CÁC PHƯƠNG PHÁP TẤN CÔNG HỆ MÃ DÒNG XÁC THỰC HẠNG NHẸ ACORN – 128 VÀ ĐỀ XUẤT CẢI TIẾN ĐỂ TĂNG ĐỘ AN TOÀN

Lê Phê Đô
Trường Đại học Công nghệ - ĐHQG HN
Email: dolp@vnu.edu.vn

***Tóm tắt nội dung*—Hệ mã xác thực nhẹ ACORN là các hệ mã xác thực tốt, tuy nhiên nó vẫn có những điểm yếu nhất định. Trong báo cáo này, chúng tôi tổng hợp một số phương pháp tấn công lên hệ ACORN: phương pháp tấn công khối của Itai Dinur và Adi Shamir và các cuộc tấn công phục hồi trạng thái.**

Tấn công khối là một dạng tấn công đại số cho phép đối phương hạ bậc của phương trình đa thức từ nguyên thủy mật mã. Các chuyên gia Md Iftekhar Salam, Harry Bartlett, Ed Dawson, Josef Pieprzyk, Leonie Simpson, and Kenneth Koon-Ho Wong đã sử dụng tấn công khối để rút ngắn vòng của ACORN. Tấn công khối trên 477 vòng khởi tạo của ACORN có thể khôi phục 128 bit khóa với độ phức tạp khoảng 2^{35}. Các chuyên gia chỉ ra, hệ phương trình tuyến tính liên quan đến trạng thái ban đầu của phiên bản đầy đủ của ACORN có thể dễ dàng tạo ra để thực hiện cuộc tấn công khôi phục trạng thái với độ phức tạp $2^{27.8}$.

Tấn công phục hồi trạng thái dựa vào tính chất trượt được các chuyên gia Meicheng và Dongdai Lin nghiên cứu. Các chuyên gia chỉ ra rằng, đối với mỗi cặp (Key, IV) có cặp khác cùng tạo ra một trạng thái tại các thời điểm khác nhau với xác suất 1.

Để khắc phục các nhược điểm của hệ ACORN – 128 hiện nay, chúng tôi đề xuất thay các thanh ghi dịch tuyến tính bởi các thanh ghi dịch phi tuyến, tăng độ dài khóa, độ dài véc tơ khởi tạo và mỗi khóa, mỗi véc tơ khởi tạo chỉ sử dụng một lần.

Các phương pháp này được dẫn chứng và minh họa cụ thể. Hy vọng, những thông tin đưa lại bổ ích cho các chuyên gia lập mã để tạo ra hệ mã xác thực nhẹ tốt hơn để sử dụng hiệu quả trong IoT.

***Keywords*-Mã dòng, Mã dòng nhẹ, Mã dòng nhẹ xác thực, ACORN, Phương pháp tấn công khối, Phương pháp tấn công dựa vào tính chất trượt, ...**

Nhằm xây dựng một hệ mật vừa thực hiện được chức năng mã hóa vừa thực hiện được vai trò xác thực, NIST đã tổ chức cuộc thi để chọn ra một hệ mật như vậy đặt tên là CEASAR trong thời hạn từ 2013 đến 2017. Vào tháng 3 năm 2014 Hệ mật ACORN là một trong 57 ứng viên lọt vào vòng 1. Tháng 6 năm 2015, trong 29 hệ mật lọt vào vòng 2 có ACORN. Đến tháng 8 năm 2016 còn 15 hệ mật được chọn vào vòng ba, trong đó cũng có ACORN. Kết quả vòng ba dự kiến sẽ công bố vào ngày 15/12/2017.

I. GIỚI THIỆU MÃ XÁC THỰC HẠNG NHẸ ACORN - 128 [1], [2], [3]

ACORN là một thuật toán mã xác thực với dữ liệu kết nối (AEAD Algorithm). Nó dùng 128 bit khóa $k = (k_0, k_1, \ldots, k_{127})$, 128 bít vec tơ khởi tạo $IV = (IV_0, IV_1, \ldots, IV_{127})$ và adlen bit dữ liệu kết nối $AD = (ad_0, ad_1, \ldots, ad_{adlen-1})$. Bản rõ gồm pclen bit $P = (p_0, p_1, \ldots, p_{pclen-1})$ được mã hóa thành bản mã $C = (ct_0, ct_1, \ldots, ct_{pclen-1})$ có cùng độ dài. Sauk khi xử lý các bit bản rõ thẻ xác thực T (gồm t bit, $64 \leq t \leq 128$) được tạo ra.

ACORN gồm 6 thanh ghi dịch tuyến tinh được ký hiệu tương ứng là a, b, c, d, e, f và một thanh ghi dịch phi tuyến ký hiệu là g. Tại bước thứ i chúng ta ký hiệu trang thái của các thanh ghi dịch a, b, c, d, e, f và g một cách tương ứng là $(a_0^i, ..., a_{60}^i)$, $(b_0^i, ..., b_{45}^i)$, $(c_0^i, ..., c_{46}^i)$, $(d_0^i, ..., d_{38}^i)$, $(e_0^i, ..., e_{36}^i)$, $(f_0^i, ..., f_{58}^i)$ và $(g_0^i, ..., g_3^i)$. Độ dài của các thanh ghi dịch tương ứng là $\eta_a = 61$, $\eta_b = 46$, $\eta_c = 47$, $\eta_d = 39$, $\eta_e = 37$, $\eta_f = 59$ và $\eta_g = 4$

Trong ACORN có 3 hàm: hàm tạo bit khóa từ trạng thái, hàm tính bit phản hồi tổng thể, và hàm cập nhật trạng thái.

II. TẤN CÔNG KHỐI [1], [5]

Tấn công khối được giới thiệu bởi Dinur và Shamir tại EUROCRYPT 2009 [5]. Cuộc tấn công được nhìn nhận như là sự tổng quát hóa phép tấn công vi sai bậc cao và tấn công đại số vi sai bậc 4. Mục đích của cuộc tấn công là khôi phục khóa mật của một hệ mật mã. Trong mô hình tấn công ban đầu, đối phương có một hộp đen để tìm đa thức chưa biết, Q được xây dựng dựa trên lk biến bí mật và lv biến công khai. Đối phương cũng được giả thiết là biết một bit đầu ra duy nhất.

Tấn công khối cũng là một dạng tấn công đại số nhằm khôi phục biến bí mật của hệ mật mã bằng cách thiết lập và giải hệ phương trình đa thức được xác định bởi hệ mật mã. Hầu hết các hệ mật khóa đối xứng được xác định bởi một đa thức chính duy nhất trong GF(2), trong đó có chứa các biến bí mật (ví dụ, khóa mật) và các biến công khai (ví dụ, bản rõ, bản mã, véc tơ giá trị ban đầu, dữ liệu kết nối). Các biến thể của hệ phương trình có thể được chuyển đổi bằng cách thay đổi các biến bí mật và các biến công khai.

Ý tưởng cơ bản của tấn công khối là tạo ra đủ số lượng cần thiết các phương trình bậc thấp bằng cách sử dụng các biến công khai. Kẻ địch có thể giải hệ phương trình bậc thấp được tạo ra để khôi phục các biến bí mật. Các hệ phương trình bậc thấp được thiết lập bằng việc tính phương trinh đa thức chính tại tất cả các giá trị có thể của các biến công khai chuyên biệt và tổng lại ta nhận được hệ phương trình. Nói chung, tấn công được thực hiện qua 2 giai đoạn: giai đoạn tiền xử lý và giai đoạn online.

A. Giai đoạn tiền xử lý

Trong giai đoạn tiền xử lý, kẻ địch có thể tiếp cận cả biến bí mật và biến công khai. Mục đích của giai đoạn này là thiết lập hệ phương trình tuyến tính của các biến bí mật bằng cách chọn các khối một cách phù hợp. Giả sử, đa thức hộp đen là $Q(K, V)$ được xây dựng dựa vào lk biến bí mật trong K và l_v biến bí mật trong V. Kẻ địch sử dụng các biến bí mật và công khai để xây dựng biến thể tuyến tính của đa thức hộp đen Q. Trong giai đoạn tiền xử lý, kẻ địch chọn khối cỡ l_c, với $1 \leq l_c \leq l_v$ và chọn ngẫu nhiên l_c biến công khai v_i thuộc V. Với tập các biến công khai được chọn, kẻ địch tính tất cả các giá trị có thể của đa thức hộp đen và kết hợp tất cả các giá trị đó lại để tìm xem có quan hệ tuyến tính hay không.

B. Giai đoạn online

Trong giai đoạn online, kẻ địch tiếp cận các biến công khai và các khối nhận được trong giai đoạn tiền xử lý, để cố gắng khôi phục các biến bí mật. Kẻ địch tính toán và lấy tổng tất cả các đầu ra của đa thức chính trên tất cả các giá trị có thể của khối đã được xác định ở vế phải và trái của phương trình tuyến tính tương ứng. Khi đó, hệ phương trình nhận được có thể giải bằng phương pháp khử Gauss để phục hồi các biến bí mật. Thuật toán 1 giới thiệu cuộc tấn công khối tổng quát [1]

Algorithm 1 Algorithm for Cube Attack

Input: Output kesytream bits, Number of linearity test, Initial cube size, Number of cubes tested
Output: Secret variables of the cryptosystem

function PREPROCESSING PHASE
  Select a random cube: Estimate the degree, d of the polynomial, choose a initial cube size $l_c \leq d - 1$ and select a subset of l_c public variables v_i
  Do the linearity test and construct the linear equation for Number of Cubes Tested do
  for Number of Cubes Tested **do**
    for Number of linearity test **do**
      if nonlinear **then**
        if Cubes Tested < Number of Cubes Tested **then**
          Select another cube of size l_c and do the linearity test
        else
          Increase the number of cube variables l_c
        end if
      else
        Compute the coefficients of the secret variables by summing over all the possible values of the cube
      end if
      if all the coefficinets are zero **then**
        Select another subset of l_c public variables
      else
        Output the coefficients
        Construct the linear equations
      end if
    end for
  end for
  Do the preprocessing phase till sufficient number of linear equations are generated
end function

1: **function** ONLINE PHASE
2: Find the right hand side of the linear equations
3: **for** Each possible cube found in prepreocessing phase **do**
4: Compute the output bit
5: Sum all the output bits for all the possible values of the cube
6: **end for**
7: Solve the linear equations to recover the secret variables
8: **end function**

C. Tấn công khối lên ACORN

Một cuộc tấn công có thể được thực hiện hoặc trong giai đoạn khởi tạo, giai đoạn mã hóa hoặc giai đoạn giải mã. ACORN không nhận bất kỳ tín hiệu trung gian nào trong suốt giai đoạn tạo thẻ xác thực, do đó tấn công khối không áp dụng trong giai đoạn này.

1) Tấn công khối trong giai đoạn khởi tạo: Trong giai đoạn khởi tạo, khóa, véc tơ khởi tạo và dữ liệu liên kết được nhập vào trạng thái trung gian của ACORN. Nói chung, khối được chọn hoặc từ khóa đầu vào, véc tơ khởi tạo hoặc tập dữ liệu kết nối đầu vào. Tuy nhiên, kẻ địch phải có khả năng tạo ra các bit khóa nếu đã chọn được khối bit từ khóa đầu vào.

Cuộc tấn công đòi hỏi bit đầu ra từ hàm đầu ra ACORN với $n_c x 2^{l_c}$ véc tơ khởi tạo được chọn, ở đây, n_c và l_c là tổng số và độ dài của khối được chọn một cách tương ứng. Vì vậy, độ phức tạp của việc tìm ra vế phải của hệ phương trình tuyến tính trong giai đoạn online của cuộc tấn công là không nhiều hơn $n_c x 2^{l_c}$. Nhận được kết quả này do ta sử dụng phương pháp khử Gauss, phương pháp này yêu cầu số phép toán xấp xỉ n_c^3 với $n_c = 128$. Tổng độ phức tạp của cuộc tấn công khoảng $n_c x 2^{l_c} + n_c^3$. Nếu hệ phương trình tạo ra không đủ, nhĩa là $n_c < 128$, kẻ địch có thể khôi phục khóa một phần bằng cách giải hệ phương trình và phần còn lại của khóa có thể tìm được bằng cuộc tìm kiếm toàn diện.

2) Tấn công trong giai đoạn mã hóa: Trong giai đoạn mã hóa, các bit bản rõ được nhập vào trạng thái trung gian của ACORN. Do đó, bản rõ trong giai đoạn này có thể coi như một biến công khai và các khối có thể được chọn từ tập bản rõ đầu vào. Trong trường hợp này, suốt giai đoạn tiền xử lý kẻ địch dùng các bit bản rõ để thiết lập hệ phương trình tuyến tính theo các bit trạng thái. Tổng độ phức tạp tính toán của cuộc tấn công khôi phục trạng thái cần $n_c x 2^{l_c} + n_c^3$ phép toán, ở đây $n_c = 128$.

3) Kết quả tấn công ACORN: Các nhà khoa học Md Iftekhar Salam, Harry Bartlett, Ed Dawson, Josef Pieprzyk, Leonie Simpson, and Kenneth Koon-Ho Wong đã sử dụng tấn công khối để rút ngắn vòng của ACORN. Tấn công khối trên 477 vòng khởi tạo của ACORN có thể khôi phục 128 bit khóa với độ phức tạp khoảng 2^35. Các chuyên gia chỉ ra, hệ phương trình tuyến tính liên quan đến trạng thái ban đầu của phiên bản đầy đủ của ACORN có thể dễ dàng tạo ra để thực hiện cuộc tấn công khôi phục trạng thái với độ phức tạp $2^{27.8}$.

III. TẤN CÔNG DỰA VÀO CẶP TRƯỢT [2]

Hai cặp (Khóa, Véc tơ khởi tạo) khác nhau mà cùng tạo ra một trạng thái giống hệt nhau tại các thời điểm khác nhau được gọi là cặp trượt. Đối với ACORN – 128, các cặp trượt tạo ra một trạng thái giống hệt nhau ở các thời điểm khác nhau khi khởi tạo hoặc xử lý dữ liệu kết nối có thể tạo ra cùng một trạng thái mà không có sự khác biệt thời gian trong mã hóa dữ liệu kết nối được chọn.

Rất khó để tìm được cặp trượt trong ACORN – 128 một cách trực tiếp. Thay vào đó, chúng ta cố gắng tìm các cặp trượt bắt đầu từ vòng thứ 256. Theo quan sát của các tác giả, ở vòng 256 đầu tiên ánh xạ từ cặp (Key, IV) sang trạng thái của vòng 256 là bijection.

Ký hiệu $s = (0, \ldots, 0, s_{37}, \ldots, s_{292})$ và $c = (0, \ldots, 0, c_{37}, \ldots, c_{292})$ hai trạng thái của vòng thứ 256 của véc tơ khởi tạo. Ký hiệu s^t là trạng thái tại vòng $t + 256$, và ký hiệu s_i^t là bit thứ i của s^t. Với $0 \leq t_c < t_s \leq 1279$ hai trạng thái s^{t_s} và s^{t_c} là đồng nhất nếu và chỉ nếu $s^{t_s - t_c + 1} = F(c, 1, 1, 1)$. Với $t_s \geq 1280$, tức là, $t_s + 256 \geq 1536$, sau đây ta sẽ thấy đây là điều điều kiện đủ nhưng không phải điều kiện cần. Tồn tại các giá trị s và c sao cho $s^{t_s - t_c + 1} = F(c, 1, 1, 1)$ nếu và chỉ nếu có s thỏa mãn $s_0^{t_s - t_c + 1} = 1$ và $s_1^{t_s - t_c + 1} = 0$ với $0 \leq i \leq 35$.

Đặt $t = t_s - t_c + 1$. Chúng ta có thể giải trực tiếp hệ phương trình với $38 \leq t \leq 257$. Tất cả các hệ này là tuyến tính. Mỗi hệ có 37 phương trình tuyến tính trong khi có 256 ẩn, bởi vậy ta có 2^{219} nghiệm. Tổng số nghiệm cho các s như vậy là khoảng 2^{227}. Cận dưới của xác suất để cặp (Key, IV) được lấy một cách ngẫu nhiên là cặp trượt tại véc tơ khởi tạo vào khoảng 2.2^{-29}.

Bây giờ, chúng ta xét khả năng để ACORN – 128 có 2 cặp (Key, IV) khác nhau cùng tạo ra một trạng thái trong giai đoạn xử lý dữ liệu kết nối ở

2 thời điểm khác nhau. Giống như phần trên, ta ký hiệu trạng thái $s = (0, \ldots, 0, s_{37}, \ldots, s_{292})$ ứng với trạng thái của vòng thứ 256 của véc tơ giá trị ban đầu với cặp (Key, IV). Nếu cặp (Key, IV) tạo ra trạng thái như vậy trong giai đoạn xử lý dữ liệu kết nối, với xác suất 2^{-37}, thì cặp trượt có thể nhận được sau khi cài đặt 1280 bit tiếp theo của dữ liệu kết nối tới $(1, 0, \ldots, 0)$. Luôn luôn tồn tại trạng thái như vậy với dữ liệu kết nối được chọn tới 37 bit. Nghĩa là, đối với mỗi cặp (Key, IV) có cặp khác cùng tạo ra một trạng thái tại các thời điểm khác nhau với xác suất 1.

IV. MỘT SỐ ĐỀ XUẤT CẢI TIẾN

Trong tấn công khối, chúng ta nhận thấy, để thiết lập được các hệ phương trình tuyến tính, thì đa thức hộp đen $Q(K, V)$ phải có tính chất tuyến tính với một số khối các biến được chọn, trong đó K gồm l_k biến bí mật, V gồm l_v biến công khai. Đặt $K = \{k_0, k_2, ..., k_{l_k-1}\}$ tập các khóa bí mật, $V = \{v_0, v_2, ..., v_{l_v-1}\}$ tập các khóa công khai. Để tấn công khôi phục khóa bí mật, kẻ địch sử dụng các khóa bí mật và công khai để xây dựng đa thức hộp đen Q. Sau đó, kẻ địch chọn ngẫu nhiên l_c biến công khai từ tập l_v biến công khai và tính đa thức hộp hộp đen Q bằng cách cho các giá trị của các biến không được chọn bằng 0. Ký hiệu Q_c là phương trình nhận được bằng cách lấy tổng tất cả các giá trị có thể của khối.

Sau khi xây dựng được Q, kẻ địch có thể sử dụng BLR test [6] để kiểm tra điều kiện tuyến tính bằng cách chọn ngẫu nhiên 2 véc tơ $x, y \in \{0, 1\}^{l_k}$ và thử $Q[0]+Q[x]+Q[y] = Q[x]+Q[y]$. Phép thử này có tính xác suất, nó xác nhận Q_c có tính tuyến tính nếu phép thử luôn thành công. Trong trường hợp ngược lại, Q_c là phi tuyến, nếu phép thử là không thành công. Xác suất để đa thức Q_c là không tuyến tính là 2^{-j}, với yêu cầu Q_c phải vượt qua BLR test j lần.

Trong tấn công dựa vào các cặp trượt, thì phải xuất hiện các cặp (Key, IV) khác nhau, nhưng cùng tạo ra cùng một trạng thái. Điều này luôn xảy ra, tức là xảy ra với xác suất 1 do tính chất tuần hoàn của dãy các trạng thái của bộ các thanh ghi dịch tuyến tính. Chúng ta có thể cải thiện điều này, bằng cách sử dụng các thanh ghi dịch phi tuyến và sử dụng độ dài khóa, độ dài của véc tơ khởi tạo lớn hơn. Hơn nữa, ta thấy nếu sử dụng khóa hoặc véc tơ khởi tạo nhiều lần, thì kẻ địch có thể lợi dụng các cặp trượt một cách hiệu quả để khôi phục khóa hoặc khôi phục bản rõ.

Với các nhận xét như vậy, chúng tôi có các đề xuất như sau:

- Sử dụng các thanh ghi dịch phi tuyến để thay thế các thanh ghi dịch tuyến tính.
- Tăng độ dài của khóa và véc tơ khởi tạo lên 256, 512. Điều này là khả thi, do các thiết bị ngày càng nhỏ hơn và tốc độ xử lý ngày càng nhanh hơn.
- Mỗi khóa và véc tơ khởi tạo chỉ sử dụng một lần.

V. KẾT LUẬN

Hệ mật mã xác thực ACORN – 128 là một hệ mật tốt, tuy vậy, nó giống như tất cả các hệ mật mã hạng nhẹ khác, đều có những hạn chế nhất định về độ mật. Báo cáo đã tổng hợp 2 dạng tấn công vào ACORN nhằm đề xuất 1 phương án cải tiến để nhận được hệ mật hoàn thiện hơn.

TÀI LIỆU

[1] Md Iftekhar Salam, Harry Bartlett, Ed Dawson, Josef Pieprzyk,Leonie Simpson, and Kenneth Koon-Ho Wong, *Investigating Cube Attacks on the Authenticated Encryption Stream Cipher ACORN*, 2016.

[2] Meicheng Liu, Dongdai Lin, *Cryptanalysis of Lightweight Authenticated Cipher ACORN*, 2016.

[3] Hongjun Wu, *ACORN: A Lightweight Authenticated Cipher*, Submission to CAESAR, 2016.

[4] Hongjun Wu, *ACORN: A Lightweight Authenticated Cipher (v3)*, 2016.

[5] Dinur, I. and Shamir, A., *Cube Attacks on Tweakable Black Box Polynomials*, In A. Joux (Ed.), Advances in Cryptology - EUROCRYPT 2009, Vol. 5479, pp. 278-299, Springer Berlin Heidelberg, 2009.

[6] Blum, M., Luby, M., and Rubinfeld, R., *Self-testing/correcting with applications to numerical problems*, Journal of Computer and System Sciences, 47, pp. 579-595, 1993.