A jednak się da (I),

czyli saga kryptologiczna w odcinkach.
Tym razem: o kryptografii klucza publicznego i podpisie cyfrowym.

Tomasz KAZANA

Problem szyfrowania przesyłanych wiadomości sięga jeszcze czasów starożytnych, więc naszego nowego deltowego cyklu artykułów o kryptologii nie możemy nie zacząć od przypomnienia najstarszego znanego systemu szyfrowania, mianowicie szyfru Cezara. Szyfr ten nie jest specjalnie wyrafinowany. Po prostu umawiamy się, że (na przykład) zamiast litery A będziemy pisać G; zamiast B – literę H; zamiast C – literę I i tak dalej, to znaczy: zamiast n-tej litery alfabetu będziemy pisać $(n+6)$-tą literę. (Tak naprawdę $(((n+5) \bmod 26)+1)$-tą literę, gdyż pod koniec musimy alfabet cyklicznie zawinąć do początku.) Równie dobrze moglibyśmy alfabet przesuwać nie o 6 pozycji, a o 3 pozycje, o 17 pozycji, czy ogólnie: o K_{Cezar} pozycji.

Przyjmujemy, że alfabet ma 26 liter.

Nieco bardziej skomplikowany jest szyfr permutacyjny. Tutaj dwie strony ustalają zawczasu permutację literek

$$K_{\text{perm}} : \{A, B, \ldots, Z\} \to \{A, B, \ldots, Z\},$$

a następnie – aby zaszyfrować tekst – przykładają tę permutację do kolejnych literek wiadomości, uzyskując *szyfrogram*, czyli tekst zaszyfrowany. Odszyfrowanie wygląda podobnie, tylko przykładamy kolejno permutację odwrotną do K_{perm}.

Rozwiązanie zadania M 1581.
(a) *Odpowiedź:* Nie.

Przypuśćmy, że taki zbiór S istnieje. Aby liczby 0 oraz 1 miały przedstawienia w postaci odpowiednich sum, musimy mieć $0, 1 \in S$. Gdyby $2 \in S$, to mielibyśmy dwa przedstawienia tej liczby jako sumy elementów S: $2 = 1+1 = 0+2$, a zatem $2 \notin S$. Podobne rozważania prowadzą kolejno do wniosków, że $3 \in S$, $4 \notin S$, $5 \in S$. Ale $6 = 3+3 = 5+1$, sprzeczność.

(b) *Odpowiedź:* Tak.

Niech $S = 2\mathbb{N}_0 \cup \{1\} = \{0, 1, 2, 4, 6, \ldots\}$. Wówczas każda liczba nieparzysta większa od 1 (czyli każdy element zbioru $\mathbb{N}_0 \setminus S$) postaci $2n+1$ ma dokładnie jedno przedstawienie w postaci sumy dwóch elementów S, mianowicie $2n$ oraz 1.

Oba powyższe przykłady wpisują się w pewien bardzo ogólny schemat szyfrowania. Mianowicie, na początku, jeszcze przed jakimkolwiek szyfrowaniem, dwie strony (ochrzcijmy je imionami Aldona i Bogumił) muszą się spotkać i ustalić coś sekretnego, czyli tak zwany klucz kryptograficzny K, w naszym przykładach, odpowiednio: K_{Cezar} oraz K_{perm}. Następnie, gdy Aldona chce przekazać Bogumiłowi wiadomość m, musi ona najpierw obliczyć wartość $c = \texttt{Enc}(K, m)$ (gdzie `Enc` jest specyficznym opisem wybranego szyfru) i wysłać właśnie ją do Bogumiła. Bogumił otrzymawszy szyfrogram c, dokonuje obliczenia $m' = \texttt{Dec}(K, c)$, gdzie `Dec` oznacza funkcję deszyfrującą. Oczywiście, żeby wszystko miało ręce i nogi, musi zawsze (dla każdego wyboru K) zachodzić $m' = m$ oraz – ze względów bezpieczeństwa – podsłuchanie szyfrogramu c przez niecną Helgę nie może ujawnić właściwego komunikatu, czyli m. Ten ostatni wymóg nie jest wcale banalny do sformalizowania, ale my pozostaniemy przy nieformalnym określeniu – chcemy po prostu, aby dla Helgi było *bardzo trudno* wywnioskować cokolwiek na temat m z wartości c. Dodatkowo w środowisku kryptologicznym przyjmuje się, że Helga zawsze wie, jakiego systemu używają Aldona z Bogumiłem – innymi słowy zna definicję `Enc`, a jedyne czego nie zna, to klucz K (to założenie to treść tak zwanej zasady Kerckhoffsa).

Więcej o samej zasadzie Kerckhoffsa oraz motywacji jej przyjęcia pisaliśmy w *Migawce* w Δ^{7}_{18}.

Przez setki lat rozwój kryptologii przebiegał dokładnie według powyższego schematu. Wymyślano przeróżne szyfry, po czym próbowano się upewnić, czy aby na pewno znajomość $\texttt{Enc}(K, m)$ (oraz definicji `Enc`) nic nie mówi o m. Okazało się dość szybko, że zarówno szyfr Cezara, jak i permutacyjny, nie są bezpiecznymi szyframi, ale za to inne współczesne szyfry, np. AES czy Triple-DES, uchodzą za bezpieczne i są stosowane powszechnie do dziś.

Wszystko pięknie, ale widać, gdzie znajduje się spora niewygoda. Żeby szyfrować, trzeba bezpiecznie ustalić wspólny tajny klucz. Zadanie to wcale nie wydaje się banalne (szczególnie, gdy Aldona i Bogumił mieszkają na różnych kontynentach, trwa wojna etc.), więc można by sformułować następujące naiwne marzycielskie pytanie:

Czy da się skonstruować szyfr taki, który nie wymaga ustalania wspólnego tajnego klucza, to znaczy, aby po prostu $c = \texttt{Enc}(m)$?

Łatwo wykazać, że postulat powyższy nie jest możliwy do zrealizowania.

Dowód. Załóżmy przeciwnie, a więc, że takie `Enc` istnieje. Wówczas:

1. Oczywiście Bogumił musi być w stanie deszyfrować przychodzące do niego szyfrogramy, a więc:
2. Istnieje taka funkcja `Dec`, że $\texttt{Dec}(\texttt{Enc}(m)) = m$; oznacza to, że:
3. `Dec` jest funkcją odwrotną do `Enc`; pamiętajmy, że:
4. Helga zna definicję `Enc` (zasada Kerckhoffsa).
5. Wystarczy więc, że Helga obliczy funkcję odwrotną do (znanej jej!) funkcji `Enc` i już jest w stanie robić to, co Bogumił, czyli deszyfrować.
6. Niestety, oznacza to, że szyfrowanie `Enc` **nie** jest bezpieczne, co kończy dowód.

Pomimo znajomości powyższego rozumowania w roku 1976 dwóch dżentelmenów – Whitfield Diffie oraz Martin Hellman – postanowiło (w pracy *New Directions in Cryptography*) bezczelnie zapytać: a może jednak się da? Więcej, już rok później panowie Ron Rivest, Adi Shamir oraz Leonard Adleman (korzystając ze wskazówek z pracy D–H) wykazali, że faktycznie: da się! Czytelnik Logiczny powinien w tym momencie poczuć co najmniej niesmak. Akapit wyżej pokazaliśmy *dowód*, że czegoś się nie da, a dalej coś tam jeszcze dywagujemy? Haczyk – jak zwykle w matematyce – tkwi w założeniach. Diffie i Hellman nie wykazali, że powyższy dowód jest błędny. Zaproponowali tylko pewne *dodatkowe* założenie, przy którym ten dowód *przestaje* być poprawny! Konkretnie chodzi o punkt 5. tego dowodu. Jeśli założymy (jak proponują D–H), że Helga ma ograniczoną moc obliczeniową, to wcale nie jest jasne, że będzie potrafiła szybko odwrócić znaną sobie funkcję `Enc`. Tym tropem poszli Rivest, Shamir i Adleman, proponując swój szyfr, znany dziś jako szyfrowanie RSA.

Zanim przejdziemy do szczegółów, spróbujmy naświetlić samą esencję pomysłu. Idea opiera się na spostrzeżeniu, że (generalnie) mnożenie jest szybkie, ale już jego odwracanie (czyli rozkład na czynniki pierwsze) – bardzo trudne. Innymi słowy, chodzi o znalezienie takich funkcji `Enc` oraz `Dec`, aby do wyliczania `Enc` wystarczała znajomość $p \cdot q$, ale już do szybkiego obliczania `Dec` – konieczne były zarówno p, jak i q. Wówczas Bogumił może wszystkim (i Aldonie, i Heldze, i Kamili, i Markowi, i Szymonowi i każdemu, kto tylko będzie chciał) bez skrępowania ujawniać opis `Enc`, ALE zachowując dla siebie czynniki p oraz q (czyli w istocie opis `Dec`). Naprawdę jest to troszkę bardziej skomplikowane i wygląda tak.

Rzeczywiście Bogumił wybiera dwie duże (rzędu setek cyfr) liczby pierwsze p i q oraz oblicza ich iloczyn $n = p \cdot q$. Następnie oblicza $\phi(n) = (p-1) \cdot (q-1)$ oraz losuje dowolną liczbę e względnie pierwszą z $\phi(n)$. Teraz znajduje taką liczbę d, że

$$(*) \qquad e \cdot d = 1 \bmod \phi(n).$$

Okazuje się, że powyższa operacja (wyznaczenie d) da się wykonać bardzo szybko (np. za pomocą rozszerzonego algorytmu Euklidesa), gdy znamy p, q oraz e, natomiast wierzymy, że jest bardzo trudna, gdy znamy tylko n oraz e. Pozostaje opisać funkcję szyfrującą i deszyfrującą ($m, c \in \{0, 1, 2, \ldots, n-1\}$):

$$\texttt{Enc}(m) := m^e \bmod n$$

$$\texttt{Dec}(c) := c^d \bmod n.$$

Aby uznać szyfrowanie RSA za poprawne i bezpieczne, należy sprawdzić, że:

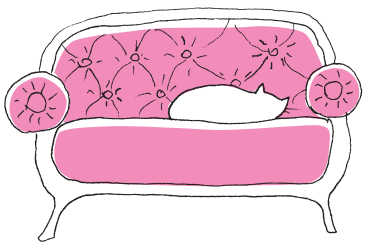
(a) $\texttt{Dec}(\texttt{Enc}(m)) = m$, czyli, że $m^{ed} = m^1 \bmod n$. Dowód tego faktu pominiemy, pozostawiając tylko uwagę, że – korzystając z kilku faktów z teorii liczb – da się go wyprowadzić wprost z równania $(*)$.

Rozwiązanie zadania F 961.
W wyniku zjawiska fotoelektrycznego na kulce zbiera się ładunek dodatni, a pochodzące od niego pole hamuje fotoelektrony. Wielkość ładunku zależy od pojemności kulki i jej potencjału $\phi : q = 4\pi\varepsilon_0 r\phi$. Maksymalny potencjał kulki $\phi_{\max}$ zależy od początkowej energii kinetycznej fotoelektronów. Ponieważ przyrost energii kinetycznej elektronów jest równy pracy sił pola kulki, więc przyjmując, że potencjał pola kulki i prędkość elektronów w nieskończoności wynoszą zero, dostajemy: $\Delta W_{\text{KIN}} = -e\phi_{\max}$ (gdzie e ładunek elektronu). Stąd $mv_{\max}^2/2 = -e\phi_{\max}$ (gdzie $v_{\max}$ maksymalna energia fotoelektronu) a $\phi_{\max} = mv_{\max}^2/2e$. Zgodnie ze wzorem Einsteina dla zjawiska fotoelektrycznego $mv_{\max}^2/2 = h\nu - A$ (gdzie ν częstość światła). Ostatecznie

$$\phi_{\max} = \frac{h\nu - A}{e} = \frac{(hc/\lambda) - A}{e} = \\ = 4{,}4 \text{ V}.$$

Wspomniany Czytelnik Logiczny proszony jest o poszukanie na marginesach tego numeru trójkąta o trzech kątach prostych.

Samo wyliczanie funkcji `Enc` musi być szybkie, żeby szyfrowanie było efektywne.



Co ciekawe, wciąż pozostaje pytanie otwarte, czy założenie o trudności problemu faktoryzacji wystarcza, aby udowodnić ściśle bezpieczeństwo RSA. Teoretycznie mogłoby się zdarzyć, że ktoś złamie RSA, ale w jakiś bardzo dziwny sposób, niewymagający po drodze rozkładu n na czynniki.

(b) Znajomość opisu `Enc` (czyli liczb n i e) nie pomoże odtworzyć funkcji `Dec`, gdyż – jak już wspomnieliśmy – wierzymy, że z n i e nie da się łatwo obliczyć d, które jest niezbędnym elementem opisu `Dec`.

Jak to wygląda w praktyce?

RSA jest przykładem *szyfru z kluczem publicznym*

Szyfrowanie RSA jest niezwykle popularne i wygodne. Aby z niego korzystać, należy postąpić dokładnie tak, jak Bogumił w opisie wyżej. Następnie parę (n, e) (tak zwany klucz publiczny) należy rozgłaszać wszem i wobec, ponieważ jej znajomość umożliwia światu bezpieczne wysyłanie komunikatów przeznaczonych tylko dla nas. Z drugiej strony – liczbę d (osobisty klucz prywatny) oraz rozkład p i q należy trzymać w wielkiej tajemnicy i nigdy nikomu nie zdradzać.

Jaki to ma związek z podpisem cyfrowym?

Podpis cyfrowy Bogumiła to taka para funkcji $(\texttt{Sign}, \texttt{Check})$, że:

- tylko Bogumił potrafi obliczać wartości $s = \texttt{Sign}(m)$ (czyli podpisywać m);
- każdy potrafi obliczać wartości $\texttt{Check}(m, s) \in \{T, F\}$ (czyli sprawdzać autentyczność podpisu s pod dokumentem m);
- funkcja `Check` zachowuje się dobrze, to znaczy $\texttt{Check}(m, \texttt{Sign}(m)) = T$ oraz $\texttt{Check}(m, x) = F$, jeśli tylko $x \neq \texttt{Sign}(m)$.

Dziś chyba każdy wie, że T to *true*, a F to *false*.

Powyższa definicja wydaje się zupełnie intuicyjna i nie wymaga szerszego komentarza. Wyzwaniem jest jednak znaleźć odpowiednie `Sign` oraz `Check`. Okazuje się, że znajomość RSA bardzo nam pomoże. Przyjmijmy bowiem:

$$\texttt{Sign}(m) = m^d \bmod n$$
$$\texttt{Check}(m, s) = T \text{ wtedy i tylko wtedy, gdy } s^e = m \bmod n,$$

gdzie n, e oraz d są takie same jak w RSA (w szczególności: n i e są publiczne, a d – znane tylko Bogumiłowi).

Innymi słowy: podpis `Sign` zachowuje się jak `Dec`, a weryfikacja podpisu s dokumentu m sprowadza się do sprawdzenia, czy $\texttt{Enc}(s) = m$.

Teraz widać, że bezpieczeństwo takiego podpisu jest takie samo, jak bezpieczeństwo RSA. Jeśli bowiem (w RSA) tylko Bogumił mógł deszyfrować, to u nas tylko on może podpisywać. Skoro wcześniej każdy mógł szyfrować (`Enc` jest publiczna), to teraz każdy może obliczać funkcję `Check`. Wreszcie – funkcja `Check` zachowuje się zgodnie z oczekiwaniami, co wynika wprost z faktu, że `Dec` jest odwrotna do `Enc`.

Postscriptum

Na zakończenie jeszcze mała obserwacja, która na razie pewnie wyda się mało użyteczna (choć zaryzykujemy stwierdzenie, że jest elegancka i zaskakująca), ale będziemy jej potrzebować w kolejnych częściach cyklu *A jednak się da*. Otóż chcemy zdefiniować protokół tak zwanego *ślepego podpisu*. To znaczy: chcielibyśmy, aby Bogumił mógł (jeśli tylko ma taką fantazję) pozwolić Aldonie na podpisanie czegoś, czego sam nigdy nie zobaczy na oczy. Możemy zrobić to tak:

1. Aldona wybiera dokument do podpisu m oraz losową liczbę $x \in \{1, 2, \ldots, n\}$;
2. Aldona wysyła do Bogumiła liczbę $c = m \cdot x^e \bmod n$;
3. Bogumił oblicza $a = c^d \bmod n$ i odsyła wynik Aldonie;
4. Aldona oblicza $s = a/x \bmod n$.

Jako ćwiczenie pozostawiamy sprawdzenie, że w protokole powyżej:
(1) Bogumił nie jest w stanie odtworzyć dokumentu m
oraz (2) liczba s jest poprawnym podpisem Bogumiła pod dokumentem m.

Rozwiązanie zadania F 962.
Energia fotonu wyraża się wzorem $E = h\nu$, a jego pęd $p = h\nu/c$. Jeżeli na powierzchnię folii w ciągu sekundy pada N fotonów, to moc światła padającego wynosi $W = Nh\nu$. Ciśnienie wytwarzane przez padające fotony wynosi $P = N\Delta p$, gdzie Δp – zmiana pędu fotonu przy odbiciu. Stąd

$$P = N2p = 2Nh\frac{\nu}{c} = \frac{2W}{c}.$$

Zgodnie z drugą zasadą dynamiki $Pt = mu$, gdzie u – prędkość uzyskana przez folię. Stąd $u = 2Wt/mc = 5 \cdot 10^{-3}$ m/s.