Misc安全雜項

MISC概述

- Misc（Miscellaneous，雜項）
- Misc 是切入 CTF 競賽領域、培養興趣的較佳入口。
- Misc 考察基本知識，對安全技能的各個層面都有不同程度的涉及，可以在很大程度上啟發思維。根據出題人的意圖，大致可以分為幾類:
- （一）考察人的腦洞，這這塊以編碼解碼為主
- （二）安全中經常會用到的，這塊以流量分析為主
- （三）給你簡單的快樂，類似於簽到題
- （四）對於安全知識的深度廣度，圖片、文件、音頻的隱寫

MISC概述

- 圖片隱寫：EXIF、寬高隱寫、LSB隱寫、盲浮水印
- 音頻隱寫：摩爾斯電碼、MP3音頻、波形圖、頻譜圖
- 文檔隱寫：word文檔隱寫、pdf文檔隱寫
- 視頻隱寫：提取幀
- 偽加密：zip偽加密、rar偽加密
- 真加密：暴力破解、字典攻擊、掩碼攻擊、明文攻擊、CRC碰撞
- web流量分析：http流量分析、webshell混淆流量分析、TLS流量分析
- USB流量分析：鍵盤流量分析、滑鼠流量分析、印表機流量分析
- 磁片取證：DiskGenius、ext3grep、extundelete
- 內存取證：volatility

MISC概述

隱寫

隱寫術成為資訊安全領域研究的焦點，每個web站點都依賴多媒體，音頻、視頻、圖像。使用隱寫，就可以將資訊潛入數字媒介而不影響載體的品質，隱寫就是把資訊隱寫到文件裏。

CTF中經常利用下麵幾種資訊載體進行隱寫

圖片文件（.jpg .png .gif .bmp）

文本文件(.docx .pdf .zip)

音視頻文件(.wav.mp3 .mp4)

其他特殊文件

判斷文件類型，就是看它的擴展名， 也可使使用特殊的文件結構工具分析文件類型，像Linux的file命令。但是所有的工具都是基於類似字典的特徵碼進行分析的，如果特徵碼字典文件中沒有收錄，有或者目標文件的一些特徵碼被破壞，那麼工具就沒法識別。這個時候就需要人工分析文件的HEX值，找特徵碼進行判斷，就得用到WinHEX、或者010editor工具。

做題三板斧

Binwalk+winhex方向：分析文件結構和內部數據

StegSolve方向：分析LSB隱寫

Stegdect方向：檢測特殊工具隱寫

常見的文件頭標識

文件類型	文件頭標識	描述
7z	37 7A BC AF 27 1C	7-ZIP compressed file
bz; bz2	42 5A 68	BZIP Archive
exe、dll、drv、vxd、sys、ocx	4D 5A	Win32 Executable
gif	47 49 46 38 39 61	Graphics interchange format file
gz; tar; tgz	1F 8B	Gzip Archive File
jpg; jpe; jpeg	FF D8 FF E0 00	JPG Graphic File
MP3	49 44 33	MPEG-1 Audio Layer 3 (MP3) audio file
png	89 50 4E 47 0D 0A	PNG Image File
rar	52 61 72 21	RAR Archive File
wav	57 41 56 45 66 6D 74	Wave Files
zip; jar; zipx	50 4B 03 04	ZIP Archive

一些工具

010編輯器

010編輯器是一個可以查看十六進制文件的軟體，一般在題目所給出文件，但又不知道文件的類型時。可以通過010編輯器打開文件，獲取文件的類型，一般文件類型在頭部出現，例如上圖文件的頭部資訊（最開始的前八個位元組）可以知曉這是一個png文件。

Wireshark

這是一個網路監聽軟體，可以監聽所連接網路，獲取通過這個網路收發資訊的內容情況

Stegsolve工具

這是一個jar工具，使用的時候需要安裝jre環境，這個工具具有分析圖片資訊圖片運算的操作（+，-，&），而且還可以將gif圖化為一幀幀的圖片觀察，等等，小巧而又強大。

File Format：文件格式，這個主要是查看圖片的具體資訊
Data Extract：數據抽取，圖片中隱藏數據的抽取
Frame Browser：幀流覽器，主要是對GIF之類的動圖進行分解，動圖變成一張張圖片，便於查看
Image Combiner：拼圖，圖片拼接

file命令

可以查看文件的類型

一些工具

binwalk命令

沒有的可以安裝 sudo apt-get install binwalk,可以查看隱藏文件，有些文件表面上是.jpg實際上是一個雜交體。

安裝：apt install binwalk

binwalk -e 可以分離文件

foremost命令

將組合文件分離形成一個output文件夾，裏面有分離出來的文件。還會生成一個audit.txt日誌資訊，十分好用的命令

安裝：apt install foremost

安裝：yum -y install foremost

用法：foremost 文件名

Zsteg

zsteg可以檢測PNG和BMP圖片裏的隱寫數據。

Winhex

WinHex是一款以通用的 16 進制編輯器為核心，專門用來對付電腦取證、數據恢復、低級數據處理、以及 IT 安全性、各種日常緊急情況的高級工具： 用來檢查和修復各種文件、恢復刪除文件、硬碟損壞、數碼相機卡損壞造成的數據丟失等。

一些工具

JD-GUI (Java反編譯工具)

JD-GUI是使用C++開發的一款Java反編譯工具，它是一個獨立圖形介面的Java源代碼".class"文件反編譯工具。

GifSplitter

GifSplitter可以把GIF動畫分解成單個圖像幀。然後，您可以選擇任何幀的GIF動畫，並修改它們。此外，GifSplitter將創建一個 .gsf 文件，它可以在Magic ASCII Studio中使用。因此，GifSplitter能夠在Magic ASCII Studio裏直接轉換Magic ASCII碼文件為ASCII藝術動畫了。

MP3stego

在隱寫中，有時候會碰到音頻裏面會有隱藏資訊，用MP3stego可以提取裏面隱藏的資訊，非常好用！

Stegdetect

一款自動化數字圖像隱寫分析工具

密碼破解工具

Passper for RAR用來對rar文件的密碼進行破解

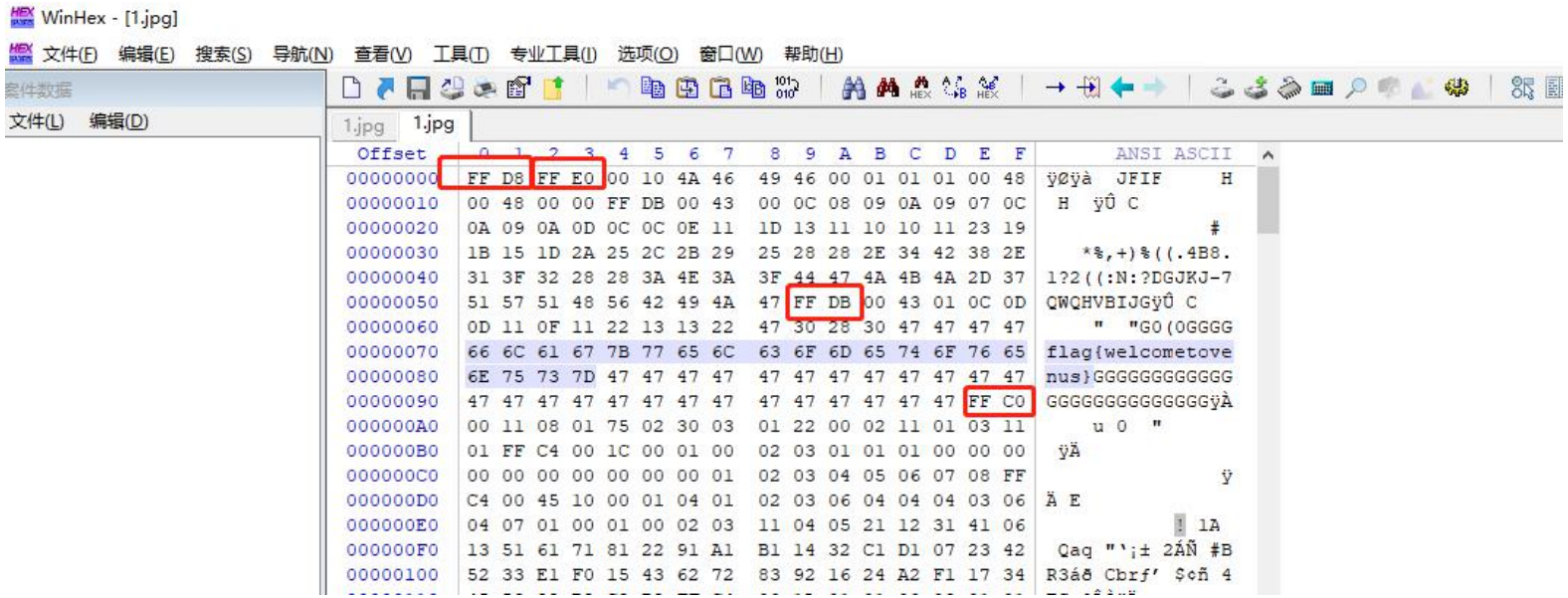

圖片隱寫

JPEG格式分析

- JPEG是一種使用有損壓縮方法保存的圖像格式，對於隱藏的資訊可能造成破壞或改變。
- 由於壓縮，輸出圖像是存儲大小和圖像品質之間的權衡。用戶可以調整壓縮級別，以實現所需的品質級別，同時縮小存儲大小。如果對圖像進行10:1的壓縮，圖像品質的影響微不足道。壓縮值越高，圖像品質的下降越高。

JPEG格式分析

- JPEG文件大體上可以分成兩個部分： 標記碼(Tag)和壓縮數據。
- 圖片的文件格式
- FF DB： 圖片起始標記
- 0xFF E0至0xFF DB: 這塊是描述，圖片的長度，高寬，色深，壓縮方法等
- 0xFF DB至0xFF C0: 標記碼，可以插入隱蔽數據，不會影響圖片的打開
- 0xFF C0:幀圖像
- 0xFF D9:圖片結束



PNG格式分析

- PNG文件是一種使用無損壓縮的光柵圖像文件格式。
- 此文件格式是作為圖形交換格式（GIF）的替代品而創建的，沒有版權限制。但是，PNG文件格式不支持動畫。PNG文件格式支持無損圖像壓縮，使其在用戶中很受歡迎。隨著時間的流逝，PNG已成為廣泛使用的圖像文件格式之一。

PNG格式分析

- PNG文件前八個位元組始終為　　89 50 4E 47 0D 0A 1A 0A
- PNG文件結束位元組為　　AE 42 60 82

附加隱寫

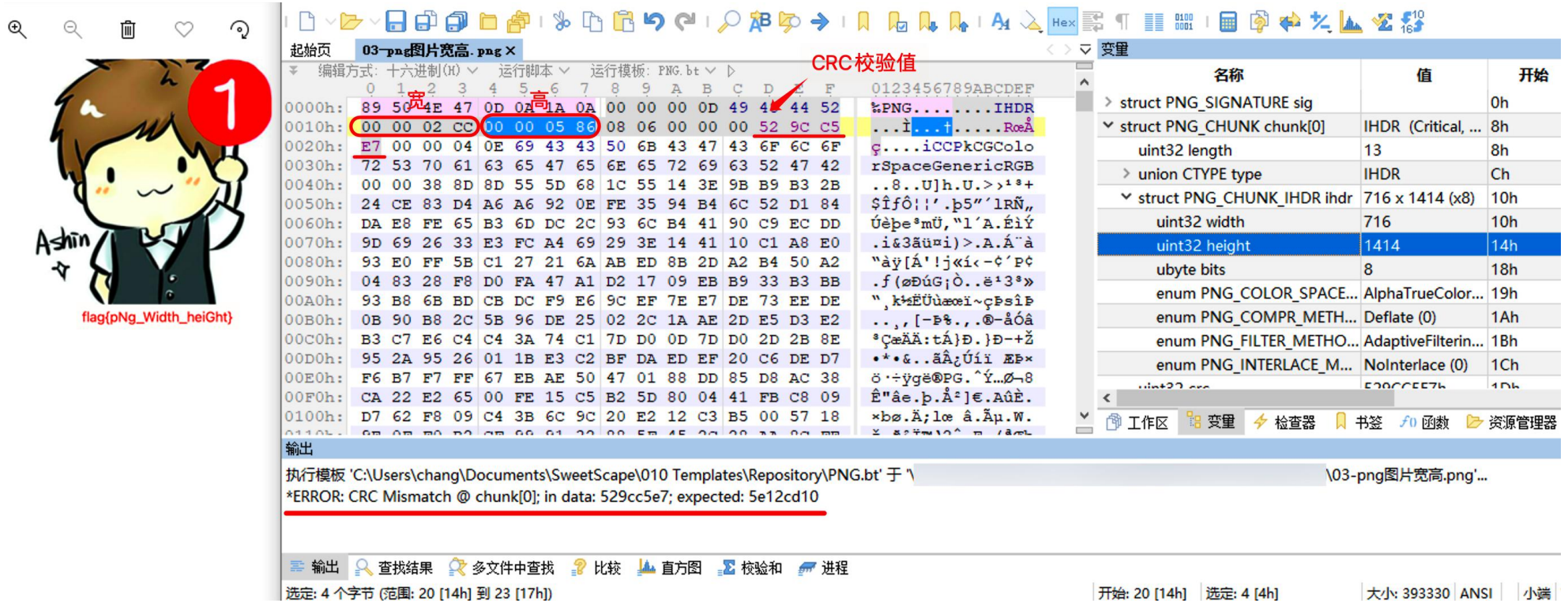
- 在附加式的圖片隱寫術中，通常用某種程式或者某種方法在載體文件中直接附加上需要被隱寫的目標，之後將載體文件直接傳輸給接受者或者發佈到網站上，然後接受者根據方法提取出被隱寫的消息
- 在CTF賽事中，關於這種圖片隱寫的有兩種經典方式:
- 直接附加字串
- 圖種的形式

附加隱寫

- 通常情況下，我們一般選擇binwalk進行識別，使用foremost進行分離
- binwalk filename 識別
- foremost filename　分離

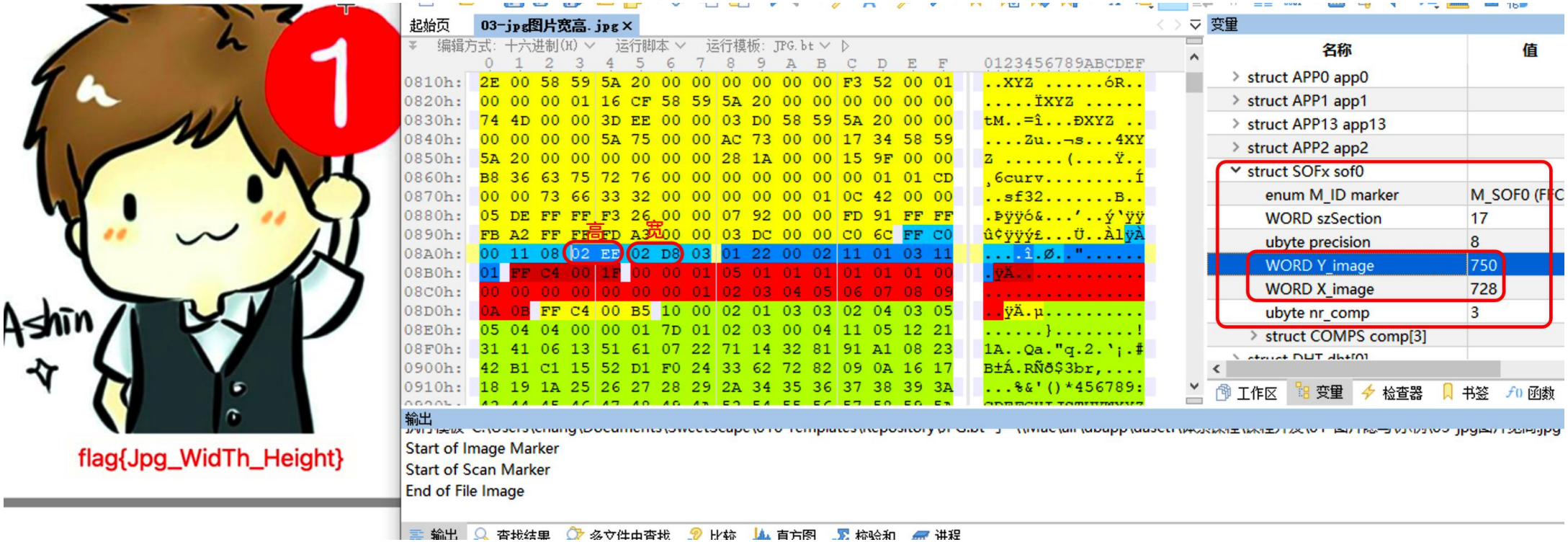
圖片寬高隱寫

- PNG


flag{pNg_Width_heiGht}
CRC校验值
宽
高
起始页
03-png图片宽高.png
变量
名称
值
开始
struct PNG_SIGNATURE sig
struct PNG_CHUNK chunk[0]
IHDR (Critical, ...
uint32 length
13
union CTYPE type
IHDR
struct PNG_CHUNK_IHDR ihdr
716 x 1414 (x8)
uint32 width
716
uint32 height
1414
ubyte bits
8
enum PNG_COLOR_SPACE...
AlphaTrueColor...
enum PNG_COMPR_METH...
Deflate (0)
enum PNG_FILTER_METHO...
AdaptiveFilterin...
enum PNG_INTERLACE_M...
NoInterlace (0)
输出
执行模板 'C:\Users\chang\Documents\SweetScape\010 Templates\Repository\PNG.bt' 于 '\03-png图片宽高.png'...
*ERROR: CRC Mismatch @ chunk[0]; in data: 529cc5e7; expected: 5e12cd10

圖片寬高隱寫

- JPG


1
Ashin
flag{Jpg_WidTh_Height}
起始页
03-jpg图片宽高.jpg
编辑方式: 十六进制(H)
运行脚本
运行模板: JPG.bt
高
宽
变量
名称
值
struct APP0 app0
struct APP1 app1
struct APP13 app13
struct APP2 app2
struct SOFx sof0
enum M_ID marker
M_SOF0 (FFC
WORD szSection
17
ubyte precision
8
WORD Y_image
750
WORD X_image
728
ubyte nr_comp
3
struct COMPS comp[3]
工作区
变量
检查器
书签
函数
输出
Start of Image Marker
Start of Scan Marker
End of File Image

圖片寬高隱寫

- BMP

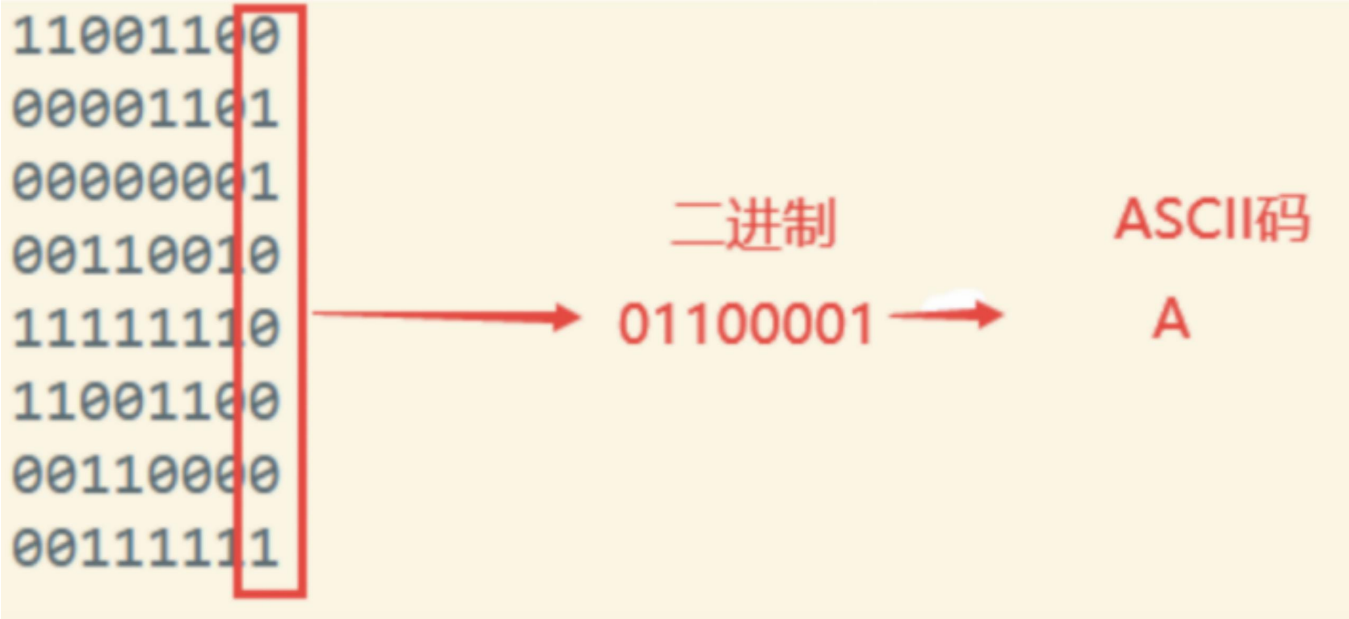

flag{bmp_wiDTh_HeigHt}
Ashin
起始页
03-bmp图片宽高.bmp
编辑方式: 十六进制(H)
运行脚本
运行模板: BMP.bt
0 1 2 3 4 5 6 7 8 9 A B C D E F 0123456789ABCDEF
0000h: 42 4D AE 4B 15 00 00 00 00 00 36 00 00 00 28 00 BM®K.....6...(.
0010h: 00 00 96 02 00 00 BC 02 00 00 01 00 18 00 00 00 ..–...¼.....
0020h: 00 00 78 4B 15 00 00 00 00 00 00 00 00 00 00 00 ..xK.....
0030h: 00 00 00 00 00 00 FF FF FF FF FF FF FF FF FF FF ......ÿÿÿÿÿÿÿÿÿÿ
宽
高
变量
名称
值
开始
struct BITMAPFILEHEADER bmfh
0h
struct BITMAPINFOHEADER bmih
Eh
DWORD biSize 40 Eh
LONG biWidth 662 12h
LONG biHeight 700 16h
WORD biPlanes 1 1Ah
WORD biBitCount 24 1Ch
DWORD biCompression 0 1Eh
DWORD biSizeImage 1395576 22h
LONG biXPelsPerMeter 0 26h
LONG biYPelsPerMeter 0 2Ah
DWORD biClrUsed 0 2Eh
工作区
变量
检查器
书签
函数
资源管理器

LSB隱寫

- 在 png 圖片中，每個顏色佔有 8 bit，即 256 種顏色，那麼就可以組合成 16777216 種顏色。
- 比如 RGB 值為 (0,255,0) 的顏色為綠色，那麼我們將其中的 255 改成 254，僅憑肉眼的話其實是看不出顏色的變化的。如果一個 png 圖片我們將它裏面的每一個 RGB 值都做一點變動，對於整個圖片來說變化也是微乎其微的了。
- LSB 隱寫就是修改 RGB 顏色分量的最低二進位位，顏色依舊看不出有什麼變化，從而達到隱藏資訊的目的。

LSB隱寫

- 比如我們想要隱藏一個 A 字元，我們可以修改圖片中某些 RGB 的最低位，使得其組合起來為 A 的 ascii 的二進位，那麼我們想要隱藏文件或者隱藏字串就可以按照上述的方法對每個 RGB 的最低位進行修改即可。



LSB隱寫

- 如果想要提取出隱藏資訊的話，只需將每個 RGB 的最低位提取出來，重新組成新的文件或者字串即可。
- 一般對於 LSB 隱寫我們常用的有兩個工具，一個是 stegsolve，多見於 Windows 系統，另外一個是 zsteg，多見於 Linux 系統。


壓縮包隱寫

壓縮包隱寫

- 在做CTF的misc題目時經常會碰到需要密碼的zip文件，打開這種需要密碼的方法無非就這四種：
- （1）根據提示或其他文件解出密碼
- （2）暴力破解、字典破解
- （3）偽加密修改
- （4）明文攻擊

壓縮包分析概述

- 壓縮的原理是把文件的二進位進行壓縮，把相鄰的 0，1 代碼減少，比如有五個連續的零 00000，可以把它變成 5 個 0 的寫法 50 來減少該文件的空間。
- 其中根據壓縮包文件的內容和結構存在著多種破解方式，如暴力破解、字典攻擊與明文攻擊等方法。

ZIP格式解析

- ZIP是一種數據壓縮和文件儲存的格式，優點在於壓縮速度快，並且普及率高。
- 在 Windows 中內置了對 ZIP 格式的支持，我們不需要單獨為他下載專門的壓縮軟體。
- 一個 zip 文件通常由三部分組成:
- 1）壓縮原始檔案數據區
- 2）壓縮原始檔案目錄區
- 3）壓縮原始檔案目錄結束標誌

偽加密

- 1、壓縮原始檔案數據區
- 50 4B 03 04：這是文件頭標記
- 14 00：解壓文件所需 pkware 版本
- 00 00：全局方式位標記（判斷有無加密）
- 08 00：壓縮方式
- 5A 7E：最後修改文件時間
- F7 46：最後修改文件日期

含义	字节数
文件头标记	4 bytes (504b0304)
解压文件所需版本	2 bytes
通用位标记	2 bytes
压缩方式	2 bytes
最后修改文件时间	2 bytes
最后修改文件日期	2 bytes
crc-32	4 bytes
压缩后尺寸	4 bytes
未压缩尺寸	4 bytes
文件名长度	2 bytes
扩展记录长度	2 bytes

偽加密

- 壓縮原始檔案目錄區
- 50 4B 01 02：目錄中文件文件頭標記 （0x02014b50）
- 1F 00：壓縮使用的 pkware 版本
- 14 00：解壓文件所需 pkware 版本
- 00 00：全局方式位標記（判斷是否為偽加密）
- 08 00：壓縮方式
- 5A 7E：最後修改文件時間
- F7 46：最後修改文件日期

含义	字节数
文件头标记	4 bytes (504b0102)
压缩使用的版本	2 bytes
解压需要的版本	2 bytes
通用位标志	2 bytes
压缩方式	2 bytes
最后修改文件时间	2 bytes
最后修改文件日期	2 bytes
crc-32	4 bytes
压缩后尺寸	4 bytes
未压缩尺寸	4 bytes
文件名长度	2 bytes
扩展字段长度	2 bytes
文件注释长度	2 bytes
磁盘开始号	2 bytes
内部文件属性	2 bytes
外部文件属性	4 bytes
本地头相对偏移量	4 bytes

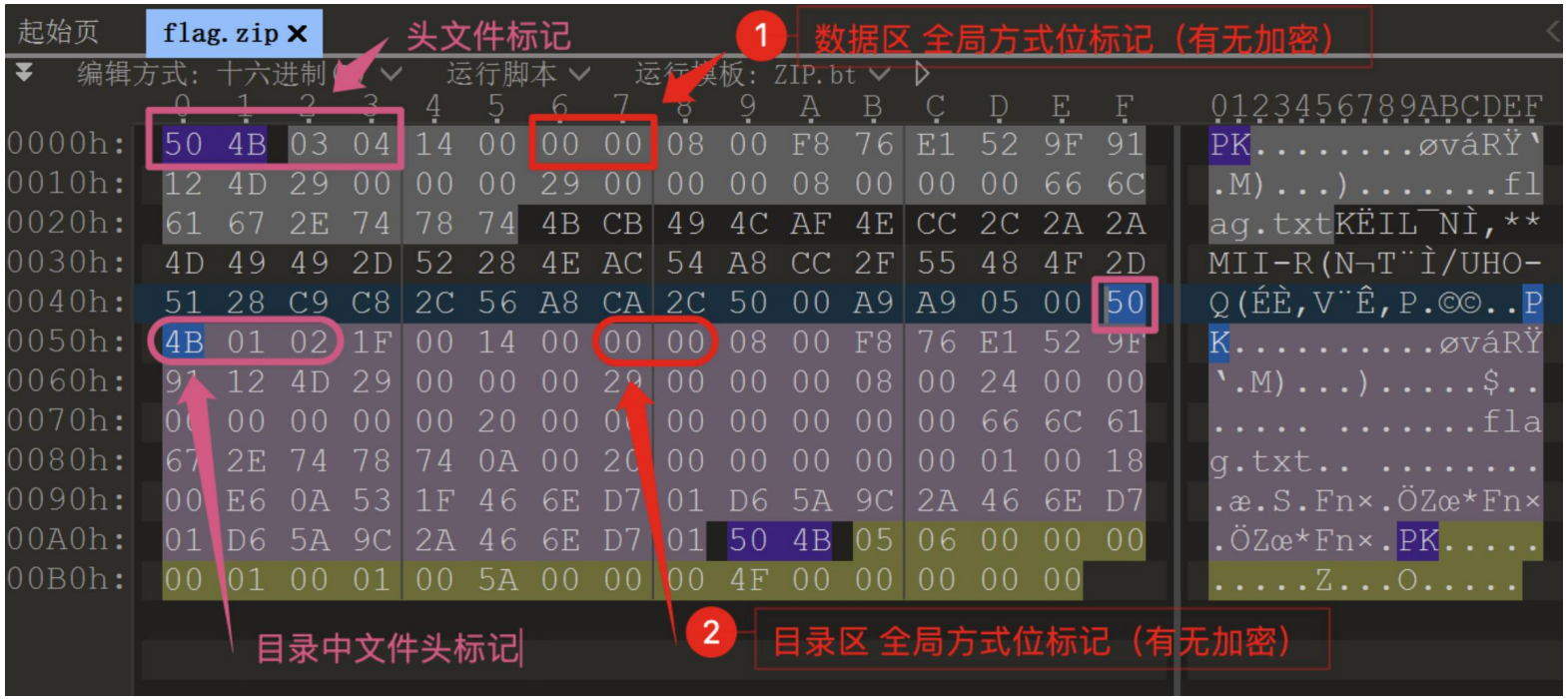
偽加密

- 3、壓縮原始檔案目錄結束標誌：
- 50 4B 05 06：目錄結束標記
- 00 00：當前磁片編號
- 00 00：目錄區開始磁片編號
- 01 00：本磁片上紀錄總數
- 01 00：目錄區中紀錄總數
- 59 00 00 00：目錄區尺寸大小
- 3E 00 00 00：目錄區對第一張磁片的偏移量
- 00 00：ZIP 文件注釋長度

含义	字节数
文件头标记	4 bytes (504b0102)
压缩使用的版本	2 bytes
解压需要的版本	2 bytes
通用位标志	2 bytes
压缩方式	2 bytes
最后修改文件时间	2 bytes
最后修改文件日期	2 bytes
crc-32	4 bytes
压缩后尺寸	4 bytes
未压缩尺寸	4 bytes
文件名长度	2 bytes
扩展字段长度	2 bytes
文件注释长度	2 bytes
磁盘开始号	2 bytes
内部文件属性	2 bytes
外部文件属性	4 bytes
本地头相对偏移量	4 bytes

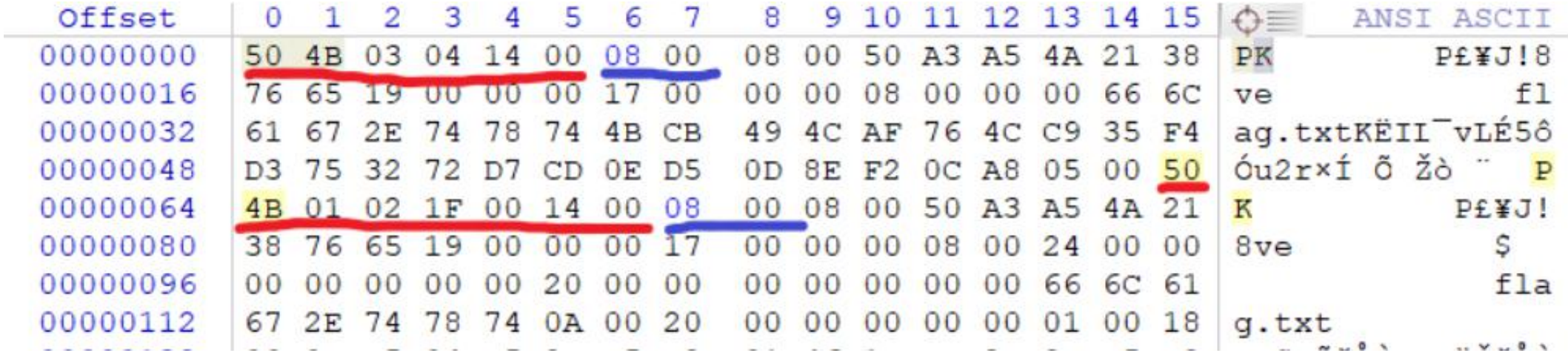
偽加密

- 偽加密時，壓縮文件數據區中的通用位標記的值無所謂，重點是壓縮原始檔案目錄區的通用位標記得存在奇數的情況，例如 0900，0100等，此時我們無法直接打開文件，且壓縮包也不存在真正的密碼，需要手動修改其標誌位。



無加密

- 壓縮原始檔案數據區的全局方式位標記應當為00 00 （50 4B 03 04 14 00 後）
- 且壓縮原始檔案目錄區的全局方式位標記應當為00 00 （50 4B 01 02 14 00 後）



Offset	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	ANSI ASCII
00000000	50	4B	03	04	14	00	08	00	08	00	50	A3	A5	4A	21	38	PK P£¥J!8
00000016	76	65	19	00	00	00	17	00	00	00	08	00	00	00	66	6C	ve fl
00000032	61	67	2E	74	78	74	4B	CB	49	4C	AF	76	4C	C9	35	F4	ag.txtKËII¯vLÉ5ô
00000048	D3	75	32	72	D7	CD	0E	D5	0D	8E	F2	0C	A8	05	00	50	Óu2r×Í Õ Žò ¨ P
00000064	4B	01	02	1F	00	14	00	08	00	08	00	50	A3	A5	4A	21	K P£¥J!
00000080	38	76	65	19	00	00	00	17	00	00	00	08	00	24	00	00	8ve $
00000096	00	00	00	00	00	20	00	00	00	00	00	00	00	66	6C	61	fla
00000112	67	2E	74	78	74	0A	00	20	00	00	00	00	00	01	00	18	g.txt

偽加密

- 壓縮原始檔案數據區的全局方式位標記應當為 00 00 （50 4B 03 04 14 00 後）
- 且壓縮原始檔案目錄區的全局方式位標記應當為 09 00 （50 4B 01 02 14 00 後）

Offset	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	ANSI ASCII
00000000	50	4B	03	04	14	00	08	00	08	00	50	A3	A5	4A	21	38	PK P£¥J!8
00000016	76	65	19	00	00	00	17	00	00	00	08	00	00	00	66	6C	ve fl
00000032	61	67	2E	74	78	74	4B	CB	49	4C	AF	76	4C	C9	35	F4	ag.txtKËII¯vLÉ5ô
00000048	D3	75	32	72	D7	CD	0E	D5	0D	8E	F2	0C	A8	05	00	50	Óu2r×Í Õ Žò ¨ P
00000064	4B	01	02	1F	00	14	00	09	00	08	00	50	A3	A5	4A	21	K P£¥J!
00000080	38	76	65	19	00	00	00	17	00	00	00	08	00	24	00	00	8ve $
00000096	00	00	00	00	00	20	00	00	00	00	00	00	00	66	6C	61	fla
00000112	67	2E	74	78	74	0A	00	20	00	00	00	00	00	01	00	18	g.txt
00000128	00	0F	F5	04	D5	9A	C5	D2	01	46	1F	CB	8A	9A	C5	D2	õ ÕšÅÒ F ËŠšÅÒ

真加密

- 壓縮原始檔案數據區的全局方式位標記應當為09 00 （50 4B 03 04 14 00 後）
- 且壓縮原始檔案目錄區的全局方式位標記應當為09 00 （50 4B 01 02 14 00 後）

Offset	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	ANSI ASCII
00000000	50	4B	03	04	14	00	01	00	08	00	EB	00	1D	49	3A	0D	PK ë I:
00000016	F9	75	12	00	00	00	04	00	00	00	08	00	00	00	64	61	ùu da
00000032	74	61	2E	74	78	74	68	73	CE	CC	27	EA	62	89	1D	04	ta.txthsÎÌ'êb‰
00000048	1C	E0	66	A9	F5	D6	39	3C	50	4B	01	02	00	00	14	00	àf©õÖ9<PK
00000064	01	00	08	00	EB	00	1D	49	3A	0D	F9	75	12	00	00	00	ë I: ùu
00000080	04	00	00	00	08	00	00	00	00	00	00	00	01	00	00	00	
00000096	00	00	00	00	00	00	64	61	74	61	2E	74	78	74	50	4B	data.txtPK

文件修改方法

- 確定是偽加密後就需要將其修改為無加密，方法很簡單，就是將壓縮原始檔案目錄區的全局方式位標記從09 00改為00 00。

Offset	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	ANSI ASCII
00000000	50	4B	03	04	14	00	08	00	08	00	50	A3	A5	4A	21	38	PK P£¥J!8
00000016	76	65	19	00	00	00	17	00	00	00	08	00	00	00	66	6C	ve fl
00000032	61	67	2E	74	78	74	4B	CB	49	4C	AF	76	4C	C9	35	F4	ag.txtKËII¯vLÉ5ô
00000048	D3	75	32	72	D7	CD	0E	D5	0D	8E	F2	0C	A8	05	00	50	Óu2r×Í Õ Žò ¨ P
00000064	4B	01	02	1F	00	14	00	09	00	08	00	50	A3	A5	4A	21	K P£¥J!
00000080	38	76	65	19	00	00	00	17	00	00	00	08	00	24	00	00	8ve $
00000096	00	00	00	00	00	20	00	00	00	00	00	00	00	66	6C	61	fla
00000112	67	2E	74	78	74	0A	00	20	00	00	00	00	00	01	00	18	g.txt
00000128	00	0F	F5	04	D5	9A	C5	D2	01	46	1F	CB	8A	9A	C5	D2	õ ÕšÅÒ F ËŠšÅÒ

Offset	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	ANSI ASCII
00000000	50	4B	03	04	14	00	08	00	08	00	50	A3	A5	4A	21	38	PK P£¥J!8
00000016	76	65	19	00	00	00	17	00	00	00	08	00	00	00	66	6C	ve fl
00000032	61	67	2E	74	78	74	4B	CB	49	4C	AF	76	4C	C9	35	F4	ag.txtKËII¯vLÉ5ô
00000048	D3	75	32	72	D7	CD	0E	D5	0D	8E	F2	0C	A8	05	00	50	Óu2r×Í Õ Žò ¨ P
00000064	4B	01	02	1F	00	14	00	08	00	08	00	50	A3	A5	4A	21	K P£¥J!
00000080	38	76	65	19	00	00	00	17	00	00	00	08	00	24	00	00	8ve $
00000096	00	00	00	00	00	20	00	00	00	00	00	00	00	66	6C	61	fla
00000112	67	2E	74	78	74	0A	00	20	00	00	00	00	00	01	00	18	g.txt
00000128	00	0F	F5	04	D5	9A	C5	D2	01	46	1F	CB	8A	9A	C5	D2	õ ÕšÅÒ F ËŠšÅÒ

其他修復方式途徑

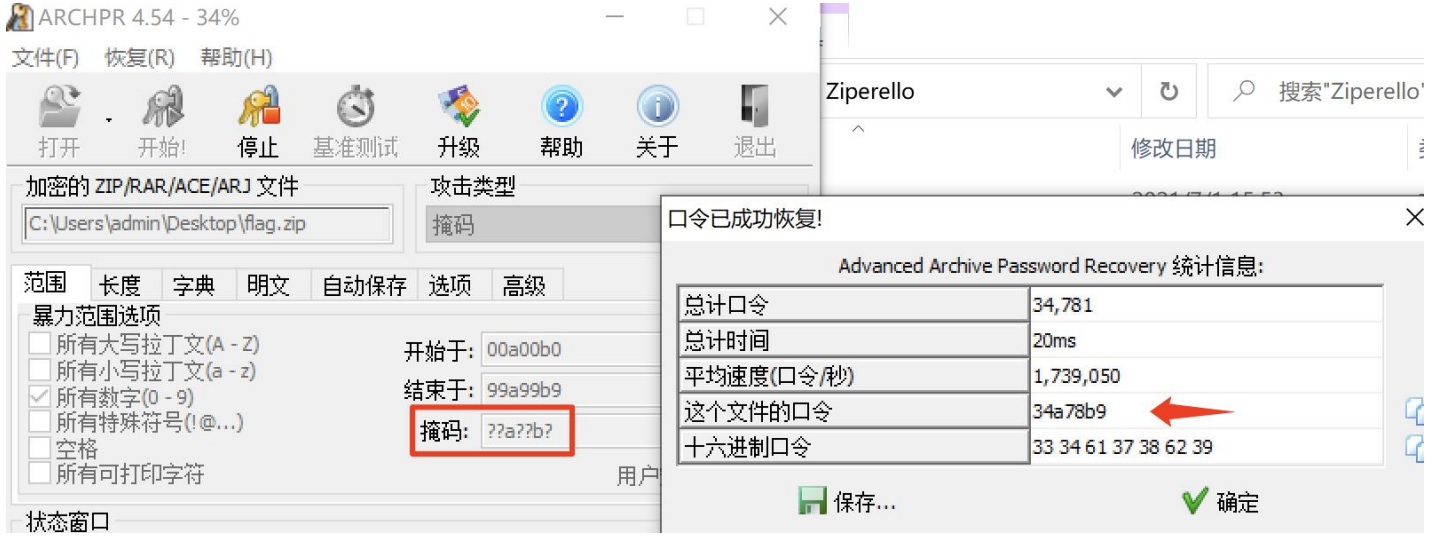
- 1）用binwalk -e 無視偽加密
- 2）在macOS和kali系統中，可以直接打開偽加密zip文件
- 3）檢測偽加密的工具ZipCenOp.jar
  - ZipCenOp.jar用法
  - java -jar ZipCenOp.jar r lm.zip
  - ##這裏的 lm.zip 是壓縮檔案名
- 4）有時用WinRAR的修復功能

暴力破解

- 使用 ARCHPR 進行暴力破解時，我們需要選擇一個範圍和長度。一般情況下可以從純數字開始。選擇範圍長度，若是純數字的話可以將長度設置的稍微大一些，比如 1-9 位。

掩碼攻擊

- 如果已知密碼的某幾位，如已知 7 位密碼中的第 3 位為 a，第 6 位為 b，那麼可以構造 ??a??b? 進行掩碼攻擊。
- 掩碼攻擊的原理相當於構造了第 3 位為 a，第 6 位為 b 的字典，因此掩碼攻擊的效率也比爆破高出不少。



CRC32碰撞

- 例如一個文件，內容長度只有 4 位元組，可以看到它的 CRC32 的值為 0x6526B899。
- 我們可以利用腳本去遍曆所有四位數字串的 CRC32 值，如果和 a.txt 的 CRC32 相同，那麼就能斷定這個字串就是 a.txt 中的內容。

流量及取證分析

Wireshark基本介紹

CTF比賽中，對於流量包的分析取證是一種十分重要的題型。通常這類題目都是會提供一個包含流量數據的pcap文件，參賽選手通過該文件篩選和過濾其中無關的流量資訊，根據關鍵流量資訊找出flag或者相關線索。

pcap流量包的分析通常都是通過圖形化的網路嗅探器——wireshark進行的，這款嗅探器經過眾多開發者的不斷完善，現在已經成為使用最為廣泛的安全工具之一。

Wireshark基本過濾

數據包篩選功能是wireshark的核心功能，比如需要篩選出特定的協議如HTTP，Telnet等，也可能需要篩選出ip地址，端口等。

1.1源ip篩選　　輸入命令：ip.src == 地址
　　目的ip篩選　輸入命令：ip.dst == 地址

1.2 mac地址篩選:

eth.dst ==A0:00:00:04:C5:84 篩選目標mac地址

eth.addr==A0:00:00:04:C5:84 篩選MAC地址

1.3 端口篩選:

tcp.dstport == 80　篩選tcp協議的目標端口為80 的流量包

tcp.srcport == 80　篩選tcp協議的源端口為80 的流量包

udp.srcport == 80　篩選udp協議的源端口為80 的流量包

1.4 協議篩選:

tcp　篩選協議為tcp的流量包

udp 篩選協議為udp的流量包

arp/icmp/http/ftp/dns/ip　篩選協議為arp/icmp/http/ftp/dns/ip的流量包

1.5 包長度篩選:

udp.length ==20　篩選長度為20的udp流量包

tcp.len >=20　篩選長度大於20的tcp流量包

ip.len ==20　篩選長度為20的IP流量包

frame.len ==20 篩選長度為20的整個流量包

1.6　http請求篩選

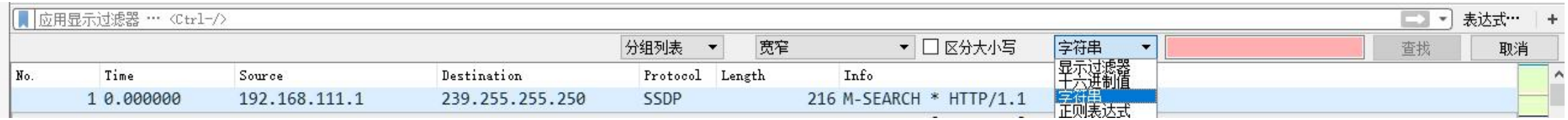
請求方法為GET：http.request.method==“GET”篩選HTTP請求方法為GET的　流量包

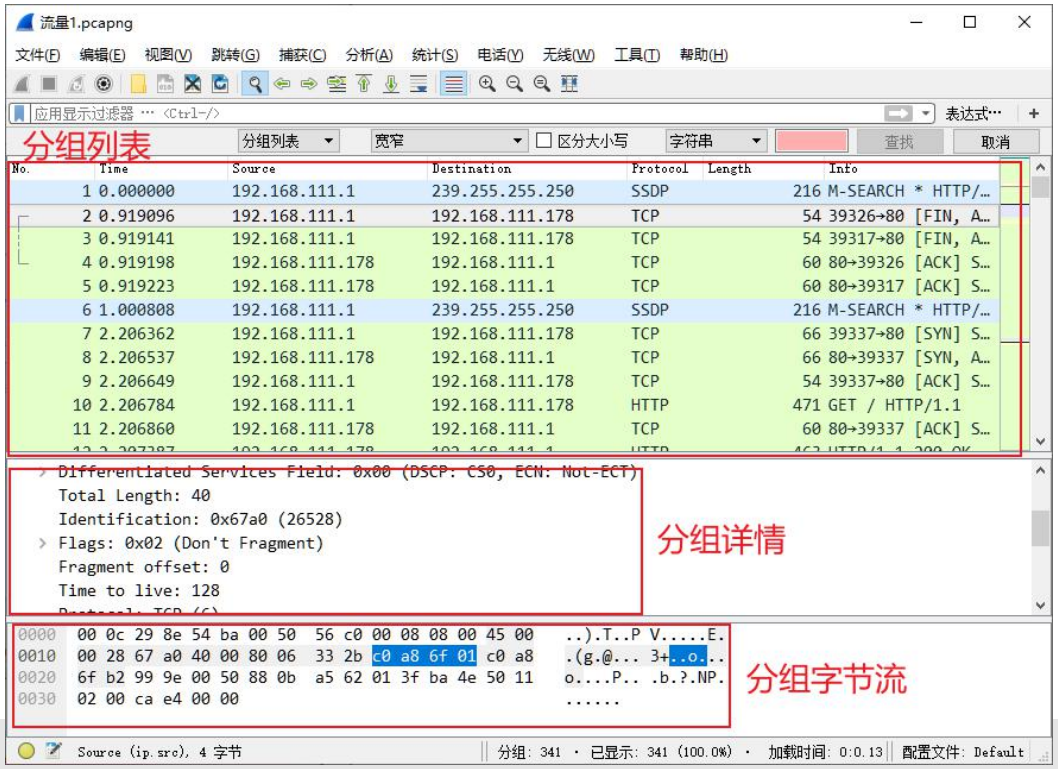
請求方法為POST：http.request.method==“POST”篩選HTTP請求方法為POST的流量包

指定URI：http.request.uri==“/img/logo-edu.gif”篩選HTTP請求的URL為/img/logo-edu.gif的流量包

請求或相應中包含特定內容：http contains “FLAG” 篩選HTTP內容為/FLAG的流量包

Wireshark數據包搜索

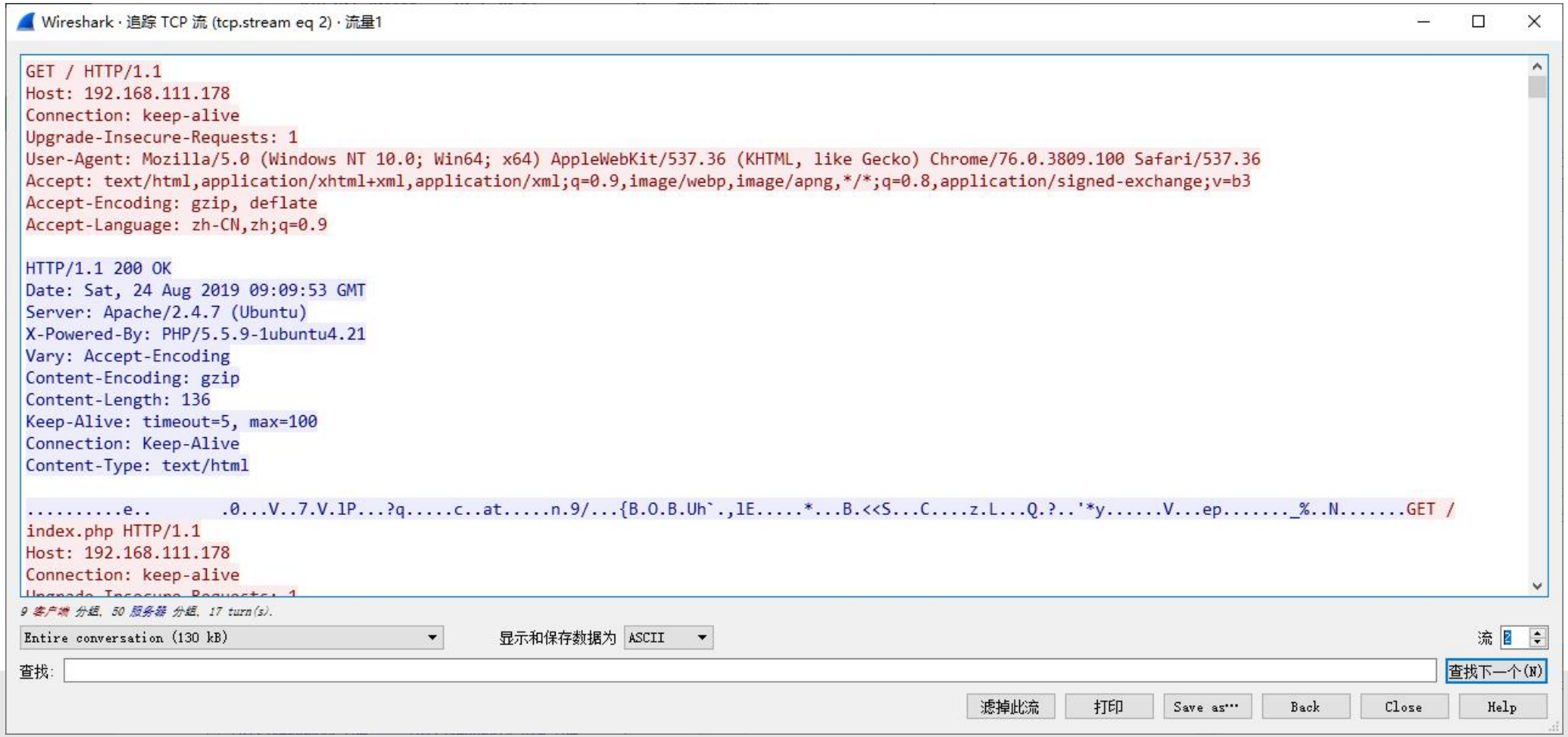




Wireshark數據包還原

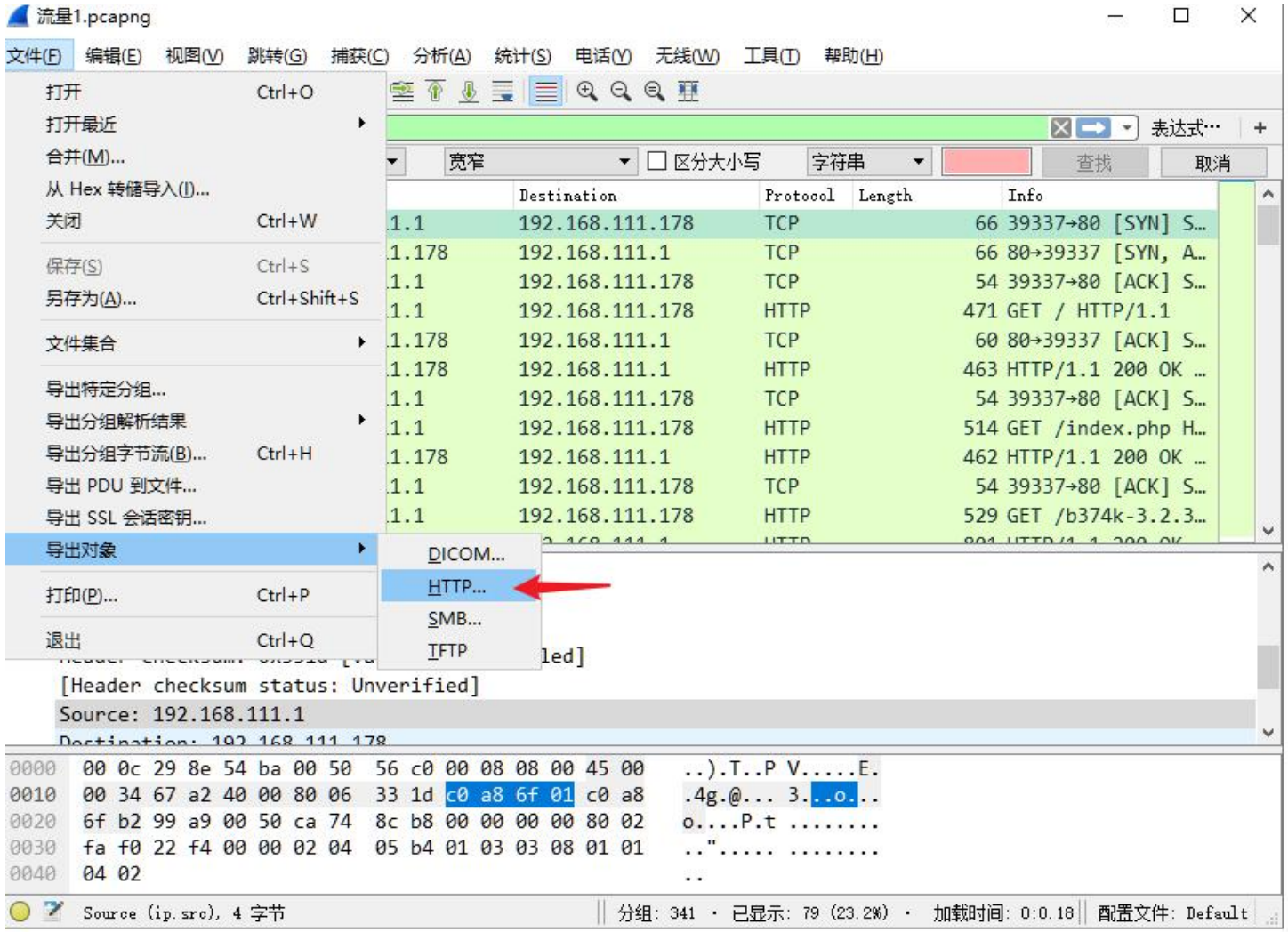
在wireshark中，存在一個交追蹤流的功能，可以將HTTP或TCP流量集合在一起並還原成原始數據。
具體操作方式如下:

選中想要還原的流量包，右鍵選中，選擇追蹤流 – TCP流/UPD流/SSL流/HTTP流。



數據提取

Wireshark支持提取通過http傳輸（上傳/下載）的文件內容，方法如下



批量提取

數據提取

選中http文件傳輸流量包，在分組詳情中找到data或者Line-based text data:text/html層，滑鼠右鍵點擊 – 選中 導出分組位元組流。

Tshark

tshark是wireshark的命令行工具，通過shell命令抓取、解析報文。tcpdump是Linux系統下的抓包工具。wireshark和tcpdump都共同使用 libpcap作為其底層抓包的庫，tshark也可以抓取報文。

tshark命令解析数据包

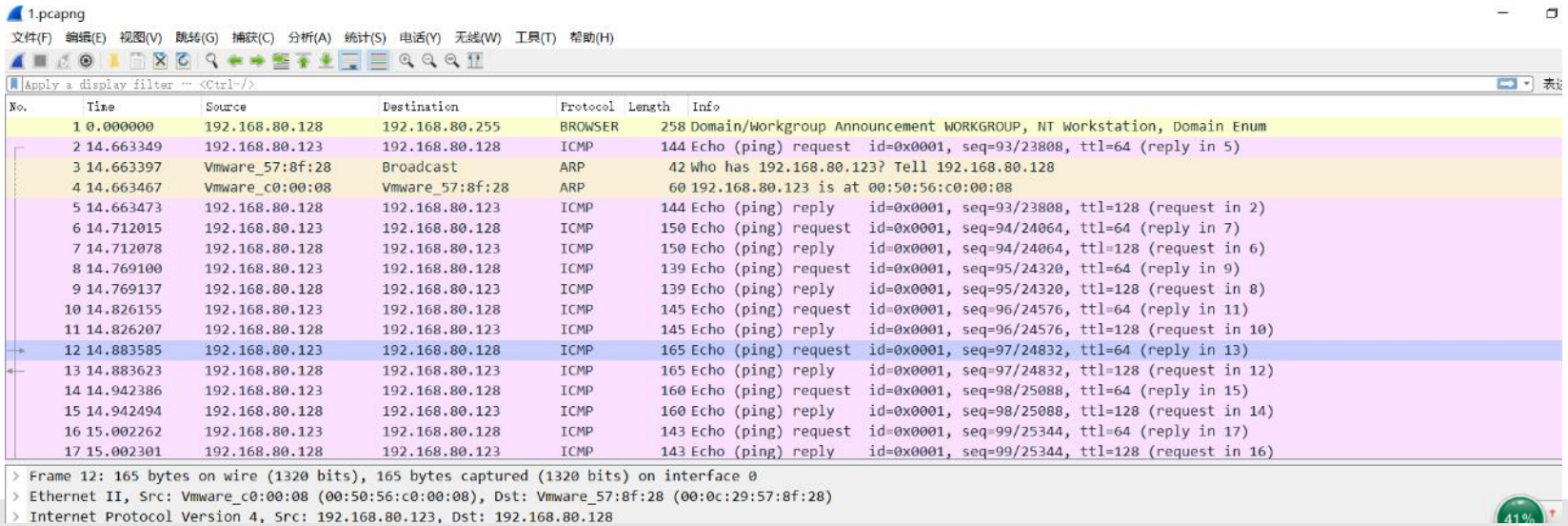
常用参数:

- -r：指定需要解析的数据包
- -T：指定数据包解析输出格式，支持格式见解码所有数据，这里介绍 -T fields，一般与-e 选项连用。
- -e：指定过滤的字段
- -E：可用于指定分隔符： separator=,：默认分隔符为缩进（\t）
- -Y：过滤指定报文

```
1. tshark [ -r <infile> ]  -T fields [ -e <field> ] -E <field print option> -Y <displaY filter>
2. tshark -r packet.pcap -T fields -e 解析的字段 -E separator=,
```

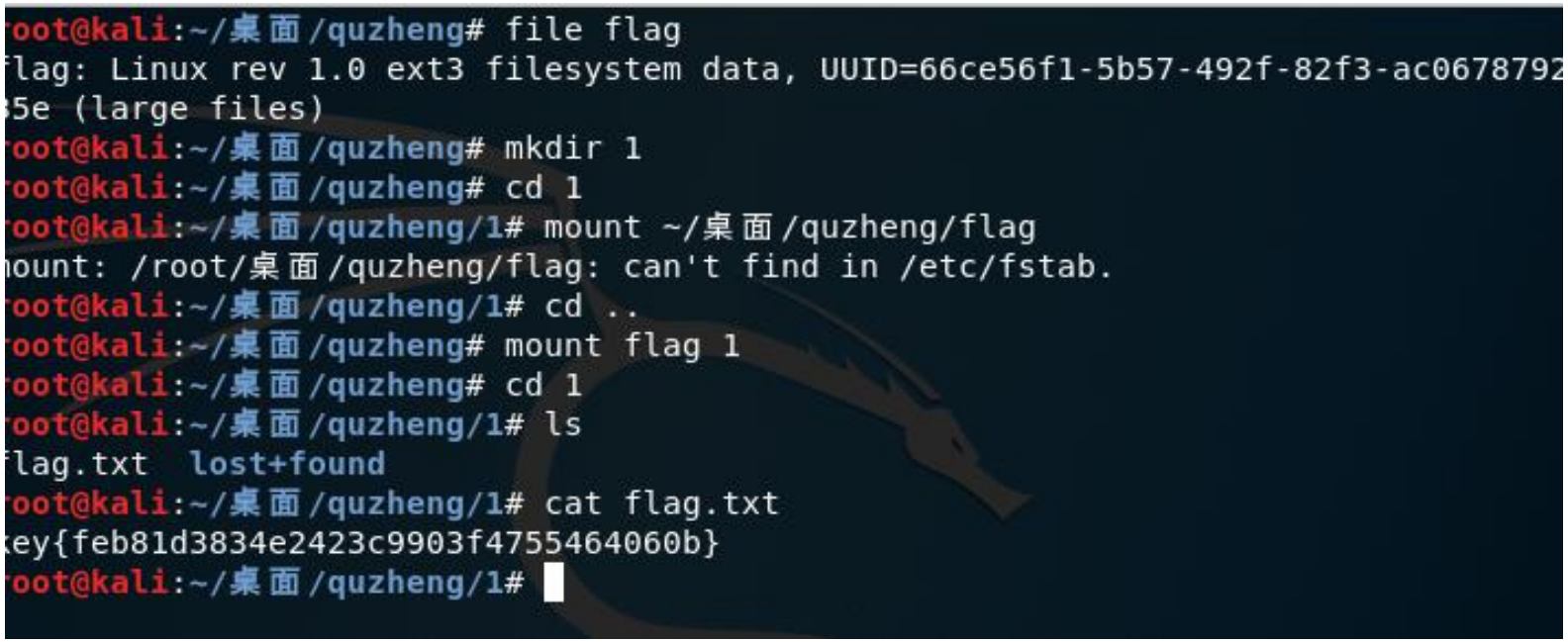
ICMP流量分析

ICMP協議是一個網路層協議。
一個新搭建好的網路，往往需要先進行一個簡單的測試，來驗證網路是否暢通；但是IP協議並不提供可靠傳輸。如果丟包了，IP協議並不能通知傳輸層是否丟包以及丟包的原因。
所以我們就需要一種協議來完成這樣的功能–ICMP協議。
ICMP協議的功能
ICMP協議的功能主要有：
1. 確認IP包是否成功到達目標地址
2. 通知在發送過程中IP包被丟棄的原因



取證分析

CTF取證類賽題會提供一個完整的磁片鏡像，參賽者需要具備一定的策略來在這個數據系統中尋找特定的flag。在電腦取證中，這類策略指的是快速理清內容的能力。沒有策略的話只能耗時耗力地查看所有的資訊。



內存取證

在CTF中，內存取證一般指對電腦及相關智能設備運行時的內存中存儲的臨時數據進行獲取與分析，提取flag或者與flag相關重要資訊。

解析Windows/Linux/Mac OS的內存結構、分析進程等內存數據、根據題目提示尋找線索和思路，提取分析指定進程的特定數據。

常見的內成員結構存在於以下三大操作系統:

- Windows操作系統
- Linux操作系統
- Mac OS操作系統

常見的內存鏡像文件格式有img、dmp、raw、vmem等。

內存取證

Volatility Framwork是一款開源的基於Python開發的鏡像分析框架，它自帶的分析插件支持分析鏡像中所保留的歷史網路連接資訊、歷史進程、歷史命令記錄等等。Kali系統自帶，其他系統可自行到Github上進行Download。
Volatility支持對32位或64位Wnidows、Linux、Mac、Android操作系統的RAM數據進行提取與分析。

volatility -f memory imageinfo #查看系統版本
volatility -f memory --profile=Win7SP1x64 cmdscan #查看cmd命令歷史
volatility -f memory --profile=Win7SP1x64 filescan | grep “flag” #查找flag文件
volatility -f memory --profile=Win7SP1x64 lsadump #跑密碼
volatility -f memory --profile=Win7SP1x64 dumpfiles -Q 0x000000007f142f20 -D ./ -u #導出記憶體中緩存的文件

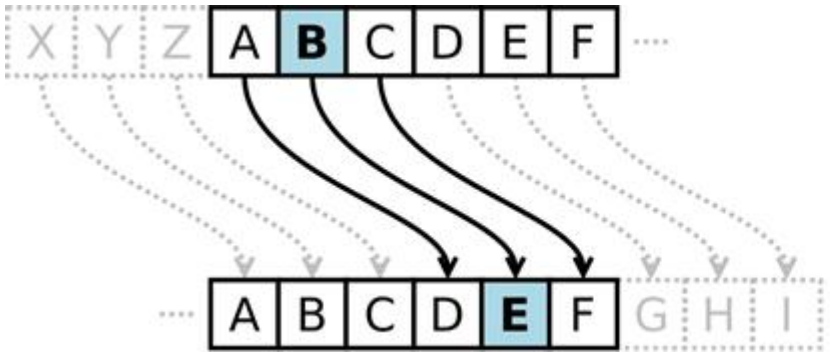
常 見 編 碼

凱撒密碼

愷撒密碼（英語：Caesar cipher），或稱愷撒加密、愷撒變換、變換加密，是一種最簡單且最廣為人知的加密技術。當年愷撒曾用此方法與其將軍們進行聯繫。

演算法

明文中的所有字母都在字母表上向後（或向前）按照一個固定數目進行偏移後被替換成密文。$C=(M+k)\ mod\ 26$ 例如，當偏移量是3的時候，所有的字母A將被替換成D，B變成E，以此類推。



明文字母表：ABCDEFGHIJKLMNOPQRSTUVWXYZ
密文字母表：DEFGHIJKLMNOPQRSTUVWXYZABC

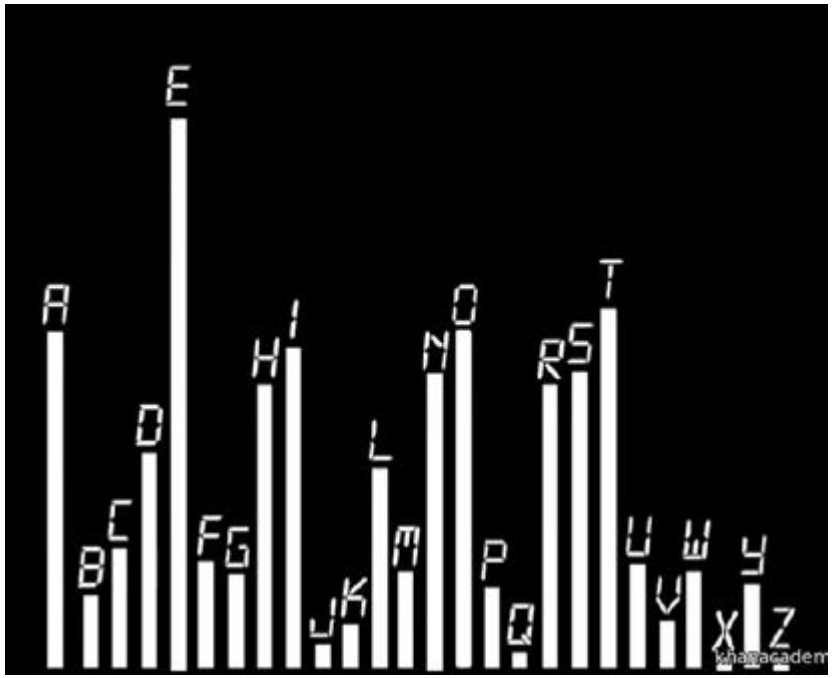
密文：Q TQSM BW LZQVS UQTS 明文：I LIKE TO DRINK MILK

簡單替換密碼

將明文中所使用的字母替換為另一套字母表，形成新的對應關係。這種替換可以是任意的一對一關係。
例如：現有一個簡單替換密碼表

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
W	Y	H	F	X	U	M	T	J	V	S	G	E	N	B	R	D	Z	L	Q	A	P	C	O	K	I

明文： crypto
密文： hzkrqb



仿射密碼

仿射密码是一种替换密码。它是一个字母对一个字母的。

它的加密函数是$e(x) = ax + b \pmod{m}$，其中

- a和m互质。
- m是字母的数目。

实例

字母A-Z对应的数字

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25

加密 若密钥对为（5，8），即函数中a为5，b为8，我们使用的是英文字母，即m为26

明文字母	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
明文数字	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
(5x+8)mod(26)	8	13	18	23	2	7	12	17	22	1	6	11	16	21	0	5	10	15	20	25	4	9	14	19	24	3
密文字母	I	N	S	X	C	H	M	R	W	B	G	L	Q	V	A	F	K	P	U	Z	E	J	O	T	Y	D

明文是　AFF I NECI P HER
密文就是 IHHWVCSWFRCP

線上網站
https://wtool.com.cn/affine.html

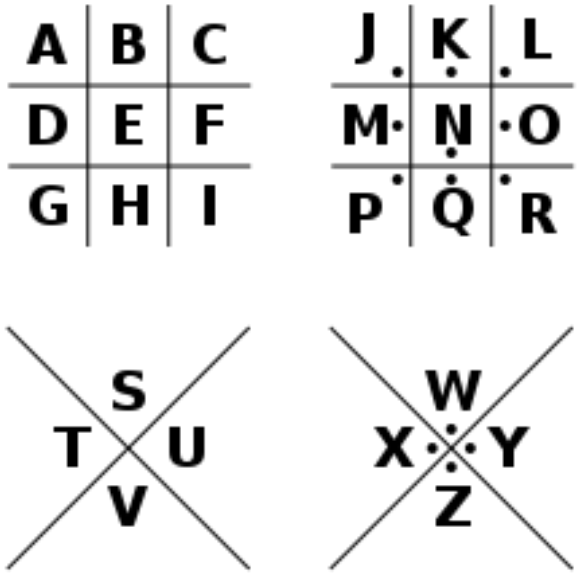
柵欄密碼

加密原理

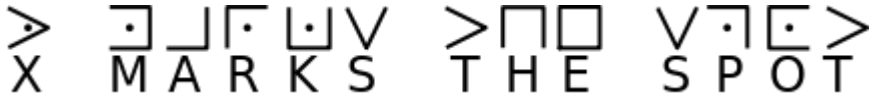
①把將要傳遞的資訊中的字母交替排成上下兩行。
②再將下麵一行字母排在上面一行的後邊，從而形成一段密碼。
③例如:
明文：THE LONGEST DAY MUST HAVE AN END
加密:
1、把將要傳遞的資訊中的字母交替排成上下兩行。
T E O G S D Y U T A E N N
H L N E T A M S H V A E D
2、 密文:
將下麵一行字母排在上面一行的後邊。
TEOGSDYUTAENN HLNETAMSHVAED
解密:
先將密文分為兩行
T E O G S D Y U T A E N N
H L N E T A M S H V A E D
再按上下上下的順序組合成一句話
明文：THE LONGEST DAY MUST HAVE AN END

豬圈密碼

豬圈密碼(Pigpen Cipher或稱九宮格密碼、朱高密碼、共濟會密碼或共濟會員密碼)，是一種以格子為基礎的簡單替代式密碼。



我們舉一個例子，如明文為 X marks the spot，那麼密文如下

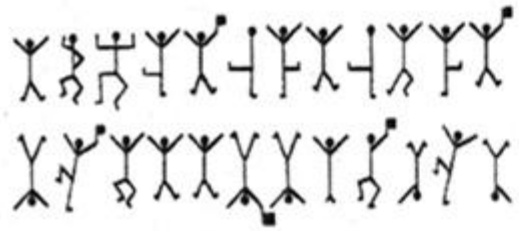


跳舞的小人

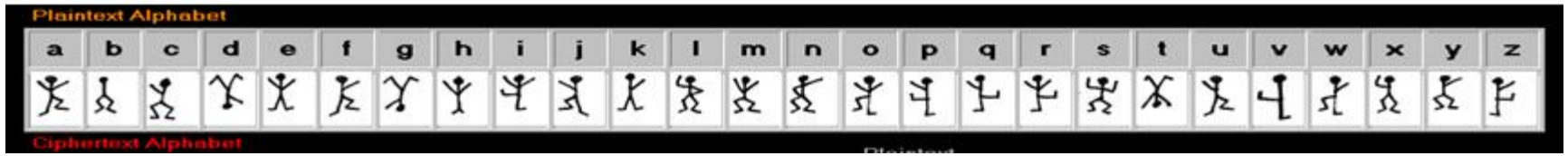
criminal's message (1)

criminal's message (2)

Elsie's reply



criminal's message (3)

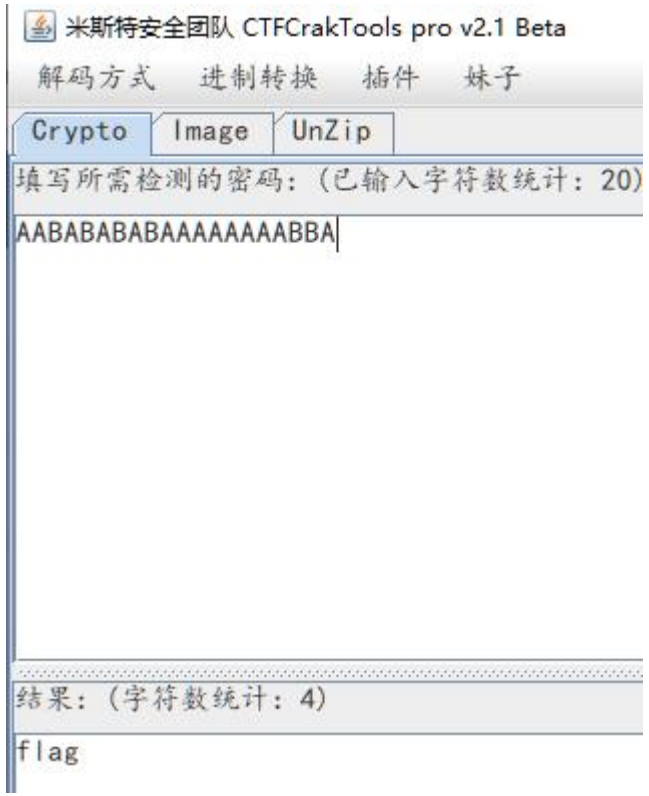


培根密碼

原理

培根密碼使用兩種不同的字體，代表 A 和 B，結合加密表進行加解密。

a	AAAAA	g	AABBA	n	ABBAA	t	BAABA
b	AAAAB	h	AABBB	o	ABBAB	u-v	BAABB
c	AAABA	i-j	ABAAA	p	ABBBA	w	BABAA
d	AAABB	k	ABAAB	q	ABBBB	x	BABAB
e	AABAA	l	ABABA	r	BAAAA	y	BABBA
f	AABAB	m	ABABB	s	BAAAB	z	BABBB



下麵這一段內容是明文 steganography 加密後的內容，正常字體是 A，粗體是 B:

To en**co**de **a** mes**s**age ea**ch** letter **of** the **pl**ain**te**x**t** **i**s replaced b**y** **a** gro**up** **of** **f**i**ve** of **th**e letters **'A'** o**r** **'B'**.

培根密碼

明文: I LOVE YOU

初始密文: ABAAA ABABBABBBABABABAABAA BBAAAABBBABABAA

隨意選一段話, 要求正好是 40 個字元。

When will you be home? Your wife is waiting for you.(不考慮空格和標點符號)

對照初始密文, 將大寫字母看作 A, 小寫字母看作 B, 將上面這段話變換為最終密文

WhEN WIIL yoU be hOmE? YOuR WiFE is WAITing FoR yoU.

或者是將粗體字看作 B, 正常字體看作 A, 變換為最終密文

線上網站:
http://rumkin.com/tools/cipher/baconian.php

棋盘密碼

棋盘密码（Polybius）

- 加密对象：小写字母
- 原理：
  - 棋盘密码是一种查表加密法，密码表如下：

	1	2	3	4	5
1	a	b	c	d	e
2	f	g	h	i,j	k
3	l	m	n	o	p
4	q	r	s	t	u
5	v	w	x	y	z

  - 密文就是字符在密码表里面对应的横纵坐标，如"a"加密为"11"，"y"加密为"54"
- 特点：
  - 数字没两个一组
  - 数字范围为1~5

明文：HELLO 密文：23 15 31 31 34

棋盤密碼

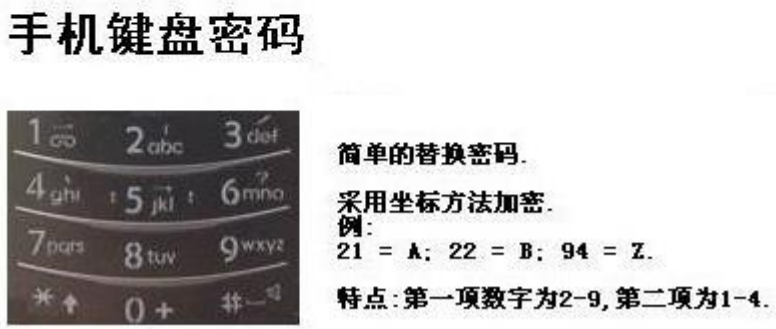
	A	D	F	G	X
A	b	t	a	l	p
D	d	h	o	z	k
F	q	f	v	s	n
G	g	i/j	c	u	x
X	m	r	e	w	y

明文：HELLO 密文：DD XF AG AG DF

鍵盤密碼

手机键盘密码

手机键盘加密方式，是每个数字键上有 3-4 个字母，用两位数字来表示字母，例如：ru 用手机键盘表示就是：7382，那么这里就可以知道了，手机键盘加密方式不可能用 1 开头，第二位数字不可能超过 4，解密的时候参考此



鍵盘密碼

电脑键盘棋盘

电脑键盘棋盘加密，利用了电脑的棋盘方阵。

	1	2	3	4	5	6	7	8	9	10	11	12	13	14
1	~	!	@	#	$	%	^	&	*	(	)	_	+	|
2	`	1	2	3	4	5	6	7	8	9	0	-	=	\
3	Q	W	E	R	T	Y	U	I	O	P	{	}		
4	q	w	e	r	t	y	u	i	o	p	[	]		
5	A	S	D	F	G	H	J	K	L	:	"			
6	a	s	d	f	g	h	j	k	l	;	'			
7	Z	X	C	V	B	N	M	<	>	?				
8	z	x	c	v	b	n	m	,	.	/				

键盘密码加密的原理同棋盘密码，只是利用了键盘作为方阵。

例：

密文：
87 34 112
55 47 87 410

明文：
mR_Gump

鍵盤密碼

电脑键盘坐标

电脑键盘坐标加密，利用键盘上面的字母行和数字行来加密，例：bye 用电脑键盘 XY 表示就是：
351613

鍵盤密碼

电脑键盘 QWE

电脑键盘 QWE 加密法，就是用字母表替换键盘上面的排列顺序。

二. QWE=ABC
即把键盘上的字母按顺序对应ABC.

注意:红色的为明码(即你手中的密码)
黑色的就是对应的密码了.

進製錶示

二進位

最近二進位就是電腦常用的進制，即逢二進一。例如：1010。

八進制

八進制即逢八進一。例如：626。

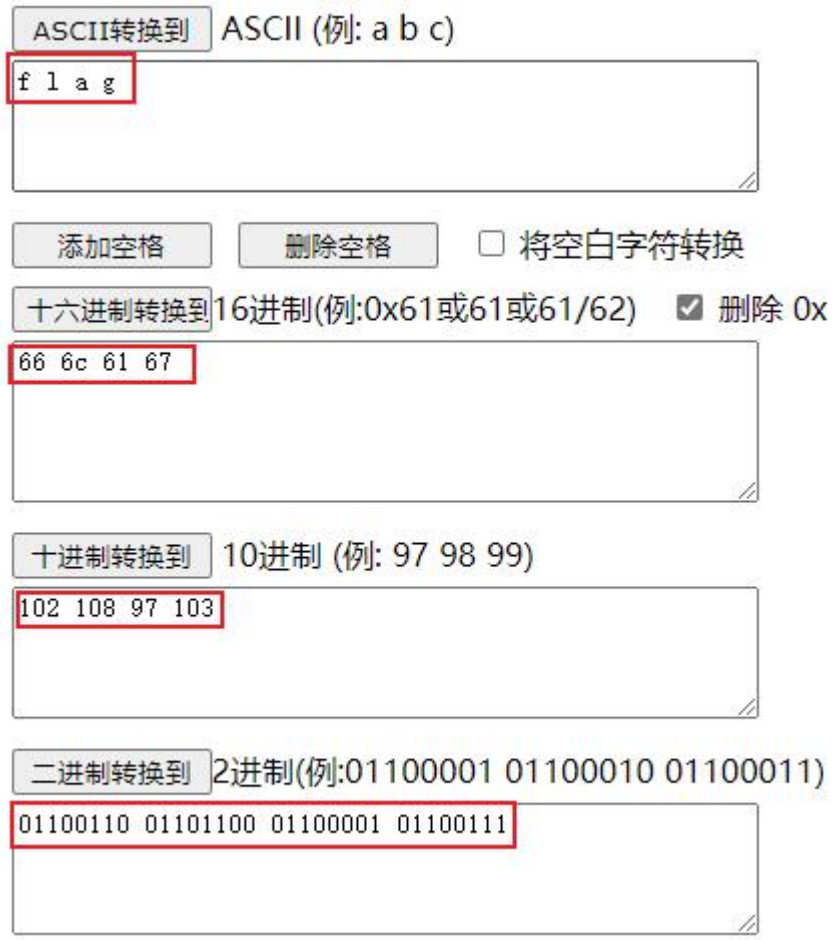
十進位

十進位就是我們在計算中常用的進制，所以就不再舉例（即逢十進一）。

十六進制

十六進制與其它進制有所不同，在10到15用英文字母進行表示。

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F



ASCII

ASCII表

(American Standard Code for Information Interchange 美国标准信息交换代码)

高四位 / 低四位		ASCII控制字符												ASCII打印字符																	
		0000						0001						0010		0011		0100		0101		0100		0111							
		0						1						2		3		4		5		6		7							
		十进制	字符	Ctrl	代码	转义字符	字符解释	十进制	字符	Ctrl	代码	转义字符	字符解释	十进制	字符	十进制	字符	十进制	字符	十进制	字符	十进制	字符	十进制	字符	十进制	字符	Ctrl			
0000	0	0		^@	NUL	\0	空字符	16	▶	^P	DLE		数据链路转义	32		48	0	64	@	80	P	96	`	112	p						
0001	1	1	☺	^A	SOH		标题开始	17	◀	^Q	DC1		设备控制 1	33	!	49	1	65	A	81	Q	97	a	113	q						
0010	2	2	☻	^B	STX		正文开始	18	↕	^R	DC2		设备控制 2	34	"	50	2	66	B	82	R	98	b	114	r						
0011	3	3	♥	^C	ETX		正文结束	19	‼	^S	DC3		设备控制 3	35	#	51	3	67	C	83	S	99	c	115	s						
0100	4	4	♦	^D	EOT		传输结束	20	¶	^T	DC4		设备控制 4	36	$	52	4	68	D	84	T	100	d	116	t						
0101	5	5	♣	^E	ENQ		查询	21	§	^U	NAK		否定应答	37	%	53	5	69	E	85	U	101	e	117	u						
0110	6	6	♠	^F	ACK		肯定应答	22	▬	^V	SYN		同步空闲	38	&	54	6	70	F	86	V	102	f	118	v						
0111	7	7	•	^G	BEL	\a	响铃	23	↨	^W	ETB		传输块结束	39	'	55	7	71	G	87	W	103	g	119	w						
1000	8	8	◘	^H	BS	\b	退格	24	↑	^X	CAN		取消	40	(	56	8	72	H	88	X	104	h	120	x						
1001	9	9	○	^I	HT	\t	横向制表	25	↓	^Y	EM		介质结束	41	)	57	9	73	I	89	Y	105	i	121	y						
1010	A	10	◙	^J	LF	\n	换行	26	→	^Z	SUB		替代	42	*	58	:	74	J	90	Z	106	j	122	z						
1011	B	11	♂	^K	VT	\v	纵向制表	27	←	^[	ESC	\e	溢出	43	+	59	;	75	K	91	[	107	k	123	{						
1100	C	12	♀	^L	FF	\f	换页	28	∟	^\	FS		文件分隔符	44	,	60	<	76	L	92	\	108	l	124	|						
1101	D	13	♪	^M	CR	\r	回车	29	↔	^]	GS		组分隔符	45	-	61	=	77	M	93	]	109	m	125	}						
1110	E	14	♫	^N	SO		移出	30	▲	^^	RS		记录分隔符	46	.	62	>	78	N	94	^	110	n	126	~						
1111	F	15	☼	^O	SI		移入	31	▼	^-	US		单元分隔符	47	/	63	?	79	O	95	_	111	o	127	⌂	^Backspace 代码：DEL					

注：表中的ASCII字符可以用“Alt + 小键盘上的数字键”方法输入。 制作：MHL QQ:1208980380 2013/08/08

文本	ASCII码
key	107 101 121
KEY	75 69 89
flag	102 108 97 103
FLAG	102 108 97 103
ctf	99 116 102
CTF	67 84 70

中文編碼

◆擴展ASCII碼：包含ASCII中已有的128個字元，又增加了128個字元，非國際標準。

GB2312編碼：用2個位元組來表示一個漢字。

對ASCII中原有的字元也按照兩個位元組進行重新編碼，稱為全角字元；

原先的ASCII字元稱為半角字元。

GBK編碼：在GB2312的基礎之上新加了20000多個字元（包括繁體字）。

GB18030編碼：在GBK的基礎之上又新加了很多少數民族的字元。

BIG5碼：臺灣和香港使用的另外一套漢字編碼方案。

◆要打開一個文本，必須知道它的編碼方式，如果用錯誤的編碼方式解讀，就會出現亂碼。

Unicode編碼

Unicode又稱為統一碼、萬國碼、單一碼，是國際組織制定的旨在容納全球所有字元的編碼方案，包括字元集、編碼方案等，它為每種語言中的每個字元設定了統一且唯一的二進位編碼，以滿足跨語言、跨平臺的要求。

```
编码示例:
明文: hello
四种编码方式:
&#x [Hex]: &#x0068;&#x0065;&#x006C;&#x006C;&#x006F;
&# [Decimal]: &#00104;&#00101;&#00108;&#00108;&#00111;
\U [Hex]: \U0068\U0065\U006C\U006C\U006F
\U+ [Hex]: \U+0068\U+0065\U+006C\U+006C\U+006F
```

線上編碼解碼：
http://www.mxcz.net/tools/Unicode.aspx

Html編碼

編碼格式：形如a這種

```
&#102;&#108;&#103;&#121;&#117;&#97;&#32;&#121;
```

特殊符号	命名实体	十进制编码	特殊符号	命名实体	十进制编码	特殊符号	命名实体	十进制编码
A	Α	Α	B	Β	Β	Γ	Γ	Γ
Δ	Δ	Δ	E	Ε	Ε	Z	Ζ	Ζ
H	Η	Η	Θ	Θ	Θ	I	Ι	Ι
K	Κ	Κ	Λ	Λ	Λ	M	Μ	Μ
N	Ν	Ν	Ξ	Ξ	Ξ	O	Ο	Ο
Π	Π	Π	P	Ρ	Ρ	Σ	Σ	Σ
T	Τ	Τ	Y	Υ	Υ	Φ	Φ	Φ

URL編碼

url編碼又叫百分號編碼，是統一資源定位(URL)編碼方式。

URL地址（常說網址）規定了常用地數字，字母可以直接使用，另外一批作為特殊用戶字元也可以直接用（/,:@等），剩下的其他所有字元必須通過%xx編碼處理。

```
1 原链接：http://www.mzf.com/?login=123
2 编码后：http%3a%2f%2fwww.mzf.com%2f%3flogin%3d123
3 明文：睡觉
4 编码后：%e7%9d%a1%e8%a7%89
```

Base64

BASE64是一種編碼方式，通常用於把二進位數據編碼為可寫的字元形式的數據。這是一種可逆的編碼方式。

編碼後的數據是一個字串，其中包含的字元為：
A-Z、a-z、0-9、+、/，共64個字元：26 + 26 + 10 + 1 + 1 = 64。

Base64 索引表

数值	字符	数值	字符	数值	字符	数值	字符
0	A	16	Q	32	g	48	w
1	B	17	R	33	h	49	x
2	C	18	S	34	i	50	y
3	D	19	T	35	j	51	z
4	E	20	U	36	k	52	0
5	F	21	V	37	l	53	1
6	G	22	W	38	m	54	2
7	H	23	X	39	n	55	3
8	I	24	Y	40	o	56	4
9	J	25	Z	41	p	57	5
10	K	26	a	42	q	58	6
11	L	27	b	43	r	59	7
12	M	28	c	44	s	60	8
13	N	29	d	45	t	61	9
14	O	30	e	46	u	62	+
15	P	31	f	47	v	63	/

Base32

Base32編碼是使用32個可列印字元（字母A-Z和數字2-7）對任意位元組數據進行編碼的方案，編碼後的字串不用區分大小寫並排除了容易混淆的字元，可以方便地由人類使用並由電腦處理。

RFC 4648 Base32 字母表

值	符号	值	符号	值	符号	值	符号
0	A	8	I	16	Q	24	Y
1	B	9	J	17	R	25	Z
2	C	10	K	18	S	26	2
3	D	11	L	19	T	27	3
4	E	12	M	20	U	28	4
5	F	13	N	21	V	29	5
6	G	14	O	22	W	30	6
7	H	15	P	23	X	31	7
填充	=						

Base16

Base16編碼使用16個ASCII可列印字元（數字0-9和字母A-F）對任意位元組數據進行編碼。Base16先獲取輸入字串每個位元組的二進位值（不足8比特在高位補0），然後將其串聯進來，再按照4比特一組進行切分，將每組二進位數分別轉換成十進位，在下述表格中找到對應的編碼串接起來就是Base16編碼。可以看到8比特數據按照4比特切分剛好是兩組，所以Base16不可能用到填充符號“=”。

Base16 编码表

值	编码	值	编码
0	0	8	8
1	1	9	9
2	2	10	A
3	3	11	B
4	4	12	C
5	5	13	D
6	6	14	E
7	7	15	F

三者的區別

1、BASE16是由大寫字母(A-F)、和數字(0-9)組成
2、BASE32是由大寫字母(A-Z)、數字(2-7)組成
3、BASE64是由大寫字母(A-Z)、小寫字母(a-z)、數字(0-9)、字元組成。

```
Base16字符表0123456789ABCDEF
Base32字符表ABCDEFGHIJKLMNOPQRSTUVWXYZ234567
Base64字符表ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789+/
```

Base家族其他編碼

base58
base58的編碼表相比base64少了數字0，大寫字母I，O，小寫字母 l (這個是L），以及符號 ‘+’和 ‘/’

base91
base91的密文由91個字元（0-9，a-z，AZ,!#$%&()*+,./:;<=>?@[]^_`{|}~”）組成

base100
Base100編碼/解碼工具（又名：Emoji表情符號編碼/解碼），可將文本內容編碼為Emoji表情符號；同時也可以將編碼後的Emoji表情符號內容解碼為文本。 **

摩斯密碼

摩爾斯電碼也被稱作摩斯密碼，是一種時通時斷的信號代碼，通過不同的排列順序來表達不同的英文字母、數字和標點符號。它發明於1837年，是一種早期的數位化通信形式。不同於現代化的數字通訊，摩爾斯電碼只使用零和一兩種狀態的二進位代碼，它的代碼包括五種：短促的點信號“·”，讀“滴”（Di）保持一定時間的長信號“—”，讀“嗒”（Da）表示點和劃之間的停頓、每個詞之間中等的停頓，以及句子之間長的停頓