STUDIA INFORMATICA 2015
Volume 36 Number 3 (121)

Dariusz RZOŃCA
Politechnika Rzeszowska, Katedra Informatyki i Automatyki

KRYPTOANALIZA ALGORYTMU SZYFRUJĄCEGO HPM14

Streszczenie. W artykule przeanalizowano jeden z opisywanych w literaturze algorytmów szyfrujących, bazujący na niestandardowym zastosowaniu kryptograficznie silnych funkcji skrótu. Wskazano jego potencjalne słabości i możliwe do przeprowadzenia ataki. Jeden z nich wymaga częściowej znajomości tekstu jawnego, drugi jedynie nierównomiernego rozkładu bitów (nieznanego intruzowi) na poszczególnych pozycjach bloku wiadomości.

Słowa kluczowe: kryptoanaliza, analiza bezpieczeństwa, algorytm szyfrujący

CRYPTOANALYSIS OF HPM14 ENCRYPTION ALGORITHM

Summary. The paper analyzes the encryption algorithm, described in the literature, based on a custom application of cryptographically strong hash functions. Potential weaknesses and possible attacks have been described. One of them requires partial knowledge of the plaintext, while the second one needs only the uneven distribution of bits (unknown to intruder) at different positions of message blocks.

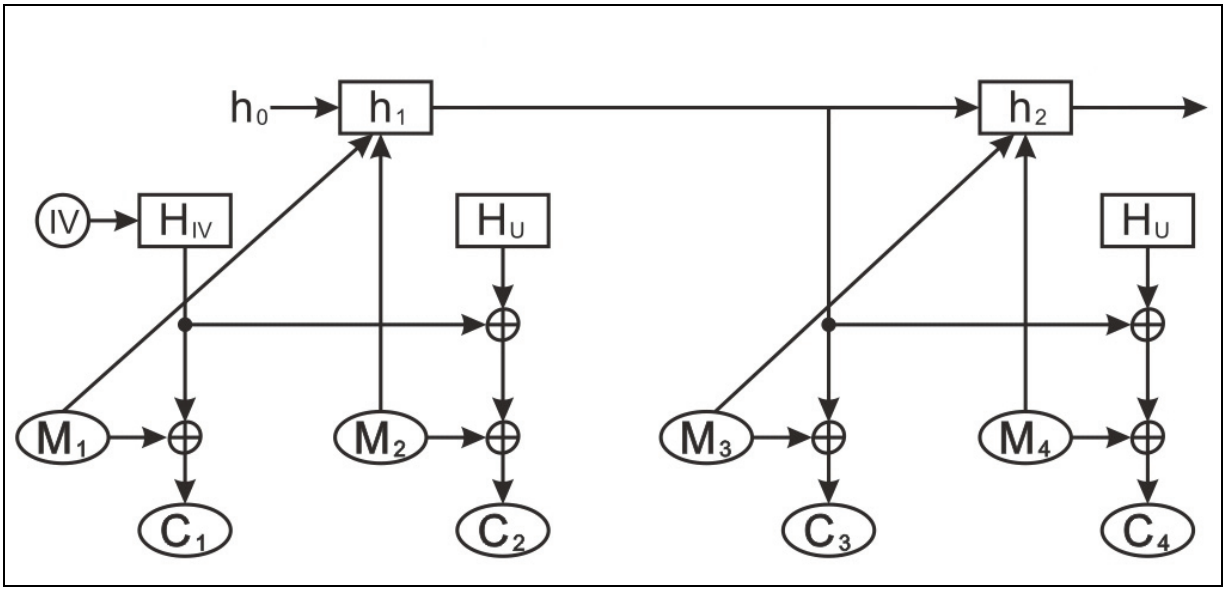
Keywords: cryptanalysis, security analysis, encryption algorithm

1. Wstęp

Zapewnienie poufności transmisji danych jest istotnym zagadnieniem. W tym celu stosowane są różne algorytmy szyfrujące, jak np. AES (*Advanced Encryption Standard*) [1]. Niekiedy zamiast typowych rozwiązań opisywane są niestandardowe algorytmy, dedykowane dla pewnych specyficznych przypadków. W niniejszym artykule przedstawiono analizę bezpieczeństwa jednego z takich algorytmów [2], wraz ze wskazaniem kilku potencjalnych luk i możliwych ataków.

2. Opis algorytmu

W pracy [2] przedstawiono koncepcję zabezpieczenia transmisji pomiędzy urządzeniami mobilnymi w sieciach PAN (*Personal Area Network*) [3]. Opisywana tam metoda przeznaczona jest do zabezpieczania połączeń punkt-punkt pomiędzy urządzeniami mobilnymi, przy użyciu dedykowanej aplikacji do transmisji plików. Przykład takiego zastosowania opisano w [2]. W trakcie nawiązywania połączenia pomiędzy dwoma smartfonami uzgadniany jest tajny klucz (H_u), wspólny dla obu stron, z wykorzystaniem kodów QR [4]. Transmitowane dane szyfrowane są dedykowanym algorytmem, oznaczonym tu jako HPM14 (akronim utworzony z początkowych liter nazwisk autorów i roku publikacji [2]). Algorytm ten bazuje na pracach [5, 6], szerzej opisano go także w dalszej części niniejszego artykułu. Jest to symetryczny, blokowy szyfr, bazujący na kryptograficznie silnych funkcjach skrótu SHA-256 lub SHA-512 [7]. Do jego cech należy połączenie procesu szyfrowania z obliczaniem skrótów kolejnych bloków transmitowanych danych, co pozwala wykryć ewentualne przekłamania, jednak nie gwarantuje integralności danych w sensie wykrycia celowych modyfikacji wprowadzonych w procesie transmisji przez osoby trzecie. Autorzy podają, że algorytm ten jest szybszy niż AES, zapewniając niegorszy poziom bezpieczeństwa [2]. Proces szyfrowania przedstawiony jest na rys. 1.



Rys. 1. Schemat procesu szyfrowania [2]
Fig. 1. Scheme of the encryption process [2]

Na rysunku użyto następujących oznaczeń:

- M_1, M_2, ... M_i – bloki tekstu jawnego,
- C_1, C_2, ... C_i – bloki kryptogramu,
- IV – wektor inicjalizacyjny,
- h_1, h_2, ... h_k – kolejne iteracje jednokierunkowej funkcji skrótu,
- H_{IV} – skrót wektora inicjalizacyjnego,

- H_u – klucz szyfrujący,
- $\oplus$ – operacja XOR.

W zależności od użytej funkcji skrótu zmienia się rozmiar bloków i kluczy. Dla SHA-256 bloki M_i, C_i, skrót H_{IV} i klucz H_u mają długość 256 bitów, dla SHA-512 ich rozmiar to 512 bitów. Proces szyfrowania przebiega następująco:

$$C_1 = M_1 \oplus H_{IV}$$
$$C_2 = M_2 \oplus H_{IV} \oplus H_u$$
$$C_i = M_i \oplus h_{\lfloor (i-1)/2 \rfloor} \text{ (dla } i \geq 3 \text{ i nieparzystego)}$$
$$C_i = M_i \oplus h_{\lfloor (i-1)/2 \rfloor} \oplus H_u \text{ (dla } i \geq 4 \text{ i parzystego).}$$

Analogicznie można zapisać proces deszyfrowania:

$$M_1 = C_1 \oplus H_{IV}$$
$$M_2 = C_2 \oplus H_{IV} \oplus H_u$$
$$M_i = C_i \oplus h_{\lfloor (i-1)/2 \rfloor} \text{ (dla } i \geq 3 \text{ i nieparzystego)}$$
$$M_i = C_i \oplus h_{\lfloor (i-1)/2 \rfloor} \oplus H_u \text{ (dla } i \geq 4 \text{ i parzystego).}$$

3. Analiza bezpieczeństwa

3.1. Atak z częściowo znanym tekstem jawnym

Łatwo zauważyć, że algorytm ten nie jest odporny na ataki ze znanym tekstem jawnym (ang. *known plaintext attack*) [8]. Znajomość pierwszych dwóch bloków tekstu jawnego M_1 i M_2 wystarcza do obliczenia na podstawie bloków kryptogramu C_1 i C_2 skrótu wektora inicjalizacyjnego H_{IV} i klucza szyfrującego H_u. Odtworzenie hasła przebiega wówczas następująco:

$$H_{IV} = C_1 \oplus M_1$$
$$H_u = C_2 \oplus M_2 \oplus H_{IV}$$

Znajomość H_u wystarcza do odszyfrowania kolejnych bloków wiadomości.

Powyższy atak wymaga znajomości dwóch pierwszych bloków tekstu jawnego M_1 i M_2, a więc 64 bajtów dla funkcji SHA-256 bądź 128 dla SHA-512. Warto zauważyć, że intruz niekiedy może dysponować taką wiedzą. Przykładowo, znajomość typu przesyłanego pliku pozwala często na określenie ze znacznym prawdopodobieństwem fragmentów nagłówka przesyłanego pliku, który może się składać z kilkunastu, a nawet kilkuset bajtów. W zależności od rodzaju pliku wiedza ta pozwala na częściowe bądź pełne odtworzenie klucza H_u.

3.2. Atak wykorzystujący kryptoanalizę statystyczną

Zauważmy, że w procesie szyfrowania dwóch kolejnych bloków jest wykorzystywany ten sam skrót $h_{\lfloor (i-1)/2 \rfloor}$. Licząc XOR kolejnych bloków kryptogramu, można go wyeliminować. Otrzymujemy więc:

$$C_1 \oplus C_2 = M_1 \oplus H_{IV} \oplus M_2 \oplus H_{IV} \oplus H_u = M_1 \oplus M_2 \oplus H_u$$

$$C_3 \oplus C_4 = M_3 \oplus h_1 \oplus M_4 \oplus h_1 \oplus H_u = M_3 \oplus M_4 \oplus H_u$$

itd.

Jeżeli rozkład częstości występowania poszczególnych znaków w wiadomości M nie jest równomierny, to rozkład w $C_i \oplus C_{i+1}$ także nie będzie równomierny i pozwoli na odtworzenie H_u. Co więcej, można wykazać, że znajomość rozkładu w M nie jest konieczna do przeprowadzenia ataku.

Twierdzenie 3.2.1

Dla dowolnego rozkładu α częstości występowania bitu o wartości 0 na poszczególnych pozycjach *n*-bitowego słowa $\alpha = (\alpha_{0,0}; \alpha_{0,1}; ...; \alpha_{0,n})$, gdzie $\alpha_{0,i} \in [0;1]$ jest częstością występowania bitu 0 na *i*-tej pozycji, generowany przez funkcję $\oplus$ (XOR) rozkład $\beta = (\beta_{0,0}; \beta_{0,1}; ...; \beta_{0,n})$ faworyzuje 0, tj. $\beta_{0,i} \in \left[\frac{1}{2};1\right]$, przy czym $\beta_{0,i} = \frac{1}{2}$ wtedy i tylko wtedy, gdy $\alpha_{0,i} = \frac{1}{2}$, czyli częstość występowania zer i jedynek na *i*-tej pozycji była jednakowa.

Dowód

Wartość $\alpha_{0,i}$ jest częstością występowania bitu 0 na *i*-tej pozycji, $\alpha_{0,i} \in [0;1]$, a więc częstość wystąpienia na tej pozycji bitu 1 wynosi $\alpha_{1,i} = 1 - \alpha_{0,i}$. Ponieważ $0 \oplus 0 = 1 \oplus 1 = 0$, a $0 \oplus 1 = 1 \oplus 0 = 1$, więc zakładając niezależność wyborów argumentów operacji $\oplus$, z częstością poszczególnych bitów podaną wyżej, częstość wystąpienia 0 na *i*-tej pozycji w wyniku operacji $\oplus$ wynosi

$$\beta_{0,i} = \alpha_{0,i}^2 + \alpha_{1,i}^2 = \alpha_{0,i}^2 + (1 - \alpha_{0,i})^2 = 2\alpha_{0,i}^2 - 2\alpha_{0,i} + 1,$$

zaś częstość wystąpienia w wyniku bitu 1 na *i*-tej pozycji wynosi

$$\beta_{1,i} = 2\alpha_{0,i}\alpha_{1,i} = 2\alpha_{0,i}(1 - \alpha_{0,i}) = -2\alpha_{0,i}^2 + 2\alpha_{0,i}.$$

Obliczmy różnicę częstości występowania poszczególnych bitów na *i*-tej pozycji.

$$\beta_{0,i} - \beta_{1,i} = 4\alpha_{0,i}^2 - 4\alpha_{0,i} + 1$$

Wyróżnik tego trójmianu kwadratowego wynosi $\Delta = 0$, a więc różnica $\beta_{0,i} - \beta_{1,i}$ jest zawsze nieujemna, przy czym równa zero jest jedynie dla $\alpha_{0,i} = \alpha_{1,i} = \frac{1}{2}$, co kończy dowód.

Skorzystanie z powyższego twierdzenia umożliwia przeprowadzenie następującego ataku. Liczymy XOR kolejnych bloków kryptogramu $C_1 \oplus C_2$, $C_3 \oplus C_4$ itd. Jak już wspomniano, otrzymujemy w ten sposób XOR kolejnych bloków tekstu jawnego z kluczem H_u, tzn. $C_1 \oplus C_2 = M_1 \oplus M_2 \oplus H_u$, $C_3 \oplus C_4 = M_3 \oplus M_4 \oplus H_u$ itd. Zgodnie z twierdzeniem 3.2.1 rozkład częstości występowania bitu 0 na *i*-tych pozycjach słów generowanych przez funkcję XOR z bloków tekstu jawnego $M_1 \oplus M_2$, $M_3 \oplus M_4$ itd. faworyzuje 0. Zliczamy więc ilość wystąpień bitu 0 na poszczególnych pozycjach słów wygenerowanych przez funkcję XOR z kolejnych bloków kryptogramu $C_1 \oplus C_2$, $C_3 \oplus C_4$ itd. Szukamy następnie znaczących odchyleń od częstości $\frac{1}{2}$ na *i*-tych pozycjach tak utworzonych słów. Jeżeli na *i*-tej pozycji częściej występuje 0, to w kluczu H_u na tej pozycji z dużym prawdopodobieństwem też jest 0, jeżeli częściej występuje tam 1, to *i*-ty bit klucza H_u z dużym prawdopodobieństwem jest jedynką. Atak taki pozwala na łatwe odtworzenie znacznych fragmentów klucza H_u, dla wiadomości zawierających nierównomierny rozkład bitów na poszczególnych pozycjach. Taka sytuacja często występuje w praktyce, np. dla plików tekstowych.

4. Podsumowanie

W artykule przedstawiono potencjalne słabości algorytmu szyfrującego HPM14, pozwalające na odtworzenie klucza i odszyfrowanie zakodowanej wiadomości. Pierwszy z opisanych ataków bazuje na znajomości początkowych fragmentów tekstu jawnego (np. nagłówków przesyłanego pliku). Drugi z ataków wymaga nierównomiernego rozkładu bitów na poszczególnych pozycjach wiadomości, nie jest jednak konieczna znajomość *a priori* tego rozkładu przez intruza.

BIBLIOGRAFIA

1. Advanced Encryption Standard (AES), Federal Information Processing Standards Publications 197, National Institute of Standards and Technology, Nov. 2001, http://csrc.nist.gov/publications/fips/fips197/fips-197.pdf.
2. Hłobaż A., Podlaski K., Milczarski P.: Applications of QR Codes in Secure Mobile Data Exchange, [w:] Kwiecień A., Gaj P., and Stera P. (Eds.): Computer Networks 2014, Communications in Computer and Information Science 431, Springer International Publishing 2014, s. 227÷286.
3. Tanenbaum A., Wetherall D.: Sieci komputerowe. Helion, Gliwice 2012.
4. BS ISO/IEC 18004:2006. Information technology. Automatic identification and data capture techniques. QR Code 2005 bar code symbology specification.

5. Hłobaż A.: Bezpieczeństwo transmisji danych pomiarowych – metoda szyfrowania wiadomości wraz z współbieżnym liczeniem skrótu. Przegląd Telekomunikacyjny – Wiadomości Telekomunikacyjne 1/2007, s. 13÷15.
6. Hłobaż A.: Bezpieczeństwo transmisji danych pomiarowych – modyfikacje metody szyfrowania wiadomości wraz z jej współbieżnym uwierzytelnianiem. Przegląd Włókienniczy – Włókno, Odzież, Skóra 1/2008, s. 39÷42.
7. Secure Hash Standard (SHS), Federal Information Processing Standards Publications 180-3, National Institute of Standards and Technology, Oct. 2008, http://csrc.nist.gov/publications/fips/fips180-3/fips180-3_final.pdf.
8. Menezes A., van Oorschot P., Vanstone S.: Kryptografia stosowana. Wydawnictwa Naukowo-Techniczne, Warszawa 2005.

Abstract

Security of the transmission requires an encryption algorithm to ensure privacy. Usually one of the standard algorithms is applied, however non-standard solutions are also described in the literature. The paper analyzes one of such custom encryption scheme [2], based on cryptographically strong hash functions, as shown in Fig. 1.

Unfortunately, it has been shown that such algorithm is vulnerable to some attacks. One of them is partially known plaintext attack. If an intruder knows, or is able to guess, the very beginning of the plaintext message, he could easily decrypt the following transmission. Such knowledge is highly probable, e.g. headers of transmitted files are very common.

Another attack described here requires only the uneven distribution of bits at different positions of message blocks. Such situation is typical for different types of uncompressed files, e.g. text files. Moreover, it has been proved that even the knowledge of such distribution is unnecessary for the intruder to perform the attack.

Adres

Dariusz RZOŃCA: Politechnika Rzeszowska im. Ignacego Łukasiewicza,
al. Powstańców Warszawy 12, 35-959 Rzeszów, Polska, drzonca@kia.prz.edu.pl.