

Hotelbeds Suppliers Integrations Developer TEST

Instructions

Upload the code into a remote repository and provide access (public permissions; if it's not possible, provide the link and we will request access)

Complete the tests

You can do it in any language (Java 8 preferred).

Try to use a dependency manager.

Try to use a framework in your application (Spring framework preferred).

Avoid to upload binary/compiled files or files created by IDE's

TDD is a plus.

Ensure that you uploaded all your code.

Have fun!

Hacker detection system

A system provided by a company allows customers to sign in using their username and password. There is a requirement for an automated system to be developed to help identify attempts to hack the system and compromise accounts. Activity log files are recorded and the new system will need to process these logs to identify suspicious activity.

Write a Java program implementing the `HackerDetector` interface (outlined below) which defines a single method 'parseLine'. The method should take one line of the log file at a time and return the IP address if any suspicious activity is identified or null if the activity appears to be normal.

```
package com.hotelbeds.supplierintegrations.hackertest.detector;
public interface HackerDetector {
    String parseLine(String line);
}
```

The parseLine method will be called each time a new log line is produced.

The log lines will be in the following format:

ip,date,action,username

IP look like 80.238.9.179 Date is in the epoch format like 1336129471 Action is one of the following: SIGNIN_SUCCESS or SIGNIN_FAILURE Username is a String like Will.Smith

A log line will therefore look like this: 80.238.9.179,133612947,SIGNIN_SUCCESS,Will.Smith

The first detection method will be to identify a single IP address that has attempted a failed login 5 or more times within a 5 minute period. On detection you should return the suspicious IP.

Our signin page can generate around 100,000 failed signing a day so memory consumption should be considered and managed.

Please bear in mind all the best practices you would normally employ when producing "done" production code:

- A well factored object oriented domain model

- Testing

- Clean code

Time Calculation

Write a function that returns the number of minutes (rounded down) between two timestamps `time1` and `time2` in RFC 2822 format (ie: Thu, 21 Dec 2000 16:01:07 +0200). Don't forget about the time zones.