

高一上學期

多元選修課程

資訊安全
與

概論

の學習紀錄

聶宜帆

目錄

—課前感想-為什麼選擇這門課—

—9/15編碼與解碼—

—9/22密碼學概論—

—9/29編碼與解碼—

—10/13隱寫術—

—10/20 Linux簡介—

—10/27高中跨校資安實務素養體驗賽-個人賽—

—11/03人工智慧AI簡介—

—11/10高中跨校資安實務素養體驗賽-團體賽—

—11/17模型逆向攻擊MNIST訓練模型—

—11/24 SQL Injection學習與實作—

—12/01封包攔截-Burp Suite攔截—

—12/08—

—12/15—

—12/22—

為什麼選擇這門課

「編碼與解碼」、「現代密碼學」、「網路封包實戰」，課程大綱映入我眼簾。小時候，應該沒有人沒幻想過自己使用神秘的摩斯密碼與人溝通，亦或是當個網路上帥氣的駭客。現在的我，看著眼前的「攻擊實務演練」，想像著與駭客的程式戰，心中默默憧憬著——或許這門課能讓我真正踏進過去只存在於想像中的世界。

那些看似複雜的技術名詞，背後究竟藏著什麼秘密？駭客的攻擊是如何被發現、被阻擋的？比起電影裡幾秒鐘完成的情節，我更好奇這些攻擊背後所運用的原理與邏輯。

帶著這種半期待、半好奇的心情，我選擇了這門課。希望在未來的課堂上，能逐漸看懂那些在我眼中原本像魔法般的指令與工具，理解它們的用途與意義，看見資安世界真正的樣貌。這門課對我來說是學習一項未知的技術，也是一扇讓我接觸另一種思考方式的入口。我期待能在這堂課裡，將那些曾經只存在於幻想中的酷炫場景，一步步化為真實。

09/15

編碼與解碼

- ASCLL 學習
- Morse Code 摩斯密碼學習
- Base64 學習

ASCII學習

- 將「HELLO」編碼

This screenshot shows a web-based ASCII conversion tool. On the left, a text input field contains the word "HELLO". On the right, the corresponding ASCII code "72 69 76 76 79" is displayed. Both sides have "明確" (Confirm) and "複製" (Copy) buttons at the top. Below each text area is a black bar with a white arrow pointing right, labeled "發給Ascii >" on the left and "< 文本到ASCII" on the right.

- 將「66 114 101 97 107 65 76 76 67 84 70 123 65 109 118 48 117 68 121 101 114 118 80 116 109 86 114 57 83 83 83 75 125」解碼

This screenshot shows a web-based ASCII conversion tool. On the left, a text input field contains the ASCII code "66 114 101 97 107 65 76 76 67 84 70 123 65 109 118 48 117 68 121 101 114 118 80 116 109 86 114 57 83 83 83 75 125". On the right, the decoded text "BreakALLCTF{Amv0uDyervPtmVr9SSSK}" is displayed. Both sides have "明確" (Confirm) and "複製" (Copy) buttons at the top. Below each text area is a black bar with a white arrow pointing right, labeled "發給Ascii >" on the left and "< 文本到ASCII" on the right.

使用的網站：<https://www.aperisolve.com/>

Morse code 摩斯密碼學習

- 將「I love you」編碼

The screenshot shows a mobile application interface. At the top left, the word "Input:" is followed by a text input field containing the text "'I love you'". Below this, the word "Output:" is followed by a text output field containing the Morse code representation ".----. . . / .-.. - --- .-.- / -.-. - --- ..-". At the bottom, there is a row of ten buttons: "Play" (orange), "Pause" (grey), "Stop" (grey), "Repeat" (blue), "Sound" (yellow, highlighted with a yellow border), "Light" (blue), "Vibrate" (grey), "Configure" (green), "Save Audio" (green), and "Share" (green).

-將「.. - . . -. - - - . . . - . - . . - . . - . . - - - - - - - - . - - . - - - . 」解碼

Base64學習

Man的Base64編碼

文字	M	a	n	
ASCII編碼	77	97	110	
位元	0 1 0 0 1 1 0 1 0 1 1 0 0 0 0 1 0 1 1 0 1 1 1 1 0			
索引	19	22	5	46
Base64編碼	T	W	F	u

教育局先導資助方案
資訊人才獎勵計畫
高中職資訊教育資源推動中心

- 將字串 「BreakallCTF{happyhackinghighhaaha}」 轉換成 Base64

BreakallCTF{happyhackinghighhaaha}

清空 加密 解密 解密為 UTF-8 字符 ▾

QnJlYWtbGxDVEZ7aGFwcHloYWNraW5naGlnaGhhYWhhfQ==

複製

- 將 Base64 字串 「QnJlYWtbTExDVEZ7NTN1c1pRM2hXVzI1ZGN0WjdkWGV9」 解碼

QnJlYWtbTExDVEZ7NTN1c1pRM2hXVzI1ZGN0WjdkWGV9

清空 加密 解密 解密為 UTF-8 字符 ▾

BreakALLCTF{53usZQ3hWW25dchZ7dXe}

複製

使用的網站：<https://www.aperisolve.com/>

利用Python解碼

將下列十六進位編碼字串解碼:

10 分

6236343a20615735305a584a755a58526659323975646d567963326c76626c
3930623239736331397962324e72

您的回答



請將前一題結果再進行解碼

10 分

您的回答



[2]
0
秒

```
import base64
hex_string = '6236343a20615735305a584a755a58526659323975646d567963326c76626c3930623239736331397962324e72'
decoded_bytes = bytes.fromhex(hex_string)
decoded_string = decoded_bytes.decode('utf-8')
print(decoded_string)
```

☒ b64: aW50ZXJuZXRfY29udmVyc21vb190b29sc19yb2Nr

[4]
0
秒

```
import base64
base64_string = 'aW50ZXJuZXRfY29udmVyc21vb190b29sc19yb2Nr'
decoded_bytes = base64.b64decode(base64_string)
decoded_string = decoded_bytes.decode('utf-8')
print(decoded_string)
```

☒ internet_conversion_tools_rock

△利用Chat gpt寫出程式後再使用google AI更正錯誤

09/22

密 碼 學 概 論

- 密碼學學習
 - 使用軟體進行訊息加密
 - 使用該軟體解題

密碼學學習



現代密碼學分類 (type of key)

對稱式密碼

symmetric key algorithms
Private-key cryptography



加密與解密都是使用同一把key

<https://en.wikipedia.org/wiki/Cipher>

非對稱式密碼

asymmetric key algorithms
Public-key cryptography



https://en.wikipedia.org/wiki/Outline_of_cryptography

臺灣省高級中等學校
資訊人才培養計畫
高中職實務教學資源與技術中心



危險的對稱式密碼 - XOR ciphers

- 使用XOR 邏輯運算建構簡易的密碼(容易被破)

A	B	A AND B	A OR B	A \oplus B	(A \oplus B) \oplus B	(A \oplus B) \oplus A
0	0	0	0	0	0	0
0	1	0	1	1	0	1
1	0	0	1	1	1	0
1	1	1	1	0	1	1

⊕ Boolean exclusive-OR operation

臺灣省高級中等學校
資訊人才培養計畫
高中職實務教學資源與技術中心

- 使用軟體進行訊息加密

Message to encrypt:

Hello

Caesar Output:

Khoor

5 characters, 1 line

Message to encrypt:

Hello

AES Output:

CA 14 5D 08 CD E6 31 6D 1C 6E 6A 7F E4 7E 81 27

47 characters, 1 line

- 使用該軟體解題

請使用XOR加密以下內容，*

10分

Cryptography is wonderful.

key為

99 AA BB CC DD EE FF

答案為

DA D8 C2 BC A9 81 98 EB CB CB A4 A4 CE 96

Message to encrypt/decrypt:

Cryptography is wonderful.

XOR Output:

DA D8 C2 BC A9 81 98 EB CB CB A4 A4 CE 96 EA 8A CC A3 B3 8A 9A EB CC CE A0 F3

77 characters, 1 line

請使用RSA解密以下內容，N=299, e=23

* 10分

60 00 C1 00 3E 00 F9 00 76 00 93 00 C1 00 02 01 FD 00 4B 00 56 00 56 00

93 00 74 00 A0 00 69 00 7A 00 C1 00 3D 00 93 00 4B 00 C1 00 02 01 3E 00

67 00 F9 00 A0 00 69 00 74 00 68 00 56 00 FD 00 B8

答案為

I love asymmetric key algorithms.

Plaintext:

I love asymmetric key algorithms.

33 characters, 1 line

Decrypted Text:

10/13

隱寫術

- 使用網站解密解密隱藏在片中的文字

Decode image

To decode a hidden message from an image, just choose an image and hit the **Decode** button.

Neither the image nor the message that has been hidden will be at any moment transmitted over the web, all the magic happens within your browser.

選擇檔案 noble_images.png

Decode

Hidden message

CTF{I love noble
wilds}

Input



使用的網站：<https://www.aperisolve.com/>

- 使用網站解密解密隱藏在片中的文字 / 照片



【校歌】

flag{love_tfg_song}

江學珠 詞
蓋面化曲

The musical score consists of three staves of music. The first staff starts with a treble clef, a key signature of one sharp (F#), and a common time signature. It contains lyrics in Chinese: "維我女校寶島名高華華學子公誠勤毅校訓孔昭齊家治國". The second staff begins with a repeat sign and contains lyrics: "志氣凌霄一肩雙挑". The third staff continues with lyrics: "修養健全人格具備科學頭腦為國家盡至忠為民族盡大孝". The fourth staff concludes with lyrics: "力行三民主義實踐國父遺教繼往開來為我女界增光耀".

Strings

```
A7vJ
c[7C
yQE?
p:K,
a`-----PAT
ps)G
=r@t-
a`-----PAT
"+?G8DAs
p8EvN
s0000000
100000
-{&wt
IEND
flag{i_love_tfg}
```

使用的網站: <https://www.aperisolve.com/>

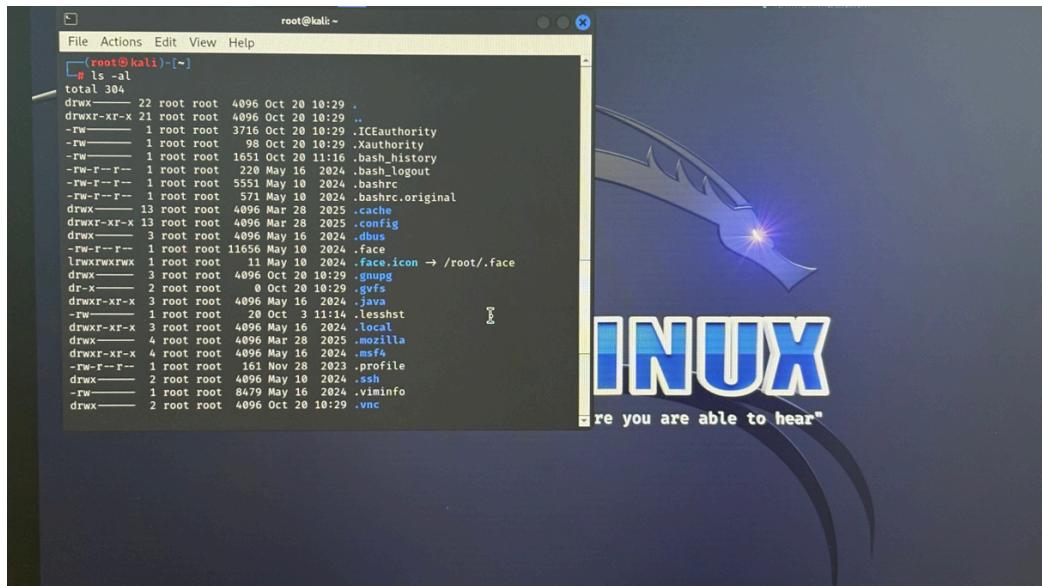
10/20

Linux 簡介

- 使用 CDX Linux 模擬環境

The screenshot shows the CDX Linux simulation environment. At the top, there's a navigation bar with links like '最新消息', 'CDX簡介', '平臺環境', '課程資訊', '活動資訊', '檔案下載', '功能選單', and a user icon. Below the navigation bar is a '機器列表' (Machine List) section with a search bar and buttons for '更新狀態' (Update Status), '新增機器' (Add Machine), '此次功能' (This Function), and '搜尋課程機器' (Search Course Machine). A table lists machines, including 'test20@gappa.fg.tpe.edu.tw' and 'fliuang_gappa.fg.tpe.edu.tw'. One machine is highlighted with a red border, showing details like '機器ID : 19314', 'IP : [10.99.221.24]', and 'RUNNING'. The status bar at the bottom indicates '確定開始時間 : 2025-10-20 23:59:59'.

- 練習各種指令



△練習 「ls -al」

A terminal window titled 'root@kali: ~' shows the output of the 'ifconfig' command, which displays network interface statistics for 'eth0' and 'lo'. It includes information like MTU, broadcast address, and error counts. Below it, the 'hostname -s' command is run, showing the host name 'kali'.

△練習 「hostname -s」

使用的網站：<https://cdx.nchc.org.tw>

- 使用picoCTF解題

▽題目

The screenshot shows a challenge page for 'Obedient Cat'. At the top, there's a user profile for 'Obedient Cat' with a blue icon. Below the profile are three category buttons: 'Easy', 'General Skills', and 'picoCTF 2021'. To the right are a user icon and a close button. Underneath the buttons, it says 'AUTHOR: SYREAL'. On the right side, there's a 'Hints ?' button with three numbered options (1, 2, 3). A green banner at the bottom left says 'Congratulations! You've solved this challenge!' and 'Assignment: CTF_Test (114.1.61)'. Below the banner, the challenge description reads: 'This file has a flag in plain sight (aka "in-the-clear"). Download flag.' A progress bar indicates '320,308 users solved' and '91% Liked'. At the bottom, there's a text input field containing 'picoCTF{s4n1ty_v3r1f13d_f28ac910}' and a blue 'Submit Flag' button.

▽利用linux解出答案

```
tfg11431037-picoctf@webshell:~$ wget https://mercury.picoctf.net/static/2d24d50b4ebcd90c704575627f1f57b2/flag
--2025-10-20 04:03:26-- https://mercury.picoctf.net/static/2d24d50b4ebcd90c704575627f1f57b2/flag
Resolving mercury.picoctf.net (mercury.picoctf.net)... 18.189.209.142
Connecting to mercury.picoctf.net (mercury.picoctf.net)|18.189.209.142|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 34 [application/octet-stream]
Saving to: 'flag'

flag                                         100%[=====] 34 --.- MB/s

2025-10-20 04:03:26 (32.0 MB/s) - 'flag' saved [34/34]

tfg11431037-picoctf@webshell:~$ cat flag
picoCTF{s4n1ty_v3r1f13d_f28ac910}
```

10/27

參加
高中跨校資安實務
素養體驗賽 - 個人賽

參賽！

ISIP HS Cybersecurity Challenges

Notifications Profile Settings

高中跨校資安實務素養體驗賽 個人賽題目

編碼與解碼

PicoCTF解題 - Vigenere

30

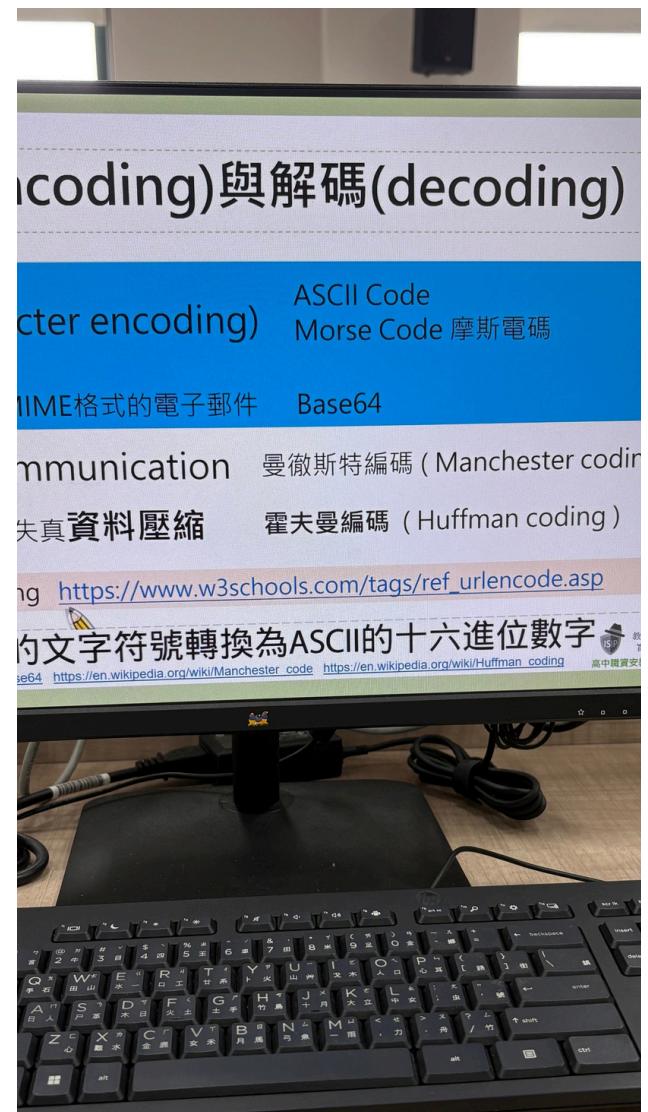
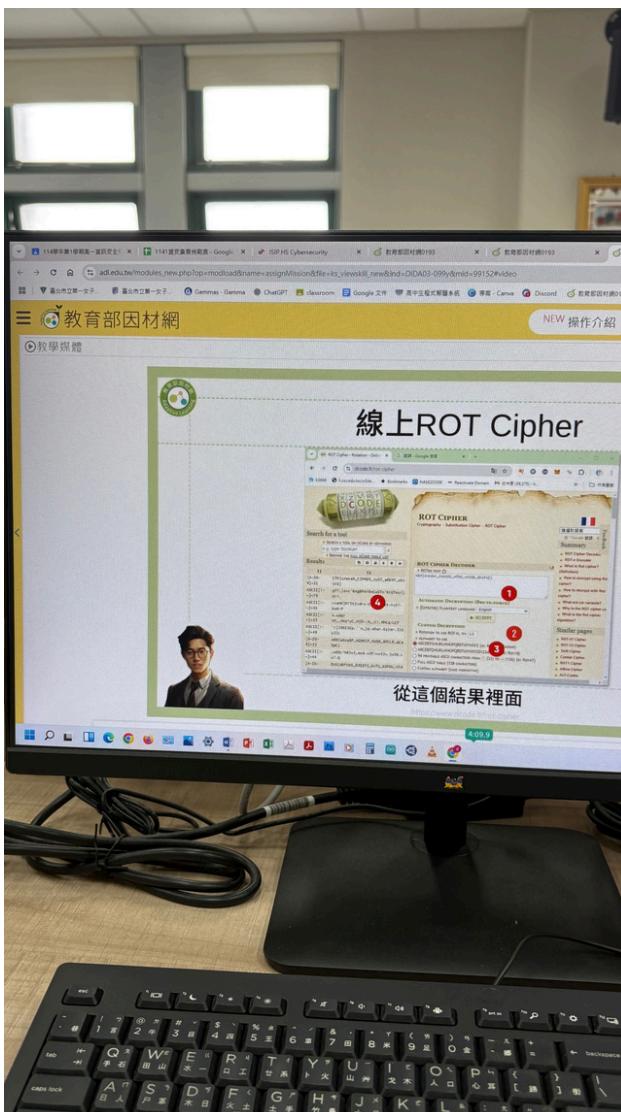
ABCTF - 凱薩加密法

30

RC3 - 凱薩加密進階

40

-利用因才網參賽的自學資源影片進行學習



- 解題



dCode is preparing a new interface. Come test and give your feedback on the new page: [Caesar Cipher!](#)

CAESAR CIPHER

Cryptography • Substitution Cipher • Caesar Cipher

CAESAR CIPHER DECODER

CAESAR SHIFTED CIPHERTEXT
xyzqc{t3_qelrdeq_t3_k33a3a_lk3_lc_qe3p3}

Test all possible shifts (26-letter alphabet A-Z)

► DECRYPT (BRUTEFORCE)

MANUAL DECRYPTION AND PARAMETERS

SHIFT/KEY (NUMBER): 3

USE THE ENGLISH ALPHABET (26 LETTERS FROM A TO Z)

USE THE ENGLISH ALPHABET AND ALSO SHIFT THE DIGITS 0-9

USE THE LATIN ALPHABET IN THE TIME OF CAESAR (23 LETTERS, NO J, U OR W)

USE THE ASCII TABLE (0-127) AS ALPHABET

USE A CUSTOM ALPHABET (A-ZD-9 CHARS ONLY) 0123456789ABCDEGHJKLMNOPQRSTUVWXYZ

► DECRYPT

See also: ROT Cipher – Shift Cipher

Summary

- Caesar Cipher Decoder
- Caesar Encoder
- What is the Caesar cipher? (Definition)
- How to encrypt using Caesar cipher?
- How to decrypt Caesar cipher?
- How to recognize Caesar ciphertext?
- How to decipher Caesar without knowing the shift?
- Is the Caesar Cipher secure?
- What are the variants of the Caesar cipher?
- How to encrypt digits and numbers using Caesar cipher?
- Why the name Caesar Cipher?
- What is August Cipher?
- What are other Caesar Cipher names?
- How to cipher CAESAR with the Caesar code?
- How to write Caesar Cipher in pseudo-code?

答案

按下去

Challenge

PicoCTF解題 - Vigenere

30

描述 你能解密這則訊息嗎？請使用此密鑄「CYLAB」解密這則訊息。

rgnoDVD{00NU_WQ3_G1G3O3T3_A1AH3S_2951c89f}

線索 : vigenere-cipher

_US3_V1G3N3R3_C1PH3R_2951a89h

Submit

VIGENÈRE DECODER

VIGENÈRE CIPHERTEXT
rgnoDVD{00NU_WQ3_G1G3O3T3_A1AH3S_2951c89f}

PLAINTEXT LANGUAGE English

ALPHABET ABCDEFGHIJKLMNOPQRSTUVWXYZ

► AUTOMATIC DECRYPTION

使用題目中的金鑰解密

KNOWING THE KEY/PASSWORD: CYLAB

KNOWING THE KEY-LENGTH/SIZE, NUMBER OF LETTERS: 3

KNOWING ONLY A PARTIAL KEY (JOKER=?): KE?

KNOWING A PLAINTEXT WORD: CODE

SHOW VIGENÈRE'S SQUARE/GRID (TABULA RECTA)

► DECRYPT

See also: Autoclave Cipher – Beaufort Cipher – Caesar Cipher

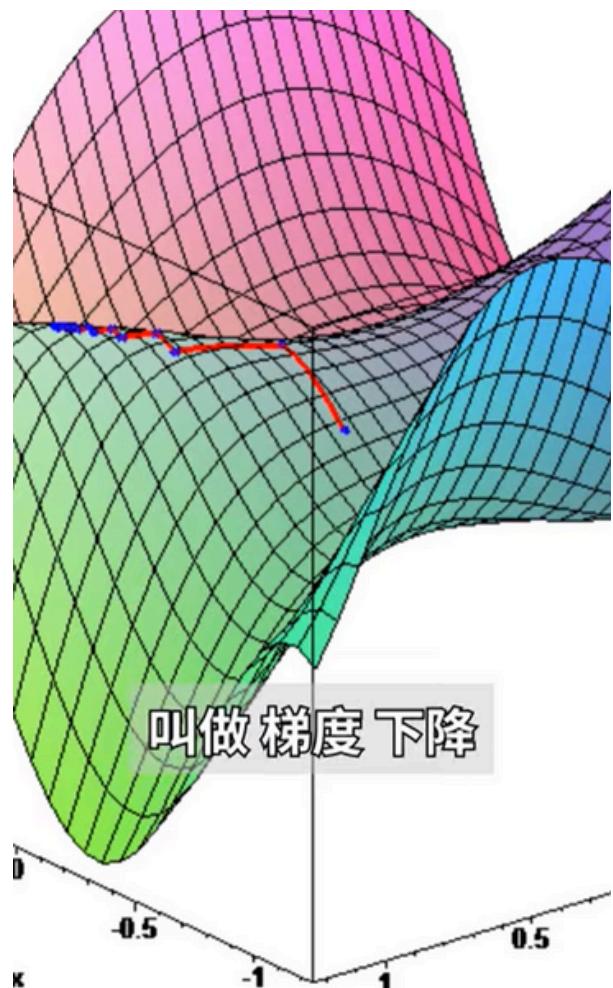
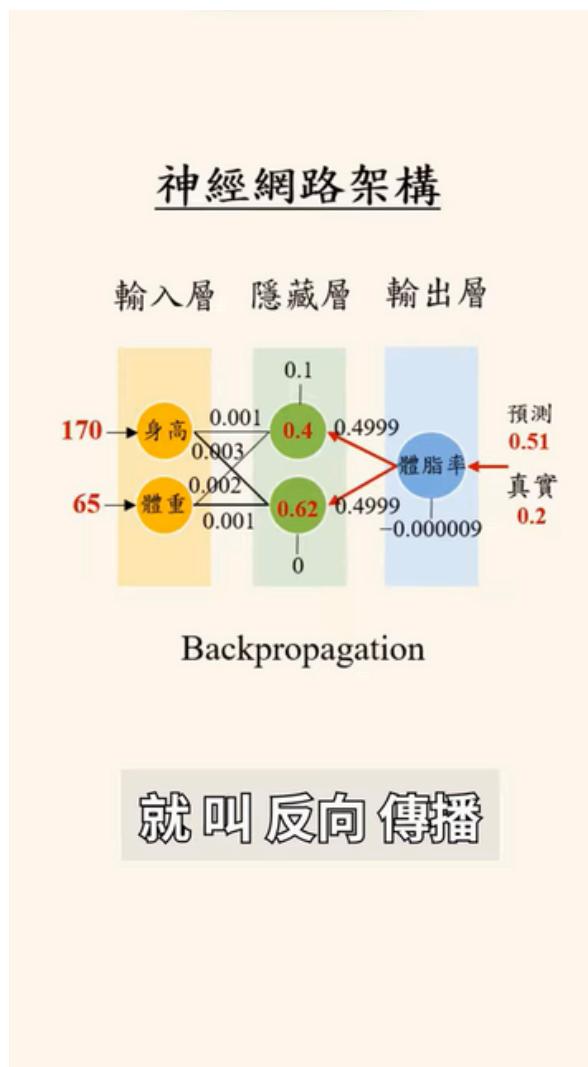
encryption key?

- What is the running key vigenere cipher?
- What is the keyed vigenere cipher?
- What are the advantages of the Vigenere cipher versus Caesar Cipher?

11/03

人工智慧AI簡介

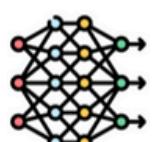
-看老師製作的YT影片學習人工智慧大概架構



影片連結：https://www.youtube.com/watch?v=5U2I2LTIY5M&list=PLUF1beASyXmfqQOctNEfVT3BVyV_3eo-E

-授課老師教學簡報

經典的神經網路



神經網路 Neural Network

最為經典的神經網路，又被稱為**全連結神經網路**，可以被應用到不同的任務上。

卷積神經網路 Convolution Neural Network

被廣泛應用在**圖像**和**語音**相關任務的一種神經網路，例如物體偵測、人臉辨識等...

遞歸神經網路 Recurrent Neural Network

在運作時會**保存先前的信息**，因此它可以考慮**上下文的資訊**，所以也被稱為「**有記憶**」的神經網路。

- 實作 — 訓練模型

請 Chat GPT 幫忙寫訓練程式

請用 MNIST 資料集寫一個卷積神經網路 CNN 的訓練程式

當然可以 !

以下是一個使用 TensorFlow / Keras 建立並訓練 CNN 的完整範例，用於 MNIST 手寫數字辨識。

這個程式會：

1. 載入 MNIST 資料集
2. 建立卷積神經網路 (CNN) 模型
3. 編譯與訓練模型
4. 評估模型準確率

■ 程式碼範例

python

複製程式碼

```
# 匯入必要的套件
import tensorflow as tf
from tensorflow.keras import layers, models
from tensorflow.keras.datasets import mnist
from tensorflow.keras.utils import to_categorical

# 1 載入 MNIST 資料集
(x_train, y_train), (x_test, y_test) = mnist.load_data()
```

最好畫出 ACCURACY 正確率跟 Epoch 關係圖 以及 loss 和 Epoch 的關係圖

非常好 !

我們可以在訓練後使用 `matplotlib` 將 準確率 (accuracy) 與 損失 (loss) 隨著訓練 epoch 的變化畫出圖表。

以下是完整程式 (包含畫圖部分)：

🧠 CNN + 訓練曲線 (MNIST)

python

複製程式碼

```
# 匯入必要的套件
import tensorflow as tf
from tensorflow.keras import layers, models
from tensorflow.keras.datasets import mnist
from tensorflow.keras.utils import to_categorical
import matplotlib.pyplot as plt

# 1 載入 MNIST 資料集
(x_train, y_train), (x_test, y_test) = mnist.load_data()

# 2 資料預處理
x_train = x_train.reshape((x_train.shape[0], 28, 28, 1)).astype('float32') / 255
```

實作結果

```

Epoch 1/10
844/844 49s 55ms/step - accuracy: 0.8516 - loss: 0.4688 - val_accuracy: 0.9853 - val_loss: 0.0506
Epoch 2/10
844/844 47s 55ms/step - accuracy: 0.9818 - loss: 0.0583 - val_accuracy: 0.9893 - val_loss: 0.0387
Epoch 3/10
844/844 45s 54ms/step - accuracy: 0.9886 - loss: 0.0370 - val_accuracy: 0.9887 - val_loss: 0.0398
Epoch 4/10
844/844 47s 56ms/step - accuracy: 0.9922 - loss: 0.0279 - val_accuracy: 0.9885 - val_loss: 0.0403
Epoch 5/10
844/844 81s 55ms/step - accuracy: 0.9924 - loss: 0.0223 - val_accuracy: 0.9883 - val_loss: 0.0384
Epoch 6/10
844/844 82s 55ms/step - accuracy: 0.9941 - loss: 0.0182 - val_accuracy: 0.9895 - val_loss: 0.0356
Epoch 7/10
844/844 45s 53ms/step - accuracy: 0.9954 - loss: 0.0152 - val_accuracy: 0.9897 - val_loss: 0.0367
Epoch 8/10
844/844 46s 55ms/step - accuracy: 0.9962 - loss: 0.0112 - val_accuracy: 0.9920 - val_loss: 0.0304
Epoch 9/10
844/844 82s 54ms/step - accuracy: 0.9974 - loss: 0.0092 - val_accuracy: 0.9905 - val_loss: 0.0421
Epoch 10/10
844/844 81s 53ms/step - accuracy: 0.9969 - loss: 0.0092 - val_accuracy: 0.9903 - val_loss: 0.0460

```

正確率隨訓練次數增加

失誤率隨訓練次數減少

● # 匯入必要的套件

```

import tensorflow as tf
from tensorflow.keras import layers, models
from tensorflow.keras.datasets import mnist
from tensorflow.keras.utils import to_categorical
import matplotlib.pyplot as plt

```

1 載入 MNIST 資料集

```
(x_train, y_train), (x_test, y_test) = mnist.load_data()
```

2 資料預處理

```
x_train = x_train.reshape((x_train.shape[0], 28, 28, 1)).astype('float32') / 255
x_test = x_test.reshape((x_test.shape[0], 28, 28, 1)).astype('float32') / 255
```

```
y_train = to_categorical(y_train, 10)
y_test = to_categorical(y_test, 10)
```

3 建立 CNN 機型

```
model = models.Sequential([
    layers.Conv2D(32, (3, 3), activation='relu', input_shape=(28, 28, 1)),
    layers.MaxPooling2D((2, 2)),
    layers.Conv2D(64, (3, 3), activation='relu'),
    layers.MaxPooling2D((2, 2)),
    layers.Conv2D(64, (3, 3), activation='relu'),
    layers.Flatten(),
    layers.Dense(64, activation='relu'),
    layers.Dense(10, activation='softmax')
])
```

顯示模型架構

```
model.summary()
```

4 編譯模型

```
model.compile(optimizer='adam',
              loss='categorical_crossentropy',
              metrics=['accuracy'])
```

5 訓練模型

```
history = model.fit(x_train, y_train, epochs=10, batch_size=64,
                     validation_split=0.1, verbose=1)
```

6 評估模型

```
test_loss, test_acc = model.evaluate(x_test, y_test, verbose=0)
print(f'\n測試集準確率: {test_acc * 100:.2f}%')
```

7 繪製訓練曲線

```
plt.figure(figsize=(12, 5))
```

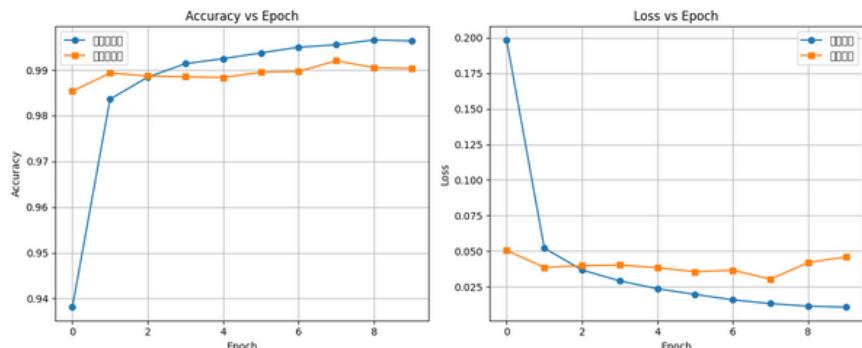
左圖: Accuracy vs Epoch

```
plt.subplot(1, 2, 1)
plt.plot(history.history['accuracy'], label='訓練準確率', marker='o')
plt.plot(history.history['val_accuracy'], label='驗證準確率', marker='s')
plt.title('Accuracy vs Epoch')
plt.xlabel('Epoch')
plt.ylabel('Accuracy')
plt.legend()
plt.grid(True)
```

右圖: Loss vs Epoch

```
plt.subplot(1, 2, 2)
plt.plot(history.history['loss'], label='訓練損失', marker='o')
plt.plot(history.history['val_loss'], label='驗證損失', marker='s')
plt.title('Loss vs Epoch')
plt.xlabel('Epoch')
plt.ylabel('Loss')
plt.legend()
plt.grid(True)
```

```
plt.tight_layout()
plt.show()
```



Colab 程式網址:

https://colab.research.google.com/drive/1HX9CJC9GJB1IJ_yCiCFB8hc4X7sIAKK#scrollTo=NvxUTM6LwN5R

11/10

參加
高中跨校資安實務
素養體驗賽 - 團體賽



上一頁 Alt + 向左鍵

下一頁 Alt + 向右鍵

重新載入 Ctrl + R

另存新檔... Ctrl + S

列印... Ctrl + P

投放...

透過 Google 智慧鏡頭搜尋

以閱讀模式開啟

翻譯成中文 (繁體)

從 Google 取得圖片說明 >

按下 檢視網頁原始碼 Ctrl + U

檢查

```
33         padding: 10px 15px;
34         cursor: pointer;
35         border-radius: 3px;
36     }
37     input[type="submit"]:hover {
38         background-color: #45a049;
39     }
40 </style>
41 </head>
42 <body>
43
44 <form action="/submit" method="post">
45     <h2>登入</h2>
46     <label for="username">帳號:</label>
47     <input type="text" id="username" name="username" required>
48
49     <label for="password">密碼:</label>
50     <input type="password" id="password" name="password" required>
51
52     <input type="submit" value="送出">
53
54         <! -- admin/V@1ue230 --> 答案!
55 </form>
56     <p hidden> -- admin/V@1ue230 </p>
57 <script defer src="https://static.cloudflareinsights.com/beacon.min.js/vcd15cbe7772f49c399c6a5babf22c124171
58 </body>
59
```

Challenge

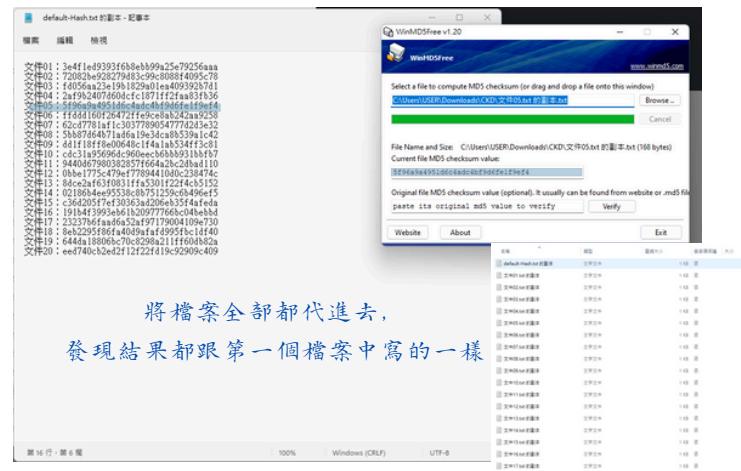
魚目混珠

20

資訊人員Crystal發現公司文件中的文件被駭客篡改。請您幫忙找出那些文件是被篡改過？

Key 格式：CTF{}

WinMD5Free應用程式 線索：WinMD5Free

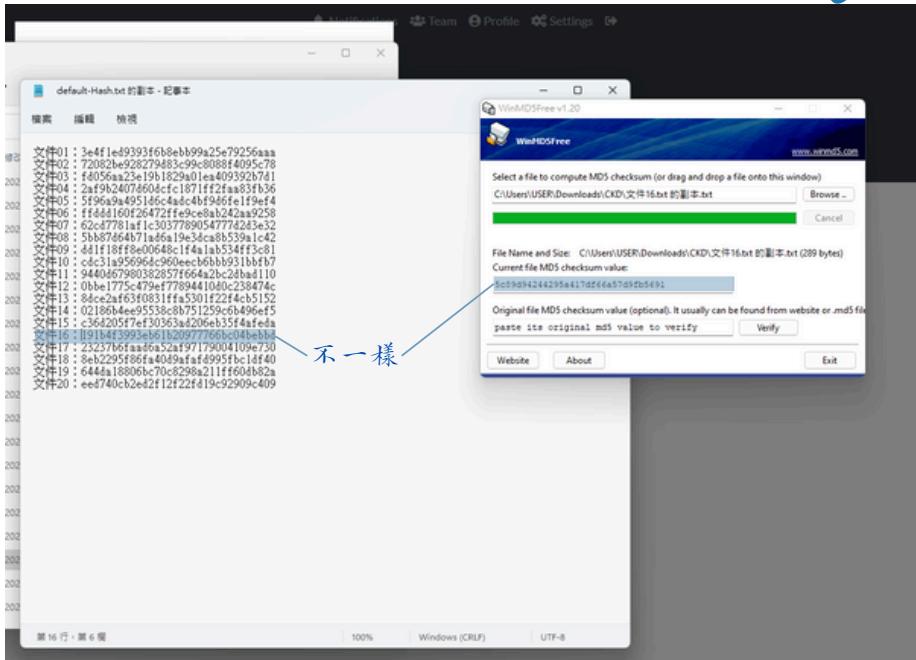


將檔案全部都代進去,

發現結果都跟第一個檔案中寫的一樣

第 16 行, 第 6 檔 100% Windows (CR/LF) UTF-8

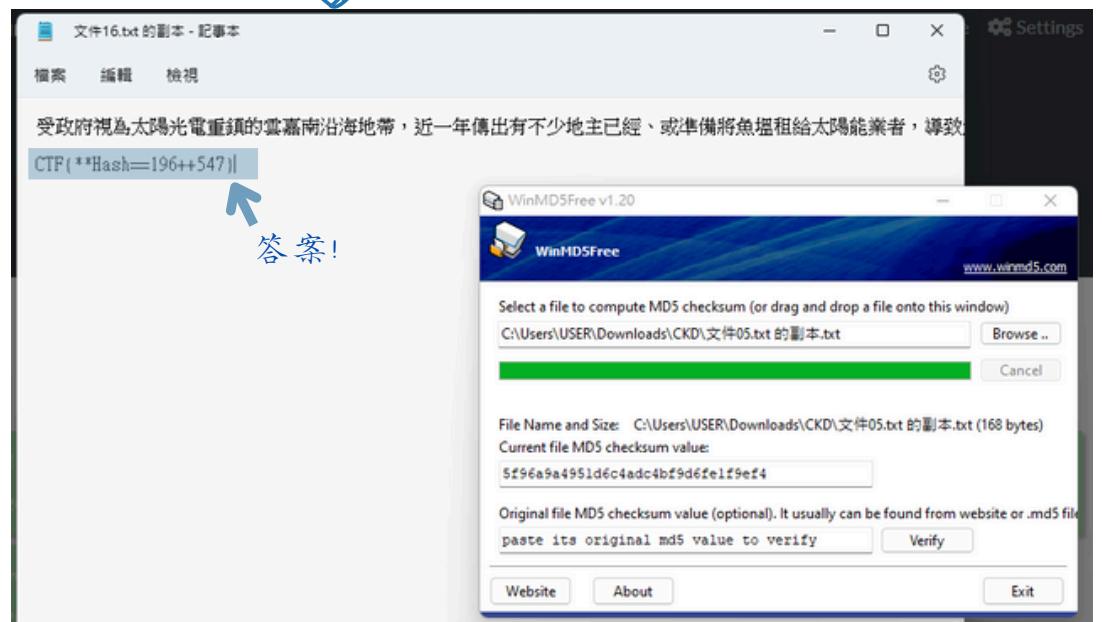
到文件16的時候發現不一樣



不一樣



打開看看，發現...



答案!

Challenge

世外桃源

20

描述：請問原始檔案的第一行文字為何？

提示：

1. 眼見不為憑
2. 線上工具：[hexed.it](#)

Key格式：CTF{}



Flag

Submit

打開題目給的連結、放入題目給的檔案

解出來 "7z" → 壓縮檔

The screenshot shows a hex editor interface with a large amount of binary data. The first few bytes are visible:

```
00000000: DB_1.pptx 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00000010: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00000018: E9 03 DA 01 C4 00 00 74 20 CC EE 04 BC 0A 1F FF 13
00000020: CD A6 53 A8 37 77 17 CC BE 09 2B 0A 82 1A 46
00000028: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00000030: E2 9C 98 1E F2 10 18 EE 09 A2 F8 C2 36 12 47 AF
00000038: C4 3A 35 C9 4C AF 36 F1 30 53 56 C9 58 57 A1 5D
00000040: 56 88 3F 0A 79 77 B2 56 C8 0E 01 82 97 40 EF
00000048: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00000050: 24 5A 55 C9 4C F2 AD 87 A4 05 50 63 D0 E3 98
00000058: F0 A2 88 80 BF 2A 42 79 5D 02 71 0D 7F CA TB A2
00000060: 57 55 18 C4 31 B3 00 00 00 00 00 00 00 00 00 00
00000068: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00000076: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00000084: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00000092: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0000009A: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
000000A8: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
000000B6: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
000000C4: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
000000D2: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
000000E0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
000000F8: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00000106: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00000114: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00000122: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00000130: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00000138: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00000146: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00000154: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00000162: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00000170: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00000178: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00000186: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00000194: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
000001A2: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
000001B0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
000001B8: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
000001C6: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
000001D4: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
000001E2: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
000001F0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
000001F8: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00000206: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00000214: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00000222: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00000230: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00000238: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00000246: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
```

題目給的網站資源：<https://hexed.it/>

答案！

The screenshot shows a Windows File Explorer window with a folder named '解壓縮' containing 15 items. A blue arrow points from the text '解壓縮' to the folder icon.

The folder contents are:

- 2025/11/10 上午 11:23 Microsoft PowerPoint 檔案
- CXD-20251008T040406Z-1-001 (1)
- Mayan_numbers (2)
- winnie-the-poohsteg (1).steg
- winnie-the-poohsteg (1)
- winnie-the-poohsteg (1).steg
- winnie_the_poohsteg (1)
- Mayan_numbers (1)
- Mayan_numbers
- DB_1.pptx
- CXD-20251008T040406Z-1-001
- winmd5free
- winnie-the-poohsteg
- index2
- CXD
- winmd5free

A blue arrow points from the text 'DB_1.pptx' to the file icon in the list.

The hex editor shows the first few bytes of the DB_1.pptx file:

```
00000000: DB_1.pptx 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00000010: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00000018: E9 03 DA 01 C4 00 00 74 20 CC EE 04 BC 0A 1F FF 13
00000020: CD A6 53 A8 37 77 17 CC BE 09 2B 0A 82 1A 46
00000028: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00000030: E2 9C 98 1E F2 10 18 EE 09 A2 F8 C2 36 12 47 AF
00000038: C4 3A 35 C9 4C AF 36 F1 30 53 56 C9 58 57 A1 5D
00000040: 56 88 3F 0A 79 77 B2 56 C8 0E 01 82 97 40 EF
00000048: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00000050: 24 5A 55 C9 4C F2 AD 87 A4 05 50 63 D0 E3 98
00000058: F0 A2 88 80 BF 2A 42 79 5D 02 71 0D 7F CA TB A2
00000060: 57 55 18 C4 31 B3 00 00 00 00 00 00 00 00 00 00
00000068: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00000076: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00000084: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00000092: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0000009A: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
000000A8: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
000000B6: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
000000C4: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
000000D2: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
000000E0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
000000F8: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00000106: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00000114: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00000122: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00000130: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00000138: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00000146: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00000154: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00000162: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00000170: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00000178: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00000186: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00000194: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
000001A2: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
000001B0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
000001B8: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
000001C6: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
000001D4: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
000001E2: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
000001F0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
000001F8: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00000206: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00000214: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00000222: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00000230: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00000238: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00000246: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
```

The Notepad window shows the text:

資料庫的核心理論與實務7/e

<https://www.books.com.tw/products/0010779579>

A blue arrow points from the text '資料庫的核心理論與實務7/e' to the URL.

The browser window shows the URL:

https://www.books.com.tw/products/0010779579

A blue arrow points from the URL to the browser window.

Challenge

馬雅數字

20

請運用馬雅數字建立Key，再用Vigenère傳統式加密法解密文字。

提示

1. 先請開啟英文字母順序表網頁 [abc.html](#)
2. 參考附圖馬雅數字對應到英文字母找出解密鑰匙，馬雅數字為20進位
3. 加密鑰匙英文字母為由左至右。對應到馬雅數字為由上至下，字母均為小寫。
4. 運用Vigenère傳統式加密法線上工具。
5. 使用解密鑰匙將密文 `cmzvwiuiuxadlhvetztxreapmzl` 轉換成明文。

線索: [vigenere.html](#)

Key格式：CTF{明文}

Mayan_num...

Flag

Submit

The screenshot shows a dark-themed browser window with several elements:

- A large blue number "20" is displayed prominently.
- Below it are other large blue numbers: "5", "13", "16", "12", and "5".
- To the right is a grid of Mayan numerals from 0 to 20, each with its corresponding value and a small eye icon.
- At the bottom, there's a table mapping Mayan symbols to letters and phonetic values, and another table mapping letters to phonetic values.
- On the far right, there's a sidebar with navigation buttons and a search bar.

→ 對照後的解密鑰匙: temple

Text To Encode

答案!

Text To Decode

KEY = TEMPLE

[Random Key](#) [Encode](#) [Decode](#) [Remove Spaces](#) [Add Spaces](#) 4

Message:
• CMZVWIUIUXADLHVENTZTXREAPMZL

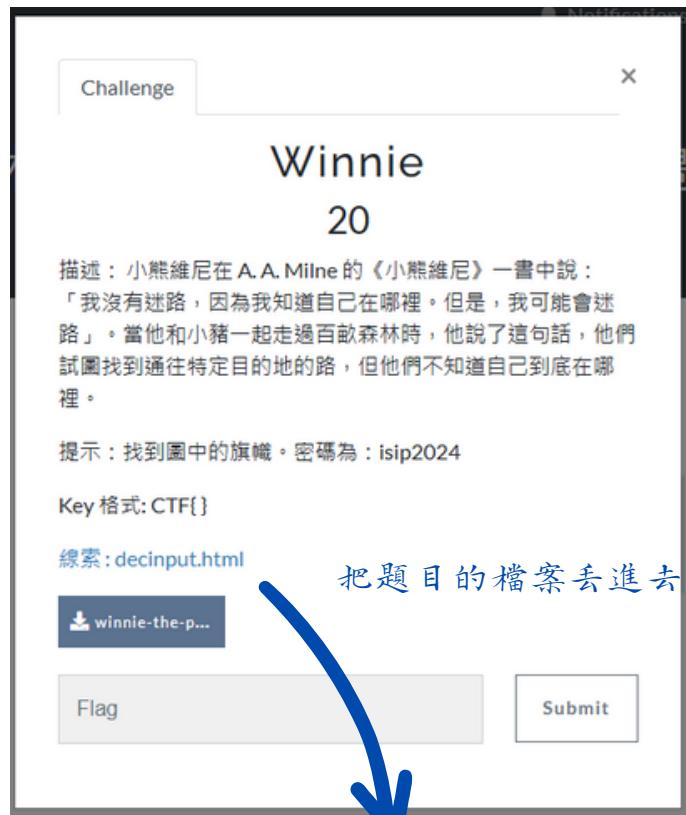
Keys: TEMPLE

7 22 14 11 15 22

AS:

• jinglebellshorseopensleigh

題目給的網站資源: <https://www.ifreesite.com/abc/abc.html>



把題目的檔案丟進去

winnie-the-poohsteg.jpg

Flag

Submit

Steganographic Decoder

This form decodes the payload that was hidden in a JPEG image or a WAV or AU audio file using the [encoder form](#). When you submit, you will be asked to save the resulting payload file to disk. This form may also help you guess at what the payload is and its file type...

Select a JPEG, WAV, or AU file to decode:

選擇檔案: winnie-the-poohsteg.jpg

Password (may be blank):

isip2024

View raw output as MIME-type `text/plain`

Guess the payload

Prompt to save (you must guess the file type yourself.)

提交

To use this form, you must first [encode a file](#).

These pages use the [steghide](#) program to perform steganography, and the files generated are fully compatible with steghide.

Please send comments or questions to [Alan Eliasen](#).

[Back to Alan's Home Server](#)

題目給的網站: <https://futureboy.us/stegano/decinput.html>

↓ 解出來的東西

Q1RGelRoZSBNYW55IEFkdmVudHVyZXMb2YgV2lubmllIHRoZSBQb29ofQ==

↓ 拿去給Base64解

Utilities Online

Base64 encode and base64 decode

Encode or decode data in Base64 format, a widely used method for representing binary data in a text format.

Base64 Encode Decode

Result

Q1RGelRoZSBNYW55IEFkdmVudHVyZXMb2YgV2lubmllIHRoZSBQb29ofQ==

CTF(The Many Adventures of Winnie the Pooh)

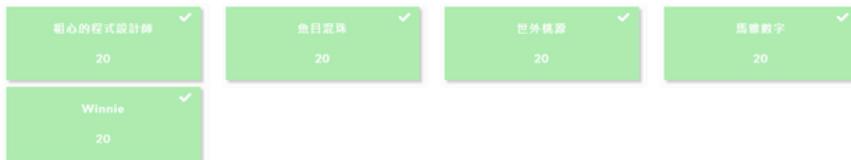
Encode Decode

答案!

網站: <https://www.utilities-online.info/base64>

高中跨校資安實務素養體驗賽 團體賽題目

數位鑑識



全部完成！

心得感想

在這次的資安挑戰賽後，我覺得自己學到了很多。[過去相較於團體賽，我更喜歡個人賽](#)。對我來說，憑自己的實力比賽，輸了也能心服口服；但若與他人一組，可能會遇到有人「躺分」或拖慢進度，讓我覺得不太公平。

然而，這次的比賽改變了我的想法。在解題的過程中，我體會到[團隊合作](#)的樂趣：題目解不出來時一起苦惱，解出來後互相分享思路、交流技巧，不僅讓彼此進步，也能學習不同的解題方法。即使有組員最後沒有成功解出題目，他們在過程中依然努力思考，只是方向可能與正解不同而已。那份共同努力、彼此鼓勵的過程，讓我深刻感受到[團隊精神](#)的重要以及[和他人合作的快樂](#)。

另外，這次比賽的分組是由老師隨機安排，我與組員們在合作前都不認識。和陌生人一起解題、討論的過程讓我覺得很新奇有趣。相較於個人賽的單打獨鬥，團體賽讓我體驗到[更多元的想法與合作的成就感](#)。

此外，當我成功解出題目，送出答案後看到螢幕上出現“**Correct**”的那一刻讓我非常感動。老實說，在課堂學習時，我曾覺得資安解題似乎只是把密文丟進網站轉換，感覺自己沒有學到什麼。但經過這次比賽我才發現，[工具只是輔助](#)，真正重要的是能否判斷題目的方向與選擇合適的方法。如果一開始抓錯重點、用錯網站，就算嘗試再多次也得不到結果。當我經過多次嘗試、終於成功解出答案時，那種[成就感](#)是前所未有的。

這次資安挑戰賽讓我重新認識了團隊合作，也讓我真正體會到「[自己動腦解出問題](#)」的樂趣。它不僅是一次比賽，更是一場讓我成長、學會信任夥伴的寶貴經驗。

11/17

模型逆向攻擊

MNIST 訓練 模型

- 利用Chat GPT寫出程式碼於google colab執行

請幫我做模型逆向攻擊MNIST訓練模型的colab程式碼



ChatGPT 上午10:28

下面給你 可直接貼進 Google Colab 執行 的完整程式碼範例，示範：

在 MNIST 訓練好的模型上

- 建立模型(簡單 CNN)

- 進行模型逆向攻擊 (Model Inversion Attack)：

透過梯度上升，從特定標籤反推出讓模型信心最高的「代表性影像」。



Colab：模型逆向攻擊 (Model Inversion) MNIST 範例

你只要整段貼到 Colab 執行即可。

python

```
# -----  
# 1. 安裝/匯入套件  
# -----  
import torch  
import torch.nn as nn  
import torch.optim as optim  
import torch.nn.functional as F  
from torchvision import datasets, transforms  
import matplotlib.pyplot as plt
```

```
device = torch.device("cuda" if torch.cuda.is_available() else "cpu")  
print("Using device:", device)
```

```
# -----  
# 2. 建立簡單 CNN 模型  
# -----
```

+ 傳送訊息或 @ChatGPT

ChatGPT 可能會出現「諸多錯誤」，但絕對該編不會使用你的個人 ChatGPT 記憶。



```
[3] 1分鐘  
train_dataset = datasets.MNIST(root='./data', train=True, download=True, transform=transform)  
train_loader = torch.utils.data.DataLoader(train_dataset, batch_size=64, shuffle=True)  
  
# -----  
# 4. 訓練模型  
# -----  
optimizer = optim.Adam(model.parameters(), lr=0.001)  
criterion = nn.CrossEntropyLoss()  
  
epochs = 2 # 為了快速示範，只跑 2 epoch  
  
print("Training...")  
for epoch in range(epochs):  
    for data, target in train_loader:  
        data, target = data.to(device), target.to(device)  
        optimizer.zero_grad()  
        output = model(data)  
        loss = criterion(output, target)  
        loss.backward()  
        optimizer.step()  
    print(f"Epoch {epoch+1}/{epochs} Done")  
  
# -----  
# 5. 模型逆向攻擊 (Model Inversion)  
#     對特定標籤，反推出最佳代表影像  
# -----  
def model_inversion(model, target_label, steps=300, lr=0.1):  
    model.eval()  
  
    # 從隨機噪音開始  
    img = torch.randn((1, 1, 28, 28), device=device, requires_grad=True)  
  
    optimizer = optim.Adam([img], lr=lr)  
  
    target_tensor = torch.tensor([target_label], device=device)  
  
    for i in range(steps):  
        optimizer.zero_grad()  
        ...
```

*** Using device: cpu

100%	9.91M/9.91M [00:00<00:00, 38.2MB/s]
100%	28.9k/28.9k [00:00<00:00, 1.01MB/s]
100%	1.65M/1.65M [00:00<00:00, 10.0MB/s]
100%	4.54k/4.54k [00:00<00:00, 6.89MB/s]

Training...

Epoch 1/2 Done

Epoch 2/2 Done

Running model inversion for label 7 ...

Model inversion result for label 7



11/24

SQL Injection

學習與實作

● 使用教學簡報學習

-SQL Injection 簡介

老師上課簡報

什麼是 SQL Injection ?

- 一種駭客攻擊手法，藉由輸入惡意 SQL 指令竄改資料庫查詢語句。
- 可能導致：
 - 登入繞過驗證
 - 瀏覽/竄改/刪除資料
 - 控制整個資料庫

▽ 簡單語法舉例與能運用在生活中保護自己的個資的部分

簡單例子（登入繞過）

- 假設有登入語法如下：

```
SELECT * FROM users WHERE username='$user'  
AND password='$pass';
```

- 使用者輸入：

username: ' OR '1'='1';/*

password: */ -- (可輸入任意或不填)

如何防範？

- 永遠不要直接把使用者輸入放進 SQL
- 例如：使用參數化查詢 (Prepared Statements)

```
$stmt = $pdo->prepare("SELECT * FROM users  
WHERE username = ? AND password = ?");
```

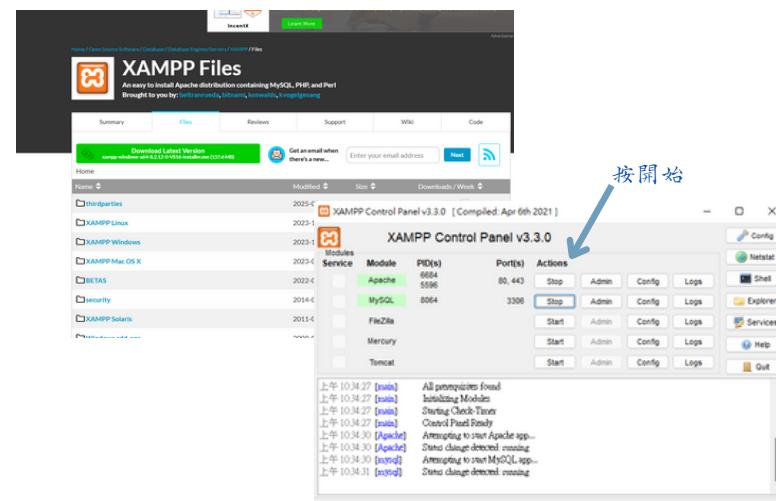
先寫好一段查詢資料庫的SQL語法草稿，但先空下兩個位置
(用?表示)，等等再把真正的帳號密碼填進去。

```
$stmt->execute([$user, $pass]);
```

把使用者輸入的帳號 \$user 和密碼 \$pass 填到剛預留的兩個?裡。

SQL Inject 安全版的登入系統實作

- 使用XAMPP進行SQL Injection的實作



▽ 在MySQL中建立資料表

This screenshot shows the phpMyAdmin interface for the 'login_test' database. In the left sidebar, the 'users' table under the 'login_test' database is selected. The main area displays SQL code for creating the 'users' table and inserting two rows of data ('admin', '1234') and ('test', 'abcd'). A blue arrow points from the text '設置帳密' (Set up account) to the 'password' column in the SQL code. At the bottom, the '执行' (Execute) button is highlighted.

This screenshot shows the results of the SQL query 'SELECT * FROM `users`' in the phpMyAdmin interface. The results table shows two rows: 'admin' with password '1234' and 'test' with password 'abcd'. A blue arrow points from the text '設好了!' (Set up is done!) to the 'test' row in the results table.

login.php - 記事本

檔案 檔案 檔案 檔案 檔案

```

<!DOCTYPE html>
<html>
<head>
    <meta charset="UTF-8">
    <title>Login Page</title>
</head>
<body>
    <h2>登入系統</h2>
    <form action="login.php" method="POST">
        <label>帳號 :</label>
        <input type="text" name="username" required><br><br>
        <label>密碼 :</label>
        <input type="password" name="password" required><br><br>
        <input type="submit" value="登入">
    </form>
</body>
</html>

<?php
$host = "localhost";
$user = "root"; // 如果你有設定密碼請修改
$pass = ""; // XAMPP 預設空白
$dbname = "login_test";

// 連線資料庫
$conn = new mysqli($host, $user, $pass, $dbname);
if ($conn->connect_error) {
    die("資料庫連線失敗：" . $conn->connect_error);
}

// 接收表單資料
$username = $_POST['username'] ?? '';
$password = $_POST['password'] ?? '';

// 使用 Prepared Statement 防止 SQL injection
$stmt = $conn->prepare("SELECT * FROM users WHERE username = ? AND password = ?");
$stmt->bind_param("ss", $username, $password);
$stmt->execute();

$result = $stmt->get_result();

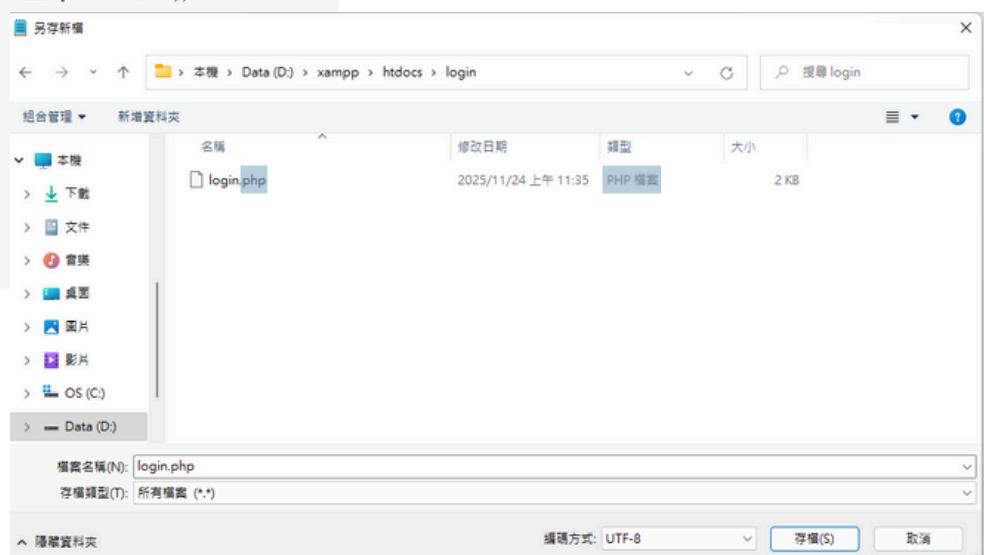
// 判斷帳號
if ($result->num_rows > 0) {
    echo "登入成功！歡迎你，" . htmlspecialchars($username);
} else {
    echo "帳號或密碼錯誤！";
}

$stmt->close();
$conn->close();
?>

```

請
Chat
gpt
幫忙
生成的
程式碼

→ 存在記事本裡 → 儲存成.php檔



在瀏覽器網址列上打上 localhost/login/login.php
成功出現登入表單

依照老師的教學簡報
在瀏覽器中打上網址

在顯示出的網站介面中輸入設定的帳號密碼

登入系統

帳號 :

密碼 :

帳號或密碼錯誤！



登入系統

帳號 :

密碼 :

登入成功！歡迎你，test
成功了！！

SQL Inject不安全版的登入系統實作

```
<?php
$host = "localhost";
$user = "root"; // XAMPP/VAMP 預設帳號
$pass = "";
$dbname = "login_test";

// 連線資料庫
$conn = new mysqli($host, $user, $pass, $dbname);
if ($conn->connect_error) {
    die("資料庫連線失敗：" . $conn->connect_error);
}

// 接收前端表單輸入
$username = $_POST['username'] ?? '';
$password = $_POST['password'] ?? '';

// ✘ 脆弱、不安全的 SQL 拼接（為了教學實驗）
$sql = "SELECT * FROM users WHERE username = '$username' AND password = '$password'";
漏洞在這裡！

// 顯示 SQL（方便你觀察注入效果）
echo "<p><b>執行的 SQL :</b> $sql</p>";

$result = $conn->query($sql);

// 驗證結果
if ($result && $result->num_rows > 0) {
    echo "<h2 style='color: green;'>登入成功！（此頁面可被 SQL Injection 攻破）</h2>";
} else {
    echo "<h2 style='color: red;'>登入失敗（輸入正確帳密或試著做 SQL Injection）</h2>";
}

$conn->close();
?>

<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>Login Page</title>
</head>
<body>
    <h2>登入系統</h2>
    <form action="login.php" method="POST">
        <label>帳號 :</label>
        <input type="text" name="username" required><br><br>

        <label>密碼 :</label>
        <input type="password" name="password" required><br><br>

        <input type="submit" value="登入">
    </form>
</body>
</html>
```

SQL
有
漏
洞
的
程
式
碼

登入失敗（輸入正確帳密或試著做 SQL Injection）

登入系統

帳號：

密碼：

執行的 SQL : SELECT * FROM users WHERE username = 'test' AND password = 'abcd'

登入成功！（此頁面可被 SQL Injection 攻破）

輸入漏洞的帳號密碼也會登入成功 帳號：

密碼：

帳號或密碼錯誤

▼ 與正常的程式碼對照

登入系統

帳號：

密碼：

帳號或密碼錯誤

登入系統

帳號：

密碼：

使用正常的程式碼就不能登入

12/01

封包攔截 --

Burp Suite攔截

利用Burp Suite監控所輸入的密碼

-試著監控看看上禮拜做的網站

The screenshot shows a web browser window titled "Login Page" with the URL "localhost/login/login.php". The page content includes fields for "帳號 : test" and "密碼 :", and a "登入" button. A blue arrow labeled "輸入之後" points from the browser's address bar down to the Burp Suite interface.

Burp Suite Community Edition v2025.10.6 - Temporary Project

Request to http://localhost/login/login.php

Time	Type	Direction	Method	URL
11:49:56 1 D...	HTTP	→ Request	POST	http://localhost/login/login.php

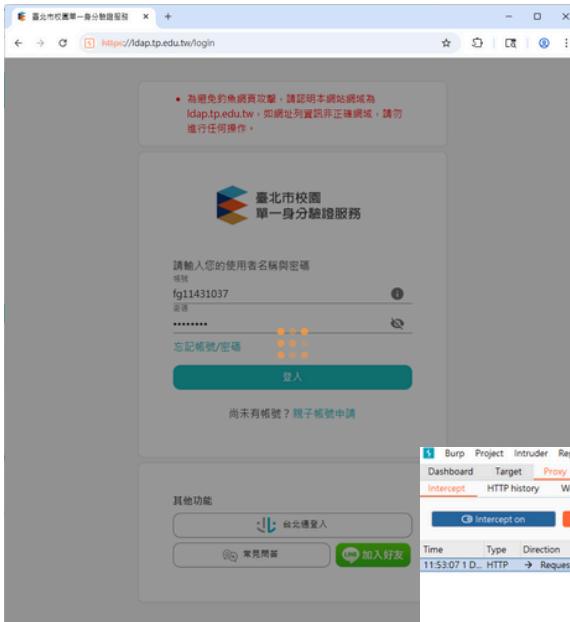
Request

Pretty Raw Hex

```
6 sec-ch-ua-mobile: ?0
7 sec-ch-ua-platform: "Windows"
8 Accept-Language: zh-TW,zh;q=0.9
9 Origin: http://localhost
0 Content-Type: application/x-www-form-urlencoded
1 Upgrade-Insecure-Requests: 1
2 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/142.0.0.0 Safari/537.36
3 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7
4 Sec-Fetch-Site: same-origin
5 Sec-Fetch-Mode: navigate
6 Sec-Fetch-User: ?1
7 Sec-Fetch-Dest: document
8 Referer: http://localhost/login/login.php
9 Accept-Encoding: gzip, deflate, br
0 Connection: keep-alive
1
2 username=test&password=1234
```

刚刚输入的帳號密碼!

-試試二代校務行政系統



Burp Suite Community Edition v2025.10.6 - Temporary Project

Dashboard Target Proxy Intruder Repeater View Help

Intercept HTTP history WebSockets history Match and replace Proxy settings

Request to https://ldap.tp.edu.tw:443 [163.21.244.100] Open browser

Time Type Direction Method URL Status code Length

11:53:07 1 D... HTTP → Request POST https://ldap.tp.edu.tw/web/login/authenticate

Request

Pretty Raw Hex

10 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/142.0.0.0 Safari/517.36

11 Accept: */*

12 Origin: https://ldap.tp.edu.tw

13 Sec-Fetch-Site: same-origin

14 Sec-Fetch-Mode: cors

15 Sec-Fetch-Dest: empty

16 Referer: https://ldap.tp.edu.tw/login

17 Accept-Encoding: gzip, deflate, br

18 Priority: 1

19 Connection: keep-alive

20 Content-Type: application/x-www-form-urlencoded

21 { "account": "fg11431037", "password": "████████", "relayState": null, "samlRequest": null }

Event log All issues

Inspector

Request attributes
Request query parameters
Request cookies
Request headers

Memory: 143.6MB Disabled



我的帳號

"account": "fg11431037",
"password": "████████",
"relayState": null,
"samlRequest": null

OMG居然成功了?!!



周宜帆

學生

帳號: fg11431037

電話: 0907017100

電子郵件: d11431037@gapps.fg.tp.edu.tw

所屬學校: 臺北市立第一女子高級中學

真的可以登入...

帳號變更
密碼變更
救援資訊變更
教育局WORKSPACE



-再試一次

亂輸的 "test"

臺北市校園
單一身分驗證服務

請輸入您的使用者名稱與密碼

帳號
test
密碼

忘記帳號/密碼

帳號不存在或密碼錯誤，登入失敗

登入

Burp Suite Community Edition v2025.10.6 - Temporary Project

尚未有

Dashboard Target Proxy Intruder Repeater Collaborator Sequencer Decoder Comparer Logger Organizer Extensions Learn

Intercept HTTP history WebSockets history Match and replace ⚙️ Proxy settings

Request to https://ldap.tp.edu.tw:443 [163.21.244.100] ⚙️ Open browser ⚙️

Time	Type	Direction	Method	URL	Status code	Length
11:54:50 1 D...	HTTP	→ Request	POST	https://ldap.tp.edu.tw/web/login/authenticate		

Pretty Raw Hex

```
14 Sec-Fetch-Mode: cors
15 Sec-Fetch-Site: sameorigin
16 Referer: https://ldap.tp.edu.tw/login
17 Accept-Encoding: gzip, deflate, br
18 Priority: u=1, i
19 Connection: Keep-Alive
20
21 {
  "account": "test",
  "password": "test",
  "relayState": null,
  "samlRequest": null
}
```

Event log All issues

21 {

"account": "test",
"password": "test",
"relayState": null,
"samlRequest": null
}

真的看的到...

Inspector

Request attributes 2

Request query parameters 0

Request cookies 1

Request headers 18

Memory: 157.9MB Disabled

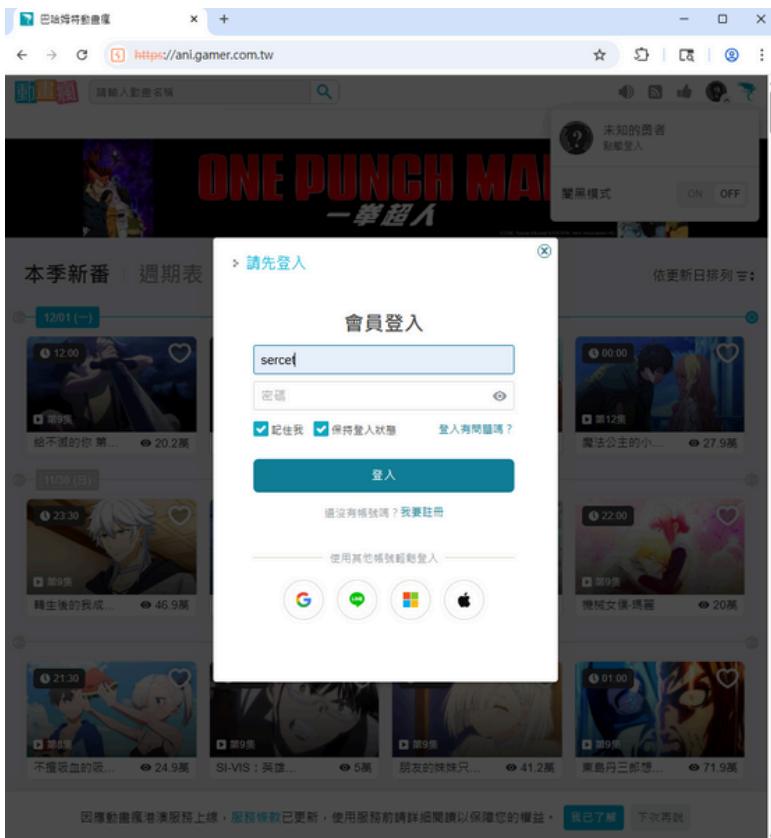
心得:

國中時的輔導課總會請講師來教導我們資安的重要，例如不要亂連不明來源的網站，密碼很容易就此被盜走之類的，那時候我總覺得這些狀況離現實很遠，系統怎麼可能那麼容易被入侵？

但在今天的課程過後，我才真正體會到網路世界到底有多危險。如果一不小心讓這類程式連接到我的電腦的話，駭客想要看到密碼，只需動動手指就行。這讓我意識到，資安並不是誇張的劇情，而是每個人都可能遇到的真實問題。

今天的課讓我體會到資安世界的有趣與危險，也更明白平時提高警覺、養成良好習慣的重要性。

-換個網站試試



△巴哈姆特動畫瘋網站的登入介面

Time	Type	Direction	Method	URL
12:01:57 1 D...	HTTP	→ Request	POST	https://api.gamer.com.tw/v1/login_precheck.php
12:01:57 1 D...	HTTP	→ Request	POST	https://analytics.google.com/g/collect?v=2&tid=G-2Q21791Y9D>m=45je5bj0h2v872385672za200zb71595720zd715957208_p=1
12:01:57 1 D...	HTTP	→ Request	GET	https://www.google.com.tw/ads/ga-audiences?v=1&t=sr&lf_rd=1&_r=4&tid=G-2Q21791Y9D&cid=915358385.1764561517>m=4
12:01:59 1 D...	HTTP	→ Request	GET	https://user.gamer.com.tw/login.php

```
Pretty Raw Hex
1 POST /user/v1/login_precheck.php HTTP/2
2 Host: api.gamer.com.tw
3 Content-Length: 13
4 Sec-Ch-Ua-Platform: "Windows"
5 Accept-Language: zh-TW, zh;q=0.9
6 Accept: */*
7 Sec-Ch-Ua: "Not_A_Brand";v="99", "Chromium";v="142"
8 Content-Type: application/x-www-form-urlencoded; charset=UTF-8
9 Sec-Ch-Ua-Mobile: ?0
10 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/142.0.0.0 Safari/537.36
11 Origin: https://user.gamer.com.tw
12 Sec-Fetch-Site: same-site
13 Sec-Fetch-Mode: cors
14 Sec-Fetch-Dest: empty
15 Referer: https://user.gamer.com.tw/login.php
16 Accept-Encoding: gzip, deflate, br
17 Priority: -1, i
18
19 userid=sercet
```

L9 **userid=sercet**

也看的到，只不過只看的到使用者名稱看不到密碼

說明：

這次選擇嘗試的是我日常會使用的網站。在照片中我們可以看到輸入捨用者名稱以後，Burp Suite中就會跳出所輸入的內容。（如果輸入了密碼的話Burp Suite就會跳掉，所以我沒有截到圖。但我覺得就算只有暴露帳號，還是蠻危險的）

12/08

???

12/15

???

12/22

???

► 與
Chat
gpt
的
部
分
對
話
紀
錄

答案!



課後感想

起初，我帶著一點憧憬、一點想像，跨進這門課的大門。那時候的我只覺得「資安」這兩個字背後有種神祕的魅力--似乎藏著某種只有少數人懂得的語言。沒想到，真正踏上這條路後，我看到的，是一個比想像中更廣、更深、更有趣的世界。

從最一開始的「編碼與解碼」、「密碼學概論」開始，那些看似亂碼，又似某種秘密對話的開端，彷彿負載著資訊世界的另一種魔法。

在「隱寫術」的那堂課，是我第一次意識到訊息可以被藏得如此安靜、如此優雅，默默存在於細節深處。

「Linux」中課程，不同的指令與介面讓我感到陌生，但也因此更想探索。

兩次跨校資安實務素養體驗賽，是最讓人心跳加速的時刻。個人賽的緊張、團體賽的默契與感動，讓我了解到原來學習不只是在吸收，而是也在等待被挑戰、被驗證的瞬間裡，獲得另一種成就。

在「人工智能簡介」中，我們看見技術與未來的交點；「SQL Injection」、「Burp Suite攔截」課程，則讓我第一次真正理解，原來資訊世界的脆弱與堅強，一線之隔。

在這學期的學習過程中，我遇過困惑、卡關，也遇過突然恍然大悟的時刻。這些過程就像一種溫柔的磨礪：跌倒使人學會重新站起，好奇使人前進，這門課讓我看到資訊世界的另一面，也使我開始思考，原來那些被我們視為理所當然的安全，都是一套套演算法與邏輯默默構築出來的。

也許我一開始是因為那份神祕與帥氣而靠近，但結束這學期時，吸引我留下的，是看見日常背後的祕密的感動，也讓我在資訊的世界裡，看見了比想像中更美麗的風景。

