

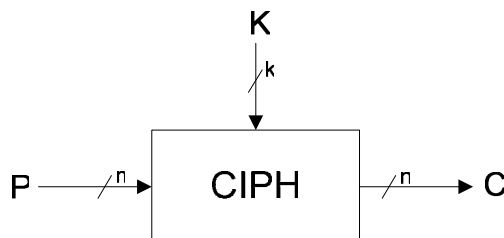
prof. dr inż. Wojciech Oszywa
mgr inż. Rafał Gliwa
Wojskowy Instytut Łączności
05-130 Zegrze

Tryby pracy szyfrów blokowych do realizacji uwierzytelnionego szyfrowania

Tryb pracy szyfru blokowego to algorytm określający, w jaki sposób zastosować n -bitowy szyfr blokowy, aby osiągnąć założony cel. Celem, któremu poświęcony został ten artykuł, jest zapewnienie jednocześnie poufności i uwierzytelnienia wiadomości, które to zagadnienie określane jest w literaturze wspólnym terminem uwierzytelnionego szyfrowania. W niniejszym artykule zaprezentowano szereg możliwości realizacji uwierzytelnionego szyfrowania wyłącznie w oparciu o szyfr blokowy, bez wykorzystania kryptograficznej funkcji skrótu.

1. Tryby pracy szyfrów blokowych

Szyfry blokowe stanowią podstawowy element wykorzystywany w kryptografii. Szyfr blokowy, oznaczany CIPH (rysunek 1), odwzorowuje n -bitowy blok tekstu jawnego P na n -bitowy blok szyfrogramu C przy użyciu k -bitowego klucza K . Parametr n nazywamy długością bloku.



Rysunek 1. Szyfr blokowy

W przypadku wiadomości, których długość jest inna niż długość pojedynczego bloku, konieczne staje się użycie szyfru blokowego w jednym z trybów pracy. W ten sposób, poza oczywistą dla szyfru blokowego funkcją zapewnienia poufności, jako element fundamentalny (*ang. primitive*), może on służyć również do generacji ciągów pseudolosowych, zapewnienia integralności danych, do realizacji wyłącznie uwierzytelnienia wiadomości lub do realizacji uwierzytelnienia i poufności jednocześnie. Zagadnienie jednoczesnego zapewnienia uwierzytelnienia i poufności wiadomości określone zostało w literaturze terminem uwierzytelnionego szyfrowania (*ang. Authenticated Encryption*) i zostało sformalizowane dopiero w roku 2000 [1]. Pierwszy dokument normatywny, definiujący tryby pracy szyfrów blokowych został opracowany przez Narodowy Instytut Standaryzacji i Technologii USA (*ang. National Institute of Standard and Technology, NIST*) w 1980 roku dla ówczesnego standardu szyfrowania DES. Dokument ten, pod nazwą FIPS 81 [7], opracowany został z uwzględnieniem specyfikacji algorytmu DES i definiował cztery tryby pracy przeznaczone do szyfrowania wiadomości:

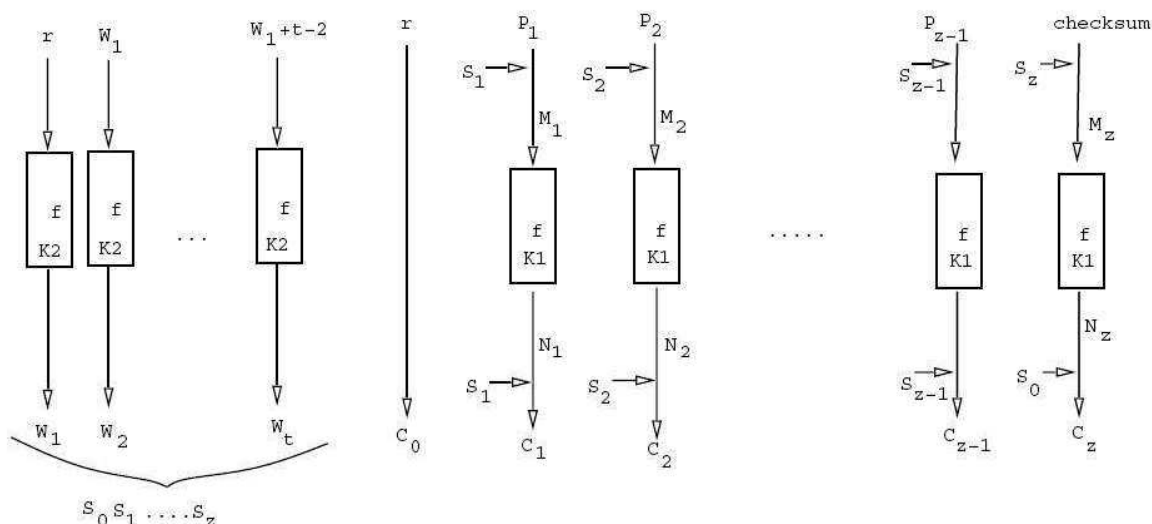
- ECB (*ang. Electronic CodeBook*) – tryb elektronicznej książki kodowej,
- CBC (*ang. Cipher Block Chaining*) – tryb wiązania bloków zaszyfrowanych,
- CFB (*ang. Cipher FeedBack mode*) – tryb sprzężenia zwrotnego szyfrogramu,
- OFB (*ang. Output FeedBack mode*) – tryb sprzężenia zwrotnego wyjścia.

Po przyjęciu przez NIST, w październiku 2000 roku, nowego standardu szyfrowania AES [8], konieczna okazała się weryfikacja dotychczasowego standardu FIPS 81. Stało się to okazją do opracowania i wdrożenia nowych trybów pracy, uwzględniających postępy w dziedzinie kryptologii i odpowiadających na postępujące potrzeby użytkowników. Do dnia dzisiejszego, w wyniku prac standaryzacyjnych NIST, opracowane zostały następujące dokumenty normatywne dotyczące trybów pracy szyfrów blokowych:

- SP 800-38A [9] (grudzień 2001), opisujący pięć trybów pracy zapewniających poufność; są to cztery zaktualizowane wersje trybów pracy z FIPS 81 oraz tryb licznikowy CTR (*ang. Counter mode*);
- SP 800-38B [10] (maj 2005), opisujący tryb pracy CMAC (*ang. Cipher-based MAC*), zapewniający uwierzytelnienie wiadomości;
- SP 800-38C [11] (maj 2004), opisujący tryb pracy CCM (*ang. Counter with CBC-MAC*), zapewniający jednocześnie poufność i uwierzytelnienie wiadomości;
- SP 800-38D [12] (październik 2007), opisujący tryb pracy GCM (*ang. Galois/Counter mode*), zapewniający jednocześnie poufność i uwierzytelnienie wiadomości (w sposób bardziej efektywny od CCM).

2. Jednoprzebiegowe tryby uwierzytelnionego szyfrowania

W 2000 roku, Charanjit Jutla z firmy IBM opracował dwa pierwsze schematy uwierzytelnionego szyfrowania, które już po jednokrotnym przetworzeniu wiadomości, zapewniają jej poufność i uwierzytelnienie. Tryby te to: IACBC (*ang. Integrity Aware Cipher Block Chaining*) oraz IAPM (*ang. Integrity Aware Parallelizable Mode*) (rysunek 2) [5]. Jutla przełamał konstrukcję kompozycji ogólnych, które były prostą kombinacją komponentu szyfrującego i komponentu uwierzytelniającego, i pokazał, że nie ma konieczności wykonywania dodatkowego przebiegu tj. powtórnego przetwarzania każdego bloku wiadomości w celu jej uwierzytelnienia. Czynność tą można zastąpić przez wykonanie prostej operacji sumowania XOR bloków tekstu jawnego, a następnie zaszyfrowanie otrzymanego wyniku.



Rysunek 2. Tryb IAPM (źródło [5])

Naliczana suma kontrolna ma wówczas postać:

$$checksum = \sum_{i=1}^{z-1} P_i \quad (\text{suma XOR}) \quad (1)$$

Aby uniknąć możliwości ujawnienia w ten sposób informacji o tekście jawnym koniecznym jest jednak „wybielenie” (*ang. whitening*) sumy kontrolnej. Przez „wybielenie” rozumiemy wykonanie operacji XOR określonego bloku z sekwencją pseudolosową. „Wybieleniu” podlegają też wszystkie

wejścia i wyjścia wywołań szyfru blokowego. W trybie IACBC, zbudowanym w oparciu o tryb szyfrowania CBC, wybielanie wejść do szyfru blokowego odbywa się dzięki łańcuchowaniu bloków, wynikającemu z konstrukcji typu CBC. Do wybielania wyjść szyfru blokowego w trybie IACBC oraz do wybielania wejść i wyjść szyfru blokowego w trybie IAPM służy specjalnie wygenerowana sekwencja pseudolosowa o określonych właściwościach. Jak wiadomo, szyfrowanie CBC z natury następuje blok po bloku: nie można rozpocząć szyfrowania $k+1$ bloku, dopóki nie dysponujemy wynikiem szyfrowania bloku k . Stąd też większe zainteresowanie przyciągnął tryb IAPM, który pozbawiony jest takiej niedogodności. Istotne znaczenie w konstrukcji trybów IACBC i IAPM odgrywa sposób generacji i właściwości wspomnianej sekwencji pseudolosowej. Generacja bazuje na losowym wektorze inicjalizującym r . Jednym ze sposobów generacji jest zaszyfrowanie kolejnych bloków $r, r+1, r+2, \dots, r+m$ (m - ilość bloków tekstu jawnego) przy użyciu szyfru blokowego z wykorzystaniem klucza innego niż używany do szyfrowania bloków tekstu jawnego. Wymaga to jednak m dodatkowych szyfrowań, przez co jest tak samo mało efektywne jak generowanie ciągu uwierzytelnienia MAC w trybie CBC-MAC. Zwiększenie efektywności omawianych trybów jednorazowych wynika z faktu, że wymagana sekwencja nie musi być w pełni pseudolosowa, lecz warunkiem wystarczającym jest, aby była parami niezależna (*ang. pairwise independent*). Oznacza to, że jeżeli mamy sekwencję postaci s_1, s_2, \dots, s_m to każdy element s_i ma być ciągiem losowym, ale tylko parami niezależnym z pozostałymi elementami. Ciąg losowy n -bitowych liczb o rozkładzie jednostajnym s_1, s_2, \dots, s_m nazywamy parami niezależnym, jeśli dla każdej pary $i, j, i \neq j$ i każdej pary n -bitowych stałych c_1 i c_2 :

$$\Pr[s_i = c_1 \cap s_j = c_2] = \frac{1}{2^{2n}} \quad (2)$$

Taka sekwencja jest efektywnie generowana za pomocą konstrukcji podzbiorów w trakcie ($\log m$) (m - ilość bloków tekstu jawnego) operacji kryptograficznych, takich jak szyfrowanie blokowe. Dzięki temu, w porównaniu do kompozycji ogólnej, utworzonej na bazie trybu CBC (szyfrowanie CBC i uwierzytelnienie CBC-MAC), która wymagałaby m wywołań szyfru blokowego do zaszyfrowania wiadomości o długości m bloków oraz dodatkowych m wywołań szyfru blokowego do uwierzytelnienia tej wiadomości, tryby IACBC i IAPM wymagają łącznej liczby $(m + \log m)$ wywołań szyfru blokowego do zapewnienia uwierzytelnionego szyfrowania. Z czasem okazało się, że wystarczający jest nawet słabszy warunek, a mianowicie, by elementy sekwencji s były parami równomiernie zróżnicowane (*ang. pairwise differentially-uniform*). Ciąg losowy n -bitowych liczb o rozkładzie jednostajnym s_1, s_2, \dots, s_m nazywamy równomiernie zróżnicowanym parami, jeśli dla każdej pary $i, j, i \neq j$ i każdej n -bitowej stałej c :

$$\Pr[s_i \oplus s_j = c] = \frac{1}{2^n} \quad (3)$$

Koszt wygenerowania sekwencji równomiernie zróżnicowanej parami jest szczególnie niski dla operacji w ciele $GF(p)$. Ciąg losowy n -bitowych liczb s_1, s_2, \dots, s_n o rozkładzie jednostajnym w $GF(p)$ nazywamy równomiernie zróżnicowanym parami w $GF(p)$, jeśli dla każdej pary $i, j, i \neq j$ i każdej n -bitowej stałej c w $GF(p)$:

$$\Pr[(s_i - s_j) \bmod p = c] = \frac{1}{p} \quad (4)$$

W takim przypadku, generacja sekwencji randomizującej sprowadza się do wykonania jednego dodatkowego szyfrowania, połączonego z prostymi i szybkimi operacjami kombinatorycznymi, a tym samym koszt uwierzytelnionego szyfrowania wynosi zaledwie $(m+1)$ szyfrowań (m - ilość bloków tekstu jawnego).

Kolejne jednorazowe tryby zapewniające uwierzytelnione szyfrowanie opracowali Gligor i Donescu. XCBC-XOR oraz XECB-XOR [4] są pokrewne odpowiednio trybom IACBC i IAPM Jutli. Główną ideą trybów XCBC oraz XECB, odróżniającą je od IACBC i IAPM, było zastosowanie do obliczeń sekwencji randomizującej nie arytmetyki $\bmod p$, lecz arytmetyki $\bmod 2^n$, która jest szczególnie efektywnie realizowana na procesorach.

Rogaway, Bellare i Black, wzorując się na konstrukcji IAPM, opracowali jednoprzebiegowy tryb pracy pod nazwą OCB [12] (*ang. Offset CodeBook*). OCB definiuje użycie konkretnej konstrukcji wyliczania parami niezależnej sekwencji randomizującej, która zawiera mnożenie przez stały element w binarnym ciele skończonym $GF(2^n)$ oraz obliczenie kodu Gray'a. Metoda ta umożliwia wcześniejsze obliczenie niektórych wartości sekwencji i zapamiętanie ich w tablicy, co prowadzi do skrócenia czasu obliczeń w trakcie działania OCB. Zaproponowana konstrukcja sprawia również, że OCB staje się trybem z jednym kluczem (IAPM wymaga drugiego klucza do generacji parami niezależnej sekwencji).

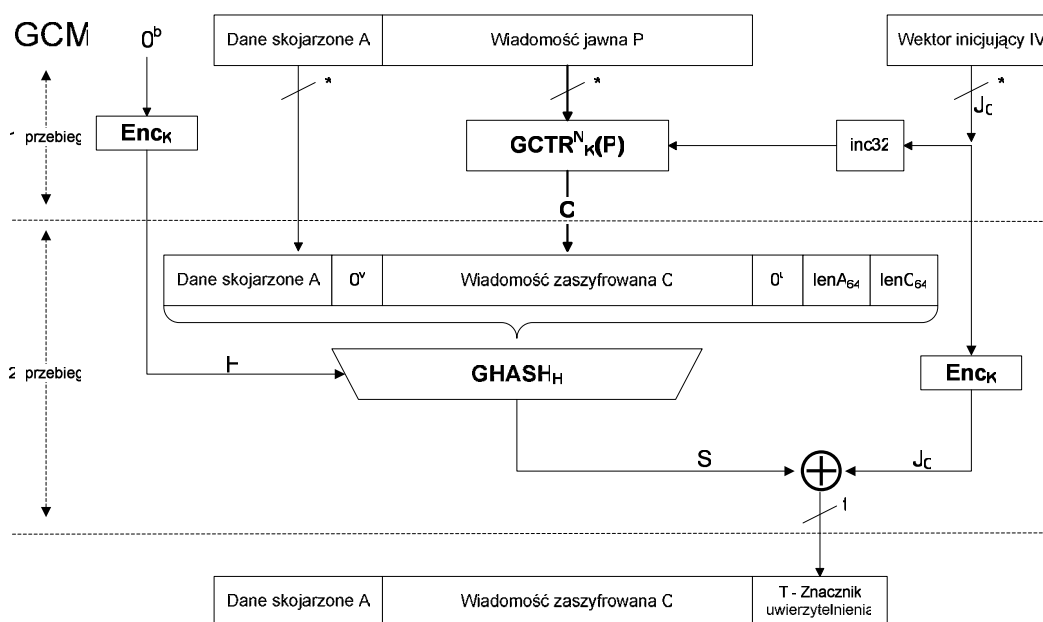
Pomimo niewątpliwiej zalety trybów jednoprzebiegowych, jaką jest szybkość przetwarzania, opracowane algorytmy nie weszły do powszechnego użytku. Stało się tak na skutek opatentowania trybów jednoprzebiegowych przez ich autorów. Obawa przed naruszeniem własności intelektualnej oraz wynikające z patentów ograniczenia zmusiły środowisko kryptologiczne do poszukiwania innych kompromisowych rozwiązań. Podjęte prace koncepcyjne doprowadziły do opracowania i dynamicznego rozwoju dwuprzebiegowych trybów uwierzytelnionego szyfrowania.

3. Dwuprzebiegowe tryby uwierzytelnionego szyfrowania

W celu zapewnienia poufności i uwierzytelnienia wiadomości, dwuprzebiegowe tryby pracy przetwarzają każdy blok wiadomości dwukrotnie. W jednym przebiegu wiadomość jest szyfrowana, w drugim przebiegu opatrywana znacznikiem uwierzytelnienia, przy czym kolejność tych operacji jest uzależniona od intencji projektanta. Szczególnie ważne jest to, że dwuprzebiegowe tryby uwierzytelnionego szyfrowania wymagają na wejściu tylko jednego klucza. Pierwszym dwuprzebiegowym trybem uwierzytelnionego szyfrowania, jaki opracowano, był tryb CCM (*ang. Counter mode with CBC-MAC*) [11]. W celu wyeliminowania pewnych słabości CCM zaprojektowano tryb EAX [2]. CCM i EAX łączą w sobie dwa bardzo dobrze znane mechanizmy kryptograficzne tj. tryb licznikowy CTR, służący do szyfrowania oraz schemat typu CBC-MAC, służący do uwierzytelniania wiadomości. Ze względu na iteracyjne właściwości mechanizmu CBC-MAC, oba tryby cechuje brak możliwości zrównoleglenia obliczeń. Kolejne prace, dotyczące poprawy efektywności przetwarzania poprzez umożliwienie zrównoleglenia obliczeń, znalazły swoje odzwierciedlenie najpierw w trybie CWC [6], a następnie w trybie GCM [12]. Mechanizm uwierzytelnienia typu CBC-MAC zastąpiono konstrukcją MAC Cartera-Wegmana, której idea jest następująca: zamiast zastosowania funkcji fundamentalnej (np. szyfru blokowego) bezpośrednio do wiadomości X , która ma podlegać uwierzytelnieniu, należy najpierw skrócić tę wiadomość do mniejszego rozmiaru, za pomocą (niekryptograficznej) funkcji należącej do Uniwersalnej Rodziny Funkcji Haszujących, a dopiero wówczas zastosować funkcję kryptograficzną do otrzymanego krótkiego ciągu wynikowego [3].

Tryb GCM (rysunek 3) zapewnia poufność danych, wykorzystując do szyfrowania tryb licznikowy CTR oraz realizuje uwierzytelnienie wiadomości i dodatkowych danych skojarzonych za pomocą uniwersalnej funkcji haszującej GHASH, zdefiniowanej na skończonym ciele binarnym Galois $GF(2^{128})$. Przetwarzanie w trybie GCM jest poprzedzone wyliczeniem wartości wstępnego bloku licznika J_0 w oparciu o wektor inicjalizujący IV . Wstępny blok licznika J_0 , po inkrementacji, służy do inicjalizacji procesu szyfrowania tekstu jawnego P , w wyniku czego otrzymujemy szyfrogram C . Następnie rozpoczyna się proces uwierzytelniania. Dodatkowe dane uwierzytelniane A oraz szyfrogram C są (oddzielnie) dopełniane taką minimalną ilością bitów '0' (jeżeli to konieczne), żeby długość ciągów wynikowych była wielokrotnością rozmiaru bloku. Do konkatencji tych ciągów dołączana jest 64-bitowa reprezentacja długości danych skojarzonych A oraz 64-bitowa reprezentacja długości szyfrogramu C . Tak sformatowany ciąg jest podawany na wejście funkcji haszującej GHASH. Wynik działania funkcji haszującej w postaci bloku S jest szyfrowany poprzez XOR-owanie z ciągiem pseudolosowym, stanowiącym wstępny blok licznika J_0 . Z wyniku szyfrowania brane jest t najstarszych bitów, które tworzą znacznik uwierzytelnienia T . Szyfrogram C oraz znacznik uwierzytelnienia T stanowią wynik działania algorytmu uwierzytelnionego szyfrowania w trybie GCM. Kolejność operacji w procesie uwierzytelnionego

szyfrowania umożliwia weryfikację autentyczności danych po stronie odbiorczej bez potrzeby ich uprzedniego deszyfrowania. W tym celu, w sposób analogiczny jak wyżej, obliczany jest znacznik uwierzytelnienia T' dla danych odebranych.



Rysunek 3. Tryb GCM - schemat uwierzytelnionego szyfrowania

Zgodność znacznika obliczonego T' z wartością znacznika odebranego T świadczy o autentyczności odebranych danych (wiadomości i danych skojarzonych). W takim przypadku szyfrogram C podlega zdeszyfrowaniu, w wyniku czego otrzymujemy tekst jawny P . W przypadku niezgodności znaczników uwierzytelnienia, wiadomość zostaje odrzucona, a na wyjściu funkcji uwierzytelnionego deszyfrowania otrzymujemy wynik BŁĄD.

Mechanizm uwierzytelnienia w trybie GCM jest oparty na uniwersalnej funkcji haszującej, zwanej GHASH, która realizuje mnożenie w ciele Galois $GF(2^{128})$, przez stały parametr H , zwany podkluczem. Podklucz funkcji GHASH jest generowany poprzez zastosowanie szyfru blokowego z kluczem K , do bloku samych zer $H=CIPH_K(0^{128})$. Konkretna instancja funkcji, oznaczana $GHASH_H$, jest używana do wyliczenia wartości funkcji haszującej z danych skojarzonych oraz z szyfrogramu, sformatowanych do bloku postaci $X=X_1||X_2||...||X_{m-1}||X_m$, takiego że:

$$X = (A || 0^v || C || 0^u || [len(A)]_{64} || [len(C)]_{64}).$$

W efekcie swojego działania funkcja GHASH oblicza sumę iloczynów:

$$S = X_1 \bullet H^m \oplus X_2 \bullet H^{m-1} \oplus \dots \oplus X_{m-1} \bullet H^2 \oplus X_m \bullet H.$$

Przez H^i (dla dodatniej liczby całkowitej i) rozumiemy i -tą potęgę bloku H , na przykład, $H^2 = H \bullet H$, $H^3 = H \bullet H \bullet H$, itd. Operacja \bullet oznacza mnożenie modułowe w binarnym ciele Galois o 2^{128} elementach. Modułem mnożenia jest wielomian nieredukowalny $f = 1 + \alpha + \alpha^2 + \alpha^7 + \alpha^{128}$. Zgodnie z konstrukcją Cartera-Wegmana wynik S funkcji haszującej, po zaszyfrowaniu tworzy znacznik uwierzytelnienia T .

4. Wnioski

Zaletą trybów uwierzytelnionego szyfrowania jest to, że jasno definiują w jaki sposób osiągnąć jednocześnie poufność i uwierzytelnienie wiadomości, wykluczając tym samym ryzyko przypadkowego powiązania ze sobą komponentu szyfrującego i komponentu uwierzytelniającego w sposób nie gwarantujący bezpieczeństwa. Ich szczególnie pożądaną cechą jest wymaganie do pracy tylko jednego klucza. Upraszcza to procedurę zarządzania kluczami i redukuje koszty związane z pobieraniem danych kluczowych z pamięci oraz z ich przechowywaniem. Biorąc pod

uwagę liczne udane ataki na kryptograficzne funkcje skrótu w ostatnim czasie, należy podkreślić fakt, że tryby uwierzytelnionego szyfrowania zapewniają poufność i uwierzytelnienie wiadomości wyłącznie w oparciu o algorytm szyfru blokowego, bez potrzeby stosowania kryptograficznej funkcji skrótu. Zmniejsza to także w istotnym stopniu koszty implementacji.

Z przeprowadzonej przez autorów analizy wynika, że warto rozważyć opracowanie schematu uwierzytelnionego szyfrowania w postaci dwuprzebiegowego trybu pracy szyfru blokowego, który zapewni uwierzytelnienie wiadomości jawnej wytworzonej u źródła (w przeciwieństwie do uwierzytelnienia szyfrogramu), a jednocześnie umożliwi zrównoleglenie obliczeń. Konsekwencją takiej kolejności operacji jest brak możliwości szybkiej weryfikacji autentyczności wiadomości (konieczność deszyfrowania wiadomości przed sprawdzeniem autentyczności). Niedogodność tą rekompensuje jednak zwiększenie poziomu bezpieczeństwa. Uwierzytelnienie wiadomości, a następnie zaszyfrowanie jej wraz z naliczonym znacznikiem uwierzytelnienia utrudnia bowiem atak na funkcję uwierzytelnienia: przeciwnikowi zostaje udostępniony wyłącznie zaszyfrowany tekst z zaszyfrowaną wartością znacznika uwierzytelnienia. Dane wejściowe funkcji uwierzytelnienia nie są dostępne (w przeciwieństwie do uwierzytelnienia szyfrogramu), a zaszyfrowana wartość znacznika nie przenosi żadnych informacji dla atakującego. Ponadto, dzięki uwierzytelnieniu wiadomości jawnej uzyskujemy gwarancję, że po pozytywnej weryfikacji znacznika uwierzytelnienia mamy do czynienia z oryginalną wiadomością wytworzoną u źródła. Tym samym wyeliminowany zostaje słaby punkt, jaki występuje w przypadku uwierzytelnienia szyfrogramu, że odbiorca, po poprawnym zweryfikowaniu autentyczności wiadomości, użyje do jej odszyfrowania niewłaściwego klucza, na skutek czego, pomimo pomyślnej weryfikacji uwierzytelnienia, otrzyma inny tekst jawny niż ten, który został wysłany. Konstrukcja trybu, umożliwiająca zrównoleglenie obliczeń wpłynie natomiast na wzrost efektywności przetwarzania, umożliwiając zastosowanie tego rozwiązania w sieciach o szybkościach rzędu dziesięciu gigabitów na sekundę.

Literatura

- [1] M. Bellare, C. Namprempre, *Authenticated encryption: Relations among notions and analysis of the generic composition paradigm*, Journal of Cryptology, vol. 21, no. 4, 2008.
- [2] M. Bellare, P. Rogaway, D. Wagner, *The EAX mode of operation*, LNCS 3017, 2004.
- [3] L. Carter, M. N. Wegman, *New hash functions and their use in authentication and set equality*. In J. of Computer and System Sciences, vol. 22, pp. 265–279, 1981.
- [4] V. Gligor, P. Donescu, *Fast encryption and authentication: XCBC encryption and XECB authentication modes*. FSE 2001, LNCS 2355, Springer-Verlag, pp. 92–108, 2002.
- [5] C. S. Jutla, *Encryption modes with almost free message integrity*. In Advances in Cryptology, EURO-CRYPT 2001, LNCS 2045, Springer-Verlag, pp. 529–544, 2001.
- [6] T. Kohno, J. Viega, D. Whiting, *CWC: A high performance conventional authenticated encryption mode*, 2004. <http://eprint.iacr.org/2003/106/>.
- [7] NIST FIPS Publication 81, *DES Modes of Operation*, U.S. DoC/NIST, 1980.
- [8] NIST FIPS Publication 197, *Specification for the Advanced Encryption Standard*, 2001.
- [9] NIST Special Publication 800-38A, *Recommendation for Block Cipher Modes of Operation - Methods and Techniques*, 2001.
- [10] NIST Special Publication 800-38B, *Recommendation for Block Cipher Modes of Operation: The CMAC Mode for Authentication*, 2005.
- [11] NIST Special Publication 800-38C, *Recommendation for Block Cipher Modes of Operation: The CCM Mode for Authentication and Confidentiality*, May 2004.
- [12] NIST Special Publication 800-38D, *Recommendation for Block Cipher Modes of Operation: Galois/Counter Mode (GCM) and GMAC*, 2007.
- [13] P. Rogaway, M. Bellare, J. Black, *OCB: A block-cipher mode of operation for efficient authenticated encryption*. ACM Transactions on Information and System Security, 2003.