

CTF Write-Up: Morse Code Challenge

Challenge Description:

In this challenge, we were tasked with decoding a message encoded in Morse code and converting the decoded message into a specific format. The final flag needed to be wrapped in `picoCTF{}` and follow specific formatting rules:

1. Convert all letters to lowercase.
2. Replace spaces with underscores (`_`).
3. Wrap the resulting message in `picoCTF{}`.

Steps to Solve:

1. **Decoding the Morse Code:** First, we downloaded the provided audio file containing the Morse code. To decode the audio, we used an online tool: [International Morse Decoders](#). This tool allowed us to input the audio and automatically converted the Morse signals into readable text.

The decoded message we obtained from the tool was:

```
WH47 H47H 90D W20U9H7
```

2. **Formatting the Decoded Message:** To convert this decoded message into the required flag format, we wrote a short Python script that:
 - Converts all characters to lowercase.
 - Replaces spaces between words with underscores.
 - Wraps the entire message in `picoCTF{}`.

Here's the Python code we used:

```
def format_morse_code(decoded_message):
    # Convert the message to lowercase
    lower_case_message = decoded_message.lower()

    # Replace spaces with underscores
    formatted_message = lower_case_message.replace(' ', '_')

    # Wrap the formatted message with picoCTF{}
    flag = f"picoCTF{{{formatted_message}}}"

    return flag

# The decoded Morse code message
decoded_message = "WH47 H47H 90D W20U9H7"

# Get the formatted flag
formatted_flag = format_morse_code(decoded_message)

# Print the result
```

```
print(formatted_flag)
```

This script takes the decoded message, formats it correctly, and prints the final flag.

3. **Final Flag:** After running the Python script, we obtained the correctly formatted flag:

```
Copy code  
picoCTF{wh47_h47h_90d_w20u9h7}
```

Conclusion:

By leveraging the **International Morse Decoder** to decode the Morse code and then using a simple Python script to format the result, we were able to quickly solve this challenge. The key steps involved recognizing the structure of Morse code, using the appropriate tool for decoding, and crafting a Python script to apply the necessary formatting rules for the flag.