情報保護學會論文誌
第15卷 第6號, 2005. 12

스트림 암호 Edon80의 주기 특성

홍 진,† 박 상 우‡
국가보안기술연구소

Period of Streamcipher Edon80

Jin Hong,† Sangwoo Park‡
National Security Research Institute

요 약

최근 eSTREAM을 통하여 제안된 스트림 암호 Edon80의 주기를 분석한다. Edon80의 설계자들은 2^{103}의 주기를 주장하였다. 본 논문에서는 이것이 키수열 주기의 평균으로서는 사실일 수 있으나, 키와 IV를 랜덤하게 선택했을 경우 무시할 수 없을 정도로 높는 확률로 비교적 짧은 주기의 키수열 또한 나타남을 보인다. 구체적인 예로, 주기 2^{55}의 키수열이 확률 2^{-71}로 나타나며, 주기 2^{11}의 키수열을 생성하는 키-IV쌍이 적어도 하나 존재함을 확인할 수 있다.

ABSTRACT

The period of a recent streamcipher proposal Edon80 is analyzed. The designers of Edon80 had projected a period of 2^{103}. Even though this could indeed be the average keystream period, we show that for a randomly chosen key-IV pair, there exists a non-dismissible probability that the produced keystream will be of relatively short period. More explicitly, a keystream of period 2^{55} may appear with probability 2^{-71}, and one can show the existence of at least one key-IV pair producing a period 2^{11} keystream.

Keywords : *Stream Cipher, Period, Edon80*

Ⅰ. 서 론

현재 유럽 연합의 암호 관련 프로젝트인 ECRYPT에서는 eSTREAM이라는 이름으로 스트림 암호 선정 작업을 진행하고 있다.. 작년 말에 공모가 발표되었으며, 올해 4월이었던 마감일까지 총 34개의 알고리즘이 제출되었다. eSTREAM은 이 중 약 20개를 선정하여 개발자들에게 이를 지난 5월 덴마크 Aarhus에서 열린 Symmetric Key Encryption Workshop (SKEW)을 통하여 소개할 기회를 제공하였다. 스트림 암호 Edon80은 이때 소개된 알고리즘 중 하나이다. Edon80은 FSE2005[1]에서 발표된 quasigroup string e-transformation이라는 개념에 기초하고 있으며, 설계자들은 기존의 결과를 바탕으로 안전성에 대한 일부 증명 가능한 결과를 제시하기도 하였다.

본 논문에서 우리는 Edon80의 주기를 분석한다. Edon80의 설계자들은 발생 키수열의 주기가 2^{103}일 것이라고 주장하였다. 본 논문은, 이것이 평균적으로는 사실일 수 있으나, 무시할 수 없을 만큼 높은 확률로 비교적 짧은 주기의 키수열이 생성되며, 극히 짧은 주기의 키수열도 생성될 가능성이 있음을 보인다. 예를 들어 2^{55}처럼 짧은 주기의 키수열이 확률

접수일 : 2005년 9월 30일 ; 채택일 : 2005년 10월 30일
† 주저자 : jinhong@etri.re.kr
‡ 교신저자 : psw@etri.re.kr

2^{-71}로 나타나며, 주기 2^{11}의 키수열을 생성하는 키-IV쌍이 적어도 하나 존재한다.

Ⅱ. Edon80 소개

본 절에서는 스트림 암호 Edon80[2]을 간단히 소개한다. Edon80은 80비트 안전성을 목표로 하는 하드웨어 기반 암호이다. 80비트 크기의 키와 64비트 크기의 IV를 사용한다.

2.1 Quasigroup

Edon80은 다음 4종의 위수-4 quasigroup을 사용한다.

$\bullet_0$	0	1	2	3
0	0	2	1	3
1	2	1	3	0
2	1	3	0	2
3	3	0	2	1

$\bullet_1$	0	1	2	3
0	1	3	0	2
1	0	1	2	3
2	2	0	3	1
3	3	2	1	0

$\bullet_2$	0	1	2	3
0	2	1	0	3
1	1	2	3	0
2	3	0	2	1
3	0	3	1	2

$\bullet_3$	0	1	2	3
0	3	2	1	0
1	1	0	3	2
2	0	3	2	1
3	2	1	0	3

Quasigroup의 개념에 친숙하지 않은 독자는 이들을 단순히 4개의 원소를 가지는 집합에 주어진 (비가환이며 결합법칙도 만족하지 않는) 곱하기 규칙으로 생각하면 된다. 각 곱하기 규칙을 서로 구별하기 위하여 2비트 숫자로 index를 주었다.

2.2 키 초기화

80비트 크기의 키 K는 키수열 생성에 실제로 사용될 80개의 순차적 quasigroup을 정하는데 사용된다. 키 K를 우선 2비트 부분키들로 나눈다.

$$K=K_0\|K_1\|\cdots\|K_{39}. \tag{1}$$

이들 부분키를 index로 사용하여 각 quasigroup 연산자 $*_i$ $(i=0,1,\cdots,79)$를 다음의 규칙에 따라 $\bullet_j$ $(j=0,\cdots,3)$ 중 하나로 고정한다.

$$*_i = \begin{cases} \bullet_{K_i} & \text{for } 0\le i<40 \\ \bullet_{K_{i-40}} & \text{for } 40\le i<80 \end{cases} \tag{2}$$

따라서 키 K는 사용될 quasigroup을 완전히 결정하고, 반대로 임의의 연속적인 40개의 연산자 $*_i$는 키를 완전히 결정한다.

2.3 IV 초기화

키를 고정시킨 상태에서, IVSetup은 64-bit IV를 유한 수열 $(a_0,\cdots,a_{79})$에 대응시킨다. 이때, 각 a_i는 2비트 값으로 키 및 IV 모두에 의존한다. IVSetup의 정확한 동작 방식은 본 논문의 논의와 무관하므로 자세히 설명하지 않는다.

2.4 키수열 생성

표 1에 주어진 quasigroup 원소들을 살펴보자. 각 행은 오른쪽으로 무한히 계속되는 quasigroup 원소들의 수열이며, 이 중 첫 번째 행은 4를 주기로 반복되는 고정된 무한 수열이다. 첫 번째 열에는 키로부터 미리 결정된 80개의 quasigroup 연산자 $*_i$를 나열했다. 그 다음 열은 IVSetup 과정에서 얻어진 유한 수열을 나타낸다. 나머지 원소 $a_{i,j}$들은 quasigroup 연산자들을 사용하여 제일 위 왼쪽 구석을 시작으로 순차적으로 얻어진다. 구체적으로, 각 $a_{i,j}$는 다음과 같이 계산한다.

$$a_{i,j}=*_i(a_{i,j-1},a_{i-1,j}) \tag{3}$$

여기서 $a_{-1,j}=j \pmod 4$ 가 초기 수열로 주어진 최상위 행이며 $a_{i,-1}=a_i$가 IVSetup으로 얻은 초기 상태이다. 마지막으로, 키수열은 가장 아래 행의 짝수 번째 원소들로 주어진다.

$$\text{키수열}=(a_{79,1},a_{79,3},a_{79,5},\cdots). \tag{4}$$

표 1. 키수열 생성

$*_i$		0	1	2	3	0	1	2	…
$*_0$	a_0	$a_{0,0}$	$a_{0,1}$	$a_{0,2}$	$a_{0,3}$	$a_{0,4}$	$a_{0,5}$	$a_{0,6}$	…
$*_1$	a_1	$a_{1,0}$	$a_{1,1}$	$a_{1,2}$	$a_{1,3}$	$a_{1,4}$	$a_{1,5}$	$a_{1,6}$	…
$*_2$	a_2	$a_{2,0}$	$a_{2,1}$	$a_{2,2}$	$a_{2,3}$	$a_{2,4}$	$a_{2,5}$	$a_{2,6}$	…
$*_3$	a_3	$a_{3,0}$	$a_{3,1}$	$a_{3,2}$	$a_{3,3}$	$a_{3,4}$	$a_{3,5}$	$a_{3,6}$	…
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	…
$*_{78}$	a_{78}	$a_{78,0}$	$a_{78,1}$	$a_{78,2}$	$a_{78,3}$	$a_{78,4}$	$a_{78,5}$	$a_{78,6}$	…
$*_{79}$	a_{79}	$a_{79,0}$	$a_{79,1}$	$a_{79,2}$	$a_{79,3}$	$a_{79,4}$	$a_{79,5}$	$a_{79,6}$	…

표 2. 부분 키-상태 쌍 예제

$*_i$		0	1	2	3	0	1	2	3	0	⋯
$\cdot_2$	1	1	2	2	1	1	2	2	1	1	⋯
$\cdot_0$	2	3	2	0	2	3	2	0	2	3	⋯
$\cdot_3$	1	2	2	0	1	2	2	0	1	2	⋯
$\cdot_0$	1	3	2	1	1	3	2	1	1	3	⋯
$\cdot_2$	0	3	1	2	0	3	1	2	0	3	⋯

앞으로의 논의는 대부분 quasigroup 연산자 $*_i$ $(i=0,1,\cdots,79)$들과 동치라고 볼 수 있는 키와 IV Setup 과정 후에 얻어지는 초기 상태 $(a_0,\cdots,a_{79})$를 중심으로 이루어질 것이다. 이들 둘을 합하여 본 논문에서는 키-상태 쌍이라 부르도록 하겠다.

2.5 주기

하나의 행으로부터 새로운 아래 행을 표 1에 나타난 방식으로 계산하는 과정을 quasigroup string e-transformation이라 부른다. 이러한 변환은 주어진 주기 수열을 새로운 주기 수열에 대응시킴은 자명하다. Edon80의 설계자들은 주어진 임의의 행에 e-transformation을 1회 적용할 때 마다 수열의 주기가 (평균적으로) 2.48배 증가한다고 주장한다. 따라서 최종적으로 얻은 키수열의 주기는

$$4\times(2.48)^{80}\times\frac{1}{2}\sim 2^{105.8} \qquad (5)$$

이 될 것으로 예상할 수 있다. 여기서 첫 번째 항 4는 시작이 되는 초기 수열의 주기를 나타낸 것이며, 끝 항 $\frac{1}{2}$은 마지막 행의 결과에서 반만을 실제 키수열로 출력하기 때문에 곱한 것이다.[1] 설계자들은 이러한 논리를 제공하면서도 실제로는 2^{106}이 아닌 2^{103}의 주기를 제시하였는데, 사소한 계산 실수가 있었던 것으로 보인다.

설계자들은 키수열의 사용에 있어서 길이 제한을 설정하지 않았다. 따라서 최대 2^{103}비트까지의 키수열 사용을 허락한 것으로 볼 수 있을 것이며, 한발 물러선다 하더라도 80비트 안전성을 추구하는 스트림 암호가 일반적으로 제공하는 2^{80}까지는 허락한다고 볼 수 있다.

1) 엄밀히는 최하위 열 주기의 짝·홀수 여부에 따라 키수열의 주기를 계산하는 방식은 달라질 것이나, 본 논문에서는 주기의 근사치에만 관심을 가지므로 고려하지 않는다.

표 3. d-행, 주기-p 키-상태 쌍의 개수

d	$p=4$	$p=8$	$p=16$
5	7.38	11.49	13.30
6	9.36	13.58	15.68
7	11.04	15.63	18.01
8	12.97	17.71	20.30
9	14.75	19.76	22.55
10	16.63	21.81	24.77
11	18.44	23.85	26.96
12	20.30	25.88	29.13
13	22.13	27.91	31.29
14	23.97	29.94	33.44
15	25.81	31.96	35.57
16	27.65	33.98	
17	29.49		
18	31.33		

Ⅲ. 바람직하지 않은 키-상태 쌍

본 절에서는 표 1을 구체적인 quasigroup의 원소들로 채우되 가장 아래 행이 주기 4의 수열이 되도록 할 것이다. 해당 키-상태 쌍은 결국 주기 2의 키수열을 가져올 것이다.

3.1 부분적인 키-상태 쌍

표 2에 주어진 순차적인 5회의 quasigroup string e-transformation을 살펴보자. 이들 각 행의 주기가 4임을 주목하자. 첫 2열에 대한 전수조사를 통하여 이와 같이 모든 행의 주기가 4이며 5행으로 이루어진 부분 키-상태 쌍을 $166\sim 2^{7.38}$개 찾을 수 있었다.

동일한 방식으로 작은 크기의 d와 p에 대해서, d개의 행으로 이루어진 주기 p의 키-상태 쌍을 전수조사하여 그 결과를 표3에 적어보았다. 표에 실제로 적은 값들은 찾은 개수의 log값이다. 예를 들어 첫 번째 항은 5개의 행으로 이루어진 주기 4의 키-상태 쌍이 대략 $2^{7.38}$개 존재함을 의미한다. 표에서 비어있는 부분은 시간 및 자원의 한계로 구하지 못한 값들이다.

각 열을 따라 아래 방향으로 내려가며 값을 살펴보면, 숫자들이 거의 등차수열을 이루며 증가함을 확인할 수 있다. 이를 근거로 40개의 행으로 이루어진 키-상태 쌍에 대하여 표 4의 결과를 추정할 수 있다. 이들 값이 정확하지는 않더라도 적어도 근사적으로는 사실일 것으로 기대할 수 있다.

표 4. 짧은 주기의 40-행 키-상태 쌍 개수 추정

	$p=4$	$p=8$	$p=16$
$d=40$	71.81	82.46	88.57

3.2 주기-2의 키-상태 쌍

40-행 주기-4의 키-상태 쌍 약 2^{72}개 중 임의로 하나를 선택해 보자. 이러한 부분 키-상태 쌍의 최하위 행은 물론 주기 4의 수열이며, 이는 약 $(1/4)^4$의 확률로 최상위 행의 초기 수열 $(0,1,2,3,0,\cdots)$과 동일할 것이다. 작은 개수의 행에 대해서는 이것이 대체적으로 사실임을 실험적으로 확인할 수 있었다.[2] 따라서 최하위 행이 초기수열과 동일한, 40-행 부분 키-상태 쌍이 대략 2^{64}개 존재할 것으로 기대할 수 있다. 부록에 이의 구체적인 예를 하나 제시하였다.

이제 이러한 특별한 형태의 40-행 부분 키-상태를 하나 고정하자. 선택한 40-행 부분 키-상태 쌍의 복사본을 본래의 키-상태 쌍 바로 아래에 이어서 붙이자. 이렇게 연결할 수 있는 것은 바로 40번째 행이 복사본의 최상위 행인 주기-4의 초기수열과 동일하기 때문이다. 상위 40개 행이 키를 완전히 결정하여 하위 행의 quasigroup 연산자들을 결정한다는 사실도 실제로는 이러한 과정이 $*_i = *_{i+40}$ $(i=0,1,\cdots,39)$의 규칙을 따르기 때문에 하위 40행에 상위 40행의 복사본을 사용하는 우리의 구성과 배치되지 않는다. 결합된 결과는 표 1에 구체적인 quasigroup 원소들을 채운 형태가 된다.

이제 구성한 80-행 키-상태 쌍 최하위 행의 짝수 번째 항이 실제 키수열임을 기억하면 주기 2의 키수열을 생성하는 키-상태 쌍을 얻었음을 알 수 있다. 결국 동일한 키수열 $(1,3,1,3,1,\cdots)$을 출력하는 2^{64}개의 (전체) 키-상태 쌍이 존재함을 보인 것이다. 물론 이들 키-상태 쌍 중 일부라도 정상적인 IVSetup 과정을 통하여 도달 가능한지는 명확하지 않으므로 이를 Edon80의 약점으로 보기에는 무리가 있다.

Ⅳ. 바람직하지 않은 키-IV 쌍

본 절에서는ky 표 1을 구체적인 quasigroup의 원소들로 채우되 최하위 행이 상당히 짧은 주기의 수열이 되도록 할 것이다. 구성한 키-상태 쌍의 개수는 충분히 많아 이들 중 의미 있는 만큼이 실제로 키-IV 쌍에 대응될 것이다. 이는 일종의 취약 키-IV 쌍을 찾은 것으로 볼 수 있다.

표를 채우는 과정은 2단계로 이루어 질 것이다. 우선 상위 40개 행을 주기 4, 8, 또는 16이 되도록 채울 것이다. 이후 나머지 40개 행은 상위 행들이 주는 제한은 만족시키면서 최대한 랜덤하게 채우게 된다.

4.1 짧은 주기를 가져오는 키-상태 쌍

40개 행으로 이루어졌으며 각 행이 주기 4인 40-행 부분 키-IV 쌍을 하나 고정하자. 이러한 부분 키-IV 쌍의 마지막 40번째 행 또한 주기가 4임에 주목하자. 40개 행이 결정되었으니 나머지 40개 행에 사용될 quasigroup 연산자 $*_i$ $(i=40,\cdots,79)$ 또한 유일하게 결정되었다. 이들을 주어진 대로 사용하면서 나머지 40개의 초기 상태 $(a_{40},\cdots,a_{79})$을 quasigroup의 원소로 랜덤하게 채운다.

설계자들의 논리를 그대로 따르면 마지막 80번째 행의 주기는 약 $4\times(2.48)^{40} \sim 2^{54.41}$로 예상할 수 있다. 이는 설계자들이 제시하는 값 2^{103}과는 큰 차이가 있는 주기 $2^{53.41}$의 키수열에 해당한다. 뿐만 아니라 이는 80비트 안전성에 대응되는 2^{80}에도 미치지 못하는 값이다.

표 4를 살펴보면 총 2^{72}개의 40-행 키-상태 쌍이 존재함을 알 수 있고, 하위 40개 행에 quasigroup 원소를 랜덤하게 선택하여 넣는 자유도가 80비트만큼 있음을 생각하면 다음의 결론에 도달할 수 있다.

- 주기 2^{53}의 키수열을 생성하는 키-상태 쌍이 최소한 2^{72+80}개 존재한다.

좀 더 정확한 표현은 "그 평균 키수열 주기가 2^{53}이 되는 2^{72+80}개의 원소로 이루어진 일군의 키-상태 쌍이 있다"는 것이다. 그러나 전체적인 모습을 보는 데에는 큰 무리가 없으므로 앞으로도 이러한 정확한 표현 보다는 위와 같이 대략적인 표현을 사용하겠다.

주기 8 또는 16의 40-행 키-상태 쌍으로 출발하면 각각 다음 결론에 도달한다.

- 주기-2^{54}의 키-상태 쌍 2^{82+80}개,
- 주기-2^{55}의 키-상태 쌍 2^{89+80}개

이상이 존재한다.

2) 실험을 통하여 나타난 중간 수열을 실제로 살펴보면 해당 확률이 $(1/4)^4$보다 약간 큰 1/240일 것으로 예상할 만한 근거를 얻을 수 있으나, 이는 전체적인 논리 진행에 큰 영향을 미치지 않으므로 여기서 논의하지 않는다.

표 5. 짧은 주기의 34-행 키-상태 쌍 개수 추정

	$p=4$	$p=8$	$p=16$
$d=34$	60.77	70.34	75.85

4.2 짧은 주기를 가져오는 키-IV 쌍

이제 앞 소절에서 찾은 키-상태 쌍이 IVSetup를 통하여 실제로 키-IV 쌍에 대응되는지 살펴보는 일이 남았다. 키가 임의로 고정된 상태에서 IVSetup 과정은 64비트 IV를 160비트 상태에 대응시킨다. 따라서 IVSetup이 적절히 랜덤성을 확보하도록 설계되었다는 가정하에서 임의의 키-상태 쌍이 IVSetup을 통하여 구체적인 키-IV에 대응될 가능성은 2^{-96}이다. 이를 바탕으로 다음의 결과를 제시할 수 있다.

- 주기-2^{53}의 키-IV 쌍 2^{56}개,
- 주기-2^{54}의 키-IV 쌍 2^{66}개,
- 주기-2^{55}의 키-IV 쌍 2^{73}개

이상이 존재한다. 전체적으로 2^{80+64}개의 키-IV 쌍이 존재하므로 랜덤하게 선택한 키-IV 쌍이 생성하는 키수열이

- 주기-2^{53}일 가능성은 2^{-88},
- 주기-2^{54}일 가능성은 2^{-78},
- 주기-2^{55}일 가능성은 2^{-71}

이상이다. 이 중 2가지는 2^{-80}보다 크므로, 해당 길이의 키수열 사용이 허락되었다면 유효한 공격에 해당된다. 물론 설계자들은 2^{103}의 주기를 제시하였으므로 이러한 짧은 길이의 키수열 사용이 허락된 것으로 보는 것이 타당할 것이다.

V. 주기의 분포

본 절에서는, 랜덤하게 키 및 IV를 사용할 경우 앞에서 다루어왔던 것보다 약간 긴 주기를 생성하는 키-IV 쌍은 약간 더 자주 마주치게 되며, 극히 짧은 주기의 키수열 또한 사용할 가능성이 있음을 보인다.

5.1 주기별 키-IV 쌍 사용 확률

앞서 구성했던 방식대로 표1을 반드시 상위 및 하위 40개 행으로 나누어야만 하는 것은 아니다. 예를 들기 위해 우선 표 3을 선형적으로 확장하여 짧은 주기를 가지는 34-행 키-상태 쌍의 수를 계산하자. 결과는 표 5에 정리하였다. 이러한 34-행 키-상태 쌍으로부터 출발하여, 우선 35에서 40번째에 해당하는 6개 행을 랜덤하게 선택한 키-상태 값으로 채우고, 나머지 하위 40개 행을 상위 40개 행이 가져오는 제한 내에서 최대한 랜덤하게 채우자. 이러한 방식은 $4\cdot 6+80=104$ 비트의 자유도를 확보한다. 최하위 행의 주기는 $4\times(2.48)^{16}\sim 2^{62.28}$로 예상할 수 있으며, 키-상태 쌍과 키-IV 쌍을 대응시키면서 96-비트 자유도를 잃으므로 다음의 결과를 얻을 수 있다.

- 주기-2^{61}의 키-IV 쌍 2^{69}개,
- 주기-2^{62}의 키-IV 쌍 2^{78}개,
- 주기-2^{63}의 키-IV 쌍 2^{84}개

이상이 존재한다. 이들 주기는 모두 2^{103} 및 2^{80}보다 짧다. 랜덤하게 키 및 IV를 사용하는 경우 이들 짧은 주기의 키수열을 사용하게 될 가능성은 다음과 같다.

- 주기-2^{61} 사용 확률 2^{-75},
- 주기-2^{62} 사용 확률 2^{-66},
- 주기-2^{63} 사용 확률 2^{-60}.

Edon80은 80비트 안전성을 목표하므로 이들 확률 값이 2^{-80}보다 크다는 점은 약점으로 볼 수 있다.

지금까지의 설명으로 앞 절에서 제시한 것보다 약간 더 긴 주기의 키수열을 찾는다면 약간 더 큰 확률로 이를 접할 가능성이 있음을 보았다.

그림1은 이러한 관계를 그래프로 나타낸다. 예를 들어 그래프의 가장 왼쪽에 위치한 "△"은 키-IV 쌍을 랜덤하게 선택한 경우 $p=16$을 활용하여 구성한

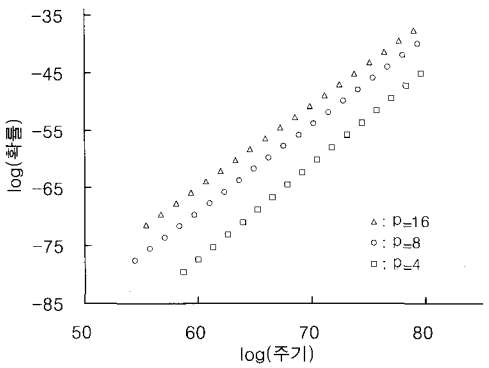

그림 1. 주기 / 확률 tradeoff

주기 $2^{55.4}$의 키수열을 최소한 $2^{-71.4}$의 확률로 접할 수 있음을 의미한다. 그래프를 살펴보면 $p=32$ 또는 $p=64$을 사용할 경우 더욱 강력한 결과를 얻을 수 있음을 쉽게 예측할 수 있다. 그러나 시간 및 연산 능력의 한계로 이를 시도하지 아니하였다.

한 가지 유념할 점은 지금까지 제시한 확률들은 해당 주기의 키수열이 나타날 가능성의 하한이라는 점이다. 제시한 값이 이들 짧은 주기의 수열이 나타날 실제 확률과 비슷한지는 알 수 없다. 예를 들어 m의 임의의 약수 n에 대하여 주기-n의 수열은 항상 주기-m의 수열이므로 위의 값들이 실제 상황보다 낮은 값임은 분명하다. 또한 동일한 주기의 수열을 여러 p값을 사용하여 얻을 수 있으므로, 위에서 제시한 값 중 동일한 주기에 해당하는 3개 값을 합하여도 하한이 될 것이다. 본 논문에서는 위에서 제시한 값들만으로도 Edon80의 주기와 관련된 문제성을 지적할 수 있으므로 이러한 복잡한 작업은 다루지 않기로 하였다.

5.2 매우 짧은 주기 키수열의 존재성

지금까지 본 논문에서는 표1을 40번째 행에서 나누어 보고, 또한 40번째 행 위에서도 나누어 보았다. 본 소절에서는 40번째 행 아래에서 나누어 보도록 하겠다.

소절3.2로 돌아가 우선 40번째 행에 초기 수열 $(0,1,2,3,0,\cdots)$이 나오도록 하는 2^{64}가지 40-행 부분 키-상태 쌍 중 하나로 상위 40개 행을 채우자. 그 다음 전과 같이 이의 복사본을 하위에 결합하되 하위 16개 행의 quasigroup 원소 $a_{64},\cdots,a_{79}$는 랜덤하게 선택하여 넣는다. 이로 32비트의 자유도를 확보할 수 있다. 계산 $4\times(2.48)^{16}\times\frac{1}{2}\sim 2^{19.97}$을 통하여 주기 2^{20}의 키수열을 생성하는 키-IV 쌍이 2^{64+32}개 존재함을 알 수 있다. 이들 중 임의의 것이 정상적인 IVSetup과정으로 도달 가능할 확률이 2^{-96}이므로, 매우 짧은 2^{20}의 주기를 가지는 키수열을 생성하는 키-IV 쌍이 1개 이상 존재할 것으로 기대할 수 있다. $p=16$을 사용하면 더욱 짧은 2^{11} 주기 키수열의 존재성을 보일 수 있다. 이러한 결과를 그림2에 정리하였다.

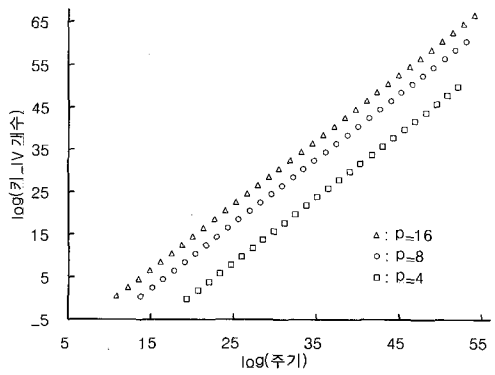

그림 2. 키-IV 쌍 개수 / 주기 tradeoff

물론 랜덤하게 사용한 키-IV 쌍이 바로 이들 소수의 키-IV 쌍 중 하나가 될 가능성은 매우 낮으나, 해당 주기가 극히 짧으므로 Edon80의 사용에 위협이 된다고 할 수 있을 것이다. 전과 마찬가지로 여기 제시한 매우 짧은 주기 키수열의 존재성은 개수의 하한을 통한 논리일 뿐이다. 실제로는 이러한 키-IV 쌍이 여러개 존재할 수 있으며 여기서 제시한 방법과 전혀 다른 방법으로 구성이 가능할 여지도 있다.

마지막으로, 그림2의 위쪽 두 개 "△"를 살펴보는 것 또한 의미 있음을 지적한다. 예를 들어 가장 위쪽의 점은 2^{54}의 주기를 가져오는 2^{67}개 키-IV 쌍이 존재함을 의미한다. 이들 중 하나를 사용하게 될 확률은 2^{-77}로 안전성 기준 2^{-80}보다 높다. 물론 더욱 큰 p 값을 사용하면 더욱 강력한 결과를 얻을 것으로 예상할 수 있다.

Ⅵ. 결 론

본 논문에서 우리는 스트림 암호 Edon80의 키-상태 쌍 중 많은 수가 주기 2의 키수열을 생성함을 보였다. 뿐만 아니라, 임의로 선택한 키-IV 쌍이 주기 2^{55}의 키수열을 생성할 확률이 2^{-71} 이상이라는 사실도 확인하였다. 주기 2^{63}의 키수열은 적어도 2^{-60}의 확률로 나타나며, 주기 2^{11}의 키수열을 가져오는 키-IV 쌍이 적어도 하나 존재한다.

위에서 다룬 주기들은 모두 설계자들이 제시한 값인 2^{103}에 비교해서 매우 작다. 이들 숫자들은 심지어는 구체적인 언급이 없는 한 80비트 안전성의 암호에 기대하게 되는 주기인 2^{80}보다도 작다. 이들 짧은 주기의 키수열이 2^{-80}보다 큰 확률로 생성됨은 분명한 약점임에 틀림없다. 또한 이와 같이 나쁜 특성을 가지는 키-IV 쌍들 혹은 취약 키-IV 쌍을 효율적으로 분류하는 것은 현재로서는 어렵고 따라서 그 사용을 쉽게 피할 방법도 없다.

Edon80의 평균 주기가 설계자의 제안처럼 2^{103}이라는 것을 받아들이면, 본 논문에서 살펴본 결과는 키수열 주기의 분포가 매우 넓으며, 이 중 유효한 개수의 키-IV 쌍이 위험한 수준으로 짧은 주기의 키수열을 발생시킴을 의미한다. 한 가지 유념할 점은 본 논문의 분석은 짧은 키수열 생성의 확률에 대한 아주 대략적인 하한을 주었을 뿐이라는 점이다. 따라서 실제 상황은 본 논문에서 제시한 것보다 나쁠 가능성이 있다.

비록 본 논문의 분석 결과가 키 혹은 내부 상태를 복구하는 데에는 활용될 수 없으나, 적어도 Edon80의 주기가 아직 충분히 분석되지 않았음은 확실하게 보여준다. Edon80을 실제 사용하기 위해서는 우선적으로 키-IV의 랜덤한 선택에 따른 키수열 주기의 분포가 파악되어야 하며, 상대적으로 낮은 주기를 가져오는 키-IV의 사용을 막는 방법이 제시되어야 할 것이다.

참 고 문 헌

[1] S. Markovski, D. Gligoroski, L. Kocarev, "Unbiased random sequences from quasigroup string transformations", FSE2005, LNCS 3557, pp. 163-180, Springer, 2005.

[2] D. Gligoroski, S. Markovski, L. Kocarev, M. Gusev, "Edon80-Hardware synchronous stream cipher", eSTREAM, the ECRYPT Stream Cipher Project, Report 2005/007, 2005.http://www.ecrypt.org/stream/에서 찾을 수 있음.

〈著者紹介〉

홍 진 (Jin Hong) 정회원
1994년 2월: 서울대학교 수학과 졸업
1996년 2월: 서울대학교 수학과 석사
2000년 8월: 서울대학교 수학과 박사
2000년 9월~2002년 9월: 고등과학원 연구원
2002년 9월~현재: 국가보안기술연구소 선임연구원
〈관심분야〉 암호 이론

박 상 우 (Sangwoo Park) 정회원
1989년 2월: 고려대학교 수학교육과 졸업
1991년 8월: 고려대학교 수학과 석사
2003년 2월: 고려대학교 수학과 박사
1991년 8월~1999년 12월: 한국전자통신연구원 선임연구원
2000년 1월~현재: 국가보안기술연구소 선임연구원
〈관심분야〉 암호 이론, 정보보호