**Description**

**Message to Agent:** We've intercepted two files: one claims to hold the solution, while the other seems like a chaotic mess filled with random S, T, and whitespace characters. It looks like nonsense, but there's something valuable hidden in there—an important token that our boss, Jason, needs to retrieve from the website. Your mission: crack it and extract the information. Be careful; direct brute forcing attacks on the website are off-limits. Get to work, Agent, and keep it clean!

**SOLUTION**

User gets the txt file.

Decode whitespace.

Decode Ook!

Use the username and password to login into the website.

They log into the website and see the JWT.

They Must now use hashcat with a list to get the secret key using the most famous jwt wordlist

<https://github.com/wallarm/jwt-secrets/blob/master/jwt.secrets.list>

The secret key is SecretKey=276c628ce33a59b398e4bc8f961874f454c57cfcc77cd0c27e4efc429095bd52945f465786371701f96ec5782d9379b287fe8aa6b5d00ce47f65f89fd12bc665

With the secret key they must change the JWT and add a valid signature to then be able to get into the /flag.

They must have their username as StarLight

The flag is there

SolvingJWTwithTOOLS\_isCool2024

FLAG: shadowCTF{SolvingJWTwithTOOLS\_isCool2024}

**Second solution with SOLUTION.TXT FILE**

The flag is encoded with:

Vigenere, ROT47,base64, brainfuck,alphuck,to hex, URL encoded, LZstring base64 compress

HAVE FUN!!!!

The flag becomes.

