ReDoS Regular Expressions DoS

Description

The Regular expression Denial of Service (ReDoS (https://www.owasp.org/index.php/Regular_expression_Denial_of_Service_-_ReDoS)) is a Denial of Service attack, that exploits the fact that most Regular Expression implementations may reach extreme situations that cause them to work very slowly (exponentially related to input size). An attacker can then cause a program using a Regular Expression to enter these extreme situations and then hang for a very long time.

Attack Mechanics

When untrusted data input is executed on a regex pattern, it may exploit vulnerable patterns into running long calculations to match for a given string. For Node.js this is extremely important due to the single-threaded event-loop architecture which means that the main Node.js process is blocked from serving any other requests.

How Do I Prevent It?

1. Avoid writing your own regular expressions
2. Use Node.js's validator.js (https://github.com/chriso/validator.js/) package to validate expected data format instead of writing your own regular expressions
3. As a last resort of writing your own regex patterns you can utilize Node.js's safe-regex (https://github.com/substack/safe-regex) package which allows detecting if a regex is prone to catastrophic backtracking, and also allows to configure threshold for maximum repetitions.

Source Code Example

Even simple regex patterns are vulnerable to ReDoS. The NodeGoat project uses the following source code to validate text format from the user based on a regex pattern:

```
// Allow only numbers with a suffix of #, for example: 'XXXXXX#'
var regexPattern = /([0-9]+)+\#/;
var testComplyWithRequirements = regexPattern.test(bankRouting)
```

If a long enough input is provided it will stall the Node.js process and render it useless (in the background the Node.js process will take 100% cpu until stopped or the regex yields a result (true or false)). Try to input the following string in the Bank Routing number in the Profile form:

```
917626121176121211231231231231 21
```