
# Estymacja kosztów łamania systemu kryptograficznego

Andrzej Chmielowiec

andrzej.chmielowiec@cmmsigma.eu

Centrum Modelowania Matematycznego Sigma

# Plan prezentacji

- Wprowadzenie
- Długość kluczy, a bezpieczeństwo
- Koszt ataku
- Warunki na nieopłacalność ataku
- Podsumowanie

# Komu może przydać się ta analiza?

- Tym którzy przetważają duże ilości informacji.
- Tym którzy przetważają cenne inforamcje.
- Tym którzy chcą chronić informacje przez długi czas.
- Tym którzy projektują system.

# Równoważne długości kluczy

| Algorytm symetryczny | Funkcja skrótu | RSA/ DSA/DH | ECC |
|---|---|---|---|
| 80 | 160 | 1024 | 160 |
| 112 | 224 | 2048 | 224 |
| 128 | 256 | 3072 | 256 |
| 192 | 384 | 7680 | 384 |
| 256 | 512 | 15360 | 512 |

# Złożoność obliczniowa ataków (1)

Przez złożoność ataku na algorytm kryptograficzny rozumiemy liczbę operacji elementarnych niezbędnych do znaleziania klucza.

- Algorytm symetryczny

$$\Phi_{SYM}(B) = e^{\ln(2)B}.$$

- Funkcja skrótu

$$\Phi_{HASH}(B) = e^{\frac{\ln(2)B}{2}}.$$

# Złożoność obliczniowa ataków (2)

- Klasyczne algorytmy asymetryczne (RSA, DSA, DH)

$$\Phi_{CLAS}(B) = e^{\beta(\ln 2^B)^{1/3}(\ln\ln 2^B)^{2/3}}.$$

- Algorytmy asymetryczne bazujące na krzywych eliptycznych (ECDSA, ECDH)

$$\Phi_{ECC}(B) = e^{\frac{\ln(2)B}{2}}.$$

# Założenia modelu

- Szybkość z jaką spada jednostka mocy obliczeniowej jest znana i dana funkcją $e^{-\alpha t}$ (dla prawa Moore'a $\alpha = \frac{2}{3}\ln(2)$).
- Funkcja premii $P(t)$ za odszyfrowanie wiadomości po czasie $t$ jest nierosnąca i znana jest jej wartość $P_0$ dla $t = 0$.
- Znany jest czas $T$ eksploatacji systemu.
- Znany jest koszt $Q$ budowy maszyny łamiącej klucze.

# Funkcja kosztu

Poniższa formuła przedstawia funkcję kosztu łamania systemu kryptograficznego.

$$C(t_1, t_2) = \frac{Qe^{-\alpha t_1}}{t_2 - t_1} - P(t_2 - t_1)\frac{T - t_1}{t_2 - t_1}.$$

Jeżeli funkcja dla pewnych wartości $t_1, t_2 \in [0, T)$ przyjmuje wartości ujemne, to łamanie systemu jest opłacalne.

# Funkcja pomocnicza

Funkcja pomocnicza

$$q(t_1) = \frac{Qe^{-\alpha t_1}}{T - t_1}$$

przyjmuje w przedziale $[0, T)$ najmniejszą wartość w punkcie $t_1 = T - \frac{1}{\alpha}$.

# Nieopłacalność ataku na kryptosystem (1)

Z faktu, że funkcja premii $P$ jest nierosnąca i $P(0) = P_0$ wynika związek

$$\begin{aligned} C(t_1, t_2) &= \frac{Qe^{-\alpha t_1}}{t_2 - t_1} - P(t_2 - t_1)\frac{T - t_1}{t_2 - t_1} \\ &\geq \frac{Qe^{-\alpha t_1}}{t_2 - t_1} - P_0\frac{T - t_1}{t_2 - t_1}. \end{aligned}$$

Jeśli zatem $\frac{Qe^{-\alpha t_1}}{t_2 - t_1} - P_0\frac{T - t_1}{t_2 - t_1} > 0$, to atak na kryptosystem jest nieopłacalny.

# Nieopłacalność ataku na kryptosystem (2)

Z poprzedniego związku otrzymujemy

$$P_0 < \frac{Qe^{-\alpha t_1}}{T - t_1} = q(t_1) \leq \alpha Q e^{-\alpha T + 1}.$$

Ostatecznie warunek na nieopłacalność łamania systemu kryptograficznego przyjmuje postać

$$Q > \frac{P_0 e^{\alpha T - 1}}{\alpha}.$$

# Niezależność od strategii atakującego

Spełnienie warunku $Q > \frac{P_0 e^{\alpha T - 1}}{\alpha}$ powoduje, że atak jest niopłacalny niezależnie od tego jaką strategię stosuje atakujący.

$$C_T = \int_{0 \leq t_1, t_2 < T} \delta(t_1, t_2) C(t_1, t_2) \geq 0.$$

# Nieopłacalność ataku - warunek ogólny (1)

Niech $\Phi(B)$ będzie złożonością obliczeniową ataku na system kryptograficzny o kluczach długości $B$ bitów. Załóżmy ponadto, że $Q_0$ oznacza koszt maszyny, która łamie taki system z kluczami długości $B_0$ bitów w chwili $T_0$. Wtedy

$$Q(t, B) = Q_0 e^{-\alpha(t-T_0)} \frac{\Phi(B)}{\Phi(B_0)}$$

określa koszt maszyny łamiącej system z kluczami długości $B$ bitów w chwili $t$.

# Nieopłacalność ataku - warunek ogólny (2)

Jeżeli system ktyptograficzny spełnia warunek

$$Q(T_1, B) > \frac{P_0 e^{\alpha(T_2 - T_1) - 1}}{\alpha}$$

$$Q_0 e^{-\alpha(T_1 - T_0)} \frac{\Phi(B)}{\Phi(B_0)} > \frac{P_0 e^{\alpha(T_2 - T_1) - 1}}{\alpha}$$

$$\Phi(B) > \Phi(B_0) \frac{P_0 e^{\alpha(T_2 - T_0) - 1}}{\alpha Q_0}$$

to atak na niego jest nieopłacalny do chwili $T_2$.

# Nieopłacalność ataku - szyfry symetryczne (1)

Dla szyfrów symetrycznych przyjmujemy, że maszyną łamiącą klucze jest COPACOBANA, dla której mamy

- $Q_0 = 740$ PLN,
- $B_0 = 56$,
- $T_0 = 2006$,
- $\alpha = \frac{2}{3}\ln(2)$,
- $\Phi(B) = e^{\ln(2)B}$.

# Nieopłacalność ataku - szyfry symetryczne (2)

Jeśli liczba bitów klucza szyfru symetrycznego spełnia nierówność

$$B_{SYM} > B_0 + \frac{1}{\ln(2)}\left(\alpha(T_2 - T_0) - 1 + \ln\frac{P_0}{\alpha Q_0}\right)$$

to atak na taki szyfr jest nieopłacalny.

# Nieopłacalność ataku - funkcje skrótu (1)

Dla funkcji skrótu przyjmujemy, że maszyną łamiącą klucze jest COPACOBANA, dla której mamy

- $Q_0 = 740$ PLN,
- $B_0 = 112$,
- $T_0 = 2006$,
- $\alpha = \frac{2}{3}\ln(2)$,
- $\Phi(B) = e^{\frac{\ln(2)B}{2}}$.

# Nieopłacalność ataku - funkcje skrótu (2)

Jeśli długość skrótu spełnia nierówność

$$B_{HASH} > B_0 + \frac{2}{\ln(2)} \left( \alpha(T_2 - T_0) - 1 + \ln \frac{P_0}{\alpha Q_0} \right)$$

to atak na taką funkcję jest nieopłacalny.

# Nieopłacalność ataku - RSA i DSA (1)

Dla kryptosystemów RSA i DSA przyjmujemy, że maszyną łamiącą klucze jest TWIRL, dla której mamy

- $Q_0 = 3.000.000$ PLN,
- $B_0 = 1024$,
- $T_0 = 2004$,
- $\alpha = \frac{2}{3}\ln(2)$,
- $\Phi(B) = e^{1.9(\ln 2^B)^{1/3}(\ln\ln 2^B)^{2/3}}$.

# Nieopłacalność ataku - RSA i DSA (2)

Jeśli długość kluczy spełnia nierówność

$$\Phi(B) > \Phi(B_0)\frac{P_0 e^{\alpha(T_2-T_0)-1}}{\alpha Q_0}$$

to atak jest nieopłacalny.

# Nieopłacalność ataku - krzywe eliptyczne (1)

Dla kryptosystemów opartych na krzywych eliptycznych przyjmujemy, że maszyną łamiącą klucze jest COPACOBANA, dla której mamy

- $Q_0 = 2.200.000$ PLN,
- $B_0 = 112$,
- $T_0 = 2006$,
- $\alpha = \frac{2}{3}\ln(2)$,
- $\Phi(B) = e^{\frac{\ln(2)B}{2}}$.

# Nieopłacalność ataku - krzywe eliptyczne (2)

Jeśli długość kluczy spełnia nierówność

$$B_{ECC} > B_0 + \frac{2}{\ln(2)}\left(\alpha(T_2 - T_0) - 1 + \ln\frac{P_0}{\alpha Q_0}\right)$$

to atak jest nieopłacalny.

# Podsumowanie

Korzystając ze wzorów należy pamiętać, że

- $T_2$ oznacza czas zakończenia ochrony kryptograficznej, a nie czas po którym dany algorytm nie będzie już stosowany,
- $P_0$ powinno raczej wyrażać maksymalną, a nie średnią wartość informacji przetwarzanej przez system.