

NOSTR 螺丝锤协议

NOSTR是个一劳永逸创建抗审查的全球“社交”
网络的最简化开源协议

目录

- NOSTR是什么鬼?
- Web2互联网的问题
- Web3区块链的问题
- NOSTR的运行原理
- Web5=闪电网络+NOSTR
- NOSTR重新定义创作者经济
- NOSTR是下一代互联网的底层协议
- 答疑

NOSTR是什么鬼？

一句话总结：

- NOSTR是个一劳永逸创建抗审查的全球“社交”网络的最简化开源协议。 - fiatjaf, NOSTR开发者
- NOSTR是一个有能力取代推特、电报和其他东西的协议。 - dergigi
- NOSTR是一个有能力让互联网巨头和加密货币市值归零的协议。 - aLE

一句话总结“拿特资产特供版”：

- NOSTR让过去的互联网估值体系失灵，并能重建一套新的互联网估值体系。 - aLE

WEB1

THE INFORMATION ECONOMY



WEB2

THE PLATFORM ECONOMY



WEB3

THE OWNERSHIP ECONOMY



Web2互联网的问题

Web2的逻辑:

- UGC是Web2的全部
- 广告是Web2的主要营收模式

但这个逻辑是有明显问题的

Web2互联网的问题

Web2声称它是免费的。

但中心化服务器（和云服务器）的运营成本，最终需要用户承担。

NOSTR fixes this

Web2互联网的问题

注册门槛是阻碍部分人使用Web2的最大阻力，比如：

- 繁琐的注册机制“邮箱/手机验证”加“账号密码”模式
- 每一个平台都要注册新的“账号密码”
- 苹果ID？对老年人非常不友好。

Web2互联网的问题

用户只有账号的使用权，没有所有权：

- 你的账号本质上属于公司
- 你的数据也本质上属于公司
- 公司可以随时出于任何理由删除你的数据、暂停使用，甚至封号
- 公司倒闭了你的数据就没有了
- 中心化的服务器存在单点故障，你的数据“从技术上”并不安全
- 垄断寻租

Web2互联网的问题

Web2的逻辑问题，所造成的创作者激励问题：

- 内容创作者本质上是在与Web2签订一种不对等的奴役制度，实质上、技术上并不拥有UGC的知识产权，也不参与广告营收分配
- Web2通常不允许直接的、点对点的经济激励，而是采用一种迂回的方式。举个例子：硬币
- Web2经济激励存在广泛的“抽成”、“按月提现”现象
- 由于缺乏直接的经济激励，据我观察，大部分内容创作者的营收，很大程度上来自于他“周边产品”的销售，比如马克杯、T恤、鸭舌帽

Web2互联网的问题

自媒体在Web2叙事下本质上是个伪命题 - aLE

NOSTR fixes this

Web2互联网的问题

Web2的逻辑问题，所造成的广告和垃圾信息问题：

- 广告代言也是内容创作者的主要营收，但，Web2用户讨厌广告
- 因为平台也需要广告营收，所以Web2充斥着广告
- Web2利用大数据让你上瘾，来增加它的广告营收
- 由于Web2是免费使用的，所以它并不抗女巫攻击，它无法解决垃圾信息问题

Web2互联网的问题

Web2逻辑的终极版本：



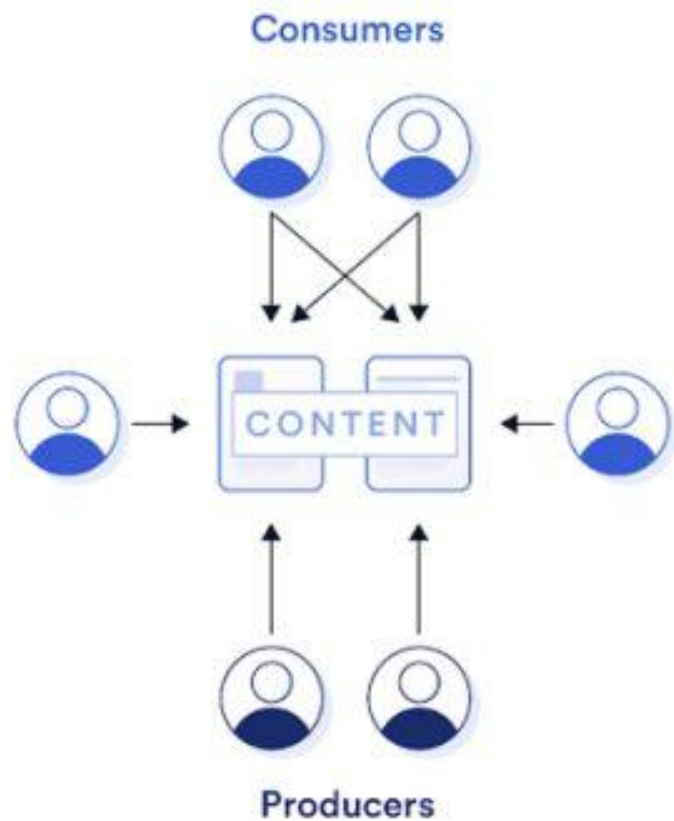
图源：《2022，我们在幽暗的谷底守护一朵花》史中，浅黑科技

Web2互联网的问题

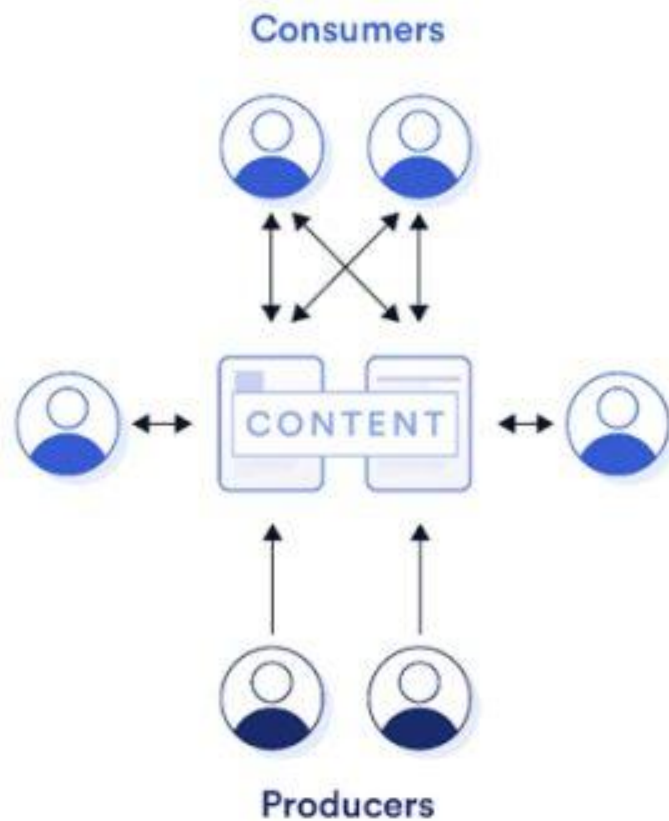
不论在情感上、科技上、还是产权上，我们也并不真正意义拥有不被侵犯的私人空间。——《关于「互联网讨论消亡」的进一步思考》陈飞樾，沙丘实验室

NOSTR fixes this

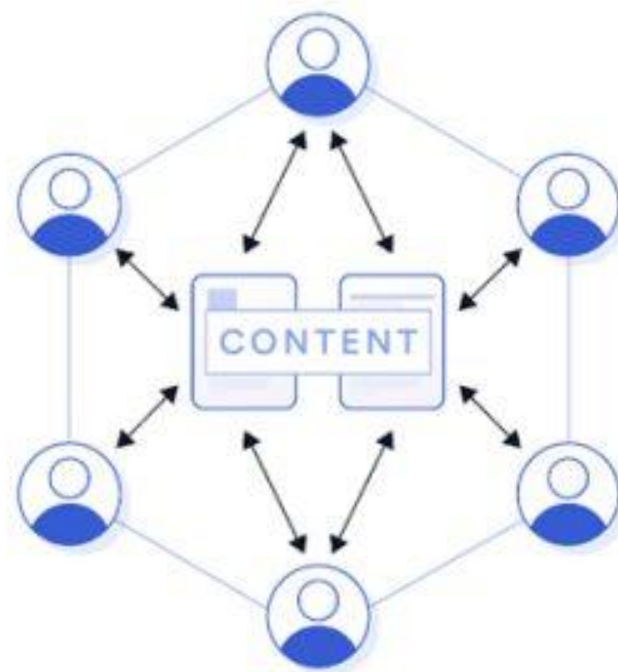
Web 1.0



Web 2.0



Web3



Web3互联网的问题

Web3的解决方案和基本逻辑：

- 去中心化的服务器（基于区块链技术）
- 去中心化身份系统（基于公钥密码学）
- 将一切信息金融化（ICO、NFT）
- Web3=区块链+智能合约

这个逻辑比Web2的问题更大

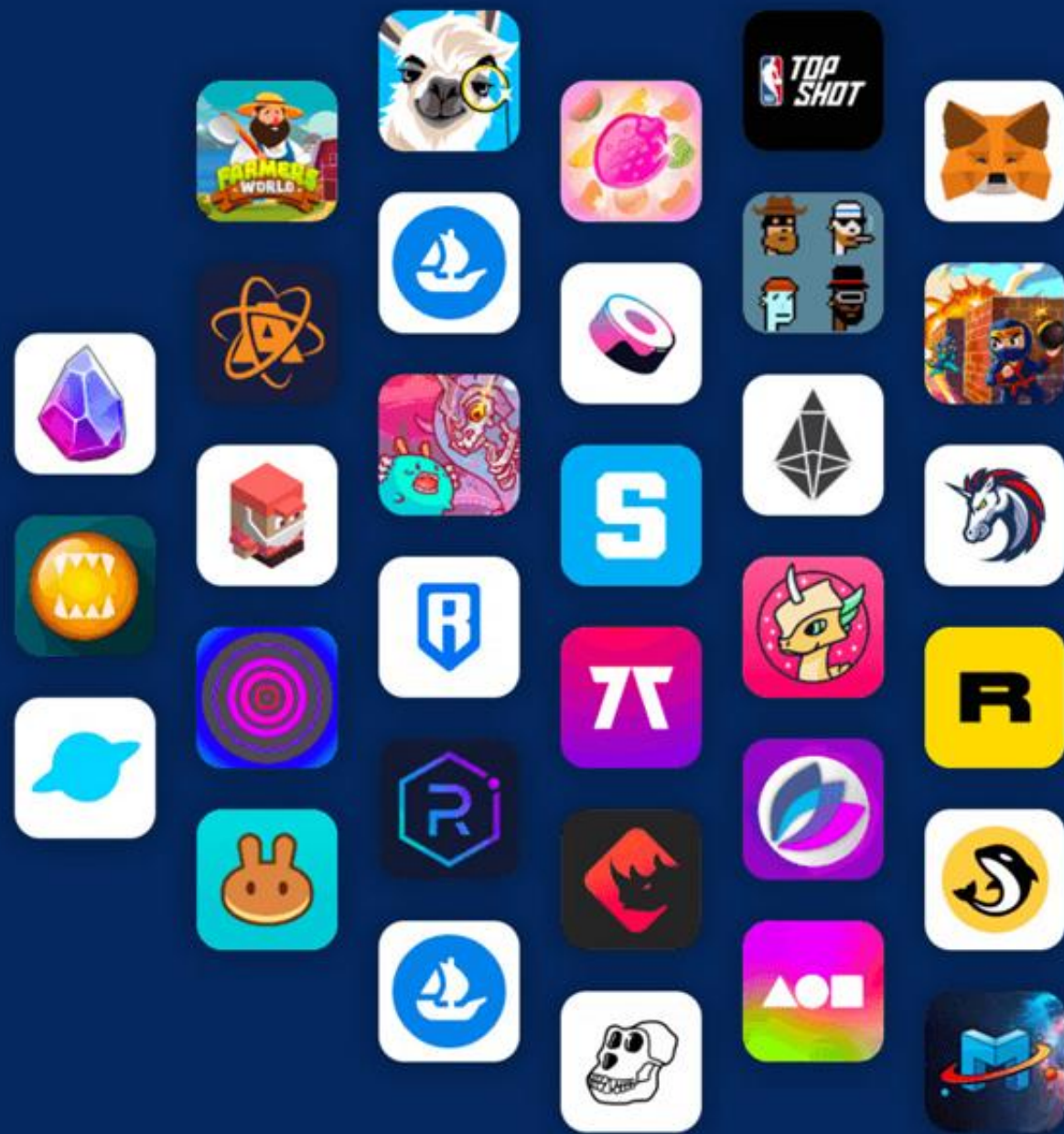
Web3互联网的问题

Web3的愿景

- 数据去中心化（借助区块链技术）
- 数据主权回归用户（借助公钥密码学）
- 去中心化的组织模式/反公司制（借助DAO和代币经济学）

看起来很理想，但其实问题非常大

DAPP



Web3互联网的问题

区块链的问题（为什么Dapp没人玩？）：

- 速度：区块链速度很慢（打包速度）
- 性能：区块链性能很低（容量、拥堵、高并发）
- 费用：区块链费用很贵（一拥堵就更贵）
- 门槛：区块链门槛很高（比Web2高）
- 上链：无法处理垃圾信息
- 同步：区块链的数据膨胀问题，以及谁来承担？
- 节点：违背“去准入门槛”原则，不可避免走向中心化
- 跨链：跨链的“稳定性”和“安全性”都非常难保证

Decentralization

Security

Scalability





Blockstream



Lightning
40,000,000 tps



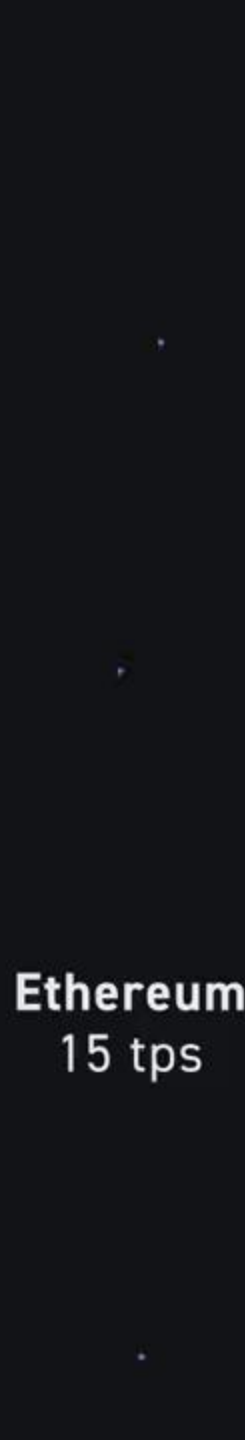
Solana
65,000 tps



Visa
24,000 tps



Paypal
193 tps



Ethereum
15 tps

NFT



DeFi



Web3互联网的问题

智能合约的问题：

- 虚拟资产所有权无法映射现实资产，必须借助中心化机构的力量（政府、法院、公证机关），虚假的去中心化叙事
- 智能合约的实际控制权问题，必须要用DAO以及发币来解决，但这种解决方式非常烂，本质上就是ICO，全体负责的真相就是无人负责。
- 信息一旦资产化（NFT），就不可能兼顾资产产品的不可能三角（收益性、安全性及流动性）
- 虚拟资产的交易促进DeFi的诞生，但DeFi不是金融创新，是项目方出货创新。DeFi的本质是金融乐高，造成系统性风险、金融危机。

Web3互联网的问题

总结:

- 由于过度金融化，导致整个币圈基本上是一个博傻的大赌场，只有赌狗和机器人在玩
- 区块链应用(Dapp)使用体验并不如Web2，真实用户少得可怜
- 区块链是一种很臃肿的系统，性能很低，无法承载“高并发”互联网应用（比如社交网络）
- 如果要强行搞互联网应用，区块链必然走向“提升性能”和牺牲“去中心化”的道路，甚至走向中心化，“安全性”和“抗审查性”严重受损

Web3互联网的问题

币圈是个法币本位大赌场 - aLE

NOSTR fixes this

INTRODUCING NOSTR

Image: @WalkerAmerica



NOSTR

Notes and Other
Stuff Transmitted
by Relays

NOSTR的运行原理

NOSTR的解释:

- Notes and Other Stuff Transmitted by Relays
- 由“中继器”传输的“笔记”和“其它东西”

NOSTR的运行原理

NOSTR的特点:

- 不依赖于任何可信任(trusted)的中心化服务器，因此具有“韧性/可迅速恢复性”(resilient)
- 基于公钥密码学的“密钥和签名”原理，所以它防篡改
- 不需要依赖P2P技术就能运行

NOSTR的运行原理

NOSTR简洁优雅：

- 客户端Client

可以用APP或者网站来类比，NOSTR“客户端”可以实现任何已知APP和网站的功能，也能实现目前的互联网(Web2)和区块链(Web3)无法实现的功能。

- 中继器Relay

可以用服务器来类比，但“中继器”是傻瓜服务器，它抓取、储存任何来自与它连接的“客户端”的信息，并转发给其它“客户端”；任何人都可以运行“中继器”。

NOSTR

A truly censorship-resistant alternative to Twitter that has a chance of working

一个真正抗审查的推特替代品

Nostr is a **Protocol**,
not an App nor a
platform

Nostr是一个“协议”，
不是一个App，也不是一个平台

Authentication

How can I create
an account on
Nostr

我怎么在Nostr上
创建账号

Whaaaaat !!

什么鬼？

Using your keys,

用你的密钥对

Public key = username

Private key = password

公钥 = 用户名

私钥 = 密码

Explain it like I'm 5

假设我跟个3岁小孩儿
一样跟我解释一下

Let's say I want to create a
post, so you can view it later

打个比方我要发个po，
然后你待会儿要看到
我发的po

NOSTR

I publish my post on relay 'A' that am connected to.
我把我的po发送给与我连接的中继器“A”

Relay 'A'

Post

发po

(把“中继器”想象成任何人都可以运行的服务器)
(Think of Relay as a server that anyone can run)

Any other user connected to this same relay will see my post

其它任何与中继器“A”连接的用户都能收到我的po

Relay 'A'

Post

Post

Post

Post

Post

If am connected to Relay 'A' and you're connected to Relay 'B'.

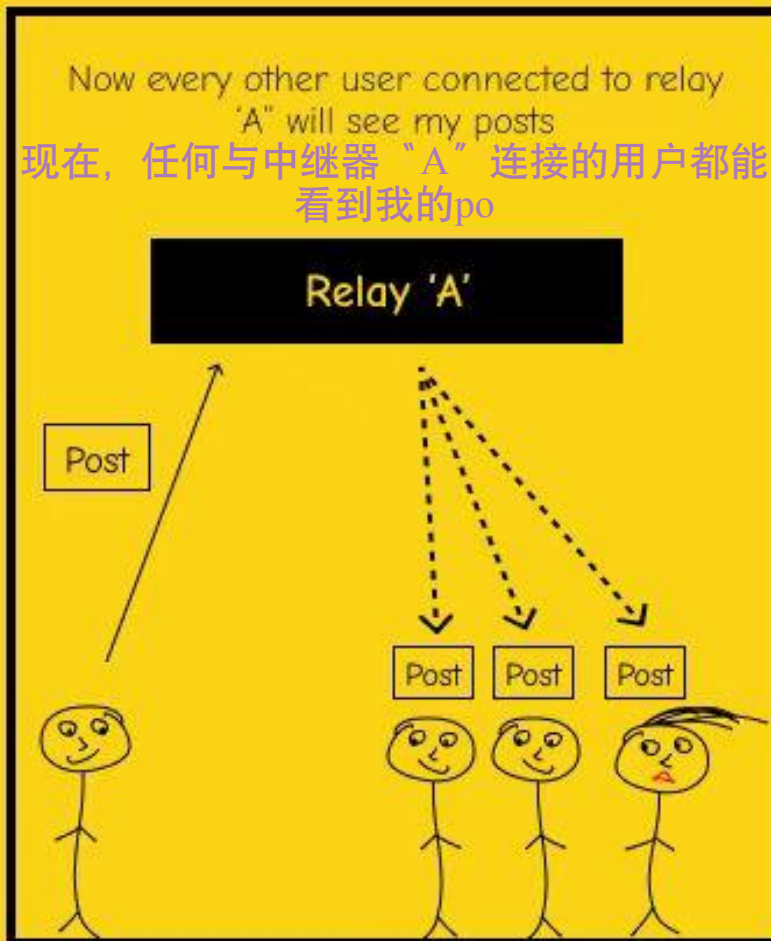
We will not be able to see each other posts
如果我连接了中继器“A”而你连接了中继器“B”

Relay 'A'

Relay 'B'

我们就看不到各自的po

NOSTR



Relays are dumb servers
中继器是傻瓜服务器

They just forward any message they get.

You need to connect your client to a relay for it to work.

There are many relays & you can run your own.

它转发任何发给它的信息
你要把中继器和客户端连接起来才能运行
中继器很多, 你可以运行你自己的中继器

NOSTR

Nostr Clients Nostr客户端

There are many client already
implemented on top of Nostr:

现在有很多建立在Nostr协议上的客户端



Twitter
Alternative
Nostr版推特



Telegram
Alternative
Nostr版电报



Games
Nostr游戏

Next Post

In the Next post we will see how events
(Posts) are **signed** before they get sent to
Relay

接下来介绍事件 (events, 即po) 如何
在它们被发送给中继器之前被“签名”

关注@coderjourney1

Will share a serie of posts related to
Nostr here.

If you're interested in learning more
make sure to **follow**

Post #1

How Nostr Works ?

A truly censorship-resistant alternative to Twitter that has a chance of working

一个真正抗审查的推特替代品

How creating a **Post**
works on Nostr ?

发Po是如何在Nostr上实现的？

告诉我如何在
Nostr上面发po

Tell me how
creating a Post
works on Nostr

把信息签名并发
送给多个中继器
by signing it and
sending in to multipl
relays

假设我跟个3岁小孩儿
一样跟我解释一下
Explain it to me
like am 5 yo

Creating A Post

Let's say you want to
publish this post,
假设我想要发布一个po文

Post 一个po文



第一步，你需要将其签名
You first have to sign it

为什么要签名？

Why signing it ?

So that Nobody can
speak for you or
impersonate you

这样就没人可以冒充你



A post (event) is signed using
your private key
签名一个po (事件event)，需要用到你的
私钥



已加密✓

我们会在之后讨论生成公钥和私钥的问题

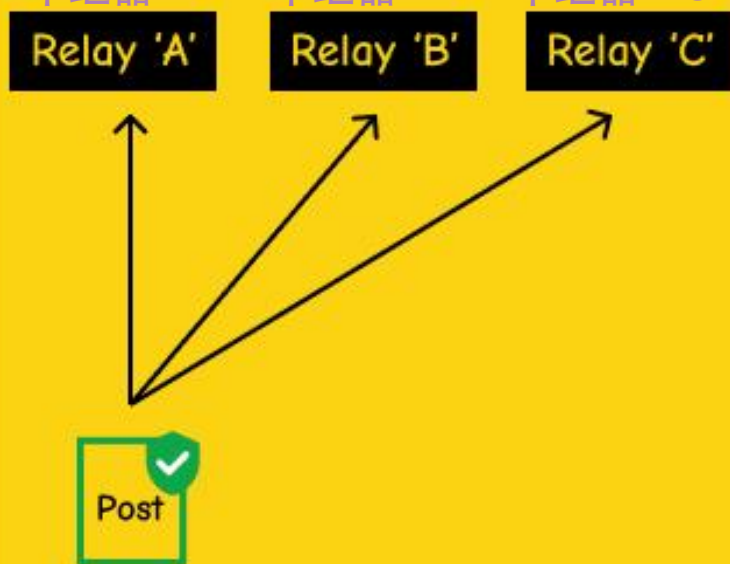
*Will talk about public/Privatekey generation in the next post

NOSTR

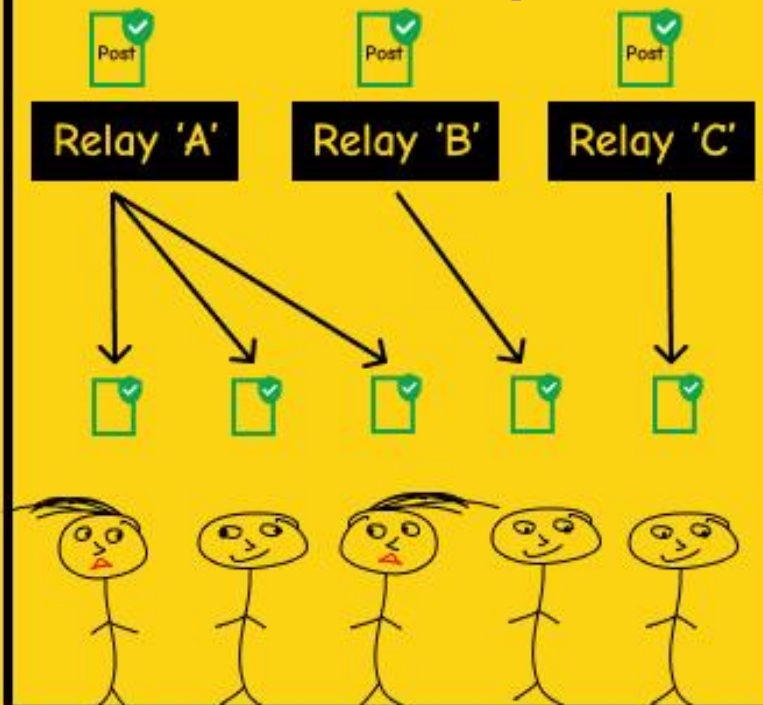
当你点“发布”，你的po会被发送给所有跟你连接的中继器

When you click “publish”, your post gets sent to all Relays you are connected to

中继器“A” 中继器“B” 中继器“C”

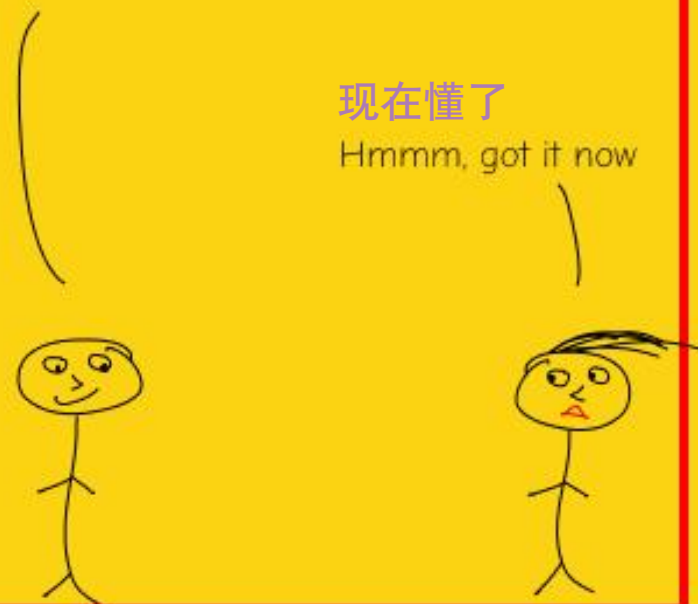


Any user **connected** to one of these Relays will receive your post
任何与这些中继器连接的用户都能收到你的po



如果我和你之间**没有连接相同的中继器**，你就看不到我的po

If me and you were **not connected** to the same relay, you won't see my post



NOSTR



Next Post

In the Next post we will see how creating an account using **public/private Key** works

待会儿我们讲，创建Nostr账号时，用到的公钥密码学

关注@coderjourney1

Will share a serie of posts related to Nostr here.

If you're interested in learning more make sure to **follow**

Post #2

NOSTR

A truly censorship-resistant alternative to Twitter that has a chance of working

一个真正抗审查的推特替代品

How to create an
account on **Nostr**?
怎么在Nostr上创建账号?

怎么在Nostr app上
创建账号
How can I create an account on Nostr app

Nostr不是一个app, 它是一个“协议”
Nostr is not an app, it's a **protocol**.

你可以生成密钥对, 然后连接任何建立在这个协议之上的app
you can generate keys and connect to any app built on top of it



Credentials

public key = username
private key = password

“公钥” = 用户名
“私钥” = 密码

Public/Private Key

公钥/私钥

To create a new key-pair, choose any
Nostr Client

你可以用任何Nostr客户端创
建新的密钥对

Nostr

Astral.
ninja

Damus

blockcore

Coracle

这些客户端都可以帮你创建密钥对
These clients above will create a key-pair
for you

Save the private key

保存你的私钥

save the private key somewhere,
and keep it **secret**.

把私钥保存在其它地方，不
要告诉别人

密钥对的两种形式

Keys Format

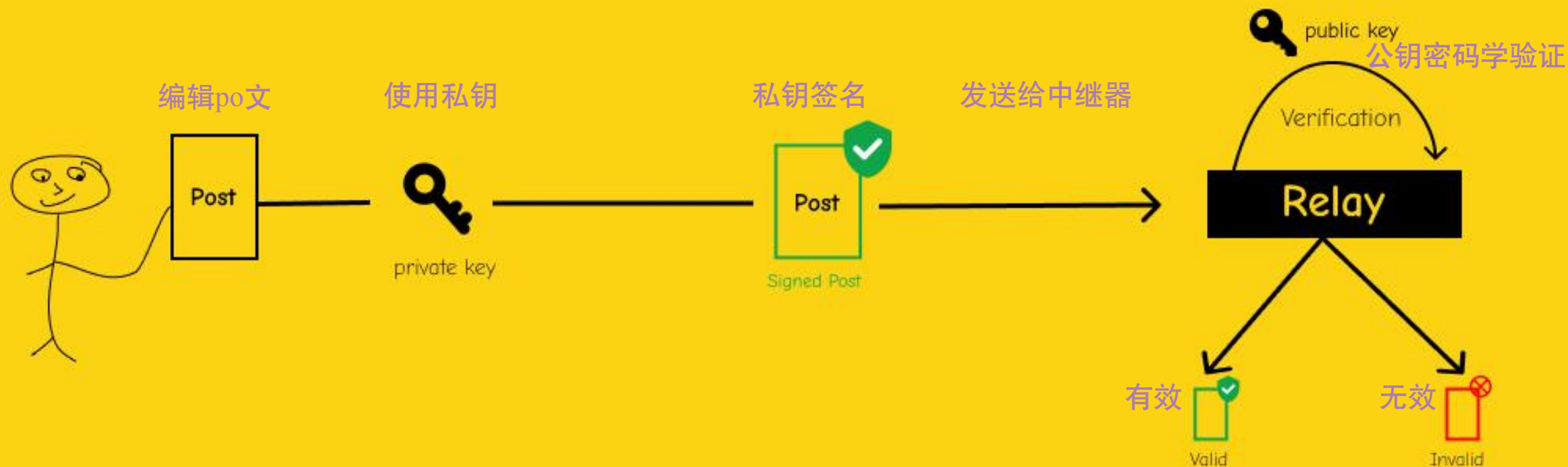


Hex
16进制

npub/nsec

NOSTR

Protocol flow 协议流程



NOSTR的运行原理

NOSTR运行原理：

- 你的ID就是你的公钥（你可以把它理解成你的用户名or账号），你发送任何信息都需要用私钥签名（把它理解成你的密码），签名由客户端验证。
- 客户端把“私钥签名的数据”传输给“你选择的”中继器（单个or多个），也从中继器中获取其它数据（来自其它客户端和用户）。

NOSTR的运行原理

NOSTR钥匙串和Web2账户的区别：

- 理论上你可以拥有无数钥匙串、你可以随时生成新的、你可以用完即弃。这意味着使用NOSTR不需要注册，这与需要“注册”和“KYC”的Web2有本质不同。
- 你可以用“A1by”等“钱包软件”生成和管理钥匙串。而非邮箱、手机。
- 你可以用闪电网络钱包实现对“A1by”的控制权，理论上只需要一个账户，就能在整个NOSTR网络冲浪。

NOSTR的运行原理

NOSTR中继器和中心化服务器的区别：

- 中继器不需要被信任，因为验证签名的是客户端
- 你可以随时丢弃中继器，因此它无法作恶
- 你也可以运行你自己的中继器（抗审查）
- 客户端只能与中继器通信，中继器只能与客户端通信
- 中继器与中继器之间互不通信同步（这一点很重要，本质上区分了区块链节点）

中继器是一种区别于“区块链节点”的“服务器去中心化解决方案”。

Web5=NOSTR+闪电网络



11,992
NODES

84,498
CHANNELS



ACINQ	3569 channel(s)
WalletOfSatoshi.com	2641 channel(s)
1ML.com node ALPHA	1723 channel(s)
CoinGate	1627 channel(s)
In.nicehash.com [Nicehash]	1369 channel(s)
Boltz	1230 channel(s)
Kraken ⚡	1190 channel(s)
bfx-lnd0	1109 channel(s)
deezy	1000 channel(s)
OpenNode.com	992 channel(s)
gameb_1	967 channel(s)
BCash_Is_Trash	914 channel(s)
bfx-lnd1	824 channel(s)
mainnet.lightningconductor....	799 channel(s)
tippin.me	719 channel(s)
southxchange.com	691 channel(s)
gameb_2	664 channel(s)
ando.masterofpearls.net	636 channel(s)
CryptoChill	594 channel(s)
LNBIG.com [lnd-01]	584 channel(s)
Bitrefill Routing	572 channel(s)
LNBIG.com [lnd-12]	539 channel(s)
Moon (paywithmoon.com)	507 channel(s)
DiamondHands 💎👋	485 channel(s)
Bitrefill	460 channel(s)
LNBIG.com [lnd-11]	450 channel(s)
Lightning.Watch	448 channel(s)
sphinxrouting-fceb9955a9	412 channel(s)
River Financial	392 channel(s)

Web5=闪电网络+ NOSTR

经济模型

- NOSTR的开发者fiatjaf同时也是比特币和闪电网络的开发者，因此NOSTR原生支持闪电网络（互联网应用+支付系统）
- 路由费是闪电网络节点的主要营收
- 比特币是NOSTR原生货币

Web5=闪电网络+ NOSTR

经济模型

- NOSTR对公司制友好，创业公司可以自建闪电网络节点，流量越大的节点，收入越高，从而摆脱对广告的依赖（但并不排斥）
- 越多人用NOSTR和闪电网络，比特币矿工手续费收入越高
- 以比特币为计价单位的商业模式会在NOSTR上成为主流，会加速超比特币化(hyperbitcoinization)。



Web5=闪电网络+ NOSTR

性能和安全性

- 闪电网络不是区块链技术，速度非常快，性能非常强，能够承载NOSTR上的高并发应用，一举重建整个互联网
- 闪电网络的安全性由比特币主网保证，比特币主网是目前最去中心化、最安全的区块链
- 闪电网络不会牺牲比特币主网的安全性和去中心化（突破不可能三角）

REACHABLE BITCOIN NODES

Updated: Wed Jan 11 03:50:11 2023 CST

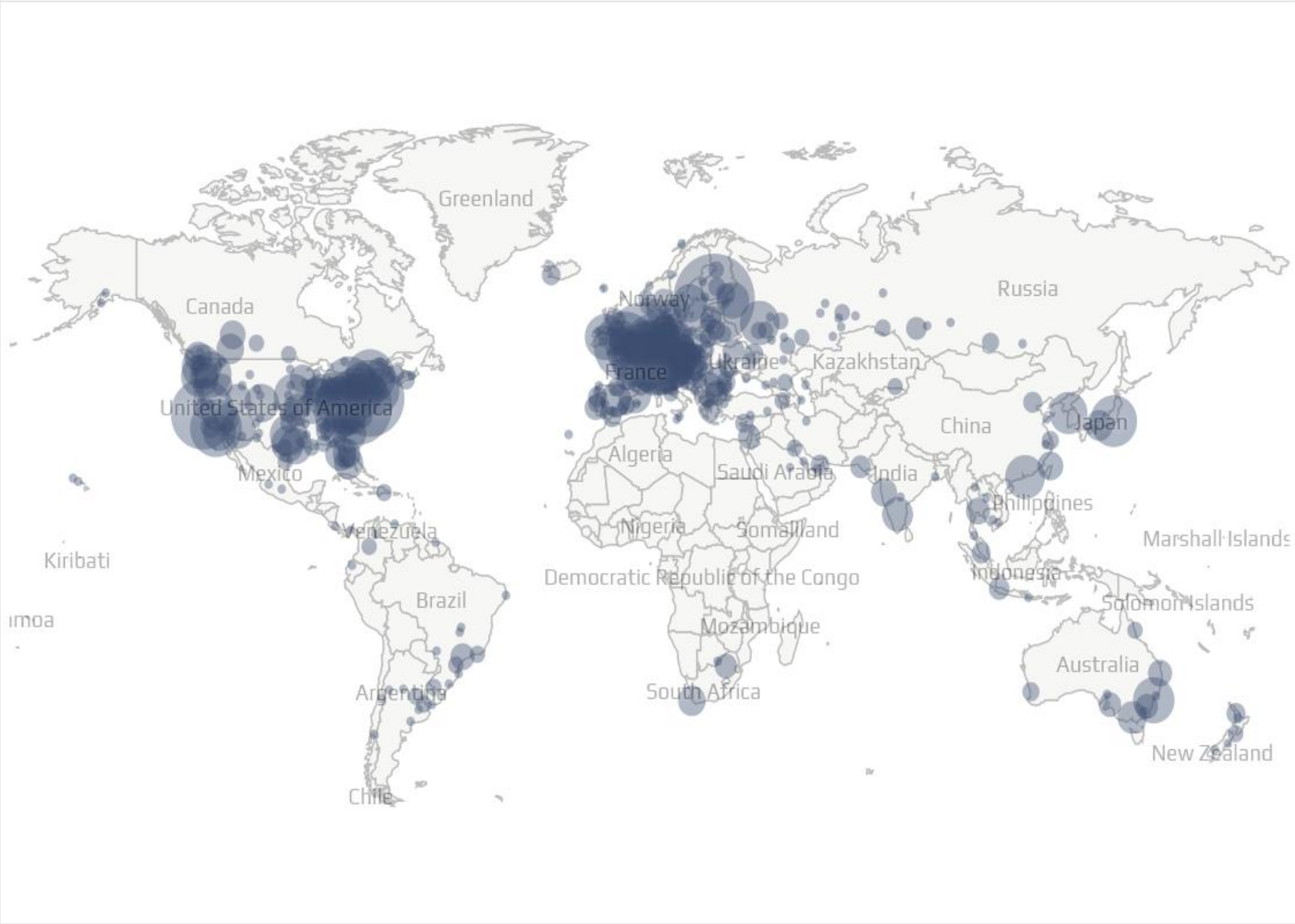
14017 NODES

CHARTS

IPv4: -6.0% / IPv6: -0.8% / .onion: -4.9%

Top 10 countries with their respective number of reachable nodes are as follows.

RANK	COUNTRY	NODES
1	n/a	7762 (55.38%)
2	United States	1636 (11.67%)
3	Germany	1325 (9.45%)
4	France	412 (2.94%)
5	Netherlands	371 (2.65%)
6	Canada	267 (1.90%)
7	Finland	230 (1.64%)
8	United Kingdom	201 (1.43%)
9	Russian Federation	170 (1.21%)
10	Switzerland	125 (0.89%)



Map shows concentration of reachable Bitcoin nodes found in countries around the world.

LIVE MAP

NOSTR重新定义创作者经济

Web5的逻辑

- UGC是Web5的全部
- Web5收入模型不基于广告
- 账号所有权、数据所有权回归用户

NOSTR重新定义创作者经济

Web5的特点

- 打破了Web2的多中心化封闭垄断局面
- 实现了低成本、自动化的数据去中心化
- 对UGC更直接的经济激励，和其它激励
- 通过限制服务器的功能来阻止服务器作恶
- NOSTR抗女巫攻击，能解决垃圾信息问题
- 数据“去中心化+自我主权”使大数据叙事失灵

NOSTR重新定义创作者经济

自媒体在Web5叙事下不再是个伪命题 - aLE

NOSTR是下一代互联网的底层协议

NOSTR+

- NOSTR+AI
- NOSTR+电商
- NOSTR+直播
- NOSTR+流媒体
- NOSTR+云服务器
- NOSTR+学术期刊、出版社、杂志
- NOSTR+...

答疑



 ale@nostrpurple.com

 ale@getalby.com



 @0x4D718

 Cakksakkas

 @lnbc1p911Driver

