Web Security and Malware Analysis
Notes on JavaScript De-Obfuscation

Some useful notes on how to de-obfuscate JavaScript:

First, try to understand which functions are called and what calls them.

For example, in assignment number 4 you have three functions:

1 - b(c,d) (careful on the JavaScript notation)
2 - authenticate_bau(c)
3 - d(e,f)

The flow of the program starts calling **authenticate_bau**.

The second step is finding out which brackets are employed, and how they are distributed. This technique is useful to determine where specific functions start and end.

In this case, function **d** ends right after the **return g;** statement, while **authenticate_bau** ends at the very end of the code.

Your goal is analyzing **authenticate_bau**. Hence, the function does two things:

- It defines the function **d(e,f)**
- It calls the function d on c (that is the argument of **authenticate_bau,** i.e., the string the user puts) and number 8.
- It compares the result of the call with the function **b()**

What does the function **b** do? You should be able to answer this if you consider the relationship between **b** and **a** (careful: if in JavaScript one parameter is not used in a function, the parameter itself is ignored).

Now, it is essential to understand what the function **d** does. The best strategy is moving line by line.

This methodology will be very useful also in assignment 5 (which is even easier).