re.81 explorer的奇怪番外7

.apk放进JEB

1.在Bytecode(dex)中找到”checkflag”

看到关键字符串”hctf{}”

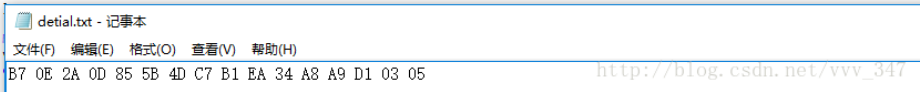
```
String v6 = checkFlag.this.editText.getText().toString();
try {
    MessageDigest v4 = MessageDigest.getInstance("MD5");
    v4.update(v6.getBytes());
    if(!Arrays.equals(v4.digest(), new byte[]{-73, 14, 42, 13, -123, 91, 77, -57, -79, -22, 52, -88, -87, -47, 3, 5})) {
        return;
    }

    MessageDigest v7 = MessageDigest.getInstance("sha-256");
    v7.update(v6.getBytes());
    checkFlag.this.textView.setText("hctf{" + checkFlag.bytes2Hex(v7.digest()) + "}");
}
```


可以知道: 用户输入v6, v4更新v6摘要 生成MD5值, v7更新v6摘要 生成SHA256

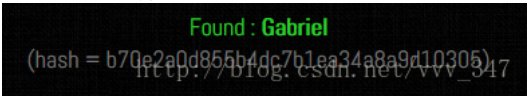
2.我们需要让MD5值(v4) 等于 字符数组

```
v4.update(v6.getBytes());
if(!Arrays.equals(v4.digest(), new byte[]{-73, 14, 42, 13, -123, 91, 77, -57, -79, -22, 52, -88, -87, -47, 3, 5})) {
    return;
```



得到MD5值

3.将其放入 decrypt online 得到字符串”Gabriel”

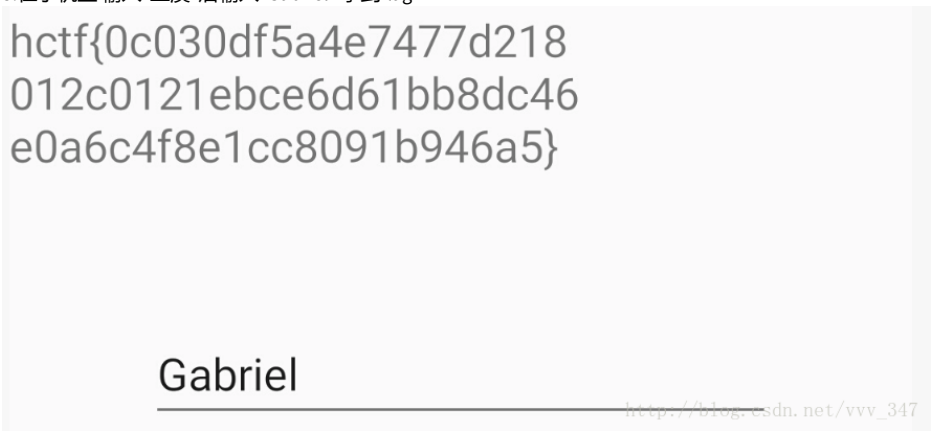


4. 在Bytecode(dex)中找到”ChooseAreaActivity”

```
if(v1.getCity().equals("兰溪")) {
    ChooseAreaActivity.this.startActivity(new Intent(ChooseAreaActivity.this, checkFlag.class));
}
```

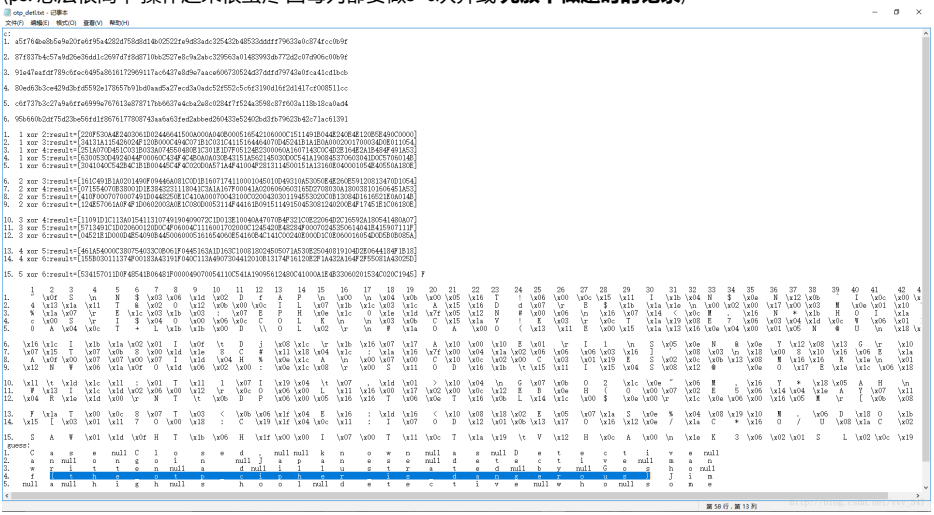

发现关键地点:”兰溪”

5.在手机上 输入”兰溪”后输入”Gabriel” 拿到flag



CRYPTO.85 进击的 Crypto [3]

针对一次性密码本 利用”多次使用相同密码本处理不同密文后不安全”
c1 xor c2 == m1 xor c2 这样可将密钥消去
(ps: 想法很简单 操作起来很蛋疼 因每列都要做5~6次异或 **先放下做题时的记录**)



1.分析一下 可以知道是6组86字节的密文 猜测6条密文使用相同密码本加密
2.做 6 * 5 / 2 == 15次异或
3. 根据15次异或结果 先转成字符 再***猜测***单词
tips: 小写字母 xor 空格 == 大写字母 根据这个能推出一些列
猜后拿到flag(见图)

crypto.86 进击的 Crypto [4]

先大致看下 6个print对应6个被注释的字符串
我们需要填充 k,x 来满足字符串输出正确
1.先求k 因为只拿r跑不出来k 使用s来跑
我们可以利用两组数据的作差 将”x * r”消去

```
s2 - s1 = (invert(k, q) * (sha2-sha1)) mod q
```

两边同乘 k 因为invert(k*q) 是逆元
那么invert(k*q) * k mod q =1
化简 k = ((sh2-sh1)/(s2-s1)) mod q
得到k = 20956958366294439995880747242168065 3
2. 再求x (这里卡了我比较久 关于数论方面的请dalao们带着自己理解来看)
取模的逆运算

```
s1 = (invert(k, q) * (SHA1(data) + x * r)) % q
invert(k, q) * (SHA1(data) + x * r) = q*n +s1
x * r = (q*n + s1)*k - SHA1(data)
x * r * invert(r,q) mod q= ((q*n + s1)*k - SHA1(data)) * invert(r,q) mod q
x = (s1*k - SHA1(data)) * invert(r,q) mod q
```

得到x = 914039385178649432938708047621093551813459555435
3.

```
data3 = 'getflag'
getflag(data3)
```

得到flag: hctf{881691912314398184476813935100212817302692520 95}

crypto.88 进击的 Crypto [5]

rsa 低加密指数广播攻击 参考资料:http://codezen.fr/2014/01/16/hackyou-2014-crypto-400-cryptonet
这里只贴上我改后的代码了

```
import gmpy

def my_parse_number(number):
    string = "%x" % number
    #if len(string) != 64:
    #    return ""
    erg = []
    while string != '':
        erg = erg + [chr(int(string[:2], 16))]
        string = string[2:]
    return ''.join(erg)

def extended_gcd(a, b):
    x,y = 0, 1
    lastx, lasty = 1, 0

    while b:
        a, (q, b) = b, divmod(a,b)
        x, lastx = lastx-q*x, x
        y, lasty = lasty-q*y, y

    return (lastx, lasty, a)


def chinese_remainder_theorem(items):
  N = 1
  for a, n in items:
    N *= n
```

```
  result = 0
  for a, n in items:
    m = N/n
    r, s, d = extended_gcd(n, m)
    if d != 1:
      raise "Input not pairwise co-prime"
    result += a*s*m

  return result % N, N


e = 10
data =
[(46751826055497113476532995147778262385590402005277478108469019914331270257233
61807211672208052862870258047285285949212515664278410499603829663266213968511493
97476666740718252255366652670490446193403784385316159836964063058419457563965793
37179482626554992950304313466836005502727652236887919362093745788148650931171190
05722483187734650127796510808637890189952840971330768913946884525594627652458745
93053221923825891160874411200179813639480600472647289083848188102671660574381 2
67435095525800379459074825852392946044698602079444547807655963514518190909378666
17716473870604463315696183050495143502762614767485881979217577416 ,
17551188754807399016342420221734945766749930201727412345251590531404061480740932
99519906533298771918319719944833643585101554027215544148674602731510782134096 9
10562345157587258017097836865059388027407602552045395637853976622843042914211 79
94616985318963125739076826635459215114946159348678398268099389525414431611851774
38893140857143904309727832232709856889899525777036311722547259008448912095839 7
18913095846278453263745348218276350579172064142268674961621652172401210837868 01
77131928438795893866425040489551258902610047574579899692834984999209528110849 34
20425298255940618942383715919103545843051361543553651912154001 49) ,
(45860339300158149755534876143213412903580729605395648495897585938015878234143 9
43518620740933432554796353031990785619052607936458699426221361646157768228672 75
17288881298907101668765802300583877618143747020452833544441385256189560337795 59
59308318399246963213964044984235974352824060625970587119586612774322385216810 26
48194264586418521660098991066550367559133222832197397410206568120343484040920 18
34027702684411427269917032301405008728971540398967191876771261341434632022274 53
28800096615947138667139053339483278319521201983885423491905171167675501563556 56
38908217568924749995496356504869440103465546416033432384905726 89 ,
21840437284422601584177601857355845296420300157767339109572377640408362726674 56
12462104004007604741871212468937124801093268936141627794707684152829007138009 83
81557060225006867369504474993253268985131077008755252462085762344201942804448 27
87965428075984951982104920901562456426672111590909100736209153905853053125 4772
02769740504046638475470903627807290778295946849959794573264149474490832194616 17
30264339290452273381290314130133222735768951788080184551057951888747292808371 88
26890369016944549160461428710615747001273521253638766267143341819170249648385 34
11272637074176429254648097889757153329386664173166959413474775 77) ,
(12012575342366442210994368605032582129674485327006093552902983877957202172783 9
38071684841031902396156899396879136388606996807602896316795628987877357547213 45
03609537429477817005495064692265649759273617485965208785426516689631310212698 51
92817468100734777151956991469098074494196508676228563008278101378187951168135 31
19117280354102271439057204066405072328422057721895983540492150402309552609035 00
82723661766573841740812297987390995585390830744895003590620101643583000871244 47
03792284470987043021801967391430574293992788883859961652030001191302279206039 22
60618046807788302643742874847024153198113792628046678578753632923,
30015914133986758133105015082922460910471726819000479872816812806794140887209 29
43939630632733778912030698647114667762001081736724287792933083204601164930405 72
82691502065429392924184368672829638740011006209907457300964556352080530189996 23
```

```
44680821396200256042478539540675850427377958553324581371362112652752444824919601767821622038488365501031636558236126052654241662743070651213369566898998986344412618116550537893845887115476859399277211164044299143290814329909133212969649572606911487047105816817745629049240808132749848099400089686295153048595186294732840183326518315879164025393862424216302505370306881626757517617) ,(424900591132489863045872349115157681019861907833297291142016664512725859851995044850764090437934281748642626205488919936938525781252475976124558897365157821129134432661931755937736031950772772953508314329333579958819697624194921803783374783954529503025911527966806588656041787720405468449504718111707560346867229226883127373595696406762683797521853984066764188562142660794195030633019889718768375468737961184427847157279181639795063438890723331211792063480798973401113271833433485059211165163132313093343800700898348978260182732139616810466849333262655891183872199817314850899477147322405082747118647810689694941383 ,(180097184358254456493726296348677722470351382294931083627136309476803383542277355720708823903786324403651631170942334135201070705819082242392109691720944027609240553345842596782765059194181916237629982497242483793027866216890491075007603483039401128406649262313456463259641332815507654547196281616004751437925673099473301307478601255575470518436208996292176369180029298464630978415394460140195798303473560840804885619174403568484658129372468090181685826351916196220410790124505318112371681056343821086341358808870935256112631869252130279133374547069197549235937145099980620734506797209055694893043986252551438015332777) ,(185571098538984059749247695501053456737030674575378136072521514699331407607983785742444395593721004289714000703511731403488236527558256014332688323635758961373642880699843141513461372061155183095973892688751505496963768132777785145915326157942946370446266839313675101816123836819144217022615922332654559500232524076806040410466296677032071331468536189891875977491326505140385842804177911344265968483694176260395337408275545521177275015658410650778353989518267478553935527314974389832742015997401897436309491746107221186590196309646699014386900280702137108333356225735264035887370928692739695816076375225727570152375060 ,(27090736422393991189249636552945539144039087911497773160371557650625344533254580764344628540515132884576739746597729079146155130899009238110101654711303800340566538298798432929136509923129089904647023409146264405964139002638684510681372603717417957076868564057020972760021529533084636175431497373175373439731293294743322573259752407656360572903799827280347295307991289986724431807314456435532652023007868110674667089564345493971442366101821646902002142914233623830177594879478477690605860139502646384207075554779319247065320407822282776895082374706165510690027657154768045195356231437691359289642773047309136001351911291) ,(19221531342713801219971420455098666365124810169512711030444063942333694562364114172361552139176582476168134219937393468391346535097965763785916484012657401095066804857900941482772687159471483908107669892419626059680168762119954068288719303591498504050604628627299671160225276355636519961578118413913001058744164531470610279168723402009469959377794138715204862239973561494796213624769358906726186769711689962808937828553399403173762034548974335960107700638910231658795639167906700770120741974418495196827830462542130080965552102416180517600397897065113788279096066792533807104187868678848857780735334749865813019681909 ,(22277916445389799876692954866506052125036892596099795492064670519272419621528759837318253665292779127861537748967309681231938516330289846519214661138299205082421968922350230121526530080908053297879027158545040193774647350499415863211113776785399848472712293535289113810322398103983587423306282629848367328743089145940536913390664697321526306054579656600439140061667264516070003507841618352122567379038304294976725469305215682426898731699093794930417366769249256999177339843020364367769712776225743916112170787273891225926246407201080202593858063362565931942590349178361085396799890148505959437244983447934838792051793) ,(20438667185329273014541270175218359454017271737603524882205170284988617772628070365661651892857305693750834062342129215886540984212900882695513521018306884299036008806910466079023965667750056941483156056164377131154242235947101866778448
```

```
67273218443350479253279231960061845711898262461185970057002138523262952260834191169438764964758526877361442277664448596129451496415695776781879266732057920583336999766633938582656240607973759756021032489262381190762825081392613689817792764844409877116240223852376311404164963595040347176139494461319495285348608562589901671620063620688524249523700500144604558899272202564535295466 ,
208510052547049339583548175529751905883839628438271222269049640483180532430498222770497155057357620515305929405140076871388825513699067117871334327924784474652414094491456254707668672805456733137108778274910109662858717476384013153386169867823706418816152384346676879364009556296426297489878399960115504081561623172011397464531488745586030347569022970668355927485620141000330507366503595124976332065087862042105884086417865415028745257336325154309073340438903724184386937954371980449964060056216051812575335078292203435075056740907356476506842356289315995804144395122093369406451027596426472360089455862342181990450 1) ,
(90184064520411178679272046746492209291605099426042905723036441834512379449124702363673082663190793492896258734476472004745987558696429992131427588258095224748108176097511875333469026672250324526941871622626830564457754559639942372022961048123863746362701884335219275034155079634772409132708797941411705683930253678910039210900159939449462057236013491496012567274337161551218834202714982659672342590685351725014456437526918609974806361016993407402655057670919206682180180029766321112564306556190612472920256547408890770141792586470846982297331600713063900206036362107363631983024072891952869539805202850325286144929260 ,
217456807181940378616945698636780828537972443803102001764776439436449724638713606690705846828077038706848309480231588897802197168293536656005647912579120820439051220480045938031933751782699429732082492761027696710387524894384007315353672573562070987377110969536792313007209035021921444765198118566464008222399662197464557294093757717970648255529584822970977979115998082428757993424388996363287637669081546587389696147073778521478439536141783054165948198468124268283006894970463896659828172093157843131086177530520153269370477125135693229631025001220385057702329174732600574089811254696075731919261340437194378681562730) ,
(17978671919734595201524186618868659656036765546673883938819333911564948587885887649580260024407841331246281393868209076159996160495960942301090418772074853970717285335254037020899190292900203677538150336517019683862637968571205072887517222623244787536816674375115977561164262940853489068410965264848486887671069724935189535511151128103749734771224124290856369970727284303277310747736879041910990727109635145752624047197512229648365533715013000081281308123918165338441914929297032099189207335234064923738740649808773470816209447609236866579435096892428925919289328933703820150406631017328475704480244960066600782150955 3 ,
232574830423317810313200040663959730985398818704330344981806282921648254768456475961228897563969872207872634787786471990335232488610794879912560444736446515960559630792146394039814096408486541588067643811077716428015000310006227815563853161604153799667073262886506475792902392829571472354691875108497532721686076971371510238161060553858839091026451266521753830084143245517459861325880906431205328541736948784727909518259034781432825317588016829248046749554245422672923121452470552323438808591118050034108325754535134236727740460190849252647561826203675666604144150602478060519434077780238996018053283187868945809604682 1) ,
(85970678808124561236695949786601350586688758349242012799241627610201375826623677045049885766149349660623231916669829431136561867838942450352670517561783314908812646297746356201796780009477719288545662392134985860786708701279246964590412958484740738793365013254371945638454146993605645711366834924262927667991211550412761626734277009997610555926092136031585766372120037710039967227431705433428850112896928454935984139656889493256705197434130714790573826022220559025936141594271860177979922194903089301168500845530334613165094578344364208219160942060102062293702736226814131763471872529500874288682959655287728324093162763700462232733803098815790618032466782991675135897764083276532864571869638548209178151319747206605210170413175115862723965742547399444114338561310552820021527511046626328995296240066429360213385680947529597032276730956327399931992615060
```

```
48399522508592521685688896842833243324070987594588134776515978198364901455727292006058624015674914131447232032058875410956114939938460192876157612457774033922125553271809409381181134668696682847583314698759337802814876161751333701826176655372126746717533228292189928645064111959824895679517476699556313948724818860906006774808944802984477517788391260149504822396276593451707118257)]


print "Please wait, performing CRT"
x, n = chinese_remainder_theorem(data)
realnum = gmpy.mpz(x).root(e)[0].digits()
print my_parse_number(int(realnum))
```

最后 “感谢各位dalao出的题和比赛
“