

Beyond Bitcoin & FORTH (BFF001)

Hilbert Hotel

with Infinite Cabinets

This is a series to recap our work on FORTH and the next generation Internet hardware, software & networking technologies, starting with David Hilbert's theory of infinity, as illustrated by the Grand Hotel with (a twist that is) rooms containing infinite cabinets, representing ownerships of digital assets (in the cabinets) using hash numbers. The Hash Filesystem replaces database servers, simplifying server node implementations based on FORTH architecture.

Liang Ng, January 2026

omnixtar.github.io/svfig

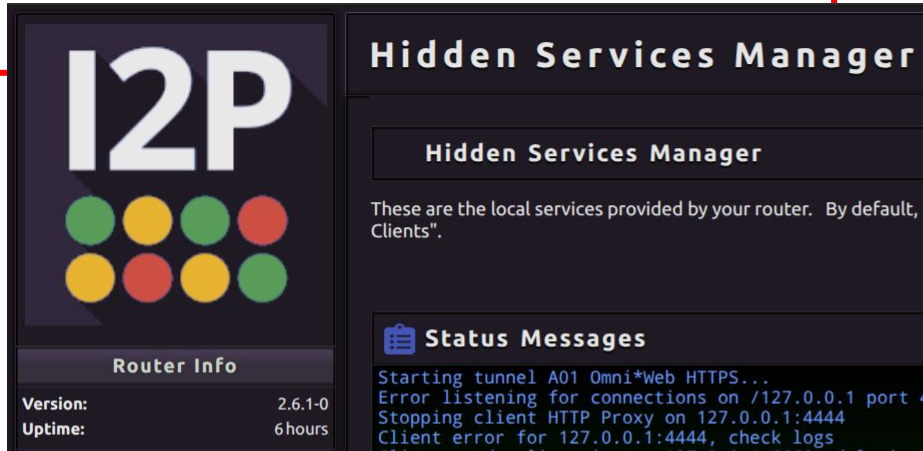
<https://www.linkedin.com/in/liang-ng>

- *alternative solution to Bitcoin and blockchain technologies for managing ownerships of digital assets*

“Who owns digital assets?”

“Buy, sell, transfer, royalties?”

- *Omnihash & DJJSON*
- *Decentralised JSON*



Pay-to-PubKey-Hash (Pay-to-Public-Key-Hash, P2PKH) is the basic form of most common form of transaction on the [Bitcoin network](#). Transactions that pay P2PKH scripts that are resolved by sending the public key and a digital signature [private key](#).

The ScriptPub

<https://bitcoinwiki.org/wiki/pay-to-pubkey-hash>

Table of C

1. Pay-to-Pubkey Hash

- 1.1. Pay-to-PubKey-Hash Review
- 1.2. Pay-to-PublicKey Hash Example
- 1.3. See also
- 1.4. References

FORTH / (Phoscript)

- *virtual machine in extreme environments*
- *abstraction of source code for monetisation*

...lishing the public key only when the coins are spent (and assuming that address is rendered ineffective.

...h the hash being smaller (20 bytes) it is easier to print and easier to embed int

```
hongwu@hongwu-Latitude-5480: /var/www/dmeta/o/auth/Graph/dmeta
hongwu@hongwu-Latitude-5480: ~
hongwu@hongwu-Latitude-5480: /var/www/d...
hongwu@hongwu-Latitude-5480: /var/www/d...
hongwu@hongwu-Latitude-5480: /var/www/d...

-rw-r--r-- 1 www-data www-data 597 Jan 22 15:12 'Hzn18Vs0lg=='
-rw-r--r-- 1 www-data www-data 597 Jan 22 15:10 'Hhj3wWn00A=='
-rw-r--r-- 1 www-data www-data 618 Jan 22 15:09 'GUD2koxhKQ=='
hongwu@hongwu-Latitude-5480: /var/www/dmeta/o/auth/Graph/dmeta$ ls -lt BYRAtdTkJw\=\=/|head
total 52
-rw-r--r-- 1 www-data www-data 618 Jan 22 17:32 Gg-6Vx8ANQ==
-rw-r--r-- 1 www-data www-data 597 Jan 22 15:51 FcSSP_buqA==
-rw-r--r-- 1 www-data www-data 597 Jan 22 15:21 E9wHrPZbIA==
-rw-r--r-- 1 www-data www-data 1150 Jan 22 15:16 CFkif5lnhg==
-rw-r--r-- 1 www-data www-data 597 Jan 22 15:16 HBJa8oIJPw==
-rw-r--r-- 1 www-data www-data 1150 Jan 22 15:16 ASE2NFkzjA==
-rw-r--r-- 1 www-data www-data 597 Jan 22 15:16 EpJddpwiPA==
-rw-r--r-- 1 www-data www-data 618 Jan 22 15:16 Hk27YiUtuA==
-rw-r--r-- 1 www-data www-data 1150 Jan 22 15:15 CemDL02Hfw==
hongwu@hongwu-Latitude-5480: /var/www/dmeta/o/auth/Graph/dmeta$ cat BYRAtdTkJw\=\=/Gg-6Vx8ANQ\=\= ;ec
ho
{"t":"2026-01-22T09:32:39.065Z","m":"from_parent_20260110_button","c":"<h4 id=\"-solution-for-truly-
decentralised-internet\">\u2013 Solution for Truly Decentralised Internet<a class=\"anchorjs-link \"
href=\"#-solution-for-truly-decentralised-internet\" aria-label=\"Anchor\" data-anchorjs-icon=\"\ue
9cb\" style=\"font: 1em \\/ 1 anchorjs-icons; padding-left: 0.375em;\"></a></h4>","x":"like","u":{"
href":"https:\\\\omnixtar.github.io\\I3P\\","origin":"https:\\\\omnixtar.github.io","protocol":"http
s:","host":"omnixtar.github.io","hostname":"omnixtar.github.io","port":"","pathname":"\\I3P\\","sear
ch":"","hash":""}}
hongwu@hongwu-Latitude-5480: /var/www/dmeta/o/auth/Graph/dmeta$
```



```
-rw-r--r-- 1 www-data www-data 597 Jan 22 15:12 'Hzn18Vs0lg=='
-rw-r--r-- 1 www-data www-data 597 Jan 22 15:10 'Hhj3wWn00A=='
-rw-r--r-- 1 www-data www-data 618 Jan 22 15:09 'GUD2koxhK0=='
```

```
{
  "t": "2026-01-22T07:12:52.930Z",
  "m": "like",
  "c": "<h4 id=\"-solution-for-truly-decentralised-internet\">\u2013 Solution for Truly Decentralised Internet<a class=\"anchorjs-link \" href=\"#-solution-for-truly-decentralised-internet\" aria-label=\"Anchor\" data-anchorjs-icon=\"\ue9cb\" style=\"font: 1em \\/ 1 anchorjs-icons; padding-left: 0.375em;\"><\a><\h4>",
  "x": "search",
  "u": {
    "href": "https:\\\\omnixtar.github.io\\I3P\\",
    "origin": "https:\\\\omnixtar.github.io",
    "protocol": "https:",
    "host": "omnixtar.github.io",
    "hostname": "omnixtar.github.io",
    "port": "",
    "pathname": "\\I3P\\",
    "search": "",
    "hash": ""
  }
}
```

total 1340

Dual chained properties replace many (all?) database functionalities?

- chained strings: hashcodes can be chained (concatenated) with strings
- chained functions (hash of a string is a string)

$$H(s) \rightarrow s, H(H(s)) \rightarrow s, \dots, H^n(s) \rightarrow s$$

```
-rw-r--r--  1 www-data www-data    97 Jan  16 15:58 1omyUgD8
```

```
hongwu@hongwu-Latitude-5480: /var/www/dmeta/o/auth/Graph/dmeta$
```



```
total 12
-rw-r--r-- 1 www-data www-data 597 Jan 22 15:12 'Hzn18Vs0lg=='
-rw-r--r-- 1 www-data www-data 597 Jan 22 15:10 'Hhj3wWn0OA=='
-rw-r--r-- 1 www-data www-data 618 Jan 22 15:09 'GUD2koxhKQ=='
hongwu@hongwu-Latitude-5480:/var/www/dmeta/o/auth/Graph/dmeta$ cat BYRAtdTkJfw\=\=/Hzn18Vs0lg\=\= ;ec
ho
{"t":"2026-01-22T07:12:52.930Z","m":"like","c": "<h4 id=\"-solution-for-truly-decentralised-internet\">\u2013 Solution for Truly Decentralised Internet<a class=\"anchorjs-link \" href=\"#-solution-for-truly-decentralised-internet\" aria-label=\"Anchor\" data-anchorjs-icon=\"\ue9cb\" style=\"font: 1em \"/> 1 anchorjs-icons; padding-left: 0.375em;</a></h4>","x":"search","u":{"href":"https://omnixtar.github.io/I3P/","origin":"https://omnixtar.github.io","protocol":"https","host":"omnixtar.github.io","hostname":"omnixtar.github.io","port":"","pathname":"/I3P/","search":"","hash":""}}
hongwu@hongwu-Latitude-5480:/var/www/dmeta/o/auth/Graph/dmeta$ ls -lt|head
total 17
drwxr-xr-x 1 root root 4096 Jan 22 15:58 .
drwxr-xr-x 1 root root 4096 Jan 22 15:58 ..
drwxr-xr-x 1 root root 4096 Jan 22 15:58 .git
drwxr-xr-x 1 root root 4096 Jan 22 15:58 .github
drwxrwxr-x 1 root root 4096 Jan 22 15:58 .vscode
drwxr-xr-x 1 root root 4096 Jan 22 15:58 I3P
drwxr-xr-x 1 root root 4096 Jan 22 15:58 I3P-Assets
drwxr-xr-x 1 root root 4096 Jan 22 15:58 I3P-Data
drwxr-xr-x 1 root root 4096 Jan 22 15:58 I3P-Scripts
drwxr-xr-x 1 root root 4096 Jan 22 15:58 I3P-Tools
```

Dual chained properties replace many (all?) database functionalities?

- chained strings: hashcodes can be chained (concatenated) with strings
- chained functions (hash of a string is a string)

$H(s) \rightarrow s, H(H(s)) \rightarrow s, \dots, H^n(s) \rightarrow s$

Example: h_f0 in ajax response

Reason for lightweight “FORTH card computer” implementation →
“database-less”

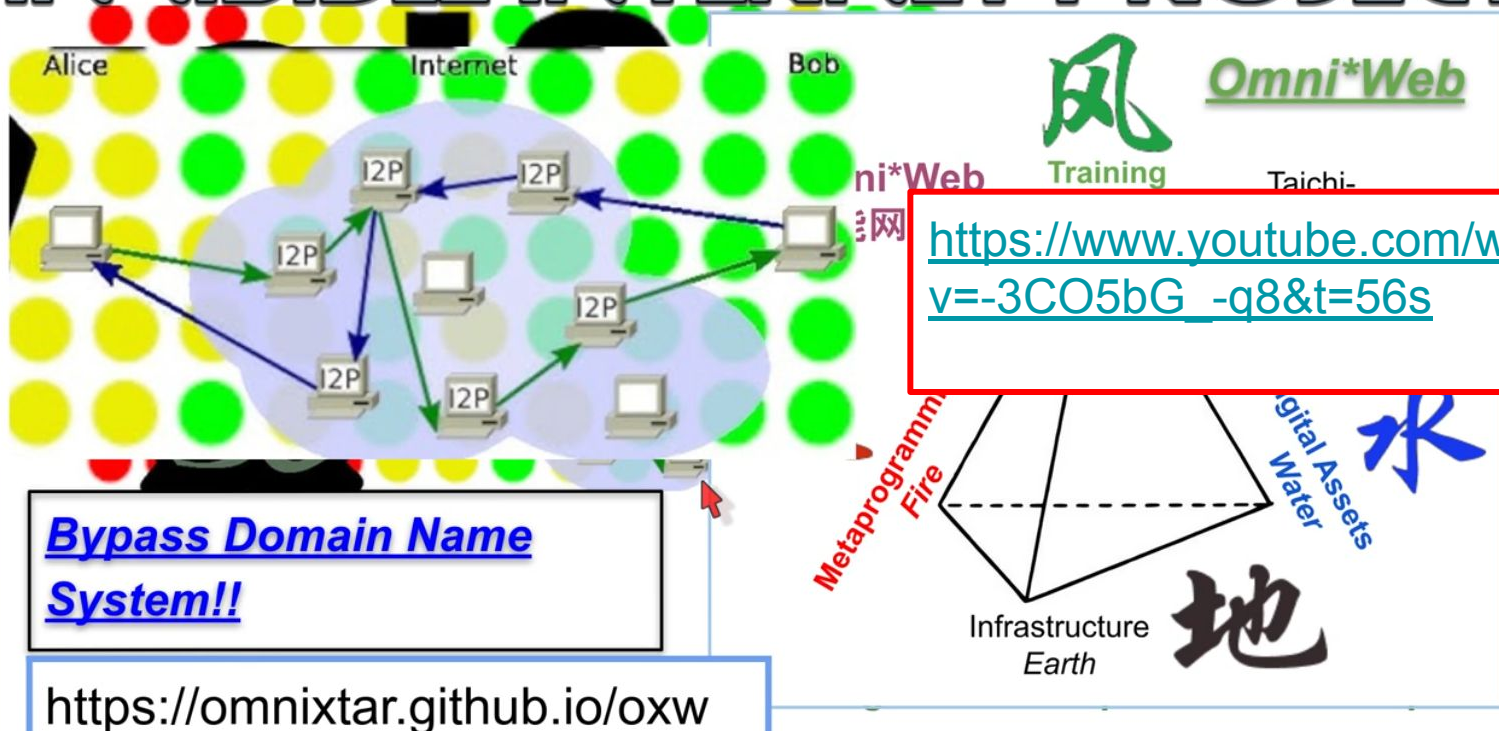
```
MDEtMjJUMDk6MzA6MDIuMzM5WilsImhfZjAiOiJlYkNUMF
BtXzJRPT0ifQ==", "b64d": "jd:", "array": "swap:", "ap:", "2dup:
", "acdwa": "over:", "HF0", "bv:", "a_cmd", "gbv:", "HF0", "i:", "0",
", "i:", "h_f0", "usa:", "je:", "dup:", "hbp", "2", "pick:", "array:", "swap
:", "r:", "apk:", "isodt:", "t:", "apk:", "dup:", "je:", "dup:", "hbp", "2", "
pick:", "swap:", "h:", "apk:", "gbv:", "HF0", "i:", "swap:", "ap:", "je:",
", "ON", "ECHO", "bv:", "dup:", "dup:", "ec:", "hbp", "9", "pick:", "sw
ap:", "2", "msss:", "w:", ":", ":", ":", "1 => 'jd:', 2 => 'cdw:', 3 => 'am:', 4
=> 'lcdw:', 5 => 'B_AUTH1', ), 1 => 'Graph/dmeta/
BYRAtdTkJFw==', 2 => array ( 0 => array ( 'a_cmd' => array (
0 => 's:', 1 => ';', ), 't_f0' => '2026-01-22T09:30:02.339Z',
'h_f0' => 'HbCT0Pm_2Q==', ), ), 3 => 'Graph/dmeta/
BYRAtdTkJFw==', )
[{"a_cmd":
["s:", ":", "t_f0": "2026-01-22T09:30:02.339Z", "h_f0": "HbCT0
Pm_2Q=="], {"r": "Graph/dmeta/
BYRAtdTkJFw==", "t": "2026-01-22T17:30:02.444+0000", "h": "
BNQgJcsdnA=="}]
```

☐ Don't allow 866196a35a9f.ngrok-free.app to prompt you again

OK



INVISIBLE INTERNET PROJECT



Omni*Web: Set Up YrOwn GLOBALLY ACCESSIBLE Web Server w/o Domain Name System in 4 Simple Steps (I2P)



MI4 Metaprogramming in Forth
56 subscribers



12



Share



Download



Clip



ONCE YOU UNWRAP
YOU CAN'T GO BACK

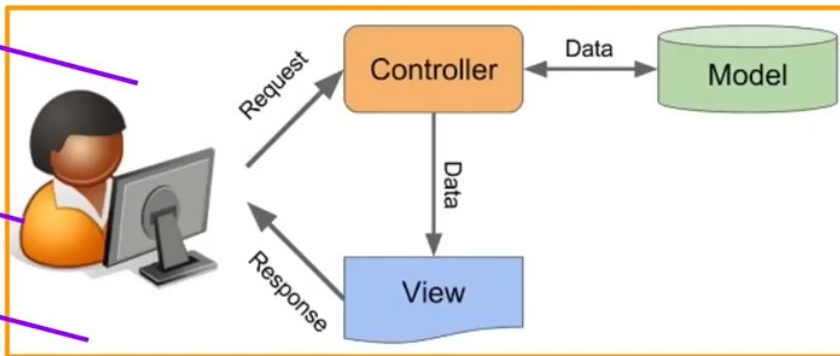




Omni*Web

Crypto-
Meta-
Programming
(CMP)

https://www.youtube.com/watch?v=P_M3PVn9J7I



Omni*Web: Crypto-Metaprogramming (CMP) as alternative to Model-View-Controller; towards Metanarchy



MI4 Metaprogramming in Forth
56 subscribers



Share



Download



Clip



All

Computer programming

Computer Science



Singapore's Tharman Destroys
an Arrogant BBC Host



BITCOIN

Bitcoin: Technical Concepts

Bech32
Blockchain Diagram
Bitcoin Encryption
Creating forks
Bitcoin mining
Blockchain
Bitcoin Improvement Proposals
Pay-to-Script Hash
Proof of Keys
UTXO
User Activated Soft Fork
OmniBOLT
Blockchain (database)
Segregated Witness
Lightning Network
Hashed Timelock Contracts
NSequence
Bitcoin Emission
Block timestamp

Pay-to-PubKey-Hash (Pay-to-Public-Key-Hash, P2PKH) is the basic form of making a [transaction](#) and is the most common form of transaction on the [Bitcoin network](#). Transactions that pay to a [Bitcoin address](#) contain P2PKH scripts that are resolved by sending the public key and a digital signature created by the corresponding [private key](#).

The ScriptPubKey and ScriptSig for a transaction is shown below:

Table of Contents



1. Pay-to-Pubkey Hash
 - 1.1. Pay-to-PubKey-Hash Review
 - 1.2. Pay-to-PublicKey Hash Example
 - 1.3. See also
 - 1.4. References

[https://bitcoinwiki.org/wiki/
pay-to-pubkey-hash](https://bitcoinwiki.org/wiki/pay-to-pubkey-hash)

Pay-to-PubKey-Hash Review

Two types of payment are referred as P2PK (pay to public key) and P2PKH (pay to public key hash).

[Satoshi](#) later decided to use P2PKH instead of P2PK for two reasons:

Elliptic Curve [Cryptography](#) (the cryptography used by your public key and [private key](#)) is vulnerable to a modified Shor's [algorithm](#) for solving the discrete logarithm problem on elliptic curves. In plain English, it means that in the future a quantum computer might be able to retrieve a private key from a public key. By publishing the public key only when the coins are spent (and assuming that addresses are not reused), such attack is rendered ineffective.

With the hash being smaller (20 bytes) it is easier to print and easier to embed into small storage mediums like





BITCOIN

Bitcoin: Technical Concepts

Rech32

Table of Contents



- *alternative solution to Bitcoin and blockchain technologies for managing ownerships of digital assets*

“Who owns digital assets?”

“Buy, sell, transfer, royalties?”

- *Omnihash & DJSON
Decentralised JSON*

Pay-to-PubKey-Hash (Pay-to-Public-Key-Hash, P2PKH) is the basic form of making a [transaction](#) and is the most common form of transaction on the [Bitcoin network](#). Transactions that pay to a [Bitcoin address](#) contain P2PKH scripts that are resolved by sending the public key and a digital signature created by the corresponding [private key](#).

The ScriptPubKey and ScriptSig for a transaction is shown below:

[https://bitcoinwiki.org/wiki/
pay-to-pubkey-hash](https://bitcoinwiki.org/wiki/pay-to-pubkey-hash)

ash Review

ed as P2PK (pay to public key) and P2PKH (pay to public key hash).

PKH instead of P2PK for two reasons:

cryptography used by your public key and [private key](#)) is vulnerable to a living the discrete logarithm problem on elliptic curves. In plain English, it um computer might be able to retrieve a private key from a public key. By hen the coins are spent (and assuming that addresses are not reused), such

bytes) it is easier to print and easier to embed into small storage mediums like



works
Softfork
Scalability
Adaptive difficulty
CVE-2012-4684-new
CVE-2013-2293
Nanopayments
Block weight
BIP UNOFFICIAL DRAFT 0
Ideal Properties of Digital
Commodities
Address reuse
Hashlock
Contingency plans
Offline transactions
Off-Chain Transactions
Funding network security
Bitcoin scalability problem
Segwit support
CVE-2012-3789
Proof of Ownership
Dump format
Test Cases
Hot wallet
Dominant Assurance Contracts
Bitcoin Binary Data Protocol
Coin analogy
CVE-2012-4683
BIP Draft – Instant Partial

A [Bitcoin address](#) is only a hash, so the sender can't provide a full public key in scriptPubKey. When redeeming coins that have been sent to a [Bitcoin](#) address, the recipient provides both the signature and the public key. The [script](#) verifies that the provided public key does [hash](#) to the hash in scriptPubKey, and then it also checks the signature against the public key.

Checking process:

Stack	Script	Description
Empty.	<sig> <pubKey> OP_DUP OP_HASH160 <pubKeyHash> OP_EQUALVERIFY OP_CHECKSIG	scriptSig and scriptPubKey are combined.
<sig> <pubKey>	OP_DUP OP_HASH160 <pubKeyHash> OP_EQUALVERIFY OP_CHECKSIG	Constants are added to the stack.
<sig> <pubKey> <pubKey>	OP_HASH160 <pubKeyHash> OP_EQUALVERIFY OP_CHECKSIG	Top stack item is duplicated.
<sig> <pubKey> <pubHashA>	<pubKeyHash> OP_EQUALVERIFY OP_CHECKSIG	Top stack item is hashed.
<sig> <pubKey> <pubHashA> <pubKeyHash>	OP_EQUALVERIFY OP_CHECKSIG	Constant added.
<sig> <pubKey>	OP_CHECKSIG	Equality is checked between the top two stack items.
true	Empty.	Signature is checked for top two stack items.

A [Bitcoin address](#) is only a hash, so the sender can't provide a full public key in scriptPubKey. When redeeming coins that have been sent to a [Bitcoin](#) address, the recipient provides both the signature and the public key. The [script](#) verifies that the provided public key does [hash](#) to the hash in scriptPubKey, and then it also checks the signature against the public key.

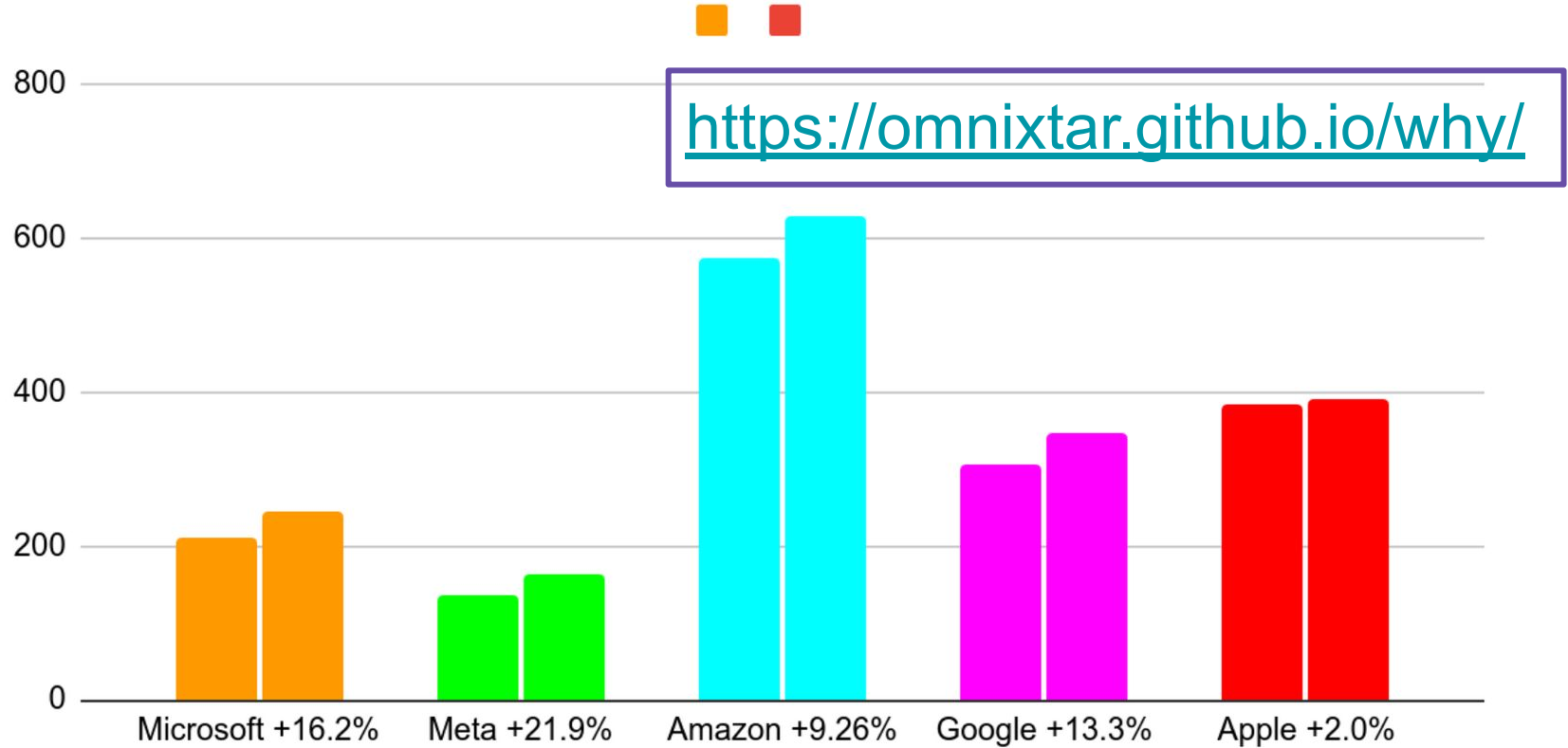
Checking process:

Stack	Script	Description
Empty.	<sig> <pubKey> OP_DUP OP_HASH160 <pubKeyHash> OP_EQUALVERIFY OP_CHECKSIG	scriptSig and scriptPubKey are combined.
<sig> <pubKey>	OP_DUP OP_HASH160 <pubKeyHash> OP_EQUALVERIFY OP_CHECKSIG	Constants are added to the stack.
	P_HASH160 <pubKeyHash> P_EQUALVERIFY OP_CHECKSIG	Top stack item is duplicated.
	pubKeyHash> OP_EQUALVERIFY P_CHECKSIG	Top stack item is hashed.
	P_EQUALVERIFY OP_CHECKSIG	Constant added.
	P_CHECKSIG	Equality is checked between the top two stack items.
		Signature is checked for top two stack items.

- *FORTH used in both Bitcoin (Blockchains) & Omnihash/DJSON (Phoscript)*
- *“Managing ownerships of digital assets” vs. “who pays tariffs”*

MMAGA Revenues 2023/24 (USD billions)

Total 2024: USD 1.776 Trillion (+10.27%)



Bitcoin uses FORTH like syntax to verify the hash of public key.

Stack	Script
Empty.	<sig> <pubKey> OP_DUP OP_HASH160 <pubKeyHash> OP_EQUALVERIFY OP_CHECKSIG

Hilbert Hotel with Infinite Cabinets: Recursion added since 1924
Hilbert Metatron / Hilbertron
virtual objects in Hilbert world:
folders, documents, trains,
ships, etc. labeled w/ hash
numbers / hash code strings



Hilbert Hotel: has infinite number of rooms.

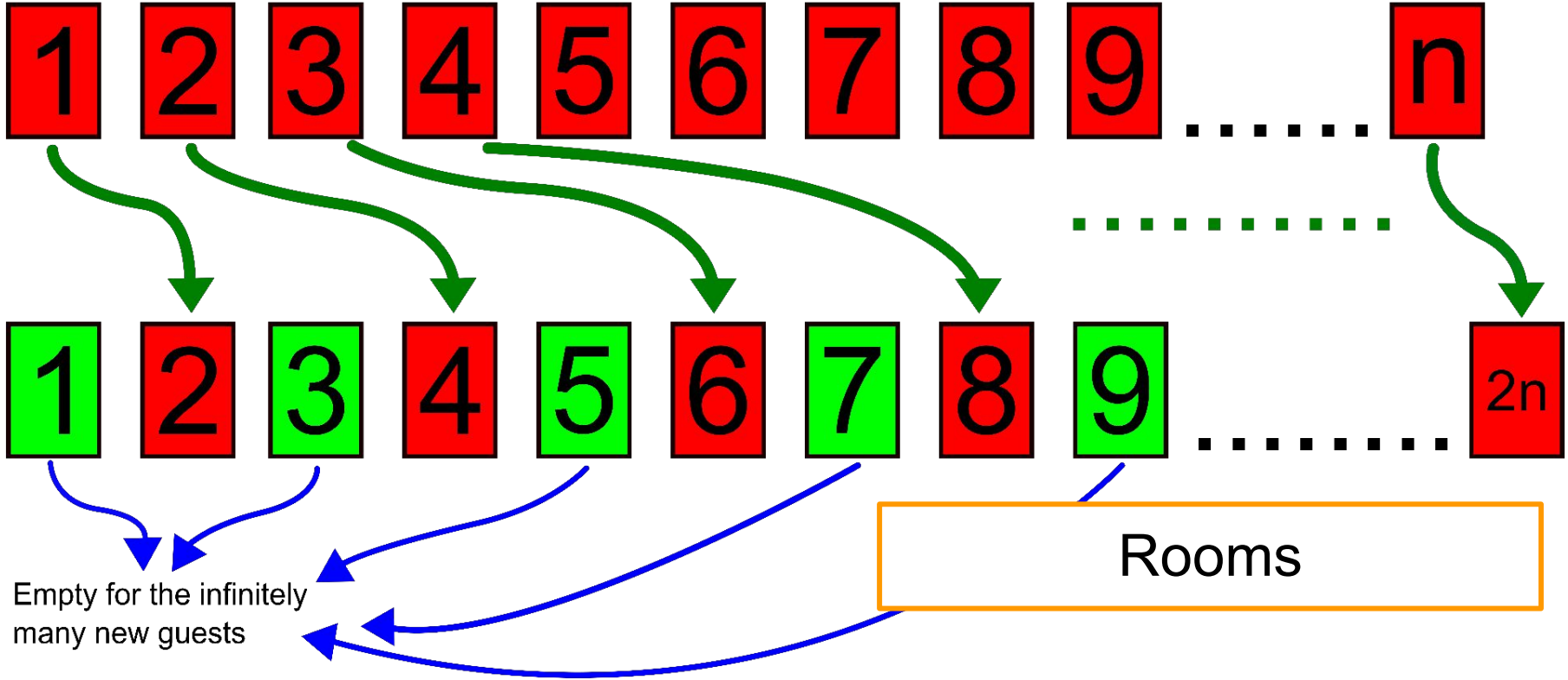
Room number is a hash number.

New: Each room has an Infinite Cabinet (Recursion)

Each drawer is labeled with hash number.



Guests



Empty for the infinitely
many new guests

Rooms

- Bitcoin uses FORTH like syntax to verify the hash of public key.

1. Extend the use of hash of public key as user identifier in decentralised network and explore various properties of hash codes,
2. Omnihash, literally meaning a hash code for representing all types of digital assets, and DJSON decentralized JSON, a special type of JSON string containing at least one Omnihash
3. Omnihash as the basis of hash filesystem, which becomes the foundation of a database system without database server

4. 3X architecture, namely database-less, domain-less and kernel-less

5. Phoscript and Omnihash as used in Web and mobile applications,
Phoscript metaprogramming shell: ported to any host programming language, including front end as well as back end environments

Phase II of 3X architecture: Sandwich model of code substitution
substitute user space services such as Web server and I2P routers with FORTH or Phoscript exclusive code base.

- Bitcoin uses FORTH like syntax to verify the hash of public key.

1. Extend the use of hash of public key as user identifier in decentralised network and explore various properties of hash codes,
2. Omnihash, literally meaning a hash code for representing all types of digital assets, and DJSON decentralized JSON, a special type of JSON string containing at least one Omnihash
3. Omnihash as the basis of hash filesystem, which becomes the foundation of a database system without database server.

4. 3X architecture, namely database-less, domain-less and kernel-less

5. Phoscript and Omnihash as used in Phoscript metaprogramming shell including front end as well as back end. Phase II of 3X architecture: Sandboxing to substitute user space services with FORTH or Phoscript exclusive code.

How to relate “ownership of digital assets” to
Ordinary Users
(who only listen to MMAGA)
(Microsoft Meta Amazon Google Apple)
or FORTH Programmers
(who may not be familiar with JSON
or database programming)
???

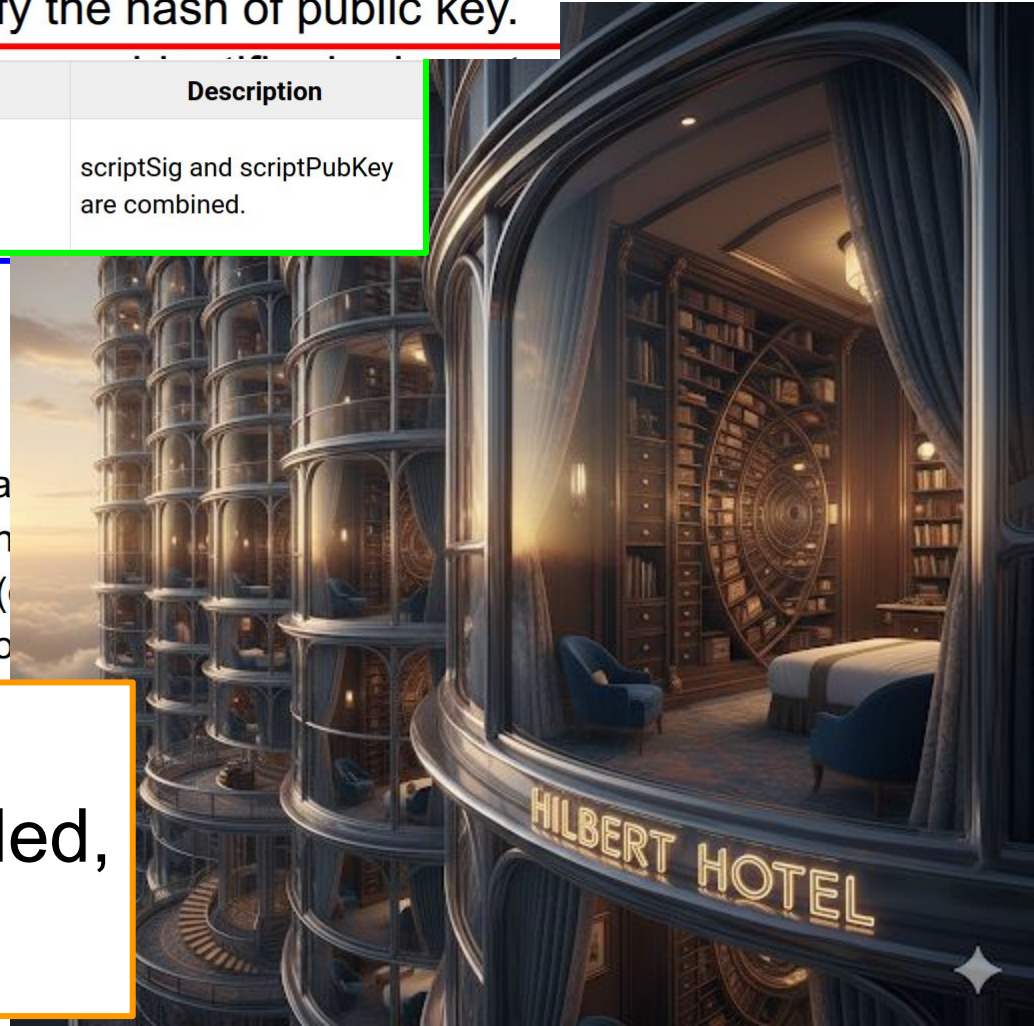
Bitcoin uses FORTH like syntax to verify the hash of public key.

Stack	Script	Description
Empty.	<sig> <pubKey> OP_DUP OP_HASH160 <pubKeyHash> OP_EQUALVERIFY OP_CHECKSIG	scriptSig and scriptPubKey are combined.

Phoscript-Linux/C-FORTH Sandwich Model

1. Minimal VM: Linux Java I2P Apache php
2. Sandwich Model (user space programs):
 - a. Top: P2C (Phos to C) P2J (Phos to Java)
[start: 0, end: replace host language fun
 - b. Middle: C-lib, J-lib (Java), PHP-lib in .o (
[start: 100%, end: replaced by Top & Bo

*Hilbert Hotel with Infinite
Cabinets*: Recursion added,
since 1924.



Intro Recap (5 minutes)

Hilbert Hotel, Infinite Cabinets, Infinity Trains (5 minutes)

DJSON Decentralised JSON & Hash Filesystem (10 min)

Examples: omnixtar.github.io/h omnixtar.github.io/contract

3X Architecture (5 min)

Conclusions, Future installment of this series (5 min)



Home



Shorts



Subscriptions



You

All

Shorts

Videos

Unwatched

Watched

Recently uploaded

Live

Subscriptions

Filters



How An Infinite Hotel Ran Out Of Room

34m views · 4 years ago



... to know if you're the manager of the Hilbert Hotel. References: Ewald ...

0:12 Dies ist das Hilbert Hotel und du bist der Manager. Jetzt wirkt es vielleicht so, als ob du für jeden der...

4K Subtitles



The Infinite Hotel Paradox - Jeff Dekofsky

25m views · 11 years ago



Sign up for our newsletter and never miss an animation: <http://bit.ly/TEDEdNewsletter> View full lesson: ...

Subtitles



hilbert hotel digital asset ownership



AI Mode All Images News Videos Shopping Short videos More ▾ Tools ▾



AI Overview



The term "Hilbert hotel digital asset ownership" refers to **Hilbert Group**, a Nasdaq-listed investment firm that specializes in quantitative trading strategies and asset management in digital asset markets, including Bitcoin and Ethereum. The name is a reference to Hilbert's paradox of the Grand Hotel, a mathematical thought experiment. [↗](#)

Hilbert Group is a public company that offers traditional investors a regulated entry point to the digital asset space. Its ownership structure involves: [↗](#)

- **Public Shareholders:** The company is listed on the Nasdaq First North Growth Market (ticker HILB B).
- **Key Personnel Ownership:** The Chief Investment Officer and a major shareholder, Russell Thompson, has expressed his intention to significantly increase his ownership in the company through open-market purchases.

About - Hilbert Group

Hilbert is a tech-enabled investment company operate...



www.hilbert.group [↗](#)



Hilbert Group Acquires Renowned Digital Asset Manager ...

25 Nov 2024 — Headquartered in the British Virgin Islands, the firm focuses on digital...



www.hilbert.group [↗](#)

Hilbert Group Acquires 100% of Fast-Growing...

1 Sept 2025 — Hilbert Group AP (Nasdaq: HILB B) today...



Interim Report Q3 2025 In Brief

FINANCIAL EVENTS

Third quarter

- Revenues amounted to KSEK 51,562.6 (KSEK 31,551.7)
- Operating profit/loss (EBIT) totalled KSEK -21,433.3 (KSEK -12,095.7)
- Results for the period were KSEK -54,330.9 (KSEK -18,820.5)
- Net Cash flow from operating activities amounted to KSEK -58,217.7 (KSEK -11,603.3)
- Earnings per share before and after dilution amounted to SEK -0.66 (SEK -0.29)

Year to date January – September

- Revenues amounted to KSEK 142,764.5 (KSEK 70,786.4)
- Operating profit/loss (EBIT) totalled KSEK -51,354.3 (KSEK -33,830.2)
- Results for the period were KSEK -105,228.0 (KSEK -37,337.6)
- Cash flow from operating activities amounted to KSEK -69,077.6 (KSEK -26,100.8)
- Earnings per share before and after dilution amounted to SEK -1.43 (SEK -0.58)

Financial overview

Nine months

Nine months

Full year

Financial overview

KSEK	Q325	Q324	Nine months 2025	Nine months 2024	Full year 2024
Revenue	51,562.6	31,551.7	142,764.5	70,786.4	112,977.5
EBIT	-21,433.3	-12,095.7	-51,354.3	-33,830.2	-49,718.6
Profit/loss for the period	-54,330.9	-18,820.5	-105,228.0	-37,337.6	-43,329.3
Earnings per share	-0.66	-0.29	-1.43	-0.58	-0.60
Equity	-677.4	16,144.7	-677.4	16,144.7	23,101.0
Total assets	88,341.0	55,407.2	88,341.0	55,407.2	64,232.0
Equity/assets ratio, %	-0.01	0.29	-0.01	0.29	0.36

SIGNIFICANT EVENTS

During the third quarter and after the reporting period

- Hilbert Group Taps Four Leading Blockchain Figures to Syntetika Advisory Board – Max Rabinovitch, John Lilic, Vladimir Maslyakov, Chirdeep Chhabra
- Hilbert Group launches a Strategic Crypto Treasury Program; managed by a newly established Treasury Committee chaired by CIO Russell Thompson
- Hilbert Group executes on SEK200+ Million Financing to Support Crypto Treasury Strategy
- Lars Seir Christensen co-founder of Saxo Bank, appointed as an Advisor to the Board

- Deus X Capital commits SEK300million worth of Bitcoin to Hilbert Group's Crypto Treasury Strategy
- Liberty Road Capital's ownership increases; the Board of Directors resolved on a directed issue of 7,012,963 B shares based on the authorisation at the last AGM, as conditions were fulfilled for an additional equity payment
- Hilbert Group closes heavily oversubscribed seed round for Syntetika Tokenisation and Decentralised Trading Platform. The round was heavily oversubscribed, attracting over \$ 6 million in investor interest – six times the original \$1 million target
- Exercise of Hilbert Employee Stock Options between July 2025 & September 2025 by Mr Niclas Sandstrom and other participants; first time since inception
- Hilbert's Group shareholders proposes board changes – Steen Jackobsen, former CIO of Saxo Bank, is nominated to the Board. Niclas Sandstrom, Russell Thompson, David Butler, Stuart Connolly and Erik Nerpin proposed to the Board, Forde Foss Skiftesvik and Tim Grant to step down
- Hilbert Group Enters Flexible 3 Year SEK 150 Million Financing Arrangement with LDA Capital; Hilbert Group gains access not only to capital but also to LDA's global network of TradFi and Web 3.0 portfolio companies, crypto native advisors and strategic investors
- Hilbert Group's Syntetika Welcomes Asia-Based Investment Leader Juliet Tang. Former Investment Manager at Huobi, HTX Ventures & Ryan Horn, Ex-Binance Leader Behind Cristiano Ronaldo's Partnership to the Advisory Board
- Hilbert Group announces early repayment of the Eur2 million senior unsecured convertible bond, due in 2025
- Hilbert Group announced full regulatory approval from the Gibraltar Financial



BITCOIN

Bitcoin: Technical Concepts

Bech32
Blockchain Diagram
Bitcoin Encryption
Creating forks
Bitcoin mining
Blockchain
Bitcoin Improvement Proposals
Pay-to-Script Hash
Proof of Keys
UTXO
User Activated Soft Fork
OmniBOLT
Blockchain (database)
Segregated Witness
Lightning Network
Hashed Timelock Contracts
NSequence
Bitcoin Emission
Block timestamp

Pay-to-PubKey-Hash (Pay-to-Public-Key-Hash, P2PKH) is the basic form of making a [transaction](#) and is the most common form of transaction on the [Bitcoin network](#). Transactions that pay to a [Bitcoin address](#) contain P2PKH scripts that are resolved by sending the public key and a digital signature created by the corresponding [private key](#).

The ScriptPubKey and ScriptSig for a transaction is shown below:

Table of Contents



1. Pay-to-Pubkey Hash
 - 1.1. Pay-to-PubKey-Hash Review
 - 1.2. Pay-to-PublicKey Hash Example
 - 1.3. See also
 - 1.4. References

[https://bitcoinwiki.org/wiki/
pay-to-pubkey-hash](https://bitcoinwiki.org/wiki/pay-to-pubkey-hash)

Pay-to-PubKey-Hash Review

Two types of payment are referred as P2PK (pay to public key) and P2PKH (pay to public key hash).

[Satoshi](#) later decided to use P2PKH instead of P2PK for two reasons:

Elliptic Curve [Cryptography](#) (the cryptography used by your public key and [private key](#)) is vulnerable to a modified Shor's [algorithm](#) for solving the discrete logarithm problem on elliptic curves. In plain English, it means that in the future a quantum computer might be able to retrieve a private key from a public key. By publishing the public key only when the coins are spent (and assuming that addresses are not reused), such attack is rendered ineffective.

With the hash being smaller (20 bytes) it is easier to print and easier to embed into small storage mediums like



works
Softfork
Scalability
Adaptive difficulty
CVE-2012-4684-new
CVE-2013-2293
Nanopayments
Block weight
BIP UNOFFICIAL DRAFT 0
Ideal Properties of Digital
Commodities
Address reuse
Hashlock
Contingency plans
Offline transactions
Off-Chain Transactions
Funding network security
Bitcoin scalability problem
Segwit support
CVE-2012-3789
Proof of Ownership
Dump format
Test Cases
Hot wallet
Dominant Assurance Contracts
Bitcoin Binary Data Protocol
Coin analogy
CVE-2012-4683
BIP Draft – Instant Partial

A [Bitcoin address](#) is only a hash, so the sender can't provide a full public key in scriptPubKey. When redeeming coins that have been sent to a [Bitcoin](#) address, the recipient provides both the signature and the public key. The [script](#) verifies that the provided public key does [hash](#) to the hash in scriptPubKey, and then it also checks the signature against the public key.

Checking process:

Stack	Script	Description
Empty.	<sig> <pubKey> OP_DUP OP_HASH160 <pubKeyHash> OP_EQUALVERIFY OP_CHECKSIG	scriptSig and scriptPubKey are combined.
<sig> <pubKey>	OP_DUP OP_HASH160 <pubKeyHash> OP_EQUALVERIFY OP_CHECKSIG	Constants are added to the stack.
<sig> <pubKey> <pubKey>	OP_HASH160 <pubKeyHash> OP_EQUALVERIFY OP_CHECKSIG	Top stack item is duplicated.
<sig> <pubKey> <pubHashA>	<pubKeyHash> OP_EQUALVERIFY OP_CHECKSIG	Top stack item is hashed.
<sig> <pubKey> <pubHashA> <pubKeyHash>	OP_EQUALVERIFY OP_CHECKSIG	Constant added.
<sig> <pubKey>	OP_CHECKSIG	Equality is checked between the top two stack items.
true	Empty.	Signature is checked for top two stack items.

Hilbert Hotel: has infinite number of rooms.

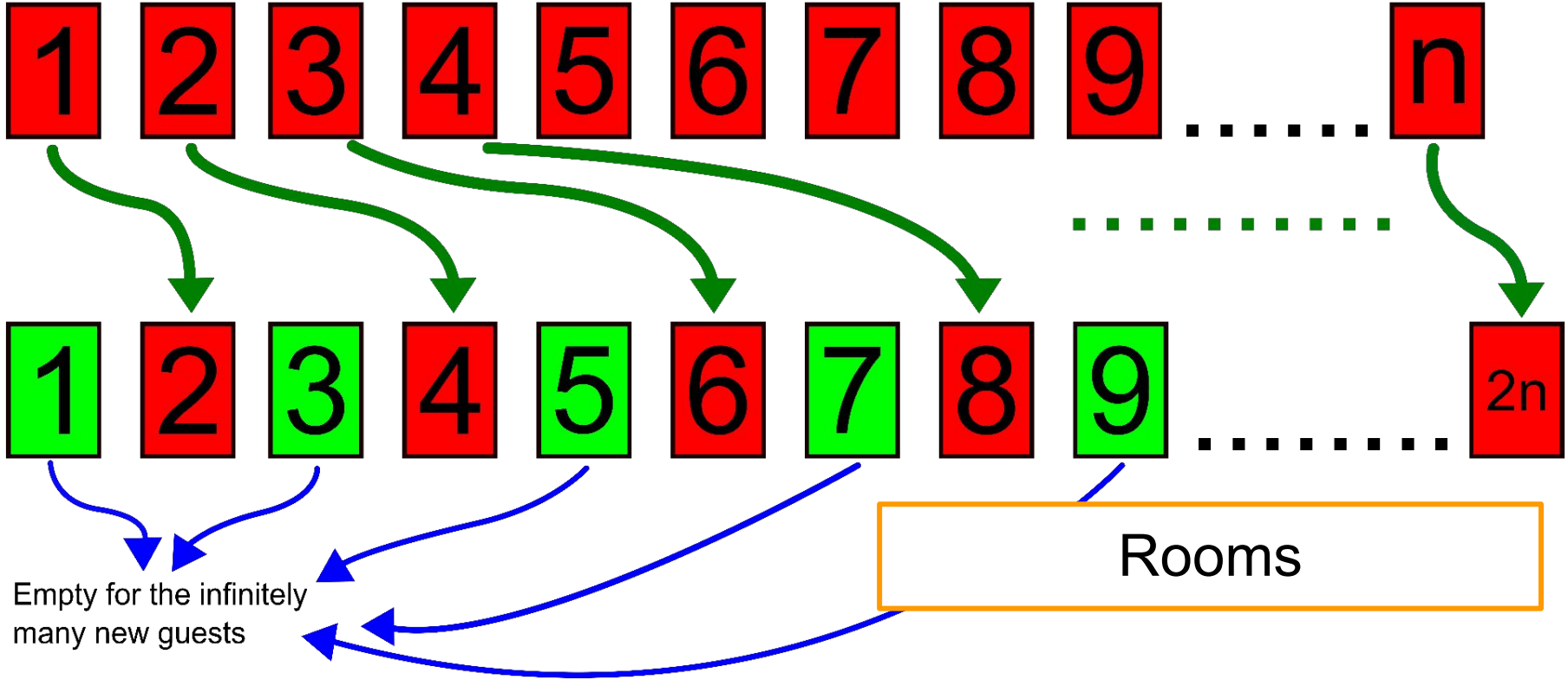
Room number is a hash number.

New: Each room has an Infinite Cabinet (Recursion)

Each drawer is labeled with hash number.



Guests



Rooms

Empty for the infinitely
many new guests

Hilbert Hotel with Infinite Cabinets

What is the practical significance of this?

Imagine: Every comment, photo, video, document YOU posted on the Internet is like a piece of paper or file in Infinite Cabinet, each labeled with a hash number.



Activities

Google Chrome

Kha 18 Dis, 20:02

omnixtar.github.io/h/

x

+

←

→

↻

🔍 omnixtar.github.io/h/

Omni* Omnihash

Omnihash

GenKeys

Save

Help

Load Keypair

Choose File

No file chosen

Your Hash is ECClo9IBig==


Your Parent Hash is root

You have 0 downlines.

{ "x": "REG", "p": "root", "h": "ECClo9IBig==", "t": "2025-12-18T11:02:03.028Z" }

- You may copy & save this DJSON

Decentralised JSON ...



Welcome to Omni*Hilbert Hotel

<https://omnixtar.github.io/h/>

Practical use: manage SVFIG memberships or any organisations!!

Load Keypair

Choose File

No file chosen

Your Hash is ECClo9IBig==
Your Parent Hash is root
You have 0 downlines.

```
{"x":"REG","p":"root","h":"ECClo9IBig==","t":"2025-12-18T11:02:03.028Z"}
```

- You may copy & save this DJJSON
Decentralised JSON ...



*Your User Identifier is the most
valuable digital asset!!*

*Hash of public key!! (Remember &
Repeat!!)*

All subsequent hashcode strings &
Omnihash(es) contain parts of User
ID (hash of public key)!!

Activities

Google Chrome

Sab 20 Dis, 20:10

zh

🔊

🔋

omnixtar.github.io/h/?ECClo9lBig==

omnixtar.github.io/h/?ECClo9lBig==

New Chrome available

Omni* Omnihash

Omnihash

GenKeys

Save

Help

Load Keypair


Choose File

No file chosen

Your Hash is H-dYqaYMkQ==

Your Parent Hash is ECClo9lBig==

You have 0 downlines.



DevTools - omnixtar.github.io/h/?ECClo9lBig==

Elements

Console

Sources

>>

4

1

2

🔄

⚙️

⋮

🎬

🚫

top

👁

🔍 Filter

Default levels

6 Issues: 2 1 3 | 1 hidden ⚙️

-----END PUBLIC KEY-----

ECClo9lBig==[h/?ECClo9lBig==:225](https://omnixtar.github.io/h/?ECClo9lBig==:225)

A2RJY6IEjA==[h/?ECClo9lBig==:234](https://omnixtar.github.io/h/?ECClo9lBig==:234)

⚠️ The resource [h/?ECClo9lBig==:1](https://omnixtar.github.io/h/?ECClo9lBig==:1)
<https://corsmirror.com/graphql?body=%7B%22persistedQueryNa...>
was preloaded using link preload but not used within a few
seconds from the window's load event. Please make sure it
has an appropriate `as` value and it is preloaded
intentionally.

>

⋮

Console

What's new

AI assistance

🗑

✕

Omni* Omnihash

Omnihash

GenKeys

Save

Help

Load Keypair

Choose File

No file chosen

Your Hash is H-dYqaYMkQ==

Your Parent Hash is ECClo9lBig==

You have 0 downlines.



Elements

Console

Sources



top



Filter

Default levels

6 Issues: 2 1 3 | 1 hidden

was preloaded using link preload but not used within a few seconds from the window's load event. Please make sure it has an appropriate 'as' value and it is preloaded intentionally.

> f('a h53:')

< undefined

> t()

< 7929297801672961

> s[s.length-1]

< 7929297801672961

> Math.pow(2,53)

< 9007199254740992

>



Console

What's new

AI assistance





14ce0dd ▾

omnixtar.github.io / [h](#) / [Phos](#) / [librsa.js](#)

↑ Top

Code

Blame

Raw



```
193     }
194
195   ▾ const cyrb53 = function(str, seed = 0) {
196       let h1 = 0xdeadbeef ^ seed, h2 = 0x41c6ce57 ^ seed;
197       for (let i = 0, ch; i < str.length; i++) {
198           ch = str.charCodeAt(i);
199           h1 = Math.imul(h1 ^ ch, 2654435761);
200           h2 = Math.imul(h2 ^ ch, 1597334677);
201       }
202       h1 = Math.imul(h1 ^ (h1>>>16), 2246822507) ^ Math.imul(h2 ^ (h2>>>13), 3266489909);
203       h2 = Math.imul(h2 ^ (h2>>>16), 2246822507) ^ Math.imul(h1 ^ (h1>>>13), 3266489909);
204       return 4294967296 * (2097151 & h2) + (h1>>>0);
205   };
206
207   ... function f_h53()
208   {
209       var S = M.S;
210       S.push(cyrb53(S.pop()));
211   }
212
```



main

omnixtar.github.io / js / omni.js

↑ Top

Code

Blame

Raw



```
35  var bnToB64=function bnToB64(bn){
36    var hex = BigInt(bn).toString(16);
37    if (hex.length % 2) { hex = '0' + hex; }
38    var bin = [];
39    var i = 0;
40    var d;
41    var b;
42    while (i < hex.length) {
43      d = parseInt(hex.slice(i, i + 2), 16);
44      b = String.fromCharCode(d);
45      bin.push(b);
46      i += 2;
47    }
48    return btoa(bin.join(''));
49  }
50  var mkeval_l=function(FN){ // *_l means long form before minified
51    var j1={n:FN}
52    j1.s=eval(j1.n).toString()
53    j1.t=(new Date()).toISOString()
54    var j2={s:JSON.stringify(j1),h:bnToB64(cyrb53(JSON.stringify(j1))) }
```


Activities

Google Chrome

Sab 20 Dis, 21:43

zh

omnixtar.github.io/h/?EC

omnixtar.github.io/h/?ECClo9lBig==

Omni* Omnihash

Omnihash

GenKeys

Save

Help

Load Keypair


Choose File

No file chosen

Your Hash is H-dYqaYMkQ==

Your Parent Hash is ECClo9lBig==

You have 0 downlines.



DevTools - omnixtar.github.io/h/?ECClo9lBig==

Elements

Console

Sources

6 Issues: 2 1 3

```
> f('a h53: b64:')
< undefined
> t()
< 'HCungsl5AQ=='
> f_b64=function(){s.push(bnToB64(s.pop()))}
< f () {s.push(bnToB64(s.pop()))}
>
```

Console

What's new

AI assistance



main

omnixtar.github.io / js / omni.js

↑ Top

Code

Blame

Raw



```
9  function f_path() // path safe b64
10  {
11      // $a = array_pop($S);
12      var $a = s.pop()
13      // echo '  in path: '.$a." ".strpos($a, '/')."\\n";
14      // if (strpos($a, '/')>=0) { $a = str_replace('/', '_', $a); } // echo $a; }
15      if ($a.indexOf('/')>=0) { $a = $a.replaceAll('/', '_'); }
16      // if (strpos($a, '+')>=0) $a = str_replace('+', '-', $a);
17      if ($a.indexOf('+')>=0) { $a = $a.replaceAll('+', '-'); }
18      s.push($a)
19  }
20
21  // mkeval_l("omnihash")
22  // DmII2G1lww==
23  // var omnihash=function(){
24  ✓ var cyrb53=function(str, seed = 0){
25      let h1 = 0xdeadbeef ^ seed, h2 = 0x41c6ce57 ^ seed;
26      for (let i = 0, ch; i < str.length; i++) {
27          ch = str.charCodeAt(i);
28          h1 = Math.imul(h1 ^ ch, 2654435761);
```



<> index.html ×



h > <> index.html > html > body > script > click() callback

```
1    <html>
23   <body>
123  <script type="text/javascript">
405  $('#omni').click(async function () {
411
412      // Get the input and crypted values.
413      var input = $('#pubkey2').val();
414      var crypted = $('#hpbkb').val();
415
416      console.log('bdmhs 2023 12 31 ', S0.K.s_pbk);
417
418      // var H = bnToB64( cyrb53( btoa(S0.K.s_pbk) ) )
419      ...var H = f_path( bnToB64( cyrb53( btoa(S0.K.s_pbk) ) ) ) ...
420      dt = new Date();
421
422      s_up=JSON.stringify({c:$('#input2').val(),x:"REG",p:S_P,h:H,nn:nn,t:dt.
toISOString(),r:$('#hash4').val()});
423
424      $('#dmhs').val( JSON.stringify({c:$('#input2').val(),x:"REG",p:S_P,h:H,
t:dt.toISOString(),r:$('#hash4').val()}) + "\n\n- You may copy & save
```



Omni* Omnihash

[Omnihash](#)[GenKeys](#)[Save](#)[Help](#)

Load Keypair

Choose File

No file chosen

Your Hash is H-dYqaYMkQ==

Your Parent Hash is ECClo9IBig==

You have 0 downlines.



{"x": "R
dYqaY
20T11:
- You m



Elements

Console

Sources



2



top ▼



Filter

Default levels ▼

6 Issues: 2 1 3

```
> f('S0.K.s_pbk ev: b2a: h53: b64: path:')
```

```
< undefined
```

```
> t()
```

```
< 'H-dYqaYMkQ=='
```

```
> S0.K
```

```
< {s_pvk: '-----BEGIN PRIVATE KEY-----\nMIIJQgIBADANBgkqhkiG9w...134o2RsyS0pIXmMUuT/A=\n-----END PRIVATE KEY-----', s_pbk: '-----BEGIN PUBLIC KEY-----\nMIICIjANBgkqhkiG9w0BAQE...D9e2VEnpiX51BBcCAwEAAQ=\n-----END PUBLIC KEY-----', pair: {...}}
```

```
>
```

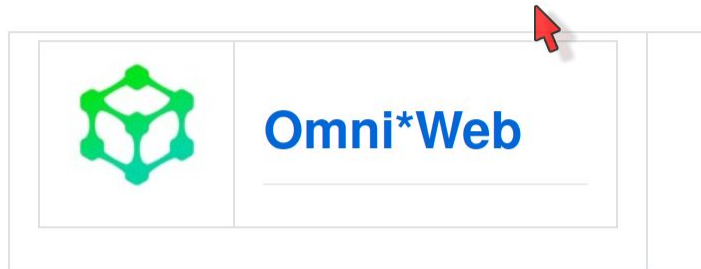
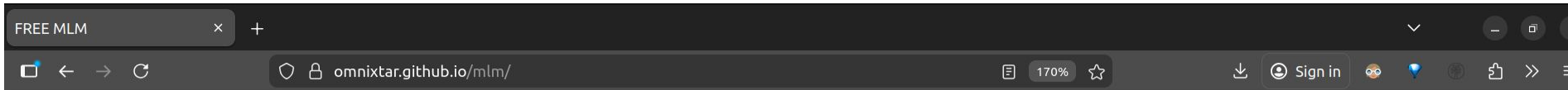


Console

What's new x

AI assistance





Omnihash: THE One & Only FREE Multilevel Marketing

1. Start with no parent hash:

- <https://omnixtar.github.io/h/>
- Demo video 

2. Start with a parent hash:

- <https://omnixtar.github.io/h/?Du8iBzaMkw==>
- Demo video 

*Membership for SVFIG using
Omnihash*

- or any organisations
- charity of commercial
- membership is the most valuable digital asset!!
- omnixtar.github.io/mlm

Omni* Omnihash

OmnihashGenKeysSave

Load Keypair

No file selected.

Your Hash is B3NfOseLg==

Your Parent Hash is undefined

You have 0 downlines.

```
{"x":"REG","h":"B3NfOseLg==","t":"2025-07-02T09:27:17.613Z"}
```

- You may copy & save this DJSON
Decentralised JSON ...



0:16 / 0:32

<https://www.youtube.com/watch?v=rH7oH7b47LE>

Get Omnihash with NO parent hash



Omni* Omnihash

Omnihash

GenKeys

Save

Load Keyr

Browse... No file selected.

Your Hash is BPcCpnRKrw==
Your Parent Hash is Du8iBzAkw==
You have 0 downlines.

```
{"x": "REG", "p": "Du8iBzAkw==", "h": "BPcCpnRKrw==", "t": "2025-07-02T09:28:25.026Z"}
```

- You may copy & save this DJSON
Decentralised JSON ...

<https://www.youtube.com/watch?v=Kp5396NiWQk>



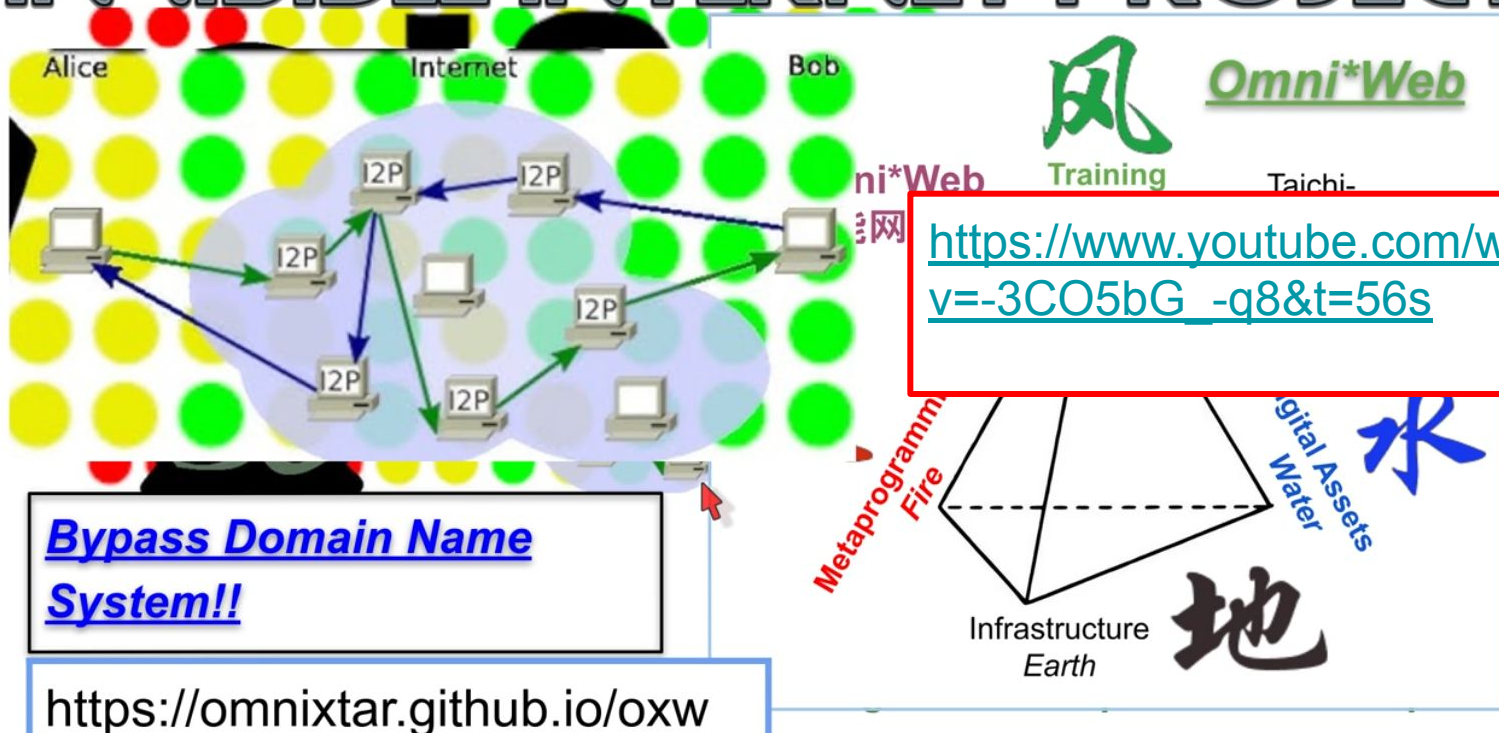
0:15 / 0:35

Get Omnihash WITH parent hash





INVISIBLE INTERNET PROJECT



Omni*Web: Set Up YrOwn GLOBALLY ACCESSIBLE Web Server w/o Domain Name System in 4 Simple Steps (I2P)



MI4 Metaprogramming in Forth
56 subscribers



12



Download



Clip

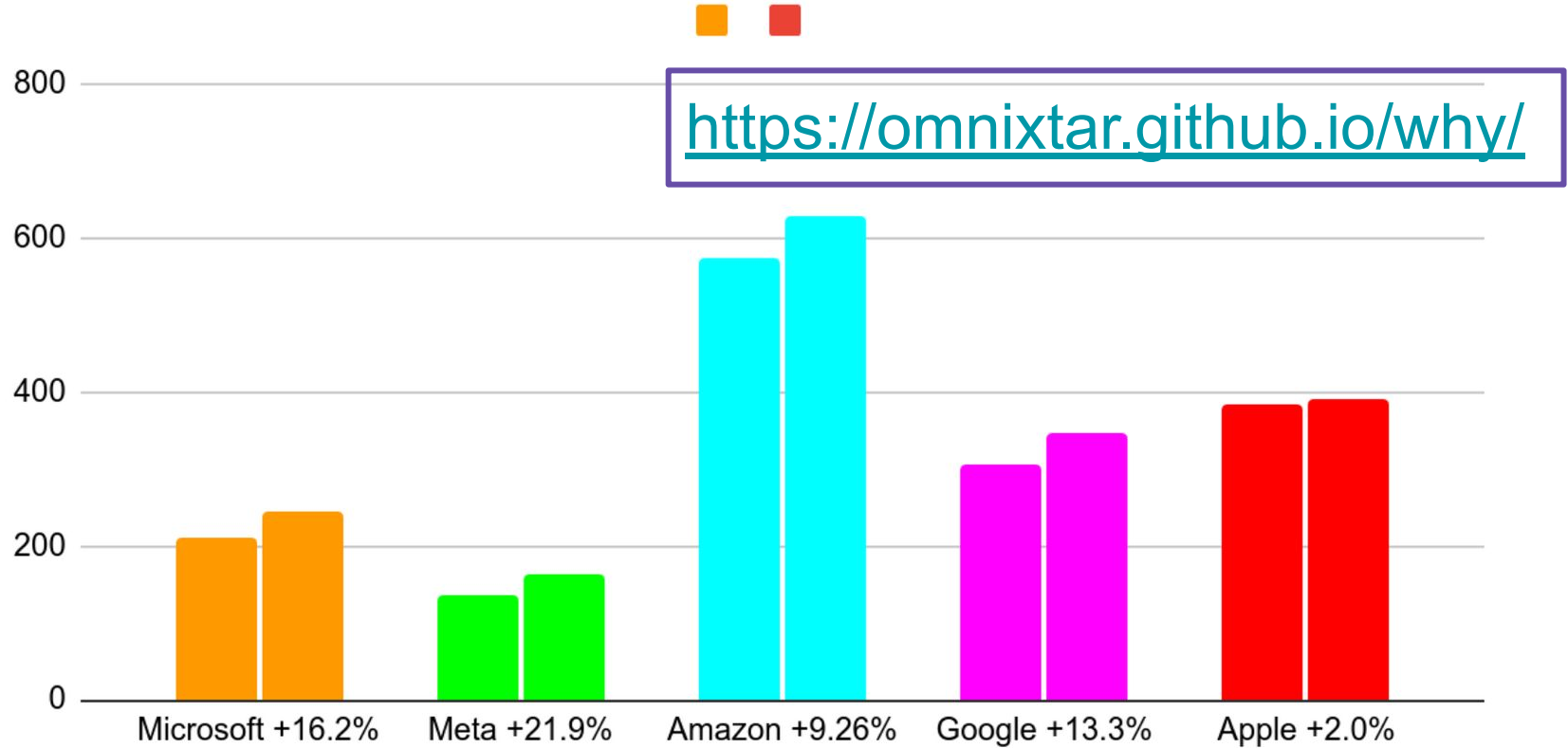


ONCE YOU UNWRAP
YOU CAN'T GO BACK



MMAGA Revenues 2023/24 (USD billions)

Total 2024: USD 1.776 Trillion (+10.27%)



Activities Firefox

yo6sgmfq7pfvvp2e4kcuhtfg7wfltt63igwcukhbmumq6lu3a3a.b32.i2p/oxm/oxm.php?nn=adam

adam session_id:adam-fer4f9qk69m9nj562u7kj5n2u9_31
session_id:adam-fer4f9qk69m9nj562u7kj5n2u9_31

Omni*Search Demo

Copy the following code and execute it in

f(' B_F '+' JSON.stringify({w:'like', t:'recipe

1602 Phos 1611 S 1621 B_F start omni:

```
fgl_s 340 < 2 > array ( 0 => array ( 0 => '{"B_AUTH":  
["SESSION:", "AUTH", "i:", "s:", ""]; 1 => "jd:", 2 => "cdw:", 3 => "am:", 4 =>  
'lcdw:', 5 => 'B_AUTH', ), 1 => array ( 'SP' => '93b164de', 'PASS' =>  
'20250211_220742', 'PBK' =>  
'LS0tLS1CRUdJTiBQVUJMSUMgS0VZLS0tLS0KU0lqQU5CZ2ZkaGt  
pRzI3MEJBUUVGQUFPQ0FhOEFNSUIDQ2dLQ0FnRUFyUUR1TWJCL3NS  
c2dXNmNSeGhOUdILWknVRkjdERSVXV1b3ozSC9XT05nYUQWEXg  
NHVwZFNUTHU0QXhWNWtkUnFBTWlMRzI2ZTZkajBqc2EzS3MvQlVNd  
0p3VW1Jd1EYmtFL0NsZFdsdCwceExVXNnQUI2OVNGOTRCSGpmW  
EZMODVJL2Vaa1NSem5RWHRsZnQvYzZlTG4yMDYyY2ZlQ0l0dE11Z2R  
vNHb5anBLVzNuS095SIAwZ2Nrckt0dWJCSG1HSjIqOW5YeXZraSszUU  
V4bS1370JlUwY2ZlQ0l0dE11Z2RvNHb5anBLVzNuS095SIAwZ2Nrckt0dWJCSG1HSjIqOW5YeXZraSszUU
```

post ajax:

DSC

cmd

123

1. Discovered by accident, PBK login video, show again.
2. DJSON video https://youtu.be/_w-E4Z4Ju5s?t=250

Login using Hash of public key.

Don't allow yo6sgmfq7pfvvp2e4kcuhtfg7wfltt63igwcukhbmumq6lu3a3a.b32.i2p to prompt you again

OK SEND GET

DJSON: Decentralised JSON -- ULTIMATE generalisation of Bitcoin address and blockchain, w/o MINING.



MI4 Metaprogramming in Forth
56 subscribers



Share



Download



Clip



All

For you



The A.I. Stock Bubble |
ChatGPT Grok Go Erotic |

Activities

Firefox

Sel 11 Feb, 22:07

yo6sgmfq7pfvvp2e4kcuhtfg7wfltt63igwcukhbmumq6lu3a3a.b32.i2p/oxm/oxm.php?nn=adam

170%

adam session_id:adam-fer4f9qk69m9nj562u7kj5n2u9_31
session_id:adam-fer4f9qk69m9nj562u7kj5n2u9_31

Omni*Search Demo

Copy the following code and execute it in

f(': B_F ' + JSON.stringify({w:'like', t:'recipe

1602 Phos 1611 S 1621 B_F start omni:

fgl_s 340 < 2 > array (0 => array (0 => {
'lcdw:', 5 => 'B_AUTH',), 1 => array ('SP'
'LS0tLS1CRUdJTiBQVUJMSUMgS0VZLS0tLS0KU0Q0LQ0FhRUFyU0R1TWJCL3NS
'NN' => 'adam',),)
fgl_s 340 < 4 > array (0 => array (0 => {
["SESSION:", "AUTH", "i:", "PBK", "i:", "o_ha
1 => 'jd:', 2 => 'cdw:', 3 => 'am:', 4 => 'lcdw'
array ('a_cmd' => array (0 => 's:', 1 => ';
'Graph/dmeta/CXAGcRkevA==',) [{"a_cm
{ "r": "Graph/dmeta/CXAGcRkevA==", "t":
fgl_s 340 < 2 > array (0 => array (0 => {
'lcdw:', 5 => 'B_AUTH',), 1 => NULL,)
fgl_s 340 < 2 > array (0 => array (0 => {
["B_AUTH": ["SESSION:", "AUTH", "i:", "s:", "i:", "o_ha
'lcdw:', 5 => 'B_AUTH',), 1 => array ('SP' => 'e3235h7f', 'PASS' => '20250211_042724', 'PBK' =>

yo6sgmfq7pfvvp2e4kcuhtfg7wfltt63igwcukhbmumq6lu3a3a.b32.i2p

post ajax:
fgl_s 340 < 2 > array (0 => array (0 => {
["SESSION:", "AUTH", "i:", "s:", "i:", "o_ha
'lcdw:', 5 => 'B_AUTH',), 1 => array ('SP' => '93b164de', 'PASS' => '20250211_220742', 'PBK' => 'LS0tLS1CRUdJTiBQVUJMSUMgS0VZLS0tLS0KU0Q0LQ0FhRUFyU0R1TWJCL3NSc2dXNmNSeGhOUdlWkNVRkJKdER5VXV1b3ozSC9XT05nYUQWExGHNHvZFNUTHU0QXhWNUtUnFBTWlMRzI2ZTZkajBqczEzS3MvQlVNd0p3VW1Jd1EyYmtFL0NsZFdcDcweExVXNnQUl2OVNGOTRCSGpmWEZMODVJL2Vaa1NSEm5RWHRsZnQvYzZjTG4yMDFyelB3OGL0dEl1Z2RvNHBSanBLVzNuS095SIAwZ2Nrckt0dWJCSG1HSJlQOW5YeXZraSszUUV4bSt3Z0JRWFhOR28yaHl2UF83akhTSVUvdhwbzZKYkpiVm8vTlZRTU9vK2tStkLSZW1manpyYTFENzdyZDMrZjMveER4bGQraTg2K3dyRlBKnWzlcDQzoElpK0pZMklkTnZxd1hxaVZSeGlrTU1ENUJpNDNmTnBiS0JsZXJBYSUpY2pRTGhUZEVE1Vlk0d2pUczNMZVMxaytLRHBZQVNNROSITZElsvVWVaSjVNVNDhtMnBodGt1M0V4WHAYa0JtZml2ZmxPRXRRdmdaZm1OQzJPbUcyYXNYZ3pWLO5qZk5kakd2ZWp0VU0YkZ6WfDkZkNRQ3o4UGxpbUZZeWtJZll0eVBMNzQ5MW9tVmZjeUlfUzBWb1hqT1VnblNsdUxPMFBbRk5oODdyTWpvczJYYTg4THpmQmh1UFJIN2pwVXd1Y0pvcG1GU3ZlYUfhdl1NjNW1WNWl1ZUZvMlJlV3UxWlhuK1pzRzhEekdMR3lCci9XYXkuNVo1NktuZzdYVzBMVDRGTlXNWE1VG95VNVNDbi9ua1dYUkp6UlhRcGFwQutMVGNPekFDUEVtak9GYnZEQWNkbGMrVWpBK0U1bXVBYmVtNnNpcVQwN3pzNkt0Qk9rbWRtNVJjMXVvOENBd0VBQVE9PQotLS0tLUVORCBQVUJMSUMgS0VZLS0tLS0=, 'NN' => 'adam',),)
☐ Don't allow yo6sgmfq7pfvvp2e4kcuhtfg7wfltt63igwcukhbmumq6lu3a3a.b32.i2p to prompt you again

DSC

cmd

123

Encrypted_Message

Any_other_remarks

Misc

SEND

GET

adam session_id:adam-fer4f9qk69m9nj562u7kj5n2u9_31
 session_id:adam-fer4f9qk69m9nj562u7kj5n2u9_31

Omni*Search Demo

Copy the following code and execute it in the browser console:

```
f(': B_F ' + JSON.stringify({w:'like', t:'recipient_hash', f:'sender_hash'}) + ' jd: rghe: g_
```

1602 Phos 1611 S 1621 B_F start omni:

```
fgl_s 340 < 2 > array ( 0 => array ( 0 => '{"B_AUTH":["SESSION:","AUTH","i:","s:","  

'lcdw:', 5 => 'B_AUTH', ), 1 => array ( 'SP' => '9fa64665', 'PASS' => '20250210_00:  

'LS0tLS1CRUdJTlBQVUJMSUMgS0VZLS0tLS0KTUIJQ0lqQlU5C72tvaGtnRzI3ME  

'NN' => 'adam', ), )
```

```
fgl_s 340 < 4 > array ( 0 => array ( 0 => '{"B_AUTH1":  

["SESSION:","AUTH","i:","PBK","i:","o_hash.json","l_cdwjs","t  

1 => 'jd:', 2 => 'cdw:', 3 => 'am:', 4 => 'lcdw:', 5 => 'B_AUTH1'  

array ( 'a_cmd' => array ( 0 => 's:', 1 => ';', ), 't_f0' => '2025-0:  

'Graph/dmeta/CXAGcRkevA==', ) [{"a_cmd":["s:",";"], "t_f0":"2  

{"r":"Graph/dmeta/CXAGcRkevA==","t":"2025-02-10T00:38:  

fgl_s 340 < 2 > array ( 0 => array ( 0 => '{"B_AUTH":["SESSI  

'lcdw:', 5 => 'B_AUTH', ), 1 => NULL, )
```

```
fgl_s 340 < 2 > array ( 0 => array ( 0 => '{"B_AUTH":["SESSI  

'lcdw:', 5 => 'B AUTH' ) 1 => array ( 'SP' => 'e3235b7f' 'PASS' => '20250210_042724' 'PBK' =>
```

DSC

cmd

123

Encrypted_Message

Any_other_remarks

Misc

adam

s: x:

AUTH GenKeys SEND GET

Front End

PBK (Public Key)

Trillion Dollar Slide:
defines user identifier!!

Ciphertext

+

PVK (Private Key)

Secret Phrase (SP)

Back End

PBK (Public Key)

+

Secret Phrase (SP)

Ciphertext

Foundation of Bitcoin
user identifier:
hash of public key

Secret Phrase (SP)



Additional Options

Backend may send its PBK (B_PBK) (prefix B) to frontend, so that

- front end may encrypt deciphered secret phrase (SP) with B_PBK,
- then send to Backend, which then decipher it using Backend private key (B_PVK).

```
"req_auth_sc":["b64d:","4","orpb:","hex:","dup2:","rsa:","2","pick:","lkey:","swap:","r  
ecr:","b64e :","ON","ECHO","bv:","ec:","2","pick:","SP","ka:","AUTH","uss:",";"]
```

```
i:","dts:","over:","over:","jsp:","rot:","rot:","swap:","_","jsd:","=chat","dc0:","_","jsd:","rot:","  
"rot:","jsp:","jnl:","swap:","log","\/","jsd:","wa:",";"],"nick":["NN","ka:","AUTH","uss:",";"],"req  
auth_sc":["b64d:","4","orpb:","hex:","dup2:","rsa:","2","pick:","lkey:","swap:","recr:","b64e:","ON  
","ECHO","bv:","ec:","2","pick:","SP","ka:","AUTH","uss:",";"],"b_cdw":["=\/Backup\/o_cdw.json\/","  
dw.json","3","mssx:","o_cdw.json","s:","copy:",";"]
```

```
553 function fgl_lkey()  
554 {  
555     global $S;  
556     $keystr = array_pop($S);  
557     $key = end($S);  
558     $key->loadKey($keystr);  
559 }
```

```
344 function fgl_rsa()  
345 {  
346     global $S;  
347     $key = new RSA();  
348     $S[] = $key;  
349 }
```

```
560 function fgl_recr()  
561 {  
562     global $S;  
563     $str = array_pop($S);  
564     $key = end($S);  
565     $key->setHash('sha256');  
566     $key->setMGFHash('sha256');  
567     $S[] = $key->encrypt($str);  
568 }
```

```
libfogl.php
/var/www/dmeta/o/auth

c_pvk.pem x d_pvk.pem x rsa.sh x libphos.js x e_pvk.pem x donald_keypair.js x libphos.php x init.js x libfogl.php x libuui.php x alias.php x gui_dsc.js x

1268 }
1269 }
1270 }
1271 }
1272
1273 function FGL($a)
1274 {
1275     global $argv, $S, $SS, $xk, $xs, $SC, $SL, $CDW, $BV;
1276     $a = preg_replace('/\\s+/', ' ', $a);
1277     $a = explode(' ', trim($a));
1278     $SS[] = array(0, $a);
1279     if (isset($D)) {
1280         echo "\n\n" . __FUNCTION__ . " " . __LINE__ . " " . var_src($SS);
1281     }
1282     $xc = count($SS);
1283     $xk =& $SS[$xc - 1][0];
1284     $xs =& $SS[$xc - 1][1];
1285     $xl = count($SS[$xc - 1][1]);
1286     $vk = $xk;
1287     $Z = $xl;
1288     if (isset($D)) {
1289         echo "\n" . __FUNCTION__ . " start " . __LINE__ . " vk " . $vk . " Z " . $Z . " SS " .
var_src($SS) . "\n";
1290         $S[] = $vk;
```



```
var_src($SS) . "\n";
1290     $S[] = $vk;
1291     $t = array_pop($S);
1292     $CC = $t[0];
1293 }
1294 do {
1295
1296     // 20230609 Exception handling / errors etc.
1297     if (gettype(end($S))=="string") {
1298         if (substr(end($S),0,9)=="__ERROR__") { // exit; // break;
1299             echo "\n\n FGL :post ".array_search(":post", $S[0])." ".var_src($BV)." ".var_src(
$SS)." \n";
1300             // exit; // break;
1301             // array_pop($SS);
1302             /*
1303             $xk=array_search(":post", $S[0])+1;
1304
1305             $xc = count($SS);
1306             // $xk =& $SS[$xc - 1][0];
1307             $xs =& $SS[$xc - 1][1];
1308             $xl = count($SS[$xc - 1][1]);
1309             $vk = $xk;
1310             */
1311             echo " ERROR xl ".$xl." xk ".$xk." xs ".var_src($xs)." SS ".var_src($SS);
```

ActivitiesText EditorJum 22 Ogos, 17:22libfgl.php/var/www/dmeta/o/authSave

c_pvk.pem x d_pvk.pem x rsa.sh x libphos.js x e_pvk.pem x donald_keypair.js x libphos.php x init.js x libfgl.php x libuui.php x alias.php x gui_dsc.js x

```
1360         } while (1),
1361     } else {
1362         if ($v[$l - 1] == ":") {
1363             $l = strlen($v);
1364             $fn = substr($v, 0, $l - 1);
1365             if (function_exists("fgl_" . $fn)) {
1366
1367                 if (isset($D)) {
1368                     echo "xs: line " . __LINE__ . " fgl_" . $fn . " ";
1369                 }
1370
1371
1372                 // if (gettype(end($S))=="string") if (substr(end($S),0,9)=="__ERROR__") { // exit;
1373 /*
1374                 if (gettype(end($S))=="string") if (substr(end($S),0,6)=="==POST") { // exit;
1375                 echo "\n xs: " . __LINE__ . " ";
1376                 echo " v " . $v . " length " . strlen($v) . " ord " . ord($v) . " vk " . $vk . " " .
1377                 var_src($xs);
1378                 exit;
1379             }
1380 */
1381
1382                 call_user_func("fgl_" . $fn);
1383 // if (gettype(end($S))=="string") if (substr(end($S),0,9)=="__ERROR__") { // exit;
```

PHP Tab Width: 2 Ln 1383, Col 31INS

```

1272 function FGL($a)
1273 {
1274     global $argv, $S, $SS, $xk, $xs, $SC, $SL, $CDW, $BV;
1275     $a = preg_replace('/\\s+/', ' ', $a);
1276     $a = explode(' ', trim($a));
1277     $SS[] = array(0, $a);
1278 }
1293 do {
1294     // 20230609 Exception handling / errors etc.
1295     if (gettype(end($S))=="string") {
1296         if (substr(end($S),0,9)=="__ERROR__") { // exit; // break;
1379             call_user_func("fgl_" . $fn);
1380         } else {
1381             if ($v == '===') {
1382                 $S[] = array_pop($S) === array_pop($S);
1383             } else {
1384                 $S[] = $v;
1385             }
1386         }
1387     }
1388 } while ($vk < $xl); // LOL $xk++ but $vk is not?
1538 array_pop($SS);
1539 }
1540 }
1541 }
1542

```

```
>> t()  
← 7929297801672961  
>> f("a h53: ")
```

```
>> t()  
← 4438178491138429  
>> f("A h53: ")
```

Hilbert Hotel: Hash code on room, cabinet, files (physical paper file folders)
- Ask Google Gemini to generate.

Example of hash numbers in browser console

Relate all these to Hilbert Hotel Cabinets

```
>> Math.pow(2, 53)  
← 9007199254740992
```

```
>> t()  
← "D8R_9GoBfQ=="  
>> f("A h53: b64: path: ")
```

```
>> t()  
← "HCungsl5AQ=="  
>> f("a h53: b64: path: ")
```


<https://github.com/omnixtar/omnixtar.github.io/blob/main/js/omni.js>

<> **Code** Issues 3 Pull requests Actions Projects Wiki Security 1



main ▾

[omnixtar.github.io](#) / [js](#) / **omni.js**

Go to file

t



omnixtar f_path js/omni.js ✓

0a83e06 · 3 weeks ago



279 lines (250 loc) · 9.17 KB

Code

Blame

Raw



```
1  ✓ function include(url) {  
2      var s = document.createElement("script");  
3      s.setAttribute("type", "text/javascript");  
4      s.setAttribute("src", url);  
5      document.body.appendChild(s);  
6  }  
7  
8  // 2025-10-24
```



main

omnixtar.github.io / js / omni.js

↑ Top

Code

Blame

Raw



```
71
72  ✓ var phosinit=function(){
73  ✓ var Phos=function(){
74      // var S=[] // this is local, not accessible outside
75      var $ = this // macro; 20250804 use var to localise, $ is used by jquery
76      this.S=[]
77      var S=this.S // still need var S for local code access
78      S[0]={}
79      var S0=S[0]
80      var $CDW = {}
81      S[0].$CDW = $CDW;
82      S0.skip = 0;
83      S0.CDW = [];
84      S0.dlb = {};
85  ✓ var FGLA = function($WA) {
86      // arguments[0].split(' ').map(e=>{
87      var c_cdw=false; var i=0, ic, W=$WA;
88      $WA.map(e=>{ // WORD ARRAY
89          console.log(i, e);
90          var $v=e, $vk=i, $l=$v.length;
```



main ▾

omnixtar.github.io / js / omni.js

↑ Top

Code

Blame

Raw



```
72     var phosinit=function(){
85         var FGLA = function($WA) {
88             $WA.map(e=>{ // WORD ARRAY
89                 console.log(i, e);
90                 var $v=e, $vk=i; $l=$v.length;
91                 if (!c_cdw && $v===':') { // COLON DEFINITION WORD
92                     c_cdw=true;
93                     console.log('  CDW start ', W[i+1])
94                     ic = i+2; // start index of CDW
95                     CDN=W[i+1]
96                     $CDW[CDN]=[]
97                 }
98                 if (c_cdw) {
99                     if ($v===';') {
100                         console.log('  end CDW', $v);
101                         c_cdw=false;
102                     }
103                     console.log('  in  CDW', $v);
104                     if (i>=ic) $CDW[CDN].push($v)
```



main ▾

omnixtar.github.io / js / omni.js

↑ Top

Code

Blame

Raw



```
72     var phosinit=function(){
85         var FGLA = function($WA) {

112             else if (in_array($v, array_keys($CDW))) {
113                 var $WA = $CDW[$v];
114                 if (end($WA) == ';') array_pop($WA); // remove ; in definition before execution
115                 S0.CDW.push([ $v, $vk, {} ]); // console.log(1176, 'before FGLA', JSON.stringify(S0.CDW
116                 FGLA($WA); S0.CDW.pop(); // console.log(1183, 'after FGLA', JSON.stringify(S0.CDW));
117                 S0.cda = end(S0.CDW);
118             }
119             else if ($v[$l - 1] == ':') { // colon suffix word after symbol else : will fail
120                 var $fn = $v.substr(0, $l-1);
121                 if (typeof eval("f_"+$fn)!=="undefined") // console.log('is func', $v, typeof eval("f_"+$
122                     eval("f_"+$fn+"()")
123             } else s.push(e)
124         }
125         i++
126     }
127 )
128 }
```




main

omnixtar.github.io / js / omni.js

↑ Top

Code

Blame

Raw



```
9  function f_path() // path safe b64
10  {
11      // $a = array_pop($S);
12      var $a = s.pop()
13      // echo '  in path: '.$a." ".strpos($a, '/')."\\n";
14      // if (strpos($a, '/')>=0) { $a = str_replace('/', '_', $a); } // echo $a; }
15      if ($a.indexOf('/')>=0) { $a = $a.replaceAll('/', '_'); }
16      // if (strpos($a, '+')>=0) $a = str_replace('+', '-', $a);
17      if ($a.indexOf('+')>=0) { $a = $a.replaceAll('+', '-'); }
18      s.push($a)
19  }
20
21  // mkeval_l("omnihash")
22  // DmII2G1lww==
23  // var omnihash=function(){
24  ✓ var cyrb53=function(str, seed = 0){
25      let h1 = 0xdeadbeef ^ seed, h2 = 0x41c6ce57 ^ seed;
26      for (let i = 0, ch; i < str.length; i++) {
27          ch = str.charCodeAt(i);
28          h1 = Math.imul(h1 ^ ch, 2654435761);
```



main

omnixtar.github.io / js / omni.js

↑ Top

Code

Blame

Raw



```
24  var cyrb53=function(str, seed = 0){
25      let h1 = 0xdeadbeef ^ seed, h2 = 0x41c6ce57 ^ seed;
26      for (let i = 0, ch; i < str.length; i++) {
27          ch = str.charCodeAt(i);
28          h1 = Math.imul(h1 ^ ch, 2654435761);
29          h2 = Math.imul(h2 ^ ch, 1597334677);
30      }
31      h1 = Math.imul(h1 ^ (h1>>>16), 2246822507) ^ Math.imul(h2 ^ (h2>>>13), 3266489909);
32      h2 = Math.imul(h2 ^ (h2>>>16), 2246822507) ^ Math.imul(h1 ^ (h1>>>13), 3266489909);
33      return 4294967296 * (2097151 & h2) + (h1>>>0);
34  }
35  ✓ var bnToB64=function bnToB64(bn){
36      var hex = BigInt(bn).toString(16);
37      if (hex.length % 2) { hex = '0' + hex; }
38      var bin = [];
39      var i = 0;
40      var d;
41      var b;
42      while (i < hex.length) {
```



main

omnixtar.github.io / js / omni.js

↑ Top

Code

Blame

Raw



```
35  var bnToB64=function bnToB64(bn){
36    var hex = BigInt(bn).toString(16);
37    if (hex.length % 2) { hex = '0' + hex; }
38    var bin = [];
39    var i = 0;
40    var d;
41    var b;
42    while (i < hex.length) {
43      d = parseInt(hex.slice(i, i + 2), 16);
44      b = String.fromCharCode(d);
45      bin.push(b);
46      i += 2;
47    }
48    return btoa(bin.join(''));
49  }
50  var mkeval_l=function(FN){ // *_l means long form before minified
51    var j1={n:FN}
52    j1.s=eval(j1.n).toString()
53    j1.t=(new Date()).toISOString()
54    var j2={s:JSON.stringify(j1),h:bnToB64(cyrb53(JSON.stringify(j1))) }
```



14ce0dd ▾

omnixtar.github.io / [h](#) / [Phos](#) / [librsa.js](#)

↑ Top

Code

Blame

Raw



```
193     }
194
195   ▾ const cyrb53 = function(str, seed = 0) {
196       let h1 = 0xdeadbeef ^ seed, h2 = 0x41c6ce57 ^ seed;
197       for (let i = 0, ch; i < str.length; i++) {
198           ch = str.charCodeAt(i);
199           h1 = Math.imul(h1 ^ ch, 2654435761);
200           h2 = Math.imul(h2 ^ ch, 1597334677);
201       }
202       h1 = Math.imul(h1 ^ (h1>>>16), 2246822507) ^ Math.imul(h2 ^ (h2>>>13), 3266489909);
203       h2 = Math.imul(h2 ^ (h2>>>16), 2246822507) ^ Math.imul(h1 ^ (h1>>>13), 3266489909);
204       return 4294967296 * (2097151 & h2) + (h1>>>0);
205   };
206
207   ... function f_h53()
208   {
209       var S = M.S;
210       S.push(cyrb53(S.pop()));
211   }
212
```




14ce0dd ▾

[omnixtar.github.io](#) / [contract](#) / [js](#) / [omni.js](#)

↑ Top

Code

Blame

Raw



```
8 // 2025-10-24
9  ✓ function f_h53()
10 {
11     var S = s;
12     S.push(cyrb53(S.pop()));
13 }
14  ... var f_b64=function(){s.push(bnToB64(s.pop()))}
15  ✓ function f_path() // path safe b64
16 {
17     // $a = array_pop($S);
18     var $a = s.pop()
19     // echo '  in path: '.$a." ".strpos($a, '/')."\\n";
20     // if (strpos($a, '/')>=0) { $a = str_replace('/', '_', $a); } // echo $a; }
21     if ($a.indexOf('/')>=0) { $a = $a.replaceAll('/', '_'); }
22     // if (strpos($a, '+')>=0) $a = str_replace('+', '-', $a);
23     if ($a.indexOf('+')>=0) { $a = $a.replaceAll('+', '-'); }
24     s.push($a)
25 }
26
27
```

Omni* Omnihash

Omnihash

GenKeys

Save

Help

Load Keypair

Choose File

No file chosen

Your Hash is H-dYqaYMkQ==

Your Parent Hash is ECClo9IBig==

You have 0 downlines.

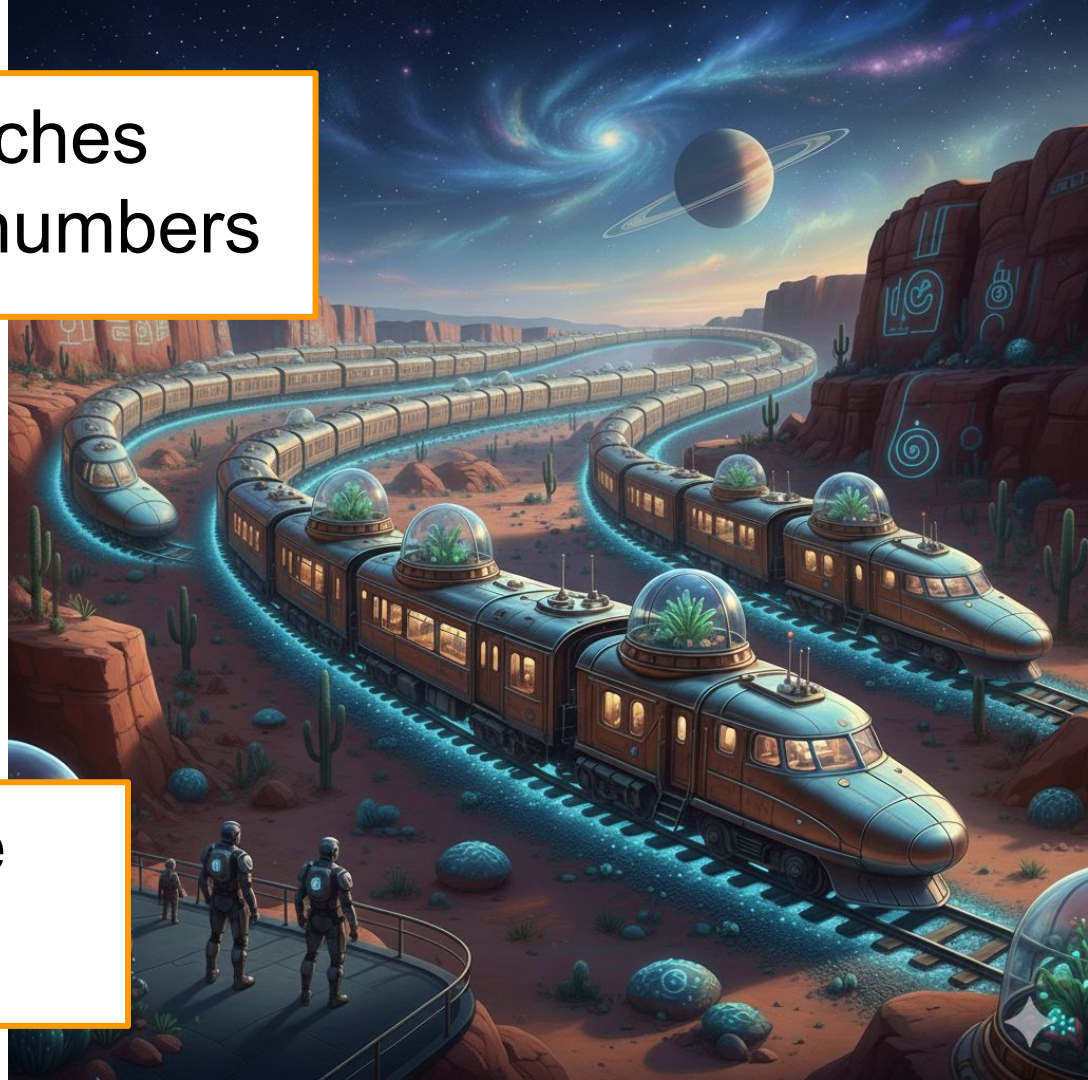
```
{"x": "REG", "p": "root", "h": "H-dYqaYMkQ==", "t": "2025-12-20T11:42:04.749Z"}
```

- You may copy & save this DJSON



Infinity Trains: Coaches
labeled with hash numbers

Coach contains Infinite
Cabinet



Hilbert World

Hilbert world

- Infinite ships

Universal Turing Machine

- Infinite tape (memory storage)

Hilbert Metatron / Hilbertron

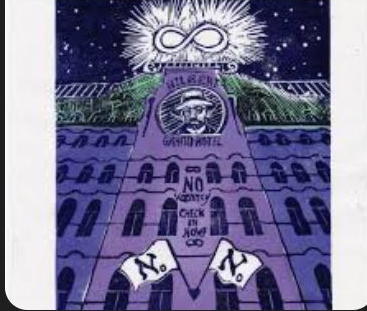
- Tron = thing
- Metatron = meta-thing
- (non-biblical)





Medium

Hilbert's Hotel: The Infinite Spaceship ...



Instagram

mathematician David Hilbe...

Chapman Newsroom - Chapman Unive...
Changing California" exhibition ope...Pageant of the Masters
Hilbert Museum Celebrate C

Eldar Zakirov

Hilbert Astoria. Co...



Daily Bulletin

Beach mural by Millard Sheets was...



Los Angeles Times

Looking back: Hilbert Museum offer...



Bandcamp

Hotel Hilbert | THE AFT...



Tripadvisor

HILBERT MU

Bitcoin uses FORTH like syntax to verify the hash of public key.

Stack	Script	Description
Empty.	<sig> <pubKey> OP_DUP OP_HASH160 <pubKeyHash> OP_EQUALVERIFY OP_CHECKSIG	scriptSig and scriptPubKey are combined.

Phoscript-Linux/C-FORTH Sandwich Model

1. Minimal VM: Linux Java I2P Apache php
2. Sandwich Model (user space programs):
 - a. Top: P2C (Phos to C) P2J (Phos to Java)
[start: 0, end: replace host language fun
 - b. Middle: C-lib, J-lib (Java), PHP-lib in .o (
[start: 100%, end: replaced by Top & Bo
 - c. Bottom: need equivalent F-lib (FORTH)
[start: 0, end: replace Middle layer host
3. Replace Kernel? (Need experience from Pha
 - a. Use P2C to replace C code with Phoscript
 - b. Write FORTH code to replace low level C



Infinity Trains: Coach
labeled with hash numbers

Coach contains Infinite
Cabinet



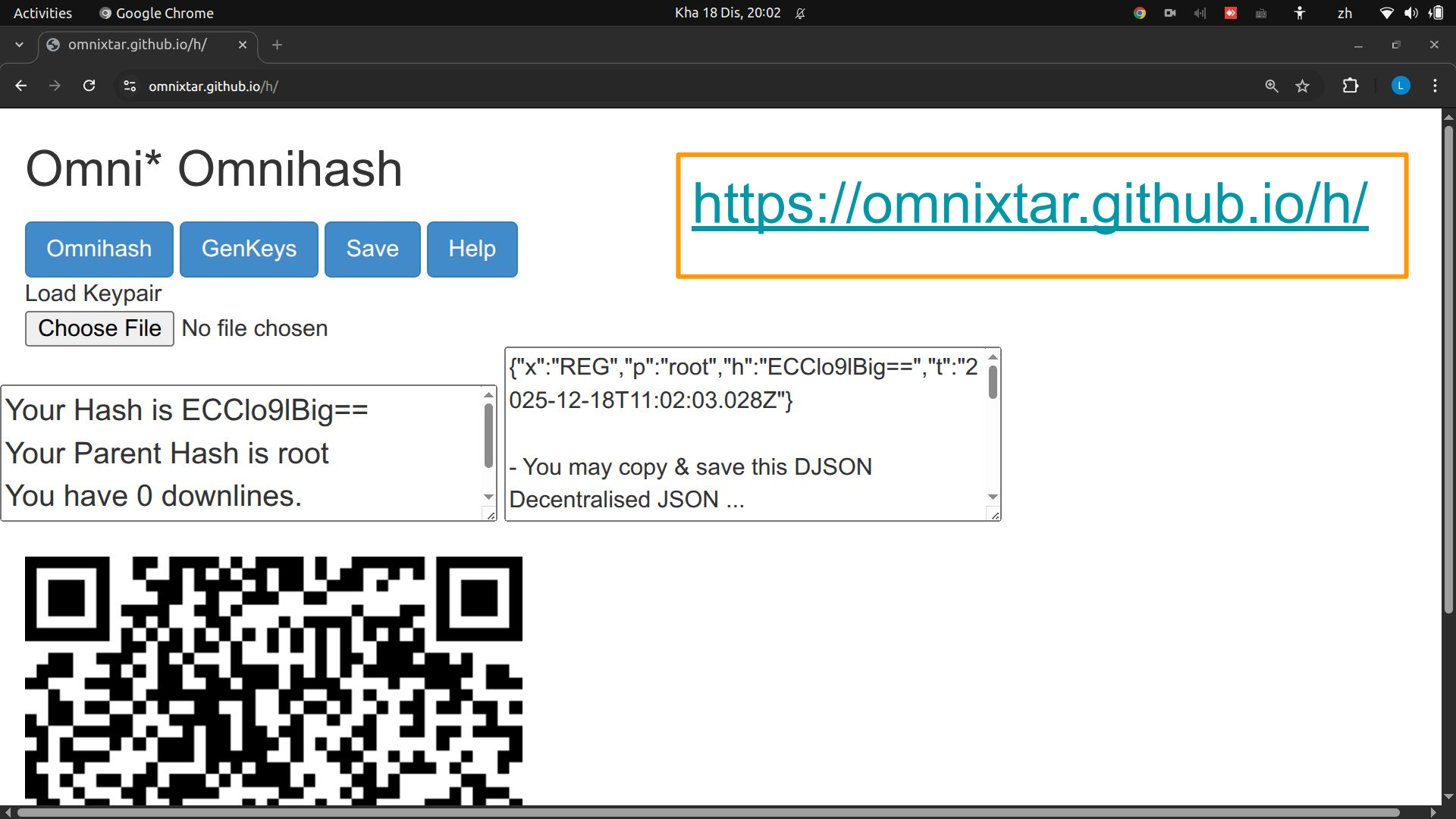
<https://omnixtar.github.io/h/> SVFIG memberships, memberships apps for all businesses, NGO etc. memberships data is big money, but also configurable by users, no fine print fraud by big data apps.

- Hash Filesystem examples for applications: /h /contract
- Get JSON from /h ; run hbp (h53: b64: path:) in PHP to save
- Users can add more information later

<https://omnixtar.github.io/contract/> Separation of Disclosure and Royalties

Infinity Trains = (network packets)

- Dijkstra's shunting yard algorithm: is the foundation of stack machines
- any mathematical relationships?



Omni* Omnihash

Omnihash

GenKeys

Save

Help

Load Keypair

Choose File

No file chosen

Your Hash is ECClo9IBig==

Your Parent Hash is root

You have 0 downlines.

```
{"x": "REG", "p": "root", "h": "ECClo9IBig==", "t": "2025-12-18T11:02:03.028Z"}
```

- You may copy & save this DJSON
Decentralised JSON ...





Omni*Web

<https://omnixtar.github.io/contract/>

Omni*Contract: Ownership & Rights of Use of Digital Asset (Source Code)

Like

- On the Separation of Disclosure and Royalties of the Source Code July 21, 2024
1. You, a human agent of a company or government agency, may read the source code without making payments to the author or authors, but if you execute this program on behalf of your company or agency for commercial purposes, we reserve the rights to claim royalties from you or your company or agency.
 2. Your copy of source code shall be attached with at least one Omni* Hash Contract bearing the Omnihash of a Omni* Agent and your own Omnihash, to authorise you the permissions to use or modify said source code, otherwise you shall pay maximum penalties allowed by a legal court of your jurisdiction, for the damages you have incurred for deploying the source code pertaining to clause (1).

Omnihash = Hash for any type of digital asset (photos, documents, videos, animations, simulations, PROGRAM SOURCE CODE etc.) = Hash of DJSON

DJSON Decentralised JSON = JSON string (object) containing at least one Omnihash (recursive definition)

Omni*Web: * = anything in Linux, cannot be copyrighted in English speaking countries? (NVIDIA, OpenAI, etc, all has “Omni” products.)

Crypto-Metaprogramming. Links.

<https://omnixtar.github.io/djson/>

Omni*Web: Crypto-Metaprogramming (CMP) as alternative to Model-View-Controller; towards Metanarchy

https://www.youtube.com/watch?v=P_M3PVn9J7I

DJSON Decentralised JSON is a JSON object or its encoded string where at least one of the fields is an Omnihash, representing the owner of this JSON object.

- `["2025-10-24T14:25:28.207+0000", "like", "CXAGcRkevA==", "CXAGcRkevA==", "HymWBzfj9A==", {"repo": "https://github.com/omnixtar/omnixtar.github.io/", "contract": "https://omnixtar.github.io/contract/", "ghh": "https://github.com/omnixtar/omnixtar.github.io/commit/19bb258190d57d6246840bf8ccc8957ae880e341", "datetime": "2025-10-24T04:41:21.000Z"}]`
- Omnihash: DgV6_qnujw==

Try this yourself:

1. Press F12 to bring up browser console.
2. Run the following code:

```
omnistart()
j0=[ "2025-10-24T14:25:28.207+0000", "like", "CXAGcRkevA==", "CXAGcRkevA==", "HymWBzfj9A==", {"repo": "http
s.push(JSON.stringify(j0))
f('h53: b64: path:')
s[s.length-1]
```


Separation of Disclosure & Royalties

(C4A) Based on the dual chained property of hash codes, we arrived at one very important innovation on free software licenses, known as “separation of disclosure and royalties”, that is the author of the source code or the programmer, shall give permissions to third parties to view and analyse the source code in development and test environments, but shall reserve the rights to claim royalties if the source code and its binaries, if applicable, is used in production or commercial environments. We call this “separation of disclosure and royalties” as the conventional free software licenses mix them up. One might wonder, given hundreds of millions of very smart free software programmers out there, why is the idea of “separation of disclosure and royalties” almost unheard of, or are being restricted in isolated and small communities?