i春秋 Crypto Writeup

原创

WLNY 于 2020-08-09 18:01:58 发布 348 收藏

RSA?

题目：

N=0x180be86dc898a3c3a710e52b31de460f8f350610bf63e6b2203c08fddad44601d96eb454a34dab7684589bc32b19eb27cfff8c07179e349ddb62898ae896f8c681796052ae1598bd41f35491175c9b60ae2260d0d4ebac05b4b6f2677a7609c2fe6194fe7b63841cec632e3a2f55d0cb09df08eacea34394ad473577dea5131552b0b30efac31c59087bfe603d2b13bed7d14967bfd489157aa01b14b4e1bd08d9b92ec0c319aeb8fedd535c56770aac95247d116d59cae2f99c3b51f43093fd39c10f93830c1ece75ee37e5fcdc5b174052eccadcadeda2f1b3a4a87184041d5c1a6a0b2eeaa3c3a1227bc27e130e67ac397b375ffe7c873e9b1c649812edcd

e=0x1

c=0x4963654354467b66616c6c735f61706172745f736f5f656173696c795f616e645f72656173736d626c65645f736f5f63727564656c797d

思路：

一开始想着分解N，在factordb上没有分解成功,观察到e = 1,由RSA原理我们知道 c = m^e mod N= m mod N，即 **m = k * N +c**

编写脚本得到k = 0时求出flag。

看我回旋13踢

题目：

synt{5pq1004q-86n5-46q8-o720-oro5on0417r1}

思路：

盲猜是凯撒密码，密钥是13，得到flag。

RSA

题目：

N=0x1564aade6f1b9f169dcc94c9787411984cd3878bcd6236c5ce00b4aad6ca7cb0ca8a0334d9fe0726f8b057c4412cfbff75967a91a370a1c1bd185212d46b581676cf750c05bbd349d3586e78b33477a9254f6155576573911d2356931b98fe4fec387da3e9680053e95a4709934289dc0bc5cdc2aa97ce62a6ca6ba25fca6ae38c0b9b55c16be0982b596ef929b7c71da3783c1f20557e4803de7d2a91b5a6e85df64249f48b4cf32aec01c12d3e88e014579982ecd046042af370045f09678c9029f8fc38ebaea564c29115e19c7030f245ebb2130cbf9dc1c340e2cf17a625376ca52ad8163cfb2e33b6ecaf55353bc1ff19f8f4dc7551dc5ba36235af9758b

e=0x10001

phi=0x1564aade6f1b9f169dcc94c9787411984cd3878bcd6236c5ce00b4aad6ca7cb0ca8a0334d9fe0726f8b057c4412cfbff75967a91a370a1c1bd185212d46b581676cf750c05bbd349d3586e78b33477a9254f6155576573911d2356931b98fe4fec387da3e9680053e95a4709934289dc0bc5cdc2aa97ce62a6ca6ba25fca6ae366e86eed95d330ffad22705d24e20f9806ce501dda9768d860c8da465370fc70757227e729b9171b9402ead8275bf55d42000d51e16133fec3ba7393b1ced5024ab3e86b79b95ad061828861ebb71d35309559a179c6be8697f8a4f314c9e94c37cbbb46cef5879131958333897532fea4c4ecd24234d4260f54c4e37cb2db1a0

d=0x12314d6d6327261ee18a7c6ce8562c304c05069bc8c8e0b34e0023a3b48cf5849278d3493aa86004b02fa6336b098a3330180b9b9655cdf927896b22402a18fae186828efac14368e0a5af2c4d992cb956d52e7c9899d9b16a0a07318aa28c8202ebf74c50ccf49a6733327dde111393611f915f1e1b82933a2ba164aff93ef4ab2ab64aacc2b0447d437032858f089bcc0ddeebc45c45f8dc357209a423cd49055752bfae278c93134777d6e181be22d4619ef226abb6bfcc4adec696cac131f5bd10c574fa3f543dd7f78aee1d0665992f28cdbcf55a48b32beb7a1c0fa8a9fc38f0c5c271e21b83031653d96d25348f8237b28642ceb69f0b0374413308481

c=0x126c24e146ae36d203bef21fcd88fdeefff50375434f64052c5473ed2d5d2e7ac376707d76601840c6aa9af27df6845733b9e53982a8f8119c455c9c3d5df1488721194a8392b8a97ce6e783e4ca3b715918041465bb2132a1d22f5ae29dd2526093aa505fcb689d8df5780fa1748ea4d632caed82ca923758eb60c3947d2261c17f3a19d276c2054b6bf87dcd0c46acf79bff2947e1294a6131a7d8c786bed4a1c0b92a4dd457e54df577fb625ee394ea92b992a2c22e3603bf4568b53cceb451e5daca52c4e7bea7f20dd9075ccfd0af97f931c0703ba8d1a7e00bb010437bb4397ae802750875ae19297a7d8e1a0a367a2d6d9dd03a47d404b36d7defe8469

思路：

正经RSA题目，密钥都知道，用libnum库编写脚本直接带走

classical

题目：

Ld hcrakewcfaxr, f hofjjlhfo hlaxuc lj f krau ev hlaxuc kxfk zfj tjui xljkeclhfoor gtk dez xfj vfooud, vec kxu pejk afck, ldke iljtju. Ld hedkcfjk ke peiucd hcrakewcfaxlh foweclkxpj, pejk hofjjlhfo hlaxucj hfd gu acfhklhfoor hepatkui fdi jeoyui gr xfdi. Xezuyuc, OrmkO3vydJCoe2qyNLmcN2qlpJXnM3SxM2Xke3q9 kxur fcu foje tjtfoor yucr jlpaou ke gcufn zlkx peiucd kuhxdeoewr. Kxu kucp ldhotiuj kxu jlpaou jrjkupj tjui jldhu Wcuun fdi Cepfd klpuj, kxu uofgecfku Cudfljjfdhu hlaxucj, Zecoi Zfc LL hcrakewcfaxr jthx fj kxu Udlwpf pfhxldu fdi guredi. F btlhn gcezd veq mtpa eyuc kxu ofsr iew.

思路：

这一大串英文乱序一看就要字频分析，用 quipqiup爆破一遍，得到一串密码

LyjtL3fvnSRlo2xvKljrK2ximSHkJ3ZhJ2Hto3x9

有点像base64，试着解码，解不出来，尝试凯撒移位后继续解码，成功

RSA

题目：

n is

96680893262749719063585923605496034909946397522735056426538437328033669985338725407066288126593756516300075860615430875794403057183717504851457447306140156633083633464717665528261926859256017272652664307449953412987821740904604553365689705011743849635723157599918552767507100280395180063522002901593200746511781873994890375020083085611566869100770683695224484271941945294625927525177329833816238993051883827270490888701647400705139719458839603911121670886621461477962756695933517067605502585093263105364157656616569412142054608104328580678323929679979565519112196637759017578061894491053281698814305675705405267996853890146089357120490439497571408105545524052389565330531551774572933411454975669533417114287608047710507040954477798160215276215461073854016379616429522281024330905150309086667463444035922619253072463547705157651517986446117491197566716259728676907938066078264795294480859631047697393915618747207695293572824906113748188758910397359108287298864195827028516965080379239555636330405629007780145398082209758357430968293569726020486275692386555639768669685423956454140718570994010780653677316026376448344385942572695314296414821620996843758704461761351805877928716785334936453371645867606673421687756618151460769388237553316850291008885829563431507024437740955656763713973630808218636900322777193640732178355779562427916216230520043644690397638594867789766546629085276987756216748714238530802734163981640105508182049700201890889620286034239102908258162198730553309738665218384965706595206243398838764099038362326440552514400350028653126267431590053700184504322536314835976677103389968011107618167279707741058474750958193204554080177773854887274759789996536695082750552943248377982115815292889994783719639155566616548644187818328800875356110899571596192047292784487756985594050514884353099887811372283042780792667932424114118223890356768204241014534555188944215889515787579899090371510578268208388646166130706358344769616882868712695614795588649338380551355760417902905098167875505494560786635319579365410840393924272386165191915236992390400296687399481182639108031814626041697849937718254068440979035725749081620313849936963449089755322776356355398124689167761344639013447783214317524899216164169801119596879210520184797608232278662339024247022674068582221814026318202422622869215938055766159163307209194507733419198786026244838512359945964722856213736917806907280449804946313623385633781738597799014557104223179533299552398817489543281987283217002969084 8

思路：

整数在factordb分解成功，直接编写脚本得到flag

```python
# -*- coding: UTF-8 -*-
import libnum
p = 3109355130292288099988302080366553661627214702287742874531483086751935101324891424488010109436581599805011541530843961006670013916437627498065000515026794985367165323349178428949398894686939609373096632565924979654587808011920628351234298085447573409710897567077883600382278940549894137479801675368937799235512277440178093018559845824089436224619424862391138228416967759586450147530819464414060227296169923028299302050766893998020507923922192423043023031807699150761996033014474530702253802487844445871758744660155954629202624531890729358460932011537463223527079563393375535092853759824221421667449640962592879745047352070795633933755350928537598242214216674496409625928997877221
q = 3109355130292288099988302080366553661627214702287742874531483086751935101324891424488010109436581599805011541530843961006670013916437627498065000515026794985367165323349178428949398894686939609373096632565924979654587808011920628351234298085447573409710897567077883600382278940549894137479801675368937799235512277440178093018559845824089436224619424862391138228416967759586450147530819464414060227296169923028299302050766893998020507923922192423043023031807699150761996033014474530702253802487844445871758744660155954629202624531890729358460932011537463223527079563393375535092853759824221421667449640962592899787722190
N = p * q
phi = (p-1)*(q-1)
e = 65537
d = libnum.invmod(e,phi)
c = 16850291008885829563431507024437740955656763713973630808218636900322777193640732178355779562427916216230520043644690397638594867789766546629085276987756216748714238530802734163981640105508182049700201890889620286034239102908258162198730553309738665218384965706595206243398838764099038362326440552514400350028653126267431590053700184504322536314835976677103389968011107618167279707741058474750958193204554080177773854887274759789996536695082750552943248377982115815292889994783719639155566616548644187818328800875356110899571596192047292784487756985594050514884353099887811372283042780792667932424114118223890356768204241014534555188944215889515787579899090371510578268208388646166130706358344769616882868712695614795588649338380551355760417902905098167875505494560786635319579365410840393924272386165191915236992390400296687399481182639108031814626041697849937718254068440979035725749081620313849936963449089755322776356355398124689167761344639013447783214317524899216164169801119596879210520184797608232278662339024247022674068582221814026318202422622869215938055766159163307209194507733419198786026244838512359945964722856213736917806907280449804946313623385633781738597799014557104223179533299552398817489543281987283217002969084848
m = pow(c,d,N)
print (libnum.n2s(m))
```