ZESZYTY NAUKOWE POLITECHNIKI RZESZOWSKIEJ 296, Elektrotechnika 36
RUTJEE, z. 36 (3/2017), październik-grudzień 2017, s. 53-58

Mateusz TYBURA[1]

ANALIZA MOŻLIWOŚCI ATAKU CZASOWEGO ORAZ SŁOWNIKOWEGO NA KOMUNIKACJĘ Z UŻYCIEM KRYPTOGRAFII ELIPTYCZNEJ

Przez tysiąclecia tworzono, udoskonalano i łamano dziesiątki rozwiązań, których jedynym celem było uniemożliwienie odczytania informacji przez postronnych. Doprowadziło to do powstania dwóch przeciwstawnych w swoich działaniach dziedzin – kryptografii i kryptoanalizy. W dobie komputerów zrezygnowano ze wszystkich dotychczasowych rozwiązań i wprowadzono zupełnie nowe, z których za najbezpieczniejsza można uznać RSA i szyfry oparte o krzywe eliptyczne. Oba są uznawane za niemożliwe do złamania. Wynika to bezpośrednio z zależności matematycznych użytych w ich definicji. W dotychczasowych badaniach wykazano już kilka ich słabości, lecz nadal nie ma rozwiązania, które działałoby w każdym jednym przypadku. Z uwagi na to postanowiono przyjrzeć się głębiej słabym punktom szyfrów eliptycznych z uwzględnieniem wszystkich dotychczas dostępnych informacji.

Słowa kluczowe: kryptografia, krzywe eliptyczne, ataki słownikowe, ataki czasowe.

1. Wstęp

Od początków istnienia ludzkiego rodzaju istniały obawy dotyczące bezpieczeństwa i prywatności. Z tego powodu, przez lata, włożono wiele wysiłków w rozwój rzeczy, takich jak kryptografia czy steganografia. Obie te idee skoncentrowano się na ukrywaniu informacji przed tymi, którzy nie mogą czytać ich, a jednocześnie pozwalają na odczyt tym, którzy mają odpowiednie uprawnienia. Wartym zauważenia jest fakt, iż umiejętność pisania i czytania, mogła stanowić jedną z pierwszych w historii ludzkości technik ukrywania treści przed postronnymi. Wynikało to z nikłego rozpowszechnienia się tychże umiejętności w czasach przed wprowadzeniem powszechnie dostępnej edukacji. Gdy ta sytuacja uległa zmianie, wszyscy kryptologowie musieli opracować bardziej skomplikować sposoby ukrywania informacji.

W starożytnej Grecji na przykład zaprojektowano szyfr, który wykorzystywał drewniany kij o pewnej średnicy i pas skórzany do szyfrowania i odszy-

[1] Mateusz Tybura, Politechnika Rzeszowska, adres e-mail: tyburam@hotmail.com

frowywania wiadomości [1]. Z kolei z imperium rzymskiego pochodzi metoda zwana szyfrem Cezara [1]. Innym ważnym przykładem działań było opracowanie w jednym z państw arabskich dziedziny zwanej kryptoanalizą. Koncentruje się ona na zastosowaniu metod matematycznych oraz znajomości języków w odszyfrowywaniu uprzednio zaszyfrowanych treści. rodzajów analiz z wykorzystaniem znajomości języków i statystyk [1].

Przez wiele kolejnych lat algorytmy rozwijano poprzez np. używanie więcej niż jednego alfabetu, usuwanie znaków białych, czy też budowanie pewnych urządzeń mechanicznych lub elektronicznych. Każdemu kolejnemu twórcy przyświecał jeden cel – uczynienie algorytmu niemożliwym do złamania.

Ważnym krokiem w rozwoju kryptografii było zastosowanie komputera. Opracowano kolejno algorytmy DES, AES, a następnie jedne z najsilniejszych będących aktualnie w użyciu rozwiązań – RSA i szyfrowanie oparte o krzywe eliptyczne. Żadnego z tych dwóch nie udało się jak dotąd w sposób uniwersalny złamać. Obydwa łączy też zastosowanie kosztownych obliczeniowo obliczeń na dużych liczbach pierwszych.

Wymienione w tytule artykułu krzywe eliptyczne były znane przez matematyków od setek lat, nim zastosowano je do szyfrowania. Dopiero w późnych latach 80-tych matematycy Neal Koblitz oraz Victor Miller niezależnie od siebie zaproponowali ich zastosowanie w kryptografii asymetrycznej [2]. Po paru latach znalazły swoje zastosowanie w programach komercyjnych i kilku otwarto źródłowych.

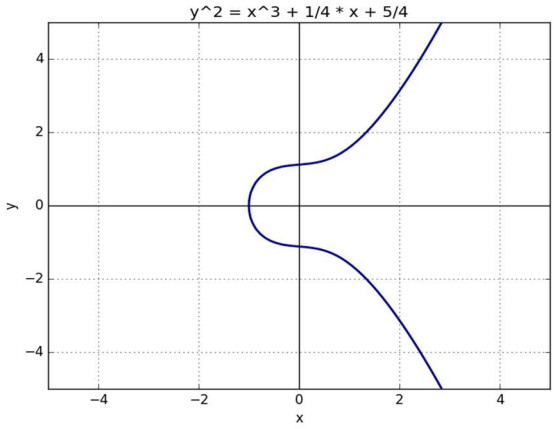
Krzywe eliptyczne *E* nad polem *K* w wielomianowej postaci równania Weierstrasse (1)

$$E: y^2+\mathrm{a}_1\mathrm{xy}+\mathrm{a}_3\mathrm{y}=\mathrm{x}^3+\mathrm{a}_2x^2+\mathrm{a}_4\mathrm{x}+\mathrm{a}_6 \tag{1}$$

gdzie: $a_1,\mathrm{a}_2,\mathrm{a}_3,\mathrm{a}_4,\mathrm{a}_6 \in K$,

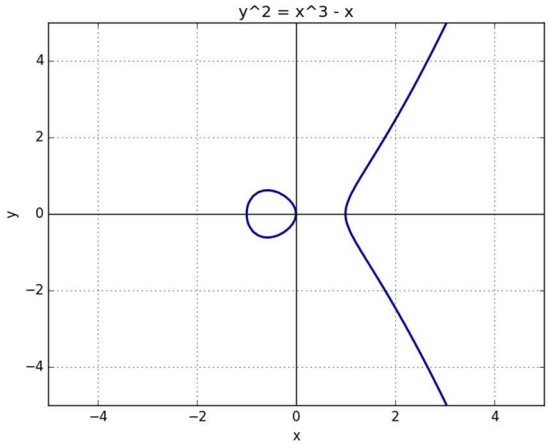
$$\Delta=-d_2^2d_8-8d_4^3-27d_6^2+9d_2d_4d_6,$$
$$\Delta \neq 0,$$
$$d_2=\mathrm{a}_1^2+4a_2,$$
$$d_4=2a_4+\mathrm{a}_1a_3,$$
$$d_6=\mathrm{a}_3^2+4a_6,$$
$$d_8=\mathrm{a}_1^2a_6+4a_2a_6-a_1a_3a_4+\mathrm{a}_2a_3^2-a_4^2.$$

Wartym zauważenia jest to, iż nazwano je w sposób mylący. Ich kształt, po narysowaniu w dwuwymiarowym układzie kartezjańskim nie jest bynajmniej elipsą (rys.1), choć zdarza się, iż w pewnym miejscu prostej występują koliste kształty (rys. 2).



Rys. 1. Przykład krzywej eliptycznej

Fig. 1. An example of elliptic curve



Rys. 2. Przykład krzywej eliptycznej

Fig. 2. An example of elliptic curve

2. Problemy

Głównym problemem związanym z analizą krzywych zarówno pod kątem ich zabezpieczania jak i łamania, jest trudność w pojęciu wszystkich aspektów matematycznych związanych z ich teorią oraz wszystkie rozważania niezbędne do ich zaimplementowania na komputerze.

Z uwagi na to, do tej pory stosowano przede wszystkim ataki typu side-channel [3]. Wykorzystywały one pewną wadę w pierwszych implementacjach szyfrów eliptycznych, polegającą na zróżnicowaniu czasu obliczeń, w zależności od użytego klucza. Bazując zatem wyłącznie na analizie odstępów czasowych, możliwym było częściowe lub całkowite przewidzenie zastosowanego klucza.

Choć problem czasu obliczeń rozwiązano, to atak czasowy wydaje się nadal w pewnym stopniu wykonywalny. Jedną z przesłanek ku temu, jest istnienie wielu krzywych, z różnymi możliwymi do zastosowania parametrami. W związku z czym potencjalnie pozostawiono furtkę, mogąca, ujawnić, np. którą z krzywych eliptycznych zastosowano, jeśli tylko czas obliczeń jest unikatowy albo choć trochę odmienny dla różnych z nich. bezpieczeństwa jak łamania jakiejkolwiek krzywej eliptycznej jest określenie, jak one działają i jakie są ich słabości. Po tym jest kluczowe, aby dowiedzieć się, że są one stosowane, aby skupić się na ich łamanie, zamiast używać złych metod odszyfrowywania danych zaszyfrowanych przy użyciu innych metod.

Kolejnym potencjalnym zagrożeniem jest rozwiązanie problemu dyskretnego logarytmu [4], występujące również w przypadku stosowania algorytmu RSA. Jego obecność wynika z zależności pomiędzy kluczem prywatnym a publicznym. Sposobu wyznaczania dyskretnego logarytmu dla dowolnie wielkich liczb, nadal nie opracowano, jednakże może to kiedyś nastąpić. Krzywe eliptyczne mają w tym temacie kilka specyficznych dla siebie ograniczeń. Jednym z nich jest chociażby fakt, że wszystkie wartości, którymi się operuje, muszą leżeć na krzywej. W związku z czym ilość ewentualnych rozwiązań jest znacznie mniejsza niż gdyby brano pod uwagę, np. całą przestrzeń liczb rzeczywistych. Co więcej, wiele, o ile nie wszystkie z dotychczas znanych, krzywych eliptycznych dogłębnie opisano. Zatem możliwe jest wcześniejsze przygotowanie odpowiednich pod kątem konkretnego ataku danych, celem np. odszyfrowania tajnych danych lub uniemożliwienia nawiązania stabilnego połączenia.

Ostatecznie koniecznym do wzięcia pod uwagę jest błąd czynnika ludzkiego. W związku, z którym, możliwym jest wprowadzenie do kodu źródłowego celowego lub zupełnie niezamierzonego błędu, w wyniku którego uzyskano by łatwiejszy dostęp do zaszyfrowanych danych.

3. Atak

Z uwagi na potencjalną trudność w wyznaczeniu metody do bezpośredniego odwracania wartości uzyskanych w procesie szyfrowania na tekst jawny postanowiono rozważyć możliwość przeprowadzenia ataku słownikowego. W tym celu zaprojektowano program komputerowy, które zadaniem było generowanie par kluczy szyfrujących oraz ich zapisanie w wyznaczonym miejscu na dysku.

Ponieważ chciano uzyskać maksymalnie dokładny pomiar zdecydowano się na napisanie programu w języku C++ z użyciem biblioteki *crypto++*. W ten sposób uzyskano pewność, że mierząc czas, uzyskiwano wyłącznie ten związany z koniecznymi do wykonania obliczeń, a nie np. wynikający z różnego rodzaju działań wykonywanych przez maszynę wirtualną, czy kompilację JIT (ang. *Just-in-time*, w locie).

Dla celów uzyskania miarodajnej statystyki cały proces powtórzono kilka razy, a w każdym z powtórzeń klucze generowano po milion razy. Przeanalizowano następnie wszystkie uzyskane dane pod względem częstotliwości występowania poszczególnych kluczy. Szukano w ten sposób potencjalnych powtórzeń, świadczących jasno o stosunkowo słabym algorytmie generatora liczb pseudolosowych zastosowanego w badanej bibliotece. W wyniku pomiarów częstotliwości występowania par kluczy, nie udało się odnaleźć ani jednego powtórzenia, dla żadnego z wygenerowanych zbiorów. Z uwagi na to, uznano, iż generator liczb pseudolosowych uczyniono wystarczająco silnym.

Kolejnym krokiem było wyliczenie wartości średniej, odchylenia standardowego, wartości pierwszego, drugiego i trzeciego kwantyla oraz wartości maksymalnej i minimalnej, jeśli chodzi o czas generowania par kluczy (tab. 1).

Tabela 1. Pomiary czasu generowania par kluczy

Table 1. Time of key pair generation process

Mierzona wartość	**Czas [μs]**
Średnia	1834.090524
Odchylenie standardowe	98.253478
Minimum	1541.000000
Pierwszy kwantyl	1776.000000
Drugi kwantyl	1823.000000
Trzeci kwantyl	1873.000000
Maksimum	4736.000000

W pomiarach zaobserwowano bliskość zdecydowanej większości czasów trwania do wartości średniej. Tylko i wyłącznie w przypadku maksimum dostrzeżono olbrzymią odległość od średniej. W każdym innym przypadku mieszczono się w najwyżej trzykrotności wartości odchylenia standardowego.

Z uwagi na uzyskane wyniki rozważono dalsze rozszerzenie działań, aż do momentu, gdy uzyskano by kompletny słownik wszystkich możliwych do uzyskania, dla zadanej krzywej, pary kluczy. Wybrano krzywą Curve2213 (M-221), o module p równym $2^{221} - 3$. Na czas rozważań pominięto ograniczenia

związane z zależnością pomiędzy kluczem prywatnym a publicznym oraz uwzględniono możliwość uzyskania kolizji. Z związku z czym ilość potencjalnych par oszacowano na $4e^{246}$. W wyniku pomiarów ustalono, że wygenerowanie miliona par zajmuje 1834090524 μs, czyli 1834.09052 s, a to z kolei 30.5681754 minut. Po uwzględnieniu szacowanej ilości par całkowity konieczny czas wyniósłby $2.326345e^{236}$ lat.

4. Wnioski

W wyniku analizy potencjalnych zagrożeń oraz uzyskanych pomiarów uzyskano jednoznaczny dowód, nie tylko na poprawę wymienionych w drugim rozdziale problemów czasowych, ale także pewność, iż niemożliwym jest przeprowadzenie ataku słownikowego z użyciem wyłącznie komputera osobistego.

Literatura

[1] S. Vaudenay, "A Classical Introduction to Cryptography: Applications for Communications Security", ISBN 9780387258805, Springer, 2005

[2] D. Hankerson, A. Menezes, S. Vanstone, "Guide to Elliptic Curve Cryptography", Springer, 2004

[3] E. Brier, M. Joye, "Weierstraß Elliptic Curves and Side-Channel Attacks", Public Key Cryptography, vol. 2274 of Lecture Notes in Computer Science, pp. 335–345, Springer-Verlag, 2002

[4] M. Musson, "Attacking the Elliptic Curve Discrete Logarithm Problem", Acadia University Master thesis, Spring Convocation, 2006

ANALYSIS OF THE POSSIBILITY OF THE TIME AND DICTIONARY BASED ATTACKS ON ELLIPTIC CURVE CRYPTOGRAPHY BASED COMMUNICATION

S u m m a r y

For millennia, dozens of solutions, which sole purpose was to prevent outsiders from reading information, have been developed, refined and broken. This led to the emergence of two opposing fields - cryptography and cryptanalysis. In the age of computers, all existing solutions have been abandoned and new ones have been introduced, with the most secure ones RSA and ciphers based on elliptic curves. Both considered impossible to break. This result directly from the math used in their definitions. Some previous researches have already shown some of their weaknesses, but there is still no solution that would work in every single case. Because of this, it was decided to take a closer look at the weak points of elliptic ciphers, taking into account all the information available to date.

Keywords: cryptography, elliptic curves, dictionary attacks, time attacks

DOI: 10.7862/re.2017.17

Tekst złożono w redakcji: wrzesień 2017
Przyjęto do druku: październik 2017