Uphantom - *the next generation of crypto databases*

White paper

Written by **Eran Ben-Shahar, Wellington, Feb 2022**

1. Introduction

In this paper we will describe *Uphantom* - a new method to distribute information over an existing network, aka the internet, in a secured, private and decentralized way.
In the basics of *Uphantom*, as the name suggests, is that the network is established as a parasite network over an existing network, like noise exists in signals passing through electrical wires. Idea is that *Uphantom* information will pass through the internet between participants while the main role of those participants is actually rather different: websites, smartphone devices, desktop machines, servers: any such internet node could potentially participate in the *Uphantom* network without disturbing its usual work. In addition, such participants will be incentivized in a unique and sustainable way to participate in the network.
Uphantom is not just unique from the ghost-like architecture point of view or the incentives, but also from the way we secure and decentralize the information. Security and privacy are probably one of the main concerns of crypto databases users. However the way *Uphantom* works, any information passed and stored on any network node is actually useless and meaningless for itself, even if a mull actor managed to decrypt it. This will be described below as well.

2. Blockchain - advantages and disadvantages

The first decentralized blockchain was conceptualized by a person (or group of people) known as Satoshi Nakamoto in 2008. Nakamoto improved the design in an important way using a Hashcash-like method to timestamp blocks without requiring them to be signed by a trusted party and introducing a difficulty parameter to stabilize the rate at which blocks are added to the chain. The design was implemented the following year by Nakamoto as a core component of the cryptocurrency bitcoin, where it serves as the public ledger for all transactions on the

network. In August 2014, the bitcoin blockchain file size, containing records of all transactions that have occurred on the network, reached 20 GB (gigabytes). In January 2015, the size had grown to almost 30 GB, and from January 2016 to January 2017, the bitcoin blockchain grew from 50 GB to 100 GB in size. The ledger size had exceeded 200 GB by early 2020.
As bitcoin turned popular, and more and more blockchain networks emerged, advantages and disadvantages were discovered:

Decentralization (advantage): Peer-to-peer blockchain networks lack centralized points of vulnerability that computer crackers can exploit; likewise, it has no central point of failure.

Security (advantage) : Blockchain security methods include the use of public-key cryptography. A public key (a long, random-looking string of numbers) is an address on the blockchain. Value tokens sent across the network are recorded as belonging to that address. A private key is like a password that gives its owner access to their digital assets or the means to otherwise interact with the various capabilities that blockchains now support. Data stored on the blockchain is generally considered incorruptible.

Transparency (advantage): because of the decentralized nature of Bitcoin's blockchain, all transactions can be transparently viewed by either having a personal node or using blockchain explorers that allow anyone to see transactions occurring live. Each node has its own copy of the chain that gets updated as fresh blocks are confirmed and added. This means that if you wanted to, you could track any cryptocurrency wherever it goes. However this may be a huge disadvantage when it comes to privacy.

Popularity (advantage): in the last 5 years blockchains became very popular with thousands of trading platforms, coins, databases and other applications using blockchains. This is making blockchain very accessible, and technical blockchain talent very keen to be involved with such projects.

High cost effectiveness (disadvantage): due to the nature of blockchains which use Proof Of Work processes (POW), the amount of effort in terms of CPU to perform a blockchain transaction is enormously bigger than other ways. This has implications on the price ("fee") of a transaction, environmental implications and how sustainable the platform is.

Low speed (disadvantage): as the blockchain grows in size, the time that it takes to perform a transaction increases, this is known as The Blockchain Scalability Problem.

Manipulation / government control risk (disadvantage): The structure of blockchain is basically giving power to the nodes with the highest CPU power, it also means that if enough nodes collaborate as mull actors, false transactions could be pushed into the ledger. “51% attack” which occasionally happen[1] but it seems like it is possible to manipulate a blockchain even without high CPU power. Taking into account that governments and surveillance organizations have got the highest CPU power available, it means that the 51% attack can give them the potential to damage the blockchain integrity.

Privacy (disadvantage): as mentioned above, the blockchain is fully transparent to anyone. Even though accounts are hidden behind anonymous identity (“a wallet”) it means that all and every wallet transaction is known to anyone - it is in the public domain. It also means that once an actor is exposed - then **all their historical activity** is exposed. Exposure of an actor’s identity is pretty simple as once an actor needs to liquidate crypto-money to real money or once they pay someone else - their real identity is disclosed and therefore all the historical transactions are exposed.

As will be described below, *Uphantom* is designed with all the blockchain advantages, while mitigating all the disadvantages.

3. Secret Sharing

Secret sharing (also called secret splitting) refers to methods for distributing a secret among a group of participants, each of whom is allocated a share of the secret. The secret can be reconstructed only when a sufficient number, of possibly different types, of shares are combined; individual shares are of no use on their own.
For *Uphantom* we selected to use Shamir’s Secret Sharing Scheme (SSSS) which is one of the first “secret sharing” cryptography algorithms invented by the Israeli cryptographer Adi Shamir (co-inventor of RSA) in 1979. The idea behind SSSA is pretty simple: we split a secret S to N parts such that with any K-out-of-N pieces you can reconstruct the original secret S, but with

[1] https://fortune.com/2018/05/29/bitcoin-gold-hack/

any K-1 pieces no information is exposed about S. That is conventionally called a (N, K) threshold scheme.
Let's look at a simple example: Let us construct a scheme to share our secret 1954 (S) with 4 (N) shares and a threshold of 3 (K).

First, we randomly choose K – 1 positive integers, so in our case, 2 positive integers. We randomly choose 43 and 12.

Then, we build a polynomial of the form

y = a0 + a1*x + a2*x^2

Where a0 is the secret, and a1 and a2 are our randomly chosen integers. We are left with:

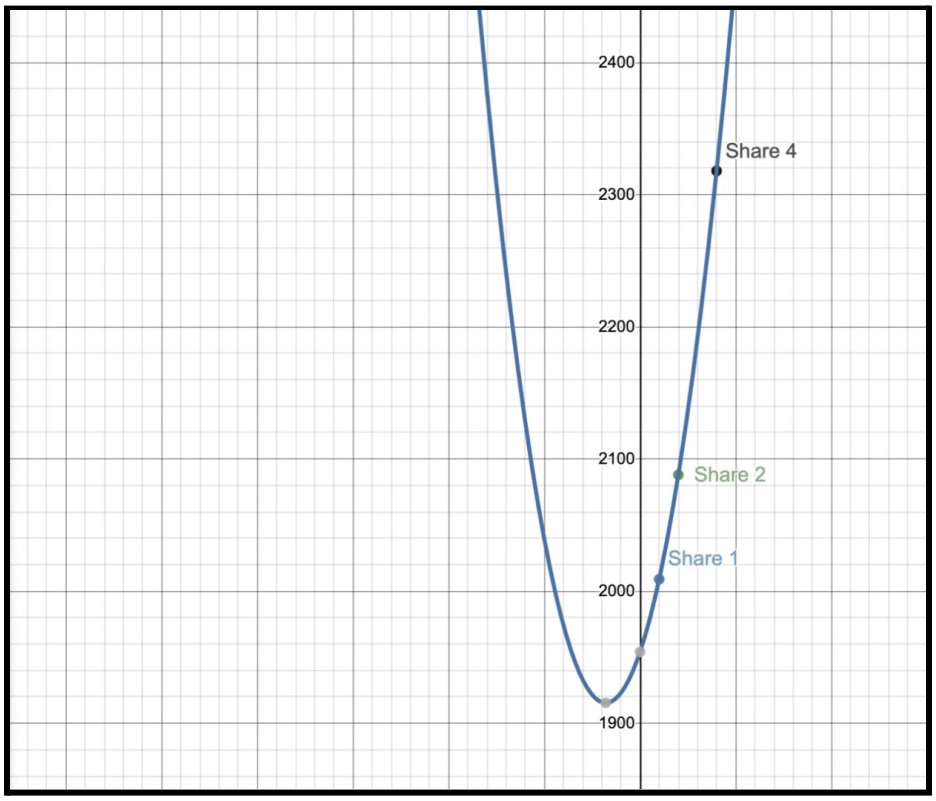
y = 1954 + 43x + 12x^2

Then, we use this formula to create 4 points [x,y] shares that we give to each participant.

	Share 1	**Share 2**	**Share 3**	**Share 4**
X	x=1	x=2	x=3	x=4
Y	y = 1954 + 43*1 + 12*1^2 = 2009	y = 1954 + 43*2 + 12*2^2 = 2088	y = 1954 + 43*3 + 12*3^2 = 2191	y = 1954 + 43*4 + 12*4^2 = 2318
[x,y]	**[1,2009]**	**[2,2088]**	**[3,2191]**	**[4, 2318]**

Recall that for reconstruction we need **any three** of the four shares together, basically, provided with three pairs, we can solve the parabola formula and find the secret S. Lets assume we have

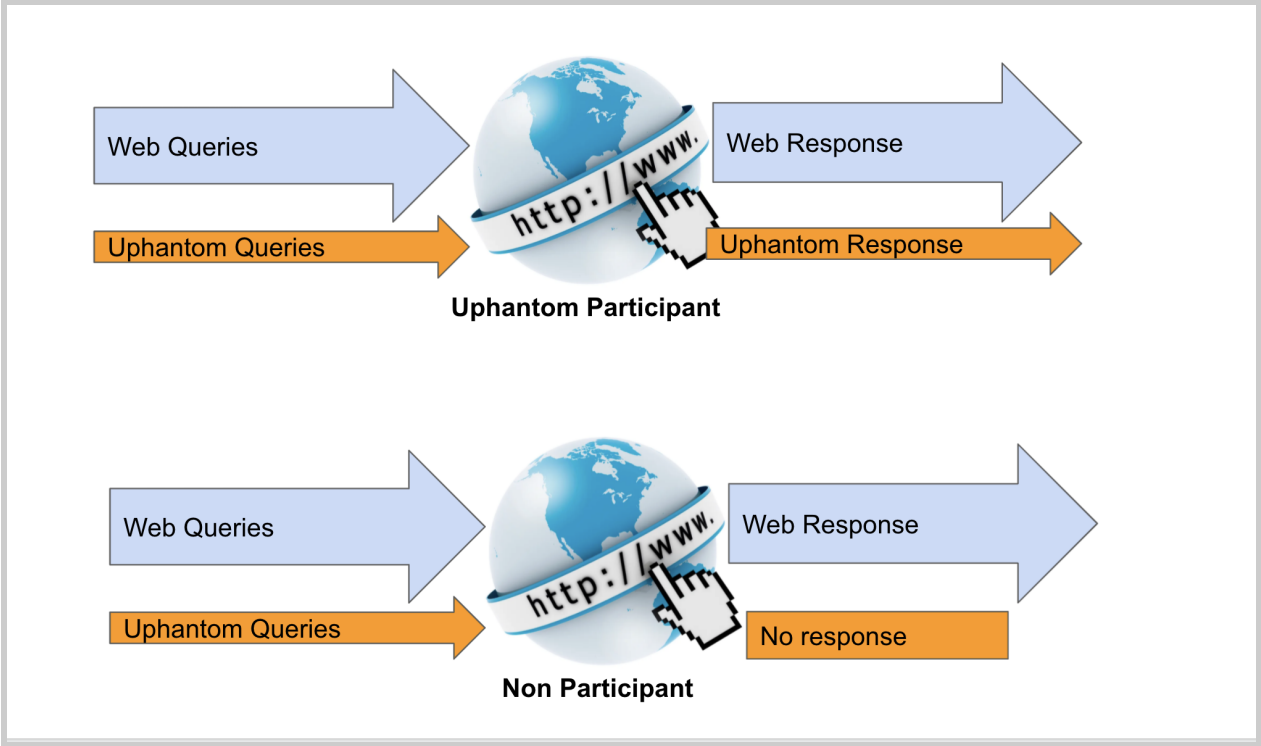
control of shares 1,2 and 4. We draw the parabola:



and find the secret where x=0: y=1954

4. **Uphantom network - white noise over the internet**

   Another idea we use for Uphantom is related to background noise. In communication systems, noise is an error or undesired random disturbance of a useful information signal. The noise is a summation of unwanted or disturbing energy from natural and sometimes man-made sources. However, in many use cases multiple information channels can be sent over one network by separating their frequencies - for instance - multiple internet connections can be sent over one wire / WIFI network by using different frequencies, another case study is to use utility wires for both power (50Hz) and broadband internet (1Ghz) signals. In electronics, signal separation needs to be done in the signal level, using techniques like Fourier Transforms, as per Uphantom and creating another layer of information over the internet, this can be done simply by using existing internet nodes (smartphone devices, desktop computers, servers) and different (but uniquely distinguished) input/output rules. This will be explained in detail below.



5. **Uphantom roles**

There are three types of participants in the Uphantom network - Connectors, Transactors and Users. Any participant can have either one, two or all of those roles.

Connectors	Connectors are network participants which allow access to information of other Connectors, and of Transactors
Transactors	Transactors are network participants which connect between users securely, and record their transactions
Users	Users are network participants which perform transactions based on the network business rules

6. Uphantom protocol

The Uphantom protocol is based solely on sending query strings over Hyper Text Transfer Protocol - HTTP - (and HTTPS) to a certain participant's URL. All variables sent are GET Method query string variables, i.e. added to the URL in the form of additional Upahntom_VARIABLE=VALUE to the URL. For example, if the participant's original URL is

https://en.wikipedia.org/w/index.php?search=happy+hour+beer

Assuming that wikipedia is participating in the Uphantom network, the URL could look like:

https://en.wikipedia.org/w/index.php?search=happy+hour+beer&Uphantom_action=get&Uphantom_account=H6q34hx47

7. Uphantom Basics

Uphantom queries are structure flexibly by the use case, and in theory have unlimited length[2]. Two Uphantom parameters are compulsory and sent with each and every query - these are Uphantom_action and Uphantom_node, and following them are Uphantom_val which holds all the transaction data.

URL query string parameter	Possible values	Description
Uphantom_action=	get	Get values of a node
	post	Sends a value to a node
Uphantom_node=	connector	
	transactor	
	user	

[2] In practice, there are limitations, please refer to https://stackoverflow.com/questions/812925/what-is-the-maximum-possible-length-of-a-query-string

Uphantom_uid	{string}	User ID
Uphantom_ukey	{string}	User key
Uphantom_targetuid		Target user id (if any)
Uphantom_val1=	{anything}	
Uphantom_val2=	{anything}	
Uphantom_val3=	{anything}	
…		
Uphantom_valN=	{anything}	

Lets look at several examples:

a. **Send $100 to a friend**

Variable	**Value**	**Description**
Uphantom_action	post	Posting data to transactor node
Uphantom_node	transactor	
Uphantom_uid	Xu743jsdf4kH	Sender user ID
Uphantom_ukey	ghf834rhfd9iIIaefhsdf6234h	Sender key
Uphantom_targetuid	jas8fQkas9429sd	Recipient User ID
Uphantom_val1	pay	Command for node
Uphantom_val2	100	Amount
Uphantom_val3	USD	Currency

Final URL:

Uphantom_action=post&Uphantom_node=transactor&Uphantom_val1=pay&Uphantom_uid=Xu743jsdf4kH&Uphantom_ukey=ghf834rhfd9iIIaefhsdf6234h&Uphantom_val2=100&Uphantom_val3=USD&Uphantom_targetuid=jas8fQkas9429sd

b. **Check my account balance**

Variable	Value	Description
Uphantom_action	get	Get data from transactor node
Uphantom_node	transactor	
Uphantom_val1	balance	Command for node
Uphantom_val2	USD	Currency
Uphantom_uid	Xu743jsdf4kH	Sender user ID
Uphantom_ukey	ghf834rhfd9iIIaefhsdf6234h	Sender key

Final URL:

Uphantom_action=post&Uphantom_node=transactor&Uphantom_uid=Xu743jsdf4kH&Uphantom_ukey=ghf834rhfd9iIIaefhsdf6234h&Uphantom_val1=balance&Uphantom_val2=USD

c. **Get the first 100 connectors from the hosting connector node**

URL:

Variable	Value	Description
Uphantom_action	get	Get data from transactor node
Uphantom_node	transactor	
Uphantom_val1	connectors	
Uphantom_uid	Xu743jsdf4kH	Sender user ID
Uphantom_ukey	ghf834rhfd9iIIaefhsdf6234h	Sender key

Final URL:

Uphantom_action=get&Uphantom_node=transactor&Uphantom_ukey=Xu743jsdf4kH&Uphantom_uid=ghf834rhfd9iIIaefhsdf6234h&Uphantom_val1=connectors

d. **Get connectors 1201 to 1300 from the hosting connector node**

URL:

:

Variable	Value	Description
Uphantom_action	get	Get data from transactor node
Uphantom_node	transactor	
Uphantom_val1	connectors	
Uphantom_uid	Xu743jsdf4kH	Sender user ID
Uphantom_ukey	ghf834rhfd9iIIaefhsdf6234h	Sender key
Uphantom_val2	1200	

Final URL:

Uphantom_action=get&Uphantom_node=transactor&Uphantom_ukey=Xu743jsdf4kH&Uphantom_uid=ghf834rhfd9iIIaefhsdf6234h&Uphantom_val1=connectors&Uphantom_val2=connectors