Side Channel Analysis Contest 2022			
8. ECDSA			
학교	국민대학교	이름	이현호

1. 주어진 코드의 특징

주어진 코드에서 ECC_Point_mul 함수가 $P=kG$ 과정을 담고 있다. 그때 사용된 Scalar Multiplication 연산 code는 [표 1]과 같다.

```
for (msb = numBits - 2; msb >= 0; msb--) {
        ECC_Point_dbl(&R[0], &R[0]);
        if (ECC_Field_getBit(scalar, msb))
        {
                ECC_Point_add(&R[0], &R[0], point);
        }
}
```

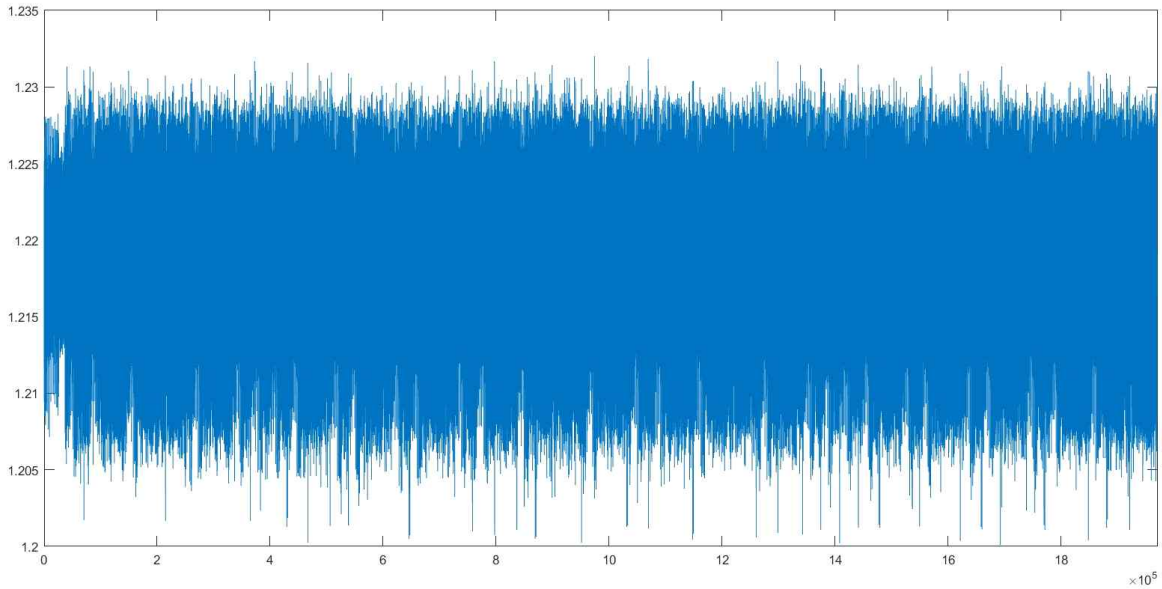
[표 1] Scalar Multiplication code

주어진 전체 코드를 통해 [표 1]에서 numBits는 변수 scalar의 비트 개수를 의미하고 ECC_Point_dbl는 ECC에서 Point doubling 연산을 의미하며 ECC_Point_add는 ECC에서 Point addition 연산을 의미하는 것을 알 수 있다. 이를 통해 [표 1]은 Left-to-Right Binary Method를 사용한 Scalar Multiplication 연산인 것을 확인할 수 있다.

즉, 변수 scalar의 특정 비트 위치의 값이 1일 때만 ECC_Point_add함수를 실행시키는 것을 알 수 있다. 또 한 Left-to-Right Binary Method는 scalar의 최상위 비트부터 최하위 비트까지의 값을 이용해 연산을 수행한다. 따라서 scalar의 최상위 비트는 항상 1로 시작할 것은 너무나 자명하다. (ex. 0b00100 = 0b100)

2. ECDSA 파형 분석

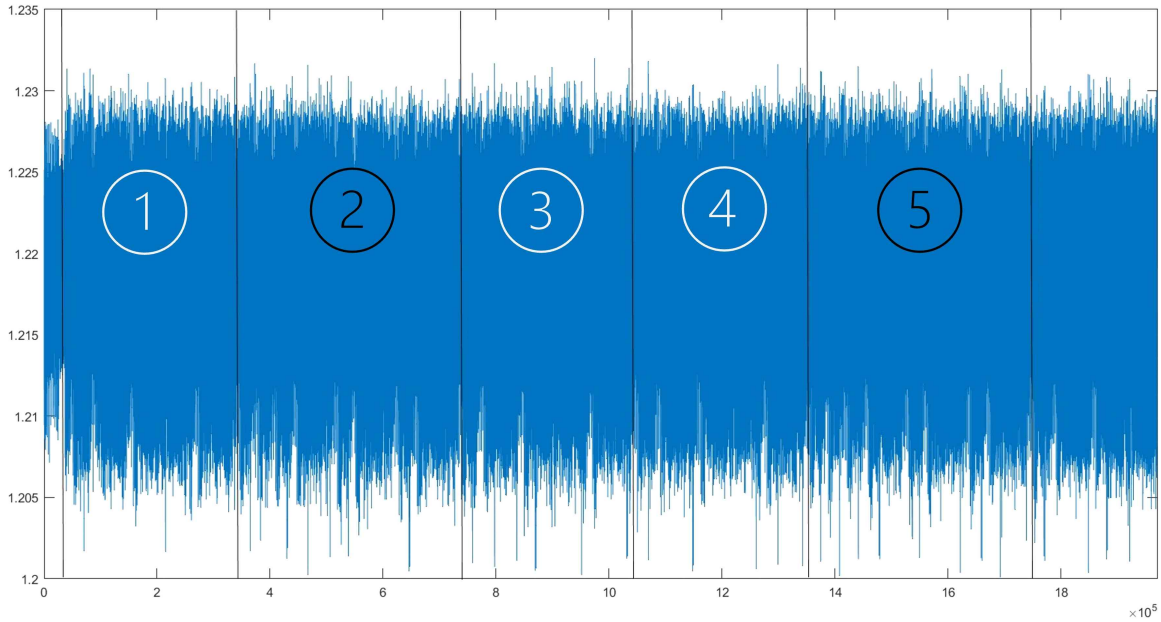
주어진 파형은 125,564,323포인트로 구성되어있다. k는 255비트 수라는 정보를 이용하여 파형을 한 구간의 포인트가 492,409인 255개의 구간으로 나눌 수 있다. 그 중 4개의 구간을 한 번에 나타낸 것은 [그림 1]과 같다.



[그림 1] 255개의 구간 중 4개의 구간

[표 1]의 연산에서 scalar의 특정 비트 위치 값에 상관없이 항상 먼저 Point doubling 연산이 일어난다. 그렇다면 파형에서도 항상 Point doubling 연산 전력 소비 파형 패턴이 존재할 것이며 특정 비트 위치 값이 1일 때만 Point addition 연산이 추가적으로 일어날 것이다. 즉, 특정 비트 위치 값이 1일 때 파형의 길이가 0일 때 보다 더 길것이며 시작은 항상 Point doubling 연산 전력 소비 파형 패턴이 나타날 것이라 예측할 수 있다. 위 정보를 바탕으로

나눈 [그림 1]에서의 같은 패턴의 파형 구간은 [그림 2] 와 같다.



[그림 2] 파형의 패턴에 따른 구간

[그림 2]의 1,3,4 구간의 패턴이 매우 비슷하며 2,5 구간의 패턴이 매우 비슷하다. 반복문에서 연산의 시작은 항상 Point doubling 연산이라는 사실에 의해 1구간이 Point doubling 연산이며 패턴이 비슷한 3,4구간도 Point doubling 연산이라 추측할 수 있다. 남은 2,5 구간은 Point addition 연산임을 알 수 있다. 그렇다면 1~2 구간이 바로 k의 특정 위치의 비트가 1일 때 나타나는 패턴이다. 비슷한 패턴을 가진 4~5 구간도 k의 특정 위치의 비트가 1라는 정보를 나타낸다. 이와 반대로 3구간은 k의 특정 위치의 비트가 0임을 나타낸다.

[그림 2]의 정보를 토대로 k의 특정 위치의 비트가 1일 때 [표 1]에서 Point doubling, Point addition 연산이 일어나는 전력파형의 길이는 약 700000포인트이며 k의 특정 위치의 비트가 0이라면 Point doubling 연산이 일어나는 전력파형의 길이는 약 300000포인트로 추측할 수 있다. 주어진 파형의 총 길이는 125564323포인트 이다.

k의 비트열에서 1의 갯수를 x, 0의 개수를 y라고 한다면 [수식 1]과 같은 연립 방정식이 성립한다.

$$\begin{cases} x+y=255 \\ 700000x+300000y=125564323 \end{cases}$$

[수식 1]

[수식 1]의 결과를 근사값으로 나타내면 $x=122,\ y=123$ 이다. $0 \le x,y$ 조건을 만족하므로 위의 추측정보가 어느 정도 신뢰성을 띄고 있음을 짐작할 수 있다.

3. 위의 추측정보를 바탕으로 분석한 k의 값은 다음과 같다.

$k=$ 0b1101111010011110110010000100001010001010111001001011001010001110100100101010011001000000111011001010011011010010111001101011111101101110010000000110110001111001011101110110010101010010010000010011000101100101011100000011000001110011010010010110110101110000

4. 위의 k 값을 바탕으로 복구한 Flag는 다음과 같다.

sImp1ep0weRAn@lySis_ISvErYGoOd!