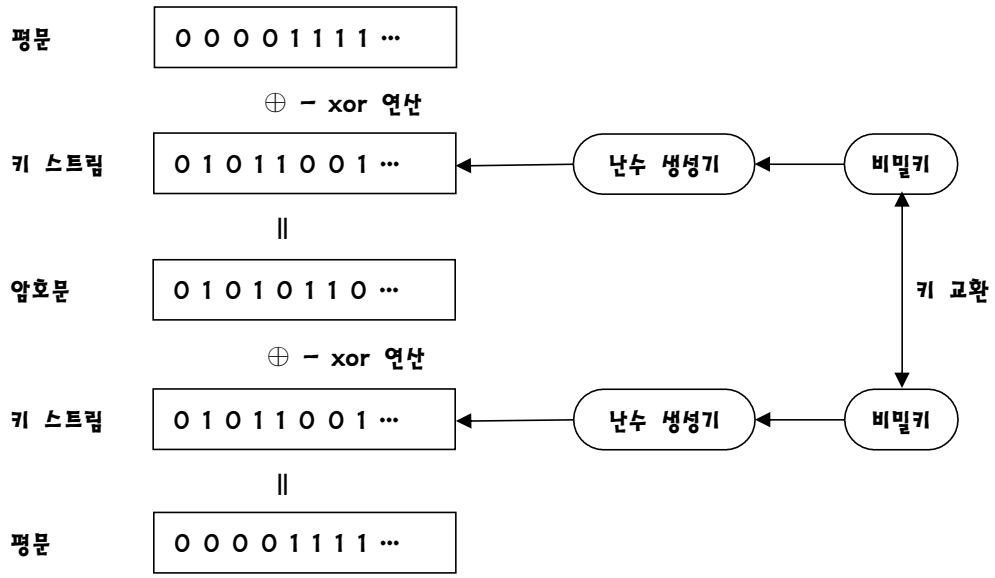
2. 스트림 암호(stream cipher)

1. 스트림 암호 개요

스트림 암호는 평문을 한 번에 'bit 또는 byte 또는 32bit'씩 연속으로 암호화 한다.

〈스트림 암호〉



① 통상적인 스트림 암호는 **유사난수(키 스트림)**를 1bit 단위로 생성하면서 암호화 한다.
→ 평문과 키 스트림의 각 값을 1bit씩 연속적으로 XOR 연산하여 암호문을 얻는다.
② 키 스트림은 일회성 암호키로 사용된다.(일회용 난수표를 유사난수로 대체한 개념)
→ 키 스트림(key steam, 키수열, 유사난수열)의 길이는 평문의 길이와 같다.
→ 그림에서 난수생성기가 키 스트림 생성기이다.
③ 송수신측은 사전에 공유된 비밀키(secret key)를 교환해야 한다.
→ 비밀키는 난수발생기에서 키 스트림을 생성하기 위한 하나의 방법이다.
→ 키 스트림을 생성하기 위한 방법은 여러 가지가 있다.
→ 키 스트림은 사전에 정의된 수열이거나 알고리즘에 의해 생성될 수 있다.
④ 스트림 암호는 하드웨어 구현이 간단하고, 고속이어서 무선통신에 많이 사용된다.
⑤ 스트림 암호 알고리즘으로 RC4가 널리 사용되며, A5/1, A5/2 등의 알고리즘도 있다.
⑥ 스트림 암호 알고리즘은 블록 암호 알고리즘보다 암호화 속도가 빠르다.

유사난수(pseudorandom number) - 의사난수(모조난수)

◈ 난수 구분

	무작위성	예측 불가능성	재현 불가능성	암호 기술에 적용
약한 의사난수	○	×	×	×
강한 의사난수	○	○	×	○
진정한 난수	○	○	○	○

◈ 무작위성

- 통계적으로 한쪽으로 치우침이 없는 수열
- 하지만, 내부적으로 주기를 가지는 수열로 **예측이 가능**하다.

◈ 예측 불가능성

- 이전에 생성된 수열로부터 다음에 발생될 수열을 **예측할 수 없는 성질**
- 이전에 생성된 의사난수열이 공격자에게 알려져도
- 다음에 생성될 의사난수를 공격자는 알 수 없다는 성질
- **소프트웨어**만으로는 진정한 난수를 생성할 수 없다. 해서, **의사난수**라 한다.
- 의사난수 생성 알고리즘을 유사난수생성기(pseudorandom number generator, PRNG)라 함
- 암호 알고리즘을 사용해서 예측 불가능성을 갖는 의사난수열을 생성할 수 있다.
- 예 : 대칭키 암호(AES), 공개키 암호(RSA), 해시함수(SHA-1) 등
- 의사난수생성기가 생성하는 난수는 입력값(seed)이 같으면 출력되는 값도 항상 같다.
- 같은 의사난수를 생성하고 싶으면, 입력값(seed)만 같은 값을 주면 된다.(반복 가능)
- 의사난수 생성에서 입력값(seed, 종자)는 매우 중요하다. seed는 암호키처럼 중요하다.
- 의사난수는 난수처럼 보이지만, 정확한 의미로는 난수가 아니다.

◈ 재현 불가능성

- 같은 수열을 **재현할 수 없는 성질**이다.
- 재현을 위해서는 생성된 수열 그 자체를 저장해 두는 수밖에 없다.
- 현재, 소프트웨어적만으로는 재현 불가능성을 갖는 난수열을 만들 수 없다.
- 즉, 난수 발생 프로그램에서 생성하는 수열은 주기를 가진다.
- **하드웨어**를 이용하면 **재현 불가능성을 갖는 난수열**을 만들 수 있다.
- 즉, 하드웨어에서 발생되는 특정 신호(온도, 소리)를 기초로 난수열을 만들 수 있다.
- 이런 하드웨어를 난수생성기(random number generator; RNG)라 한다.

2. 동기식 스트림 암호시스템(synchronous stream cryptosystem)

① 키 스트림을 생성하기 위해 **내부상태**(internal state)를 **유지**한다.
- 내부상태는 다양하게 구현할 수 있다.
- 내부상태의 한 가지 예로 뒤에서 다룰 LFSR이 있다.
- 이전 내부상태에서 새로운 내부상태와 유사난수**(키 스트림)**를 구한다.

② 동기식 스트림 암호시스템에서 **키 스트림은 평문이나 암호문과 관계없이 생성**된다.
- 키 스트림이 평문과 무관하게 생성되므로 정보 유출의 가능성이 적다

③ 암호화 및 복호화는 생성된 키 스트림과 입력값(평문, 암호문)을 XOR하는 방식이다.

④ 암복호화할 문자열에서 특정 위치의 비트를 변경하면 그 결과도 같은 위치의 비트만이 변경된다. 즉, 문자열의 다른 위치의 비트는 변경되지 않는다.

⑤ 전송 도중에 암호문의 특정 비트가 변경되었을 때 복호화 과정에서 다른 비트에는 영향을 미치지 않는다.
- 하지만, 전송 도중에 암호문의 특정 비트가 손실(제거)되거나 잘못된 비트가 추가되면,
- 오류가 난 시점 이후의 복호화는 실패하게 된다.

⑥ 따라서, 전송 중에 **송수신자 사이에 정기적인 동기화(synchronize)가 필요**하다.
- 암호문에 일정한 간격으로 동기 표시자를 추가하여, 동기화를 확인한다.

⑦ One-time pad 암호는 간단한 동기식 스트림 암호 방식의 한가지이다.
- One-time pad는 암호화를 수행할 때마다 랜덤하게 선택된 키 스트림을 사용한다.

⑧ 실제로, 선형 귀환 시프트 레지스터(LFSR)를 이용하여 구현한다.
- LFSR은 시프트 레지스터의 일종으로 **의사난수를 주기적으로 생성**할 수 있다.

◈ 스트림 암호 / 블록 암호

- 스트림 암호는 블록 암호 알고리즘을 이용하여 구현할 수 있다.
- 스트림 암호 방식의 핵심 요소는 키 스트림 생성기이다.
- 스트림 암호의 안전성은 키 스트림 생성기의 출력에 의존한다.
- 암호화와 복호화 측은 동일한 키 스트림 생성기를 가지고 있어야 한다.

One-time pad 암호

① One-time pad 암호는 **동기식 스트림 암호 방식의 한가지**이다.
→ Gilbert Vernam이 설계하였다.

② One-time pad 암호는 xor 연산을 사용한다.

③ One-time pad 암호에서 평문과 암호문은 어떠한 관계도 존재하지 않는다.
→ 이유는 암호시마다 무질서(random)하게 선택된 키 스트림을 적용하기 때문이다.
→ 공격자가 암호문을 획득하여도 평문 또는 키를 추측할 근거가 존재하지 않는다.

④ 평문에 어떤 패턴이 있어도 생성된 암호문에는 어떤 **통계적 특성**도 존재하지 않는다.
→ 이유는 역시 무질서(random)하게 선택된 키 스트림을 적용하기 때문이다.
→ 단지, 전수조사 공격으로 암호를 깰 수 있다.

⑤ One-time pad 암호는 이상적이지만, 현실에 사용되기는 매우 어렵다.
→ 이유는 암호시마다 무질서(random)하게 선택된 키를 송수신자가 공유해야 하므로

⑥ One-time pad 암호를 현실에서 구현한 것이 선형귀한시프트레지스터(LFSR)이다.
→ LFSR은 시프트 레지스터의 일종으로 의사난수를 **주기적**으로 생성할 수 있다.
→ 여기서, **진정한 난수**와 **의사난수**의 차이점을 정확하게 이해해야 한다.
→ **진정한 난수**는 예측이 불가능하지만 **의사난수**는 예측 가능할 수 있다.

◈ One-time pad / One-time password

- One-time pad 암호 : 일회용 비밀번호(키스트림)를 사용하는 암호시스템
- One-time password : 개체를 인증하기 위한 일회용 비밀번호

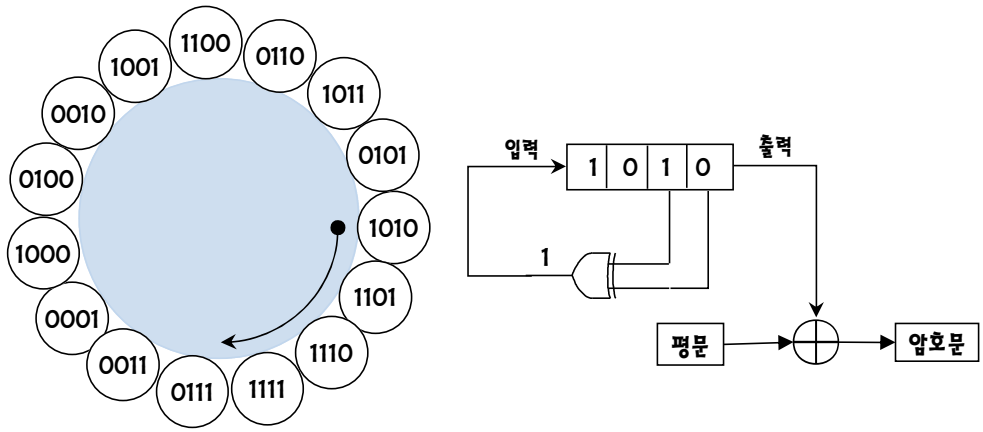
탐구

선형 귀환 시프트 레지스터(LFSR) / 선형 되먹임 시프트 레지스터

① LFSR은 Linear Feedback Shift Register 약어이다.
→ LFSR은 시프트 레지스터의 일종으로 의사난수를 **주기적**으로 생성할 수 있다.

② 다음은 4비트 크기를 가지는 LFSR의 한 예이다. 초기 비트 값은 시드(seed)라 한다.
→ 여기서, 입력 비트는 이전 상태의 **마지막 2비트를 XOR 연산**한 값이다.

〈마지막 2비트를 XOR 연산〉



③ 레지스터에 입력되는 값은 이전 상태 값들의 **선형함수**로 계산되는 구조이다.
→ 생성되는 값은 그 이전 값에 의해 결정된다. LFSR의 동작은 **결정론적**이다.

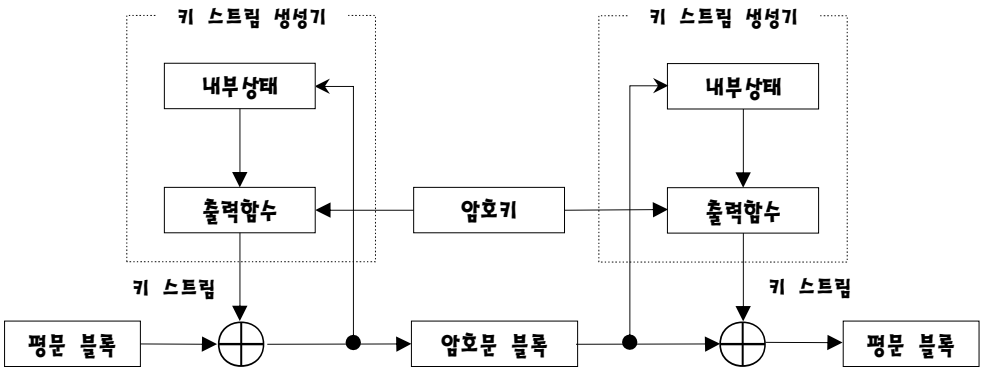
④ LFSR에서 생성되는 값은 그 이전 값에 의해 결정된다.
→ 입력 값에 영향을 미치는 출력을 "탭"이라 한다.(예에서, 마지막 2비트)

⑤ 레지스터가 가질 수 있는 값은 유한개이다. 값은 **주기적**으로 반복되는 수열 형태이다.
→ XOR 연산을 사용할 경우, 레지스터 값은 "0000"은 될 수 없다.(예에서, 최대 15개)
→ 레지스터 크기가 n비트이면, 수열 주기는 $2^n - 1$이다.
→ 하지만, 선형함수에 따라 주기가 길고 무작위로 보이는 수열을 생성할 수 있다.

⑥ 여러 개의 LFSR 출력으로부터 비선형 특성을 갖는 출력을 발생시킬 수 있다.

3. 자기 동기식 스트림 암호시스템(self-synchronous stream cryptosystem)

다음은 자기 동기식 스트림 암호시스템 구조이다.



① 키 스트림을 생성할 때 암호키와 함께 "이전에 암호화된 문자열" 일부를 사용한다.
- 키 스트림의 각 비트가 이전 암호문 비트의 함수로 결정된다.
- **키 스트림의 각 비트는 이전 평문 또는 암호문에 종속적**이다.

② 이와 같은 방식은 암호문을 구성하는 특정 바이트에 오류가 발생하더라도 그 바이트가 더 이상 내부 상태에 영향을 주지 않는 상황이 되면 **자동적으로 다시 암호화 측과 복호화 측**의 키 스트림이 **동기화**된다.
- 이를 "**자기 동기성**"이라 한다.
- 즉, 자기 동기식 스트림 암호 방식이라 한다.

③ 다시 설명하면, 키 스트림 생성기 내부상태가 이전 암호문의 n 바이트에 의존하면 의미 없는 데이터 n 바이트 전송하여 송수신측 키 스트림을 다시 동기화시킬 수 있다.

④ 즉, 전송 도중에 암호문의 특정 비트가 손실(제거) 또는 변경되거나 잘못된 비트가 추가되는 오류가 발생되어도, **일부분만이 복호화에 실패**하게 된다.
- 암호문 전송 도중에 오류가 발생되어도 오류 전파는 유한하다.

⑤ 키 스트림과 암호문이 서로 종속 관계에 있어서 암호문이 해독되기 쉽다.

⑥ 오류를 정정할 수 있는 기능을 포함시킬 수 있다.

⑦ 자기 동기식 스트림 암호는 **비동기식**(asynchronous) 스트림 암호라고도 한다.

⑧ CFB 운용모드는 일종의 **자기 동기식 스트림** 암호 방식의 암호화 모드이다.

기출문제 분석

1. 암호학적으로 안전한 의사난수 생성기에 대한 설명으로 옳은 것은? [2018년 국가 9급]

① 생성된 수열의 비트는 정규분포를 따라야 한다.
② 생성된 수열의 어느 부분 수열도 다른 부분 수열로부터 추정될 수 없어야 한다.
③ 시드(seed)라고 불리는 입력 값은 외부에 알려져도 무방하다.
④ 비결정적(non-deterministic) 알고리즘을 사용하여 재현 불가능한 수열을 생성해야 한다.

♤ **암호학적으로 안전한 의사(pseudo) 난수 생성기**

① 생성된 수열의 비트는 정규분포를 따라야 한다.(×)
→ 난수는 불규칙하므로 정규분포가 될 수 없다.
③ 시드(seed)라고 불리는 입력 값은 외부에 알려져도 무방하다.(×)
→ 시드는 난수 발생을 위한 초기값으로 보안을 위해서는 중요하다.
→ 시드 값을 알면 난수 발생을 추적할 수 있다.
④ 비결정적 알고리즘을 사용하여 재현 불가능한 수열을 생성해야 한다.(×)
→ 의사난수는 컴퓨터에 의해 만들어지는 난수이며, 결정적이다.
→ 의사난수는 재현이 가능하다.
→ 진정한 난수(현실에서 동전 던지기나 주사위 놀이)는 재현이 불가능하다.

정답 : ②

2. 암호화에 대한 설명으로 옳지 않은 것은? [2017년 국가 7급]

① AES는 블록 크기가 192비트이며, 키는 192비트와 256비트 두 가지를 사용한다.
② Rabin은 RSA와 같은 원리로 암호화하고, 2차 합동에 근거하고 있다.
③ DES는 대칭키 방식으로서 16개 라운드로 구성되어 있다.
④ One-Time Pad는 암호화를 수행할 때마다 랜덤하게 선택된 키 스트림을 사용한다.

♤ **AES(Advanced Encryption Standard) – 고급 암호 표준**

• AES는 블록 크기가 192비트이며, 키는 192비트와 256비트 두 가지를 사용한다.(×)
→ AES는 블록 크기가 128비트이며, 키는 128, 192, 256비트 3가지를 사용한다.

정답 : ①

3. 스트림 암호(stream cipher)에 대한 설명으로 가장 옳지 않은 것은? [2019년 서울 9급]

① Key stream generator 출력값을 입력값(평문)과 AND 연산하여, 암호문을 얻는다.
② 절대 안전도를 갖는 암호로 OTP(One-Time Pad)가 존재한다.
③ LFSR(linear feedback shift register)로 스트림 암호를 구현할 수 있다.
④ Trivium은 현대적 스트림 암호로 알려져 있다.

♤ **스트림 암호(stream cipher)**

- Key stream generator 출력값을 입력값(평문)과 AND 연산하여, 암호문을 얻는다.(×)
  → 평문과 키 스트림의 각 값을 1bit씩 연속적으로 XOR 연산하여 암호문을 얻는다.
- 트리비움(trivium)은 효율적인 동기식 스트림 암호의 한 종류이다.

정답 : ①