## **Regular Expression Denial of Service (ReDoS)**

Suppose we have a user that submits benign input to an API. On the server side, a developer could match any input against a regular expression. After a usually constant amount of time, the API responds. In some instances, an attacker may be able to cause significant delays in the API's response time by submitting a crafted payload that tries to exploit some particularities/inefficiencies of the regular expression matching engine. The longer this crafted payload is, the longer the API will take to respond. Exploiting such "evil" patterns in a regular expression to increase evaluation time is called a Regular Expression Denial of Service (ReDoS) attack.

Let us assess an API that is vulnerable to ReDoS attacks together.

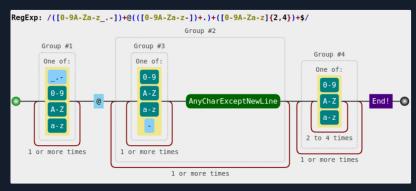
Proceed to the end of this section and click on Click here to spawn the target system! or the Reset Target icon. Use the provided Pwnbox or a local VM with the supplied VPN key to reach the target application and follow along.

The API resides in http://<TARGET IP>:3000/api/check-email and accepts a parameter called email.

Let's interact with it as follows.



Submit the above regex to regex101.com for an in-depth explanation. Then, submit the above regex to https://jex.im/regulex/ for a



Let's submit the following valid value and see how long the API takes to respond.



You will notice that the API takes several seconds to respond and that longer payloads increase the evaluation time.

The difference in response time between the first cURL command above and the second is significant.

The API is undoubtedly vulnerable to ReDoS attacks.

(1) Terminate Pwnbox to switch location

