CTF HW0x0A

Yet Another Oracle

> CTF{S3Nd me A FriEND ReQu3St xD - aHR0cHM6Ly93d3cuZmFjZWJvb2suY29tL3Byb2ZpbGUucGhwP2lkPTEwMDAwMTI2OTIxMTk2Mw==}

就LSB oracle拿去擴充就可以了。

原本lab-2是給一個bit做二分搜，這次作業只要拿最後4個bit做16分搜就可以做出答案了。

或者，我們可以用接關的方式來做：每次做一次1-bit LSB oracle attack的時候，作到不能再做的時候，其實我們會得到一個lower bound和upper bound。

那麼再重新做一次LSB oracle attack的時候，如果有以下這種狀況發生：

old lower bound < old upper bound < now mid

now mid < old lower bound < old upper bound

我們就可以直接知道它傳回來的lower bit是0還是1，但是我們卻省下了一次次數。

因此接個4次就可以把所有key leak出來。

- 因為分搜有一定風險，所以code要多跑幾次才會有flag。

Can you Decrypt?

> CTF{f01lOW my iNsT4gRAm xD - aHR0cHM6Ly93d3cuaW5zdGFncmFtLmNvbS9vYWxpZW5vCg==}

- Step1: 先解p，因為p是很多個小數相乘，這裡直接用pollard p-1 attack就可以幹出來了。
- Step 2: 接下來我們知道 q1 ~= 3q2，因此可以利用FermatSieve演算法算出估計的值，進而拿到q1 和 q2。
- Step 3: 有 p/q1/q2 ，就可以算出phi。
- Step 4: 接下來我們就可以算出d，之後c^d次方會發現還會多一個四次方。
- Step 5: 用模開方根演算法連續做兩次，得到16種結果一一比對就可找到答案。