UNITED STATES DISTRICT COURT
SOUTHERN DISTRICT OF NEW YORK

SECURITIES AND EXCHANGE COMMISSION,	:	
	:	
Plaintiff,	:	
	:	
v.	:	**19 Civ. 09439 (PKC)**
	:	
TELEGRAM GROUP INC. and TON ISSUER INC.,	:	
	:	
Defendants.	:	

BRIEF OF *AMICUS CURIAE* THE TON COMMUNITY FOUNDATION IN SUPPORT OF TELEGRAM GROUP INC. AND TON ISSUER INC.

February 14, 2020

Dimitry Joffe (DJ-6498)
JOFFE LAW P.C.
765 Amsterdam Avenue, 2C
New York, New York 10025
(917) 929-1964
Dimitry@joffe.law

Counsel to Amicus Curiae
the TON Community Foundation

TABLE OF CONTENTS

The TON Community Foundation (the "Foundation"), through its undersigned counsel, respectfully submits its *amicus curiae* brief in support of Defendants Telegram Group Inc. and TON Issuer, Inc., in the above-captioned action (the "Amicus Brief"), attached as Exhibit A hereto.

INTEREST OF *AMICUS CURIAE*

The Foundation (https://ton-foundation.org/) is a non-profit association of Telegram Open Network ("TON") ecosystem participants -- top-notch computer scientists, programmers, developers, validators, founders, and employees of TON-based commercial and non-commercial projects. The main mission of the TON Community Foundation is to promote the fastest, most effective and equitable development of TON as a decentralized system based on collaboration and cooperation.

As an *Amicus Curiae* in this Action, the Foundation represents a professional community of active participants in the TON project in whose interest it is to see the TON Blockchain mainnet launched as soon as possible. Neither the Foundation nor its individual members are affiliates of the Defendants or any of them, nor did they or their counsel receive any assistance, compensation, or any promise of compensation from the Defendants, their counsel, or any other parties in connection with this Amicus Brief. None of the Defendants or their counsel has contributed to or reviewed this Amicus Brief prior to its filing.

THE TON COMMUNITY FOUNDATION

There are currently approximately 20 teams in the community developing different solutions based on TON. In total, according to the Foundation's data, in just six months of active TON testing, about 8,000 people participated in the groups created by the community members. TON community has about 2,000 active participants, mostly concentrated in chats and channels

inside the Telegram messenger, in particular in the chat rooms @tondev en, @tondev ru, @ton_research; around the channels @infoton, @tondev, @ton_overview, @ton_china, TON China Website: https://tonbus.com/, and others.[1]

The following members of the Foundation have provided factual information and participated in the drafting of this Brief: Dmitry (Mitja) Goroshevsky, Dmitry Shtukenberg and Leonid Kholodov of TON Venture Studio Ltd., d/b/a TON Labs.

The Brief has been unanimously approved for filing by the following founding members of the Foundation:

1. TON Labs (Mitja Goroshevsky aka @Futurizt);
2. Mercuryo (Viacheslav Akhmetov aka @akhme);
3. MixBytes (Sergey Boogerwooger aka @boogerwooger);
4. Adgram (Sergey Shashev aka @favoritefx);
5. EverStake/AtticLab (Sergii Vasylchuk aka @sergattic);
6. Prometeus Labs (Vladislavs Semjonovs aka @VladislavsSem);
7. AtomicWallet.io (Konstantin Gladych);
8. P2P.org (Eugene Koinov);
9. TON Center (Kirill Emelyanenko aka @rulon);
10. TON China (Tooz Wu aka @toozwu);
11. TON France (Philippe Rodriguez aka @Ph1lr0d);
12. Alexey Pryanishnikov aka @Priani4ek;
13. Combot/TON.sh (Sergey Chernikov aka @flugdreka);

[1] See, e.g., https://t.me/tondev_en https://t.me/tondev_ru, https://t.me/ton_research, https://t.me/TONgramDev, https://t.me/tonkhasia.

14. Copperbits (Petr Korolev aka @skywinder);

15. Credentia LLC (Stepan Gershuni aka @sgershuni);

16. Alexandr Vat aka @Vatic;

17. Combot (Fedor Skuratov aka @teodorix);

18. Button Wallet (Nick Kozlov aka @enormousrage);

19. Chrono.tech (Mikhail Savchenko aka @mikefluff);

20. TON Spain (Daniel Perez).

SUMMARY OF THE ARGUMENT

The purpose of this Amicus Brief is to state the position of the community of TON Blockchain developers that the TON Blockchain is fully operational, with a state-of-the-art prelaunch security and a developed suit of services, and is ready to be launched as a mainnet in a matter of seconds. This is an unqualified unanimous opinion of the community of independent specialists with extensive blockchain experience who are involved in the actual work on the TON Blockchain and who write its code, protocol, smart contracts, tools, and applications.

This position of TON Blockchain developers is in fundamental conflict with the opinions expressed by the SEC's expert Professor Maurice P. Herlihy in his report dated December 27, 2019 (the "Herlihy Report"), in which Prof. Herlihy opines that (a) the TON Blockchain code lacks critical components necessary for the launch; (b) its security is not a proven 100%; and (c) services and applications that will eventually be purchased by Gram holders do not exist, and/or the TON Blockchain is not mature enough to support them. Herlihy Report ¶ 8.

The Foundation believes that it would assist this Court to hear the unanimous view of its 20 member companies representing over 2000 computer scientists, engineers, programmers, and entrepreneurs who have been actually building this cutting-edge technology with their hands and

in whose direct interests it is for the TON Blockchain to become wildly successful, as they all expect it to be. This Brief thus presents, on behalf of the community of TON Blockchain developers unaffiliated with Telegram, unique factual information and perspective that can assist the Court, beyond the help that the parties or their attorneys are able to provide, by explaining the current state of the TON Blockchain and testing Professor Herlihy's assumptions, factual assertions, and ultimate conclusions.

The fundamental problem with Prof. Herlihy's approach is that none of the Bitcoin, Ethereum or Tezos networks (to name just a few of the successful blockchains) would have survived his academic scrutiny – and would not have been launched had Prof. Herlihy had his say on the matter. Those blockchains did launch without asking Prof. Herlihy for his opinions, and become successful; today, if Prof. Herlihy's opinions are accepted in this Action, they would threaten the future of this innovative industry, for under Prof. Herlihy's regime no new bitcoins, ethereums, tezos, or TONs would be ever launched. The Court should decline the SEC's invitation to accept such an innovation-suffocating regime.

Sextus Empiricus, a classical philosopher circa 160-210 A.D., wrote in his treatise *Against the Professors*: "Those who talk should do and only those who do should talk." Too strong a statement, perhaps; but at the very least the Court should have the benefit of hearing from both sides – those who do and those who talk -- on this far-reaching issue.

ARGUMENT

1. The TON Blockchain is mischaracterized in Prof. Herlihy's Report.

As a threshold matter, a few basic definitional misconceptions about the TON Blockchain in the Herlihy Report call for correction.

First, the TON Blockchain is ***not*** "just a ledger; an append-only list of entries" as Prof. Herlihy defines "blockchain." Report ¶ 9. Prof. Herlihy uses the blockchain definition from 2010, which has since become obsolete (ten years is a long time for computing technologies).[2] Modern blockchains in general, and the TON Blockchain in particular, claim verifiable computation of smart contract execution among their core parameters.

Second, the TON Blockchain's smart contract is ***not*** "a script executed by the validators and recorded on the blockchain," as Prof. Herlihy defines it in his Report ¶ 11. This definition of a "smart contract" too is obsolete, borrowed from the bygone era, which obsolete definition Prof. Herlihy uses in an attempt to analyze a cutting-edge comprehensive technology such as the TON Blockchain. The TON Blockchain's smart contract is not a "script" because "scripts" do not have states and do not store data, whereas the TON Blockchain's smart contract does change states, and it stores data.

There are very limited use cases of scripts. Scripts are supported by Bitcoin for example, but Bitcoin does not support smart contracts. See https://en.bitcoin.it/wiki/Script. TON smart contracts, in contrast, are computationally universal, or Turing-complete.[3]

Third, Prof. Herlihy claims that "[t]he most common use of blockchain technology is to manage a cryptocurrency, or coin, a unit of accounting and store of value." Herlihy Report ¶ 9.

---

[2] See Back, Adam et al., *Enabling Blockchain Innovations with Pegged Sidechains*, White paper, Genius.com 2010) (defining blockchain as "a well-ordered collection of blocks, on which all users must (eventually) come to consensus. This determines the history of asset control and provides a computationally unforgeable time ordering for transactions."), available at https://genius.com/Adam-back-enabling-blockchain-innovations-with-pegged-sidechains-annotated.

[3] A system is called "computationally universal" if it can compute every function computable by systems in that class (or can simulate each of those systems). Typically, the term "computationally universal" is tacitly used with respect to a Turing-complete class of systems. See https://en.wikipedia.org/wiki/Turing_completeness.

In Ethereum and in other modern blockchains, smart contracts are often used to implement business logic in addition to their basic ledger function. What makes the TON Blockchain unique is that literally everything in its network is based on interaction with smart contracts. Technically speaking, the TON Blockchain is just a back-end distributed computing device that executes numerous programs performing arbitrary computations. Therefore, the use that Prof. Herlihy cites as the most common use of blockchain is not applicable to the TON Blockchain. Grams as well as other Currencies are managed in TON as the so-called Currency Collections. But all Grams will be located in smart contracts so in a way TON is the smart contract platform more than a cryptocurrency one.

Fourth, Prof. Herlihy claims that the TON Blockchain "collects multiple prior blockchain design ideas and combines them into one complex, unified structure." Report ¶ 18.

As a threshold matter, there is nothing wrong with combining prior ideas into a unified structure; indeed, it sounds like a great accomplishment. In any event, the same statement could be made about any other blockchain design (and about any invention for that matter). Every new technology, research or product is based on some prior efforts and ideas. The first blockchain developer took the prior research in cryptography and aggregated it into a single complex concept and, eventually, into a product (Bitcoin). And so does the latest blockchain technology, the TON Blockchain.

Moreover, as discussed above, the TON Blockchain has unique architecture in that it is all based on smart contracts. Unlike in any other blockchain, everything in TON is smart contracts. Without them even a simple transfer of system currency is not possible.

2. The TON Blockchain is fully operational and could be launched as the mainnet on a 5-second notice.

As of this writing, the TON Blockchain is fully operational and could be launched on a 5-second notice. There are two working testing networks of TON (testnets) right now. Any one of them could be turned into the Main network (mainnet) in a matter of 5 seconds, which is the approximate time of one block validation in TON. That is all the time it will take for delivering Grams from developers' wallets into the wallets of investors and users in one mass-transfer, and the network will become the TON Main network at that exact moment. And since most of the validators are in the hands of independent developers on these testnet networks already, this network will be immediately fully decentralized. In a sense, one could say that the TON Network has already launched.

Prof. Herlihy takes a contrary position in his Report, asserting that "[t]he publicly-released 'testnet' version of the TON Blockchain code, while complete enough to run simple transactions on simulated assets, lacks critical components that would be required in a fully developed and running system, such as i) the core 'BFT Consensus' protocol, ii) the code that controls validator selection, iii) code that controls validator incentives and rewards, iv) the code that controls how validator misbehavior is detected and deterred, and v) the 'vertical blockchain' mechanism for repairing corrupted blocks." Report ¶ 8. These assertions are incorrect; as shown below, the TON Blockchain does not lack any critical components required for it to function as a fully developed system.

With respect to Prof. Herlihy's points (i) through (iv), the TON Blockchain has a complete implementation of (i) the BFT consensus protocol; (ii) the code that controls the validator election process; (iii) the code that controls validator incentives and rewards, as well as (iv) the code that controls how validator misbehavior is detected and deterred.

On point (i), Prof. Herlihy admits in his Report that he was unable to reconstruct the TON Blockchain consensus protocol: "the public documents do not disclose the BFT consensus protocol executed by the validators, and that protocol is difficult to reconstruct from the validator code in the public release. Moreover, it is impossible to tell whether the code represents the final version of the protocol." (¶ 28). Prof. Herlihy then concludes that "[w]ithout a clear exposition of the BFT consensus protocol, along with a thorough security and a performance analysis, I cannot consider the TON Blockchain close to deployment." Report ¶ 28.

The fact that Prof. Herlihy was unable to reconstruct the consensus protocol and, therefore, could not verify that the code represents the final version of the protocol, speaks of the limitations of his analysis, but says nothing about the protocol itself. Indeed, one of the independent TON Blockchain developers (and a contributor to this Brief on behalf of the Foundation) TON Labs has fully reconstructed the consensus protocol from the available Node code-base, and published the process and its results at https://docs.ton.dev/86757ecb2/p/07ddda-walk-through-the-catchain. This document was created 3 months before Dr. Nikolai Durov published his version of the protocol overview at ton.org: https://test.ton.org/catchain.pdf.

The TON Blockchain consensus protocol is a variation of the Practical Byzantine Fault Tolerance ("PBFT") protocol, proposed by Miguel Castro and Barbara Liskov of the Laboratory for Computer Science, Massachusetts Institute of Technology, in the *Proceedings of the Third Symposium on Operating Systems Design and Implementation* (New Orleans, Feb. 1999), more than 20 years ago, in the pre-blockchain era (at http://pmg.csail.mit.edu/papers/osdi99.pdf). That protocol is well-studied, and all the complex ideas in it have been proven since its 1999 publication. See https://cs.nyu.edu/courses/fall18/CSCI-GA.3033-002/papers/pbft-proof.pdf.

Moreover, many of the newly introduced elements of the algorithm in TON have been formally proven in https://test.ton.org/catchain.pdf by Dr. Durov himself.

On points (ii)-(iii): the original code repository is available at https://github.com/ton-blockchain/ton. The validator election process is specified in TON whitepapers and implemented in the following smart contract: https://github.com/ton-blockchain/ton/blob/master/crypto/smartcont/elector-code.fc. In particular, it specifies the way how validators are rewarded for their work.

On point (iv) of Prof. Herlihy's Report: for Byzantine Agreement algorithms there is no need to reliably separate "good" and "bad" nodes. It is enough to know that some portion of nodes (2/3 in TON case) can be trusted -- but no need to know for sure which nodes are malicious. This property is required since otherwise a "traitor" (an absolutely trusted node, which at a critical moment behaves maliciously or ("become Byzantine") can ruin the blockchain. Thus, detection and punishment of malicious node may improve efficiency and economy of the blockchain, but is not critical for its functioning, so this fine-tuning of the algorithms may be easily postponed or abandoned entirely. See, e.g., fork detection code: https://github.com/ton-blockchain/ton/blob/master/catchain/catchain-receiver-source.cpp#L154; Consensus algorithm has a very strict messaging protocol, rather hard to violate. Also, there is a validator signature check code, see https://github.com/ton-blockchain/ton/blob/77842f9b637dd2efcd684a4668e9ff4a173449f8/catchain/catchain-receiver.cpp#L79. All messages are to be properly signed -- if the signature is invalid, the block is ignored.

On point (v) of the Herlihy Report ¶ 8: with respect to the "vertical blockchain" mechanism for repairing corrupted blocks, this mechanism is redundant and not critical to the

TON Blockchain performance, and it can be added at a much later stage of network operation, or not added at all. Indeed, the "vertical blockchain" mechanism does not exist in other blockchains; yet, other blockchains have been operating for years in a secure manner without it. In the TON Blockchain protocol, the "vertical blockchain" is also not necessary – it is an experimental feature that could reduce forks and provide an additional security layer post-launch.

Prof. Herlihy further states: "When a corrupted or incorrect block is detected, the public documents assert that the "vertical blockchain" mechanism is used to fix the error, creating a new block to shadow the old block, and potentially triggering cascading adjustments in later blocks. I was unable to find code for this functionality. I cannot be convinced of the TON Blockchain's security without a careful analysis of the cascading error-recovery code, since complex algorithms are required to unwind the error correctly and efficiently."

If the corrupted or incorrect block is detected, the malicious block or node is blocked and excluded from the consensus on the protocol level. There is no need for any corrections of the blocks afterward. As explained above, the "vertical blockchain" mechanism is just an additional and not essential part of the blockchain design, which maybe added post-launch, or never added at all.

Prof. Herlihy's other objections regarding the performance of the TON Blockchain's performance and scalability are likewise unfounded. Thus, Prof. Herlihy states that "[o]ur goal was to estimate the eventual performance and scalability of a full running TON Blockchain, but we were unable to make such an estimate because the current testnet release is so much smaller in scale (fewer nodes and transactions) than the anticipated full TON Blockchain. For example, only 36 validator nodes were detected, while the public documents project as many as 1000 validators when the system is fully deployed."

Prof. Herlihy either misread or misunderstood the TON Blockchain code, which provides for 36 as the minimum number of validators and for 1,000 as their maximum number. As long as the number of validators fluctuates within this range, their number does not impact the operational capabilities of the network. In the testnet version, the number of validators vary, reaching hundreds at times.

When Bitcoin network was launched, there was only one computer that ran it, that of Satoshi Nakamoto her/him/them-selves, with a small fallowing of very few enthusiasts for the whole of 2009 and most of 2010.[4] Ultimately, this parameter cannot be used to conclude, as Prof. Herlihy does, that the network is inoperable or cannot be launched.

Prof. Herlihy is also incorrect that "running a validator node is comparable to running a small data center: it will require substantial technical expertise configuring and maintaining servers, software, and network interfaces, and a long-term commitment to do so." Report ¶ 19. There are automated scripts deploying TON nodes on a simple server available to any developer. Almost anyone who knows how to code can run a validator node. Those scripts are open-sourced and are available, for example, on GitHub, see https://github.com/Kiku-Reise/TON and here: http://github.com/koinov

Prof. Herlihy's lament that "it is impossible to predict from observing the testnet release whether the TON Blockchain will meet its performance and scalability goals" is irrelevant as the

[4] Most of the blocks mined in 2009 have very few transactions in them. The majority of them just include a single coinbase transaction, which is the required transaction encoding payout of the block reward to the miner. Coupled with the anemic hash rate, we can speculate that there were very few users of Bitcoin in 2009. It is known that a few enthusiasts like Hal Finney (the first user of Bitcoin) downloaded and ran Satoshi's code. Hal previously stated that he actually mined one of the first 100 blocks. But there is no evidence that any of these early adopters, including Hal Finney, did much more than run the code for a short time before losing interest, see https://eklitzke.org/how-many-bitcoins-did-satoshi-nakamoto-mine.

same could be said about any system or software. Many innovative products (if not all) are launched without many features that make them great and robust in the end. However, the lack of such features pre-launch that does not predict the product's post-launch success or failure.

As shown above, Prof. Herlihy admittedly could not understand the TON Blockchain protocol. The people who do – the developers represented by the Foundation – do not try to predict if TON will meet its performance and scalability goals simply because it is impossible to predict for this or any other system or software. But the TON Blockchain is not different in this respect from any other successfully launched blockchain, and nothing in the Herlihy Report shows otherwise.

The above, in sum, shows that the TON Blockchain has been reviewed and tested by independent teams such as TON Labs and other community members no less experienced in blockchain than Prof. Herlihy. Prof. Herlihy, by contrast, has failed to reconstruct the Blockchain protocol, misinterpreted its code, and overlooked its available suit of services – all of which cast considerable doubt on whether Prof. Herlihy has had an adequate factual basis for his opinions.

3. The TON Blockchain is as secure as other successful blockchains had been at the pre-launch stage.

The TON Blockchain is no less secure as other comparable blockchains were at launch. By insisting that the blockchain "must not only be secure and efficient, it must be provably secure and efficient otherwise no one will trust it enough to use it," Prof. Herlihy in his report applies the "provably secure" standard to the TON Blockchain – the standard which Prof. Herlihy leaved undefined, and which has not been used to measure comparable operational blockchains.

Software verification (computer science area dedicated to provably correct software) has a long and complicated history dating back to 1960s, when C.A.R. Hoare published his

influential paper "An Axiomatic Basis for Computer Programming" in 1969 (http://extras.springer.com/2002/978-3-642-63970-8/DVD3/rom/pdf/Hoare_hist.pdf). However, even now, half a century later, verification of complex programs is still an accomplishment infrequently attained in practice.

For example, all complicated software is developed in the so-called high-level languages; the resulting code is then converted into machine code by programs called compilers. No program can be considered correct without a correct compiler, just like one cannot trust the words of a non-English speaking witness if her interpreter cannot be trusted. Currently, the most advanced verified compiler appears to be CompCert (http://compcert.inria.fr/) designed for the C language. Even this compiler is not perfect,[5] it is good enough to serve as critical embedded software in such mission-critical systems as aircraft navigation equipment or nuclear power plants' reactor control systems. And the verification it provides for the C language is quite advanced compared to the compiler-level verification available for C++ language used in such blockchains as Bitcoin and Ethereum; the Go language (also used in Ethereum); the Haskell language used in Cardano; or to almost any other modern language.

The CompCert compiler is usually rather small in comparison with the general-purpose software, and performs a limited set of operations, but that set is guaranteed to be correct. Compare the following two examples:

(a) a formally proved operating system kernel seL4 (https://dornerworks.com/wp-content/uploads/2018/09/DornerWorks-Design-Services-seL4-Microkernel.pdf). The source files

---

[5] The compiler fails to support the full C99 standard for C – for example, it does not support variable-size arrays and unstructured switches in C, so a working C program may need to be simplifies in order to be compilable by CompCert, see CompCert manual: http://compcert.inria.fr/man/manual.pdf (similar to adapting complicated texts for children.)

for it have about 10 thousand lines of code. This is a very basic and simple operating system (a kernel, to be precise -- the core of an operating system, just very basic OS functions). It has no user-friendly interfaces, no wide selection of software -- but it is possible to prove that it works correctly in some sense. This basic nature of the system makes it possible to prove that it works correctly enough for example to control a quad copter, see http://ts.data61.csiro.au/projects/TS/SMACCM/.

(b) Linux OS has millions lines of code. Linux is a general-purpose system: it has many features, a very user-friendly interface, and supports numerous types of equipment -- but it is not verified (and its verification task is hundred times harder than for a simple system as in (a) above).[6] Most general-purpose programs -- including blockchains -- are run under Linux or similar operating systems.

A formally provable verification is not the only means to get a verified software program. For instance, one of the TON smart contracts developed by TON Labs is verified at the machine code level, including parts of the blockchain and its virtual machine. This process does not involve proving that the interpreter can be trusted; rather, it verifies that a particular story is translated correctly. Even though this particular smart contract only contains few hundred lines of code in Solidity, it has been analyzed by top specialists for months. This example demonstrates how complicated the task of formal verification is and why almost no software, even most critical as that used by banks and payment processing systems, has been formally verified or will be regularly subjected to this procedure in the near future.

---

[6] A review of currently verified operating systems could be found here: https://ts.data61.csiro.au/publications/nicta_full_text/955.pdf.

But how can any software actually work if it is not "proved"? Numerous methods exist to solve the problem, and those methods have been used in the TON Blockchain. In particular:

- extensive testing: the TON Blockchain has been tested for months in various environments by many engineers and independent teams of developers;

- proper code style, community review, and development procedures in place: the TON Blockchain has open-sourced all of its components for community review (https://test.ton.org/) , and has two testnet versions run by two different independent organizations, see https://test.ton.org/testnet/ and http://net.ton.dev/;

- parallel development of multiple versions of the same component: the TON Blockchain has two completely independent implementations written in two different languages by two different unrelated and independent organizations, see https://github.com/ton-blockchain/ton and https://github.com/tonlabs. The differences detected in these separate implementations have greatly contributed to the TON Blockchain's reliability and dependability.

This is how the process works in general: any software, even as well-established as MS Windows, has issues that are discovered and fixed throughout its life-cycle (aptly demonstrated by frequent Windows updates).

Prof. Herlihy cites an example of DAO – a blockchain-based hedge fund established by some of the founders of the popular Ethereum blockchain (whom Prof. Herlihy derisively refers to as "celebrity programmers") – that had a security vulnerability exploited by hackers. Report ¶ 13. But this example actually proves the opposite point: despite the fact that Ethereum has had security and vulnerability issues, as well as bugs, it was successfully launched, managed to resolve numerous problems any similar project is bound to face, and continues to drive innovations in computer science and finance to this day. Ethereum was not "provably secure" when it was launched – but that did not prevent Ethereum from growing into the world's second largest blockchain.

Prof. Herlihy also opines that "a program that is 99.9% secure, however elegantly structured, is not secure at all." Report ¶ 13. Prof. Herlihy, however, offers no examples of software that is 100% secure – because, as shown above, such software does not exist.

Prof. Herlihy explains that "attackers may be well-funded: widely-used blockchains such as those proposed for Facebook or Telegram users may be the target of nation-state actors who attempt disruption for strategic reasons." Prof. Herlihy lost sight of the fact that Telegram itself had been a target of the Russian state and survived its attacks. It is public knowledge that the Russian Federal Security Services (known by its Russian acronym FSB, the KGB successor) unsuccessfully attempted to breach Telegram and to force Mr. Durov to turn over its security keys. The Russian Federal Service for Supervision of Communications, Information Technology, and Mass Media (Roskomnadzor) banned Telegram in Russia and tried to block it. Yet, attempts to block the service failed, and the Telegram network keeps on freely operating in Russia without any additional fine-tuning. This is not a bad example of a system built by Nikolai Durov that nobody managed to hack, even the state-backed intelligence agency. Is this real-life experience any less of a gold standard for security than a peer-reviewed conference or a white paper publication, which Prof. Herlihy offers as "the gold standard for establishing security" (as discussed below)?[7]

More generally, there is no complex software on the market today, and unlikely to be in the future, that is absolutely 100% secure from any external challenge or an attack from a superior technology, especially plotted at the state level. To demand a 100% security from such

---

[7] What would Nassim Nicholas Taleb, the best-selling author of *The Black Swan*, *Antifragile* and *Skin in The Game*, say? In his view, "the odds of an academic 'researcher' producing anything eventually used by society is of the order of .00001%. That includes scientists. The odds for a baker: 100%," https://twitter.com/nntaleb/status/935495725567369216.

attacks is an unrealistic and unattainable ideal, and is a wrong standard against which to measure the TON Blockchain, or any other blockchain for that matter, at the pre-launch stage.

Prof. Herlihy further claims that "the gold standard for establishing security is a publication at a peer-reviewed conference. For example, see publications on core blockchain algorithms for Facebook and Algorand." Report ¶ 16. The comparison does not seem relevant. Community trusts Pavel and Nikolai Durov who had previously created the product (Telegram messenger) used by millions of people and known for years of its efficient and secure performance.

Moreover, the fact that Prof. Herlihy cites some peer-reviewed algorithms does not means that all secure algorithms have been peer-reviewed, nor does it mean that those algorithms that have been peer-reviewed are secure. Prof. Herlihy does not – as he cannot -- claim that a peer-review raises the security of the reviewed blockchain from 99.9% -- which is "not secure at all" in his opinion -- to 100%.

Peer review or no peer review, many blockchains face security and centralization issues. For instance, despite a prolonged peer review process, the Grin protocol based on MimbleWimble appeared to be faulty and lost almost all of its core features long after its actual launch, https://medium.com/dragonfly-research/breaking-mimblewimble-privacy-model-84bcd67bfe52. The case gave rise to a long discussion lasting to this day; the community keeps on asking whether this was an unknown attack type, whether the attack was detected or missed by the peer review, whether anyone knew about it and why it was not prevented. Yet, the Grin network is up and running, allowing innovation to thrive.

In the same vein, Prof. Herlihy refers to self-published white papers as an indicator of blockchain security: "organizations often self-publish documents that include mathematical

analyses of security threats along with proofs of correctness. Examples include Algorand, Facebook and Stellar." Report ¶ 16.

First, the TON Blockchain already possesses a significant descriptive first-party and third-party database, which exceeds Algorand's or Facebook's. There are at least two TON node and TVM implementations (by ton.org and TON Labs), whereas the above-mentioned projects only have one implementation each.

Second, the requirement itself is not well-grounded: as mentioned earlier, verifying the whole code-base is unacceptably complex, and the above-mentioned solutions are not an exception. Some small parts of the equation can be proven, but that does not guarantee it is correct as a whole. As summarized by Donald Knuth (an American computer scientist, mathematician, and professor emeritus at Stanford University, 1974 recipient of the ACM Turing Award): "Beware of bugs in the above code; I have only proved it correct, not tried it," https://en.wikiquote.org/wiki/Donald_Knuth.

Prof. Herlihy also cites "third-party auditing" as an indication of security: "An organization may engage an independent third party to audit and publicly certify its code." TON Labs is an independent third-party developing its own implementation of the TON protocol, including all its components, developer tools and infrastructure, and has published its research and code. TON Labs has verified the correctness of many TON components, implemented it independently using a completely different programming language and without any assistance from the TON team. This is a process that no other existing blockchain actually went through before the launch, as all of them were developed by only one founding team. It is no exaggeration to say that TON is one of the most peer reviewed and verified protocols in the blockchain history.

In any event, this certification requirement has not been applied to other successful blockchains. Nobody certified Bitcoin before it was launched. Indeed, "[w]hen we heard about bitcoin for the first time, many of us cryptographers -- myself included – did not think it was going to work." Alejandro Hevia, University of Chile, at https://www.wired.com/story/why-you-cant-trust-most-cryptocurrency-white-papers/. The famous Bitcoin white paper by Satoshi Nakamoto was neither peer-reviewed nor verified.

Prof. Herlihy also complains that "there is no systematic analysis proving that either the design or specific implementation of these components is secure and correct. I, and in my opinion potential users cannot evaluate the security and effectiveness of the TON Blockchain (and the utility of the Gram token) until these software components have been developed, released, and their security and performance evaluated and established." Report ¶ 8.

There was no "systematic analysis proving that either the design or specific implementation of these components are secure and correct" of Bitcoin or Ethereum protocols available pre-launch either. This requirement has not been applied to any of most successful blockchain systems at a comparable stage, nor should it be applied to the TON Blockchain.

Prof. Herlihy further states that "[t]he public documents assert that a smart contract controls how absent or incorrect validators are punished. I was unable to find any such contract in the public release. The validator punishment smart contract is an inviting attack route, and I cannot be convinced of the TON Blockchain's security without a careful analysis of its code."

The industry term for this function is "slashing conditions." They are not mandatory in the TON Blockchain. Some of operating Proof-of-Stake blockchains, such as Tezos, Polkadot and Ouroborous, claim that they are planning to implement "slashing" in the future, but they do not rush to implement this feature in order not to discourage validating. There is no network-

threatening necessity in the implementation of slashing conditions, particularly in the TON Blockchain, where the design of its consensus algorithm makes it a forkless protocol.

"The whole point of the Ouroboros[8] is that it does not require slashing to guarantee security. For example, Casper[9] directly depends on the slashing mechanism to argue its properties, e.g., if validators disagree, some of them are slowly slashed out of their stake until their stake is irrelevant. Ouroboros does not require this because it manages to provide all the necessary safety proofs without need for slashing," https://forum.cardano.org/t/how-can-ouroboros-claim-solving-pos-while-not-punishing-malicious-validators/15922/3. The same is true for the Cardano Blockchain, and the same is true for the TON Blockchain. Indeed, there is a well-described and implemented mechanism for punishing a node trying to support an invalid block in the TON protocol, as described in https://docs.ton.dev/86757ecb2/p/07ddda-walk-through-the-catchain. The TON Consensus does not require a slashing condition to perform. If a node fails to validate correctly, it becomes a part of the Byzantine node cluster and does not participate in the consensus procedures. In case more than 1/3 of the nodes fail to perform the validation, the network loses all of its value, including the value of the malicious node stake. There is thus no gain in refusing validation.

Finally, Prof. Herlihy asserts that "I was unable to find code for rewarding the fishermen function. I cannot be convinced of the TON Blockchain's security without a careful analysis of how correct accusations are rewarded and false accusations deterred, since there is a potential for

[8] Ouroboros is the consensus protocol for the Cardano Blockchain https://www.cardano.org/en/ouroboros/.

[9] Casper is a family of consensus protocols for the Ethereum 2.0 Blockchain https://github.com/ethereum/wiki/wiki/Proof-of-Stake-FAQ.

denial-of-service attacks." A "fisherman" node is not crucial to the TON Blockchain's performance or security for the exact same reasons that the "vertical blockchain" is not critical.

4. The TON Blockchain can support an extensive suit of services already developed and additional services to be developed after the mainnet launch.

Developers working on the TON Blockchain have created an extensive suit of services, tools and smart contracts that successfully run on the current testnet implementation of the Blockchain and will eventually be purchasable by Grams on the mainnet after launch. A list of selected examples in the open GitHub and other repositories includes:

1. https://github.com/tonlabs (21 public repositories);
2. https://github.com/formony/ton_client;
3. https://github.com/atomex-me/xeus-fift;
4. https://github.com/broxus/ton-client;
5. https://github.com/serokell/fift-asm-dsl;
6. https://github.com/akme/tonmon;
7. https://github.com/button-tech/ton-sync-payments-channel;
8. https://github.com/button-tech/ton-delegation-pool;
9. https://github.com/broxus/ton-api;
10. https://github.com/serokell/ton-paychan;
11. https://github.com/button-tech/gram-eth;
12. https://github.com/aquigni/TON-MTProxy;
13. https://github.com/aqoleg/multisig;
14. https://github.com/dblokhin/ton-charity-foundation;
15. https://github.com/Tynik/python-fift;

16. https://github.com/Skydev0h/ton-freestyle;

17. https://github.com/button-tech/blockchain-gram-testnet;

18. https://github.com/TONRnD/TON-Swap;

19. https://github.com/serokell/ton.nix;

20. ton.sh;

21. toncenter.com;

22. gram-wallet.org.

23. Tengram.com

24. http://github.com/koinov

25. https://github.com/nicegram/wallet-ios

In addition to the above-referenced services in the public domain, many additional developed services are stored in private repositories.

There are even enterprise-grade solutions already based on the TON Blockchain technology. For example, TON Labs Operating System (TON OS) Node Enterprise Edition which already shows traction with customers such as Coinbase.

Accordingly, Prof. Herlihy's statement that "[t]oday, few if any of [the services that will eventually be purchasable by Gram holders] exist, based on my review, and the TON Blockchain is not yet mature enough to support them," is demonstrably incorrect as a factual matter.

But Prof. Herlihy is mistaken on a more fundamental level: his very premise that the TON Blockchain should be mature enough pre-launch to support services is questionable. Why should the TON Blockchain support any services before it is launched, and why cannot these services be added post-launch? There is more than enough developer activity to prove that the community around the TON Blockchain has already developed a number of solutions on TON.

Notably, Bitcoin and Ethereum blockchains had no services or applications, no plans to develop any, and no "proven" maturity to support them at their launch dates.

CONCLUSION

As shown above, the TON Blockchain meets all the requirements for a successful launch, which have been verified by independent teams of developers – top-notch blockchain specialists. Yet, the SEC's expert Prof. Herlihy holds the TON Blockchain to the standards of performance, security and maturity that either do not currently exist, or have never been applied to other existing successful blockchains. Had Prof. Herlihy's exacting academic standards been applied to Bitcoin, Ethereum or other successful blockchains, they would have failed these standards and would not have been launched. None of these known blockchains has ever been analyzed for security or compliance with its claimed parameters -- all just launched after some research that might or might not have been peer-reviewed, and following the launch underwent initial tune-up and stabilization, development, and security verification; added services and applications; and have functioned for years in a secure and efficient manner. Nothing in Prof. Herlihy's Report justifies treating the TON Blockchain any differently.

But there is a more fundamental problem with Prof. Herlihy's approach: it contradicts the real-life experience of many successful and secure blockchains such as Bitcoin, Ethereum, or Tezos, none of which would have been able to come up to the proof demanded by Prof. Herlihy's unrealistic standards of pre-launch performance, security, and maturity. Under Prof. Herlihy's scrutiny, none of those blockchains would have been launched.

Yet, they all did launch, and the new blockchain world was born. Now, Prof. Herlihy's approach threatens the future of this brave new world: his unrealistic standards, not found in nature, would kill future bitcoins – just like it would have killed Bitcoin if given a chance in 2009. The danger of this approach goes beyond this particular case about the TON Blockchain --

it will stop or considerably chill many other promising and competitive cutting-edge technologies in the future.

The Foundation respectfully submits that the Court should not grant its seal of approval to Prof. Herlihy's academic opinions -- unfounded in fact, contradicted by past experiences, and quite dangerous to future innovations.

February 14, 2020

Respectfully Submitted,

Dimitry Joffe (DJ-6498)
JOFFE LAW P.C.
765 Amsterdam Avenue, 2C
New York, New York 10025
(917) 929-1964
Dimitry@joffe.law

Counsel to Amicus Curiae
the TON Community Foundation