BASEDPORT

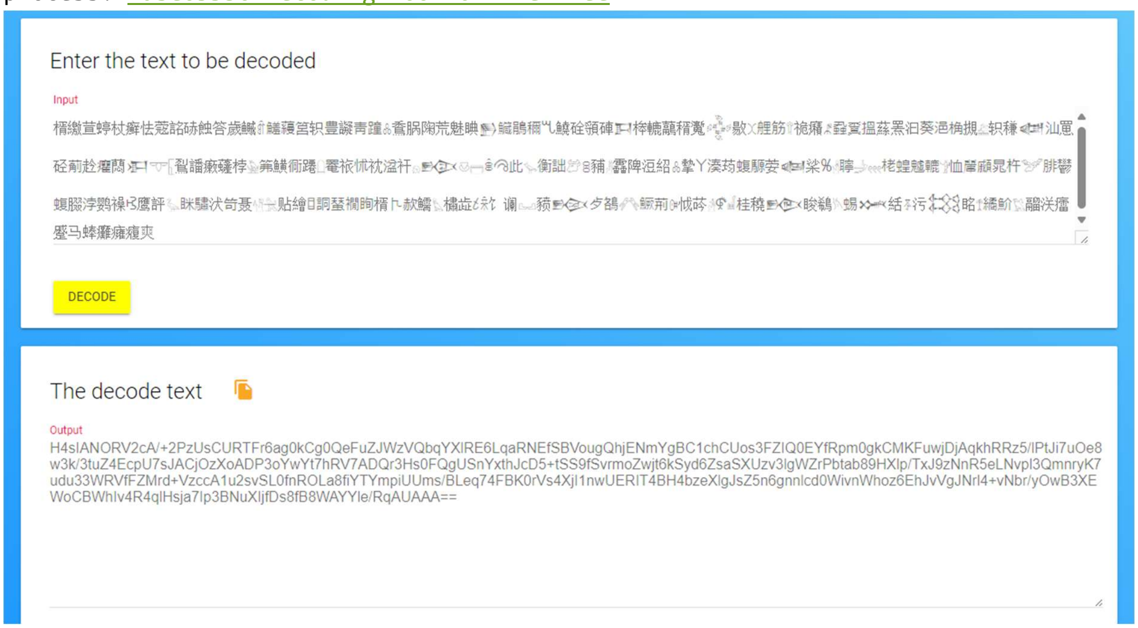
Description:

Thats a lot of Based Ports!

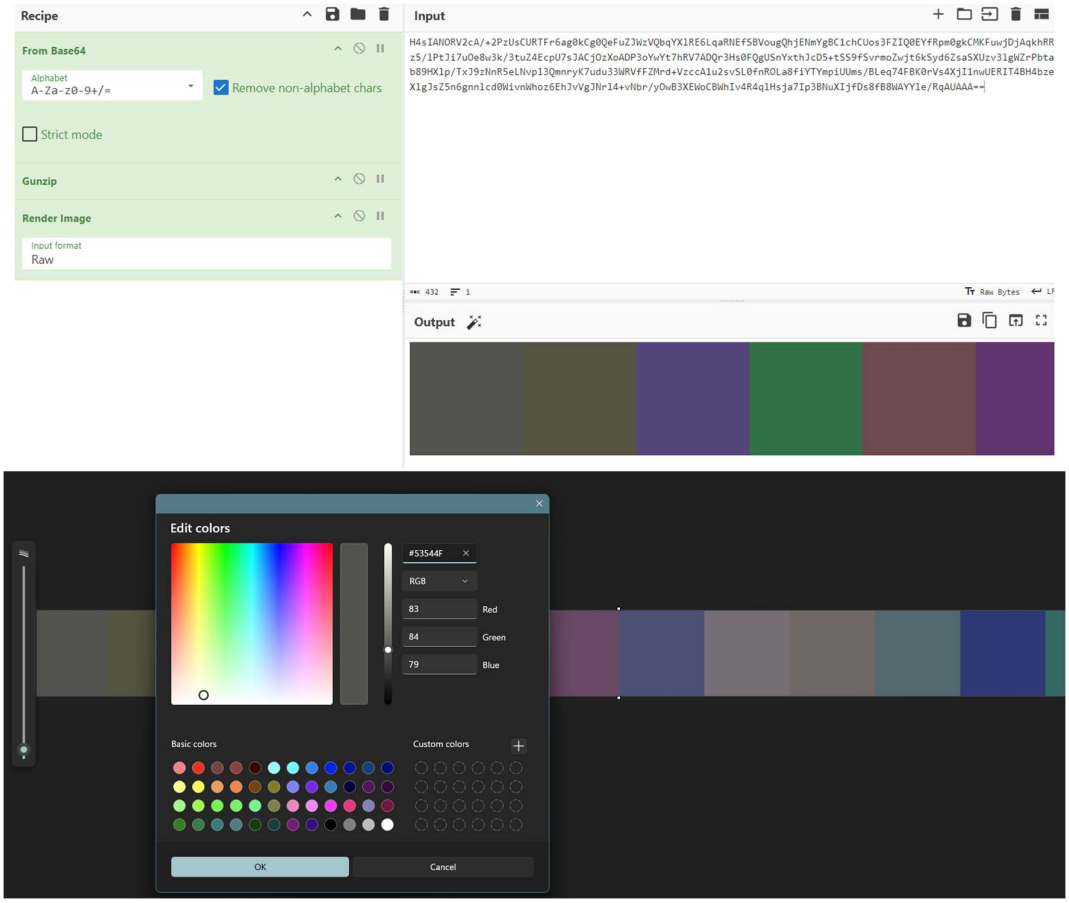
Basedport.txt:

椁緻萓崞杖癬怯蔻詻硤蝕答歲鱖[illegible]鎈穫筥轵豊譺靑蹱[illegible]喬朒陶荒魅睓[illegible]鱵鵑稱[illegible]饒硂頻碑[illegible]栙轆藹
稍虋[illegible]敫[illegible]艃筋[illegible]裞癪[illegible]蝨[illegible]搵蕬羆汨葵浥桷摫[illegible]轵穖[illegible]汕罳硁萷趁癯蔄[illegible]駑譒瘷蘾梈[illegible]箳鱑
衕躧[illegible]罨袨怵祧溢衦[illegible]此[illegible]衡詘[illegible]豧[illegible]靍陴逗紹[illegible]摯[illegible]湊玝螋驅荌[illegible]桬[illegible]聹[illegible]栳蝗
鷈轆[illegible]恤巃顪晁杵[illegible]腓鬱螋餟涍鸚褬[illegible]鷹評[illegible]眯驌沃笴聂[illegible]貼繪[illegible]詷蝵襉昫椁[illegible]款鱬[illegible]槵齸[illegible]讕
[illegible]豮[illegible]夕鴾[illegible]鱖荊[illegible]怴莳[illegible]桂穘[illegible]睃鶬[illegible]蜴[illegible]絬[illegible]污[illegible]晗[illegible]繘魪[illegible]鬸渓瘤盬马蜯癱瘫
瘦爽

I Googled to find out what kind of encoding uses Chinese characters and discovered it was Base 65536. I found a cool website to decode it and proceeded with the process: Base65536 Decoding Tool Online Free



After pasting the encoded text, I noticed more encoded characters with a pattern similar to **Base64**. I immediately decoded it using ***CyberChef***, where the magic features suggested a file type detection. It turned out to be an image, indicating that the additional encoded text was actually within the image.



Given my experience with pixel-based encoding from multiple CTFs, I decided to try my first method to extract the color codes. I wrote a script to automate the process, allowing me to decode the information efficiently.

Script

```
color_codes = [
    "#53544F", "#555443", "#54467B", "#327147", "#6E4A50",
    "#61336F", "#6A4966", "#4C5275", "#776D75", "#6E6967",
    "#52686F", "#313976", "#336A63", "#61397D"
]

def hex_to_chars(hex_code):
    hex_code = hex_code.lstrip('#')
    chars = []
    for i in range(0, len(hex_code), 2):
        hex_pair = hex_code[i:i+2]
        chars.append(chr(int(hex_pair, 16)))
    return ''.join(chars)

decoded_values = [hex_to_chars(code) for code in color_codes]
print(decoded_values)
```

```
┌──(osiris㉿ALICE)-[~/Downloads/CTF/STOUTCTF/Misc/BasedPort]
└─$ python solver.py
['STO', 'UTC', 'TF{', '2qG', 'nJP', 'a3o', 'jIf', 'LRu', 'wmu', 'nig', 'Rho', '19v', '3jc', 'a9}']
```

Flag	STOUTCTF{2qGnJPa3ojIfLRuwmunigRho19v3jca9}