온라인 게임 상의 사용자 인증에 적용 가능한 셀룰러 오토마타 기반 해쉬함수에 대한 충돌쌍 공격

이창훈* · 이제상** · 조성언*** · 김태훈**** · 김수균*****

Collision Attack on Cellular Automata based Hash Function Applicable to Authentication on Online Game

Changhoon Lee* · Jesang Lee** · Sung-Eon Cho*** · Tai-hoon Kim**** · Soo-Kyun Kim*****

요 약

본 논문에서는 온라인 게임 상의 사용자 인증에 적용 가능한 이차원 셀룰라 오토마타 기반 해쉬함수에 대한 충돌쌍 공격을 제안한다. 이것은 라운드 함수에서 사용되는 비선형 함수의 비선형 성질을 이용하여 입력된 메시지 차분을 확률 2^{-28}으로 상쇄시키는 공격이다. 또한, 최근 발표된 Wang의 분석 기법을 이용하여 확률 1로 만족하는 충돌쌍 공격을 제시한다.

ABSTRACT

In this paper, we present a collision attack on hash function with 2-dimensional cellular automata[1], which is useful for providing authentication on online game. This attack can find a collision message pair with 2^{28} computation using property of nonlinear function. We also extend basic attack with probability 2^{-28} to improve attack with probability 1 using Wang's analysis technique.

키워드

해쉬함수, 셀룰러 오토마타, 충돌쌍 공격

Ⅰ. 서 론

셀룰러 오토마타(CA: cellular automata)는 스스로 조직화하고 재생산할 수 있는 모델로서, 국소적 상호작용을 통하여 동시에 상태가 갱신되는 셀들로 구성된 유한 상태 머신이다. 이것은 Neumann에 의해 처음 소개되었으며[2], Wolfram에 의해 처음으로 암호학에 응용되었다[3]. CA의 특징 중에서 확산과 국소적인 상호 작용은 암호시스템을 설계하는데 적합하여 LFSR의 대안으로 제시되었으며, 부울 방정식의 해법, 의사난수 생성기, 암호 알고리즘 설계 등과 같은 다양한 응용분야에서 사용되고 있다. CA는 AND, OR, NOT, XOR와 같은 단순한 연

* 이창훈(주저자)
** 이제상 고려대학교 정보경영공학전문대학원
*** 조성언(교신저자) 순천대학교 정보통신공학부
**** 김태훈 한남대학교 멀티미디어학부
***** 김수균 배재대학교 게임공학과

접수일자 2008. 11. 28

산을 이용하여 상태를 갱신하며, 특히 비선형 성질을 만족하는 CA 법칙을 이용한 암호알고리즘이 국내에서 다수 발표되었다[4,5]. 그러나 확산 효과가 좋지 못하다는 구조적 취약점 때문에 대부분의 기 제안된 암호 알고리즘이 분석되었다[6,7,8,9].

그런데, CA 기반 해쉬함수들은 비교적 좋은 효율성을 갖기 때문에 빠른 시간 내 인증을 요구하는 온라인 게임에 유용하게 사용될 수 있다. 2005년 CA기법을 이용한 해쉬함수 2CAH가 제안되었다[1]. 2CAH는 셀의 동시변환규칙과 비트의 위치변환규칙을 이용하여 고정된 해쉬값을 출력한다.

해쉬함수는 임의 길이의 비트 열을 입력으로 받아 고정된 길이의 비트 열을 출력하는 함수이다. 일반적으로 해쉬함수는 함수 h와 입력 x가 주어지면, $h(x)$를 계산하는 것은 쉬워야 한다. 하지만, 암호학적으로 안전한 해쉬함수는 다음과 같은 성질을 만족해야 한다.

- preimage resistance : 해쉬 값 y가 주어졌을 때 $h(x)=y$를 만족하는 입력 x를 찾는 것이 계산 상 불가능하다.
- second preimage resistance : 입력 x와 출력 $h(x)$가 주어졌을 때, $h(x)=h(x')$을 만족하는 입력 $x \neq x'$를 찾는 것이 계산상 불가능하다.
- collision resistance : $h(x)=h(x')$을 만족하는 서로 다른 임의의 두 입력 쌍 x, x'을 찾는 것이 계산상 불가능하다.

2CAH는 셀의 동시변환규칙과 비트의 위치변환규칙을 이용하여 고정된 해쉬값을 출력한다. 본 논문에서는 동시변환규칙에만 메시지가 입력된다는 점을 이용하여 2^{-28}의 확률로 성공 가능한 충돌쌍 공격을 소개한다. 더 나아가 Wang 등이 제시한[10,11] 메시지 수정 방법을 이용하여 확률 1로 CAH3의 충돌쌍을 찾는다.

본 논문의 구성은 다음과 같다. 2장에서는 셀룰러 오토마타에 대한 기본적인 내용을 소개하고, 3장에서는 2CAH를 설명한다. 4장에서는 동시변환규칙의 차분 성질을 이용한 전체라운드의 충돌쌍 공격을 제시한다. 5장에서는 Wang의 메시지 수정 방법을 이용하여 4장의 충돌쌍 공격을 향상시킨다. 마지막으로 6장은 본 논문의 결론이다.

Ⅲ. 셀룰러 오토마타(CA)

CA는 Neumann과 Wolfram에 의해 스스로 조직화되고 재생산할 수 있는 모델로 소개되었으며[2,3], 최근에는 Chaudhuri, Nandi, Chowdhury 등 여러 학자들에 의해 연구되고 있다[12,13]. CA란 동역학계(dynamic system)를 해석하는 한 방법으로 공간과 시간을 이산적으로 다루고, 이산적인 공간을 셀룰러 공간(cellular space)의 기본단위인 각 셀의 취할 수 있는 상태를 유한하게 처리한다. CA는 배열에 따라서 1차원과 2차원으로 나뉘는데, 본 논문에서 소개되는 2CAH는 2차원 CA로 구성된 해쉬함수이다.

1.1 차원 CA

가장 단순한 구조를 가지는 1차원 CA에서는 모든 셀들이 선형으로 배열되어 있다. 가장 중요한 것은 국소적 상호 작용이 세 개의 셀, 즉 자기 자신과 인접한 두 셀에 의해 이루어지는 반지름이 1인 CA이다. CA를 설명하기 위하여 다음 기호들이 정의된다.

- t : 시간단계
- r : 반지름
- s_i^t : 시간 t에서 i번째 셀의 위치
- s_i^{t+1} : 시간 $t+1$에서 i번째 셀의 위치

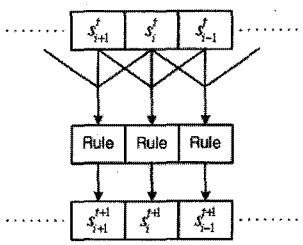

그림 1. 반지름이 1인 CA에 대한 상태전이 함수
Fig. 1 State-transition function of CA (radius=1)

반지름이 1인 CA에 대한 상태전이함수(state-transition function)는 그림 1과 같으며 반지름이 r인 1차원 CA의 상태전이함수는 다음과 같이 정의된다.

$$s_i^{t+1} = f(s_{i+r}^t, \cdots, s_{i+1}^t, s_i^t, s_{i-1}^t, \cdots, s_{i-r}^t)$$

GF(2)상에서 반지름이 1인 CA의 상태전이 함수는 i 번째 셀 s_i^t을 이웃한 2개의 셀(s_{i-1}^t, s_{i+1}^t)과의 상호작용을 통하여 s_i^{t+1}로 갱신한다. 상호작용에 이용되는 상태전이 함수를 법칙(Rule)이라 명하며, 법칙은 부울 함수($Z_{2^3} \rightarrow Z_2$)로 정의할 수 있다. GF(2)상의 서로 이웃한 3개의 셀이 가지는 상태의 경우의 수는 2^3이며, 2^{2^3}의 상태전이 함수가 존재한다. 갱신 법칙의 두 가지 예로서 법칙 78과 92는 표 1과 같이 정의되며, 부울 함수로 정의하면 표 2와 같다.

표 1. 갱신 법칙
Table. 1 Update rule

	111	110	101	100	011	010	001	000
법칙78	0	1	0	0	1	1	1	0
법칙92	0	1	0	1	1	1	0	0

표 2. 상태전이 함수
Table. 2 State-transition function

법칙	논리 함수
78	$s_i^{t+1} = (\overline{s_{i-1}^t} \bullet s_{i+1}^t) \oplus (s_i^t \bullet \overline{s_{i+1}^t})$
92	$s_i^{t+1} = (\overline{s_{i-1}^t} \bullet s_i^t) \oplus (s_{i-1}^t \bullet \overline{s_{i+1}^t})$
150	$s_i^{t+1} = s_{i-1}^t \oplus s_i^t \oplus s_{i+1}^t$

여기서 s_i는 i번째 셀의 상태 값, $\wedge$는 AND 연산, $\vee$는 OR 연산, $\oplus$는 XOR 연산을 각각 의미한다. 법칙의 이름은 모든 입력 값에 대한 출력 값을 순차적으로 나열하여 2진수로 표현한 다음 2진수를 10진수로 변환하여 표기한다. 따라서 법칙 78과 92는 모든 입력 값에 대한 출력 값을 비트열로 표현하면 각각 "01001110"과 "01011100"이며 10진수로 표현할 경우 78과 92이다.

반지름이 2인 CA의 상태전이 함수는 i번째 셀 s_i^t을 이웃한 4개의 셀(s_{i-2}^t, s_{i-1}^t, s_{i+1}^t, s_{i+2}^t)과의 상호작용을 통하여 s_i^{t+1}로 갱신하며, 법칙은 부울 함수($Z_{2^5} \rightarrow Z_2$)로 정의할 수 있다. 서로 이웃한 5개의 셀이 가지는 상태의 경우의 수는 2^5이며, 2^{2^5}의 상태전이 함수가 존재한다.

CA는 셀들에 적용되는 법칙에 따라서 다음과 같이 분류한다. 모든 셀들의 법칙이 XOR 논리로만 이루어진 CA를 linear CA라고 하고, 셀들의 법칙이 XOR/XNOR의 조합으로만 이루어진 CA를 additive CA라고 하고, 셀들의 법칙이 AND-OR 논리로 이루어진 CA를 nonadditive CA라고 한다. CA에서 모든 셀이 같은 법칙을 사용하면 uniform CA라고 하고, 셀에 적용된 법칙을 2개 이상 사용하면 Hybrid CA라고 부른다. 각 셀의 논리에서 고려해야 할 또 다른 것은 CA를 구성하는 양 끝 셀의 경계조건에 따라서 다음과 같이 분류한다. 경계조건이 "0"으로 가정하는 NBCA(Null Boundary CA)와 경계조건이 서로 연결되는 PBCA(Periodic Boundary CA) 등으로 분류한다. 그림 2는 경계조건이 "0"인 Hybrid NBCA의 한 예이다.

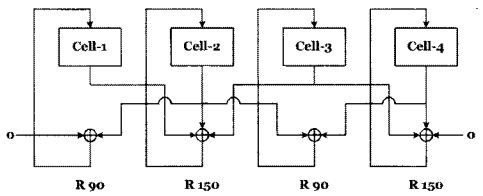

그림 2. Hybrid NBCA
Fig. 2 Hybrid NBCA

2. 2차원 CA

2차원 CA에서는 모든 셀들이 2차원으로 배열되어 있다. 그림 3에서 위 그림은 국소적 상호 작용이 아홉 개의 셀, 즉 자기 자신과 인접한 여덟 개의 셀에 의해 다음 상태에 영향을 미치는 구조를 갖는 2차원 CA이고, 그림 3에서 아래 그림은 국소적 상호 작용이 일곱 개의 셀, 즉 자기 자신과 인접한 여섯 개의 셀에 의해 다음 상태에 영향을 미치는 구조를 갖는 2차원 CA이다.

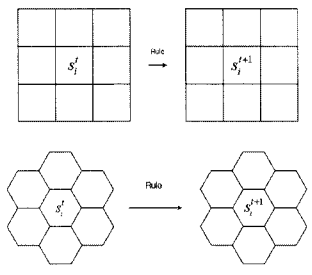

그림 3. 2차원 CA의 상태전이 함수의 예
Fig. 3 State-transition function of 2-dimensional CA

2CAH에서는 셀 공간을 2차원 배열로 구성하고, 1차원 3-이웃 CA의 상태전이 함수를 사용한다.

Ⅲ. 해쉬함수 2CAH

본 장에서는 2005년 한국 멀티미디어 학회 논문지에 김재겸이 제안한 2CAH[1]에 대하여 간략히 소개한다.

1. 표기

X_l : l번째 셀의 현재 상태
X_{l-1} : l번째 셀의 왼쪽 이웃의 현재 상태
X_{l+1} : l번째 셀의 오른쪽 이웃의 현재 상태
Y_l : l번째 셀의 다음 상태
Y_{l-1} : l번째 셀의 왼쪽 이웃의 현재 상태
Y_{l+1} : l번째 셀의 오른쪽 이웃의 현재 상태
$\overline{X_l}$: l번째 셀의 현재 상태의 보수
$\oplus$: bitwise XOR
OR : bitwise OR
$a \ll n$: a의 현재 상태를 정수 n만큼 왼쪽으로 로테이션
M_l : l번째 32비트 메시지 블록
A_3 : 32비트 블록 A의 오른쪽 끝을 0번으로 A의 왼쪽 끝을 31번 비트라고 하였을 때, 24번 비트에서 31번 비트까지의 8비트
A_2 : 32비트 블록 A의 오른쪽 끝을 0번으로 A의 왼쪽 끝을 31번 비트라고 하였을 때, 16번 비트에서 23번 비트까지의 8비트
A_1 : 32비트 블록 A의 오른쪽 끝을 0번으로 A의 왼쪽 끝을 31번 비트라고 하였을 때, 8번 비트에서 15번 비트까지의 8비트
A_0 : 32비트 블록 A의 오른쪽 끝을 0번으로 A의 왼쪽 끝을 31번 비트라고 하였을 때, 0번 비트에서 7번 비트까지의 8비트

2. 셀룰라 공간

2CAH의 셀룰라 공간은 1차원 CA와 달리 셀의 공간을 36개의 사각형들로 분할되어 있는 평면도형으로 설계하였다. 각 셀은 각각 C_0부터 C_{35}번까지의 셀 번호를 가지며, 각 셀은 8비트의 입력 값을 가진다.

표 3. 2CAH 셀룰라 공간
Table. 3 2CAH cellular space

C_0	C_1	C_2	C_3	C_4	C_5	C_6	C_7	C_8
C_9	C_{10}	C_{11}	C_{12}	C_{13}	C_{14}	C_{15}	C_{16}	C_{17}
C_{18}	C_{19}	C_{20}	C_{21}	C_{22}	C_{23}	C_{24}	C_{25}	C_{26}
C_{27}	C_{28}	C_{29}	C_{30}	C_{31}	C_{32}	C_{33}	C_{34}	C_{35}

3. 상태전이 함수

2CAH에 사용되는 상태전이 함수 L과 NL은 다음과 같다.

가. 동시변환 규칙 L

동시변환 규칙 L은 선형함수로 구성되며, 다음과 같이 정의된다.

$$Y_l = (X_{l-1} \oplus X_l \oplus X_{l+1})$$

단, 표3에서 C_0, C_9, C_{18}, C_{27} 셀의 왼쪽 이웃은 각각 C_8, C_{17}, C_{26}, C_{35} 로 정의하고, 표3 C_8, C_{17}, C_{26}, C_{35} 셀의 오른쪽 이웃은 각각 C_0, C_9, C_{18}, C_{27} 셀로 정의한다.

나. 동시변환 규칙 NL

동시변환 규칙 NL은 비선형함수로 구성되며, 식은 다음과 같이 정의된다.

$$Y_l = (X_{l-1} \oplus X_l \oplus X_{l+1}) OR (X_l \oplus \overline{X_{l+1}})$$

단, 표3에서 C_0, C_9, C_{18}, C_{27} 셀의 왼쪽 이웃은 각각 C_8, C_{17}, C_{26}, C_{35} 로 정의하고, 표3 C_8, C_{17}, C_{26}, C_{35} 셀의 오른쪽 이웃은 각각 C_0, C_9, C_{18}, C_{27} 셀로 정의한다.

4. 2CAH

2CAH는 2005년 김재겸에 의하여 개발된 해쉬함수로서 임의 길이의 메시지를 입력받아 160비트 해쉬값을 출력한다. 임의 길이의 메시지가 입력되면 2CAH 해쉬함

수는 메시지의 길이가 512비트의 배수가 되도록 덧붙이기(padding)를 수행하고, Merkle-Damgård 구성 방법을 이용하여 160비트 해쉬값을 출력한다. 2CAH의 압축함수 구조는 그림 4와 같다.

2CAH에서는 키를 사용하는 경우와 고정된 임의 상수값을 사용하는 경우로 분리된다. 본 논문에서는 두 경우 모두 충돌쌍 공격이 가능하므로, 288비트의 임의의 상수 $K_j(0 \le j \le 8)$를 사용하겠다.

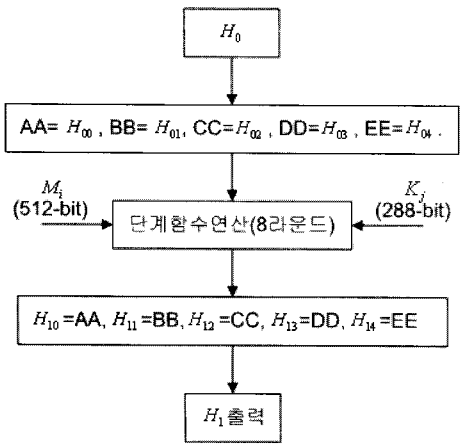

그림 4. 2CAH의 1 블록 압축함수 구조
Fig. 4 2CAH 1 block compression function

2CAH의 초기값, 단계함수 및 출력값은 다음과 같다.

가. 초기값

해쉬함수 연쇄변수의 초기 상수값 H_0는 16진수로 다음과 같다.

$$H_{00} \leftarrow 67452301_x$$
$$H_{01} \leftarrow efcdab89_x$$
$$H_{02} \leftarrow 98badcfe_x$$
$$H_{03} \leftarrow 10325476_x$$
$$H_{04} \leftarrow c3d2e1f0_x$$

이 값을 다음과 같이 초기화한다.

$$AA = H_{00},$$
$$BB = H_{01},$$
$$CC = H_{02},$$
$$DD = H_{03},$$
$$EE = H_{04}$$

나. 단계함수연산

단계함수연산은 표 4와 같이 정의된다.

표 4. 2CAH 해쉬함수의 단계함수 연산
Table. 4 Step function of 2CAH hash function

$C_0 \leftarrow AA_3, C_9 \leftarrow AA_2, C_{18} \leftarrow AA_1, C_{27} \leftarrow AA_0,$
$C_2 \leftarrow BB_3, C_{11} \leftarrow BB_2, C_{20} \leftarrow BB_1, C_{29} \leftarrow BB_0,$
$C_4 \leftarrow CC_3, C_{13} \leftarrow CC_2, C_{22} \leftarrow CC_1, C_{31} \leftarrow CC_0,$
$C_6 \leftarrow DD_3, C_{15} \leftarrow DD_2, C_{24} \leftarrow DD_1, C_{33} \leftarrow DD_0,$
$C_8 \leftarrow EE_3, C_{17} \leftarrow EE_2, C_{26} \leftarrow EE_1, C_{35} \leftarrow EE_0$
$i=0, j=0$
while(i<8) // 라운드함수
{
// 라운드 입력 메시지 및 키 셋팅
$C_1 \leftarrow M[2i]_3, C_{10} \leftarrow M[2i]_2,$
$C_{19} \leftarrow M[2i]_1, C_{28} \leftarrow M[2i]_0,$
$C_3 \leftarrow K[j]_3, C_{12} \leftarrow K[j]_2,$
$C_{21} \leftarrow K[j]_1, C_{30} \leftarrow K[j]_0$
$C_5 \leftarrow K[j+1]_3, C_{14} \leftarrow K[j+1]_2,$
$C_{23} \leftarrow K[j+1]_1, C_{32} \leftarrow K[j+1]_0$
$C_7 \leftarrow M[2i+1]_3, C_{16} \leftarrow M[2i+1]_2,$
$C_{25} \leftarrow M[2i+1]_1, C_{34} \leftarrow M[2i+1]_0$
for $k \leftarrow 0$ **to** 2 // 선형단계1
$Y_l = (X_{l-1} \oplus X_l \oplus X_{l+1})$
$(K[j]_3 \| K[j]_2 \| K[j]_1 \| \mathrm{K[j]}_0) \lll 1$
$(CC_3 \| CC_2 \| CC_1 \| CC_0) \lll 7$
$(DD_3 \| DD_2 \| DD_1 \| DD_0) \lll 13$
for $k \leftarrow 0$ **to** 2 // 비선형단계1
$Y_l = (X_{l-1} \oplus X_l \oplus X_{l+1}) OR (X_l \oplus \overline{X_{l+1}})$
$(K[j]_3 \| K[j]_2 \| K[j]_1 \| \mathrm{K[j]}_0) \lll 1$
$(CC_3 \| CC_2 \| CC_1 \| CC_0) \lll 7$
$(DD_3 \| DD_2 \| DD_1 \| DD_0) \lll 13$

```
// 라운드 입력 메시지 갱신
M[2i] ≪ 16, M[2i+1] ≪ 24
C1←M[2i]3, C10←M[2i]2,
C19←M[2i]1, C28←M[2i]0
C7←M[2i+1]3, C16←M[2i+1]2,
C25←M[2i+1]1, C34←M[2i+1]0
for k←0 to 2 // 선형단계2
  Yl = (Xl-1 ⊕ Xl ⊕ Xl+1)
  (K[j]3 ‖ K[j]2 ‖ K[j]1 ‖ K[j]0) ≪ 1
  (CC3 ‖ CC2 ‖ CC1 ‖ CC0) ≪ 7
  (DD3 ‖ DD2 ‖ DD1 ‖ DD0) ≪ 13
for k←0 to 2 // 비선형단계2
  Yl = (Xl-1 ⊕ Xl ⊕ Xl+1) OR (Xl ⊕ ¬Xl+1)
  (K[j]3 ‖ K[j]2 ‖ K[j]1 ‖ K[j]0) ≪ 1
  (CC3 ‖ CC2 ‖ CC1 ‖ CC0) ≪ 7
  (DD3 ‖ DD2 ‖ DD1 ‖ DD0) ≪ 13
i+=1, j+=1
}
```

다. 출력값

메시지 M의 블록 길이가 l이라고 가정하면, l블록 메시지를 반복적으로 처리한 후의 연쇄변수 H_l은 다음과 같다.

$$H_{l0} \leftarrow C_0 \| C_9 \| C_{18} \| C_{27},$$
$$H_{l1} \leftarrow C_2 \| C_{11} \| C_{20} \| C_{29},$$
$$H_{l2} \leftarrow C_4 \| C_{13} \| C_{21} \| C_{30},$$
$$H_{l3} \leftarrow C_6 \| C_{15} \| C_{24} \| C_{33},$$
$$H_{l4} \leftarrow C_8 \| C_{17} \| C_{26} \| C_{25}$$

Ⅳ. 셀룰러 오토마타 기반 해쉬함수 2CAH에 대한 기본 충돌쌍 공격

본 장에서는 2CAH의 전체라운드 차분 특성을 구성하고, 구성된 차분특성을 이용한 충돌쌍 공격을 소개한다.

1. 2CAH의 전체라운드 차분 특성 분석

본 공격에서 사용하는 메시지 워드의 차분은 다음과 같다.

$$\Delta M = M \oplus M' = (\Delta M_0, \Delta M_1, \cdots, \Delta M_{15})$$
$$\Delta M_{14} = 01000000_x,\ \Delta M_i = 0 (0 \le i \le 15, i \ne 14)$$

7라운드까지 입력되는 메시지의 차분은 0이므로, 7라운드 이후 전체 셀룰러 공간의 차분은 0이 된다. 8라운드에 사용되는 메시지 M_{14}에 최초로 0이 아닌 차분 01000000_x이 입력되어 셀룰러 공간 $C_{8,1}$의 차분은 01_x이 되고, 나머지 공간들은 차분이 0이 된다. 선형단계1 이후 전체 셀룰러 공간의 차분은 확률 1로 그림 5와 같이 구성된다.

단계함수연수 NL에 의해 전체 셀룰러 공간의 차분이 0이 되는 확률을 계산한다. 우선, 비선형 함수 NL은 비트별로 연산한다. 즉, 비선형 함수 NL은 3 비트를 입력으로 1 비트를 출력하는 부울 함수로 생각할 수 있다. 비선형 함수 NL의 차분 분포표는 표 5와 같다. 표 5에 의해서 선형단계1과 비선형단계1 이후에 전체 셀룰러 공간의 차분이 사라질 확률은 2^{-14}이다.

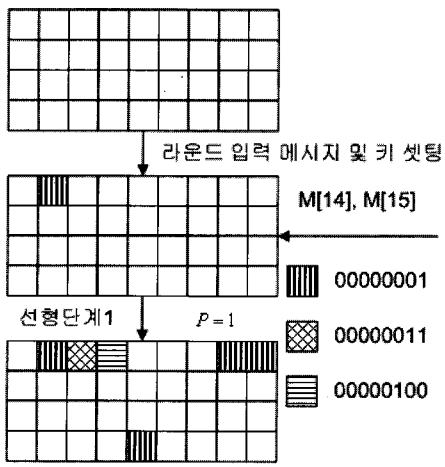

그림 5. 선형단계1 이후의 차분 특성
Fig. 5 Differential characteristic after linear step 1

표 5. 비선형 함수 NL의 차분 분포표
Table. 5 Difference distribution of nonlinear function NL

입력차분 \ 출력차분	0	1	확률
000	8	0	1
001	4	4	2^{-1}
010	4	4	2^{-1}
011	8	0	1
100	4	4	2^{-1}
101	4	4	2^{-1}
110	4	4	2^{-1}
111	4	4	2^{-1}

2. 2CAH 충돌쌍 공격

서로 다른 메시지 M, M'에 대하여 메시지 워드 M_{14}에 01000000_x의 차분이 있는 경우 선형 함수 L과 비선형 함수 NL을 통과하면, 2^{-14}의 확률로 차분이 사라진다. 차분이 0이 된 후 라운드 입력 메시지를 갱신하여 셀룰러 공간에 입력하면 새로운 위치에 메시지 차분이 생기게 된다. 하지만, 위치만 바뀌고 최하위 비트의 차분은 그대로 남게 된다. 선형단계2와 비선형단계2를 통과하면, 위와 마찬가지로 2^{-14}의 확률로 차분이 사라진다. 따라서, ΔM을 만족하는 메시지 M, M'에 대하여 해쉬 출력 값이 같을 확률은 2^{-28}이다. 따라서 주어진 메시지 차분을 만족하는 2^{28}개의 메시지 쌍이 주어지면 평균적으로 1개의 충돌쌍을 찾을 수 있다.

V. 해쉬함수 2CAH에 대한 향상된 충돌쌍 공격

본 절에서는 앞 절에서 제시한 기본 충돌쌍 공격을 Wang의 메시지 수정 공격을 이용하여 확률 1로 향상된 충돌쌍 공격을 소개한다. 2CAH를 분석하기에 앞서 공격 기본 개념이 되는 Wang의 분석 방법을 소개한다.

1. Wang의 분석 방법

Wang의 일반적인 분석 방법은 두 단계로 나뉘어 진다. 첫 번째 단계에서는 덧셈 차분과 ⊕차분을 동시에 이용하여 차분 특성을 찾고 두 번째 단계에서는 차분 특성을 만족하는 충돌쌍을 높은 확률로 찾아낸다.

- 첫 번째 단계 : 차분 특성 찾기
  - 우선 확률 1로 성립하는 부울 함수의 입출력 차분 관계식을 얻는다.
  - 충돌을 일으킬 충돌쌍의 덧셈 차분을 정의한다.
  - 덧셈 차분과 ⊕차분, 그리고 부울 함수의 입출력 차분 관계식을 이용하여 주어진 충돌쌍의 덧셈 차분에 대해 충돌을 일으킬 차분 특성을 찾는다.

- 두 번째 단계 : 차분 특성을 만족시키는 충분 조건의 집합 생성
  - 차분 특성을 만족시키는 연쇄 변수들에 대 한 충분 조건들을 생성해 나간다.
  - 충분 조건들에 대해 모순이 생길 경우, 첫 번째 단계로 다시 가서 모순이 생기지 않도록 하는 새로운 차분 특성을 찾는다.

- 세 번째 단계 : M찾기
  - Basic modification : 입력 메시지가 독립적으로 적용되는 첫 번째 라운드에 대해 차분 특성을 만족시키는 512 비트 메시지 M을 구한다.
  - Advanced modification : 두 번째 라운드에서의 갱신된 워드들에 대한 조건을 만족시키도록 메시지 M을 수정한다. 이때 수정된 메시지 M에 의해 첫 번째 라운드의 차분 특성을 유지하도록 하기 위하여 수정된 메시지 워드를 기준으로 연속적으로 메시지를 수정함으로써 첫 번째 라운드의 차분 특성을 유지하게끔 한다.

- 네 번째 단계 : M' 찾기
  - 두 번째 단계에서 메시지 M이 주어지면, 첫 번째 단계에서 정의한 충돌쌍의 차분을 갖는 M'는 확률 1로 충돌을 일으키게 된다.

2. 2CAH에 대한 향상된 충돌쌍 공격

7라운드까지 입력되는 메시지 차분은 0이므로, 7라운드 이후 전체 셀룰러 공간의 차분은 0이 된다. 8라운드에서 $\Delta M_{14} = 0x01000000$을 입력하면 셀룰러 공간 $C_{8,1}$의 차분은 01_x이 되고, 나머지 공간들은 차분이 0이 된다. 선형단계1 이후 전체 셀룰러 공간의 차분은 그림 5와 같은 차분 특성을 만족하게 되고, 표 5를 이용하여 비선형단계1의 NL을 한번 통과할 때 입력된 차분이 상쇄되도록 입력되는 메시지를 이용하여 셀의 값을 수정한다. 예를 들면, [그림 25]와 같은 차분을 특성을 만족하는 셀룰러 공간의 비트가 다음과 같을 때 비선형단계1의 첫 번째 NL을 통과하면 차분이 0이 된다.

$C_{9,1}[1,0] = \{0,0\}.$

$C_{8,1}[2,1,0] = \{0,0,0\},$

$C_{7,1}[2,1,0] = \{0,1,0\},$

$C_{7,4}[0] = \{0,0\},$

$C_{6,1}[2,1,0] = \{1,0,1\},$

$C_{6,4}[0] = \{0\},$

$C_{5,1}[2,1,0] = \{0,0,1\},$

$C_{5,4}[0] = \{1\},$

$C_{4,1}[2,0] = \{0,0\},$

$C_{4,4}[0] = \{1\},$

$C_{3,1}[0] = \{1\},$

$C_{3,4}[0] = \{1\},$

$C_{2,1}[0] = \{0,0\},$

$C_{1,1}[0] = \{1\}$

위와 같이 각 셀의 차분이 0이 되도록 하는 셀들의 충분조건을 찾고, 선형단계1을 역연산하여 메시지 M_{14}를 찾아낸다. 본 논문에서 찾은 2CAH에 대한 충돌쌍은 표 6과 같다.

표 6. CAH3 충돌쌍 메시지
Table. 6 CAH3 collision message

IV	0x67452301 0xc3d2e1f0	0xefcdab89	0x98badcfe	0x10325476
K	0x13579bdf 0x13579bdf 0x13579bdf	0x02468ace 0x02468ace	0x13579bdf 0x13579bdf	0x02468ace 0x02468ace
M	0xcaf3b62d 0x370d82c5 0xc0a3bd79 0xd7b2d8c7	0x6b2f4072 0x2f948842 0x166854d6 0x324e7e96	0x4fd6901e 0xaf85441f 0x683827de 0x054d265a	0x8fc4b671 0x60db6596 0xdb1933b6 0x3398031a
M'	0xcaf3b62d 0x370d82c5 0xc0a3bd79 0xd7b2d8c7	0x6b2f4072 0x2f948842 0x166854d6 0x324e7e96	0x4fd6901e 0xaf85441f 0x683827de 0x044d265a	0x8fc4b671 0x60db6596 0xdb1933b6 0x3398031a
해쉬 값	0xe3d7d76f 0xffdb37cf	0x9df7eff7	0x26eeed71	0xf97d7bfe

VI. 결 론

본 논문에서는 2차원 셀룰라 오토마타를 기반으로 한 해쉬함수를 분석하였다. 본 논문에서 제안한 공격을 이용하여 충돌쌍을 2^{-28}의 확률로 찾을 수 있었다. 뿐만 아니라 Wang의 메시지 수정 공격을 이용하여 기본 충돌쌍 공격을 확률 1로 향상시켰다. 이를 통하여 이 해쉬함수는 구조적으로 매우 취약함을 알 수 있다. 이 결과들은 CA가 아무리 암호학적으로 유용한 프리미티브 일지라도 유의해서 사용해야 함을 보여준다.

참고문헌

[1] 김재겸, "이차원 셀룰라 오토마타에 기반하는 해쉬함수", Journal of Korea Multimedia Society Vol. 8, No. 5, 2005.

[2] J. V. Neumann. The Theory of Self- Reproducing Automata, A. W. Burks (ed), Univ. of Illinois Press, Urbana and London, 1966.

[3] S. Wolfram, "Cryptography with Cellular Automata," Advances in Cryptology - CRYPTO 85, LNCS Vol.218, pp. 429-432, 1985.

[4] 신상욱, 윤재우, 이경현, "셀룰러 오토마타에 기반한 안전한 해쉬 함수," 한국통신정보보호학회 논문지, 제8권, 제4호, pp. 71-82, 1998. 12.

[5] 이준석, 장화식, 이경현, "셀룰라 오토마타를 이용한 스트림 암호," 한국멀티미디어학회 논문지, 제5권, 제2호, pp. 191-197, 2002. 4.

[6] 류한성, 이제상, 이창훈, 성재철, 홍석희, "셀룰러 오토마타 기반 블록 암호에 대한 부분키 공격", KoreaCrypt 2007.

[7] 임홍수, 홍득조, 성재철, 이상진, "셀룰러 오토마타 기반 스트림 암호 알고리즘에 대한 분석," 한국정보보호학회 동계 학술대회, 제14권, 제2호, pp. 20-24, 2004. 12.

[8] 정기태, 이제상, 장동훈, 성재철, 이상진, "셀룰러 오토마타 기반 해쉬 함수 분석," 한국통신정보보호학회 논문지, 제14권, 제6호, pp. 111-123, 2004. 12.

[9] 최준근, 류한성, 이제상, 홍석희, "이차원 셀룰러 오토마타 기반 해쉬함수에 대한 충돌쌍 공격", 한국방송공학회 동계 학술대회, 2008.

[10] X. Wang, X. Lai, D. Feng, H. Chen, X. Yu, "Cryptanalysis of the Hash Functions MD4 and RIPEMD", Advances in Cryptology -- Eurocrypt' 2005, LNCS 3494, Springer-Verlag (2005), pp. 1-18.

[11] X. Wang, H. Yu, "How to Break MD5 and Other Hash Functions", Advances in Cryptology -- Eurocrypt' 2005, LNCS 3494, Springer-Verlag (2005), pp. 19-35.

[12] P. Chaudhuri, D. Chowdhury, S. Nandi and S. Chatterjee, "Additive Cellular Automata - Theory and Applications", IEEE Computer Society Press, Vol. 1, CA, USA, 1997.

[13] S. Nandi, B. K. Kar, P. Pal Chaoudhuri, "Theory and Applications of Cellular Automata in Cryptography", IEEE Transaction on Computer, Vol. 43, No. 12, 1994.

저자소개

이창훈(Changhoon Lee)

2001년 2월 한양대학교 자연과학부 수학전공 이학사
2003년 2월 고려대학교 정보보호 대학원 이학석사
2008년 2월 고려대학교 정보경영공학전문대학원 공학박사
2008년 4월~2008년 12월 고려대학교 정보보호연구원 연구교수
※관심분야 : 정보보호, 암호학, 멀티미디어/USN 보안

이제상(Jesang Lee)

2003년 2월 고려대학교 수학과
2006년 8월 고려대학교 정보보호 대학원 공학석사
2006년 9월~현재: 고려대학교 정보경영공학전문대학원 박사과정
※관심분야 : 대칭키암호, 해쉬함수, 암호해독

조성언(Sung-Eon Cho)

1989년 2월 한국항공대학교 항공통신정보공학과 공학사
1991년 8월 한국항공대학교대학원 항공통신정보공학과 공학석사
1997년 2월 한국항공대학교 대학원 항공전자공학과 공학박사
1997년 3월 ~ 현재 순천대학교 정보통신공학부 교수
※관심분야 : 무선통신시스템, Wireless USN

김태훈(Tai-hoon Kim)

1995년 성균관대학교 공학사
1997년 성균관대학교 공학석사
2004년 한국정보보호진흥원 선임연구원
2006년 국군기무사령부 사무관
2007년 이화여자대학교 연구교수
2007년~현재 한남대학교 멀티미디어학부 조교수
※관심분야 : 정보보호, 정보보증, 보안수준관리

김수균(Soo-Kyun Kim)

2006년 2월 고려대학교 컴퓨터공학과 (이학박사)
2006년 3월 ~ 2008년 2월 : 삼성전자 통신연구소 책임연구원
2008년 3월 ~ 현재 : 배재대학교게임공학과 전임강사
※관심분야 : Computer graphics