Terabyte Threat Analysis
Acceptance Criteria Consideration

Team Lima
For Paul Reid on behalf of BT

4th March, 2013

1 Acceptance Criteria

As per the document dated 12th February, 2013 regarding the specifications for the software project that Team Lima produced, this retrospective serves as a consideration as to whether the final result does or does not meet the needs initially identified and agreed upon. For each section, the name of the requirement, the previous description of the requirement, and a justification are provided.

1.1 Importer tool

The importer tool must be able to take .nfcapd files, convert them into CSV and place them into the HDFS. If a malformed file is created, it is not expected to do anything other than throw an error - no error correction attempt will be made.

As specified, the importer tool does watch a directory and convert .nfcapd files to .csv files. The resultant file does appear in the HDFS. No consideration has been made for malformedness, so a failure of the nfdump utility will halt importing.

1.2 MapReduce

The Hadoop MapReduce jobs are not expected to produce 100% correct results at identifying threats. The system is unable to be tweaked to such degrees without all of BT's data, something that must be honed during real operation. Provided the CSV is read in and statistics and metrics pertaining to possible threats are logged reasonably, this section is functional.

We have created a series of MapReduce jobs to identify statistics and metrics as to threats. They are all rule-based, containing a variable that identifies when a given flow is considered to be worth investigation. Given the low amount of test data we had, we are aware that we did not produce 100% correct results, but this is functional given the above criteria nonetheless.

1.3 Cleaner

An investigation to ensure that the HBase cleaner only deletes required rows will be made.

Although we have no way to physically prove that this is the case, the fact we pass in the timestamp of the earliest acceptable row to HBase and use their methods to perform the rest leads us to believe that it is functional. Test cases have confirmed that in those instances, only the correct rows were removed.

1.4 Monitor

Confirmation that the monitor passes through the correct inputs to the PGSQL database will be checked and tested.

Given the change in the purpose of the Monitor to mostly a simple passthrough, this was not truly necessary any more. However, the Statistics module does aggregate the data prior to output. This stage has been tested and is believed to correctly perform its calculations.

1.5 Web UI

The frontend will be checked for updating bugs, framework errors and syntax/layout mistakes to ensure it complies with web standards correctly.

Due to use of so many frameworks, a significant portion of the generated code is of such a form that another developer has confirmed that the structure is valid. The portions that have been created by the team have been passed through the W3C validator in a final test that confirms adherence to specification. Finally, scripting has been confirmed not to throw warnings or errors to the browser.