Cyber Security (CIS2201) - Assignment

Security Information and Event Management

Due date: 31/03/2017, Demonstration: 06/04/2017

Many Information Technology systems record security information and events in log files. These log files often contain valuable information which can be used to identify threats. The phrase *Security Information and Event Management (SIEM)* is used to define the field of auditing log files to make *sense* of the log data. There are some significant challenges facing SIEM. For example:

- New logging mechanisms are produced alongside new technology;
- SIEM analysts need to learn new log file formats as there is no standard; and
- The volume of information generated makes it almost impossible to make sense of the data without technological support.
- If an activity is identified within SIEM data that has legal implications, then it is necessary to take provisions to prevent modification.

Using a software architecture of your choice, you are to produce an SIEM application to help an analyst make sense of the security data. Your system must have the following high-level functionality:

1. Capability to load and hold any Comma Separated Value (CSV) file in memory upto the size of 200MB. A CSV file is a table representation, where a line of data is a row, and columns are separated by a delimiting comma (','). Some CSV files use difference delimiting values (e.g ';') and you may want to consider making your system sufficiently flexible.
2. Allow the user to group (categorise) data based upon the values provided in at least one of the columns. I.e. group the "name" column.
3. Based upon the categorisation, present the frequency information of each type to the user.
4. Create a mechanism to generate a hash sum for a provided data file and store it for future use.
5. A comparison function to determine when a file has changed based on its hash value.

For example, consider the following HTTP log entries:

```
0,tcp,http,REJ,0,0,0,0
0,udp,other,SF,146,0,0,0,0
1069,udp,other,SF,146,105,0,0,0,0
```

If the user wants to group data based upon values in column 2 (or 1 if you are counting from 0), your system must be able to communicate frequency information (`tcp` = 1 and `udp` = 2) in graphical form to the user.

You are required to submit:

1. A report detailing the design, development, testing and reflective analysis of your software; and
2. The source code and an Executable (where possible).

The software and executable will be used for marking purposed. A demonstration session of 10 minutes may be needed if the software can not easily be executed on the examiner's PC.

Before you start, please refer to the making criteria on the following pages.

Report Criteria

Student Name: **Student ID:** **Marker:**

Criteria	Marks Awarded	Comments
System specification **Proposed specification of your system** *Must have: Functionality, IDE and language* **Measured:** Detail and completeness	/10	
System design **UML Models and interface designs (wireframes, etc.)** *Must have: 1 class, 1 activity and 1 sequence diagram. 1 wireframe* **Measured:** Correctness of UML and quality of interface designs	/10	
Implementation note **Log of development challenges and fixes** *Must have: At least three detailed entries* **Measured:** Completeness and quality	/10	
Testing **Both functional and usability testing** *Must have: Unit test & functionality testing table* **Measured:** Completeness of testing	/10	
Reflective analysis **A personal reflection detailing the success of the project and what you would do different next time** *Must have: A reflective evaluation of 500 words in length* **Measured:** Level of detail and reflection	/10	
Total	/50	

Software Criteria

Student Name: **Student ID:** **Marker:**

Criteria	Marks Awarded	Comments
Loading of log files *Loading, parsing, and storing CSV data in memory* *Must have: Ability to process 100,000 lines of the "unlabeled" KDD dataset[1] in a 5 minute period.* **Measured:** Speed and efficiency [1] `http://kdd.ics.uci.edu/databases/kddcup99/kddcup.testdata.unlabeled_10_percent.gz`	/10	
Specify column *The user can select a column for analysis* *Must have: A mechanism to select a column of interest* **Measured:** Selection is used throughout the analysis	/10	
Presentation of information *The information is presented in a way where it is easy to view frequency of each type* *Must have: A GUI to present the data* **Measured:** Quality and ease of aiding the user in sense making	/10	
Hashing & Quality and feel *Quality of the software and user experience* *Implementation of hashing function* *Must have: A logical GUI & hashing mechanism* **Measured:** The ability to detect change via hashing; The system is easy to use and behaves as expected	/10	
Additional functionality *Functionality that was not specified* *Optional for a higher mark* **Measured:** Usefulness and impact on SIEM analysis	/10	
Total	/50	

Totals

Score /100: **Grade:**

General Comments: