**4주차 요약 정리**

**1. 대칭키 암호 분류**

* 고전 암호: 시저, 치환, 전치 등 (20세기 이전)
* 현대 암호: 블록 암호 / 스트림 암호 (20세기 이후)

**2. 블록 암호**

* 일정한 크기의 블록 단위로 평문을 암호화
* SPN, Feistel 구조 사용
* 암호 설계 원리: 혼돈(Confusion), 확산(Diffusion)
* 예시: DES, AES

**3. DES (Data Encryption Standard)**

* **64비트 블록**, **56비트 키**, **16라운드 Feistel 구조**
* 구성: 키 스케줄링, 초기/최종 순열, S-box, P-box
* 현재는 **전수조사, 차분/선형 분석**에 취약 → **사용 권장되지 않음**

**4. 3DES (Triple DES)**

* DES를 **2번(2TDES)** 또는 **3번(3TDES)** 반복 적용
* **중간 일치 공격**으로 인해 **보안성 기대보다 낮음**
* 속도 느리고 사용 줄어듦

**5. AES (Advanced Encryption Standard)**

* **128비트 블록**, **128/192/256비트 키 지원**
* 구조: **SubBytes**, **ShiftRows**, **MixColumns**, **AddRoundKey**
* **높은 안전성과 효율성** → 오늘날 가장 많이 쓰이는 블록 암호

**6. 국산 블록 암호**

* **SEED**: KISA 주도, 128비트 블록, 128~256비트 키
* **ARIA**: 국정원 주도, 128/192/256비트 키 지원
* 둘 다 국제 표준 등록됨 (ISO/IEC 등)

**7. 패딩 방식**

* **Zero Padding**: 부족한 공간에 0 채움
* → 단점: 평문이 0으로 끝나면 구분 어려움
* **PKCS#7**: 패딩 바이트 수를 패딩 값으로 사용
* → **정확한 복원 가능**, 보안성↑

**8. 블록 암호 운영 모드**

* **ECB**: 각 블록 독립 암호화 (패턴 노출 위험), 병렬 가능
* **CBC**: 앞 암호문 블록과 XOR, **IV 필요**, 순차 암호화
* **CTR**: 카운터 값 기반 XOR, **병렬 처리 가능**, 키 스트림 사전 생성 가능

**9. 스트림 암호**

* **비트 단위** 실시간 암호화
* 평문과 같은 길이의 **키 스트림 생성 후 XOR**
* 패딩과 블록 운영 방식 불필요
* 예시: **RC4**, **ChaCha20**
* **경량 기기 (IoT, RFID)** 등에 적합

**10. 난수 생성기**

* **TRNG**: 자연현상 기반, **예측 불가한 진짜 난수**
* **PRNG**: 시드 기반, 반복 구조, 효율적이지만 예측 가능성 존재
* **CSPRNG**: 암호학적으로 안전한 PRNG (보안용 사용)

**11. 일회용 암호 (One-time Pad, OTP)**

* **키 스트림을 딱 한 번만 사용**
* **완벽한 이론적 보안성 (Perfect Secrecy)** 제공
* 단점: **키 분배의 어려움**, **TRNG 기반 키 필요**