajomości oryginalnych liczb pierwszych. Formuła zastosowana przez angielskiego matematyka była niemal identyczna z tą, którą zastosował Ron Rivest. Innymi słowy, Ellis i Cocks opracowali system, który trzy lata później został ponownie odkryty przez trzech matematyków z MIT i wprowadzony do powszechnego użytku jako RSA.

1.5.6. Upowszechnienie kryptografii

Spopularyzowanie kryptografii asymetrycznej doprowadziło do przełamania kryptologicznego monopolu NSA. Kryptografia cieszyła się coraz większym zainteresowaniem w sektorze publicznym, co utrudniało wszelkie próby ograniczania dostępu do opartych na niej zabezpieczeń. Mimo to agencja nadal starała się kontrolować rozwój badań nad nowymi technologiami z tego zakresu. Naukowcy, którzy nie pracowali dla NSA, zobowiązani byli przedstawiać swoje artykuły przed ich publikacją specjalnie w tym celu utworzonej komisji. Jej zadaniem było wyławianie treści potencjalnie niebezpiecznych dla bezpieczeństwa narodowego. Na tym etapie była to już jednak walka skazana na niepowodzenie.

Na początku lat osiemdziesiątych zainaugurowano serię konwentów o nazwie Crypto, które odąd odbywały się corocznie. Spotykali się na nich kryptolodzy z całego świata w celu wymiany doświadczeń i wyznaczania nowych kierunków badań. O ile jednak w tym przypadku NSA mogła się jeszcze pokusić o próby wpływania na prezentowane treści, o tyle prężnie rozwijający się internet stanowił medium, którego nawet agencja nie mogła sobie podporządkować. Raz zamieszczony w sieci artykuł lub program był ściągany i kopiowany przez niezliczone rzesze użytkowników, co uniemożliwiała jego wycofanie, nawet gdyby udało się przekonać do tego samego autora. Najlepszym przykładem jest tu PGP. Solą w oku NSA była w tamtym czasie grupa ludzi określających się jako *Szyfropunki*. Pisali oni programy, artykuły i opracowania z dziedziny kryptografii, a następnie udostępniali je w sieci. Oczywiście nikt tu nie pytał NSA o zgodę. Grupka ta doskonale odzwierciedlała nastroje panujące w amerykańskim społeczeństwie, coraz bardziej wzburzonym rządowymi próbami kontrolowania przepływu informacji i ingerowania w prywatność obywateli. Wkrótce więc kryptograficzni anarchiści stali się ulubieńcami mediów i symbolem walki o swobody obywatelskie.

Kolejnym etapem kryptograficznych zmagania było zapoczątkowanie przez NSA w 1993 roku projektu *EES* (skrót od ang. *Escrowed Encryption Standard*) opartego na chipach *Capstone* i *Clipper*. Miały to być powszechnie dostępne urządzenia umożliwiające szyfrowany przekaz danych. Capstone służyć miał posiadaczom komputerów przenośnych, natomiast Clipper był przeznaczony do montowania w telefonach i faksach. Każde z urządzeń miało na trwałe zaimplementowany klucz szyfrowania, do którego służby rządowe mogły uzyskać dostęp, gdyby zachodziło podejrzenie, iż właściciel chipu prowadzi działalność niezgodną z prawem. Tak przynajmniej wyglądało to w teorii, nie da się jednak ukryć, że deponowanie kluczy dawało organizacjom rządowym ogromne pole do nadużyć. W rezultacie nie trzeba było długo czekać na protesty środowisk liberalnych. Były one tym bardziej uzasadnione, iż zastosowany w obu chipach algorytm Skipjack opracowany został w NSA i dane na jego temat były tajne. W rezultacie agencja miała monopol na produkcję obu urządzeń, a to budziło uzasadnione poniekąd podejrzenia co do prawdziwych intencji rządu. Ostatecznie NSA zdecydowała się upublicznić algorytm, jednak to jedynie pogorszyło sprawę. Jego kryptoanaliza przeprowadzona w środowiskach prywatnych wykazała bowiem, iż po pierwsze, nie jest on tak mocny jak AES, a po drugie, możliwe jest wykorzystanie go bez deponowania kluczy. Znacznie osłabiło to rozwój nowej technologii i ostatecznie udaremniło jej upowszechnienie na większą skalę.

Obecnie każdy może bez przeszkód korzystać z dobrodziejstw kryptografii. Nie jest ona domeną tylko i wyłącznie organizacji rządowych, choć z całą pewnością dysponują one pod tym względem znacznie większym potencjałem z uwagi na to, że skupiają najlepszych fachowców i inwestują w badania nad nowymi technologiami. Trudno powiedzieć, jakie dokładnie są możliwości agencji takich jak NSA w odniesieniu do kryptografii, z pewnością jednak możliwości „zwykłego śmiertelnika” zwiększyły się w ciągu ostatniego półwiecza niepomierne.

Skorowidz

A

- Achterbahn 128/80, 112
 - ActiveX, 221
 - adaptive-chosen-plaintext attack, 91
 - Additional Decryption Keys, 175
 - AddRoundKey, 272
 - ADFGVX, 20
 - ADFGX, 16, 20
 - tablica podstawień, 17
 - ADK, 175
 - Adleman Leonard, 41
 - Advanced Encryption Algorithm, 271
 - AES, 165, 271
 - AddRoundKey, 274
 - bezpieczeństwo, 274
 - ByteSub, 272
 - generowanie kluczy, 271
 - rozszerzanie, 272
 - selekcja podkluczy, 272
 - MixColumn, 273
 - operacje w rundzie, 272
 - opis algorytmu, 271
 - pojedyncza runda algorytmu, 272
 - runda zerowa, 272
 - schemat, 272
 - schemat mnożenia macierzy stanu, 274
 - ShiftRow, 273
 - wielkość przesunięcia, 273
 - zależność liczby rund algorytmu od długości klucza, 271
- aes_decrypt(), 236, 237
 - aes_encrypt(), 236
 - AES256, 182
 - Alberti Leone Battista, 24
 - Alert Protocol, 154
 - alfabet
 - identyfikacja, 292
 - algorytm
 - 2DES, 269
 - 3DES, 165, 269
 - a sposoby użytkowania systemu, 241
 - A5, 111
 - Achterbahn 128/80, 112
 - AES, 271
 - AES256, 182
 - Arcfour, 111
 - Asmutha-Blooma, 123
 - bcrypt, 282
 - Blowfish, 239, 282
 - Boyar-Damagarda, 119
 - CAST-128, 165, 280
 - CAST5, 280
 - Chauma, 119
 - DES, 263
 - DES z S-blokami zależnymi od klucza, 270
 - DESX, 270
 - Diffiego-Hellmana, 40, 71
 - DSA, 126, 248, 284
 - E0, 112
 - Euklidesa, 58
 - Gordona, 69
 - Harna-Yanga, 119
 - HC-256/128, 112
 - Helix, 111
 - HMAC, 165
 - IDEA, 259
 - ISAAC, 112
 - Karnina-Greene'a-Hellmana, 122
 - Keccak, 254
 - kryptografia kwantowa, 133
 - LSB, 145
 - Maurera, 69
 - MD5, 243
 - MUGI, 112
 - najmniej znaczącego bitu, 145

- algorytm
 - Nanoteq, 111
 - obliczeniowo bezpieczny, 150
 - Patchwork, 145
 - podpisu, 228
 - PrimeInc, 69
 - próba czasu, 241
 - Rabbit, 112
 - Rambutan, 111
 - RC4, 111
 - Rijndael, 271
 - rozpraszający w podpisach cyfrowych, 246
 - RSA, 287
 - SEAL, 112
 - SHA-1, 246
 - SHA-2, 249
 - SHA-3, 254
 - Skipjack, 44, 115
 - SOBER-128, 112
 - Twofish, 165, 275
 - VMPC, 112
 - w SSH, 165
 - WAKE, 111
 - wektorowy, 122
 - wielomianu interpolacyjnego Lagrange'a, 122
 - wyznaczania wartości skrótu podpisu, 229
 - XPD/KPD, 111
 - z kluczem publicznym, 288
- algorytmy blokowe, 288
- algorytmy strumieniowe, 111
 - A5, 111
 - Achterbahn 128/80, 112
 - Arcfour, 111
 - E0, 112
 - HC-256/128, 112
 - Helix, 111
 - ISAAC, 112
 - MUGL, 112
 - Nanoteq, 111
 - Rabbit, 112
 - Rambutan, 111
 - RC4, 111
 - SEAL, 112
 - SOBER-128, 112
 - VMPC, 112
 - WAKE, 111
 - XPD/KPD, 111
- algorytmy szyfrujące, 259
 - AES, 271
 - Blowfish, 282
 - CAST5, 280
 - DES, 263
 - DSA, 284
 - IDEA, 259
 - pozostałe, 288
 - RSA, 287
 - Twofish, 275
- Al-Kindi, 22
- alternatywa wykluczająca, 72
- American Standard Code for Information Interchange, 39
- analiza
 - częstości, 22, 28, 30
 - Kasickiego, 28
 - różnicowa mocy, 94
 - sygnaturowa, 147
- AND, 72
- ANSI, 39
- Apache, 231
- aproxymacja liniowa, 92
- Arcfour, 111
- arytmetyka modulo, 52
 - RSA, 63
- arytmetyka zegarowej tarczy, 52
- ASCII, 39, 63
- atak
 - algebraiczny, 96
 - algorytmiczny, 96
 - bocznym kanałem, 94
 - geometryczny, 146
 - Man in the Middle, 95, 116, 119
 - Meet in the Middle, 95, 269, 270
 - metodą kluczy powiązanych, 91
 - na algorytmy szyfrujące, 242
 - na wartości skrajne, 102
 - na zredukowaną przestrzeń kluczy, 94
 - pierwszy atak kryptoanalityczny, 22
 - przez podmianę bloku danych, 102
 - przez powtórzenie, 93
 - przez wstawienie bloku danych, 102
 - przez zablokowanie usługi, 93
 - przez zalew SYN, 93
 - słownikowy, 94, 151
 - tablicowy, 238
 - typu DoS, 93
 - typu watermark, 105
 - urodzinowy, 95, 100
 - z wnętrza systemu, 95
 - z wykorzystaniem wirusa, 97
- ataki kryptoanalityczne, 89, 92
 - algebraiczny, 96
 - algorytmiczny, 96
 - analiza różnicowa mocy, 94
 - bocznym kanałem, 94
 - kleptografia, 97
 - kryptoanaliza liniowa, 91, 92
 - kryptoanaliza różnicowa, 91
 - łamanie metodą pełnego przeglądu, 90

łamanie z adaptacyjnie wybranym tekstem jawnym, 91
 łamanie z szyfrogramami, 90
 łamanie z wybranym szyfrogramem, 90
 łamanie z wybranym tekstem jawnym, 90, 91
 łamanie ze znanym tekstem jawnym, 90
 Man in the Middle, 95
 metoda akustyczna, 94
 metoda różnicowej analizy błędów, 95
 metodą kluczy powiązanych, 91
 metody kryptoanalityczne, 89
 na zredukowaną przestrzeń kluczy, 94
 przez powtórzenie, 93
 przez zablokowanie usługi, 93
 słownikowy, 94
 socjotechnika, 97
 urodzinowy, 95
 z wnętrza systemu, 95
 z wykorzystaniem wirusa, 97
 atropa, 141
 autoklucz, 27

B

Babbage Charles, 28
 BBS, 109
 BC, 104
 bcrypt, 238, 239, 282
 bezpieczeństwo baz danych, 240
 bezpieczeństwo systemu kryptograficznego, 82
 atak metodą pełnego przeglądu, 83
 badania kryptoanalityczne, 148
 czynnik ludzki, 153
 hasła, 151
 kontrola dostępu do danych, 153
 najslabszy element, 150
 plan zamknięcia, 149
 pracownicy, 153
 procedury kontra socjotechnika, 152
 reguły udostępniania danych, 152
 system hybrydowy, 150
 testowanie oprogramowania, 148, 149
 wspomaganie technikami niekryptograficznymi, 151
 zasada Kerckhoffa, 148
 bezpieczna powłoka, 162
 bezpieczne przeglądanie poczty, 196
 bezpieczne usuwanie plików, 200
 bezpieczny system, 242
 kryptograficzny, 149
 konstrukcja, 147
 kryteria, 150
 wybór i implementacja, 148
 bezwzględna zawartość informacyjna języka, 85

bezwzględna nadmiarowość języka, 85
 bibliografia, 307
 biblioteka libmccrypt, 232
 biblioteka mccrypt, 232
 szyfry blokowe, 232
 szyfry strumieniowe, 233
 biblioteka OpenSSL, 162
 bijekcja, 62
 BIL gates, 140
 bity pseudolosowe, 108
 Bletchley Park, 36
 Block Chaining, 104
 bloki danych, 98
 rejestr przesuwający, 101, 108
 wektor inicjujący, 99, 105
 bloki szyfrogramu, 98
 Blowfish, 165, 239, 282
 funkcja algorytmu, 283
 ogólny schemat, 283
 opis, 282
 Blum Blum Shub, 109
 błąd krytyczny, 156
 Bomby, 35
 Boolean Integrase Logic, 140
 bramka logiczna procesora, 54
 bramki BIL, 140
 brute-force attack, 90
 ByteSub, 272

C

CA, 127
 Calvert Frank, 298
 Capstone, 44
 CAST-128, 165, 280
 CAST5, 280
 klucze maskujące, 280
 klucze przesunąć, 280
 opis, 280
 przekształcenia, 281
 rundy, 281
 CBC, 99, 107
 CBCC, 104
 centrum certyfikacji, 127, 222
 Certum, 157, 222
 struktura wielopoziomowa, 129
 Certificate Request, 155
 Certificate Signing Request, 158
 Certificate Verify, 156
 certification authority, 127
 Certum, 157, 222
 certyfikat, 127
 dodatkowe parametry, 206
 eksportowanie do pliku, 206

- certifikat
 - eksportowanie na serwer, 206
 - lista odwołań certyfikatów, 130
 - obsługa, 205
 - okres ważności, 130
 - podręczne menu, 207
 - potwierdzenie tożsamości, 207
 - proces certyfikacji, 207
 - przesyłanie pocztą, 206
 - tworzenie, 203
 - unieważniony, 130
- certifikat cyfrowy
 - aktywacja, 222
 - algorytm podpisu, 228
 - algorytm wyznaczania wartości skrótu podpisu, 229
 - centra certyfikacji, 222
 - dotatkowo chroniony klucz prywatny, 223
 - elementy certyfikatu, 228
 - identyfikator wersji certyfikatu, 228
 - identyfikator wystawcy, 229
 - instalacja
 - w Microsoft Outlook, 224
 - w Mozilla Thunderbird, 227
 - w MS Outlook Express, 226
 - klucz publiczny właściciela certyfikatu, 229
 - kopia zapasowa klucza prywatnego, 224
 - lista programów pocztowych, 220
 - nazwa i identyfikator właściciela certyfikatu, 229
 - numer seryjny certyfikatu, 228
 - obsługa, 220
 - obsługa 128-bitowej siły szyfrowania, 221
 - obsługa ActiveX, 221
 - obsługa cookies, 221
 - odcisk palca, 230
 - okres ważności, 229
 - potwierdzenie wprowadzonych danych, 223
 - rozszerzenia alternatywne, 229
 - struktura, 228
 - umowa certyfikacyjna, 223
 - uzyskiwanie, 221, 222
 - uzyskiwanie informacji, 229
 - wybór parametrów generowania kluczy, 223
 - wymagania techniczne, 220
 - zmiana adresu poczty elektronicznej, 224
- certifikat OpenPGP, 205
 - przesłanie certyfikatu do globalnego serwera, 205
 - przesłanie certyfikatu pocztą, 205
 - wykonanie kopii zapasowej kluczy, 205
- certifikat SSL, 157
 - aktywacja certyfikatu, 158
 - CSR, 158
 - Firefox, 160
 - generowanie pary kluczy, 158
 - instalowanie, 159
 - Internet Explorer, 160
 - Netscape Navigator 9.x, 160
 - odcisk palca, 162
 - pełna wersja, 160
 - pobranie, 157
 - potwierdzenie dostępu do domeny, 159
 - sprawdzanie, 160
 - ważność, 162
 - wydanie, 159
 - wyświetlanie, 160
 - żądanie podpisania, 159
- certifikat X.509, 203, 204
 - hasło klucza prywatnego, 204
 - importowanie, 205
 - przesyłanie certyfikatu, 205
- certyfikaty tożsamości, 126
- CFB, 101, 107
- Chadwick John, 305
- Champollion Jean François, 295
- charakterystyki, 91
- chińskie twierdzenie o resztach, 70
- chosen-ciphertext attack, 90
- chosen-plaintext attack, 90
- ciała izomorficzne, 62
- ciało, 61
- ciągi, 254
- cienie, 122
- Cipher Block Chaining, 99
 - with Checksum, 104
- Cipher Feedback, 101
- Cipher Spec Protocol, 154
- ciphertext-only attack, 90
- Claws Mail, 203
- Client Certificate, 155
- Client Hello, 155
- Client Key Exchange Message, 155
- Clipper, 44
- CMC, 105
- Colossus, 33, 37
- cookies, 221
- crypt(), 239
- Crypto, 44
- CSR, 158
 - tworzenie, 159
- CTR, 103, 107, 165
- cyfrowe znaki wodne, 145
- cykl życia klucza, 129
 - rozprowadzanie klucza, 129
 - użytkowanie aktywne, 129
 - użytkowanie pasywne, 129
 - wycofanie, 130
 - zmęczenie, 129
- Cyklometr, 35

częstość występowania liter w języku polskim, 76
 częściowe powtórzenia, 23
 czynnik zaciemniający, 120
 czyszczenie wolnej przestrzeni dyskowej, 193

D

dane certyfikatu w przeglądarce, 161
 DASS, 117
 Data Encryption Standard, 40, 263
 Demon, 39
 denaturacja, 140
 denial of service, 97
 denial-of-service attack, 93
 DES, 39, 236, 263

- bezpieczeństwo, 266
- deszyfrowanie, 268
- klucz szyfrowania, 263
- matematyczna postać szyfrowania, 268
- modyfikacje, 269
- operacje przekształcania klucza, 263
- P-bloki, 267
- permutacja końcowa, 268
- permutacja początkowa, 263
- permutacja rozszerzona, 265
- permutacja z kompresją, 265
- permutacja zwykła, 267
- podział tekstu na bloki, 263
- pojedyncza runda algorytmu, 264
- S-bloki, 266
- skrócenie klucza algorytmu, 83
- szyfrowanie bloków, 263
- wydłużenie ciągu, 265
- z S-blokami zależnymi od klucza, 270

 des_decrypt(), 236
 des_encrypt(), 236
 designated confirmer signatures, 120
 DESX, 270

- wydłużenie efektywnej długości klucza, 271

 deszyfrowanie, 50, 262

- algorytm RSA, 287
- DES, 268
- z wykorzystaniem nukleotydów, 141

 Diffie Whit, 40
 Digital Signature Algorithm, 248
 Digital Signature Standard, 72
 długość klucza, 28
 DNA, 138

- denaturacja, 140
- nukleotyd, 138
- pierścień, 138
- podwójna helisa, 138
- polimeraza DNA, 140
- primer, 140
- reakcja łańcuchowa polimerazy, 140

sekwencjonowanie, 140
 starter, 140
 struktura, 138
 ukrywanie informacji, 140
 zapis łańcucha, 139
 zasady, 138
 dopełnianie, 106

- metodą 101, 256

 dostrajany szyfr blokowy, 105
 doświadczenie losowe, 73
 dowody z wiedzą zerową, 124
 DSA, 41, 248, 284

- podpisywanie wiadomości, 284
- wariant drugi, 286
- wariant pierwszy, 285
- weryfikacja podpisu, 285

 DSS, 72
 dualizm korpuskularno-falowy, 131
 dummies, 141
 D-Wave

- One, 136
- Two, 136

 dwójkowy system liczbowy, 53
 dysk z Fajstos, 306
 dyski szyfrowane, 212
 dyski wirtualne, 186

- dynamiczne, 214
- obsługa, 215
- tworzenie, 212
- ukryte, 215

 dystrybucja klucza, 21, 31
 działanie dwuargumentowe, 60
 dzielenie próbne, 66
 dzielenie sekretów, 121

- schemat progowy, 143
- secret sharing, 121
- secret splitting, 121

 dziesiętny system liczbowy, 53

E

E0, 112
 ECB, 98
 ECC, 65
 ECM, 66
 EES, 44
 efekt

- dyfuzji, 261
- lawinowy, 246, 248
- superpozycji, 135

 eksperymenty szczelinowe, 131
 Electronic Code Book, 98
 elektroniczna książka kodowa, 98
 elektroniczne uwierzytelnianie użytkowników, 41

- element odwrotny, 60
 - element przeciwny, 60
 - Elliptic Curve Cryptography, 65
 - Elliptic Curve Method, 66
 - Ellis James, 43
 - EME, 105
 - ENIAC, 38
 - Enigma, 33
 - kryptoanaliza, 37
 - walec odwracający, 34
 - zasada działania, 34
 - entropia, 83, 86
 - kryptosystemu, 86
 - EOF, 146
 - Eratosthenes, 55
 - Evans Arthur, 299
 - Exportable, 176
- F**
- faktoryzacja, 54, 58
 - algorytmy, 66
 - dużych liczb, 65
 - dzielenie próbne, 66
 - FCSR, 109
 - FEAL, 91
 - Feedback with Carry Shift Register, 109
 - Feistel Horst, 39
 - File Share Encryption, 198
 - administrator, 199
 - administrator grupy, 199
 - biała lista, 200
 - czarna lista, 199
 - edycja użytkowników, 199
 - lista użytkowników, 198
 - mechanizm chroniących katalogów, 200
 - ochrona plików, 199
 - role użytkowników, 198
 - szyfrowanie indywidualnych plików, 200
 - tworzenie chronionego katalogu, 198
 - użytkownik, 199
 - zmiana lokalizacji katalogu, 200
 - filtr polaryzacyjny, 133
 - Finish, 156
 - fizyka kwantowa w kryptografii, 131
 - forwardowanie portów, 163
 - PuTTY, 169
 - FoxServ, 231
 - fraza kluczowa, 152
 - funkcja
 - l-1, 61
 - aes_decrypt(), 236, 237
 - aes_encrypt(), 236
 - bcrypt, 239
 - crypt(), 239
 - des_decrypt(), 236
 - des_encrypt(), 236
 - Eulera, 57
 - F, 275
 - g, 275
 - h, 277
 - hash_pbkdf2(), 239
 - HMAC, 99
 - jednokierunkowa, 51
 - przykłady, 51
 - zastosowanie, 52
 - matematyczna, 50
 - mcrypt_create_iv(), 233
 - mcrypt_enc_get_iv_size(), 233
 - mcrypt_generic(), 233
 - mcrypt_generic_deinit(), 233
 - mcrypt_generic_init(), 233
 - mcrypt_module_close(), 234
 - mcrypt_module_open(), 233, 235
 - MD5(), 236, 239
 - mdecrypt_generic(), 233
 - na, 62
 - PBKDF2, 238
 - pseudolosowa PRF, 156
 - ROR, 277
 - różnowartościowa, 61
 - rtrim(), 236
 - SHA(), 236
 - SHA2(), 236
 - sprzężenia zwrotnego, 108
 - wyprowadzania klucza, 238
 - wzajemnie jednoznaczna, 62
 - zapadkowa, 51
 - funkcje mieszające, 99
 - SHA-3, 256
 - funkcje skrótu, 118, 238, 257
 - GOST, 257
 - HAVAL, 257
 - jednokierunkowe, 243
 - MD2, 257
 - MD4, 257
 - MD5, 243
 - N-Hash, 257
 - RIPE — MD160, 257
 - SHA-1, 246
 - SHA-2, 249
 - SHA-3, 254
 - Snefru, 257
 - funkcje szyfrujące w PHP, 231
- G**
- GCHQ, 43, 111
 - General Number Field Sieve, 66
 - generatory ciągów rzeczywiście losowych, 110

- generatory liczb losowych, 86
 - generatory pseudolosowe, 107
 - BBS, 109
 - bezpieczeństwo, 108, 110
 - Bluma-Micaliego, 110
 - kongurencyjne, 107
 - oparte na rejestrze przesuwającym ze sprzężeniem zwrotnym, 108
 - oparte na teorii złożoności, 109
 - RSA, 109
 - generatory strumienia klucza, 107
 - generowanie
 - ciągów pseudolosowych, 107
 - kluczy, 271
 - liczb pierwszych, 67
 - podkluczy, 261
 - skrótów hasła, 238
 - strumienia klucza szyfrującego, 102, 103
 - wektora inicjującego, 105
 - GIMPS, 56
 - GNFS, 66
 - GNU Privacy Assistant, 203
 - GnuPG, 203
 - formaty zaszyfrowanych plików, 209
 - GpgEX, 208
 - GpgOL, 208
 - Kleopatra, 203
 - konfigurowanie serwera, 211
 - obsługa certyfikatów, 205
 - obsługa serwerów, 211
 - odczytywanie zaszyfrowanej wiadomości, 208
 - podpis elektroniczny wiadomości, 207
 - podprogramy, 203
 - przewodzenie korespondencji, 206
 - spakowanie pliku do archiwum, 208
 - sprawdzenie poprawności podpisu pliku, 210
 - szyfrowanie plików, 208
 - szyfrowanie wiadomości, 207
 - tworzenie certyfikatu, 203
 - OpenPGP, 205
 - X.059, 204
 - usuwanie oryginalnego pliku, 208
 - weryfikacja podpisu elektronicznego wiadomości, 208
 - wysyłanie wiadomości w postaci plików, 208
 - wyszukiwanie certyfikatów na serwerze, 211
 - GnuPG for Outlook, 203
 - GOST, 257
 - GPG, 203
 - GPG Explorer eXtension, 203
 - GPG4win, 203
 - GpgEX, 203, 208
 - GpgOL, 203, 208
 - group signatures, 120
 - grupa, 60
 - abelowa, 60
 - addytywna, 60
 - multiplikatywna, 60
- ## H
- handshake, 155
 - Handshake Protocol, 154
 - hash_pbkdf2(), 239
 - hasła, 52, 151
 - fraza kluczowa, 152
 - metody wyboru bezpiecznych haseł, 151
 - uwierzytelnianie klienta, 165
 - HAVAL, 257
 - HC-256/128, 112
 - Hebern Eduard Hugo, 34
 - heksadecymalny system liczbowy, 53
 - Helix, 111
 - Hellman Marty, 40
 - Hello Extensions, 155
 - hierarchia certyfikacji, 162
 - hieroglify, 292
 - tajemnica hieroglifów, 297
 - zapis fonetyczny, 297
 - historia kryptografii, 15
 - homofony, 23, 50
- ## I
- ID 500, 135
 - IDEA, 165, 259
 - deszyfrowanie, 261
 - efekt dyfuzji, 261
 - generowanie podkluczy, 261
 - deszyfrujących, 262
 - operacje pojedynczego cyklu, 259
 - pojedyncza runda algorytmu, 260
 - pojedyncza runda dekryptażu, 262
 - przekształcenia MA, 261
 - przekształcenia początkowe, 259
 - schemat dekryptażu, 261
 - idealne kryptosystemy, 241
 - identyfikator
 - certyfikatu, 230
 - wersji certyfikatu, 228
 - wystawcy, 229
 - iloczyn kartezjański, 49
 - implementacja
 - kryptosystemu, 148
 - protokołu TLS, 157
 - informacja progowa, 125
 - informatyka molekularna, 139

- infrastruktura klucza publicznego, 127
 - cykl życia klucza, 129
 - zaufanie, 128
 - złożoność, 128
 - injekcja, 61
 - internetowe magazyny kluczy, 202
 - IP, 263
 - IP Spoofing, 93
 - ISAAC, 112
 - IV, 99
 - jako licznik, 105
 - izomorfizm ciał, 62
 - izomorfizmy, 61
 - funkcja różnowartościowa, 61
 - przekształcenia izomorficzne, 62
 - surjekcja, 62
- J**
- jądro protokołu kanału podprogowego, 125
 - jednokierunkowe funkcje skrótu, 118, 243
 - jednorazowość klucza, 31
 - jednorazowy IV, 106
 - język fleksyjny, 302
 - język mitów, 298
 - pałac króla Minosa, 299
 - wojna trojańska, 298
 - język starosłowiański, 306
- K**
- kamień z Rosetty, 294, 296
 - kanały podprogowe, 125
 - kartusz, 295
 - Ramzesa, 297
 - Kasicki Friedrich W., 28
 - KEA, 114
 - Kerberos, 116
 - Key Exchange Algorithm, 114
 - klatka Faradaya, 94
 - Kleopatra, 203
 - kleptografia, 97
 - klient scp, 166
 - klucz
 - cykl życia, 129
 - rozdzielanie kluczy, 129
 - zmęczenie klucza, 129
 - klucz 56-bitowy, 83
 - klucz jednorazowy, 30
 - dystrybucja, 31
 - losowość, 31
 - klucz pierwotny, 27
 - klucz prywatny, 41
 - klucz publiczny, 40, 41, 43
 - infrastruktura, 127
 - właściciela certyfikatu, 229
 - klucz rozszerzony, 272
 - klucz szyfrujący
 - tworzenie, 114
 - ustalenie, 70
 - klucz tajny, 40
 - known-plaintext attack, 90
 - Kober Alice, 301
 - Koch Hugo Aleksander, 34
 - kod ASCII, 39, 63
 - kod MAC, 52, 99
 - kodowanie informacji genetycznej, 138
 - kodowanie obrazu, 143
 - w skali szarości, 144
 - za pomocą półkoli, 144
 - kolizja, 96
 - kompaktowanie, 189
 - komputer kwantowy, 135
 - komputery DNA, 139
 - koncepcja
 - autoklucza, 27
 - udziałów, 142
 - kongruencja, 52
 - generatory kongruencyjne, 107
 - wzór, 107
 - koniunkcja logiczna AND, 73
 - konkatenacja, 103
 - algorytm SHA-1, 248
 - konstrukcja gąbkowa, 254, 256
 - krok szyfrowania, 50
 - kryptoanaliza, 22
 - Enigmy, 37
 - inskrypcji, 291
 - kodu, 292
 - liniowa, 92
 - operacje XOR i AND, 92
 - oparta na prawdopodobieństwie, 29
 - rozwój, 23
 - różnicowa, 91
 - atak metodą kluczy powiązanych, 91
 - charakterystyki, 91
 - szyfrów homofonicznych, 23
 - szyfrów polialfabetycznych, 73
 - znalezisk, 291
 - kryptografia, 20
 - alternatywna, 131
 - asymetryczna, 40
 - liczby pierwsze, 54
 - DNA, 138
 - informacja ukryta w DNA, 140
 - informatyka molekularna, 139
 - szyfrowanie z wykorzystaniem nukleotydów, 141
 - ukrywanie wiadomości, 141

- era komputerów, 38
 - fizyka kwantowa, 131
 - historia, 15
 - II wojny światowej, 33
 - jako gwarancja bezpieczeństwa, 151
 - konstrukcja bezpiecznego systemu, 147
 - krzywych eliptycznych, 65
 - kwantowa, 132
 - matematyczne podstawy, 47
 - molekularna, 141
 - klucz szyfrowania, 141
 - stoper, 141
 - oparta na kodach, 138
 - oparta na kratach, 137
 - początki, 20
 - podstawowe pojęcia, 48
 - postkwantowa, 137
 - radio, 32
 - rozwój, 23
 - statystyka, 47
 - telegraf, 32
 - upowszechnienie, 44
 - w PHP i MySQL, 231
 - w praktyce, 147
 - w służbie historii, 291
 - w teorii, 89
 - wielu zmiennych wykorzystująca równania
 - drugiego stopnia, 138
 - wizualna, 142
 - kodowanie obrazu, 143
 - schemat progowy, 143
 - udziały, 142
 - zegar, 116
 - kryptograficznie zabezpieczone współdzielenie plików, 198
 - kryptosystem, 50
 - hybrydowy, 289
 - implementacja, 148
 - miara entropii, 86
 - monoalfabetyczny, 51
 - NTRU, 137
 - określanie bezpieczeństwa, 82
 - piloalfabetyczny, 51
 - RSA, 63
 - wybór, 148
 - książki kodowe, 32
 - kubit, 135
 - kwant, 131
 - kwantowe wyzarzanie, 136
 - kwas dezoksyrybonukleinowy, 138
 - kwas rybonukleinowy, 138
- L**
- lanes, 254
 - lattice based cryptography, 137
 - Least Significant Bit, 145
 - libmccrypt, 232
 - liczba e, 59
 - liczba informacji, 83
 - liczby Fermata, 57
 - liczby naturalne, 54
 - liczby pierwsze, 54
 - generowanie, 67
 - liczby Fermata, 57
 - małe twierdzenie Fermata, 57
 - Mersenne'a, 56
 - mocne, 67
 - najmniejsza wspólna wielokrotność, 58
 - największy wspólny dzielnik, 58
 - NWD, 58
 - NWW, 58
 - RSA, 63
 - sito Atkina, 55
 - sito Eratostenesa, 55
 - sito Sundarama, 55
 - świadek złożoności, 68
 - test Fermata, 69
 - test Lehmana, 68
 - test Rabina-Millera, 68
 - test Solovaya-Strassena, 69
 - testy pierwszości, 67
 - wyszukiwanie, 55
 - na komputerze, 56
 - liczby względnie pierwsze, 57
 - wyszukiwanie, 58
 - Linear Feedback Shift Register, 108
 - liniowy rejestr przesuujący ze sprzężeniem zwrotnym, 108
 - lista CRL, 130
 - lista odwołań certyfikatów, 130
 - ln, 59
 - logarytm dyskretny, 40, 70
 - atak Man in the Middle, 71
 - generator pseudolosowy, 110
 - zastosowania, 72
 - logarytm naturalny, 59
 - logarytmy, 59
 - logika boolowska z wykorzystaniem inhibitorów, 140
 - losowość klucza, 31
 - losowy IV, 105
 - LRW, 105
 - LSFR, 108
 - filtry, 111
 - Lucyfer, 39

Ł

łamanie

- metodą pełnego przeglądu, 90
- polialfabetów, 28
- z adaptacyjnie wybranym tekstem jawnym, 91
- z szyfrogramami, 90
- z wybranym szyfrogramem, 90
- z wybranym tekstem jawnym, 90
- ze znanym tekstem jawnym, 90

M

macierz

- alokacji permutacji z rozszerzeniem, 266
- kompresji, 265
- MDS, 275
- permutacji klucza, 265
- permutacji końcowej, 268
- permutacji początkowej, 264
- przekształceń, 92

małe twierdzenie Fermata, 57

master-secret, 156

maszyna wirnikowa, 34

Mauborgne Joseph, 30

mcrypt, 232

mcrypt_create_iv(), 233

MCRYPT_DEV_RANDOM, 233

MCRYPT_DEV_URANDOM, 233

mcrypt_enc_get_iv_size(), 233

mcrypt_generic(), 233

mcrypt_generic_deinit(), 233

mcrypt_generic_init(), 233

mcrypt_module_close(), 234

mcrypt_module_open(), 233, 235

MCRYPT_RAND, 233

MD2, 257

MD4, 257

MD5, 162, 243

- bezpieczeństwo, 246
- efekt lawinowy, 246
- obliczenia końcowe, 246
- operacje dla cyklu pierwszego, 245
- pętla główna, 244
- pojedyncza operacja matematyczna, 245
- przekształcenia początkowe, 243
- stała addytywna, 246
- tekst jawny, 243
- wykorzystywane funkcje, 244
- zmiennie łańcuchowe, 243

MD5(), 236, 239

mdecrypt_generic(), 233

MDS, 275

Message Authentication Code, 99

Message Digest, 243

Meta-Introducer Non-Exportable, 176

metoda

- akustyczna, 94
- dopełnienia ostatniego bloku wiadomości, 106
- krzywej eliptycznej, 66
- różnicowej analizy błędów, 95
- szyfrowania pakietów, 156

metody kryptoanalityczne, 89

metody kryptografii, 21

- analiza częstości, 22
- oparte na koncepcji autoklucza, 27
- szyfr digraficzny, 29
- szyfrowanie z użyciem homofonów, 23
- szyfry homofoniczne, 23
- szyfry polialfabetyczne, 24
- tabuła recta, 26
- tarcza Albertiego, 24

Microsoft Outlook, 224

- informacje o certyfikacie, 229
- opcje wiadomości, 226
- ścieżka certyfikacji, 230
- wybór certyfikatu, 225
- zabezpieczenia poczty e-mail, 225

Microsoft Outlook Express, 226

mieszanie, 87

MixColum, 272

mnożnik, 108

moc zbioru, 51

- zdarzeń elementarnych, 74

mocne liczby pierwsze, 67

modyfikacje DES, 269

Mozilla Thunderbird, 227

- okno dialogowe Certyfikaty, 227
- podpisywanie i szyfrowanie wiadomości, 227

MTI, 71

MUGI, 112

Multiplication-Addition, 261

Multivariate-quadratic-equations cryptography, 138

MySQL

- baza przed zaszyfrowaniem, 237
- bcrypt, 238
- klucz do szyfrowania kluczy, 241
- menedżer kluczy, 241
- ochrona hasła dostępu, 238
- wzbogacanie o sól, 238
- pbkdf2, 238
- projekt struktury systemu, 240
- przechowywanie wycofanych kluczy, 241
- przeszukiwanie zaszyfrowanych danych, 240
- przykładowe funkcje, 236
- sejf na klucze, 241
- szyfrowanie danych, 236, 237
- algorytm AES, 237
- funkcja SHA2, 238

udoskonalenia, 240
zarządzanie kluczami, 241

N

nadmiarowość języka, 85, 86
najmniejsza wspólna wielokrotność, 58
największy wspólny dzielnik, 58
Nanoteq, 111
nazwa i identyfikator właściciela certyfikatu, 229
negatywny wzorzec, 34, 36, 67
N-Hash, 257
niepodrabialne podpisy cyfrowe, 119
niezaprzeczalne podpisy cyfrowe, 118
NIST, 41, 246
nomenklatory, 23
Nonce, 103
Non-Exportable, 175
notacja wielkiego O, 82
NSA, 40
NTRUEncrypt, 137
NTRUSign, 137
nukleotydy, 138
 zapis danych, 140
Number used once, 103
numer seryjny certyfikatu, 228
NWD, 58
NWW, 58

O

obraz cyfrowy, 142
obsługa certyfikatów, 205
 cyfrowych, 220
obsługa serwerów, 211
oczekiwane Chi, 80
odcisk palca, 230
odległość jednostkowa, 86
odwrotna inżynieria oprogramowania, 148
OFB, 102, 107
OFBNLF, 104
one-time pad, 30, 72
OpenPGP
 domyślny serwer, 211
 GnuPG, 203
OpenSSL, 162
operacja AND, 73, 92
operacja XOR, 72, 87, 92
operacje w cyklu
 IDEA, 259
 SHA-1, 247
 SHA-2, 251
operator złączenia, 103
Opportunistic Encryption, 172
oprogramowanie steganograficzne, 146

osobista strona domowa, 231
ostrzeżenie, 156
Output Feedback, 102
Output Feedback with a Nonlinear Function, 104

P

Pageant, 166
Painvin, 16
pakiet Symantec Encryption Desktop, 169
para kluczy szyfrujących, 40, 43
 generowanie, 170
 tworzenie, 173
paradoks urodzin, 95
Password Based Key Derivation Function, 238
PBC, 104
pbkdf2, 238
P-bloki, 267
PCBC, 104
PCR, 140
permutacja, 50
 końcowa, 268
 początkowa, 263
 rozszerzona, 265
 tekstu w czasie szyfrowania, 87
 z kompresją, 265
 zwykła, 267
Personal Home Page, 231
pętla algorytmu
 SHA-1, 248
 z rodziny SHA-2, 252
pętla główna
 MD5, 244
 SHA-1, 247
PFB, 104
PGP, 42, 169
 funkcje skrótu, 243
 generator liczb losowych, 111
 moduły, 173
 Secure Viewer, 94
 system hybrydowy, 150
PGP Disk, 186
 bezpieczeństwo, 189
 czyszczenie wolnej przestrzeni dyskowej, 193,
 195
 harmonogram, 195
 liczba przejść, 193
 warianty harmonogramu, 196
 dodatkowi użytkownicy, 187
 dyski typu Dynamic, 189
 ekran powitalny, 192
 hasło podczas uruchamiania systemu, 190
 klucz prywatny na tokenie, 190
 modyfikowanie partycji, 193
 ochrona dwupoziomowa, 191

- PGP Disk
- odzyskiwanie hasła, 192
 - opcje przyłączania i odłączania dysków, 188
 - para kluczy szyfrowania, 190
 - ponowne zaszyfrowanie dysku, 189
 - rozmiar dysku, 186
 - sposób zabezpieczania dostępu do dysku, 191
 - statystyka wybranego dysku, 194
 - szyfrowanie całego dysku lub partycji, 189
 - tworzenie dysków wirtualnych, 186
 - uwierzytelnianie z wykorzystaniem platformy TPM, 191
 - wielokrotne nadpisanie na wybranym obszarze, 193
 - wybór algorytmu szyfrowania, 187
 - wybór napędu, 193
 - wybór systemu plików dla tworzonego dysku, 187
 - wyświetlenie właściwości dysku wirtualnego, 187
 - wznowienie szyfrowania, 192
 - zabezpieczenie dostępu do dysku, 190
 - zatrzymanie szyfrowania, 192
- PGP Keys, 173
- All keys, 173
 - Email this Key, 173
 - importowanie otrzymanego klucza publicznego, 177
 - lista dodatkowych kluczy deszyfrowania, 175
 - lista kluczy głównych, 177
 - lista kluczy odwołujących, 175
 - maksymalna głębia zaufania do klucza, 176
 - My Private Keys, 173
 - ograniczenie domeny, 176
 - opcja podpisywania klucza, 175
 - operacje na kluczach, 173
 - przypisanie stopnia zaufania do klucza, 174
 - rekonstruowanie klucza, 177
 - Search for Keys, 173
 - Signature Type, 175
 - tworzenie dodatkowych podkluczy, 174
 - wycofanie złożonego podpisu, 176
 - wyświetlanie właściwości klucza, 174
 - zapasowa kopia klucza, 177
- PGP Messaging, 177
- adnotacje, 178
 - definiowanie komentarza, 179
 - definiowanie parametrów konta pocztowego, 180
 - dodawanie nowej usługi, 182
 - główne okno modułu, 180
 - konfigurowanie polityk dla definiowanej usługi, 181
 - konfigurowanie zaawansowanych ustawień serwera pocztowego, 180
 - określanie wykorzystywanego protokołu, 181
 - polityka zabezpieczeń, 178
 - przyciski do podpisywania w MS Outlook, 179
 - przyciski do szyfrowania w MS Outlook, 179
 - Service, 178
 - szyfrowanie w miarę możliwości, 178
 - usługa, 178
 - ustawienia dotyczące adnotacji, 178
 - zmiana zabezpieczenia wiadomości przesyłanych na serwer pocztowy IMAP, 179
- PGP NetShare, 199
- PGP Shredder, 200
- ustawienia, 201
 - usuwanie plików, 201
- PGP Viewer, 196
- dodatkowe ustawienia modułu, 198
 - konfiguracja podstawowych ustawień, 198
 - konfigurowanie opcji wyświetlania tekstu, 198
 - Preferences, 198
 - Show Remote Images, 198
 - Text Encoding, 198
 - View Message Source, 198
 - współpraca z klientami pocztowymi, 196
 - wyświetlanie wiadomości, 197
- PGP Zip, 182
- edytowanie archiwum, 185
 - opcje tworzenia archiwum, 183
 - rozpakowanie zaszyfrowanego archiwum, 185
 - ustawienia zapamiętywania hasła, 184
 - usuwanie pliku lub katalogu, 185
 - wybranie klucza publicznego odbiorcy, 184
- PHP, 231
- funkcje szyfrujące, 231
 - biblioteka mcrypt, 232
 - dostępne algorytmy, 232
 - wskazówki, 235
 - wywoływanie, 233
 - serwer, 231
 - szyfrowanie zmiennej algorytmem AES, 234
 - wpisywanie zaszyfrowanych zmiennych do bazy danych, 235
- pierścień, 60, 138
- przemienny, 61
- piksel, 142
- PIN, 52
- pismo demotyczne, 293
- pismo hieratyczne, 292
- pismo hieroglificzne A, 300
- pismo hieroglificzne B, 300
- pismo klinowe, 305
- pismo koptyjskie, 293
- pismo linearne, 299
- odczytanie, 301, 305
 - syllabariusz, 301
 - syllaby łączące, 302
 - zapis starożytnej greki, 305
- pismo linearne A, 300

- pismo linearne B, 300
- pismo tocharyjskie, 305
- PKI, 127
 - cykl życia klucza, 129
 - model teoretyczny, 127
 - rozprowadzanie klucza, 129
 - ścieżka certyfikacji, 128
 - użytkowanie aktywne klucza, 129
 - użytkowanie pasywne klucza, 129
 - w praktyce, 128
 - zaufanie, 128
 - złożoność, 128
 - zmęczenie klucza, 129
- Plaintext Block Chaining, 104
- Plaintext Feedback, 104
- plan zamknięcia, 149
- platforma TPM, 191
- pliki dokumentacji serwerów, 158
- pliki identyfikujące, 164
- Plink, 166
- podciasto, 61
- podklucze, 174
- podpis cyfrowy, 41, 117
 - algorytm DSA, 284
 - algorytm rozpraszający, 246
 - algorytm SHA-1, 248
 - automatyczne podpisywanie wszystkich wysyłanych wiadomości, 228
 - centra certyfikacji, 222
 - certyfikat cyfrowy, 221
 - jądro protokołu kanału podprogowego, 125
 - jednokierunkowe funkcje skrótu, 118
 - konfiguracja programu pocztowego, 224
 - niepodrabialny, 119
 - niezaprzeczalny, 118
 - podpisy grupowe, 120
 - podpisy pośrednie, 120
 - podpisy ślepe, 119
 - podpisy z wyznaczonym potwierdzającym, 120
 - popularne algorytmy i schematy, 286
 - potwierdzanie tożsamości innych użytkowników, 228
 - skaner antywirusowy, 228
 - składanie i weryfikacja, 220
 - struktura certyfikatu, 228
 - uzyskiwanie certyfikatu, 222
 - warianty, 120
 - wymagania techniczne, 220
- podpis elektroniczny, 41, 117
- podstawianie, 21
- podwójna helisa, 138
- Pokój 40, 32
- polaryzacja, 132
- polimeraza DNA, 140
- połączenie w trybie szyfrowanym, 160
- port 11371, 211
- port 389, 211
- Porta Giovanni Battista, 29
- Post-quantum Cryptography, 137
- potrójny DES, 269
- poufność przekazywana, 163
- poziom szyfrowania, 39
- prawdopodobieństwo, 73
 - rozkład, 73
 - wystąpienia pary liter, 75
 - zdarzenia, 74
- prawdopodobnie bezpieczny, 149
- pre-master-secret, 155
- Pretty Good Privacy, 42
- PRF, 156
- primer, 140
- procedura uwierzytelniania, 126
- proces certyfikacji, 207
- program pocztowy
 - dane o zaimportowanych certyfikatach, 228
 - informacje o certyfikacie, 229
 - konfiguracja, 224
 - Microsoft Outlook, 224
 - Microsoft Outlook Express, 226
 - Mozilla Thunderbird, 227
 - wirusy, 228
- projekt
 - EES, 44
 - GIMPS, 56
- propagacja, 104
- Propagating Cipher Block Chaining, 104
- protokoły kryptograficzne, 113
 - dowody z wiedzą zerową, 124
 - dzielenie sekretów, 121
 - kanały podprogowe, 125
 - podpis cyfrowy, 117
 - uwierzytelniające, 126
 - wymiany kluczy, 114
 - znakowanie czasowe, 123
- protokoły uwierzytelniające, 126
- protokoły wymiany kluczy, 114
 - DASS, 117
 - Denning-Sacco, 117
 - Diffiego-Hellmana, 114
 - KEA, 114
 - Kerberos, 116
 - Needhama-Schroedera, 117
 - Neumana-Stubblebine'a, 117
 - Otwaya-Reesa, 116
 - wide-mouth frog, 115
 - Yahaloma, 117
- protokół
 - alarmowy, 154, 156
 - arbitrażowy, 124
 - bezpieczeństwa, 154

- protokół
- bezpiecznego uwierzytelnienia
 - rozproszonego, 117
 - Bluetooth, 112
 - DASS, 117
 - Denning-Sacco, 117
 - Diffiego-Hellmana, 114
 - dowodu z wiedzą zerową, 125
 - dystybucyjny, 124
 - Fiata-Shamira, 126
 - Guillou-Quisquarera, 126
 - hkp, 211
 - http, 211
 - HTTPS, 159
 - KEA, 114
 - Kerberos, 116
 - ldap, 211
 - łązący, 124
 - MTI, 71
 - Needhama-Schroedera, 117
 - Neumana-Stubblebine'a, 117
 - określania formatu pakietów, 154
 - określania pakietów, 156
 - Otwaya-Reesa, 116
 - podpisu cyfrowego, 117
 - grupowy, 120
 - klasyczny, 118
 - niepodrabialny, 119
 - niezaprzeczalny, 118
 - pośredni, 120
 - ślepe podpisy, 119
 - z wyznaczonym potwierdzającym, 120
 - pokerowy, 126
 - połączenia, 163
 - rzucanie monetą, 126
 - Schnorra, 126
 - secret sharing, 121
 - secret splitting, 121
 - SSH, 125, 162
 - SSH1, 165
 - SSH2, 165
 - SSL, 154
 - STS, 71
 - TLS, 154
 - ujawnianie tajemnic, 126
 - uwierzytelniania użytkownika, 163
 - uzgadniania, 154, 155
 - warstwy transportowej, 163
 - wide-mouth frog, 115
 - współdzielenia sekretów z oszukującymi, 123
 - wszystko albo nic, 126
 - wykorzystujący podpis cyfrowy, 125
 - Yahaloma, 117
 - zmiany specyfikacji szyfru, 154
 - znakowanie czasowe, 123
 - zobowiązania bitowe, 126
- proxy signatures, 120
- próba czasu, 241
- przeglądanie poczty, 196
- przejście, 194
- przekształcenia
 - ByteSub, 273
 - izomorficzne, 61, 62
 - MA, 261
 - PHT, 279
- przestawianie, 21
- przestrzeń tekstu, 48
- przestrzeń zdarzeń elementarnych, 73
- przyrost, 108
- przystawka szyfrująca, 37
- PSCP, 166
- Pseudo-Hadamard Transforms, 275
- PSFTP, 166
- Public Key Infrastructure, 127
- pusty ciąg, 233
- PuTTY, 166
 - forwardowanie portów, 169
 - główne okno programu, 168
 - pierwsze połączenie z serwerem, 168
 - tryb połączenia, 168
- PuTTY link, 166
- PuTTYgen, 166
 - klucz publiczny, 167
- PuTTYtel, 166

Q

- Quadratic Sieve, 66
- quantum annealing, 136
- quantum entanglement, 132
- qubit, 135
- QuickStego, 146

R

- Rabbit, 112
- Rambutan, 111
- raport z półwiecza, 303
- RAW, 168
- RC4, 111, 165
- RC4+, 111
- RC4A, 111
- rcp, 166
- reakcja łańcuchowa polimerazy, 140
- Record Protocol, 154
- redukcja modularna, 52
- rejestr przeniesienia, 109

rejestr przesuwający, 101, 108
 FCSR, 109
 LSRF, 108
 z przeniesieniem w sprzężeniu zwrotnym, 109
 Rejewski Marian, 35
 relacja, 49
 jednoznaczna, 50
 system szyfrowania, 50
 Remote copy, 166
 Remote Shell, 162
 Request Hello, 155
 Require Encryption
 PGP, 172
 Revokers, 175
 Rijndael, 271, 274
 RIPE — MD160, 257
 Rivest Ron, 41, 243
 Rlogin, 166, 168
 RNA, 138
 rodzaje szyfrowania, 98
 szyfry blokowe, 98, 113
 szyfry strumieniowe, 107, 113
 ROR, 277
 $ROTL^n(x)$, 249
 $ROTR^n(x)$, 249
 rounds, 239
 rozdzielanie
 kluczy, 129
 sekretu, 121
 rozkład prawdopodobieństwa, 73
 rozkładanie na czynniki pierwsze, 54
 rozpraszanie, 87
 rozszerzenie ciała, 61
 rozszerzony klucz, 238
 Różycki Jerzy, 35
 RSA, 41, 43, 63, 125, 287
 algorytm szyfrowania, 58
 bezpieczeństwo kryptosystemu, 65
 chińskie twierdzenie o resztach, 70
 generator, 109
 generowanie pary kluczy, 287
 implementowanie algorytmu, 64
 klucze publiczne, 64
 obliczanie wartości tajnego klucza, 65
 odwrócenie działania funkcji szyfrującej, 64
 proces szyfrowania, 63
 standard ASCII, 63
 szyfrowanie i deszyfrowanie, 287
 RSA-232, 65
 RSH, 162
 rtrim(), 236
 runda zerowa, 272
 rząd grupy, 60
 rząd złożoności, 82
 rzucanie monetą, 126

S, Ś

S-bloki, 87, 92, 266
 AES, 272
 struktura, 267
 schemat progowy, 143
 Scherbius Artur, 34
 Schliemann Heinrich, 298
 Scp, 166
 scytale, 21
 SEAL, 112
 secret sharing, 121
 algorytm Asmutha-Blooma, 123
 algorytm Karnina-Greene'a-Hellmana, 122
 algorytm wektorowy, 122
 algorytm wielomianu interpolacyjnego
 Lagrange'a, 122
 secret splitting, 121
 Secure copy, 166
 Secure Hash Algorithm, 246
 Secure Shell, 162
 Secure Socket Layer, 154
 Secure Viewer, 94
 sejf na klucze, 241
 sekwencja zaczepów, 108
 sekwencjonowanie, 140
 Server Certificate, 155
 Server Hello, 155
 Server Hello Done, 155
 Server Key Exchange Message, 155
 serwery
 instrukcje, 158
 obsługa, 211
 sesja SSH, 163
 ustanawianie bezpiecznego połączenia, 164
 uwierzytelnianie klienta, 164
 uwierzytelnianie serwera, 164
 SHA(), 236
 SHA-1, 246
 bezpieczeństwo, 249
 blok danych, 247
 efekt lawinowy, 248
 funkcje cykli, 247
 obliczenia końcowe, 248
 operacje w cyklu, 247
 pętla algorytmu, 248
 pętla główna, 247
 przekształcenia początkowe, 246
 stała Kt, 247
 wykorzystanie, 248
 zmienna addytywna, 248
 zmienna, 246
 SHA-2, 96, 249
 dane wejściowe kolejnego cyklu, 252
 dodatkowe różnice między algorytmami, 253

- SHA-2
 - działania, 249
 - operacje w cyklu, 251
 - pętla algorytmu, 252
 - początkowe wartości skrótu, 250
 - pośrednia wartość skrótu, 253
 - przekształcenia początkowe, 250
 - $ROTL^n(x)$, 249
 - $ROTR^n(x)$, 249
 - rundy algorytmów, 251
 - $SHR^n(x)$, 249
- SHA2(), 236, 239
- SHA-224, 249, 250, 251, 253
- SHA-256, 249, 250, 251, 253
- SHA-3, 254
 - ciągi, 254
 - funkcja mieszająca, 256
 - funkcja rundy, 254
 - ogólny opis, 254
 - stan funkcji, 254
 - struktura przebiegu algorytmu, 256
 - tablica stanu, 254
 - zakres permutacji, 254
- SHA-384, 249, 250, 251, 253
- SHA-512, 249, 251, 253
- Shamir Adi, 41
- Shannon Claude Elmwood, 83
- shares, 142
- ShiftRow, 272
- Shred Free Space, 200
- $SHR^n(x)$, 249
- sieć zaufania, 202
- siła szyfrowania, 221
- sito Atkina, 55
- sito Atkina-Bernsteina, 55
- sito Eratostenesa, 55
- sito kwadratowe, 66
- sito Sundarama, 55
- Skipjack, 115
- słownik tekstu jawnego, 48
- słownik tekstu zaszyfowanego, 48
- słowo klucz, 28
- słowo puste, 48
- Snefru, 257
- SNFS, 66
- SOBER-128, 112
- socjotechnika, 97
 - bezpieczeństwo systemu kryptograficznego, 152
- sól kryptograficzna, 238, 239
- Specialized Number Field Sieve, 66
- spin, 132
- splątanie kwantowe, 132
- sponge, 254
- Spoofing, 163
- sposoby użytkownika systemu, 241
- sprawdzanie certyfikatu, 160
- SSH, 162, 168
 - forwardowanie portów, 163
 - klient, 166
 - podprotokoły, 163
 - protokół połączenia, 163
 - protokół uwierzytelniania użytkownika, 163
 - protokół warstwy transportowej, 163
 - PuTTY, 166
 - sesja, 163
 - SSH1 a SSH2, 165
 - tunelowanie połączenia, 163
 - wykorzystywane algorytmy, 165
- SSL, 154
 - funkcje skrótu, 243
- stała addytywna, 246
- stała Eulera, 59
- standard
 - DSS, 166, 284
 - podpisów cyfrowych, 72
 - szyfrowania danych, 40
 - UNICODE, 39
 - X.509, 130, 203, 220
- starter, 140
- Station-to-Station, 71
- steganografia, 19
 - kanał podprogowy, 125
 - oprogramowanie, 146
 - współczesna, 144
 - znaki wodne, 145
- Steganography Studio, 147
- Steganos Security Suite, 146
- Stegdetect, 147
- Stego Suite, 147
- stegoanaliza
 - analiza sygnaturowa, 147
 - programy, 146
- StegoMagic, 146
- StegoVideo, 146
- StegSpy, 147
- stoper, 141
- string, 233
- struktura DNA, 138
- STS, 71
- subkeys, 174
- subliminal channel, 125
- subpiksel, 142
- sumowanie modulo 2, 72
- superpozycja, 135
 - stanów, 132
- surjekcja, 62
- Symantec Encryption Desktop, 169
 - analiza szczegółowych ustawień, 172
 - dodanie klucza publicznego do katalogu globalnego, 171

- File Share Encryption, 198
- generowanie pary kluczy szyfrujących, 170
- hasło chroniące klucz prywatny, 171
- konfigurowanie ustawień szyfrowania komunikacji, 172
- konfigurowanie ustawień wysyłania wiadomości, 172
- kreator tworzenia kluczy, 170
- PGP Disk, 186
- PGP Keys, 173
- PGP Messaging, 177
- PGP Shredder, 200
- PGP Viewer, 196
- PGP Zip, 182
- szyfrowanie całego dysku, 189
- Web of Trust, 201
- zaawansowane ustawienia generowania kluczy, 171
- Symantec Encryption Management Server, 170
- symbol pusty, 24
- SYN flood, 93
- system deszyfrowania, 50
- system kryptograficzny, 50
 - bezpieczeństwo, 147
 - ElGamala, 72
 - Massey-Omury, 72
 - możliwości techniczne środowiska, 150
 - oparty na modułach, 149
 - użytkowanie, 150
 - wydajność, 150
 - złożoność, 149
 - zbiór znaków, 48
- system liczbowy
 - dwójkowy, 53
 - dziesiętny, 53
 - heksadecymalny, 53
 - szesnastkowy, 53
- system podpisu elektronicznego oparty na skrótach, 137
- system szyfrowania, 50
- system zabezpieczania informacji, 242
- szesnastkowy system liczbowy, 53
- szyfr nie do złamania, 30
- szyfrogram
 - zbiór znaków, 49
- Szyfropunki, 44
- szyfrowanie, 50, 262
 - algorytm RSA, 287
 - całego dysku lub partycji, 189
 - danych
 - w MySQL, 236
 - w pamięciach masowych, 105
 - w telefonii komórkowej GSM, 111
 - funkcyjne, 50
 - jednokierunkowa funkcja zapadkowa, 52
 - monoalfabetyczne, 51
 - pojedyncze szyfrowanie, 51
 - niefunkcyjne, 50
 - plików, 208
 - polialfabetyczne, 51
 - rodzaje, 98
 - tryby, 98
 - w trybie CBC, 100
 - w trybie CFB, 101
 - w trybie OFB, 103
 - z kluczem jednorazowym, 72
 - z wykorzystaniem nukleotydów, 141
 - z zastosowaniem RSA, 125
- szyfry
 - ADFGX, 16, 20
 - Albertiego, 24
 - blokowe, 39, 87, 98
 - BC, 104
 - Blowfish, 282
 - CBC, 99
 - CBCC, 104
 - CFB, 101
 - CTR, 103
 - DES, 263
 - długość wiadomości, 106
 - dopełnianie, 106
 - ECB, 98
 - LRW, 105
 - OFB, 102
 - OFBNLF, 104
 - PBC, 104
 - PCBC, 104
 - PFB, 104
 - tryby, 98
 - w bibliotece mcrypt, 232
 - wektor inicjujący, 105
 - wyбір trybu, 106
 - wykorzystanie, 113
 - XTS-AES, 105
 - Cezara, 21, 48, 50, 51, 53
 - digraficzne, 29
 - faraonów, 294
 - homofoniczne, 23
 - szyfrowanie niefunkcyjne, 50
 - monoalfabetyczne, 21
 - Playfaira, 29
 - podstawieniowe, 23
 - polialfabetyczne, 24
 - rozkład częstości, 75
 - testy zgodności, 73
 - polowe, 32
 - RSA, 63
 - strumieniowe, 107
 - algorytmy, 111
 - generatory ciągów rzeczywiście losowych, 110

- szyfry
 - generowanie ciągów pseudolosowych, 107
 - wykorzystanie, 113
 - Vigenere'a, 26, 72
 - wieloliterowe, 29
 - z kluczem jednorazowym, 30
 - ścieżka certyfikacji, 128, 162
 - ścieżka Hamiltona, 139
 - ścieżki zaufania, 202
 - ślepe podpisy cyfrowe, 119
 - świadek złożoności, 68
 - święte rysunki, 292
- T**
- tabela Trithemiusa, 26
 - tablica Polibiusza, 20
 - tablica stanu, 254
 - tablica szyfru Playfaira, 29
 - tablica Trithemiusa, 27, 51
 - tablica Vigenere'a, 77
 - tabula recta, 25
 - tajny klucz deszyfrujący, 64
 - tarcza Albertiego, 24
 - technika
 - IP Spoofing, 93
 - szyfrowania asymetrycznego, 43
 - wydłużenia efektywnej długości klucza, 271
 - techniki biometryczne, 142
 - tekst jawny
 - przestrzeń, 48
 - słownik, 48
 - zbiór znaków, 48, 49
 - zmniejszenie nadmiarowości, 87
 - tekst kodowany, 48
 - tekst tajny
 - przestrzeń, 48
 - telegraf, 32
 - Telnet, 166, 168
 - teoria informacji, 83
 - entropia, 83
 - mieszanie, 87
 - nadmiarowość języka, 85
 - odległość jednostkowa, 86
 - rozpraszanie, 87
 - zawartość informacyjna języka, 85
 - teoria złożoności, 82
 - test
 - AKS, 69
 - Baillie-PSW, 69
 - Chi, 73, 80
 - Fermata, 69
 - Fi, 73, 78
 - Kappa, 73, 74
 - Lehmana, 68
 - oparty na krzywych eliptycznych, 69
 - Rabina-Millera, 68
 - Solovaya-Strassena, 69
 - testy pierwszości, 67
 - testy zgodności, 73
 - polialfabetyczności, 79
 - TLS, 154
 - biblioteka OpenSSL, 162
 - Certificate Request, 155
 - Certificate Verify, 156
 - Client Certificate, 155
 - Client Hello, 155
 - Client Key Exchange Message, 155
 - Finish, 156
 - Hello Extensions, 155
 - ikona kłódki, 160
 - implementacja protokołu, 157
 - pobranie certyfikatu testowego, 157
 - lokalizacja, 154
 - nawiązywanie połączenia, 155
 - podprotokoły, 154
 - protokół alarmowy, 156
 - protokół określania pakietów, 156
 - przesyłanie danych, 156
 - przezroczystość, 154
 - Request Hello, 155
 - Server Certificate, 155
 - Server Hello, 155
 - Server Hello Done, 155
 - Server Key Exchange Message, 155
 - sprawdzanie certyfikatu, 160
 - struktura, 154
 - wyznaczanie kluczy, 156
 - tocjent Eulera, 57
 - TPM, 191
 - transkryptor, 139
 - Transport Layer Security, 154
 - Triple DES, 269
 - Trithemius Johannes, 25
 - trojaczki Kober, 302
 - TrueCrypt, 146, 212
 - automatyczne odłączanie dysków, 219
 - automatyczne przyłączanie zaszyfrowanych dysków, 217
 - dane losowe do wygenerowania kluczy, 215
 - dyski ukryte, 215
 - ochrona, 216
 - przyłączanie, 216
 - dyski wirtualne
 - dodatkowe opcje, 214
 - kreator, 213
 - narzędzia powiązane, 217
 - obsługa, 215
 - przyłączanie, 215
 - tworzenie ukrytych dysków wirtualnych, 215

- główne okno programu, 213
 - informacje dotyczące zaznaczonego dysku, 217
 - obsługa całych dysków, 215
 - opcje, 217
 - pamięć cache, 219
 - usuwanie hasła, 217
 - plik klucza, 218
 - generowanie losowe, 218
 - tworzenie, 218
 - polecenia, 217
 - preferencje zdefiniowane w programie, 219
 - próba nadpisania ukrytych danych, 217
 - przyłączanie dysku zewnętrznego, 216
 - szyfrowane dyski i partycje, 212
 - wybór szyfrowanego dysku, 213
 - zmiany w konfiguracji programu, 218
 - Trusted Introducer Exportable, 176
 - Trusted Platform Module, 191
 - tryb propagującego wiązania bloków zaszyfrowanych, 104
 - tryb sprzężenia zwrotnego
 - szyfrogramu, 101
 - tekstu jawnego, 104
 - wyjściowego, 102
 - z funkcją nieliniową, 104
 - tryb szyfrowania dysków, 105
 - tryb wiązania blokowego, 104
 - tryby szyfrowania, 98
 - BC, 104
 - CBC, 99
 - CBCC, 104
 - CFB, 101
 - CTR, 103
 - ECB, 98
 - LRW, 105
 - OFB, 102
 - OFBNLF, 104
 - PBC, 104
 - PCBC, 104
 - PFB, 104
 - szyfry blokowe, 98
 - XTS-AES, 105
 - tunelowanie połączenia, 163
 - tunelowanie portów, 169
 - Turing Alan, 36
 - tweakable-block-cipher, 105
 - twierdzenie o resztach, 70
 - Twofish, 165, 275
 - dodanie kluczy szyfrowania, 280
 - efekt lawinowy, 280
 - funkcja F, 275
 - funkcja g, 275, 278
 - funkcja h, 277
 - funkcja ROR, 277
 - klucz rozszerzony, 280
 - opis, 275
 - pojedyncza runda algorytmu, 275
 - przekształcenie PHT, 279
 - tworzenie
 - algorytmu, 241
 - dysków wirtualnych, 186
 - par kluczy, 173
 - szyfrowanych dysków, 212
- ## U
- udziały, 142
 - ujawnianie tajemnic, 126
 - układy równań wielomianowych, 121
 - ukryte dyski, 215
 - ochrona, 216
 - przyłączanie, 216
 - UNICODE, 39
 - uogólnione sito ciała liczbowego, 66
 - urządzenia szyfrujące
 - Colossus, 38
 - Enigma, 33
 - scytały, 21
 - ustanawianie bezpiecznego połączenia, 164
 - usuwanie plików, 200
 - uwierzalnianie klienta, 164
 - hasło, 165
 - klucze publiczne klienta, 165
 - pliki identyfikujące, 164
 - uwierzalnianie poprzez szyfrowanie, 287
 - uwierzalnianie serwera, 164
 - użytkowanie aktywne, 129
 - użytkowanie pasywne, 129
- ## V
- V*, 48
 - Ventris Michael, 303
 - Vernam Gilbert, 30
 - Vigenere Blaise, 27
 - VMPC, 112
- ## W
- W*, 48
 - WAKE, 111
 - walec odwracający, 34
 - WampServer, 231
 - wariant, 50
 - Web of Trust, 201, 203
 - skompromitowanie klucza, 202
 - ścieżki zaufania, 202

- wektor inicjujący, 99, 105
 - jako licznik, 105
 - jednorazowy, 106
 - losowy, 105
 - whitening, 271
 - wiadomość e-mail
 - podpisy elektroniczne, 220
 - podpisywanie, 207
 - szyfrowanie, 207
 - wiązanie bloków zaszyfrowanych, 99
 - wide-mouth frog, 115
 - wielokrotne wielomianowe sito kwadratowe, 66
 - dla podwójnie dużych liczb, 66
 - wirtualne szyfratory, 38
 - współczynnik zbieżności, 75
 - współdzielenie plików, 198
 - współdzielenie sekretu, 121
 - wybór kryptosystemu, 148
 - wycofanie klucza, 130
 - względnie bezpieczny, 149
 - wzór kongruencyjny, 107
- X**
- XEX, 105
 - Xiao Steganography, 146
 - XOR, 72
 - XPD/KPD, 111
 - XTS-AES, 105
- Y**
- Young Thomas, 295
- Z**
- zabezpieczanie połączeń internetowych, 154
 - protokół SSH, 162
 - protokół TLS, 154
 - zabezpieczenie komunikacji serwer-klient, 177
 - zakres permutacji, 254
 - zapadka, 52
 - zapis ideograficzny, 292
 - zapis sylabiczny, 292
 - zarządzanie kluczami publicznymi
 - użytkowników, 127
 - zasada Kerckhoffa, 148
 - zasada nieoznaczoności Heisenberga, 132
 - zasady, 138
 - zaszyfrowane archiwum, 182
 - zawartość informacyjna języka, 85
 - zbiór relacji, 50
 - zbiór reszt modulo n , 52
 - zbiór znaków
 - rozłączny, 48
 - równy zbiorowi liter alfabetu, 48
 - szyfrogramu, 49
 - tekstu jawnego, 48, 49
 - tekstu zaszyfrowanego, 48
 - V, 48
 - W, 48
 - zdarzenia, 73, 74
 - elementarne, 73
 - prawdopodobieństwo, 73
 - zegar
 - znaczenie w kryptografii, 116
 - zero-knowledge proofs, 125
 - ziarno, 86, 94
 - generatorów z rejestrem przesuwającym, 108
 - Zimmerman Phil, 42
 - zjawiska losowe, 110
 - złożoność algorytmów, 82
 - czasowa, 82
 - przestrzenna, 82
 - złożoność systemu, 242
 - zmęczenie klucza, 129
 - zmienna addytywna, 248
 - znacznik czasu, 123
 - z datą przyszłą, 124
 - znacznik EOF, 146
 - znacznik końca pliku, 146
 - znaki wodne, 145
 - algorytm LSB, 145
 - algorytm Patchwork, 145
 - tworzenie, 145
 - znacznik końca pliku, 146
 - znakowanie czasowe, 123
 - funkcje skrótu, 123
 - protokół arbitrażowy, 124
 - protokół dystrybucyjny, 124
 - protokół łączący, 124
 - zobowiązania bitowe, 126
 - Zygalski Henryk, 35
- Ż**
- żądanie
 - certyfikatu, 159
 - podpisania certyfikatu, 158
 - Request Hello, 155

PROGRAM PARTNERSKI

— GRUPY HELION —



1. ZAREJESTRUJ SIĘ
2. PREZENTUJ KSIĄŻKI
3. ZBIERAJ PROWIZJĘ

Zmień swoją stronę WWW w działający bankomat!

Dowiedz się więcej i dołącz już dzisiaj!

<http://program-partnerski.helion.pl>

GRUPA
Helion

Kryptografia to dziedzina nauki, której sedno stanowią sposoby bezpiecznego przekazywania informacji. Jest ona niemal tak stara jak nasza cywilizacja, a dziś rozwija się w sposób niezwykle dynamiczny. Gdy tylko narodziły się pierwsze metody zapisu i komunikowania się, pojawiła się też konieczność zabezpieczenia informacji przed tymi, którzy mogliby wykorzystywać je na niekorzyść osób nimi dysponujących. Od bezpieczeństwa ważnych informacji zależały często losy całych państw i narodów. O rozstrzygnięciach wielkich bitew nierazko decydowały inteligencja i determinacja pojedynczych osób, które potrafiły skutecznie szyfrować (albo deszyfrować) nadawane (lub przechwytywane) komunikaty.

O tej fascynującej dziedzinie wiedzy opowiada książka *Podstawy kryptografii. Wydanie III*. Wprowadza ona czytelnika w podstawowe zagadnienia kryptografii bez przygniataania nadmiarem teorii i skomplikowaną matematyką. Kusi za to barwnymi opisami i pasjonującymi przykładami „kryptograficznych wojen”. Można dzięki niej poznać historię rozwoju technik szyfrowania informacji, matematyczne podstawy kryptografii, stojącą za nią teorię oraz praktyczne zastosowania tej nauki. Niezależnie od tego, czy chcesz poznać kryptografię na własny użytek, potrzebujesz tej wiedzy do celów zawodowych, książka ta jest doskonałym przewodnikiem po świecie szyfrów, kluczy i algorytmów zabezpieczających dane. Znajdziesz w niej informacje na temat protokołów SSL i SSH, a także szczegółowy opis algorytmu SHA3.

- _ Przegląd klasycznych sposobów szyfrowania
- _ Matematyczne podstawy kryptografii
- _ Praktyczne zastosowanie mechanizmów matematycznych
- _ Teoria kryptoanalizy i informacji oraz jej praktyczne wykorzystanie
- _ Przegląd protokołów kryptograficznych
- _ Klucze publiczne i prywatne
- _ Zasady zabezpieczania danych, połączeń i systemów komputerowych
- _ Potwierdzanie tożsamości za pomocą podpisów elektronicznych
- _ Zabezpieczanie stron internetowych i szyfrowanie baz danych

Naucz się chronić cenne dane!

	<i>Sprawdź nasze szkolenia!</i>	KOD KORZYŚCI Sięgnij po więcej! ▶ 
 helion.pl	 SZKOLENIA AKADEMIA IT & BUSINESS	ISBN 978-83-283-8569-6
 HELION SA ul. Kościuszki 1c 44-100 Gliwice tel.: 32 230 98 63 helion@helion.pl	HELIONSZKOLENIA.PL	 9 788328 385696
INFORMATYKA W NAJLEPSZYM WYDANIU		Cena: 79,00 zł