[2023 JBUCTF] crypto

same\_nonce\_2

Write-Up

문제 개요

제공 파일 : same\_nonce\_2.py

텍스트, 스크린샷, 소프트웨어이(가) 표시된 사진

자동 생성된 설명

33 ~ 48 byte 크기의 secret과, 16byte key, 12byte nonce 생성

secret을 AES-GCM으로 암호화해서 암호화 값인 enc\_secret과 인증 값인 secret\_tag를 생성

enc\_secret 값은 base64 인코딩하여 저장한다.

17 ~ 32 byte 크기의 p1, p2를 생성한 후, p1과 p2를 AES-GCM으로 암호화해서 암호화 값과 인증 값을 합치고, base64인코딩을 한 값을 ciphers 리스트에 저장한다.

1번 입력

위에서 p1, p2를 암호화 한 값과 인증 값을 합친 값을 base64인코딩해서 저장하고 있는 ciphers 리스트의 값들을 출력한다.

2번 입력

enc\_secret의 값을 출력한다.

3번 입력

Base64 인코딩 된 값을 입력 받아 디코딩한 후, 그 값의 마지막 16byte를 인증 값으로, 나머지는 암호문으로 해서 복호화를 한다. 이때 인증 값이 암호문에 맞는 값이 아니면 예외가 발생해서 Failed를 출력하며, 맞는 값이면 그 암호문의 복호화 값이 secret 값인지 확인 후 flag를 출력한다.

4번 입력

exit() 함수를 호출해서 프로그램을 종료한다.

문제 풀이

GCM 상세 참고 자료

# Galois/Counter Mode : <https://url.kr/hs62co>

# NIST SP 800-38D,: [https://csrc.nist.gov/pubs/sp/800/38/d/final](https://csrc.nist.gov/pubs/sp/800/38/d/final%20%20) 꼭 보세요~

텍스트, 도표, 평면도, 스크린샷이(가) 표시된 사진

자동 생성된 설명

텍스트, 도표, 폰트, 스크린샷이(가) 표시된 사진

자동 생성된 설명

텍스트, 도표, 평면도, 스크린샷이(가) 표시된 사진

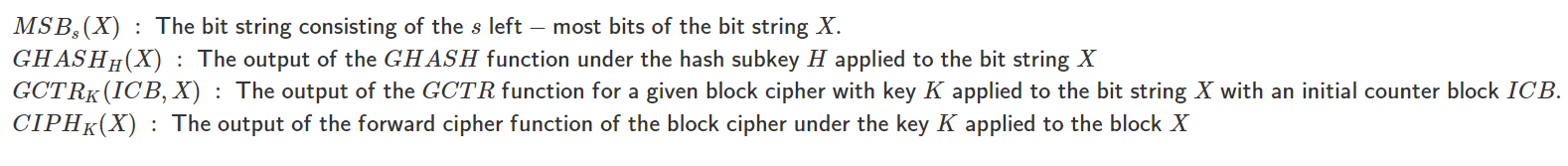
자동 생성된 설명

텍스트, 폰트, 스크린샷, 화이트이(가) 표시된 사진

자동 생성된 설명

텍스트, 폰트, 스크린샷, 대수학이(가) 표시된 사진

자동 생성된 설명



아래의 수식에서 C1, C2는 문제 코드에서 p1, p2의 암호화 값이고, T1 T2는 C1, C2의 인증 값이다.

C3는 enc\_secret이고, T3는 enc\_secret의 인증 값을 나타낸다.

텍스트, 폰트, 친필, 스크린샷이(가) 표시된 사진

자동 생성된 설명

텍스트, 폰트, 스크린샷, 대수학이(가) 표시된 사진

자동 생성된 설명

텍스트, 폰트, 화이트, 대수학이(가) 표시된 사진

자동 생성된 설명

텍스트, 폰트, 화이트, 대수학이(가) 표시된 사진

자동 생성된 설명

텍스트, 폰트, 화이트이(가) 표시된 사진

자동 생성된 설명

위 수식들을 통하여 p1, p2의 암호문 C1, C2의 인증 값인 T1, T2를 이용해서 enc\_secret의 인증 값인 T3를 구할 수 있다는 것을 알 수 있다.

이제 이 수식을 파이썬과 sagemath를 이용해서 구현하겠다.

exploit.py

텍스트, 스크린샷이(가) 표시된 사진

자동 생성된 설명

먼저 알아낼 수 있는 값인 C1, C2, T1, T2, C3의 값을 알아낸다.

그리고 이 값들을 이용해서 T3 값을 계산하기 위해서는 GF(2^128)에서의 연산과 3차 방정식의 해를 구해야 하므로 sagemath라는 툴을 사용하겠다.

calc\_tag.sage

텍스트, 스크린샷이(가) 표시된 사진

자동 생성된 설명

exploit.py를 실행 후, 출력 결과를 exploit.sage 코드에 넣어서 실행하면 3차 방정식이므로 최대 3개의 enc\_secret의 인증 값 후보들이 나온다. 이 인증 값들을 입력을 기다리고 있는 exploit.py에 입력해서 flag를 얻을 수 있다.

FLAG

scpCTF{635835b528cdcc44c8eb000321368e83f66c2f3d9feb}