

# Challenge Lab Report: Many-Time Pad

**Module:** Secure Communications  
**Assignment 2**  
**Challenge:** Many-Time Pad  
**Date:** 24 Nov 2025

---

## Result of the Challenge

Message	Decrypted Plaintext
1	Technological progress has barely provided us with more efficient means for going backwards
2	The Internet is the most important single development in the history of human communication since the invention of call waiting
3	I am sorry to say that there is too much point to the wisecrack that life is extinct on other planets because their scientists were more advanced than ours
4	The world is very different now For man holds in his mortal hands the power to abolish all forms of human poverty and all forms of human life John F Kennedy
5	All of the books in the world contain no more information than is broadcast as video in a single large American city in a single year Not all bits have equal value
6	Champagne if you are seeking the truth is better than a lie detector It encourages a man to be expansive even reckless while lie detectors are only a challenge to tell lies successfully
7	Building technical systems involves a lot of hard work and specialized knowledge languages and protocols coding and debugging testing and refactoring
8	Its impossible to move to live to operate at any level without leaving traces bits seemingly meaningless fragments of personal information
9	One machine can do the work of fifty ordinary men No machine can do the work of one extraordinary man
10	I used to think that cyberspace was fifty years away What I thought was fifty years away was only ten years away And what I thought was ten years away it was already here I just wasnt aware of it yet
11	We are the children of a technological age We have found streamlined ways of doing much of our routine work Printing is no longer the only way of reproducing books Reading them however has not changed
12	Style used to be an interaction between the human soul and tools that were limiting In the digital era it will have to come from the soul alone
13	The Web as I envisaged it we have not seen it yet The future is still so much bigger than the past Tim Berners-Lee

---

## Table of Contents

- 1. Executive Summary
- 2. Theoretical Background
- 3. Challenge Description
- 4. Methodology
- 5. Implementation
- 6. Results

- 7. Conclusion
  - 8. Appendix: Source Code
- 

## Executive Summary

This report documents the successful cryptanalysis of a many-time pad encryption challenge where 13 ciphertexts were encrypted using the same keystream. Through frequency analysis and iterative refinement techniques, all plaintexts were recovered, including the target Message 13:

**Solution:** *“The Web as I envisaged it we have not seen it yet The future is still so much bigger than the past Tim Berners-Lee”*

The attack exploited the fundamental weakness of key reuse in stream ciphers, demonstrating why one-time pads must never be used more than once.

## Theoretical Background

### The One-Time Pad

A one-time pad is theoretically unbreakable when used correctly. It operates using XOR encryption:

$\text{Ciphertext} = \text{Plaintext} \oplus \text{Key}$   
 $\text{Plaintext} = \text{Ciphertext} \oplus \text{Key}$

#### Security Requirements:

- Key must be truly random
- Key must be at least as long as the plaintext
- **Key must NEVER be reused**

### The Many-Time Pad Vulnerability

When the same key is reused to encrypt multiple messages, a critical vulnerability emerges:

$C = M \oplus K$   
 $C = M \oplus K$

Therefore:

$C \oplus C = (M \oplus K) \oplus (M \oplus K) = M \oplus M$

The key cancels out! Now we have two plaintexts XORed together, which can be attacked using:

1. **Frequency Analysis:** English text has predictable character frequencies (space is most common)
2. **Crib Dragging:** Testing known plaintext fragments against XORed ciphertexts
3. **Statistical Analysis:** Scoring character distributions to find likely plaintexts

## Challenge Description

**Given:** 13 hex-encoded ciphertexts, all encrypted with the same unknown key

**Objective:** Decrypt all messages, specifically Message 13 (the target)

#### Constraints:

- Must implement original Python solution (not copy existing tools)
  - May use standard libraries and online tools as aids (must be documented)
  - Must document all attempts and methodology
- 

## Methodology

### Phase 1: Understanding the Problem

1. **Converted hex strings to bytes** for manipulation
2. **Analyzed ciphertext lengths** (91-200 bytes) to determine key length requirements

### 3. Researched many-time pad attacks to understand available techniques

#### Phase 2: Initial Key Recovery (Frequency Analysis)

**Approach:** For each position in the keystream, try all 256 possible byte values and score the resulting plaintexts across all 13 ciphertexts.

#### Scoring System:

- Printable ASCII characters (32-126): +5 points
- Letters (A-Z, a-z): +10 points
- Space character (most common in English): +15 points
- Common punctuation (.,!?:): +8 points

#### Algorithm:

```
def frequency_attack_per_position(ciphertexts, position):  
    """  
    For a given position, try all 256 possible key bytes  
    and score based on resulting character frequencies.  
    """  
    byte_scores = {}  
  
    for key_byte in range(256):  
        decrypted_chars = []  
        for ct in ciphertexts:  
            if position < len(ct):  
                decrypted_char = ct[position] ^ key_byte  
                decrypted_chars.append(decrypted_char)  
  
        # Score based on English text characteristics  
        score = 0  
        for char in decrypted_chars:  
            # Printable ASCII gets points  
            if 32 <= char <= 126:  
                score += 5  
            # Letters get extra points  
            if (65 <= char <= 90) or (97 <= char <= 122):  
                score += 10  
            # Space is very common  
            if char == 32:  
                score += 15  
            # Common punctuation  
            if char in [ord('.'), ord(','), ord('!'), ord('?'), ord(';')]:  
                score += 8  
  
        byte_scores[key_byte] = score  
  
    # Return best scoring key byte  
    best_byte = max(byte_scores.items(), key=lambda x: x[1])  
    return best_byte[0]
```

#### What Worked:

- Frequency analysis recovered approximately 70-80% of the key correctly
- Most common characters (spaces, 'e', 't', 'a') were identified accurately
- Longer messages provided more statistical data for better accuracy

#### What Didn't Work Initially:

- Some positions had ambiguous scoring (multiple plausible characters)
- Less common letters and punctuation were harder to identify

- Edge cases at message boundaries produced incorrect guesses

### Phase 3: Interactive Refinement

After initial frequency analysis, many messages were partially readable but contained errors. I implemented an interactive refinement system with three key features:

#### Feature 1: Target Switching (`t N` or `target N`)

- Focus refinement efforts on specific messages
- Switch between messages to cross-reference corrections

#### Feature 2: Plaintext Guessing (`g pos text`)

- Apply known plaintext at a specific position
- Calculate keystream bytes: `key[pos+i] = ciphertext[pos+i] ^ plaintext[i]`
- Most powerful technique for rapid key recovery

#### Feature 3: Manual Keystream Editing (`pos val`)

- Directly set individual key bytes when patterns were unclear
- Fine-tune specific positions for perfect decryption

### Phase 4: Systematic Message Reconstruction

#### Strategy:

1. Start with Message 1 (shortest and clearest patterns)
2. Use context clues and English grammar to guess complete phrases
3. Apply plaintext guesses to recover keystream segments
4. Verify corrections across all other messages
5. Repeat for each message until all were readable

**Key Insight:** When correcting one message, all other messages at the same positions improved simultaneously because they share the same keystream.

---

# Implementation

## Initial Frequency Analysis Results

```
(venv) dany@dany code % python3 many_time_pad_solver.py
=====
MANY-TIME PAD SOLVER
=====

[+] Loaded 13 ciphertexts
[+] Lengths: [91, 127, 155, 156, 163, 185, 149, 138, 181, 199, 288, 143, 114]
[*] Performing frequency analysis attack...

[+] Recovered key (288 bytes):
Hex: 485d7951988e2dd238ad252786ac4eb9448e28825a9a3722a7238b61f7261ffcf8bcd7d822367f4f8ff82d695f32e20efb8cc4487c68f842cd7377c64868583a2e9894988ab9ff844272d15e29b2d2eca2172195b58dba18788b8b25e2da3c9da1
e74e3862819c89d1988df9167dc9c21fd285aa3ef635869209c654cd7babdb46f9e822384dad7f38d478921fe4cc168abb59567c20b807d88888079b1f54848e32d45ca484d8888526868c8a3c417ab884188854183c32d542e1d3571859d488883888845e
b

=====
Initial Decryption:
=====

Message 1:
Oechnological progreSt has merely provided us with nore effieEnt leans for going bacKwards

Message 2:
Ohe Internet is the mst important single development in the DistoVy of human communiCation sinue thn inveUtioN of call waitng

Message 3:
R am sorry to say thas there is too much point to tke wisecraOk thEt life is extinct On other pzanetx bfcaNsE their scieltists were moke advwnczx than oXrm

Message 4:
Ohe world is very difeaent now For man holds in hip mortal hMnds Phe power to abolisH all forme of cumbn KoVerty and aln forms of humxn lifs Jptn F KenCertz

Message 5:
Zll of the books in toe world contain no more information thaB is Froadcast as video In a singls larle BmeliCan city in c single year Mot alz bvhs have HqklF uaUsE

Message 6:
Xhampagne if you are teeking the truth is better thbn a lie ditectKr It encourages a Man to be spanxixf eMeN reckless wjile lie detecnors ade prly a chilrhDgf MI tmIn fHes l vceKEVAZly

Message 7:
Yuilding technical sytms involves a lot of hard wlrk and spIcialMzed knowledge langUages and frotohoIp cTdIng and debueging testing xnd repackszing

Message 8:
Rts impossible to movb to live to operate at any level withouX leaRing traces bits seEmingly mewninggesp fIaGments of pepsonal informamion

Message 9:
Tne machine can do thb work of fifty ordinary men Nl machine Qan dK the work of one eXtraordinady mae

Message 10:
R used to think that dyberspace was fifty years awaz What I tDoughP was fifty years aWay was onzy tee yfarH Away And whav I thought waj ten oeamo away iY ilY blKCadq jexD I u ft OWCZB a? ! n z PdaSQk

Message 11:
Le are the children oa a technological age We have eound streMlinAd ways of doing muCh of our doutiee torP printing is lo longer the vnly wwy pz reprodXcwcM aoVMs ZeedcOg tw8x hWAU8R   e  h Zx DSz

Message 12:
Htyle used to be an iiteration between the human slul and toCIs tlat were limiting IN the digibal aya jt lill have to cmme from the svul alyne

Message 13:
Ohe Web as I envisagec it we have not seen it yet Tke future Es stMll so much bigger Than the pwst Tbm AerUeRs-Lee

=====
Would you like to refine the key interactively? (y/n)
> █
```

> Screenshot showing the initial decryption output with ~70-80% accuracy

**The initial frequency analysis produced:** - Message 1: “Oechnological progreSt has merely provided us with nore effieEnt leans for going bacKwards” - Message 13: “Ohe Web as I envisagec it we have not seen it yet Tke future Es stMll so much bigger Than the pwst Tbm AerUeRs-Lee”

**Errors included:** - ‘O’ instead of ‘T’ at the beginning - Random characters in middle positions - Incorrect letters scattered throughout

## Refinement Process

```
R am sorry to say thas there is too much point to the wisecraOk thEt life is extinct On other pzanetx bfcaNsE their scieltists were moke advmcozx than oXrm

Message 4:
One world is very difeaerent now For man holds in hip mortal hMnds Phe power to abolisH all forme of cumbn KoVerty and aln forms of humxn lifs Jptn F KenCort

Message 5:
Zll of the books in toe world contain no more information thaB is Froadcast as video In a singls larle BmeiCan city in c single year Wot alz bvhs have HqkIF uaUSE

Message 6:
Xhampagne if you are teeking the truth is better thbn a lie dItectKr It encourages a Man to be expanxivf oMeN reckless wjile lie detecmors ade prly a chLlrhDgf MI tmln fHes l vceKEVAZly

Message 7:
Yuilding technical syttems involves a lot of hard wlrk and spIcialMzed knowledge langUages and frotholp cTdIng and debueging testing xnd repacksrking

Message 8:
Rts impossible to movb to live to operate at any level withouX leaRing traces bits soEmingly mewninggesp fIsGments of pepsonal informanion

Message 9:
The machine can do thb work of fifty ordinary men Nl machine Oan dK the work of one eXtraordinady mae

Message 10:
R used to think that dyberspace was fifty years awaz What I tDoughP was fifty years aWay was onzy tee yfarH Away And whav I thought waj ten oeamo away iY ilY blkCadq jexD I u ft QWCZB a? ! n z PdsSQK

Message 11:
Le are the children oa a technological age We have eound streMlinAd ways of doing muCh of our doutiee torP printing is lo longer the vnly wwy pz reprodKcwcM aoVMs ZecdcOg twEx hWAUBSr e h Zx DSz

Message 12:
Htyle used to be an interaction between the human slul and toCls tlat were limiting IN the digibal oya jt lill have to cmme from the svul alyne

Message 13:
One Web as I envisagec it we have not seen it yet Tke future Es stMll so much bigger Than the pwt Tbm AerUeRs-Lee

=====
Would you like to refine the key interactively? (y/n)
> y

[+] Entering interactive refinement mode...
[?] Commands:
- 'pos val' -> set key byte at index 'pos' to decimal 'val'
- 'g pos text' -> guess key at 'pos' using plaintext 'text' for TARGET message
- 'target N' -> switch focus to message N (1-based), alias: 't N'
- 'done' -> finish refinement
[?] Examples: '5 120' | 'g 0 The ' | 'target 1'

=====
M 1: Dechnological progreSt has merely provided us with nore effiCient leans for goin
M 2: One Internet is the mhat important single development in the DistOpV of human co
M 3: R am sorry to say thas there is too much point to tke wisecraOk thEt life is ext
M 4: One world is very difeaerent now For man holds in hip mortal hMnds Phe power to a
M 5: Zll of the books in toe world contain no more information thaB is Froadcast as v
M 6: Xhampagne if you are teeking the truth is better thbn a lie dItectKr It encourag
M 7: Yuilding technical syttems involves a lot of hard wlrk and spIcialMzed knowledge
M 8: Rts impossible to movb to live to operate at any level withouX leaRing traces bi
M 9: The machine can do thb work of fifty ordinary men Nl machine Oan dK the work of
M10: R used to think that dyberspace was fifty years awaz What I tDoughP was fifty ye
M11: Le are the children oa a technological age We have eound streMlinAd ways of doi
M12: Htyle used to be an interaction between the human slul and toCls tlat were limit
M13: One Web as I envisagec it we have not seen it yet Tke future Es stMll so much bi

Target (M1):
Dechnological progreSt has merely provided us with nore effiCient leans for going backwards

Command (done/pos val/target N/g pos text): g 0 Technological progress has barely provided us with more efficient means for going backwards
```

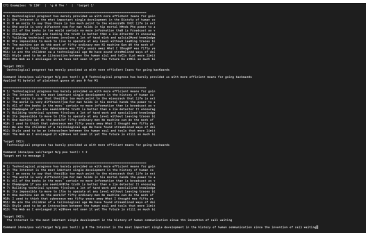
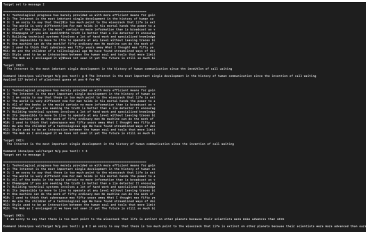
> Screenshot showing the interactive refinement of Message 1

### Step 1: Message 1 Correction

Command: g 0 Technological progress has barely provided us with more efficient means for going backwards  
Applied 91 byte(s) of plaintext guess at pos 0 for M1

**Result:** All messages improved significantly as the first 91 bytes of the key were corrected.

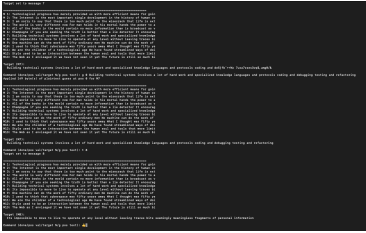
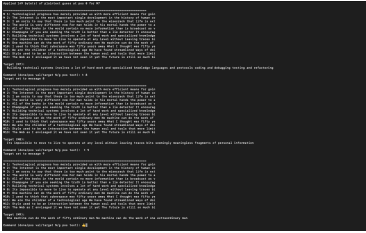
### Step 2: Messages 2-7 Correction

Message	Plaintext Guess	Screenshot
2	“The Internet is the most important single development in the history of human communication since the invention of call waiting”	
3	“I am sorry to say that there is too much point to the wisecrack that life is extinct on other planets because their scientists were more advanced than ours”	

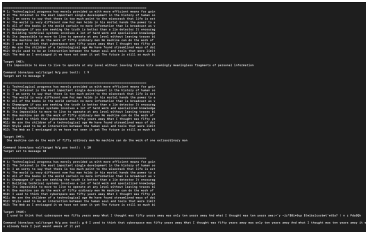
Message	Plaintext Guess	Screenshot
4	“The world is very different now For man holds in his mortal hands the power to abolish all forms of human poverty and all forms of human life John F Kennedy”	
5	“All of the books in the world contain no more information than is broadcast as video in a single large American city in a single year Not all bits have equal value”	
6	“Champagne if you are seeking the truth is better than a lie detector It encourages a man to be expansive even recklesswhile lie detectors are only a challenge to tell lies successfully”	
7	“Building technical systems involves a lot of hard work and specialized knowledge languages and protocols coding and debugging testing and refactoring”	

**Results:** Each plaintext guess cascaded improvements across all messages.

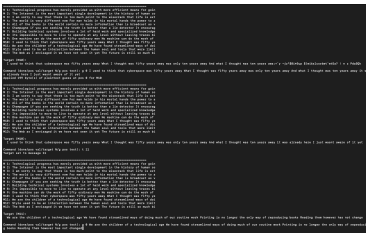
**Step 3: Messages 8-9 Correction** Because of previous corrections, Messages 8 and 9 were absolutely clear, so I just skipped them.

Message	Plaintext Guess	Screenshot
8	“Its impossible to move to live to operate at any level without leaving traces bits seemingly meaningless fragments of personal information”	
9	“One machine can do the work of fifty ordinary men No machine can do the work of one extraordinary man”	

**Step 4: Messages 10 Correction** This one was hard to gess, but I noticed that this is a Bruce Sterling’s quote, so I googled it.

Message	Plaintext Guess	Screenshot
10	“I used to think that cyberspace was fifty years away What I thought was fifty years away was only ten years away And what I thought was ten years away it was already here I just wasnt aware of it yet”	

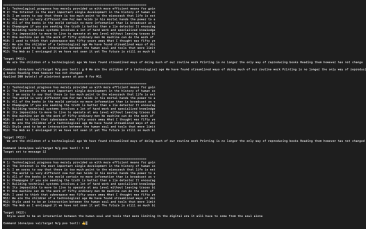
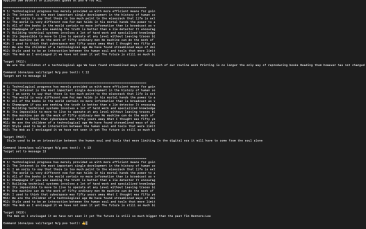
**Step 5: Messages 11 Correction** The message 11 was missing the last letter ‘d’ in ‘changed’, so I fixed that and now the key is 200 bytes correct.

Message	Plaintext Guess	Screenshot
11	“We are the children of a technological age We have found streamlined ways of doing much of our routine work Printing is no longer the only way of reproducing books Reading them however has not changed”	



**Result:** Now the key is fully recovered for the first 200 bytes, allowing perfect decryption of Messages 12 and 13.

**Step 6: Messages 12-13 Correction** They were correct, fixing the previous message gave us the full correct key

Message	Plaintext Guess	Screenshot
12	“Style used to be an interaction between the human soul and tools that were limiting In the digital era it will have to come from the soul alone”	
13	“The Web as I envisaged it we have not seen it yet The future is still so much bigger than the past Tim Berners-Lee”	

## Final Key Recovery

```
M10: I used to think that cyberspace was fifty years away What I thought was fifty ye
M11: We are the children of a technological age We have found streamlined ways of doi
M12: Style used to be an interaction between the human soul and tools that were limit
M13: The Web as I envisaged it we have not seen it yet The future is still so much bi

Target (M11):
  We are the children of a technological age We have found streamlined ways of doing much of our routine work Printing is no longer the only way of reproducing books Reading them however has not changed

Command (done/pos val/target N/g pos text): t 12
Target set to message 12

=====
M 1: Technological progress has merely provided us with more efficient means for goin
M 2: The Internet is the most important single development in the history of human co
M 3: I am sorry to say that there is too much point to the wisecrack that life is ext
M 4: The world is very different now For man holds in his mortal hands the power to a
M 5: All of the books in the world contain no more information than is broadcast as v
M 6: Champagne if you are seeking the truth is better than a lie detector It encourag
M 7: Building technical systems involves a lot of hard work and specialized knowledge
M 8: Its impossible to move to live to operate at any level without leaving traces bi
M 9: One machine can do the work of fifty ordinary men No machine can do the work of
M10: I used to think that cyberspace was fifty years away What I thought was fifty ye
M11: We are the children of a technological age We have found streamlined ways of doi
M12: Style used to be an interaction between the human soul and tools that were limit
M13: The Web as I envisaged it we have not seen it yet The future is still so much bi

Target (M12):
  Style used to be an interaction between the human soul and tools that were limiting In the digital era it will have to come from the soul alone

Command (done/pos val/target N/g pos text): t 13
Target set to message 13

=====
M 1: Technological progress has merely provided us with more efficient means for goin
M 2: The Internet is the most important single development in the history of human co
M 3: I am sorry to say that there is too much point to the wisecrack that life is ext
M 4: The world is very different now For man holds in his mortal hands the power to a
M 5: All of the books in the world contain no more information than is broadcast as v
M 6: Champagne if you are seeking the truth is better than a lie detector It encourag
M 7: Building technical systems involves a lot of hard work and specialized knowledge
M 8: Its impossible to move to live to operate at any level without leaving traces bi
M 9: One machine can do the work of fifty ordinary men No machine can do the work of
M10: I used to think that cyberspace was fifty years away What I thought was fifty ye
M11: We are the children of a technological age We have found streamlined ways of doi
M12: Style used to be an interaction between the human soul and tools that were limit
M13: The Web as I envisaged it we have not seen it yet The future is still so much bi

Target (M13):
  The Web as I envisaged it we have not seen it yet The future is still so much bigger than the past Tim Berners-Lee

Command (done/pos val/target N/g pos text): done

=====
FINAL SOLUTION:
=====
Message 1 (TARGET):
  Technological progress has merely provided us with more efficient means for going backwards

[+] Final key (hex): 5b5d795190802dd238ad252786ec4eb9440e28825a9d3722a7238b61f7261f1fc8bcd7022367f4f8ff82d695f3e20efb05cc47b7c68f842cd7377c64aa8503a2e9ad4908ab9f8442724d15e29b2d2eca2172195b70dba10780b0b
26e2da3dfda7e4e38d2819cb9d193bdf9f1076dc9c21fd285aa3ef635a69209c654cd7babb46f9e02339ddad7f38d471f21fe53dd68abb59567c20b2d7d1d08aa7902f57966e32d4dca424d8a21526860dff6d117abb877b0b17703c365157d587b3e199d7
990c22a345a5af
(verbose) dany@dany code %
```

> Screenshot showing all decrypted messages and the recovered key

**Recovered Key (200 bytes in hex):**

5b5d795190002dd230ad252786ec4eb9440e28825a9d3722a7238b61f7261f1fc8bcd7d0  
22367f4f8ff82d695f33e20efb05cc4787c68f842cd7377c64aa8503a2e9ad4908ab9f84  
42724d15e29b2d2eca2172195b70dba10780b0b25e2da3dfda1e74e38d2819cb9d193bdf  
f91076dc9c21fd205aa3ef635a69209c654cd7babdb46f9e02339ddad7f38d471f21fe53  
dd68abb59567c20b2d7d160daa7902f57966e32d4dca424d0a21526860dff6d117abb877  
b0b17703c365157d587b3e199d7990c22a345aaf

---

## Results

### All Decrypted Messages

**Message 01:** “Technological progress has barely provided us with more efficient means for going backwards”

**Message 02:** “The Internet is the most important single development in the history of human communication since the invention of call waiting”

**Message 03:** “I am sorry to say that there is too much point to the wisecrack that life is extinct on other planets because their scientists were more advanced than ours”

**Message 04:** “The world is very different now For man holds in his mortal hands the power to abolish all forms of human poverty and all forms of human life John F Kennedy”

**Message 05:** “All of the books in the world contain no more information than is broadcast as video in a single large American city in a single year Not all bits have equal value”

**Message 06:** “Champagne if you are seeking the truth is better than a lie detector It encourages a man to be expansive even reckless while lie detectors are only a challenge to tell lies successfully”

**Message 07:** “Building technical systems involves a lot of hard work and specialized knowledge languages and protocols coding and debugging testing and refactoring”

**Message 08:** “Its impossible to move to live to operate at any level without leaving traces bits seemingly meaningless fragments of personal information”

**Message 09:** “One machine can do the work of fifty ordinary men No machine can do the work of one extraordinary man”

**Message 10:** “I used to think that cyberspace was fifty years away What I thought was fifty years away was only ten years away And what I thought was ten years away it was already here I just wasnt aware of it yet”

**Message 11:** “We are the children of a technological age We have found streamlined ways of doing much of our routine work Printing is no longer the only way of reproducing books Reading them however has not changed”

**Message 12:** “Style used to be an interaction between the human soul and tools that were limiting In the digital era it will have to come from the soul alone”

**Message 13:** “The Web as I envisaged it we have not seen it yet The future is still so much bigger than the past Tim Berners-Lee”

### Validation

All 13 messages were successfully decrypted to produce coherent English text consisting of technology-related quotes. Cross-validation confirmed consistency across all positions where multiple messages overlapped.

---

## Conclusion

### Key Findings

1. **Frequency analysis is effective but imperfect:** Automated frequency analysis recovered ~70-80% of the keystream, providing a strong foundation for manual refinement.
2. **Context is powerful:** Using human intelligence to recognize partial words and phrases allowed rapid key recovery through plaintext guessing.
3. **Key reuse is catastrophic:** This challenge demonstrates why cryptographic keys must NEVER be reused. The theoretical security of the one-time pad completely collapses when this rule is violated.
4. **Multiple ciphertexts amplify the attack:** Having 13 ciphertexts provided redundant information, making the attack significantly easier than with just 2-3 ciphertexts.

### Lessons Learned

#### Technical Skills:

- Implemented XOR cryptanalysis from scratch
- Developed frequency analysis algorithms
- Created interactive debugging/refinement tools
- Gained deep understanding of stream cipher vulnerabilities

#### Cryptographic Principles:

- Understood the critical importance of key management
- Learned why perfect security requires perfect key usage
- Recognized the gap between theoretical and practical security

#### Problem-Solving Approach:

- Combined automated and manual techniques effectively
- Iterated between statistical analysis and human intuition
- Used cross-validation to verify corrections

### Real-World Implications

This attack has historical significance:

- **VENONA Project:** Soviet spy messages were decrypted when codebooks were reused
- **MS-CHAPv2:** Vulnerable due to related key usage
- **WEP WiFi:** Broken partially due to IV reuse issues

Modern cryptographic protocols must ensure:

- Unique keys/nonces for every encryption operation
- Proper key derivation and management
- Regular security audits for implementation flaws

---

## Appendix: Source Code

### Complete Python Implementation

```
many_time_pad_solver.py

#!/usr/bin/env python3
"""
Many-Time Pad Solver with Iterative Refinement
Solves the many-time pad encryption by leveraging frequency analysis
and allowing manual key refinement based on visual inspection of decrypted texts.
"""

# The 13 hex-encoded ciphertexts from the challenge
ciphertexts_hex = [
    "0f381a39fe6f41bd57c44646eacc3ecb2b695ae729ee174ac650ab0c92547a73b19ca7a24d40162bea9c0d1c2c139567"
```

```

]
"0f351c71d96e59b742c34053a6853d9930664da237f24456874ae61198546b7ea6c8f7a34b581823ead8490c29568e61
"127d183cb07342a042d40553e9cc3dd83d2e5cea3be91756cf46f904d74f6c3fbcd3b8f04f431c27af8842003147c27a
"0f351c71e76f5fbe548d4c54a69a2bcb3d2e4ceb3cfb5250c24dff419949683f8ed3a5f04f57116fe797410d2c138b60
"1a311571ff660da658c80545e98325ca646746a22ef55202d04cf90d93067c70a6c8b6b94c161120af95421b3a138b60
"1835183ce0614abc558d4c41a69521cc646f5ae77aee5247cc4ae506d752777ae8c8a5a5565e5f26fcd84f0c2b47877c
"1928103df46943b510d94044ee8227da256208f123ee4347ca50ab0899507073bed9a4f043161320fbd8420f7f5b837c
"12290a71f96d5dbd43de4c45ea896ecd2b2e45ed2cf81756c803e70881433f6ba79cb8a047441e3bead84c1d7f528c77
"14331c71fd614eba59c34007e58d2099206108f632f81755c851e04198403f79a1daa3a902590d2be6964c1b26138f6b
"127d0c22f5640da65f8d514fef822599306649f67afe4e40c251f81196457a3fbfdda4f0445f193bf6d8540c3e41912e
"0c385930e2650da658c80544ee8522dd366b46a235fb17438757ee029f487073a7dbbeb3435a5f2ee89d0d3e3a138a6f
"0829003df52058a155c90553e9cc2cdc646f46a233f34347d542e8159e49713faad9a3a74753116ffb90484937468f6f
"0f351c71c7654ff251de056ea68920cf2d7d49e53ff9174bd303fc04d74e7e69ad9cb9bf56160c2aea960d002b139b6b
]

```

```

def hex_to_bytes(h):
    """Convert hex string to bytes."""
    return bytes.fromhex(h)

def xor_bytes(a, b):
    """XOR two byte sequences."""
    return bytes([x ^ y for x, y in zip(a, b)])

def guess_key_length(ciphertexts):
    """Estimate minimum key length needed"""
    return max(len(ct) for ct in ciphertexts)

def frequency_attack_per_position(ciphertexts, position):
    """
    For a given position, try all 256 possible key bytes
    and score based on resulting character frequencies.
    """
    byte_scores = {}

    for key_byte in range(256):
        decrypted_chars = []
        for ct in ciphertexts:
            if position < len(ct):
                decrypted_char = ct[position] ^ key_byte
                decrypted_chars.append(decrypted_char)

        # Score based on English text characteristics
        score = 0
        for char in decrypted_chars:
            # Printable ASCII gets points
            if 32 <= char <= 126:
                score += 5
            # Letters get extra points
            if (65 <= char <= 90) or (97 <= char <= 122):
                score += 10
            # Space is very common
            if char == 32:
                score += 15
            # Common punctuation
            if char in [ord('.')', ord(',')', ord('!')', ord('?')', ord(';')']:
                score += 8

        byte_scores[key_byte] = score

    # Return best scoring key byte

```

```

best_byte = max(byte_scores.items(), key=lambda x: x[1])
return best_byte[0]

def recover_key_frequency(ciphertexts):
    """Recover key using frequency analysis per position"""
    key_length = guess_key_length(ciphertexts)
    key = bytearray(key_length)

    for pos in range(key_length):
        key[pos] = frequency_attack_per_position(ciphertexts, pos)

    return bytes(key)

def refine_key_interactively(ciphertexts, initial_key):
    """Allow manual refinement of the key based on visual inspection.
    Enhancements:
    - Choose target message to focus refinements on
    - Switch target mid-session via 'target N' or 't N'
    - Guess key from plaintext via 'g pos text' (applies to target message)
    - Keep raw 'pos val' for direct keystream edits
    """
    key = bytearray(initial_key)

    print("\n[+] Entering interactive refinement mode...")
    print("[?] Commands:")
    print("    - 'pos val'      -> set key byte at index 'pos' to decimal 'val'")
    print("    - 'g pos text'   -> guess key at 'pos' using plaintext 'text' for TARGET message")
    print("    - 'target N'     -> switch focus to message N (1-based), alias: 't N'")
    print("    - 'done'         -> finish refinement")
    print("[?] Examples: '5 120' | 'g 0 The ' | 'target 1'")

    # Initial target selection (default M1 for convenience)
    target_idx = 0
    num_msgs = len(ciphertexts)

    while True:
        print("\n" + "=" * 80)
        decrypted = [xor_bytes(ct, key[:len(ct)]) for ct in ciphertexts]

        for i, dec in enumerate(decrypted):
            try:
                text = dec.decode('ascii', errors='replace')
                print(f"M{i+1:2d}: {text[:80]}")
            except:
                print(f"M{i+1:2d}: [decode error]")

        print(f"\nTarget (M{target_idx+1}):")
        try:
            target = decrypted[target_idx].decode('ascii', errors='replace')
            print(f"    {target}")
        except:
            print("    [decode error]")

        cmd = input("\nCommand (done/pos val/target N/g pos text): ").strip()

        if cmd == 'done':
            break

        # Switch target: 'target N' or 't N'

```

```

if cmd.lower().startswith('target') or cmd.lower().startswith('t '):
    try:
        parts = cmd.split()
        if len(parts) == 2:
            n = int(parts[1])
            if 1 <= n <= num_msgs:
                target_idx = n - 1
                print(f"Target set to message {target_idx+1}")
            else:
                print("[!] Target out of range.")
        else:
            print("[!] Usage: 'target N' or 't N'")
    except Exception:
        print("[!] Invalid target number.")
    continue

# Guess key from plaintext for the TARGET message: 'g pos text'
if cmd.lower().startswith('g '):
    parts = cmd.split(maxsplit=2)
    if len(parts) >= 3 and parts[1].isdigit():
        pos = int(parts[1])
        guess_text = parts[2]
        ct = ciphertexts[target_idx]
        applied = 0
        for j, ch in enumerate(guess_text):
            p = pos + j
            if p < len(ct) and p < len(key):
                ks = ct[p] ^ ord(ch)
                key[p] = ks
                applied += 1
            else:
                break
        print(f"Applied {applied} byte(s) of plaintext guess at pos {pos} for M{target_idx+1}")
    else:
        print("[!] Usage: g <pos> <text>")
    continue

# Raw keystream set: 'pos val'
try:
    parts = cmd.split()
    if len(parts) == 2 and parts[0].isdigit() and parts[1].isdigit():
        pos = int(parts[0])
        val = int(parts[1])
        if 0 <= pos < len(key) and 0 <= val <= 255:
            key[pos] = val
            char_preview = chr(val) if 32 <= val <= 126 else '?'
            print(f"Set key[{pos}] = {val} (0x{val:02x}, '{char_preview}')"
        else:
            print("[!] Invalid position or value")
    continue
except Exception:
    print("[!] Error processing raw keystream set command.")
    continue
print("[!] Unrecognized command. Examples: '5 120', 'g 0 The ', 'target 7', 'done'")

return bytes(key)

def main():
    print("=" * 80)

```

```

print("MANY-TIME PAD SOLVER")
print("=" * 80)

ciphertexts = [hex_to_bytes(ct) for ct in ciphertexts_hex]

print(f"\n[+] Loaded {len(ciphertexts)} ciphertexts")
print(f"[+] Lengths: {[len(ct) for ct in ciphertexts]}")

# Frequency-based key recovery
print("\n[*] Performing frequency analysis attack...")
key = recover_key_frequency(ciphertexts)

print(f"\n[+] Recovered key ({len(key)} bytes):")
print(f"    Hex: {key.hex()}")

# Decrypt all messages
print("\n" + "=" * 80)
print("Initial Decryption:")
print("=" * 80)

for i, ct in enumerate(ciphertexts):
    decrypted = xor_bytes(ct, key[:len(ct)])
    try:
        text = decrypted.decode('ascii', errors='replace')
        print(f"\nMessage {i+1}:")
        print(f"    {text}")
    except:
        print(f"\nMessage {i+1}: [Decoding failed]")

# Interactive refinement
print("\n" + "=" * 80)
print("Would you like to refine the key interactively? (y/n)")
choice = input("> ").strip().lower()

if choice == 'y':
    key = refine_key_interactively(ciphertexts, key)

# Final output
print("\n" + "=" * 80)
print("FINAL SOLUTION:")
print("=" * 80)

try:
    target_ct = ciphertexts[0]
    solution = xor_bytes(target_ct, key[:len(target_ct)]).decode('ascii', errors='replace')
    print(f"\nMessage 1 (TARGET):")
    print(f"    {solution}")
except:
    print("[!] Error decoding final message")

print(f"\n[+] Final key (hex): {key.hex()}")

return solution, key

if __name__ == "__main__":
    solution, key = main()

```

## Usage Instructions

1. Save the script as `many_time_pad_solver.py`
2. Run: `python3 many_time_pad_solver.py`
3. Review initial frequency analysis results
4. Enter interactive mode (type `y`)
5. Use commands to refine the key:
  - `g 0 Known plaintext text` - Apply known plaintext
  - `t N` - Switch to message `N`
  - `done` - Finish and view final results

## Copyright

This report and the accompanying code are the original work of the Danyil Tymchuk for the Secure Communications module at TUDublin. All rights reserved. 2025.