Hw0x01 Writeup

XOR-revenge

查資料的過程中知道題目是 Galois LFSR，也會有對應的 characteristic polynomial 及 companion matrix，但問題是 output bits 之間會空轉 36 次。

看到這篇 writeup[1] 後我意識到可以用原本的 companion matrix M 做出 M^{37} 來跳過空轉的 state。

接著再看到這篇 writeup[2] 發現可以把某個時間點的 64-bit state 當成 64 個未知數，然後找出 64 個等式就能用高斯消去法得到 state。我選擇還原的時間點是產生加密 flag 用的最後一個 bit 前的 state $\vec{s}_0 = [s_0\ s_1\ s_2\ \cdots\ s_{63}]^T$，也就是下個 `getbit` 的結果是 s_0。用後面多給的 70 bits 的確可以找出 64 個等式。

$$M^{37i}\vec{s}_0 = \vec{s}_i,\ i = 1, \ldots, 64$$

M^{37i} 的第一個 row 跟 $\vec{s}_i$ 的第一個 element (i.e. 多給的第 i 個 bit) 可以跟 $\vec{s}_0$ 形成一個等式，共 64 個等式，可以求出 $\vec{s}_0$。

最後用 M^{-37} 反推回所有的 output bits。

---

lsb

題目是給出 $pt \bmod 3$ 後的結果，跟 $\bmod 2$ 時候的想法類似，把 pt 表示成一個 3 的多項式

$$\begin{aligned} pt &= c_k 3^k + c_{k-1} 3^{k-1} + \cdots + c_1 3 + c_0 \\ c_0 &= pt \bmod 3 \end{aligned}$$

如此一來，密文 ct 乘上 3^{-e} 解密後會得到 $3^{-1}pt$

$$\begin{aligned} 3^{-1}pt &= c_k 3^{k-1} + \cdots + c_2 3 + c_1 + 3^{-1}c_0 \\ c_1 &= [3^{-1}pt \bmod 3 - (3^{-1}c_0 \bmod n) \bmod 3] \bmod 3 \end{aligned}$$

$c_2, \ldots, c_k$ 以此類推。

由於不知道 k 是多少，但是知道會比 pt 表示成 2 的多項式的項數還少，而且多做的結果都是 0，我就讓他做滿 2048 次。

---

node

有個 elliptic curve $y^2 = x^3 - 3x + 2$，計算 $4 * (-3)^3 + 27 * 2^2 = 0$ 知道是個 singular curve，並可以寫成

$$y^2 = (x-1)^2(x+2)$$

這個 curve 是個 node，所以有個 mapping ϕ 可以把 elliptic curve 上的點映射到 $\mathbb{F}_p^*$

$$\phi(P(x,y)) = \frac{y + \sqrt{3}(x-1)}{y - \sqrt{3}(x-1)}$$

原本 curve 上的加法 f 倍的 G，經過 ϕ 映射後

$$\begin{aligned} fG &= F \\ \phi(fG) &= \phi(G)^f = \phi(F) \end{aligned}$$

變成 discrete logarithm problem，加上 $p - 1$ 容易分解，可以用 Pohlig-Hellman 來解。

DH

這題的關鍵是選一個特定的 g 來建立一個很小的 subgroup，結果我花了好幾天卡在別的方向，直到看到這篇 "OT or NOT OT" 的 writeup[3] 才恍然大悟。

重複嘗試直到選到的 p 滿足 $p \equiv 1 \pmod 4$，接著隨便選一個 $x \in \mathbb{F}_p$，算出 $g = x^{(p-1)/4}$，這樣 g 的 order 為 4。

題目裡還有檢查避免出現 1 或 $p-1$，所以要多嘗試幾次，直到 `a` 和 `b` 都為奇數。

最後密文分別乘上 g 生成的 subgroup 的 4 個 modular inverse，其中一個可以還原 flag。

AES

照著老師上課講解的方法做 correlation power analysis，我選擇用 plaintext 針對第一個 round 做分析。

題目有給 50 組 plaintext 和 trace，每個 trace 有 1806 個 sample point，照投影片上的 notation 來看，$D = 50, T = 18096, K = 256$。對這 50 組資料一次一個 byte 處理並反推回 key。

1. 每一個 plaintext byte 配一個 key hypothesis 依序做 AddRoundKey 和 SubBytes 會得到一個 $D \times K$ 的矩陣。
2. 對每個 element 計算 Hamming weight。
3. 計算這些 Hamming weight 和 trace 的 correlation
4. 找出 correlation matrix 最大值對應哪個 key hypothesis，就可能是 key 的一部份。
5. 所有可能的 key 拼起來就是原來的 key。

---

1. CakeCTF 2021 WriteUps | 廢文集中區↩
2. CTF | 2020 CISCN初赛 Z3&LFSR WriteUp | MiaoTony's小窝↩
3. zer0pts CTF 2021 Crypto Writeups :: rkm0959↩