1116 17 0035 村瀬晃

2. Vigenere 暗号解読

2.1. 解読した平文と暗号鍵

平文は intelligentinformationengineering で,

Intelligent information engineering と読むことができ，インテリジェント情報工学と訳すことができた.

暗号鍵は VIGENERECODE である.

2.2. 解読の手順とソースコード

「最初の 8 文字は INTELLIG」と与えられていたのでリスト 1 の Attack 関数で平文の最初 4 文字が i,n,t,e となる鍵を見つけることにした. 8 文字すべてを探すことも考慮したが, for 文を 8 回回すのに時間がかかりすぎるため 4 文字ずつにしようと考えた.

最初の文字が「i,n,t,e,l,l,I,g」 となる鍵が「VIGENERE」だと分かったので, 次に続く文字が「e,n,t」か「e,n,c,e」なのではないかと推測し, 順々に試した. 「e,n,t」が後に続くと推測したところ複数のリスト 1 のコードの出力結果のうち, 表 1 のような結果がでた.

最後に, t の次に来るのが「i」だと推測し,当てはめたところうまく解読できた.

リスト 1 に今回使ったコードを示す.使用言語は Python で, 実行環境は Windows10 である.

表 1

Key might be VIGENERECODZ Plaintext might be intelligentnnformationesgineering... intelligentnnformationesgineering

リスト 1 Python による暗号解読

```
 1  # 復号アルゴリズム
 2  def Dec(cipher, key, table):
 3      plain = ""
 4      text_l = "abcdefghijklmnopqrstuvwxyz"
 5      text_u = "ABCDEFGHIJKLMNOPQRSTUVWXYZ"
 6      for i,c in enumerate(cipher):
 7          plain += text_l[table[text_u.index(key[i % len(key)])].index(c)]
 8      return plain
 9  def TableGen():
10      table = []


11      text = "ABCDEFGHIJKLMNOPQRSTUVWXYZ"
12      for i in range(26):
```

```
13              table.append(text)
14              text = text[1:] + text[0]
15      return table
16  def Attack(cipher):
17      cipher = cipher
18      alphabet_upper = "ABCDEFGHIJKLMNOPQRSTUVWXYZ"
19  #DVZIYPZKGBWMINUVZEKMQBHRBQTIRVZRI
20      for letter_1 in alphabet_upper:
21          for letter_2 in alphabet_upper:
22              for letter_3 in alphabet_upper:
23                  for letter_4 in alphabet_upper:
24                                  tmp_key = letter_1 + letter_2 + letter_3 +
    letter_4
25                                  tmp_plain = Dec(cipher, tmp_key, TableGen())
26                                  if ("i" in tmp_plain[0]) and ("n" in tmp_plain[1])
    and ("t" in tmp_plain[2]) and ("e" in tmp_plain[3]):
27                                      print("Key might be {}".format(tmp_key))
28                                      print("Plaintext might be
    {}".format(tmp_plain[:40] + "..."))
29                                      print(tmp_plain[0:])
30                                      return tmp_plain
31  def Attack2(cipher):
32      cipher = cipher
33      alphabet_upper = "ABCDEFGHIJKLMNOPQRSTUVWXYZ"
34  #DVZIYPZKGBWMINUVZEKMQBHRBQTIRVZRI
35      for letter_1 in alphabet_upper:
36          for letter_2 in alphabet_upper:
37              for letter_3 in alphabet_upper:
38                  for letter_4 in alphabet_upper:
39                                  tmp_key = "VIGENERE"+letter_1 + letter_2 +
    letter_3 + letter_4
40                                  tmp_plain = Dec(cipher, tmp_key, TableGen())
41                                  #if ("l" in tmp_plain[4]) and ("l" in tmp_plain[5])
    and ("i" in tmp_plain[6]) and ("g" in tmp_plain[7]):
42                                  if ("e" in tmp_plain[8]) and ("n" in tmp_plain[9])
    and ("t" in tmp_plain[10]) and ("i" in tmp_plain[11]):
```

```
43                                      print("Key might be {}".format(tmp_key))
44                                      print("Plaintext might be
    {}".format(tmp_plain[:40] + "..."))
45                                      print(tmp_plain[0:])
46                                      print()
47  def Attack3(cipher):
48      cipher = cipher
49      alphabet_upper = "ABCDEFGHIJKLMNOPQRSTUVWXYZ"
50  #DVZIYPZKGBWMINUVZEKMQBHRBQTIRVZRI
51      for letter_1 in alphabet_upper:
52          for letter_2 in alphabet_upper:
53              for letter_3 in alphabet_upper:
54                                  tmp_key = letter_1 + letter_2 + letter_3
55                                  tmp_plain = Dec(cipher, tmp_key, TableGen())
56                                  if ("e" in tmp_plain[0]) :
57                                      print("Key might be {}".format(tmp_key))
58                                      print("Plaintext might be
    {}".format(tmp_plain[:40] + "..."))
59                                      print(tmp_plain[0:])
60  if __name__=="__main__":
61      cipher = "DVZIYPZKGBWMINUVZEKMQBHRBQTIRVZRI"
62      cipher2="dhtgltqinforeweivtbnginewntnn"
63      cipher3="txfivjdrmatidxqnomceerinv"
64      #cipher = Attack(cipher)
65      #cipher_2=Attack2(cipher2.upper())
66      cipher3 = Attack2(cipher)
```