vinteum

VINTEUM YEAR FOUR REPORT

Building Continuity in Bitcoin Open Source

04

(AUGUST 2025 - AUGUST 2026)

vinteum

VINTEUM YEAR FOUR REPORT

A Note from the Executive Director

When we published Vinteum's three-year report, much of the story was about building the organization itself: establishing our mission, creating programs, supporting our first contributors, and helping form a technical Bitcoin community in Brazil. My concern with the sustainability of Bitcoin open-source development predates Vinteum, and it is deeply meaningful to see how much the ecosystem has matured since then. I am proud that Vinteum has become a trusted part of it, but even more proud to see people we supported finding grants, jobs, collaborators, and recognition beyond the organization itself. Contributors who once needed close guidance are taking on greater responsibility, helping newer developers, and building paths that no longer depend on Vinteum being at the center.

As Vinteum grew, I also had to confront the limits of the way I had been leading it. For a long time, I carried a large share of the organization's context, decisions, relationships, and day-to-day execution myself. That gave us speed early on, but eventually became a constraint. **Learning to build an organization that does not depend on me being involved in every detail has been one of the most important and difficult parts of this stage.** It has meant documenting what was previously intuitive, trusting others with responsibilities I had grown used to carrying, and creating more room for other people to lead. The work described in this report is still unfolding. Some results are already visible, while others will take longer to understand. That is inherent to supporting open-source development, research, and people who are still developing deep expertise.

> *What matters to me is that Vinteum is becoming more capable of doing this work with continuity, judgment, and a broader group of people carrying responsibility for it.*

Lucas Ferreira
Founder and Executive Director
Vinteum

vinteum

VINTEUM YEAR FOUR REPORT

Contents

Year Four Overview

Following Vinteum's third anniversary, we published **our first comprehensive report** documenting the organization's origins, mission, programs, and the development of the Bitcoin open-source ecosystem around our work in Brazil.

This report focuses on the twelve months that followed, August 2025 to August 2026. During this period, Vinteum continued supporting established open-source contributors while helping more developers and researchers move into sustained work across Bitcoin Core, rust-bitcoin, Lightning, wallets, mining, privacy, testing, cryptography, and alternative node implementations. Several alumni from the first Bitcoin Dev Launchpad cohort progressed into independent grants, full-time Bitcoin roles, or long-term open-source contribution, while a second cohort moved into fellowships and more specialized technical work.

At the same time, Vinteum strengthened its internal capacity across engineering, research, governance, and operations, while Casa21, BitDevs, university initiatives, and the broader developer community continued to expand around these efforts.

> *What follows is a closer look at the people, projects, lessons, and organizational changes that shaped Vinteum's fourth year.*

Leadership and Governance

As Vinteum has grown, its leadership structure has become more defined across strategy, engineering, research, governance, and operations.

Founder and Executive Director Lucas Ferreira remains responsible for Vinteum's overall strategy and direction, fundraising, contributor development, and allocation of resources. A significant part of this role is maintaining enough technical and ecosystem context to understand where projects need support, which contributors are developing particular strengths, how their work is being received by maintainers, and where funding, mentorship, or introductions can have the greatest effect.

Much of this work can be understood as **matching people, problems, capital, institutions, and timing.** This includes helping contributors identify useful directions, connecting them with maintainers and reviewers, supporting transitions into external grants or employment, and maintaining relationships with funders and organizations that may provide longer-term paths. Lucas also remains closely involved with contributors as they develop, following their progress and maintaining relationships across the projects where they work to understand when additional mentorship, a change of direction, or a new opportunity may be useful.

Bruno Garcia, Vinteum's Head of Engineering, combines his own work in Bitcoin Core and bitcoinfuzz with technical leadership across Vinteum's developer programs. He mentors fellows and grantees, reviews technical progress, helps evaluate project direction, and brings the perspective of active Bitcoin open-source development into how Vinteum trains and supports contributors.

Edil Medeiros serves as Vinteum's Head of Research. Drawing on his experience as a professor and researcher, he helps shape the pedagogical and research structure of Vinteum's programs, particularly for contributors working in areas such as cryptography, privacy, protocol research, and technical investigation. His role is especially important in helping contributors move from implementation toward more rigorous research questions and methods.

André Neves contributes to Vinteum's strategic direction through his role on the board, which he serves alongside Lucas Ferreira and Bruno Garcia. He participates in strategic discussions, fundraising, partnerships, and major organizational decisions. His technical and entrepreneurial background brings a distinct perspective to questions at the intersection of open-source development, institutional strategy, and long-term sustainability.

Strengthening Operations

Vinteum intends to remain a small and efficient organization, but its growth has made stronger operational capacity necessary. **Thaiz Batista's arrival as Head of Operations** marks an important step away from a model in which too much organizational context and execution remained concentrated in the founders.

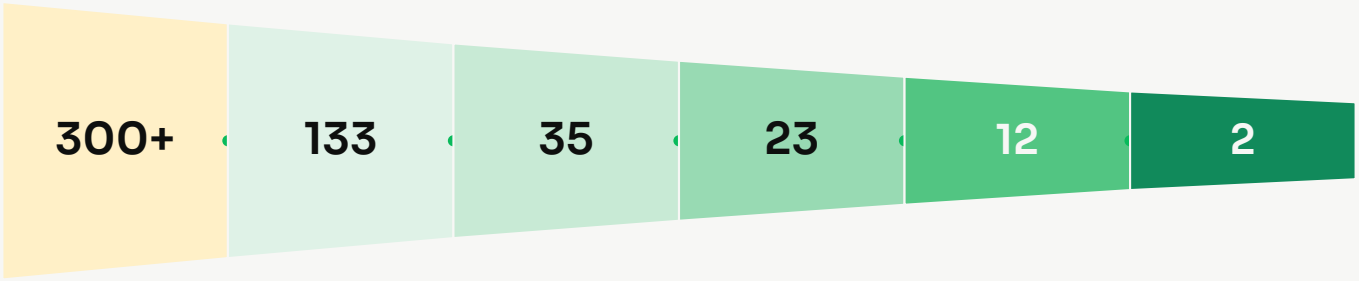
The objective is straightforward: build enough operational resilience that Vinteum can support a growing developer and research community without requiring the same small group of people to carry every process and piece of institutional knowledge.

Thaiz brings more than a decade of experience in mission-driven organizations. She previously served as COO of Instituto Mises Brasil, where she helped strengthen operations and structure its postgraduate program. More recently, she served as the Executive Director of Students For Liberty Brasil, with responsibilities across fundraising, program execution, organizational leadership, and team coordination in a global reach organization.

At Vinteum, her role spans financial and administrative processes, internal planning, grants and fellowships management, Casa21, events, partner relationships, payments, reporting, and program coordination. A central part of the work is turning processes that developed organically during Vinteum's early years into systems that can be documented, delegated, and maintained over time.

Vinteum in **Numbers**

Developer Pipeline



300+	133	35	23	12	2
Applications received for BDL2	Developers selected to begin the program	Completed the assignments and developed a proof of concept	Selected for the in-person residency in Florianópolis	Selected for Vinteum fellowships after BDL2	Senior participants moved directly into Vinteum grants

Contributors in the Ecosystem

8 Active Vinteum grantees

13 Active Vinteum fellows

13 Additional Brazilian developers now working full-time in Bitcoin open source after engaging with Vinteum's programs, community, mentorship, or network

36 Members of the Vinteum community currently engaged in Bitcoin open-source development

Community

67 BitDevs events, from August 2025 to Aug 2026

9 Cities with recurring BitDevs communities

120+ Students reached through Bitcoin Students Day

8 Universities represented in Bitcoin Students Day activities

22 Recurring weekly online sessions across office hours, study groups, public mentorship, and English conversation

First Cohort: One Year Later

BDL1 outcomes are now visible and diverse.

Several alumni have secured independent grants: Lucas Balieiro, Moisés Pompílio, João Leal, Guilherme Martins, and Joãozinho through OpenSats, and Luis Schwab through the BDK Foundation. Others moved into professional roles, including Caio at Boltz, Luca at Second, and Gustavo Stingelin at Lightning Labs. Erick Cestari and ThgO.O became Vinteum grantees.

These trajectories reinforce an important lesson: **there is no single timeline into Bitcoin open source.** Participants entered with different levels of technical experience, progressed at different speeds, and found their way through different combinations of structured programs, BitDevs, Casa21, Summer of Bitcoin, universities, independent contribution, and direct relationships with projects.

Vinteum's role is not to act as a gatekeeper, but to make those paths easier through technical context, mentorship, funding, introductions, and community. ***The goal is not to own the pipeline, but to strengthen an ecosystem capable of producing and supporting contributors through many different paths.***

A longer follow-up on the first cohort and individual alumni trajectories is available on the Vinteum blog.

BITCOIN DEV LAUNCHPAD

Second Cohort: Program and Lessons Learned

> *The second Bitcoin Dev Launchpad cohort began with a clearer objective:* ***not simply to bring new people into Bitcoin open source, but to understand how to support contributors in developing the judgment and independence needed to sustain useful work over time.***

Vinteum is still learning how to do this well. BDL2 became an important experiment in understanding which forms of training, review, and mentorship actually help participants grow into contributors capable of working responsibly inside open-source projects.

Artificial intelligence made this harder to evaluate. AI tools can help participants navigate unfamiliar code, documentation, and technical problems, but they can also produce plausible work that conceals gaps in understanding. We relied more heavily on individual discussion and iterative review, asking participants to explain their decisions, assumptions, alternatives, and uncertainties. **The relevant question was no longer simply whether someone could produce a solution, but whether they could understand, defend, review, and improve it.** This also made the program less scalable, since evaluating genuine understanding requires **sustained individual discussion, review, and mentorship.**

The same concern shaped how participants were introduced to upstream projects. Maintainers were increasingly dealing with low-quality AI-assisted submissions, and sending inexperienced contributors directly into repositories risked transferring part of Vinteum's educational burden to them. Participants often worked first in their own forks, where Vinteum grantees, fellows, and peers could review their work before it reached maintainers. Several maintainers positively commented on receiving more mature contributions after this internal review process.

BDL2 also broadened what counted as meaningful progress. **Proofs of concept, technical investigations, benchmarks, code review, reproducing bugs, improving tests, and even determining that an idea should not be submitted upstream** could all demonstrate useful technical development. A mature pull request remained a valuable outcome, but raw pull-request volume became a weaker measure of progress than depth of understanding and quality of judgment.

Second Cohort

PROGRAM AND LESSONS LEARNED

This led to another important lesson about mentorship.

Close mentorship can accelerate learning and provide context that is difficult to acquire from documentation alone, but visible progress can remain dependent on someone else continuously defining the next step. **Good mentorship should therefore become less necessary over time.** Its purpose is not to remove uncertainty, but to help contributors become better at navigating it: identifying useful problems, interpreting feedback, recovering from failed approaches, and deciding what is worth pursuing independently.

For that reason, BDL2 increasingly valued **depth, judgment, consistency, review ability, and growing independence over visible output alone.** These qualities are harder to measure during a short program, but they are closer to what we believe long-term open-source contribution actually requires.

With the exception of TheMhv, all of the fellows featured in the next section came through BDL2. Their work spans Bitcoin Core, rust-bitcoin, wallets, privacy, mining, cryptography, network analysis, testing, and alternative node implementations. It is still too early to evaluate these trajectories in the same way as the first cohort. Many are making meaningful progress but have not yet demonstrated the independence expected of long-term contributors.

That uncertainty is itself part of what Vinteum is trying to learn.

Bringing talented people into the ecosystem is only the beginning. The longer-term test of BDL2 is whether participants continue developing after the program's structure is removed, and whether Vinteum can improve the conditions that support that transition.


vinteum


Prize Breakdown - Best Overall
• 1st Place: 2.5M Sats + Bitcoin++ Tix

vinteum

VINTEUM YEAR FOUR REPORT

Current Fellows: A Work in Progress

Our Fellowship Program bridges technical education and long-term open-source funding.

Fellows work full-time on Bitcoin open source with financial support, close mentorship, and regular interaction with experienced contributors, but the fellowship is deliberately a development stage, not a grant for someone already expected to operate independently. There is no single model for a successful fellowship.

Some fellows arrive as experienced software engineers who need Bitcoin-specific context and relationships with maintainers. Others need experience inside large open-source projects, stronger technical foundations, or time to pursue research-heavy directions.

With the exception of TheMhv, all of the fellows featured below came through the second Bitcoin Dev Launchpad cohort. **Their reports are snapshots of trajectories still unfolding**: the projects they are joining, the questions they are pursuing, the skills they are developing, and the responsibilities they are beginning to assume.

vinteum

CURRENT FELLOWS

K-YU

Caio, also known as K-YU, is working full-time on bitcoinfuzz under the mentorship of Vinteum Head of Engineering Bruno Garcia. Over the past months, his fellowship has evolved from expanding fuzzing coverage into using **differential fuzzing to investigate the boundaries of Bitcoin consensus across independent implementations**. One of his main projects was completing the integration of `libbitcoin-system` into bitcoinfuzz.

Getting the new target working required investigating divergences in script verification, transaction validation, address parsing, and block deserialization. Several of these investigations led to upstream fixes in `libbitcoin-system`, while others required improvements to bitcoinfuzz's documentation, CI, corpora, and test targets. **The value of this work goes beyond finding crashes.**

By comparing how different Bitcoin implementations process the same inputs, K-YU has been learning to distinguish bugs from permissible implementation differences and to identify when a divergence exposes consensus-relevant behavior. This kind of reasoning has become an increasingly important part of his fellowship.

As the `libbitcoin-system` integration matured, K-YU began shifting part of his attention toward Bitcoin Core. He has been learning Core's development workflow, build system, tests, and contributor documentation while making his first code reviews, including work related to PSBTv2, BitcoinKernel bindings, and transaction encapsulation. These contributions are also helping him deepen his C++ knowledge and understand how validation logic is expressed and reviewed in Bitcoin's most widely used implementation.

In July, K-YU joined 2140's Consensus Code Week in Amsterdam, where discussions with other developers helped connect many of the topics he had been studying independently and sharpen his understanding of the distinction between Bitcoin consensus rules and implementation-specific behavior. His work is expanding beyond adding fuzz targets. By studying implementations side by side, reviewing Bitcoin Core, and investigating why implementations disagree, **K-YU is building the technical judgment needed for deeper work on Bitcoin consensus and validation.**

vinteum

CURRENT FELLOWS

Renato Britto

Renato Britto joined the fellowship after nearly six years working on production backend systems, including identity and authentication infrastructure at Coinbase. He left that career after completing Vinteum's Bitcoin Dev Launchpad to work on Bitcoin open source full-time. He now works across the rust-bitcoin ecosystem with weekly mentorship from maintainer Tobin Harding, as the ecosystem pushes toward stable 1.0 releases.

One of his main projects is improving how Rust applications talk to Bitcoin Core. He **contributed to a new Bitcoin Core RPC that exposes an OpenRPC specification, and is using that interface to generate a production client for corepc**. He has continued reviewing the main OpenRPC work, keeping corepc current with each Core release, and scoping future IPC support.

More recently he has been working on release compatibility.

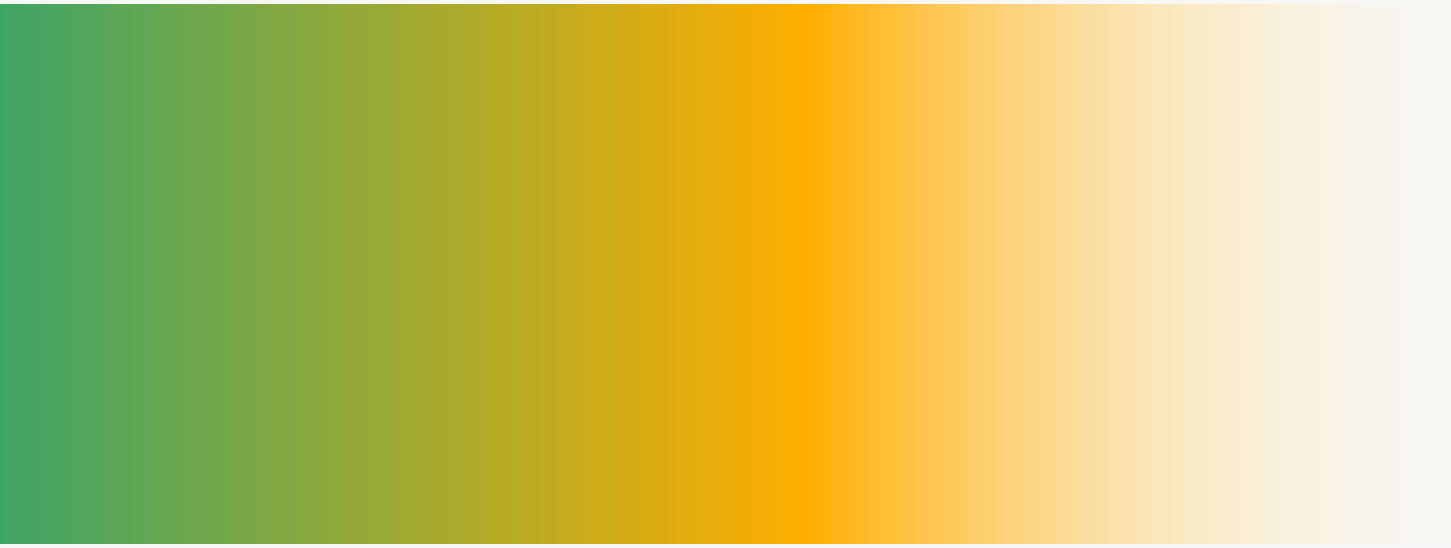
Breaking API changes between releases have made upgrading costly enough that most projects are still pinned to old versions while newer work sits unused on master. **Renato is building and testing a compatibility layer that lets projects migrate a piece at a time**, proving it against real downstream users like rust-miniscript and BDK, and feeding every problem he hits back into the library's design.

Alongside that, Renato reviews incoming contributions across the organization, keeps dependencies current, and keeps CI healthy on both master and the LTS branches. On the quality assurance side, **a new qa-assets repository he started gives the fuzzers real seed corpora to work from**, part of building out corpus-based fuzzing infrastructure. Across API design, downstream compatibility, CI, fuzzing, release engineering and review, **Renato is growing into someone who can carry a portion of rust-bitcoin's maintenance burden**.

vinteum

CURRENT FELLOWS

Victor André



Victor André is working full-time on Bitcoin peer-to-peer privacy and network analysis, guided by Edil Medeiros. **His fellowship focuses on understanding what information remains observable when Bitcoin nodes use BIP324 encrypted transport.** Victor built a **Warnet-based BIP324 Traffic Lab and a passive analysis pipeline to study signals such as packet size, timing, direction, flow structure, and network ports.**

His experiments so far suggest that **the BIP324 handshake is the strongest passive signal, while other events such as block arrivals, compact blocks, large transactions, and address responses produce significantly more false positives.**

More recent work has focused on refining the methodology and studying traffic-analysis techniques that could improve classification accuracy. **The research also highlighted the role of Bitcoin Core's default P2P port as an inexpensive signal for identifying Bitcoin traffic.** This led Victor to implement and discuss approaches for **port randomization in Bitcoin Core**, with a persistent randomized listening port emerging as the preferred direction.

He has also begun contributing to the `rust-bitcoin` ecosystem around BIP324, including work to add bitcoinfuzz coverage for `rust-bitcoin`'s implementation. This gives him another implementation through which to study the protocol while contributing tests, fixes, and review where useful.

vinteum

CURRENT FELLOWS

Jayr Motta

Jayr Motta is contributing full-time to Mujina, **an open-source firmware project for Bitcoin mining hardware.** Over the past months, his work has spanned protocol integration, firmware development, and increasingly hands-on hardware experimentation. One of his main contributions has been bringing Stratum V2 support to Mujina.

As the implementation matured, Jayr worked closely with reviewers to address feedback and improve the integration, while also studying questions around channel design and BIP 323. His work connects Mujina to the broader Stratum V2 ecosystem and helps explore how the protocol behaves on real mining hardware rather than only in software environments. Jayr has also investigated how mining work is distributed across ASIC chips, including experiments with nonce-space slicing and instrumentation to study nonce distribution.

While another contributor later took the lead on this research, the work gave him deeper context on how firmware coordinates large numbers of chips and how those design decisions affect mining performance. More recently, his focus has shifted toward Mujina's configuration architecture. He reviewed MIP-0001, implemented a proof of concept to test the proposed design, and is now working toward a more complete configuration system while continuing to contribute to firmware improvements such as fan control.

At the same time, Jayr is building a small hardware lab to test his work against physical devices and deepen his understanding of board design, ASICs, cooling systems, assembly, and manufacturing constraints. **This is helping him bridge the gap between writing mining firmware and understanding the hardware it controls.**

vinteum

CURRENT FELLOWS

m4ycon

m4ycon is working full-time on **Bitcoin network monitoring and observability**, studying how transactions and blocks propagate through the network and building tools that allow this behavior to be measured and analyzed. Over the past months, he continued his work around compact block relay. A `peer-observer` contribution for monitoring compact block reconstruction was merged, providing infrastructure for analyzing when nodes can reconstruct a compact block immediately and when additional network communication is required.

Building on this work, Maycon studied Bitcoin Core PR #35558, which proposes pre-filling compact blocks with additional transactions to improve reconstruction rates. He ran patched Bitcoin Core nodes, collected detailed logs, analyzed the resulting data, and contributed his findings back to the review discussion. Another major project is the **Cluster Mempool Observer**, inspired by a research idea proposed by b10c.

As the project developed, Maycon began describing it more broadly as a mempool recorder and replayer: infrastructure capable of preserving the state and evolution of a node's mempool so researchers can later ask what was happening at a particular moment and analyze how transactions were organized and prioritized.

In June, he built the initial backend, and in July focused on data integrity between the observer and Bitcoin Core while beginning development of a frontend. His goal is to make the tool publicly accessible and eventually use the recorded data to investigate questions around mempool performance and behavior. **The project could give researchers a way to study Bitcoin Core's evolving mempool architecture using historical data rather than only observing a node in real time.**

Alongside these projects, Maycon maintains a heavily instrumented Bitcoin node for long-term network research. He has improved how its detailed debug logs are stored and compressed, allowing months of network activity to be retained for future investigations. Vinteum's infrastructure has also been used for his compact block experiments, including tests designed to produce more realistic measurements across independently connected nodes.

Maycon's fellowship is increasingly moving from operating observability infrastructure toward **designing experiments and building research tools of his own**. By combining long-running network measurements, Bitcoin Core review, and purpose-built data collection, he is developing the tools and experience needed to investigate how Bitcoin behaves under real network conditions.

vinteum

CURRENT FELLOWS

Octavio Lucca

Octavio Lucca is working full-time on **Bitcoin network observability**, with a growing focus on bringing monitoring and analysis infrastructure to Floresta. His fellowship began with hands-on work across the `peer-observer` stack, including its data collection, archiving, and replay infrastructure. More recently, he completed a series of improvements to the archive replayer, making it more robust and moving its core logic into a reusable library that can be tested independently from the command-line interface.

This makes historical network data easier to validate, replay, and use in future analysis. As Octavio began adapting these tools to Floresta, he encountered differences between Floresta's RPC interface and the data exposed by Bitcoin Core. This led him to start contributing directly to Floresta, particularly its `getpeerinfo` RPC.

His work has added and refined fields related to peer services, connection types, inbound status, relay behavior, permissions, time offsets, and compact block support, bringing the interface closer to Bitcoin Core while accounting for differences in Floresta's architecture. These improvements are enabling a broader **Floresta observability effort**.

Octavio has already brought a Floresta-compatible fork observer deployment online, allowing stale blocks and chain forks to be studied using data from an independent Bitcoin implementation. He is also developing a Floresta-specific observer using Prometheus and Grafana, with the goal of making its network behavior publicly observable in much the same way existing infrastructure monitors Bitcoin Core nodes. Alongside this, Octavio is also working on a DNS monitoring project that tracks Bitcoin DNS seed behavior over time.

This adds another view into how nodes discover peers and how the network's bootstrapping infrastructure behaves in practice, complementing the peer and fork observability work with data from an earlier stage of the connection lifecycle. **The project creates a useful feedback loop between observability and implementation development:** missing data in Floresta leads to RPC improvements, those improvements unlock better monitoring, and the resulting observations can in turn expose behavior worth investigating in the node itself.

Through this work, Octavio is helping extend **Bitcoin network observability** beyond a single implementation while contributing directly to Floresta's tooling and interoperability.

vinteum

CURRENT FELLOWS

bc1cindy

Cindy is working full-time on Bitcoin privacy research and tooling, with mentorship from Yuval "nothingmuch" and Armin Sabouri. **Her work began around Payjoin, but has expanded into a broader investigation of how wallet behavior and transaction structure can reveal information on-chain.** Over the past months, Cindy systematically analyzed transaction fingerprints across Payjoin integrations and other Bitcoin wallets.

Wallets often construct transactions differently through choices involving input and output ordering, nSequence, locktimes, script types, sighash behavior, fee-rate rounding, change handling, and coin selection. Her work tracks both chain-observable transaction-level fingerprints and relay-observable behavioral fingerprints at the network layer. **Even when users participate in privacy-enhancing protocols, these implementation differences can make their transactions distinguishable.**

Initially, Cindy worked directly with individual projects to identify and reduce these leaks. Her research led to issues and patches across Cake Wallet, Bull Bitcoin, Liana, **`bitcoin_base`**, and **`rust-payjoin`**, including changes to ordering, change behavior, locktime patterns, and coin selection. As the number of fingerprints and integrations grew, she shifted from addressing each case individually toward building reusable research infrastructure.

This led to **lumen-fingerprints**, a project that uses Floresta to study how transactions across the Bitcoin blockchain are constructed and measure how identifying particular wallet behaviors may be. Cindy is also developing decluster, an experimental evidence-weighted de-anonymization framework that combines wallet fingerprints, transaction amounts, payment-graph structure, and techniques such as dense-subset-sum to study common-ownership inference. Rather than treating every signal as definitive, the model scores different pieces of evidence and can account for cases where collaborative transactions create misleading ownership links.

This offensive research serves a defensive purpose. By demonstrating how transactions can be clustered and wallets identified, Cindy can help developers understand which behaviors need to change and evaluate whether privacy protocols actually make users blend into a broader anonymity set. Her work is now increasingly connected to the **Fungi protocol**, where she has reviewed concurrent PSBT development and is helping shape future privacy improvements.

Her next focus includes privacy-preserving transport over Tor and applying findings from this research as the protocol develops.

vinteum

CURRENT FELLOWS

Lucas Lima

Lucas Lima is working full-time on Bitcoin protocol research and implementation, with a focus on **Fountain Codes, btcd, and network-level experimentation**. His fellowship combines cryptography study with hands-on protocol development. One of his first major projects was implementing BIP 152 Compact Block Relay in btcd, including block reconstruction, missing-transaction requests, relay modes, fallback behavior, and tests based on Bitcoin Core.

He has continued refining the implementation in response to review. His main research direction explores whether **Fountain Codes could allow pruned Bitcoin nodes to contribute historical block data without retaining the entire blockchain**. The proposed architecture divides blocks into epochs and encodes them into smaller fragments, or droplets, distributed across nodes.

A bootstrapping node could then recover historical blocks after collecting enough droplets from multiple peers. Over the past months, Lucas moved this work from simulations into a **proof of concept integrated directly into btcd**. The implementation groups blocks into epochs, generates and stores droplets, and adds peer-to-peer functionality for nodes to exchange them.

He has also packaged the modified node into Warnet and run it in isolated regtest environments. The research has attracted substantive technical feedback. Discussions on Delving Bitcoin raised questions around storage versus bandwidth tradeoffs, peer requirements, malicious peers, fingerprinting, and whether historical block availability currently justifies the added complexity.

Lucas is now preparing larger Warnet experiments to measure bandwidth consumption, peer requirements, and reconstruction costs rather than treating adoption as a predetermined outcome.

Alongside this work, he continues studying applied cryptography and experimenting with other Bitcoin interfaces, including changes to `libbitcoinkernel` in his Bitcoin Core fork. **The value of the fellowship is not dependent on Fountain Codes eventually becoming part of Bitcoin.**

By turning the idea into a working implementation and measuring its behavior under more realistic network conditions, Lucas can help establish where the approach is useful, where its tradeoffs become prohibitive, and whether further work is justified.

vinteum

CURRENT FELLOWS

Yan-pi

Yan is working full-time on **Bitcoin wallet infrastructure**, with a growing role in Bitcoin Dev Kit and regular mentorship from BDK maintainer Leonardo Lima and Vinteum Head of Research Edil Medeiros. A major focus of his fellowship is research into **descriptor metadata and wallet synchronization**. Descriptor wallets typically rely on a gap limit when restoring or importing a wallet, deriving addresses until enough consecutive unused ones are found.

This can become inefficient when previously used addresses are spread across a large derivation range. Yan is exploring whether wallets can instead preserve compact metadata about which script indices have already been derived, revealed, or used. Building on BIP 393's proposal for descriptor annotations, the project investigates encoding this information alongside descriptors, including compact representations such as Elias-Fano encoding.

If viable, an imported wallet could locate relevant scripts more efficiently instead of rediscovering its history primarily through the traditional gap-limit process. Over the past two months, Yan has moved the project from Leonardo's initial proof of concept toward a **reusable BDK API**. The work includes data structures for script metadata, encoding and decoding methods, wallet import and export functionality, and APIs for deriving specific indices.

With most of this infrastructure implemented, the research is moving into benchmarking and validation to determine whether the added metadata produces meaningful synchronization improvements. At the same time, Yan has become increasingly involved in BDK itself. He contributed support for TRUC-related policy rules in bdk_wallet, continues adapting that work for the upcoming 4.0.0 release, and has increasingly prioritized reviews, testing, and participation in BDK team discussions.

Rather than measuring progress only through his own pull requests, he has been learning where review and validation can be most useful to the project. His fellowship has also expanded into Bitcoin data analysis. During Vinteum's in-person deep dives, Yan helped explore a revival of BlockSci, contributing build fixes and participating in discussions around a new architecture for the project.

Yan's progression reflects a shift from learning how to contribute to becoming more deeply embedded in the development process of a major wallet library. **His current work combines implementation, API design, research, and review**, with the descriptor metadata project providing a concrete opportunity to test whether a new wallet design can translate into measurable improvements for users.

vinteum

CURRENT FELLOWS

allocz

allocz is working full-time on **btcd performance, resource usage, and testability**. A major focus of his fellowship has been understanding why btcd performs significantly worse than Bitcoin Core during Initial Block Download. Earlier investigations identified disk I/O caused by UTXO cache misses as an important bottleneck, leading to a skip_prevout_fetch optimization that substantially reduced unnecessary reads.

More recently, allocz identified another major source of resource usage. While investigating why an archival btcd LevelDB database could grow to around 90 GB compared with roughly 12 GB for Bitcoin Core's chainstate, he found that btcd retained spent UTXOs indefinitely in its spend journal to support arbitrary chain rollbacks. After discussing the behavior with maintainers, he implemented **depth-based pruning** for this data.

Combined with his earlier I/O optimization, the change significantly reduced database size and disk activity. On his test setup, with blocks stored on a hard drive and LevelDB on an SSD, allocz can now complete a full btcd Initial Block Download in under 40 hours. On the unmodified version with everything stored on HDD, the same process had previously failed to complete even after months.

He is now exploring further improvements, including Calvin Kim's SwiftSync implementation for btcd and faster signature validation.

As an experiment, allocz replaced btcd's existing cryptographic backend with his own `secp256k1` library. Even with an intentionally inefficient proof of concept that performs additional serialization, **signature validation performance nearly doubled**, suggesting substantial room for improvement in full-validation workloads. The secp256k1 library is another major part of his fellowship.

It provides a portable pure-Go implementation while optionally using `libsecp256k1` when CGO is available. His optimization work has also propagated upstream, including an optimization to Go's math/big.Int.ModInverse that was merged into the Go standard library. Alongside performance work, allocz has been strengthening btcd's testing infrastructure.

He improved its RPC integration-test harness, introduced debugstream support for asserting debug events, and added tests for features and bug fixes that were previously difficult to validate automatically. He has also benchmarked alternative key-value databases and continues reviewing btcd pull requests, including identifying a potentially security-sensitive issue before it was merged. His fellowship increasingly connects performance optimization with **safer, more measurable development practices**.

By reducing I/O and storage requirements, improving cryptographic performance, and building better test infrastructure, allocz is helping make btcd faster while improving the tools available to validate future changes.

vinteum

CURRENT FELLOWS

Vinicius Cestari

Vinicius Cestari is working full-time on **Bitcoin Core and related infrastructure, with a particular focus on inter-process communication and `libmultiprocess`**, the library used by Bitcoin Core to expose functionality across process boundaries.

Over the past months, Vinicius has concentrated on understanding this infrastructure through code review and direct contributions. He has submitted fixes and improvements to `libmultiprocess` covering serialization helpers, documentation, generated code, process error handling, tests, and build behavior, with several already merged.

At the same time, he has been reviewing and testing IPC-related changes in Bitcoin Core, gradually building the context needed to provide more meaningful feedback on complex changes. One experimental application of this work has been exploring how Bitcoin Core's IPC interfaces could support downstream software such as LND. Working with the 2140 Bitcoin Core fork, Vinicius initially prototyped exposing existing RPC functionality through IPC before shifting toward a more generic Cap'n Proto interface after feedback from the project.

The broader goal is to understand how applications could consume Bitcoin Core functionality through structured IPC rather than relying exclusively on JSON-RPC.

His work has also helped him understand Bitcoin Core's existing multiprocess architecture in practice. In July, he began tracing messages between the `bitcoin-node` and `bitcoin-wallet` processes and documenting the execution flow, complementing code review with a more system-level understanding of how IPC calls are generated, transported, and handled.

He has also begun experimenting with **Bitcoin network measurement**. Vinicius built a scanner for identifying operating-system fingerprints exposed by reachable Bitcoin nodes and started operating his own bitnodes crawler. This work resulted in an analysis of OS fingerprints across more than 16,000 Bitcoin nodes and provides another avenue for developing his understanding of Bitcoin's networking behavior.

Vinicius's fellowship is increasingly centered on **building deep familiarity with a technically complex part of Bitcoin Core rather than maximizing the number of isolated contributions**. Through implementation, review, debugging, and experimentation, he is developing the systems-level context needed to contribute meaningfully to Bitcoin Core's evolving multiprocess architecture.

vinteum

CURRENT FELLOWS

Isaque Franklin

Isaque Franklin is following a **research-heavy fellowship path focused on developing the depth required to eventually contribute to Bitcoin's cryptographic infrastructure**. Guided by Vinteum Head of Research Edil Medeiros, much of his time is deliberately dedicated to applied cryptography, `libsecp256k1`, and the mathematical foundations behind the code.

Over the past months, Isaque has been studying `libsecp256k1` from the bottom up, examining field and group operations, modular inversion, safeGCD, exhaustive testing, and the optimizations used throughout the library.

Rather than treating the code as a black box, his goal is to understand why these implementations work and how their mathematical assumptions translate into production cryptographic software. This work has already resulted in public technical writing. In July, he published detailed explorations of modular inversion through Fermat's Little Theorem in the earliest secp256k1 implementation and of the field-element representation used by modern `libsecp256k1`.

He is also studying the library's exhaustive testing infrastructure and beginning to review increasingly complex cryptographic code as his understanding develops. A second major area of study is **MuSig2 and Nested MuSig2**.

Isaque has been working through the Nested MuSig2 paper and reviewing its implementation in a `libsecp256k1` fork, initially concentrating on understanding the construction before moving toward a more security-oriented review. This work is complemented by an applied cryptography study group covering security definitions, computational hardness, Diffie-Hellman, authenticated encryption, and related foundations.

He has also been studying hardware wallet authentication protocols, including the architecture behind Jade's Blind PIN Server, with the longer-term goal of understanding how these designs can be specified and independently implemented. In parallel, he contributed a merged Wasabi Wallet fix implementing a missing `k_max` check introduced by BIP 352, contributed to the ongoing BlockSci revival, and continues experimenting with Odin bindings for `libsecp256k1`. **The pace of this fellowship is intentionally different from contribution paths centered on pull-request volume.**

Given the unusually high cost of mistakes and the high barrier to meaningful review in cryptographic code, Isaque's work is focused on building the theoretical understanding, code-reading ability, and security judgment required to contribute responsibly in an area where Bitcoin has relatively few specialized reviewers.

vinteum

CURRENT FELLOWS

TheMhv

TheMhv is currently working primarily on **Cashu and the Cashu Development Kit (CDK)**. Over the past months, his work has increasingly moved beyond implementing isolated features toward reviewing protocol changes and understanding the cryptographic assumptions behind them. A significant part of his recent work has involved **deterministic DLEQ nonces**.

He reviewed the corresponding NUT specification and CDK implementation, and later worked on simplifying CDK's storage model once deterministic nonce derivation made storing those nonces unnecessary. He has also contributed to replay protection for mint quote lookups, updated earlier work as specifications evolved, and continued reviewing and maintaining open CDK pull requests. At the same time, the fellowship is giving him room to **deepen his cryptography foundations**.

Through Vinteum's crypto study group, he has studied topics including chosen-plaintext and chosen-ciphertext attacks, while independently exploring BLS signatures and pairing-based cryptography in the context of possible future Cashu protocol designs.

In July, he began building a small Cashu BLS proof of concept to understand the construction through implementation rather than only from the literature. His work now sits at the intersection of implementation, protocol review, and cryptographic study. Cashu specifications are still evolving, some contributions have waited significant periods for review, and questions around backwards compatibility and changes to cryptographic behavior remain part of the development process.

The next challenge in his fellowship is less about finding things to work on and more about developing a clearer technical direction. Cashu remains his main project, but his growing interest in cryptography and protocol design creates several possible paths. The coming months should help determine where he can build the deepest expertise and become most useful.

vinteum

VINTEUM YEAR FOUR REPORT

Contributor Reports

Vinteum supports contributors whose work has become important to the projects and technical areas in which they operate. The objective is not only to support individual output, but to provide the continuity needed for experienced contributors to take on deeper technical responsibilities, support other developers, and strengthen the projects around them.

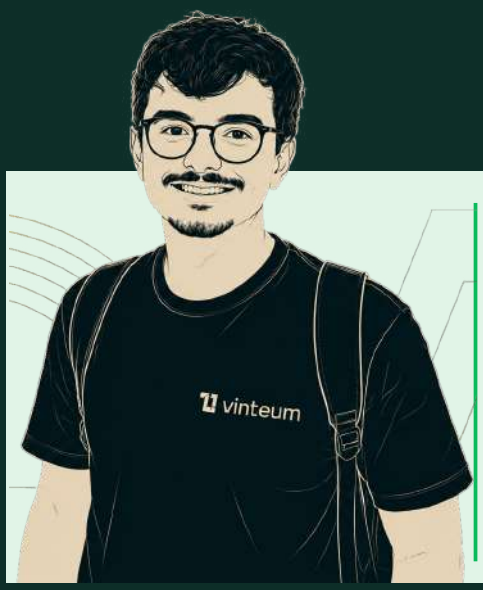
Most of the contributors featured in this section are supported through Vinteum grants. The section also includes Bruno Garcia, Vinteum's Head of Engineering, whose role combines direct technical contribution with engineering leadership across the organization. While his broader work mentoring and guiding contributors is discussed in the Leadership and Governance section, his report here focuses on his own work in Bitcoin Core, bitcoinfuzz, testing, and research.

The following reports examine the concrete work carried out over the past year, including implementation, review, maintainership, research, project coordination, and the development of more durable open-source infrastructure.

vinteum

CONTRIBUTOR REPORTS / HEAD OF ENGINEERING

Bruno Garcia / Bitcoin Core, bitcoinfuzz



Over the past year, Bruno has remained a consistent contributor to Bitcoin Core, with work concentrated in ***networking, fuzzing, and the script and wallet subsystems.*** *A significant thread of the work strengthened Bitcoin Core's fuzz testing coverage: he expanded the* ***connman*** *fuzz target with new invariants for added-node management, outbound-byte accounting, network-activity state, and local-service registration, added wallet fuzz targets, and cleaned up the script and descriptor fuzz harnesses by removing redundant and expensive operations.*

He also contributed additional fixes across Bitcoin Core's P2P and networking code. One of his main focuses during this last year was building **`bcore-mutation`**, a mutation testing tool for Bitcoin Core, together with the infrastructure needed to run it systematically. Mutation testing deliberately introduces small changes into a codebase and checks whether the existing test suite detects them, making it useful for identifying gaps that ordinary coverage metrics may miss.

Using mutation-testing reports, Bruno identified and addressed missing tests - e.g. Branch-and-Bound coin selection edge cases, bare-multisig descriptor parsing coverage, the OP_PUSHDATA2 65535-byte threshold, and other cases. He also landed roughly two dozen changes that matured the tool itself.

These included new mutation operators, improvements that eliminate useless or no-op mutants, persistent SQLite storage for projects, runs, and mutants, contextual git-diff generation, and support for running the framework against the libsecp256k1. **Review remained another major part of Bruno's work.** He submitted 124 pull-request reviews across roughly fifteen repositories, including 52 Bitcoin Core PRs and 56 in bitcoinfuzz.

His Bitcoin Core reviews were concentrated primarily in fuzzing and test coverage, with additional work across P2P networking, wallet, RPC, private-broadcast, and HTTP-server changes. Bruno also continued as a driving maintainer and lead contributor to bitcoinfuzz, where he merged 144 pull requests from the contributor community, and authored 117 commits of his own.

A major focus was expanding differential fuzzing across independent Bitcoin implementations. He added a cross-implementation verify_script target covering Bitcoin Core, btcd, gocoin, and NBitcoin; extended PSBT parsing to include PSBTv2 (BIP-370) across rust-psbt, NBitcoin, libwally-core, and Bitcoin Core; introduced new `libbitcoinkernel` block and transaction targets; and added a MuSig2 signing-session target. A second major thread was the infrastructure required to run many independent implementations inside the same fuzzing environment.

Bruno worked on symbol-prefixing for vendored secp256k1 and libbitcoinkernel, Go and CGo integration issues, AFL++ support, Docker and CI improvements, leak detection, and dependency maintenance. This work helped make the tool reliable enough to continuously compare implementations written in different languages and surface real discrepancies across projects including gocoin, btcd, libbitcoin-system, bitcoinj, NBitcoin, libwally-core, and Floresta. Bruno also contributed to **Fuzzamoto,** particularly around its compact-block intermediate representation.

His work included fixes to `HeaderAndShortIds` construction, improved transaction prefill behavior, and a performance improvement that avoids unnecessary work when compiling compact blocks. He also contributed improvements to RPC-driven scenarios, Docker support, and developer tooling. A common thread across these efforts is making it easier to detect correctness problems systematically, strengthen existing test suites, and compare behavior across independent Bitcoin implementations.

Bruno also completed his MSc in 2025, with research motivated directly by his work on Bitcoin software testing. His thesis focused on the intersection of **mutation testing and fuzzing,** resulting in three papers on peer-to-peer software testing, differential fuzzing for identifying equivalent mutants, and the use of mutation testing to evaluate fuzz targets. The research feeds directly back into the projects he works on, particularly `bcore-mutation`, bitcoinfuzz, and his broader efforts to improve testing methodology across Bitcoin implementations.

vinteum

GRANTEE REPORTS

Davidson Souza / Floresta

Over the past year, Davidson Souza has focused on maturing Floresta into a more structured, security-conscious, and reliable. With Vinteum's support, Floresta developed a new visual identity, moved into its own dedicated organization, and introduced a public roadmap organized around quarterly goals. These changes have made the project's priorities more transparent, improved coordination across a distributed team, and made it easier for contributors to understand where their work can have the greatest impact.

Security and reliability have become central priorities. Floresta adopted a formal security policy and internal procedures for responding to critical vulnerabilities, while continuing to expand its testing, hardening, and monitoring infrastructure. Davidson combines manual code analysis, automated testing, custom monitoring harnesses, and AI-assisted analysis to identify defects and unusual behavior.

This work has already uncovered a small number of security-relevant issues, which were promptly investigated and patched. As lead maintainer, Davidson also dedicates a significant portion of his time to reviewing contributions across Floresta and rustreexo. Over the past twelve months, the projects merged 271 pull requests, all reviewed by Davidson, while he authored 52 of them himself.

His own contributions have focused primarily on improving security and reliability, addressing edge cases detected through monitoring, removing performance bottlenecks, and simplifying the codebase to make it easier to maintain and extend. Davidson has also continued working on research aimed at significantly accelerating Floresta's initial block download. Since early 2026, he has been collaborating with JoseSK999 on an assumevalid-based implementation of Swift Sync.

The pull request is approaching completion, and the authors of the Swift Sync proposal are now concentrating their efforts on helping finalize the implementation in Floresta as a way to demonstrate and evaluate its concepts in practice. This collaboration also reflects a broader role Floresta is beginning to play in the Bitcoin ecosystem. Beyond serving as an alternative full-node implementation, the project is increasingly being viewed as a testing ground for new protocol ideas.

Its modular architecture and comparatively flexible development environment allow researchers and developers to experiment with proposals, study their tradeoffs, and produce working implementations before broader adoption is considered.

In parallel, Davidson and other contributors are exploring a faster Swift Sync design that does not rely on assumevalid. This work involves investigating how greater concurrency could improve synchronization without introducing unnecessary complexity or compromising the project's security model. Davidson has been running a series of experiments to better understand the performance gains, architectural requirements, and tradeoffs involved.

Another area of active development is Floresta's compact-filter infrastructure. Davidson is working on improvements that would allow users to rescan for historical transactions before the node has finished processing the entire blockchain. The proposed changes are also intended to reduce the disk space required to store filters and make their behavior more reliable.

Together, these improvements could provide a smoother experience for wallets and applications that rely on Floresta for private transaction discovery. His exploratory work has also examined how Floresta performs on mobile hardware. In an early benchmark, Davidson completed an initial block download on a Pixel phone in approximately seven hours, close to the practical limit of his internet connection.

The experiment provides an encouraging indication that fully validating Bitcoin infrastructure may be practical on increasingly constrained and widely available consumer devices. Beyond improving Floresta as a complete node, Davidson is investigating how its individual components can serve a broader range of developers. This includes exploring integrations with second-layer protocols and making Floresta's internal crates more useful to projects that may not need the entire node implementation.

He is also collaborating with other open-source efforts, including work related to an Electrum protocol design better suited to personal servers and discussions around a reusable libbitcoinkernel interface that could also be adopted by Floresta. Taken together, this work reflects an important stage in Floresta's development. The project is moving beyond rapid experimentation toward clearer governance, stronger security practices, measurable development goals, and an architecture designed for broader reuse across the Bitcoin ecosystem.

At the same time, it is beginning to establish itself as a practical testing ground for protocol research, giving researchers a place to turn new ideas into working implementations and evaluate them under real-world conditions.

vinteum

GRANTEE REPORTS

Plebhash /SRI - Stratum V2 Reference Implementation

*Over the past year, Plebhash has focused on building the infrastructure required for broader, production-grade adoption of Stratum **V2**. His work has increasingly centered on connecting the Stratum **V2** Reference Implementation to Bitcoin Core and making its components easier for mining pools, software developers, and other infrastructure providers to integrate.*

Throughout 2025 and 2026, he devoted a significant portion of his time to **`bitcoin_core_sv2`**, a Rust crate that uses Bitcoin Core's new inter-process communication interface to support the Stratum V2 Job Declaration and Template Distribution protocols.

The project became the first active consumer of Bitcoin Core's IPC interface, providing an important real-world test of the new architecture. Through this integration, Plebhash and the SRI community were able to identify and report issues upstream, helping improve both the Stratum V2 implementation and Bitcoin Core's emerging IPC infrastructure. Job Declaration is a central component of Stratum V2's decentralization model.

It allows miners to propose their own block templates instead of relying entirely on a pool operator to select transactions. Making this functionality practical for existing mining infrastructure requires more than defining the protocol. It also requires stable libraries, clear integration paths, and implementations that can interact reliably with Bitcoin Core.

In 2026, Plebhash led a major refactoring of the SRI Job Declarator Server, transforming it into the reusable jd_server_sv2 library. The new crate builds on **`bitcoin_core_sv2`** while preserving the flexibility to support integrations that do not use Bitcoin Core IPC. Its goal is to make it easier for mining pools to incorporate the Job Declaration Protocol into their own infrastructure without having to adopt the complete SRI stack.

Plebhash has also led development around sv2-uniffi, an effort to generate C++ and Python bindings from SRI's Rust codebase.

By making SRI components accessible from languages already widely used throughout the mining industry, this work lowers the integration barrier for pool operators and other companies whose existing systems are not written in Rust.

These developments have taken place during an important period for Stratum V2 adoption. In 2026, mining companies and infrastructure providers including ANTPOOL, Foundry, F2Pool, MARA Foundation, and CKPool publicly committed to integrating Stratum V2.

As interest has shifted from experimentation toward deployment, the availability of reusable libraries, language bindings, and reliable Bitcoin Core integration has become increasingly important.

Plebhash has also taken on a broader technical leadership role through the Stratum V2 Working Group. Since its creation, he has led the technical portion of its monthly calls, helping mining companies, protocol developers, and implementers coordinate on unresolved design and interoperability questions.

The group's discussions have included conventions for identifying pools and miners through tags in a coinbase transaction's scriptSig, hashrate limits for Stratum V2 Standard Channels, and changes that resulted in BIP 323. Other topics have included preparing the Template Distribution Protocol for potential consensus changes related to BIP 141, designing a protocol extension for non-custodial payouts directly through coinbase transaction outputs, and considering the inclusion of Extended Channels within the Group Channels model.

This combination of implementation work and protocol coordination places Plebhash at an important point in the Stratum V2 ecosystem. He is helping translate the protocol from a specification and reference implementation into infrastructure that existing mining operations can realistically adopt. His work strengthens the connection between Bitcoin Core, miners, and pools while advancing the longer-term goal of giving individual miners greater control over block construction.

vinteum

GRANTEE REPORTS

ThgO.O /BTCPay Server

Over the past year, ThgO.O progressed from a Vinteum Fellow to a grantee supported by Vinteum in collaboration with the BTCPay Server Foundation. He also joined BTCPay Server's core development team, focusing on making self-hosted Bitcoin payments more secure, accessible, and practical for merchants, organizations, and developers.

One of his main contributions was improving shared custody with BTCPay Server. He developed a guided multisignature wallet setup and a more granular permission model, allowing store owners to define signing policies, invite cosigners, collect keys, and coordinate transaction signing directly through the interface. Wallet capabilities such as viewing, creating, signing, and broadcasting transactions can now be assigned separately, including through the Greenfield API. This allows participants and integrations to perform specific operations without receiving broad administrative access.

As part of this work, ThgO.O also introduced a native macOS ARM64 release of BTCPay Vault, improving hardware wallet support on Apple Silicon devices. He also redesigned Payment Requests with better search, filtering, reporting, and more relevant labels. Another major focus was BTCPay Server's plugin ecosystem.

As the maintainer of Plugin Builder, he drove the project's development through its 1.0 release, adding ratings and reviews, identity verification with Nostr and Github, plugin ownership tools, stronger automated testing, and compatibility controls between plugins and server versions. He also helped integrate the public plugin directory directly into BTCPay Server, making community-developed extensions easier to discover, evaluate, and maintain.

ThgO.O has also been developing an experimental plugin that uses Floresta as a lightweight on-chain backend for BTCPay Server. His work has covered wallet setup and recovery, address and transaction management, persistent metadata, and compatibility with the behavior BTCPay expects from its blockchain backend.

He also contributed fixes upstream to Floresta as issues were identified during the integration. Although still experimental, the plugin could enable BTCPay deployments wrth lower storage and hardware requirements, making independently verified, self-hosted payment infrastructure more accessible in constrained environments.

On the lightning side, he contributed to **`BTCPayServer.Lightning`** and began developing Lightning Manager, a plugin intended to provide a consistent wallet management interface across implementations such as LND, Core Lightning, Phoenixd, and Blink.

He also continued maintaining the Decentralized Pix Plugin. His work has also created a valuable feedback loop between BTCPay Server and the projects it integrates. While developing the Floresta backend, he identified issues and contributed fixes upstream, helping test Floresta against the expectations of a mature production application.

The same approach guides Lightning Manager, which supports multiple backends while exposing only the capabilities available in each implementation. Beyond the individual features, this work helps make BTCPay Server less dependent on any single node or Lightning implementation and provides upstream projects with practical feedback from a widely used merchant platform. Beyond code, ThgO.O worked directly with communities and merchants using these tools.

At the Global Bitcoin Circular Economies Summit in El Zonte, he and R0ckstar Dev launched the Vouchers plugin and helped local merchants configure BTCPay Server for daily use. He later presented the new multisignature and permission features at BTCPay Day Prague.

Across wallet security, plugins, alternative node backends, Lightning, and merchant onboarding, his work has followed a consistent goal: lowering the barriers to self-custodial Bitcoin payments and expanding the range of environments in which BTCPay Server can be used securely.

GRANTEE REPORTS

Erick Cestari /bitcoinfuzz/Smite

Over the past year, Erick Cestari has focused on security research across Bitcoin and Lightning, combining differential fuzzing, fuzzing infrastructure, manual review, offensive testing, and protocol analysis. His grant was paused for four months while he worked as a security engineering intern at Lightning Labs, where he reviewed LND and related projects before returning to his Vinteum-supported work.

A major part of his work has taken place through bitcoinfuzz, where he is one of the maintainers. Erick added fuzz targets covering Lightning P2P message parsing, BOLT4 onion decoding, ElligatorSwift decoding, Schnorr signature verification, AES-256-CBC, and PSBT parsing, while also reviewing and maintaining contributions from others. During the period covered by the report, this work led to **15 bugs being added to bitcoinfuzz'strophy list** across projects including rust-lightning, rust-bitcoin, LND, lightning-kmp, bitcoin-kmp, libwally-core, lightning-onion, and secp256k1.

Several of these findings also demonstrated the value of differential fuzzing beyond conventional bug discovery. When two implementations disagree on the same input, the underlying issue may be an ambiguous protocol requirement rather than a simple implementation error. One investigation began with a BOLT11 signature verification mismatch in lightning-kmp, exposed a documentation gap in secp256k1, and ultimately resulted in a clarification to BOLT11.

Other findings led to fixes and specification improvements across Bitcoin and Lightning implementations. Erick also submitted responsible disclosures to projects including btcd, Core Lightning, Eclair, Floresta, LDK, LND, lightning-kmp, rust-bitcoin, libwally-core, and secp256k1. In parallel, he reopened discussion around onion-message spam in Lightning after identifying weaknesses in the current design, including very large permitted hop counts and an attribution problem that can arise when maliciously constructed paths alternate repeatedly between two peers.

Discussions around this work contributed to an onion v2 design agreed upon during the Vienna Agreement, separating the final payload from the header so hop limits can be reduced without reducing the payload available to the final recipient.

His protocol work also included BOLTs #1343, which introduced `option_onion_messages_only_channels`, allowing nodes to explicitly signal that they accept onion messages only from channel peers.

Beyond bitcoinfuzz, Erick has continued contributing to Smite, a project for snapshot fuzzing Lightning implementations.

His work spans message codecs, input minimization, intermediate representation tooling, and failure triage. He has also been developing BOLT4 onion support intended to extend Smite's coverage into the Sphinx layer itself.

Smite has become the subject of his undergraduate thesis, focused on how snapshot fuzzing can increase code coverage and uncover new vulnerabilities in Lightning implementations. Erick also expanded fuzzing infrastructure across several projects, including libwally-core, Stratum V2, Floresta, BDK, and rust-bitcoin. His work ranged from introducing new harnesses to improving corpus management and CI integration, helping turn previously discovered failures into persistent regression tests.

In rust-bitcoin, this approach was subsequently adopted through the project's qa-assets repository. Another area of work has been Fuzzamoto, which exercises full Bitcoin Core nodes rather than isolated libraries. Erick added support for Bitcoin Core's BIP324 encrypted P2P transport, allowing existing fuzzing scenarios to run over either the legacy or encrypted transport and extending coverage into code paths that had previously been absent from Fuzzamoto's testing.

Across these projects, Erick's work sits at the intersection of implementation security, protocol correctness, and reusable testing infrastructure. Individual findings can result in a bug fix, a specification clarification, or a new regression test, while improvements to fuzzing infrastructure make it possible to investigate the same classes of problems continuously across multiple implementations.

vinteum

GRANTEE REPORTS

Chris G. /Floresta

*Since receiving a Vinteum grant in April **2026,** Chris G. has focused on strengthening the technical and organizational foundations of Floresta. His work combines **software architecture, development planning, contributor coordination, and protocol research,** helping the project mature while preserving the flexibility needed to experiment with new approaches to Bitcoin validation.*

One of Chris' early areas of focus was bringing more structure to the project's development process. He helped establish clearer quarterly roadmaps, align contributors around shared priorities, and improve how broader technical goals are translated into concrete work. As Floresta expands across multiple areas, this structure helps the team balance immediate implementation work with longer-term architectural questions. He has also proposed improvements to **engineering practices around testing, automation, observability, and developer experience,** while actively contributing to architectural discussions and changes across the codebase.

This architectural work includes the modularization of Floresta's components and the development of its domain-model crate, with the goal of establishing clearer boundaries between subsystems and reducing unnecessary coupling. These architectural changes are closely related to **Floresta's potential as an embeddable Bitcoin backend.** The project is increasingly being designed not only as a standalone node, but as infrastructure that can provide blockchain validation and related services to wallets, applications, developer tools, and other software.

Cleaner internal interfaces and stronger separation of concerns make those integrations easier to build and maintain. Chris has also taken on broader maintainer responsibilities. As the project grows, concentrating architectural decisions, reviews, planning, and releases in a small number of contributors becomes a constraint.

By working on architectural proposals, mentoring contributors, guiding implementation decisions, and serving as one of the project's technical contacts, Chris is helping distribute ownership more broadly across the team. In parallel, Chris has been developing an independent reference implementation of Utreexo in Lisp. Utreexo is the cryptographic accumulator used by Floresta to reduce the state required for Bitcoin validation.

Reimplementing it independently provides a way to study its fundamental operations without inheriting the abstractions and design decisions of the existing production implementation. The work has both practical and research value. It gives Chris a deeper understanding of Utreexo's data structures and algorithms, while providing a compact implementation that can help expose assumptions, clarify aspects of the protocol, and support future work on stronger validation or formal reasoning.

Implementing Utreexo independently in Lisp also provides a different way of expressing and reasoning about the protocol, making it easier to compare design choices across implementations and separate the protocol itself from the abstractions of any particular codebase. This research complements Floresta's growing use as an environment for protocol experimentation. Alongside Utreexo, the project is increasingly being used to explore synchronization techniques and alternative approaches to node architecture.

Chris' work emphasizes that this flexibility needs to be supported by **clear interfaces, meaningful tests, useful diagnostics, and engineering practices that make experimental assumptions and tradeoffs visible.** Although his grant began only in April 2026, Chris is already contributing across several areas of the project: **architecture, maintainership, development processes, and protocol research.** Together, this work is helping Floresta evolve from a small experimental implementation into a more structured and reusable Bitcoin project, while preserving the openness to experimentation that has been central to its development.

vinteum

GRANTEE REPORTS

Johnny Santos /P2Pool V2

*Johnny Santos joined Vinteum in March **2026** to work on **P2Poolv2** and help strengthen the open-source Bitcoin mining software stack. The project had been maintained largely by Jungly, and Johnny's role has quickly expanded from isolated contributions into a broader effort to make **P2Poolv2** easier to run, test, debug, and extend.*

*His first major project grew out of a Bitcoin Dev Launchpad proof of concept: **implementing high-bandwidth Compact Block Relay for P2Poolv2.***

Adapting BIP 152 to the project's network architecture proved more complex than initially expected, requiring work across synchronization, message flow, edge cases, and block reconstruction. The implementation is now close to completion and has also helped expose assumptions elsewhere in the node's sync logic. That work led Johnny to focus more heavily on **testing P2Poolv2 under realistic network conditions.**

The team began running nodes on testnet4 and found numerous issues involving header synchronization, RocksDB configuration, block confirmation, buffers, parallelization, disconnected headers, and libp2p behavior. Johnny also deployed a persistent P2Poolv2 node on Vinteum's infrastructure, creating a longer-running environment for observing failures and validating fixes. The result has been substantial improvement in initial synchronization.

In Johnny's current tests, a new node can reach roughly height 20,000 in around eight minutes while using less than 80 MiB of RAM and a small share of two virtual CPU cores. He has also contributed deployment and CI improvements, including prebuilt container images and testing across multiple architectures, lowering the barrier to running P2Poolv2 in different environments.

Another emerging area is **Bitcoin Core's experimental Cap'n Proto interface.** To experiment with it from P2Poolv2, Johnny created bitcoin-capnp, derived from Plebhash's sv2-bitcoin-core work, and is exploring an actor-compatible client that fits P2Poolv2's runtime model. Alongside this, Johnny has contributed testing and debugging around Hydrapool and P2Poolv2's Stratum server, including preparation for workloads involving large numbers of simultaneous miner connections.

His work is increasingly moving from individual features toward **understanding and improving P2Poolv2 as a complete system,** including synchronization, block relay, Bitcoin Core integration, deployment, and the infrastructure needed for other contributors to work on the project effectively.

GRANTEE REPORTS

Leonardo Lima /Bitcoin Dev Kit

*Over the past year, Leonardo Lima has consolidated his role as a maintainer across the Bitcoin Dev Kit ecosystem. Supported by Vinteum since December **2023**, he narrowed his focus entirely to BDK at the end of **2024** and became a maintainer in early **2025**.*

He is now lead maintainer of bdk_esplora, rust-esplora-client, and rust-electrum-client, and secondary maintainer across several of BDK's core wallet and chain libraries.

One of his most significant technical efforts this year was a **breaking refactor of `bdk_chain` canonicalization,** now merged. Canonicalization determines how a wallet resolves conflicting or replaced transaction histories into a consistent view of the chain, making it a correctness-critical part of BDK. The previous design tightly coupled canonicalization to synchronous data sources and their error handling, making asynchronous and fallible backends difficult to integrate.

Leonardo redesigned the process as a two-phase, sans-IO pipeline and introduced a generic ChainQuery interface in bdk_core, removing the old ChainOracle abstraction. **Canonicalization can now operate independently of whether chain data comes from a local source, full node, light client, synchronous API, or asynchronous backend.** The design went through several rounds of review with other maintainers before being merged.

Alongside this work, Leonardo has been contributing to **removing signing responsibilities from bdk_wallet,** with the goal of allowing wallets to avoid holding private key material directly. Delegating signing to rust-bitcoin also required contributions to rust-miniscript so descriptor-derived keys could work with its signing APIs. He has also introduced structure-aware fuzzing for BDK.

Wallet fuzzing is challenging because meaningful states involve structured combinations of transaction graphs, chain data, and descriptors rather than arbitrary byte inputs. Leonardo has been developing fuzz targets that generate valid but adversarial wallet states, work he presented at Bitcoin+ + Exploits Edition in Florianópolis. His responsibilities now extend well beyond individual contributions.

As a BDK maintainer, Leonardo triages issues and pull requests, reviews and merges changes by rough consensus, publishes releases, guides contributors, and participates in API and architecture decisions across the ecosystem. His current research builds directly on the canonicalization refactor. Most BDK wallets today access blockchain data through Electrum or Esplora servers, exposing information about the addresses they query.

Leonardo is exploring **new chain sources based on Bitcoin Core's bitcoinkernel library and multiprocess IPC interfaces,** which could allow wallets to validate against a full node controlled by the user. The sans-IO architecture in bdk_chain makes these alternative backends easier to integrate. He is also exploring a more UTXO-oriented model for **`bdk_chain`**.

While the current architecture is designed around full transaction histories, applications such as Lightning and Ark may need only particular UTXOs and transactions, potentially allowing for a lighter model. A growing part of Leonardo's impact comes through **mentorship and project stewardship.** Through the BDK Foundation mentorship program, he mentored Peter Tyonum, who later became lead maintainer of **`bdk-cli`** and **`bdk_testenv`**, and he currently mentors Vinteum fellow Yan.

He also runs office hours and study sessions through Vinteum, helping newer contributors understand BDK and enter the project with more context.

vinteum

GRANTEE REPORTS

Pins /LND

Now in his second year as a Vinteum grantee, Pins has continued to deepen his work across LND and the broader Lightning protocol ecosystem. His contributions now include ***30 commits merged into LND, three into btcd, and two into the BOLTs specification,*** *with another* ***25*** *LND commits tagged for inclusion in an upcoming release.*

Together, this work spans protocol behavior, implementation improvements, testing infrastructure, and maintenance of one of the most widely deployed Lightning implementations.

During 2026, his primary technical focus shifted toward coverage-guided fuzz testing for LND. Pins developed fuzzing targets for the Channel Link and Funding Manager state machines, both of which coordinate complex channel behavior across multiple states, messages, and failure conditions.

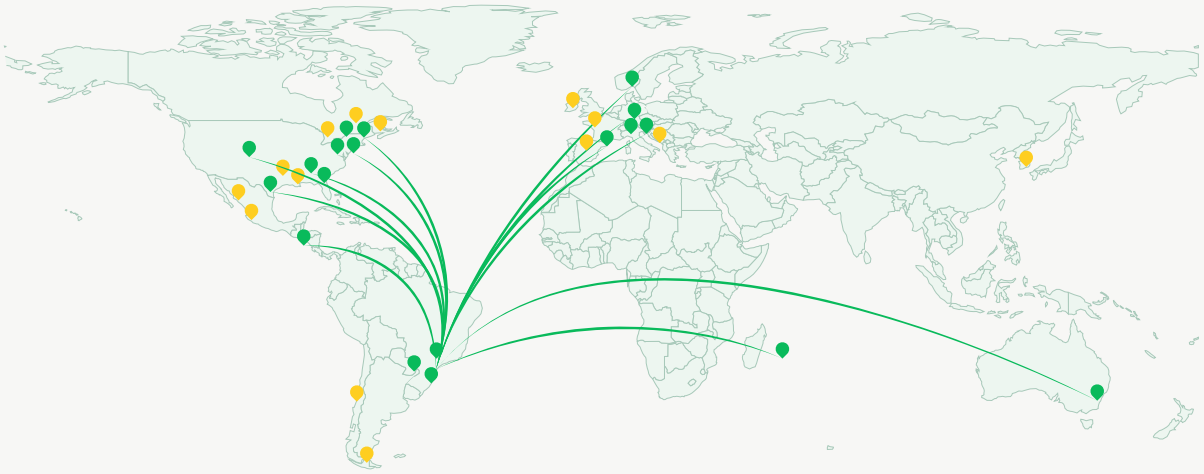
His next objective is to extend this work to the cooperative and force-close protocols. These targets are intended to exercise execution paths and state transitions that are difficult to cover through conventional tests alone. By generating varied and unexpected inputs, they can help uncover crashes, inconsistent state handling, and other implementation defects before they affect users.

The work also expands LND's reusable testing infrastructure, making it easier to investigate and reproduce failures in complex protocol components. Alongside implementation work, Pins remains active in technical education and contributor development. He organizes collaborative reading sessions around advanced technical material and hosts weekly open working sessions on Vinteum's Discord, where contributors discuss ideas, work through implementation problems, and collaborate on Lightning development.

These activities continue the educational work described in our previous report, but are increasingly connected to his role as an experienced project contributor. **By helping newer developers understand the LND codebase, interpret technical feedback, and work through problems collectively, Pins contributes not only through his own patches but also by making it easier for others to participate effectively.**

As Lightning implementations take on more functionality and support increasingly complex protocol behavior, systematic testing and sustained maintenance become more important. Pins' work this year reflects that shift, combining **protocol contribution, fuzz testing, implementation maintenance, and contributor support** to strengthen LND and the developer community around it.

Connecting Brazilian Contributors to the Global Bitcoin Ecosystem



- Atlanta
- Austin
- Nashville
- São Paulo
- Sydney
- Frankfurt
- Barcelona
- Boston
- Vienna
- Berlin
- Amsterdam
- Laramie
- Washington DC
- Toronto
- Florianópolis
- Prague
- Oslo
- New York
- Asunción
- Port Louis
- San Salvador

Cities visited for conferences, workshops, residencies, or in-person collaboration.

- Toronto
- San Francisco
- Seoul
- London
- Madrid
- Austin
- Atlanta
- Vienna
- Buenos Aires
- Brescia
- Santiago
- Amsterdam
- Boston
- New York
- Houston

Cities represented by guests invited to Vinteum programs and events.

Bitcoin open-source development is global, but much of the work happens asynchronously and at a distance. For contributors still developing technical context, project relationships, and reputation, ***spending time in person with maintainers, mentors, and peers can accelerate integration into the ecosystem in ways that are difficult to reproduce online.*** *Vinteum treats international travel and in-person collaboration as part of its contributor-support strategy.*

This includes helping Brazilian developers attend technical conferences and working gatherings abroad, bringing international contributors to Brazil, and supporting smaller retreats where teams can spend concentrated time reviewing code, discussing architecture, and transferring context. During the past year, this took several forms. For TABConf **2025,** in Atlanta, Vinteum rented a shared house for twelve members of the community, reducing participation costs to airfare and local expenses for most attendees. The group included winners of the previous Satsconf hackathon, whose flights were also covered by Vinteum. Davidson Souza presented Floresta's work on mobile, while Pins led a workshop on **building Taproot Lightning channels from scratch.** Davidson, Leonardo Lima, and Luis Schwab also attended the Rust Bitcoin Summit in Nashville.

In December, seven members of the Vinteum community traveled to Mauritius for the ₿trust Gathering, ₿trust Dev Day, and Africa Bitcoin Conference. At ₿trust Dev Day, Plebhash presented work on **CPU mining and sovereign block templates with Stratum V2,** while Bruno Garcia presented bitcoinfuzz and differential fuzzing across Bitcoin projects. The trip also created an opportunity to deepen collaboration among ₿trust, Vinteum, B4OS, and Bitshala. The teams exchanged technical ideas, discussed potential joint work, and compared approaches to contributor development, strengthening relationships among organizations supporting Bitcoin open-source contributors across different regions.

Some of the most valuable trips were tied directly to project collaboration. Leonardo spent approximately one month in Sydney working intensively with long-time BDK contributors Evan Lin and Lloyd Fournier on bdk_chain. Bruno Garcia participated in two Bitcoin Core developer retreats, in Frankfurtt and Barcelona, spending concentrated time with other Core contributors on technical discussion, review, and project coordination.

Cindy spent a week in Boston working directly with Armin Sabouri on wallet privacy and protocol research, while also participating in the MIT Bitcoin Expo. Erick Cestari and Pins also joined the annual Lightning specification meeting in Vienna, a recurring gathering of protocol developers from LND, Eclair, Core Lightning, and LDK to coordinate interoperability and discuss changes to the shared Lightning specification across implementations.

Erick Cestari, Jãonoctus, Luis Schwab, and Davidson traveled to Berlin for Lightning++, where Davidson presented a study of the specification changes required to integrate Utreexo with Lightning and Erick presented his work on differential fuzzing of Lightning implementations. **Erick gave that presentation less than a year after entering Vinteum's first Bitcoin Dev Launchpad cohort.** In Amsterdam, Bruno Garcia, K-YU, Davidson, Moisés Pompílio, Luis Schwab, and Chris G. joined a Floresta offsite followed by the Consensus Code Week, both hosted by 2140.

The gathering brought together developers working across consensus-sensitive implementations and interfaces, including Floresta, libbitcoin, btcd, and Core/bitcoinkernel, giving Vinteum contributors an opportunity to compare approaches directly with developers working on related validation and consensus problems. Vinteum's international exchanges also extend into research and protocol-focused gatherings.

Edil Medeiros participated in the University of Wyoming Bitcoin Research Institute Workshop in Laramie, DCI Global activities, and the Bitcoin Education Institute Annual Conference in Washington, DC, while also joining Bitcoin++ Consensus Edition in Toronto to engage more directly with developers and researchers working on consensus and protocol questions.

> *Vinteum has also increasingly* ***brought this exchange in the other direction.*** *During the second BDL residency in Florianópolis, Librería de Satoshi's B4OS residency was taking place in the same city. Working with Librería CEO Dulce Villarreal, Vinteum coordinated two days of overlapping programming so participants from both programs could study, discuss technical topics, and spend time together rather than remaining in separate cohorts.*

Many alumni from the first BDL cohort also returned for the residency and remained for **Bitcoin++ Exploits Edition,** where Vinteum was a major supporter. This gave participants additional exposure to international developers working on Bitcoin security, protocol development, wallets, and testing, including Niklas Gögge, Eugene Siegel, Sjors Provoost, Max Hillebrand, niftynei, Matthew Zipkin, Daniela Brozzoni, and others. D++ also spent two full weeks in São Paulo mentoring participants in Hack4Freedom, providing sustained hands-on support rather than a single workshop or conference appearance.

In parallel, Vinteum now brings selected grantees and fellows to São Paulo each month for **intensive in-person deep dives** focused on the technical areas they are currently working on. Participation varies depending on the subject, with sessions led by Edil Medeiros, Leonardo Lima, and Johnny Santos. The format allows contributors working on related problems to spend concentrated time reading specifications and code, reviewing one another's work, and developing shared context that would be harder to build through regular online meetings alone.

Alongside these exchanges, Vinteum has organized focused developer retreats around **Floresta, Stratum V2, DIY hardware signers, and Bitcoin security and fuzzing.** These are small working gatherings designed around code review, architecture discussions, knowledge transfer, and sustained collaboration rather than conference-style programming. A detailed account of these retreats and their technical outcomes is available on the Vinteum blog.

The Floresta team alone participated in two such offsites during the year, including the Casa21 retreat that brought contributors together with Utreexo researchers and the later Amsterdam gathering.

Other contributors also used international events to deepen their relationships across the ecosystem. ThgO.O attended Plan B Forum in El Salvador and BTC Prague, while Vinteum's Executive Director participated in events including the Oslo Freedom Forum, HRF's Global Bitcoin Summit, TABConf, Acelerando Bitcoin, Africa Bitcoin Conference, OP_NEXT in New York, and the MIT Bitcoin Expo.

He also participated in a closed Chaincode gathering designed to bring protocol developers and application developers together to discuss how Lightning behaves in real-world deployments and which technical and product constraints continue to limit adoption. **In many cases, the most valuable part of the trip was not the event itself, but the concentrated time contributors spent working directly with people they normally interact with only online.**

These exchanges help contributors gain technical context, become known to the people reviewing their work, encounter new directions, and form relationships that can lead to deeper collaboration, grants, employment, or greater responsibility within projects. They also bring knowledge back into the Brazilian community, where it can spread through Casa21, BitDevs, fellowships, study groups, mentorship, and future contributors. The objective is not conference attendance for its own sake, but to make the Brazilian Bitcoin developer community **more connected to the global networks of people building, reviewing, researching, and maintaining Bitcoin open-source infrastructure.**

Building a Bitcoin Research Pipeline

Vinteum's research strategy is increasingly focused on ***connecting open-source development with more rigorous technical research.*** *Part of this work begins inside our own programs. Some contributors arrive primarily as software developers and gradually move toward research as implementation exposes harder questions.*

Others show an early aptitude for cryptography, privacy analysis, protocol design, measurement, or technical investigation. The objective is to help them move beyond implementing predefined tasks toward **formulating questions, evaluating evidence, studying prior work, and communicating results clearly.** This does not mean creating a research program separated from open-source development.

We want research questions to remain connected to the systems and communities where their conclusions matter. Depending on the problem, the relevant output may be a paper, prototype, benchmark, dataset, independent implementation, technical report, or change to production software. **Building a research pipeline means creating pathways through which questions, people, methods, code, and institutions can find one another, not only funding isolated projects.**

Developing researchers internally is only part of the strategy. Vinteum has also been building relationships with academics already studying subjects relevant to Bitcoin, particularly when their work intersects with questions faced by open-source contributors. The goal is to create more sustained dialogue between researchers and the people building and maintaining Bitcoin software, giving researchers greater exposure to implementation constraints and unresolved technical problems while giving developers access to deeper theoretical expertise and research methods.

During the past year, this included participation in the **University of Wyoming Bitcoin Research Institute Workshop,** continued engagement with the **MIT Digital Currency Initiative through DCI Global,** and participation in the **Bitcoin Education Institute Annual Conference.** We have also begun or continued conversations with researchers connected to institutions including the Karlsruhe Institute of Technology, the University of Lisbon, Masaryk University, and the Autonomous University of Barcelona.

In Brazil, Vinteum has held preliminary conversations with the University of São Paulo about a potential initiative supporting master's and doctoral students researching **post-quantum cryptography and its implications for Bitcoin.** The discussions remain at an early stage, and Vinteum is currently raising funds to make the partnership possible. If successful, the initiative could serve as a foundation for a broader national effort supporting Bitcoin-related graduate research across Brazilian universities.

The connection between research and implementation is also becoming more visible within Vinteum's own engineering work. Bruno Garcia is currently pursuing a doctorate and has had research papers accepted both independently and in collaboration with Erick Cestari in the area of Bitcoin fuzzing. This illustrates the kind of overlap Vinteum wants to encourage: **research informed by direct experience with production code, and engineering strengthened by more rigorous methods of testing, measurement, and evaluation.**

Funding Infrastructure

Funding open-source development means more than funding developers. Contributors also need infrastructure to run nodes, test implementations, fuzz code, benchmark changes, host development services, and experiment without being constrained by the hardware available to them personally. Over the past year, Vinteum began investing more directly in this layer as well, both by building shared infrastructure at ***Casa21*** *and by covering dedicated server costs for grantees when their work requires it.*

At Casa21, we installed a Dell PowerEdge R730 with 56 CPU cores, 128 GB of RAM, eight 1.2 TB 10K SAS drives, and an additional 1 TB SSD. Rather than dedicating the machine to a single project, we use virtualization to make computing resources available across different contributors and workloads. The server is already supporting development and experimentation across several parts of the Bitcoin ecosystem.

Contributors use it to run Bitcoin Core and Floresta nodes, Utreexo infrastructure, P2Poolv2 on testnet4, and Stratum V2. It also provides environments for fuzzing Floresta and Rustreexo, testing Bitcoin Core pull requests on dedicated nodes, running Payjoin infrastructure, and other development workloads. We also recently received a donated Dell PowerEdge R710 with two Xeon processors, 64 GB of RAM, and 300 GB of storage.

Unlike our shared server, this machine will be dedicated as a bare-metal server for Fuzzamoto, providing additional infrastructure for continuous fuzz testing of Bitcoin software. Not every workload makes sense on the infrastructure we operate ourselves. For contributors who need dedicated resources, we also cover external infrastructure costs when they are directly related to their open-source work.

This includes dedicated bare-metal servers and other compute resources used for development, testing, benchmarking, and fuzzing. Together, these resources give fellows and grantees a place to run longer-running or resource-intensive experiments without relying entirely on their personal machines. They also make Casa21 more than a physical place to work and meet: it is becoming a shared technical infrastructure for Bitcoin open-source development.

These expenses are small compared with the cost of funding a developer, but they can significantly expand what that developer is able to do. Our goal is straightforward: if a contributor we support needs computing infrastructure to test an idea, investigate a performance problem, continuously fuzz a codebase, or reproduce an issue, access to hardware should not be the bottleneck.

We are also happy to receive hardware donations to expand this infrastructure. Servers, memory, SSDs, hard drives, and other useful computing hardware can directly increase the resources available to Bitcoin open-source contributors working through Vinteum and Casa21. If you or your company have hardware that could be put to work supporting Bitcoin development, get in touch.

Casa21 and the Developer Community

A significant part of the organization's role is creating the **physical and social infrastructure where contributors can meet, study, collaborate, and remain connected to the wider Bitcoin ecosystem. Casa21** has become the physical center of that infrastructure in São Paulo. It serves as a workspace for fellows and grantees, a venue for technical events, and a meeting point for developers, researchers, students, and community organizers. Contributors use the space for regular work, project meetings, technical discussions, and in-person deep dives where fellows and grantees review proposals, study technical material, compare progress, and work through difficult implementation or research questions alongside peers and mentors.

These interactions are especially valuable for people entering open source. Many practical norms of contribution are difficult to teach through coursework alone. Learning how to ask useful questions, review someone else's work, discuss uncertainty, respond to criticism, or recognize that an idea is not yet ready often happens through repeated exposure to experienced contributors.

vinteum

CASA21 AND THE DEVELOPER COMMUNITY

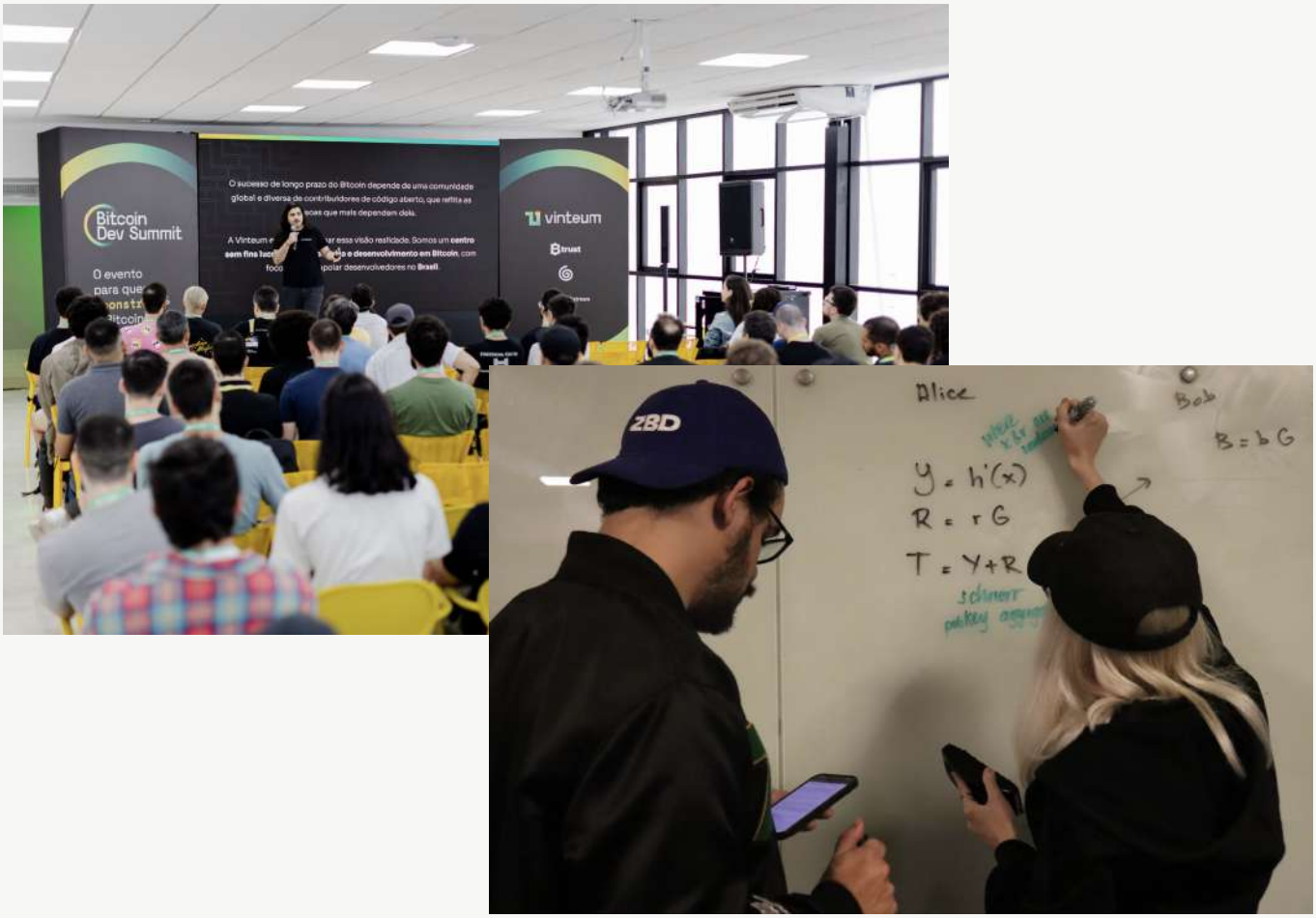
Much of this community also operates online through Vinteum's Discord. Recurring activities include technical study groups, collaborative reading sessions, project-specific discussions, English practice, open working sessions, and regular office hours hosted by grantees and experienced contributors. The Discord server also hosts **Show and Tell,** a weekly community check-in where contributors share what they worked on, discuss blockers, and outline what they plan to tackle next. Most grantees and fellows also hold weekly **Office Hours,** working in public and welcoming interruptions that can turn into spontaneous technical discussions and collaboration. Together, these formats create regular accountability while giving fellows, grantees, and other contributors greater visibility into one another's work and more opportunities for peer feedback and serendipitous exchange. Together, **Casa21** and the online community provide entry points outside formal selection processes. Someone may first join a discussion, follow a study group, begin attending office hours, and only later decide to contribute to a project or apply for structured support.

Vinteum also supports larger gatherings that bring the Brazilian developer community into closer contact **with international contributors and one another.** Vinteum was a major partner of Satsconf and the broader São Paulo Bitcoin Week, creating additional technical programming around the concentration of developers already gathered for the conference. This included the **Bitcoin Dev Summit,** a dedicated gathering for open-source contributors.

Vinteum was also a major supporter of **Bitcoin++ Exploits Edition in Florianópolis,** which brought developers focused on Bitcoin security, testing, wallets, and protocol development into direct contact with participants and alumni from Vinteum's programs. Beyond these larger gatherings, this work increasingly extends across Brazil through **locally organized BitDevs groups.** Recurring meetings now take place in São Paulo, Florianópolis, Curitiba, Belo Horizonte, Brasília, Porto Alegre, Uberlândia, São Carlos, and Ribeirão Preto, with the latter two alternating monthly as part of BitDevs Interior. These communities maintain a shared focus on substantive technical discussion around BIPs, pull requests, research papers, Lightning, mining, wallets, privacy, and other areas of Bitcoin infrastructure, usually through a Socratic format.

vinteum

CASA21 AND THE DEVELOPER COMMUNITY



In a country as large as Brazil, they allow students and developers to participate in serious technical conversations without first relocating to São Paulo or entering a Vinteum program. Vinteum supports organizers, connects groups with contributors, and helps circulate expertise and opportunities between cities. **The long-term objective is for these communities to develop their own continuity and local leadership while remaining connected to a broader national network.** Universities are another important entry point. Through student-focused events, workshops, relationships with professors, and student groups, Vinteum is working to make Bitcoin development and research more visible to people studying computer science, engineering, mathematics, and related fields. **Bitcoin Students Day** has already brought together students from USP, Inteli, Unicamp, UFMG, PUC Minas, and COTEMIG, introducing them to Bitcoin's technical foundations, open-source projects, and paths into contribution.

Vinteum is now exploring expansion to additional universities and regions, including Brasília and Brazil's Northeast, working with local students and academic partners to adapt the initiative to each community. Other initiatives have created entry points for groups that remain underrepresented in Bitcoin development. Vinteum sponsored and hosted Evento's **Hack4Freedom** at **Casa21,** supporting an independent initiative that brought women from technical and educational communities together for workshops, mentoring, and a hands-on hackathon involving Lightning, Nostr, privacy, AI, and open-source development. **A healthy developer ecosystem needs more than a funding pipeline.** It needs places where people can encounter the work, form relationships, test their interest, and continue participating even when they are not inside a formal program. **Casa21,** Discord, BitDevs, universities, and open technical events help create that broader environment around Bitcoin development in Brazil.

Our Funders and Supporters

Everything described in this report depends on funders willing to support work across different timelines. Their support allows Vinteum to sustain established contributors, provide fellowships to emerging developers and researchers, operate ***Casa21,*** *develop educational programs, and invest in technical work whose importance may only become fully visible over time. We are especially grateful to* ***Xapo Bitcoin Limited,*** *whose support covers Vinteum's core operating costs and provides the institutional foundation behind much of the work described in this report.*

This stability allows Vinteum to maintain its team and infrastructure, provide continuity between grants, and respond to opportunities or needs that are difficult to predict in advance. We are also deeply grateful to **₿trust and OpenSats** for supporting Vinteum's educational and fellowship programs, allowing participants to move from structured technical education into full-time open-source work with the time, mentorship, and financial support needed to develop as contributors.

Additional donations from individuals and organizations support contributors, programs, events, and technical areas throughout the year, creating flexibility to bridge funding gaps and act on opportunities outside the organization's annual planning cycle. We are equally grateful to the maintainers, mentors, researchers, universities, community organizers, companies, and partner organizations that contribute time, review, knowledge, introductions, infrastructure, and opportunities. Much of Vinteum's work depends on forms of support that are not financial but are no less important.

We are deeply grateful to everyone who has trusted Vinteum to do this work. The progress described throughout this report would not be possible without that trust.

Looking Ahead

Year Four made Vinteum's next challenge clearer. The organization has shown that it can help people move from technical education into meaningful Bitcoin open-source work, but the intensive support behind that progress is difficult to scale. This constraint does not come only from Vinteum's own mentorship capacity.

It also depends on how many projects are ready to absorb new contributors, how much reviewer bandwidth experienced maintainers have available, and whether there are enough meaningful problems that can responsibly be handed to people who are still developing independence. As more developer education programs succeed in bringing people into Bitcoin open source, this capacity can become saturated.

This will require more deliberate decisions about when to launch another cohort, how many participants can be supported well, and which parts of the program should remain intentionally small and intensive. **We do not want to maximize the number of people entering the pipeline, but to increase the number who eventually become independent, trusted, and sustained contributors.**

> *For current fellows, the coming period will be an important test. Many have made meaningful progress but are still developing the technical judgment, project relationships, and independence required for long-term contribution. Vinteum will continue helping them deepen their work, integrate more closely with maintainers, and identify sustainable paths through grants, employment, or continued project support.*

At the same time, Vinteum wants to create a **more continuous entry point into its developer programs.** Seminars, study groups, Discord activities, university initiatives, BitDevs, and Casa21 can allow people to develop at different speeds and become visible to mentors throughout the year, rather than concentrating discovery and selection into occasional cohorts. Initiatives such as the Mastering Bitcoin and Mastering Lightning seminars provide a foundation for this model.

Vinteum intends to explore structures in which experienced community members, alumni, volunteers, and potentially dedicated education fellows can facilitate recurring technical activities using shared materials and clearer processes. This would allow the community to support more early-stage learning without requiring Vinteum's leadership team to operate every session directly.

Vinteum also plans to expand Bitcoin Students Day to additional universities and regions, continue supporting locally led BitDevs communities, and is exploring a move to a **larger Casa21** capable of accommodating more contributors, technical gatherings, parallel activities, and potentially short-term accommodation for developers visiting São Paulo. Stronger operational leadership should make these ambitions more sustainable by allowing more organizational responsibility to be delegated while Vinteum's leadership focuses on fundraising, contributor development, research relationships, and allocating resources across the ecosystem.

Vinteum

VINTEUM YEAR FOUR REPORT

The central challenge for the coming year is not expansion alone, but consolidation with direction.

We know what we are building toward at Vinteum. The task now is to be deliberate about the paths that can get us there: which programs should grow, which should remain deliberately small, which contributors are ready for greater independence, and where limited funding and mentorship can create the greatest long-term value.

The work ahead is to turn momentum into continuity. The measure of success next year will not be how much we do, but how durable our work becomes.

vinteum