

**CLAIMS****WHAT IS CLAIMED IS:**

1. A cloud data storage infrastructure operating a data security system, comprising:

a connection to a telecommunications network coupled to one or more client devices;

a supply of data from one or more client devices being received into memory or data storage of a server computing device; and

a machine executable code loaded into the server computing device wherein steps of the code are executed by a computer processor in the server computing device;

wherein, the code is configured to:

receive a data file,

disassemble the data file into blocks of data;

generate addresses for random storage locations in a cloud based network for each of the blocks of data,

store, the blocks of data into the random storage locations within the cloud data storage infrastructure, and

record the addresses of the random storage locations into an array of pointers.

2. The system of claim 1, wherein the code is further configured to randomly size the blocks of data into a range of sizes between a preset minimum and a preset maximum block size.
  
3. The system of claim 1, wherein the code is further configured to interrogate a selected random storage location and determine whether or not the selected random storage location is empty, and if a storage space in the selected random storage location is empty, the code is further configured to interrogate the selected random storage location and determine whether or not the selected random storage location is large enough to accommodate one of the blocks of data, and if a storage space in the selected random storage location is large enough, write the one of the blocks of data to a storage medium in the selected random storage location, and if the selected random storage location is not large enough to accommodate the one block of data, the code is further configured to reject the selected random storage location and then select another random storage location.
  
4. The system of claim 1, wherein the code is further configured to:  
  
interrogate an empty selected random storage location and determine whether or not the selected random storage location is large enough to accommodate the one block of data, and if the selected random storage location is not large enough to accommodate the one block of data, then to  
  
reserve enough space within the selected random storage location to write a

pointer to a next random storage location,

write as much of the one block of data to the selected random storage location as fits within the selected random storage location,

write a pointer to the next random storage location to the end of the first storage location,

write any remaining data from the one block of data to the next random storage overflow location.

5. The system of claim 1, wherein the code is further configured to:

interrogate a selected random storage location and determine whether or not the selected random storage location is large enough to accommodate the one block of data; and if the selected random storage location is not large enough to accommodate the one block of data, then

extend and / or indexing to the next location in the array of pointers and reserve there enough space to write an additional pointer to a next random storage overflow location;

write any remaining data from the one block of data to a next available random storage overflow location; and

write an additional pointer to the storage overflow location to the array of pointers to the storage locations.

6. The system of claim 1, wherein the code is further configured to:
  - store the array of pointers as a sequential array of pointers,
  - retrieve the random blocks of data in a sequence according to the sequential array of pointers, and
  - re-assemble the blocks of data into the data file according to the sequence of the sequential array of pointers.
  
7. The system of claim 1, wherein the code is further configured to randomly vary a routing of the random blocks of data through the cloud based network.
  
8. The system of claim 1, wherein the code is further configured to release any storage locations which stored blocks of data which have been retrieved and which are no longer needed to be stored.
  
9. The system of claim 1, wherein the code is further configured to delete the data file by:
  - renaming and overwriting a file type string of the data file according to a parameter comprising the number times to overwrite;
  - renaming and overwriting a file name string of the data file according to a parameter comprising the number times to overwrite.; and
  - overwriting file contents of the data file according to a parameter comprising the number

times to overwrite.

10. The system of claim 1, wherein the array of pointers is stored to a storage media that is isolated within a location separate from the storage media of the blocks of data.

11. The system of claim 1, wherein the code is further configured to re-assemble the blocks of data into a copy of the data file in response to a call for the data file by retrieving the blocks of data from the random storage locations according to the array of pointers.

12. The system of claim 1, wherein the code is further configured to:

use a first communications channel in the telecommunications network to record the addresses of the random storage locations; and

transmit the blocks of data through a second communications channel, wherein the second communications channel is isolated from the first communications channel and vice versa.

13. The system of claim 1, further comprising a plurality of communications channels within an isolated data block handling network infrastructure, configured to permit and execute parallel uploading and downloading of data blocks to and from a plurality of random storage locations.

14. The system of claim 1, wherein the code is further configured to: execute the steps of disassembly of data files into data blocks on the server side of the telecommunications network and re-assemble the blocks of data back into a copy of the data files on either the server side or provide to a client side device or to another service provider system access in the form of isolated channels to access data block, and pointers key storage locations through which to download copies of pointers based keys, and / or data blocks.

15. A computer program product for operating a data security system in a cloud data infrastructure, the computer program product comprising:

one or more computer readable storage media, and program instructions collectively stored on the one or more computer readable storage media, the program instructions comprising:

receiving a data file,

disassembling the data file into blocks of data;

generating addresses for random storage locations in a cloud based network for each of the blocks of data,

storing, through the telecommunications network, the blocks of data in the random storage locations within a data storage infrastructure of the cloud based network, and

recording the addresses of the random storage locations into an array of pointers.

16. The computer program product of claim 15, wherein the program instructions further comprise: interrogating an empty selected random storage location and determine whether or not the selected random storage location is large enough to accommodate the one block of data, and if the selected random storage location is not large enough to accommodate the one block of data, then

reserving enough space within the first storage location to write a pointer to a next random storage location,

writing as much of the one block of data to the selected random storage location as fits within the selected random storage location,

writing a pointer to the next random storage location to the end of the first storage location, and

writing any remaining data from the one block of data to the next random storage overflow location.

17. The computer program product of claim 15, wherein the program instructions further comprise:

storing the array of pointers as a sequential array of pointers,

retrieving the random blocks of data in a sequence according to the

sequential array of pointers, and

re-assembling the blocks of data into the data file according to the sequence of the sequential array of pointers.

18. The computer program product of claim 15, wherein the array of pointers is stored into a storage media that is isolated within a location separate from the storage media of the blocks of data.
19. The computer program product of claim 15, wherein the program instructions further comprise re-assembling the blocks of data in response to a call for the data file by retrieving the blocks of data from the random storage locations according to the array of pointers.
20. The computer program product of claim 15, further comprising program code configured to operate a plurality of communications channels within the isolated data block handling network infrastructure, capable to permit and execute parallel uploading and downloading of data blocks to and from a plurality of random storage locations.
21. The computer program product of claim 15, further comprising program code capable to execute the steps of:



disassembling data files into data blocks on the server side of the telecommunications network, and

re-assembling the blocks of data back into a copies of data files on either the server side, or providing to a client side device or to another service provider system access by operating and controlling isolated communication channels coupled to data block, and pointers key storage locations providing the facility to download copies of pointers based keys, and / or data blocks.