UNIVERSITY of York

MSc/MEng/MMath Degree Examinations 2018/19

DEPARTMENT OF COMPUTER SCIENCE

Topics in Privacy & Security (PSEC)
Open Individual Assessment

Issued: Wednesday, 6th February, 2019

Submission due: 12 noon, Wednesday 13th March, 2019

Feedback and marks due: Wednesday 10th April, 2019

All students should submit their answers through the electronic submission system:

http://www.cs.york.ac.uk/student/assessment/submit/

by **12 noon, Wednesday 13th March, 2019**. An assessment (or part of an assessment) submitted after this deadline will be marked initially as if it had been handed in on time, but the Board of Examiners will normally apply a lateness penalty to the whole assessment.

The feedback and marks date is guided by departmental policy but, in exceptional cases, there may be a delay. In these cases, all students expecting feedback will be emailed by the module owner with a revised feedback date. The date that students can expect to see their feedback is published on the module descriptor:

https://www.cs.york.ac.uk/modules/psec.html

Your attention is drawn to the section about Academic Misconduct in your Departmental Handbook:

https://www.cs.york.ac.uk/student/handbook/

Any queries on this assessment should be addressed to Dr Radu Calinescu, radu.calinescu@york.ac.uk

Answers that apply to all students will be posted on the PSEC VLE page.

Rubric

Answer all questions. Note the page limits for each question. Parts of answers that go beyond the page limit may not be marked. References must be listed at the end of the document and do not count towards page limits.

Your exam number should be on the front cover of your assessment. You should not be otherwise identified anywhere on your submission.

General Marking Criteria

You will be given credit for clear and concise descriptions, and for organising your assessment into well-defined, meaningful sections.

Demonstration of appropriate research is expected for all parts of the assessment. You should cite relevant published work to support your statements and arguments throughout your assessment. Failure to demonstrate such research will lead to a loss of marks.

Question 1: Online Reputation Systems [60 marks]

Business travellers and holidaymakers use online review systems like TripAdvisor to inform their choices of hotels, restaurants and other types of travel-related venues. However, there is a growing concern that many of the reviews posted on *online reputation systems* are fake [1,2], and – in the case of review systems like TripAdvisor – have a significant negative impact on the travel industry and its customers.

As a cyber security consultant, you are hired by the new company GENuine REviews (GENRE) to propose an approach they should follow to limit the fraction of fake hotel, restaurant and museum reviews on their future online travel website.

You must write a report detailing and justifying your proposed approach to the GENRE online travel website. Your report should cover the following aspects:

(i) **[15 Marks]** An analysis of the types of cyberattacks that may lead to fake hotel, restaurant and museum reviews ending up on a travel website.

(ii) **[15 Marks]** An approach that GENRE can use to significantly limit the fraction of fake reviews due to the cyberattacks from part (i) of the question, assuming that all reviews are posted by anonymous users (i.e., users who are not required to set up an account on the GENRE system).

(iii) **[20 Marks]** An enhanced variant of the approach from part (ii) of the question for the scenario where GENRE only accepts reviews from registered users (i.e., users who are required to set up a GENRE account, and to sign into their accounts in order to submit reviews).

(iv) **[10 Marks]** A comparison of the advantages and disadvantages of your approach variants from parts (ii) and (iii) of the question, and a recommendation as to which variant GENRE should adopt and why. A hybrid approach that combines the two variants is acceptable if appropriately justified.

Further Marking Guidance. You are expected to identify relevant research literature and other trustworthy sources, and to use them appropriately to answer the above question parts. Answers without accompanying rationale and research, which reproduce information from other sources without extending and adapting it for the GENRE system, or which do not consider alternative solutions will not gain high marks.

Your answer to this question must not exceed six A4 pages (minimum font size 11pt) plus references.

References

1. Paul Resnick, Ko Kuwabara, Richard Zeckhauser and Eric Friedman, “Reputation Systems”. In: *Communications of the ACM,* vol. 43, issue 12, pages 45-48, December 2000. Available online (only from the university network) at http://dl.acm.org/citation.cfm?id=355122. Last accessed on 17 November 2018.

2. Kevin Hoffman, David Zage and Cristina Nita-Rotaru. A Survey of Attack and Defense Techniques for Reputation Systems, *ACM Computing Surveys*, vol. 42. no. 1, December 2009. Available online (only from the university network) at http://dl.acm.org/citation.cfm?id=1592452. Last accessed on 17 November 2017.

Question 2: Anomaly Detection [40 marks]

Many web servers use the *common log format* (CLF) below to generate a one-line log entry for each HTTP request they handle:

```
remotehost ident authuser [date] "request" status bytes
```

where the elements in the log entry have the following meanings:

`remotehost`	Fully qualified domain name of the remote host the client sent the request from, or its IP address.
`ident`	The identity information reported by the client, if available.
`authuser`	The username under which the user has authenticated, if available.
`[date]`	Date, time and time zone of the request.
`request`	The request line exactly as it came from the client.
`status`	The HTTP status code returned to the client.
`bytes`	The number of bytes in the object returned to the client, excluding all HTTP headers.

Log files containing CLF entries can be analysed to detect certain classes of security problems. This is typically achieved using *rule-based detection* techniques that look for patterns associated with known types of attacks, such as cross site scripting and code injection [1]. Multiple commercial and open-source tools exist that support this type of analysis.

Your task is to design and implement a complementary software tool that uses *anomaly detection* to identify potential causes of concern in CLF log files. The tool should take as input three command-line arguments:

- The name of a *training CLF log file* containing entries that correspond to “normal” (i.e., attack-free) web server activity;
- The name of a *test CLF log file* for a time period during which the web server might have been subjected to attacks.
- An *outlier coefficient* α (i.e., a positive number) that the tool should use to identify potential attacks in the test log file, as described below.

The tool should apply standard statistical analysis [2] to the entries from the training log file. This analysis should calculate the mean M and (sample) standard deviation s for the number of hourly accesses to each webpage listed in the training log file. The number of hourly webpage accesses is expected to differ significantly between

- early morning, i.e., 06:00-07:59
- daytime, i.e., 08:00-17:59
- evening, i.e., 18:00-21:59
- overnight, i.e., 22:00-05:59,

and between weekdays (i.e., Monday to Friday) and weekends (i.e., Saturday and Sunday). Therefore, different mean and standard deviation values should be calculated for each of these eight time of day/type of day combinations. After completing this analysis of the training log file, the tool should report as outliers (i.e., suspicious) the entries from the test log file that lie outside the interval $[M - \alpha s, M + \alpha s]$.

The tool output should be in the format:

[date] webpage num_accesses [M−αs,M+αs]

where *[date]* represents the start of the one-hour time period during which the web server received *num_accesses* requests for *webpage* instead of a number of requests in the interval [*M* – α*s*, *M* + α*s*]. As an example, the output line

```
[02/Nov/2018:09:00:00 0000] /secret.html 43 [16.78,32.12]
```

reports that the webpage `secret.html` was accessed `43` times between 9am and 9:59:59am on the 2nd of November 2018, instead of a number of accesses in the expected [*M* – α*s*, *M* + α*s*] interval, which is `[16.78,32.12]` in this case.

Write a report covering the following aspects of your development and use of the tool:

(i) **[15 marks]** Compare and contrast rule-based detection and anomaly detection using web server CLF log files, in terms of types of attacks they can identify, advantages and limitations.

(ii) **[10 marks]** Describe how the tool works (e.g., by means of pseudocode accompanied by a suitable description), analysing the time and space complexity of your log-based anomaly detection.

(iii) **[10 marks]** Sample `training_log` and `test_log` files containing synthetic data that correspond to normal and to unknown patterns of access to GNU Mailing List Manager (http://www.gnu.org/software/mailman/) web pages are available in the 'Assessment/2018-19 Open Assessment Files' area on the module VLE. Summarise the results of an experiment that uses your tool to detect anomalies in the log file `test_log`, using the log file `training_log` to learn the normal pattern of access for this web server. Discuss the differences between the anomaly detection results obtained using the outlier coefficient values α=1.0, α=2.0, α=3.0, α=4.0 and α=5.0.

(iv) **[5 marks]** Briefly discuss what types of web sites your tool might be suitable for. For what types of web sites is it unlikely to produce useful results?

Your report should not exceed four A4 pages (minimum font size 11pt) plus references.

You must attach as an appendix to your **single-file submission** the source code for the tool. If needed, use a ZIP archive to submit all components of the assessment as **a single file**.

Note: Use a programming language of your choice to implement the software tool.

References

[1] Roger Meyer, "Detecting Attacks on Web Applications from Log Files", SANS Institute, 2008. Available online at http://www.sans.org/reading-room/whitepapers/logging/detecting-attacks-web-applications-log-files-2074. Last accessed on 20 November 2018.

[2] Online Statistics Education: A Multimedia Course of Study (http://onlinestatbook.com/). Project Leader: David M. Lane, Rice University. Last accessed on 20 November 2018.