CertiK white paper

draft for open community review and subject to change

CertiK: Building Fully Trustworthy Smart Contracts and Blockchain Ecosystems

www.certik.org

May 3, 2018

IMPORTANT NOTICE

NOTHING IN THIS WHITEPAPER CONSTITUTES LEGAL, FINANCIAL, BUSINESS OR TAX ADVICE AND YOU SHOULD CONSULT YOUR OWN LEGAL, FINANCIAL, TAX OR OTHER PROFESSIONAL ADVISER BEFORE ENGAGING IN ANY ACTIVITY IN CONNECTION HEREWITH. NEITHER CERTIK FOUNDATION LTD. (THE FOUNDATION), ANY OF THE PROJECT TEAM MEMBERS WHO HAVE WORKED ON THE CERTIK PLATFORM (AS DEFINED HEREIN) OR PROJECT TO DEVELOP THE CERTIK PLATFORM IN ANY WAY WHATSOEVER (THE CERTIK TEAM), ANY DISTRIBUTOR/VENDOR OF CTK (THE DISTRIBUTOR), NOR ANY SERVICE PROVIDER SHALL BE LIABLE FOR ANY KIND OF DIRECT OR INDIRECT DAMAGE OR LOSS WHATSOEVER WHICH YOU MAY SUFFER IN CONNECTION WITH ACCESSING THIS WHITEPAPER, THE CERTIK FOUNDATION WEBSITE AT HTTPS://CERTIK.ORG OR ANY OTHER WEBSITES OR MATERIALS PUBLISHED BY THE FOUNDATION.

This Whitepaper is intended for general informational purposes only and does not constitute a prospectus, an offer document, an offer of securities, a solicitation for investment, or any offer to sell any product, item or asset (whether digital or otherwise). The information herein below may not be exhaustive and does not imply any elements of a contractual relationship. There is no assurance as to the accuracy or completeness of such information and no representation, warranty or undertaking is or purported to be provided as to the accuracy or completeness of such information. Where this Whitepaper includes information that has been obtained from third party sources, the Foundation and/or the CertiK team have not independently verified the accuracy or completion of such information. Further, you acknowledge that circumstances may change and that this Whitepaper may become outdated as a result; and the Foundation is under no obligation to update or correct this document in connection therewith.

This Whitepaper does not constitute any offer by the Foundation, the Distributor or the CertiK team to sell any CTK (as defined herein) nor shall it or any part of it nor the fact of its presentation form the basis of, or be relied upon in connection with, any contract or investment decision. Nothing contained in this Whitepaper is or may be relied upon as a promise, representation or undertaking as to the future performance of the CertiK Platform. The agreement between the Distributor and you, in relation to any sale and purchase of CTK is to be governed by only the separate terms and conditions of such agreement.

By accessing this Whitepaper or any part thereof, you represent and warrant to the Foundation, its affiliates, and the CertiK team as follows:

(a) you acknowledge, understand and agree that CTK may have no value, there is no guarantee or representation of value or liquidity for CTK, and CTK is not for speculative investment;

(b) none of the Foundation, its affiliates, and/or the CertiK team members shall be re-

sponsible for or liable for the value of CTK, the transferability and/or liquidity of CTK and/or the availability of any market for CTK through third parties or otherwise;

(c) in any decision to purchase any CTK, you have not relied on any statement set out in this Whitepaper;

(d) you will and shall at your own expense ensure compliance with all laws, regulatory requirements and restrictions applicable to you (as the case may be); and

(e) you acknowledge, understand and agree that you are not eligible to purchase any CTK if you are a citizen, national, resident (tax or otherwise), domiciliary and/or green card holder of a geographic area or country (i) where it is likely that the sale of CTK would be construed as the sale of a security (howsoever named) or investment product and/or (ii) in which access to or participation in the CTK sale or the CertiK Platform is prohibited by applicable law, decree, regulation, treaty, or administrative act, and/or (including without limitation the United States of America, Canada, New Zealand, People's Republic of China and the Republic of Korea).

The Foundation, the Distributor and the CertiK team do not and do not purport to make, and hereby disclaims, all representations, warranties or undertaking to any entity or person (including without limitation warranties as to the accuracy, completeness, timeliness or reliability of the contents of this Whitepaper or any other materials published by the Foundation). To the maximum extent permitted by law, the Foundation, the Distributor, their related entities and service providers shall not be liable for any indirect, special, incidental, consequential or other losses of any kind, in tort, contract or otherwise (including, without limitation, any liability arising from default or negligence on the part of any of them, or any loss of revenue, income or profits, and loss of use or data) arising from the use of this Whitepaper or any other materials published, or its contents or otherwise arising in connection with the same. Prospective purchasers of CTK should carefully consider and evaluate all risks and uncertainties (including financial and legal risks and uncertainties) associated with the CTK sale, the Foundation, the Distributor and the CertiK team.

All contributions will be applied towards the Foundation's objects, including without limitation promoting the research, design and development of, and advocacy for a decentralised community system which would promote the establishment of a formal verification framework for building fully reliable, secure and hacker-resistant smart contacts and blockchain ecosystems.

The information set out in this Whitepaper is for community discussion only and is not legally binding. The agreement for sale and purchase of CTK and/or continued holding of CTK shall be governed by a separate set of Terms and Conditions or CTK Purchase Agreement (as the case may be) setting out the terms of such purchase and/or continued holding of CTK (the Terms and Conditions), which shall be separately provided to you or made available at https://certik.org. In the event of any inconsistencies between the Terms and Conditions and this Whitepaper, the Terms and Conditions shall prevail.

No regulatory authority has examined or approved of any of the information set out in this Whitepaper. No such action has been or will be taken under the laws, regulatory requirements or rules of any jurisdiction. The publication, distribution or dissemination of this Whitepaper does not imply that the applicable laws, regulatory requirements or rules have been complied with.

All statements contained in this Whitepaper, statements made in press releases or in any place accessible by the public and oral statements that may be made by the Foundation, the Distributor and/or the CertiK team may constitute forward-looking statements (including statements regarding intent, belief or current expectations with respect to market conditions, business strategy and plans, financial condition, specific provisions and risk management practices). You are cautioned not to place undue reliance on these forward-looking statements given that these statements involve known and unknown risks, uncertainties and other factors that may cause the actual future results to be materially different from that described by such forward-looking statements, and no independent third party has reviewed the reasonableness of any such statements or assumptions. These forward-looking statements are applicable only as of the date of this Whitepaper and the Foundation and the CertiK team expressly disclaims any responsibility (whether express or implied) to release any revisions to these forward-looking statements to reflect events after such date.

This Whitepaper may be translated into a language other than English and in the event of conflict or ambiguity between the English language version and translated versions of this Whitepaper, the English language version shall prevail. You acknowledge that you have read and understood the English language version of this Whitepaper.

CertiK: Building Fully Trustworthy Smart Contracts and Blockchain Ecosystems

Draft for open community review and subject to change

The CertiK platform is envisaged to be a formal verification framework for building fully trustworthy smart contacts and blockchain ecosystems. Different from the traditional testing approaches to detect bugs, the CertiK platform attempts to mathematically prove that blockchain ecosystems are bug-free. The Foundation has developed modular verification techniques to decompose such an otherwise prohibitive proof task into smaller ones that can be automatically solved in a decentralized style. These proof objects can be built and encoded in the CertiK platform's transactions and will then be validated by other participants. Thus, the CertiK platform's blockchain is intended to work as certificates to exhibit the end-to-end correctness and security of the verified smart contracts, libraries of decentralized applications (DApp), and the implementations of the blockchain itself. That is also why these are called certified blockchain ecosystems.

The CertiK team compromises world-class formal verification experts who are professors from Columbia University and Yale University, as well as senior software engineers from Google, Facebook, and FreeWheel. Previously, the founders of the CertiK team had successfully built the world's first fully verified concurrent OS kernel, named CertiKOS, using the Coq proof assistant. This CertiKOS work is a core component of an NSF Expeditions in Computing project [DeepSpec], is nominated and selected as research highlights of CACM, and has been widely considered "a real breakthrough" toward hacker-resistant systems [YaleNews, IBTimes, YDN]. These previous successes indicate that the CertiK techniques can be revolutionary to blockchain ecosystems by making them truly reliable and secure.

Contents

1 Introduction

Blockchain technologies, pioneered by Bitcoin [5] and Ethereum [6], provide a globally-consistent ledger that does not rely on a central trusted authority. These ledgers can record the transactions of virtual currencies by a collaboration of network nodes. The Proof-of-Work (POW) [5] or Proof-of-Stake (POS) [16] mining schemes set up a theoretically unaffordable computational cost to prevent false transactions. Therefore, it seems that the ledgers are "trustable" even without a central authority. Based on this trust, smart contracts [17] and other forms of decentralized applications (DApp) can be stored in the ledgers and form the blockchain ecosystems, whose source code are entirely "transparent" to the public. In these ecosystems, the central authority is replaced by the consensus among network nodes, and the value is created through a system of decentralized trust.

However, blockchain ecosystems are not truly trustable. Due to their transparent policy and the potentially dramatic benefits one may receive from a successful attack, these ecosystems are in reality highly sensitive to attacks and are far more vulnerable than expected:

- Although the protocol is well-designed and highly reliable, the ledger implementations like other complex systems may have flaws and do not fully meet the protocol (or specifications). For example, there are 703 open issues and 2,186 closed issues reported for the official Ethereum Virtual Machine (EVM) implementation on Jan 15, 2018 [15]. Some of these issues may very well compromise the guarantees of blockchain ecosystems from the root.

- The implementation of cryptographic software library is also error-prone [18]. These bugs can expose security risks that may allow the digital signature mechanism to be bypassed and lead to a huge financial loss.

- Due to the transparent property, smart contracts and other DApps have to expose all their protocol design and even their source code to everyone, including malicious users. These features make some applications, like digital wallets, attractive but defenseless towards hackers. For instance, a variant of a well-documented reentrancy attack was recently exploited in TheDAO [19] digital wallet, leading to the theft of more than $50M worth of Ether.

- Since blockchain ecosystems are decentralized (or unsupervised), any actions have to reach consensus among the majority of network nodes in order to become effective. Thus, once the DApps are released, it may become hard and slow to fix bugs. Take TheDAO attack as an example. Recovering TheDAO funds required a hard fork of the blockchain. This poses a new requirement for DApps – that they have to be truly trustworthy before being uploaded to the ledgers.

State-of-the-art approaches There are many ways to improve the reliability and security of system software, but none of them can fully address these challenges introduced by blockchains. Testing is currently the most widely used approach to enhance the trust of systems. However, as Dijkstra said, program testing can be used to show the presence of bugs, but never to show their absence [8]. It is obvious that using testing alone cannot eliminate the zero-day vulnerability issues.

Formal verification is an alternative approach that aims to mathematically prove that the system is correct with respect to specifications. However, it is still difficult to formally verify practical and complex systems. Traditional verification techniques like the model checking [11] are limited to ensuring functional correctness and suffer from the state explosion problem [12] when dealing with concurrent/decentralized programs. Besides, some researchers [13] insist on developing mechanized proofs for functional correctness using proof assistants. This approach enables the handling of richer properties but requires substantial proof efforts. In fact, while such "formal verification" concept dates back to the 1960s, complete formal proofs of non-trivial sequential systems only became feasible recently, as demonstrated by seL4 in 2009 [7]. This result was encouraging, and it seemed not too far away from building a entirely verified concurrent/decentralized practical system using reasonable proof efforts. However, nine years have passed while this last step is still insurmountable. In the single-core setting, the cost of such verification is already prohibitive; seL4 took 11 person-years to verify 7,000 lines of C code. Several researchers [9, 10] believe that it is impossible to fully verify practical concurrent/decentralized systems like blockchain ecosystems, and even if it can be done, the costs will far exceed the sequential ones. To address these challenges, one will have to answer the following questions:

- *What to prove?* Most of the existing verification services can only prove that the program satisfies a list properties, e.g., "no stack overflow", "all exceptions have been handled", etc. However, such a list of properties is insufficient to ensure that the program implements the functionality correctly. Instead, the functional correctness of these programs have to be proven. However, writing down the functional specifications alone is a complicated affair. It requires a deep understanding of the entire system and a rigorous means of expressing the desired system behaviors.

- *How to scale the proof development?* The current proof cost has become a significant obstacle. There is a need to further cut down the proof efforts and makes it possible for one project to borrow intelligence and computation resources from a broader community.

- *How to let others trust the proofs?* Developing a proof method is hard, but it is even harder to convince people that the method is sound. It is not very meaningful to force others to trust some so-called "black-box" proofs without understanding how and why these proofs work. There is a need to allow people to validate proofs on their local machines and encourage everyone to participate in this validation procedure.

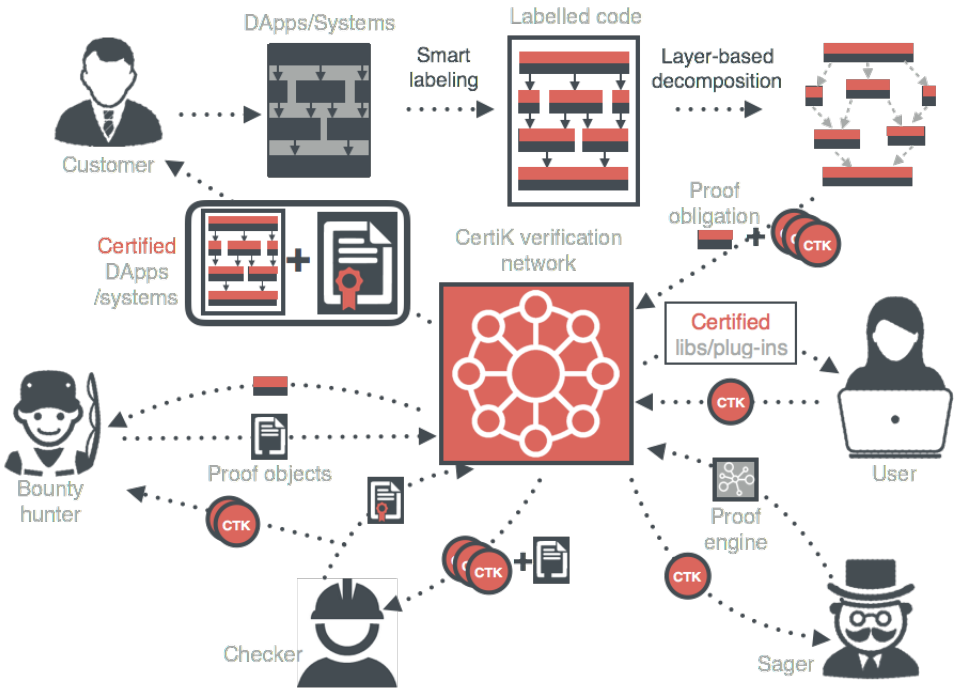


Figure 1: The CertiK framework and community.

CertiK Platform overview It is believed that the answers to the above questions are rooted in the blockchain itself. This belief has guided the Foundation to develop a one-stop solution, named the CertiK Platform, which provides a powerful set of Certified Kits for building fully trustworthy blockchain ecosystems (see Fig. 1):

- *Smart labeling.* The CertiK Platform has designed a novel approach to specify DApps/systems using labels. These labels are expressive enough to formally state the desired properties and are compatible with the existing programming languages (e.g., Solidity). By utilizing deep learning techniques with manually established labeled code base for training, the CertiK Platform intends to introduce a framework, named smart labeling, to understand decentralized programs not only at the syntax level but also at the semantics level and then adding proper labels to the source code automatically.

- *Layer-based decomposition.* The CertiK team is among the first to achieve modular verification by realizing a novel concept, named layered deep specifications [1, 2, 3, 4]. This technique uncovers the insights of layered design patterns and makes it possible to decompose a complex proof task into smaller ones and verify each of them at their proper abstraction level.

- *Pluggable proof engine.* These decomposed proof obligations are much easier to un-

Figure 2: Screenshot of verifying "in-place swap" and "account transfer" methods using CertiK.

tangle and can even be solved by some automatic verifiers (e.g., SMT solvers [14]). To enable extensibility, the CertiK Platform is intended to provide an open protocol such that more advanced solving algorithms can be freely plugged into this system.

- *Machine-checkable proof objects.* The CertiK Platform constructs mechanized proof objects (or counterexamples) such that these proofs can be quickly checked by anyone using their own machine. These proof objects can be viewed as the "certificates" [13] to the verified programs.

- *Certified DApp libraries.* In order to improve the code quality and reliability of the entire blockchain community, the CertiK Platform offers a series of certified libraries and plug-ins to the integrated development environment (IDE) for building more trustworthy DApps. The use of these tools will cost a small amounts of CTK as virtual crypto "fuel", but will provide more assurance during the development time.

- *Customized certification services.* For DApps/systems (e.g., digital wallets) with high-reliability requirements, the CertiK Platform intends to provide customized certification services. In this case, verification experts will help specify/verify the programs and generate a detailed, comprehensive report.

Figure 2 presents how to use the online services on the CertiK Platform to verify two simple functions: in-place swap (line 7 to 15) and account transfer (line 16 to 25). Specifications to these functions can be expressed using CertiK labels, e.g., "@pre," "@post," and

"@inv," which stand for pre-condition, post-condition, and invariants respectively. Since these labels are written in the comments, there is no need to modify the compiler and no switch costs for developers. The functions together with specifications (or labels) will then be processed and decomposed by the CertiK Platform and sent to verifier to solve. After the verification, the CertiK Platform's back-end will return a detailed and comprehensive evaluation report. Counterexamples will be provided if the proof obligation cannot be satisfied. For example, the invariant that the balance is always non-negative is violated by the transfer function at line 21. As shown Fig. 2, the CertiK Platform's certified kits can even offer real-time feedback to help developers fix the discovered bugs.

The native digital fuel of CertiK Platform ("CTK") is a major component of the ecosystem on the CertiK Platform. CTK is a non-refundable functional utility fuel which will be used as the unit of exchange between participants on the CertiK Platform, as well as the economic incentives which will be consumed to encourage participants to contribute and maintain the ecosystem on the CertiK Platform (as described below). CTK is an integral and indispensable part of the CertiK Platform, because in the absence of CTK, there would be no common unit of exchange to reward users to incentivise them for work done on the CertiK Platform, thus rendering the ecosystem on the CertiK Platform unsustainable.

The certified kits on the CertiK Platform will be further powered by the CertiK community and create a decentralized style of work such that everyone can construct proofs, validate proofs, improve solver algorithms, and also contribute new proof obligations. The CertiK Platform introduce a new mining scheme, named Proof-of-Proof (PoP), involving the distribution of CTK incentives. The traditional mining scheme, e.g., PoW, raises the computational cost by finding a formatted value (which includes all transactions to appear in the block) whose SHA256 hash matches some difficulty threshold. Finding such a hash does not create any "real value." In comparison, the computational difficulties of our PoP lie in the proof search which solves concrete problems. This PoP scheme unifies the whole community through the flow of CTKs among five different roles:

- *Customers* can submit programs/systems that need verification (through the CertiK Platform's services) or any proof obligations (that meet the open protocol) to the CertiK Platform's network. This is done by initiating and broadcasting a special "proof request" transaction associated with some CTK incentives offered for anyone who constructs the proofs.

- *Bounty hunters* are the ones who aim for CTK incentives and would like to share their computation resources. They will construct and broadcast the proof objects, and then wait for the proofs to be validated. Due to the significant importance of this role, only users who possess a certain amount of CTKs are allowed to take this role.

- *Checkers* can get CTK incentives by recording regular transactions or check the submitted proof objects. Bounty hunters can only receive their incentives once their proofs are validated and checkers can also get a small portion of these incentives.

- *Sages* are the ones who plug in their proof engines via the CertiK Platform's open protocol. Their engines may be randomly used by bounty hunters and will be evaluated through A/B testing. They can also get some CTK incentives depending on the evaluation result of their engines. Outstanding engines will be studied and spread by the community.
- *Users* can subscribe to all CertiK Platform's certified libraries and IDE plug-ins to build their own DApps/systems with some CTKs.

These five roles will balance, guard, and improve the CertiK Platform's community. Along with the CTK flows, real value is generated by posing and solving proof obligations, validating proof objects, and creating advanced proof engines. The Foundation believes that the proof/verification requirements are universal, which will keep the CertiK Platform's community active.

Statement of CTK. CTK does not in any way represent any shareholding, participation, right, title, or interest in the Foundation, its affiliates, or any other company, enterprise or undertaking, nor will CTK entitle CTK holders to any promise of fees, revenue, profits or investment returns, and are not intended to constitute securities in Singapore or any relevant jurisdiction. CTK may only be utilized on the CertiK Platform, and ownership of CTK carries no rights, express or implied, other than the right to use CTK as a means to enable usage of and interaction with the CertiK Platform. In particular, you understand and accept that CTK:

(a) is non-refundable and cannot be exchanged for cash (or its equivalent value in any other virtual currency) or any payment obligation by the Foundation or any affiliate;

(b) does not represent or confer on the CTK holder any right of any form with respect to the Foundation (or any of its affiliates) or its revenues or assets, including without limitation any right to receive future revenue, shares, ownership right or stake, share or security, any voting, distribution, redemption, liquidation, proprietary (including all forms of intellectual property), or other financial or legal rights or equivalent rights, or intellectual property rights or any other form of participation in or relating to the CertiK Platform, the Foundation, the Distributor and/or their service providers;

(c) is not intended to be a representation of money (including electronic money), token, security, commodity, bond, debt instrument or any other kind of financial instrument or investment;

(d) is not a loan to the Foundation or any of its affiliates, is not intended to represent a debt owed by the Foundation or any of its affiliates, and there is no expectation of profit; and

(e) does not provide the CTK holder with any ownership or other interest in the Foundation or any of its affiliates.

The contributions in the CTK sale will be held by the Distributor (or its affiliate) after the CTK sale, and contributors will have no economic or legal right over or beneficial interest in these contributions or the assets of that entity after the CTK sale.

To the extent a secondary market or exchange for trading CTK does develop, it would be run and operated wholly independently of the Foundation, the Distributor, the sale of CTK and the CertiK Platform. Neither the Foundation nor the Distributor will create such secondary markets nor will either entity act as an exchange for CTK.

2 Market Analysis

In the past two years, due to the extreme popular of Bitcoin, the blockchain technology and DApps have become more and more popular. The price skyrocket of virtual currencies derives the exponential growth of the number of smart contracts and other DApps. There are more than one million contracts deployed on Ethereum on January 2018 [20], while this number was only about 0.12 million a year and half ago [21]. Many people [22] believe that blockchain will fundamentally change every aspect of our lives. The development infrastructure is improved daily, and it can be expected that this exponential growth of deployed smart contract will continue in the future. Based on the current growth rate, the total amount of such DApps will probably reach 10 million in the new one to two years.

Market size estimation. Most of the existing smart contracts and DApps are dealing with virtual currencies, making their reliability and security highly sensitive. There is a high demand for verification services. Existing smart contract verification service providers charge several thousand to one million dollars for a single service, although their techniques cannot adequately address the challenges mentioned in Sec. 1. It is evident that this kind of verification service is highly profitable and has a high barrier to entry. If only takes one hundred thousand dollars as the average charge for a service, the market for such verification services alone can be as large as one trillion dollars (i.e., $10M \times \$100K$).

In addition, the techniques used in the CertiK Platform can cover a broader market than existing service providers thanks to our certified DApp libraries and the IDE plug-ins that provide real-time and interactive verification feedback. These services can reduce the development costs by shortening the development cycle and replacing some of the testing infrastructure. According to our collected data, the testing development and maintenance take about 40% of the total development cost [23] and, a DApp with a good quality service typically takes around one million dollars to build. In this sense, if the CertiK Platform certified libraries and plug-ins replace 20% of the testing infrastructure, it will be a trillion-dollar market (i.e., $20\% \times 40\% \times 10M \times \$1M$).

Potential competitors Quantstamp [24] proposes a verification protocol for smart contracts written in Solidity. It utilizes the traditional model checking techniques and requires

an intensive amount of human effort for reviewing the source code and writing the specification manually. This limits the scalability of their approach. Also, it is unclear how to extend the Quantstamp techniques to verify complex systems like the blockchain itself. Despite all these issues, the estimated valuation of Quantstamp is more than $350 million. Solidified [25] and Securify are another two companies claiming to provide verification services for smart contracts. Their services can only be used to check/verify a list of fixed properties rather than the functional correctness.

Zeppelin is a testing/verification service provider that takes a significant percentage of the existing market. But most of their current verification services are done manually. Besides, they also develop a widely used open-source framework for the smart contract development, named OpenZeppelin. However, the libraries provided by OpenZeppelin are either not verified or do not offer mechanized proof objects.

Runtime Verification is a traditional formal verification company and now provide verification services for smart contracts. Similar to the CertiK team, they also have a strong academic background and have proposed one semantics of the EVM, named KEVM. But their work still remains at the research stage. It is a big unknown how their techniques can be applied to industry-grade systems.

3 CertiK Technical Chain

This section describes TheDAO attack as an example to explain the chain of techniques used in the CertiK Platform. Figure 3 shows the pseudocode to replay TheDAO attacks. The vulnerable bank contract (line 1 to 19) maintains the account balance; it adds the deposit value to the balance and reduces the balance with the withdrawal value. However, there is a server bug that the server first sends the money (at line 14) before updating the balance (at line 17) in the withdrawal method. Thus, attackers (line 20 to 32) can utilize the fallback and synchronization features of smart contracts to perform "multiple spend attacks". The attacker first calls the withdraw method (at line 29). When the money is sent back (at line 14), the fallback function (at line 23) of the attacker is triggered, and another withdraw method is invoked again. Due to the fact that balance has not been updated yet, the bank will issue another send. This simple issue of TheDAO digital wallet causes the theft of more than $50 millions worth of Ether.

CertiK labeling. In order to detect and prevent these kinds of bugs, one has to be able to precisely specify the expected behaviors of this bank contract. This can be achieved by using CertiK labels that are lightweight but expressive.

Consider the bank contract example. Its specification can be merely written as a single equation: "balance = deposit - withdraw." Thus, one can insert "@pre" (at line 7), and "@post" (at line 8) labels with this equation before all the method declarations to ensure that this specification is satisfied. Figure 3 only shows the labels for the withdrawal method and use postfix "_post" to represent the value after the method's execution.

```
/* The vulnerable bank contract */
contract Bank{
   uint balance;
   function depositBalance() {
     balance = balance + msg.value;
   }
   /*@CTK TestBalance
       @pre balance = deposit - withdraw
       @post balance_post = deposit_post - withdraw_post
       @inv balance <= deposit - withdraw
       @fun "": lambda v. withdraw_post = withdraw + v
   */
   function withdrawBalance() {
     if (msg.sender.call.value(balance)() == false) {
         throw;
     }
     balance = 0; /* BUG! Reduce balance after sending money */
   }
}
/* The malicious contract that attacks the bank*/
contract Attacker {
   address bankAddress;
   function() { /* fallback function call withdrawnBalance recursively */
     bankAddress.call(bytes4(sha3("withdrawBalance()")));
   }
   function deposit() {
     bankAddress.call.value(2).gas(20764)(bytes4(sha3("depositBalance()")));
   }
   function withdraw() { /* triggers withdrawBalance in the contract Bank*/
     bankAddress.call(bytes4(sha3("withdrawBalance()")));
   }
}
```

Figure 3: Pseudocode illustrating TheDAO attack.

However, only ensuring this equation before and after the method call is not strong enough. Temporal breakdown of this equation during the method call may lead to severe consequences. The control flow can be messed up by malicious fallback functions, and this broken point makes the whole contract vulnerable. To solve this issue, one can insert a "@inv" label meaning that the followed property holds at any point of the execution. We weaken the equation to the invariant "balance $\leq$ deposit - withdraw" such that these values do not have to be updated at the same time. Since the fallback function is a "black box" to the bank contract, we can only utilize the "@fun" label to specify its known effects: the money equals to the balance has been withdrawn.

Using this label-based language, the specification of this bank contract can be easily expressed in a formal and comprehensive way. CertiK labels can be used to write any properties in the first class logic. The Foundation plans to add higher-order support shortly

such that these labels are rich enough to specify almost all deployed smart contracts, DApps, and blockchain systems.

This label-based language is well designed such that it is entirely possible to label source programs automatically. The Foundation intends to establish a large training set containing certified DApp libraries that the CertiK team is building manually using selected DApps from popular domains. The Foundation intends to apply deep learning techniques to build a smart labeling framework. With this framework, most of the shared logic and properties can be automatically labeled such that the specification and proof efforts can be dramatically reduced.

CertiK proof engine. The labeled programs will then be compiled using the in house developed CertiK compiler. Different from the general-purpose compiler, CertiK compiler recognizes the label language and can parse the labeled programs into an internal model for DApps. This model can be viewed as abstract automata defining how the DApps will change the system state (consisting of all global and local variables). This model is language independent such that the back-end of the proof engine can be unified.

The proof obligation saying that the behavior of executing programs running on the CertiK Platform's internal DApp model meet the specification (generated from labels) can be converted into a set of constraints. Take the invariant proof for the program in Fig. 3 as an example. At the beginning of the withdraw function (line 12), constraints for the @pre label is generated, i.e., "C1: balance = deposit - withdraw". Also, one has to validate the invariant defined using the @inv label by checking if "I2: ¬(balance ≤ deposit - withdraw)" can be satisfied. Since I2 is always false given C1, one knows the invariant holds at the beginning. Then, at line 13, the function call of the message sender increases the withdraw value with the current balance. For such value updates, one can introduce a new version of the variable, e.g., "withdraw_1" and encode the update into the constraints "C2: withdraw_1 = withdraw + balance". To validate the invariant at this point, one may check if "I2: ¬(balance ≤ deposit - withdraw_1)" can be satisfied. Here, given C1 and C2, one may have that "deposit - withdraw_1 = 0". Thus, any balance with positive initial value will break the invariant. In this way, the withdrawal bug leading to TheDAO attack can be easily detected using the CertiK proof engine.

This procedure can be done by SMT solvers [14], and counterexamples (or hints) will be generated if the problem can be solved. The soundness of the counterexamples (or the proof) can be easily checked with respect to the proof obligation, which forms the basis of the Proof-of-Proof mechanism.

Furthermore, to improve the performance of this solving procedure, the Foundation intends to design an open protocol such that any SMT solvers can be plugged into the CertiK Platform's network. These solvers will be randomly selected to prove some already verified programs and the results, including the execution time and the quality of the generated hints, will be evaluated. The solvers with better performance have a higher chance to be selected.

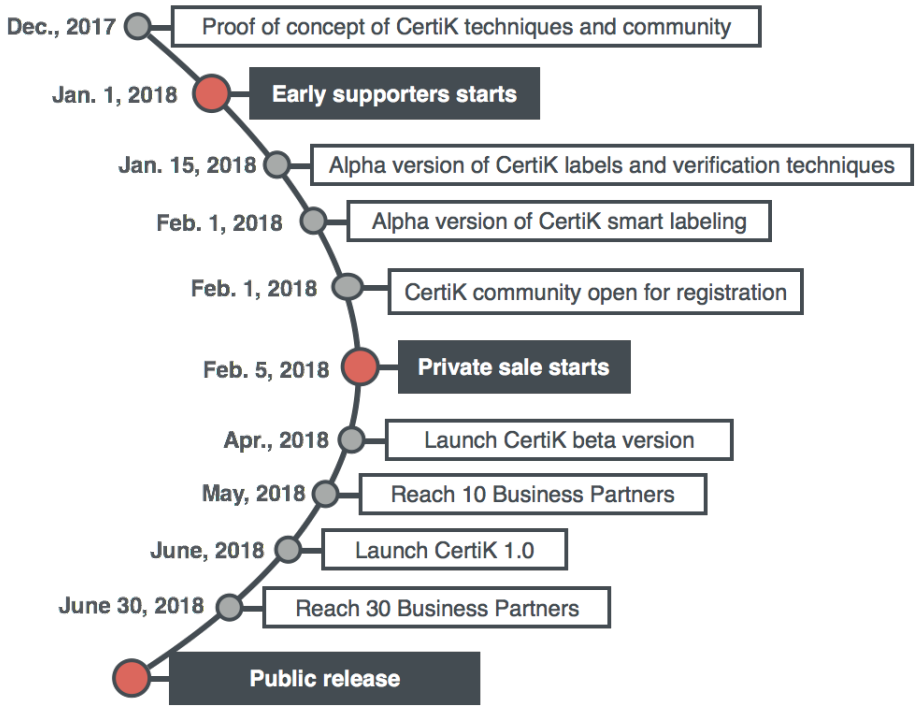


Figure 4: The roadmap of the CertiK Platform.

Layer-based decomposition As we explained previously, SMT solvers often encounter the state explosion problem [12] when dealing with complex systems. To address this issue and apply CertiK Platform's techniques to a broader domain, the Foundation intends to introduce a novel layered-based approach. By developing the programming language support for building and composing layer-based specifications, the CertiK Platform enables a disciplined way of decomposing a complex system into a large number of small components and proof obligations. Without using layers, one might have to consider arbitrary interactions between the current component and its environment: an invariant held in one function can be easily broken when it calls a function defined in another module or communicates with another entity. A layered approach aims to sort and isolate all components based on a carefully designed set of abstraction levels so one can reason about one small abstraction step at a time. This can dramatically simplify the environment model that needs to be considered at each layer. In the past, the founders of the CertiK team have successfully built the world's first fully verified concurrent OS kernel, named CertiKOS [1, 2], using the Coq proof assistant [26]. CertiKOS consists of 6,500 lines C and assembly and is divided into more than 60 layers. The whole proof efforts are only about two person-years.

Based on these successes, the CertiK team plans to create a new layered-based verification framework that is suitable to reason about blockchain ecosystems. The key idea is to model the behaviors of the environment, including the context contracts and context nodes,

in a compositional way. Recall TheDAO attack example, the root of that bug is the neglect of the context's fallback functions. By proving that each layer component meets its layer specification under arbitrary context and linking all the proofs together, the end-to-end correctness of the entire system can be guaranteed.

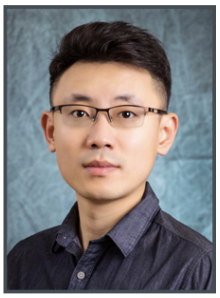
4 Roadmap and Project Plan

Figure 4 shows the roadmap of the CertiK Platform. The proof of concept of the CertiK Platform's techniques and community started in December 2017. The product development plan is quite aggressive. The Foundation plans to launch the alpha version of CertiK smart labeling and the layered verification techniques by the end of February 2018. These prototypes can then be played with and improved by an online community established by the CertiK team. To demonstrate the power of its approach, the Foundation aims for establishing business partnerships with at least ten organizations in the blockchain community once the beta version of the CertiK Platform is launched in April. These kinds of partnerships will be further expanded to reach at least 20 partners by the end of June 2018. The Foundation will then focus on developing new verification techniques, maintaining the CertiK Platform's community, and further spread this idea of decentralized certification.

The Distributor of CTK shall be an affiliate of the Foundation. At the early stage, the code development and the cluster establishment will take up the most significant portion of the budget. Research, legal, and financial consulting are also necessary. Thereafter, once the alpha version is complete, the Foundation will allocate more resources on building and maintaining the CertiK community to promote the ideology of the Foundation and expand the influence of the CertiK Platform. The Foundation will continuously provide tutorials about the CertiK Platform's services and provide detailed instructions for developers/miners/users on how to participate in the verification process. The Foundation plans to make a series of technical talk videos about the CertiK Platform and maintain them on social media. Further, the Foundation plans to utilize its academic resources to give lectures, hold seminars, and even organize summer schools on how to build trustworthy blockchain ecosystems.

In addition, the Foundation plans to build a strong development team by hiring at least 20 software engineers and research scientists. The Foundation will keep evolving the CertiK Platform, making sure its technologies are always leading the market.

5 Team Leaders



Prof. Ronghui Gu (Co-Founder)
Assistant Professor, Columbia University
Ronghui Gu is a tenure-track Assistant Professor of Computer Science at Columbia University. He obtained his Ph.D. in Computer Science from Yale University in 2016, where his dissertation won the Distinction Dissertation Award at Yale and was nominated for the ACM Dissertation Award. He obtained his B.S. from Tsinghua University in 2011. Prof. Gu is an expert in formal verification of system software. He was the primary designer and developer of CertiKOS, the world's first fully verified concurrent OS kernel. His OSDI16 paper on CertiKOS has been nominated and selected for publication in the Research Highlights section of the CACM.

Prof. Zhong Shao (Co-Founder)
Chair of Computer Science Department, Yale University
Thomas L. Kempner Professor, Yale University
Zhong Shao is Thomas L. Kempner Professor and Department Chair in the Department of Computer Science at Yale University. He earned his Ph.D. in Computer Science from Princeton University in 1994. During his early career, he was a key developer of the SML/NJ compiler and the main architect of its FLINT certifying infrastructure. In recent years, Shao has been a leading figure working on the highly visible research fields on cybersecurity, programming languages, operating systems, and certified software. He and his FLINT group at Yale have developed the world's first hacker-resistant concurrent operating system CertiKOS—a major milestone toward building cyber-physical systems that are provably free from software vulnerabilities. Shao is the author or co-author of 90 articles in top scientific journals and conferences.

Dr. Vilhelm Sjöberg (Research Scientist)
Associate Research Scientist, Yale University
Ph.D., University of Pennsylvania
John C. Reynolds Doctoral Dissertation Award Winner, 2016
Vilhelm Sjöberg is an associate research scientist at Yale University. He received his Ph.D. in Computer Science from the University of Pennsylvania in 2015. He is an expert in software verification, programming languages, and type systems. Currently he is interested in language support for layered verified systems like CertiKOS. Dr. Sjöberg is the winner of 2016 ACM SIGPLAN John C. Reynolds Doctoral Dissertation Award.

6 Risks

You acknowledge and agree that there are numerous risks associated with purchasing CTK, holding CTK, and using CTK for participation in the CertiK Platform.

6.1 Uncertain Regulations and Enforcement Actions

The regulatory status of CTK and distributed ledger technology is unclear or unsettled in many jurisdictions. It is impossible to predict how, when, or whether regulatory agencies may apply existing regulations or create new regulations with respect to such technology and its applications, including CTK and/or the CertiK Platform. Regulatory actions could negatively impact CTK and/or the CertiK Platform in various ways. The Foundation (or its affiliates) may cease operations in a jurisdiction in the event that regulatory actions, or changes to law or regulation, make it illegal to operate in such jurisdiction, or commercially undesirable to obtain the necessary regulatory approval(s) to operate in such jurisdiction.

After consulting with a wide range of legal advisors and continuous analysis of the development and legal structure of virtual currencies, the Foundation will apply a cautious approach towards the sale of CTK. For the sale of CTK, the Foundation is working with Tzedek Law LLC, a boutique corporate law firm in Singapore with a good reputation in the blockchain space.

6.2 Loss of Talent

The development of the CertiK Platform depends on the continued co-operation of the existing technical team and expert consultants, who are highly knowledgeable and experienced in their respective sectors. The loss of any member may adversely affect the CertiK Platform or its future development.

6.3 Failure to Develop

There is the risk that the development of the CertiK Platform will not be executed or implemented as planned, for a variety of reasons, including without limitation the event of a decline in the prices of any digital asset, virtual currency or CTK, unforeseen technical difficulties, and shortage of development funds for activities.

6.4 Other Risks

In addition to the aforementioned risks, there are other risks (as more particularly set out in the Terms and Conditions) associated with your purchase, holding and use of CTK, including those that the Foundation cannot anticipate. Such risks may further materialise as unanticipated variations or combinations of the aforementioned risks. You should conduct full due diligence on the Foundation, its affiliates and the CertiK team, as well as understand the overall framework and vision for the CertiK Platform prior to purchasing CTK.

References

[DeepSpec] DeepSpec: The science of deep specifications. http://deepspec.org/.

[YaleNews] CertiKOS: A breakthrough toward hacker-resistant operating systems. *Yale News*, 2016.

[IBTimes] CertiKOS: Yale develops world's first hacker-resistant operating system. *Internaltional Business Times*, 2016.

[YDN] Yale computer scientists unveil new OS. *Yale Daily News*, 2016. .

[1] R. Gu, J. Koenig, T. Ramananandro, Z. Shao, X. Wu, S. Weng, H. Zhang, and Y. Guo. "Deep specifications and certified abstraction layers." In *42nd ACM Symposium on Principles of Programming Languages (POPL'15)*.

[2] R. Gu, Z. Shao, H. Chen, X. Wu, J. Kim, V. Sjöberg, and D. Costanzo. "CertiKOS: An Extensible Architecture for Building Certified Concurrent OS Kernels." In *12th USENIX Symposium on Operating Systems Design and Implementation (OSDI'16)*.

[3] D. Costanzo, Z. Shao, and R. Gu. "End-to-end verification of information-flow security for C and assembly programs." In *37th ACM SIGPLAN Conference on Programming Language Design and Implementation (PLDI'16)*.

[4] H. Chen, X. Wu, Z. Shao, J. Lockerman, and R. Gu. "Toward compositional verification of interruptible OS kernels and device drivers." In *37th ACM SIGPLAN Conference on Programming Language Design and Implementation (PLDI'16)*.

[5] S. Nakamoto. "Bitcoin: A peer-to-peer electronic cash system." http://bitcoin.org/bitcoin.pdf.

[6] Wood, G., 2014. Ethereum: A secure decentralised generalised transaction ledger. Ethereum Project Yellow Paper, 151.

[7] G. Klein, K. Elphinstone, G. Heiser, J. Andronick, D. Cock, P. Derrin, D. Elkaduwe, K. Engelhardt, R. Kolanski, M. Norrish, T. Sewell, H. Tuch, and S. Winwood. "seL4: Formal verification of an OS kernel." In 22nd ACM Symposium on Operating Systems Principles (SOSP 09).

[8] E. W. Dijkstra. "Notes on structured programming." In Structured programming, pages 182. Academic Press, 1972.

[9] S. Peters, A. Danis, K. Elphinstone, and G. Heiser. "For a microkernel, a big lock is fine." In Asia Pacific Workshop on Systems (APSys 15).

[10] M. von Tessin. "The Clustered Multikernel: An Approach to Formal Verification of Multiprocessor Operating-System Kernels." PhD thesis, School of Computer Science and Engineering, The University of New South Wales, March 2013.

[11] Clarke, Edmund M., Orna Grumberg, and Doron Peled. "Model checking." MIT press, 1999.

[12] McMillan, Kenneth L. "Symbolic model checking." In Symbolic Model Checking, pp. 25-60. Springer, Boston, MA, 1993.

[13] Shao, Zhong. "Certified software." Communications of the ACM 53, no. 12 (2010): 56-66. Harvard

[14] De Moura, Leonardo, and Nikolaj Bjrner. "Z3: An efficient SMT solver." Tools and Algorithms for the Construction and Analysis of Systems (2008): 337-340.

[15] https://github.com/ethereum/go-ethereum/issues.

[16] King, Sunny, and Scott Nadal. "Ppcoin: Peer-to-peer crypto-currency with proof-of-stake." self-published paper, August 19 (2012).

[17] Buterin, Vitalik. "A next-generation smart contract and decentralized application platform." white paper (2014).

[18] http://heartbleed.com/.

[19] V. Buterin. Critical update re: Dao vulnerability, 2016.

[20] https://etherscan.io/accounts/c, Jan, 2018.

[21] How many contract are currently deployed on the ethereum blockchain?, September, 2016.

[22] IBM Sees Blockchain as the Next Big Thing.

[23] Application testing costs set to rise to 40% of IT budget.

[24] Quantstamp whitepaper.

[25] Solidified whitepaper.

[26] The Coq proof assistant. The Coq development team. http://coq.inria.fr.

[27] John C. Reynolds Doctoral Dissertation Award, 2016.