CS463 MP2: Secure Index Based Grep

Seeun Oh (seeunoh2), Rohan Jyoti (jyoti1), Gaurav Lahoti (lahoti2)

Implementation

File Map: We map each file path to a random integer.

Word Index: We map each word in the file to the corresponding integer. If many files have the same word, we store all of them as a single string and later do string manipulation.

Add/Upload: We retrieve the values from filemap/indexmap file, create the hashmaps in the memory, add the additional files to the hashmaps (empty initially) and upload them to the server.

Delete: If file exists in filemap, we delete the entry in filemap and respective keyword entries in indexmap. Then send the command for the server to delete the corresponding file.

Search: Using the indexmap, we retrieve the string corresponding to the keyword, parse it and download the relevant files from the server. We decrypt the files on the client side and use filemap to store them with original filename.

Encryption/Decryption: Same as in MP1.

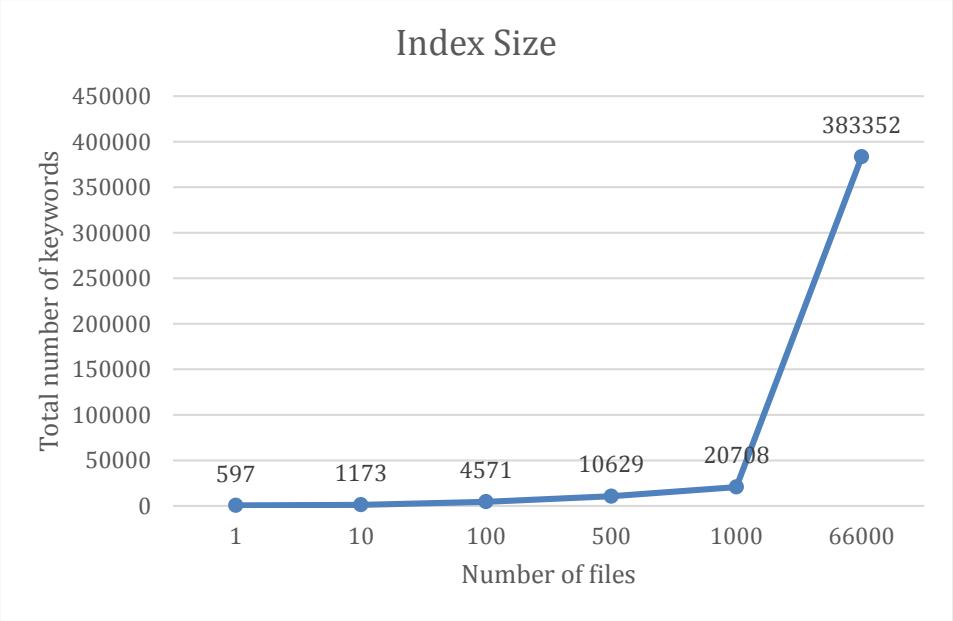
Performance

Time Trials (ms)	File Size (bytes)	Index Size (bytes)	Index Size (#words)	Add (ms)	Delete (ms)	Search Frequent (ms)	Search Rare (ms)
1 File	14.8 KB	10 KB	597	225	61	180	177.33
10 Files	44.9 KB	34 KB	1173	262	72	474	180
100 Files	333 KB	237 KB	4571	823	161	4106	201.33
500 Files	1.21 MB	1.02 MB	10629	3915	313	19980	200
1000 Files	2.7 MB	2.0 MB	20708	13995	3045	39848	217
66000 files	342.5 MB	128.8 MB	383352	27841822	45432	104578	2256

The frequent and rare keywords for no. of files: 1: {the, swisher}, 10: {the, gordon}, 100: {to, implement}, 500: {1, berkeland}, 1000: {0, index-flat}, 66000: {1, hawkeyes}

Time for adding and deleting "enron_mail_20110402/maildir/dorland-c/deleted_items/20" from 66000 files: 64890ms and 49284ms

Results



As expected, the add/upload performance of this MP is much slower than that of MP1 since we have to not only encrypt and transfer but also build a local index. The search performance is better than that of MP1 because of the indexing. In fact, the search is O(1) due to the data structure used; the rest of the time is attributed to downloading the corresponding files from the server. In terms of security, both MPs are very similar since there are no design differences in the encryption/decryption portion. However, a possible implication of this MP being less secure is the fact that the adversary can now record the groups of files being downloaded together and possibly deduce and construct a pattern. This leaks information about files which contain similar keywords.