

CTF for Starters 資料

自己紹介

- 名前： 西永俊文
- 所属： 金沢大学 情報セキュリティ研究室 修士 1 年
- Twitter: @tnishinaga
- やったこと：
 - セキュリティキャンプ 2013 参加 ,2014-2015 チューター
 - 技術書「BareMetal で遊ぶ Raspberry Pi」執筆

もくじ

- フォレンジック問題を解こう
- 暗号問題を解こう (時間があれば)
 - シーザー暗号
 - ROT13
 - base64 問題

フォレンジック問題
を解こう

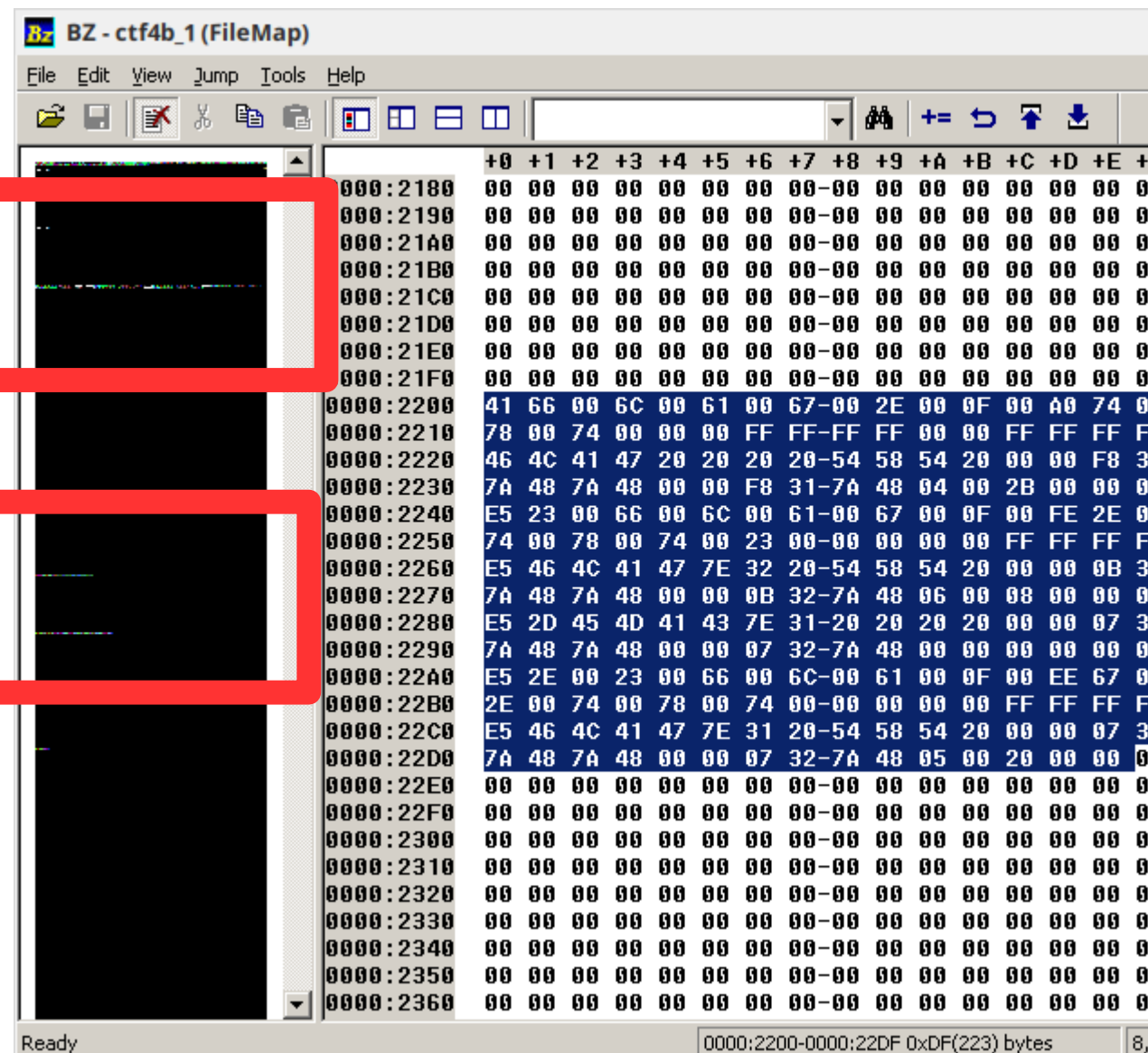
(初級編)

フォレンジック問題

- 謎のファイルに含まれた flag を解析する問題
- バイナリエディタを使って解く
 - Starling - <http://www.vector.co.jp/soft/win95/util/se079072.html>
 - Bz - <http://www.forest.impress.co.jp/library/software/binaryeditbz/>
- 目 grep が大切
 - <http://www.slideshare.net/murachue/grep-8132856>

grep

なんかあるところ



フォレンジック問題の解き方

1. 目 grep で怪しいところを見つける
2. ドラッグで選択してコピー
3. 新しいファイルをバイナリエディタで開いてペースト
4. 保存して開く

例題

- 問題 1

- mondai/ctf4s_1

- 問題 2

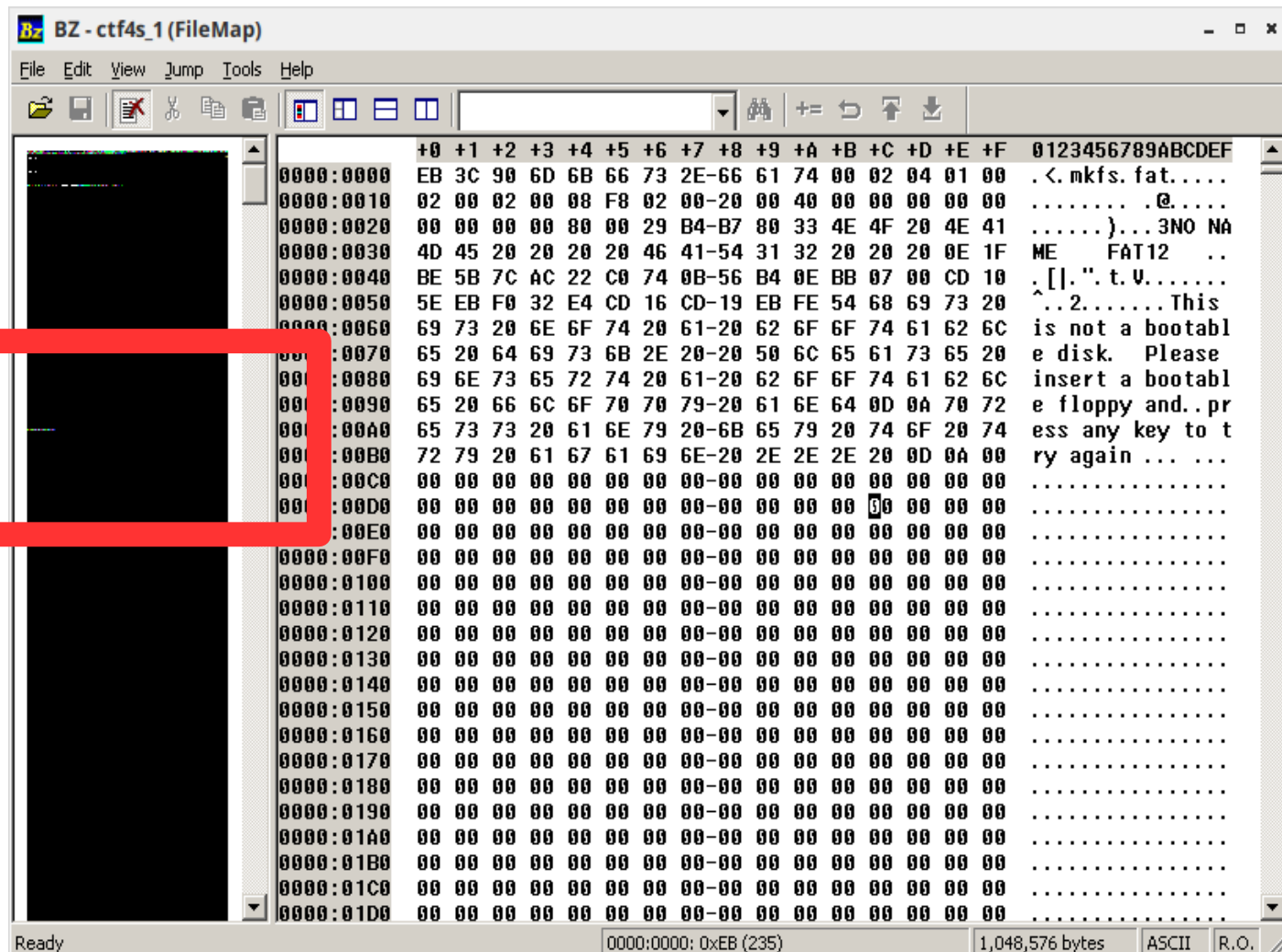
- mondai/ctf4s_2

問題 1 の答え

● 解き方 1

- ・ バイナリエディタで怪しいところをチェック

あやしい
ところを
クリック

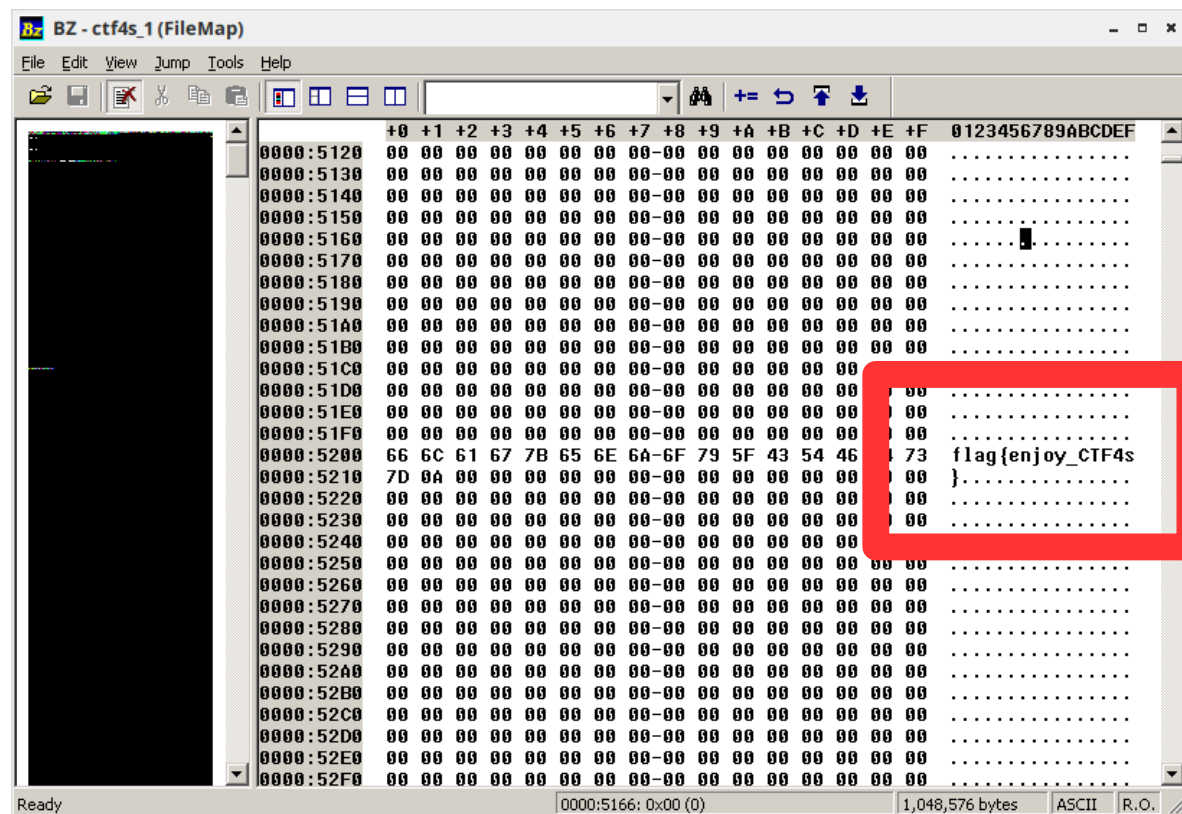


```
0000:0000 EB 3C 90 6D 6B 66 73 2E-66 61 74 00 02 04 01 00 .<.mkfs. fat....
0000:0010 02 00 02 00 08 F8 02 00-20 00 40 00 00 00 00 00 ..... @.....
0000:0020 00 00 00 00 00 00 29 B4-B7 00 33 4E 4F 20 4E 41 .....}...3NO NA
0000:0030 4D 45 20 20 20 20 46 41-54 31 32 20 20 20 0E 1F ME FAT12 ..
0000:0040 BE 5B 7C AC 22 C0 74 0B-56 B4 0E BB 07 00 CD 10 .[|. ". t.v.....
0000:0050 5E EB F0 32 E4 CD 16 CD-19 EB FE 54 68 69 73 20 ^..2.....This
0000:0060 69 73 20 6E 6F 74 20 61-20 62 6F 6F 74 61 62 6C is not a bootabl
0000:0070 65 20 64 69 73 6B 2E 20-20 50 6C 65 61 73 65 20 e disk. Please
0000:0080 69 6E 73 65 72 74 20 61-20 62 6F 6F 74 61 62 6C insert a bootabl
0000:0090 65 20 66 6C 6F 70 70 79-20 61 6E 64 0D 0A 70 72 e floppy and..pr
0000:00A0 65 73 73 20 61 6E 79 20-6B 65 79 20 74 6F 20 74 ess any key to t
0000:00B0 72 79 20 61 67 61 69 6E-20 2E 2E 2E 20 0D 0A 00 ry again ... ..
0000:00C0 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
0000:00D0 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
0000:00E0 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
0000:00F0 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
0000:0100 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
0000:0110 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
0000:0120 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
0000:0130 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
0000:0140 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
0000:0150 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
0000:0160 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
0000:0170 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
0000:0180 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
0000:0190 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
0000:01A0 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
0000:01B0 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
0000:01C0 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
0000:01D0 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 .....
```

問題 1 の答え

●解き方 1

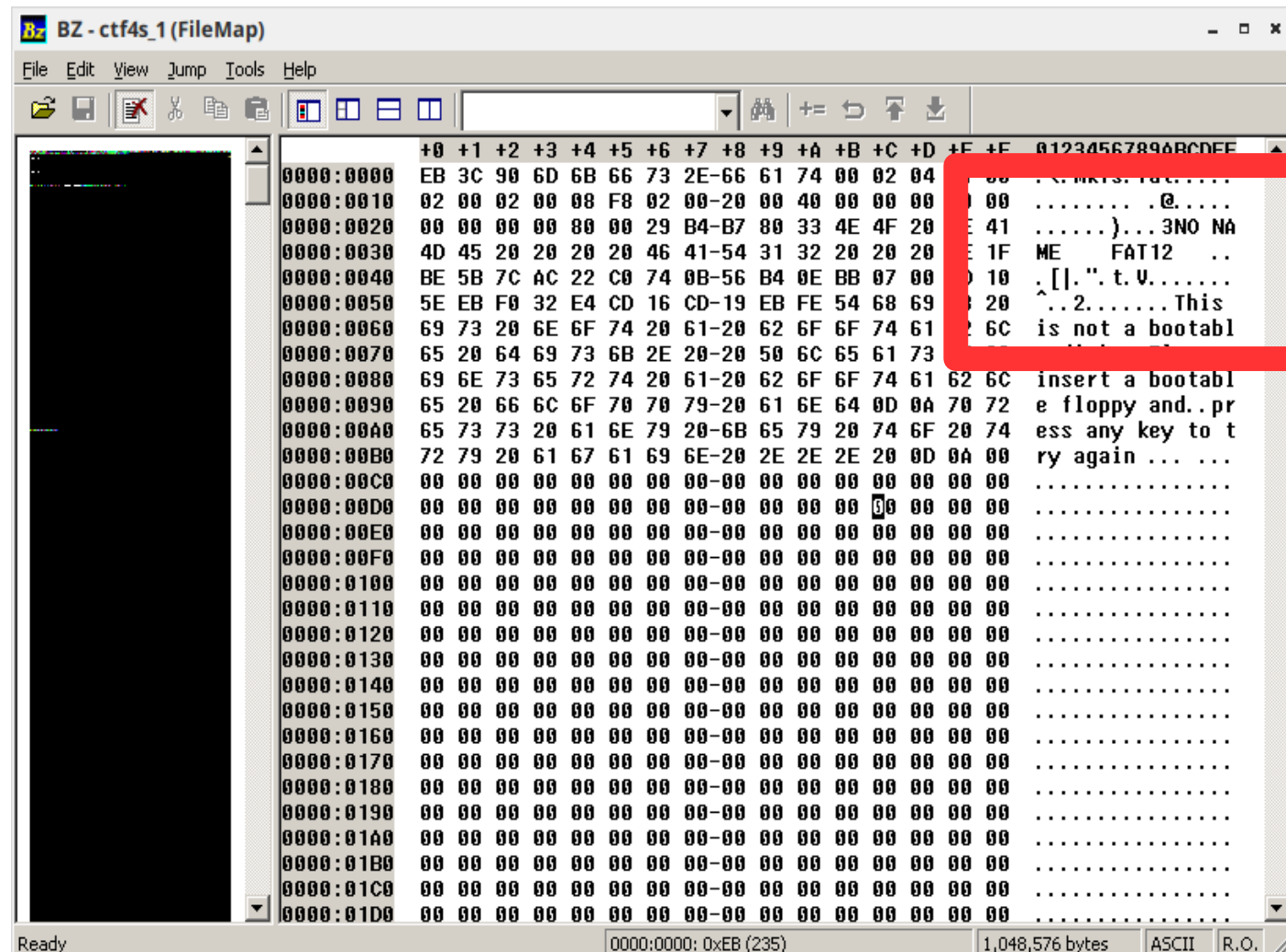
- ・バイナリエディタで怪しいところをチェック



問題 1 の答え

●解き方 2

- ファイルの頭（ヘッダ）をみて何のファイルか読み取る
- FAT12 とあるのでマウントできる
 - `mount -t vfat ctf4s_1 hoge`
- 後は開いて取り出すだけ



問題 1 の答え

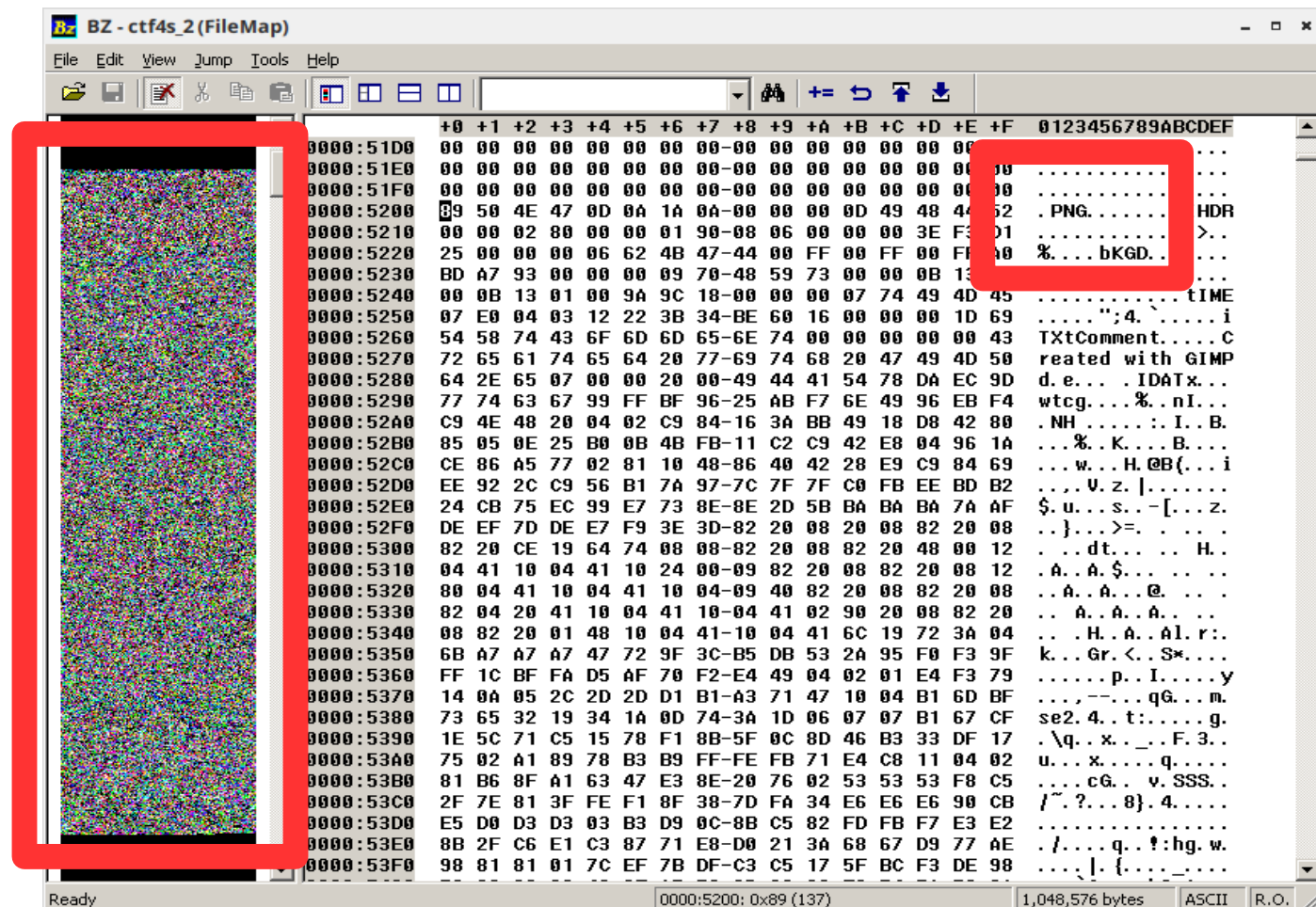
- 解き方 2(もっと楽なの)
 - file コマンドを使えば
ファイルの素性がわかる
 - FAT12 とあるので
マウントできる
 - `mount -t vfat ctf4s_1 hoge`
 - 後は開いて取り出すだけ

```
tnishinaga@arch-tx201:~/Dropbox/勉強会/CTF4s/2016_kanazawa/mondai
File Edit Tabs Help
[tnishinaga@arch-tx201 mondai]$ file ctf4s_1
ctf4s_1: DOS/MBR boot sector, code offset 0x3c+2, OEM-ID "mkfs.fat", sectors/cluster 4, root entries 512, sectors 2048 (volumes <=32 MB) , Media descriptor 0xf8, sectors/FAT 2, sectors/track 32, heads 64, serial number 0x3380b7b4, unlabeled, FAT (12 bit)
[tnishinaga@arch-tx201 mondai]$
```

問題 2 の答え

● 解き方 1

- バイナリエディタで怪しいところをチェック
 - 謎のデータがある
 - PNG?
- 怪しいところを選択してコピー.
新しいファイルを開いてペースト
- 保存



問題 2 の答え

●解き方 1

- 保存したファイルを file コマンドで調査
- PNG なので普通に開く

```
tnishinaga@arch-tx201:~/Dropbox/勉強会/CTF4s/2016_kanazawa/mondai
File Edit Tabs Help
[tnishinaga@arch-tx201 mondai]$ file hoge
hoge: PNG image data, 640 x 400, 8-bit/color RGBA, non-interlaced
[tnishinaga@arch-tx201 mondai]$
```



Flag{Welcome_to_CTF4s}

暗号問題を解こう (初級編)

シーザー暗号とは

- 換字暗号. ある文字をある文字に置き換えるだけ
 - 例 : H->O, O->G, G->E, E->H のとき
 - 暗号化前 : HOGHEHOG
 - 暗号化後 : OGEHOGEH
- 解くのは簡単
 - 方法 1: 換字表を入手する
 - 方法 2: 頻出単語から推測する
 - 英語だと e, t, a, o, i が頻出単語
 - 暗号文の頻出単語にこれらを割り当ててみて意味が通るか考える
 - 方法 3: ツールを使う
 - <http://www.xarg.org/tools/caesar-cipher/>

ROT13 とは

- シーザー暗号の一種.
- あるアルファベットの 13 文字後に置き換える
- 解き方も同じ

例題

- 暗号文 : Jrypbzr gb PGS sbe Fgnegref!!
- ヒント : ROT13

こたえ

- Welcome to CTF for Starters!!
- シーザー暗号とわかっているなら
 - ツールを使うと楽
 - <http://www.xarg.org/tools/caesar-cipher/>
- どの暗号かわからない時
 - 勝手に調べてくれるところに投げしてみる
 - <http://www.quipqiup.com/>
 - たいていヒントとして暗号の手がかりがある

base64 問題の解き方

- データを文字列で表すためのエンコード方式（暗号ではない）
- 元データを 6bit ずつに分けて変換.
- 足りないところは「 = 」で埋める
 - 例 : “hoge” => “aG9nZQ==”
- Chrome なら開発者向けのコンソールを使って変換できる
 - `btoa(“base64 エンコードしたい文字列”)`
 - `atob(“base64 デーコードしたい文字列”)`
- 例題 :
 - <http://ctf.katsudon.org/problem/1>