Forum:Hack Forums

Thread: Finding duplicates in directory

K-On! - *February 3, 2014, 10:01 pm*
There you have it, hopefully this will help someone.

http://pastebin.com/iZAN34d0

PigBacon - *February 4, 2014, 2:57 am*
What if I've got a few gigs worth of distro isos in my dir? Reading everything into memory isn't a great idea - best to use a streaming OpenSSL method to hash, don't you think?

K-On! - *February 4, 2014, 6:03 am*
(Feb 4, 2014 03:57 AM)plausibility Wrote: *[CITE]* What if I've got a few gigs worth of distro isos in my dir? Reading everything into memory isn't a great idea - best to use a streaming OpenSSL method to hash, don't you think?
I definitely know this is not the best way to do this, but hey, it's something! Might be useful for some situation but not for big amounts of data/files, OR then just revise the code and find another way :P

Dr. Strangelove - *February 4, 2014, 12:33 pm*
I was thinking about adding something like that to my FTP client here: http://is.gd/r91ixK

CodeX - *February 4, 2014, 12:53 pm*
You could just use os.stat to get the file size, you can assume that one hash wouldn't be the same as another if the file size is different, while not true for hashes it is certainly true to raw data in the files themselves and that's what counts in this case, this would save a lot of work by using data information available from the file system. If the files are of an equal size then you could look to hashing or for what it's worth just do a string comparison on the two, e.g. Code def same(fileA,fileB):

```
    a = 1
    while a:
        a = fileA.read(512)
        if a != fileB.read(512): return False
    return TrueHashing should be slower than a string comparison as you have to read and process
```

the entire file in the first place for the hash. Using a string comparison could mean that you won't have to read the complete file if two files aren't the same.

If two files are equal then you can delete one or remove it from the list of files being compared which will make things faster if there are a lot of duplicates of the same files.

At the moment I can't think of any advantages to using hashes unless you intend to reuse the hash.

It could also be better to walk through the files instead of creating a list as you can keep the memory usage down too which can be an issues in some cases when you have vast amounts of files.