MONA concept - UPL

This platform focuses on the privacy and integrity of user data.

Privacy is understood as the right of each individual, company, agency or administration in keeping your communications with third parties safe. It is not the purpose of this ecosystem being used to commit crimes in any way, as well as organizing, financing or coordinating any action that may harm third parties, their assets or interests. It is essential and expected, the necessary mechanisms and regulations are established in order to guarantee both requirements.

It is considered data integrity the fact of receiving information, be it the means of dissemination of this (analog, digital or others) in the way that the sender of the same has sent it to its recipient. Otherwise, it would be receiving biased, inaccurate or false information from a wrongful intermediary or operator. It is part of this integrity to receive the information in the relevant time depending on the speed of dissemination of this depending on the medium.

Based on the above concepts, a platform was designed to meet such challenges to the extent possible.

How does it work:

The following scenario describes two individuals or parties establishing a secure communication mechanism. For our example, part (a) asks part (b) for the latter's consent to send (x) information: document, file, program, multimedia content, etc.

The above can be undertaken in several ways, being the most common: use of telephone devices, via text messages, social network, email, etc. Once the part (a) has received the shipment confirmation, the part (b) proceeds to request the system (download App) to configure it according to what was agreed with (a). Said basic configuration consists of setting a lifetime for said information. This concept is incorporated so as not to retain any information after the time of receipt of (b) or the time set by the user. In this way, in a scenario in which said platform is compromised, only the content existing at that time will be accessible, but not the previous or future content. Later we will delve into the concept of data encryption that is stored on that platform.

The user (b) installs the App on their terminal or device, it has a lifetime and is for single use only. In other words, the App can be configured to be deleted once the information has been received and saved outside it, the latter can be on the device itself, removable media, the forwarding of the same or a combination of the above. Even if the user wishes to keep said application, the synchronous encryption engine which is described in the attached document [Extemporaneous formula] must be downloaded to receive a new reception of information.

The above is very useful because if the user's device is compromised (b) our platform will proceed to notify the user of his situation. This is achieved with proprietary mechanism that is incorporated into the main platform, for obvious reasons it is omitted in this description or concept of operation.

Once the above is done, the user in his terminal generates a pseudo-user which serves to: that our platform generates a (smart contract) and its respective Wallet. As well as to send this pseudo-user to the part (a). Then the part (a) proceeds to make a query to our system to verify if in effect said user is valid. If so, and using your encryption key (not part (b)), you send this information to our platform. Same that can be consulted by the part (a) within the time limit set by it.

For each process of interaction with our platform both parties are under the supervision of the user verification module (App), this is so to avoid different attacks, these may be from an authentic but intentionally bad user. It is for this reason that if any irregularity is detected, the system notifies the involved party of their condition and the service is aborted.

The part (a) once the system confirms the existence (validity) of the user or part (b) proceeds to send its information to the system. The above is done so that when (a) requesting a user search service and after verification of it, our system shares a single-use encryption key, which is valid only for that user (b) and the time of the shipment request.
This mechanism is used to encrypt the information sent to part (b). In the case of (b) in the same way the system asks for your key or encryption engine only when it proceeds to download your information.

Upon completion of the above, our system eliminates both contracts and their respective wallet.
When doing the above, the funds or Tokens return to the main wallet of our system to start the cycle again with other users. In this way, none of the parties (a) or (b) retain Tokens beyond the lifetime of their respective contracts. Being for (a) a very short event since these are only used to (encode) the information to be sent to part (b). After said shipment or storage, the contract and the wallet associated with that transaction are eliminated, but not before recovering any surplus of remaining Tokens after the information is sent.

In the case of part (b) your wallet retains said Tokens until the term or validity set by the user in relation to retention of information or until it downloads said information. After which our system proceeds to eliminate said contract and its respective wallet.

If both parties decide to share information, they must again carry out the process.
For this reason they must re-download their respective encryption engines or synchronous keys.

How information is stored or (encoded):

In a publication entitled: [Data storage in the Blockchain network] this process is described.
Briefly described, any content to be sent is converted to hexadecimal format. To subsequently apply characterization method [Concept of calculation neuron and fractal generation]. At the end of this process we have a mathematical summary or fractal of the original content itself that with its proper encryption engine or key can be reverted to the original state by returning the original information exactly and in the same order it was entered before performing said process.

Due to the nature of this last procedure, from an extensive original data we obtain a summary, therefore in a way it is as if we compress the information with a compression radius impossible to reach by a conventional method.

This summary or fractal is what we send in the form of a transaction to the recipient, that is, the tokens stored in their wallet are equivalent to the original encrypted and compressed message. For the above, the part we take is the decimal or fraction of token. Therefore a user would never have a unit integer. At the end of said operation and as described above the residual of said fraction is returned to the Main wallet of our main smart contract.

So, in case of intrusion by third parties, what the wallets that are waiting for delivery to the customer, are only tokens that do not make direct reference to any information. The foregoing given that they require a decompression process that is not possible if either party is not involved in the proper order and with the system verification guarantees. In the scenario that a user is compromised this does not harm the rest since the keys of each parts are unique and for a single use, as well as the option that has the longest life (24hrs) only applies to (b). This last parameter is suggested and can be adjusted according to requirements.

In the future document will be delved into the issue of device-device security as well as anti-spam (boot) methods developed for this platform and its owners.

Subject to abundance.

Julio C. Gómez.
Io-exchange.com (c)