Tenfingers: a decentralized sharing protocol

Ludvig Larsson aka Valmond/Loulou
ludviglarsson@protonmail.com
tenfingers.org

Version 1.04

Abstract:
A decentralized sharing system will allow any entity to safely share any data with anyone or only a select few. Anyone trading some storage space will be able to share data that might be both anonymous, decentralized and take-down safe.
This works by reciprocal sharing; Alice shares Bob's data because Bob shares Alice's data.
Sharing with several nodes assure the data is accessible from several regions in the world which assures a high availability. Strong encryption and obfuscation assures resilience to take-down because nodes do not know what they share.

1. Introduction

FOSS, Free Open Source Software have gone from a fringe movement to enabling the majority of computer related work. There is a plethora of Operative Systems and most software can be found in the Free Open Source Software sphere, but there is something that is not yet free for everyone, and the Tenfingers Sharing Protocol attempts to change that: Decentralized Online Hosting.

There are already some tries with decentralized sharing, but historically they all have some restrictions; needing a crypto currency to function, needing a charitable node that shares your data for free, it is centralized or the data is not encrypted.

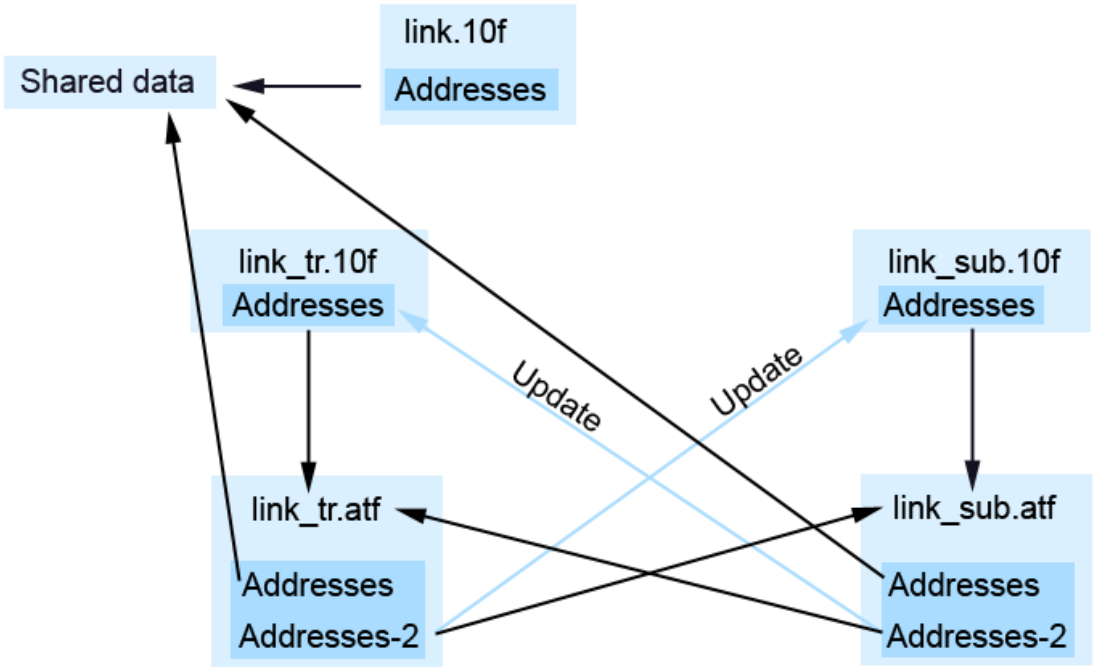
2. Sharing

Computers nowadays usually have unused storage space that we can leverage and trade for an online presence.
If we want resilience, we can share our data with tens or hundreds of other nodes, which makes each online megabyte ‘cost’ (more or less) ten to a hundred megabytes of disk space and in return, tens or hundreds of computers will serve our data.
For example, A 10MB web page, shared a hundred times, would cost less than a couple of gigabytes of local storage space.

3. Network

Nodes are identified by an asymmetric RSA key, they will connect to other nodes to gather information about the network and learn about new nodes.
The nodes are stored in a list and will be contacted from time to time to verify they exist and that they are running.
Shared data is stored on nodes and a link file can be generated, containing the addresses to those nodes, and some extra information like the name of the data. The link file grants access to both downloading the data and to decrypt it.
As a user shares data, another two data are also generated (a translation file and a substitution translation file) and are shared in the same manner. They contain a list of the addresses pointing to the original data and another list pointing to the other one of these two files. They are not necessarily shared by the same nodes. This information will be updated when a node changes address, so that neither the base link nor the translation files will not go stale over time.
See fig 1. The translation file and the substitution files contains addresses to the base payload (the shared data) but also addresses to the 'other' accompanying file so that we always will have the possibility to download the data. They are updated when data is downloaded.

Fig 1:



Light blue boxes denotes files
Darker blue boxes contains a lists of node addresses who store a specific data for us, the black arrows shows what they store (link.10fs addresses downloads the original shared data, link_tr.10fs'

addresses downloads link_tr.atf (address translation file), Address-2 in link_tr.arf downloads link_sub.10f et cetera).
Address-2 are containing addresses of the reciprocal _tr/_sub files addresses, this is so that when nodes change IP:PORT addresses, we can update them in the shared .atf files.
Addresses in the .atf files pointing on the original data, will also change over time if the nodes who store the original data changes address or the nodes themselves has been changed.

4. Incentive

The incentive is based on: I'm sharing your data because you are sharing my data.
A node no longer sharing our data will be discarded and replaced by a new node sharing our data (we of course will stop sharing the old nodes data and start sharing the new nodes data).
Coupled with redundancy there is a high chance the data is always accessible even if your computer is not online.

5. Privacy

All communication is started with an asymmetric RSA exchange sharing a AES256 key-pair generated on the fly, assuring we talk with the right node (the RSA part), and that our data cannot be spied on (the RSA and then the AES part).
The data itself is also encrypted with AES256 which means nodes do not know what they are sharing. Only those having access to the link file, which contains the addresses of the nodes storing the data and the AES key-pair), will be able to both access the information and decrypt it.
As you can share as many copies as you want to, the probability that all of them are in the same country or region of the world greatly diminishes with the increasing number of copies shared. This makes it resilient to any kind of take-down.

6. Conclusion

We have a lot of unused storage space, we can use it to enable a secure and decentralized presence online for anyone at no more cost than sharing some of our bandwidth.