Server side clustering

What information could be useful:

- details of application that crashed (name, version)
- list of installed packages and libraries
- informations about OS (kernel and system version, installed modules)

Possible way to process and cluster these informations

If any logs has similiar content, same ocucurrence of keywords
we can check if users have same applications installed and
if so we can cluster these crashes as a same type

K-means algorithm. It uses an iterative refinement technique.

K inital means are randomly generated.
Assignment step: Assign each observation to the cluster whose mean yields
the least within-cluster sum of squares

Update step: Calculate the new means to be the centroids of the observations
in the new clusters.

K-means clustering aims to partition n observations into k clusters in which each observation belongs to the cluster with the nearest mean, serving as a prototype of the cluster. For each log we could create a vector based on numbers of occurrences of each keyword and then apply the algorithm to get different clusters.

However, the K-means algorithm highly depends on the initial state and easily get trap into a local optimal solution. We can improve it by using a tabu search (TS) algorithm. TS avoids returning to recently visited solutions by constructing a list which is a called a Tabu List (TL). TS generates many trial solutions in a neighbourhood of the current solution and select the best one among all. The process of generating trial solutions is composed to avoid generating any trial solution that has already been visited recently. The best trial solution in the generated solutions will become a current solution. TS can accept downhill movements to avoid getting trapped in local optimal. TS can be terminated if the number of iterations without any improvement exceeds a predetermined maximum number of iteration.