RuffNotes Update WhitePaper.md 2ce6a73 3 days ago

2 contributors

296 lines (170 sloc) 30.9 KB

Ruff IoT Blockchain Whitepaper [^1]

Working Draft, October 2017, Rev. 0.7.1

Summary

Ruff is a platform designed to improve commerce by combining the Internet of Things and blockchain technology. Ruff incorporates a distributed operating system with an open blockchain, using virtual business to business networks and a consensus algorithm to realize better offline solutions for information flow and product sourcing needs. In CAP choices, the traditional blockchain sacrifices usability to enhance the consistency and distribution fault tolerance. Ruff, through a combination of edge computing and blockchain applications, enhances usability so as to meet the Internet of Things demand for real-time flexibility . The key problem that we want to solve is about providing trusted interoperability and paid interoperability between different IoT device systems, and using these devices' computing power to build an open Ruff ecosystem.

Background

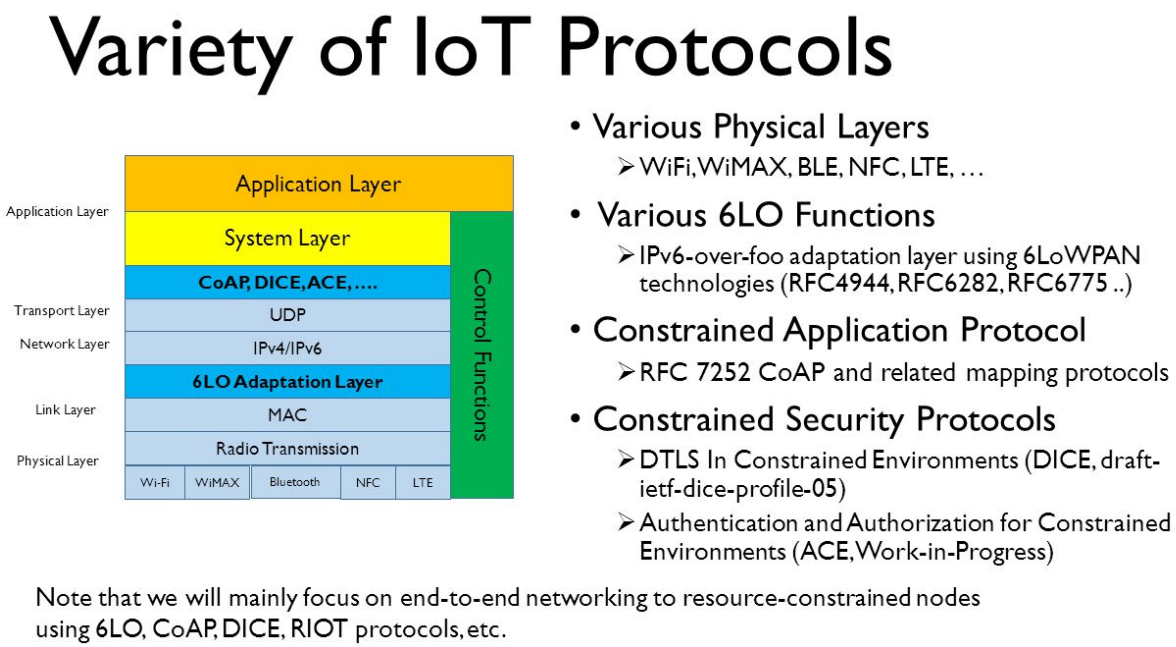
The Internet of Things often proves an isolated, closed system wherein wide area IoT and local IoT are unable to interact. Privatized industrial systems and IDC-based IT networks are difficult to connect. However, IoT data often requires high consistency and security, which is a problem that no centralized technology can readily solve. Modern IoT technologies are often accompanied by redundant nodes and mixed up clouds, but blockchain offers the possibility of a total solution to consistency and security issues. Unfortunately, blockchain development currently suffers from lack of infrastructure, high technical threshold, and excessive technical risk. So far, there are no mature solutions to the scalability problems that are frequently pointed out. As a result, the current distributed app development is sparse and remains at the virtual level, not yet having real world interactivity.

Fragmentation

From its beginning the Internet of Things has always been fragmented. For instance, it still takes a specific mobile app to open each of a growing number of bike sharing models, while in telecommunications you can generally call anyone on any carrier. Why is there no simpler solution for transportation options? There is no standard for equivalent nodes, and the result is break down. Moreover, beyond transportation tools, the IoT connected things we have around us today like doors, lights, alarms, coffee machines, and all so on are on closed isolated networks. This fragmentation is not just prevalent among different product types but also among products belonging to the same types as well. There is simply not enough homogeneity among IoT connected things for current models to offer the networkability we already enjoy in other ways through computers and mobile devices. Do these problems mean there is no possible solution at present? The answer is negative. What is needed is the introduction of operating system and middleware concepts, compatibility with the fragmented spectrum IoT hardware devices, and a unified programming interface.

Standardization

IT technology is standardized overall. For example, personal computers interact with servers through the http protocol presented on the browser, which is a form of standardization. The network for exchanges between different bitcoin nodes is also standardized. Only by constantly standardizing fragmented products will it be possible to unify communication among nodes or create a consensus network among nodes. The entire Internet of Things industry has been striving for standardization for over 20 years. On the physical level, standards exist for WiFi, BLE, Zigbee and so on, and at the level of industrial networks there are Modbus, Profibus, industrial ethernet, and so on, through in industrial networks different standards can be incompatible. Nevertheless standardization has not been achieved at the application level. Machine B and Machine B may be able to connect, but Machine A does not know any instructions for manipulating or requesting data from Machine B. Worse still, even within devices different drivers and different software vendors follow different proprietary protocols, making it difficult for applications to interact.



Ease of use

Sometimes commands will not be readable. WIth GPIO_14 high level - low level operations, serial port compatibility and switch roles may sometimes be unknown. You often need the following definition.

```
RCC_APB2PeriphClockCmd(RCC_APB2Periph_GPIOB, ENABLE);
GPIO_LED.GPIO_Pin = GPIO_Pin_1 | GPIO_Pin_11 | GPIO_Pin_14 | GPIO_Pin_15;
GPIO_LED.GPIO_Mode = GPIO_Mode_Out_PP;
GPIO_LED.GPIO_Speed = GPIO_Speed_50MHz;
GPIO_Init(GPIOB, &GPIO_LED);
```

However, this approach is difficult to be promote in the application engineer community, as its threshold is considerable, it is easy to write buggy code, and readability continues to remain low. Engineers prefer the following programming methods:

```
$('led-green').on();
$('led-red').off();
```

The above paragraph uses Ruff OS programming code, and more can be found at http://www.ruff.io. As of December 2017, 13,521 engineers worldwide have registered with the Ruff community, with more than half of them purchasing the Ruff board and deploying code. When there is a standard application layer protocol between nodes, the communication between nodes remains at the abstract level of interactions between devices, such as payments, requests, verifications, and so on. When multiple devices connect interactively the abstraction level will tend to be very high. At the most basic level, this is the interaction between different applications. Unifying standards in the application layer is the most urgent challenge to solve for the Internet of Things. It is also an important infrastructure consideration for the interconnectivity and interoperability of all things in the future. IoT standardization will not be implemented via a centralized cloud. Rather, starting from computing at the edge computing node level, the traditional module + cloud model will abandoned. This application logic will form a unified programming model based outside of firmware.

Reliable interoperability between IoT devices

Every smart device has an address, and when the device is sold the merchant enters this address into the hardware and provides a QR code for containing a private address key in the hardware's packaging. The control center, by obtaining the private key, sends the device a bind command (signed with the private key) that enables complete control of the hardware. After the binding, the control center can delete the device's private key and save it's own private key.

- Peer-to-peer control: The control center operates the device by initiating a control TX with its own signature to the device. (No chain involvement is required, but control center and equipment are online when needed.)
- Chain-state based control： When the control end can not establish a point-to-point connection with the device, the control end can consume tokens and write a "state change" TX or "control command" TX on the chain to operate the target device. (Each of these TX are equally necessary.) The target device can synchronize its status or control commands directly with the chain, or through a trusted light node (such as a bridge device). Blockchain thus solves the problems inherent in cloud operation and the maintenance costs and stability issues of connecting all devices to the cloud.
- Automated control： There is no need to use contracts to set the logic of "turn off the air conditioner when the temperature is below 15 ° C". Such custom control logic can be implemented using the console app built with traditional development language, reducing equipment support contract costs, but also reduce the likelihood of main chain jamming when running contracts.

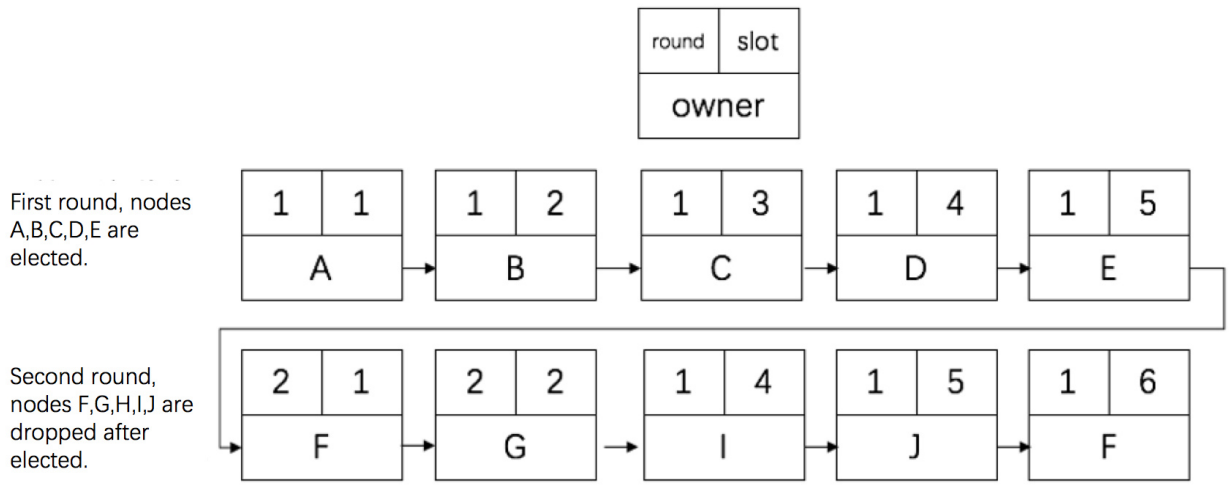
Based on the implementation of the above, multiple vendors can together form an open ecosystem on Ruff.

Time sequence data

Most IoT data is arranged in time sequences, making it a natural fit for the blockchain. Timestamp data can be applied to prevent replay attacks and solve problems caused by concurrent deadlocks. In the past this data was not effectively combined on isolated centralized networks, with the ultimate issue of data consistency in circulation. ERP, MES, WMS and other centralized systems simply cannot ensure consistent product traceability in the production, storage, and circulation processes. Ruff's edge computing node will synchronize the timestamp as the core data value, thus controlling the business logic within the local area network. Timestamps will be synchronized across the entire blockchain network, tracing the behavior of all nodes in the network at the same time to make the state of the network restorable at a given point.

Consensus mechanism

To account for the computing power of master devices in the Internet of Things we chose DPoS as Ruff's consensus algorithm. According to this algorithm, people who hold tokens across the network can choose block producers through a voting system. Anyone can participate in block production once elected. This system produces a block every 9 seconds. At any one moment, only one producer is authorized to produce a block. If the block is not successfully delivered within a certain time period it will be skipped over. Any full node can be a candidate for a specific transaction, and the system generates 105 proxy agents from candidates via a voting mechanism. The following figure shows the number of proxy agents under the circumstances of 5 possible block distributions:



Web resources are not free, and a reward will be issued whenever a block containing IoT contracts is generated.

Block generation in our system is counted based on 105 blocks per period. At the beginning of each period, the process of picking 105 proxy agents from candidates is called a round. There is a fixed number of 105 slots in a round, and there will be 105 blocks. All candidates can vote (VOTETX) to select the next round of proxy agents. Each block contains its own round and slot numbers in addition to its own height. Round numbers are continuous, while slot numbers may not be continuous.

Candidates must initiate a transaction type Vote on a turn, and select a given number of proxy agents from a valid candidate list. (If the total number of candidates exceeds 318, then previous round proxies will not be allowed to be re-elected.)

When the current round of block slots reaches 105:

- If a quorum for a vote cannot be obtained, all current proxies will be reelected by default for the next round.
- If a voting quorum is reached, the 105 proxies with the highest vote count will be selected for the next round.

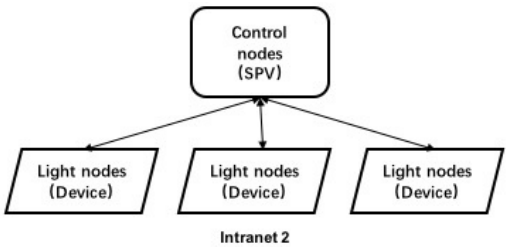
Based on the hash of the last slot block of the previous round, a definite list of round proxies' block orders can be obtained.

Each block, distributed by the agent responsible for the slot, is distributed at a 9 second interval from the previous block. Other agents not currently responsible for block distribution will be responsible for transaction collection, and when it is their turn to distribute must give priority to transactions not yet included. (Transaction collection is determined by round.)

If for example it is proxy agent N's turn but the agent's node is dropped, then the next agent will automatically distribute the next block after 18 seconds. A slot vacancy will be reported for the previous round, and for the current round the number of blocks still reach 105.

By analyzing the chain data for vote transactions and block slots, as well as round value. You can therefore determine which candidates do and do not work well. During a round, an agent may initiate a poll to disqualify certain candidates. If a proxy agent fails to deliver a block on time, for example, the agent will be blocked from distribution for the next 24 hours. Then all candidates will vote on whether to disqualify the candidate or not.

Chain control



In the typical structure above, the light nodes and the control node are on the same intranet. When a light node binds to the control node, the control node is allowed to control the light node from the master chain.

We use command signatures to achieve chain-based control.

- **Device initialization：**

The manufacturer of the light node device writes a public-private key pair into the device and then prints the private key on the device's instructions or package in the form of a two-dimensional code.

- **Device bindin：**

When activated the light node is placed in a network environment where it can connect to and communicate with the control node. The control node then initiates a binding command, which contains the public key of the control node. After the command is created, the device's private key must be submitted to sign the binding command. After the signature is complete the binding command is sent to the device. After receiving the binding command, the device verifies the signature. The verification is performed by recording the control node's public key. The control node can no longer store the light node's private key.

- **Control command verification：**

The command received by the light node will verify and agree to execute the command as long as it has the signature of its recorded control node.

- **Logging command history onto the master chain：**

If necessary, the control node will log all control commands that have been issued, and whenever communication with the backbone is restored, publish the history and control result records to the master chain.

Lightning chain differential mechanism

The control node can create a fixed format contract on the master chain using a CreateContract transaction. The contract is generally formatted in the terms "If you give me a certain number of tokens, I will allow you to use the following command under certain conditions." The successful creation of the contract will return to save the contract block height and transaction hash (together referred to as the Contract Address).

The user can initiate a Call transaction to a contract address. After saving the transaction to the master chain, the control node will continually check for transactions from the initiated contract. Once a transaction is detected, the control node checks to see if the user has enough tokens to complete it. If so, the transaction will be successful, initiating a Return transaction containing the result of the contract execution and a not-yet committed code for a Review transaction. Review is a multi-signed “Seq” signature transaction.

The Return transaction is confirmed by the master chain after the contract is successfully executed, at which time the user's tokens will be transferred to the control node's account. After some time, if the user feels that the hardware is not handling the subsequent commands correctly, then he can extract the Review transaction’s previously signed Return transaction, sign his own digital signature, and submit a Review transaction to the master chain, and the negative report will take effect. As this sequence is fixed, users will have to make registered complaints by due process.

If a particular contract gathers numerous such reports, the control app will interactively prompt the user that the contract has a large number of negative reviews.

Node classification

Internet of Things nodes are often very small computing units. Due to their power consumption requirements, their computing power is very low, their memory is very small, and the MCU can not exceed 512 kb. The Linux version of a node is at the router level, and the storage is likewise very small, with MCU only having 1M Flash memory. It is very difficult for such a node to participate in consensus processes. Therefore, the structure of the Internet of Things is necessarily composed of multiple nodes comprising a network. There will be one or more applications in this network, and applications will go through the application interface and interact on the blockchain. The computing power required for local applications will be drawn from the edge computing unit, potentially a gateway or router. Applications can manage the local network in a centralized or decentralized way and interact with the chain. According to this, we classify the nodes in the Ruff ecosystem as follows:

Light nodes (Executors)

The application control interface will request network authentication information, and after verifying correct implementation will issue a contract to the user, such as the release of property access rights. The light node role can be assumed by a simple device without storage capacity, and the cost can be as low as a few dollars.

Full nodes（Recorders）

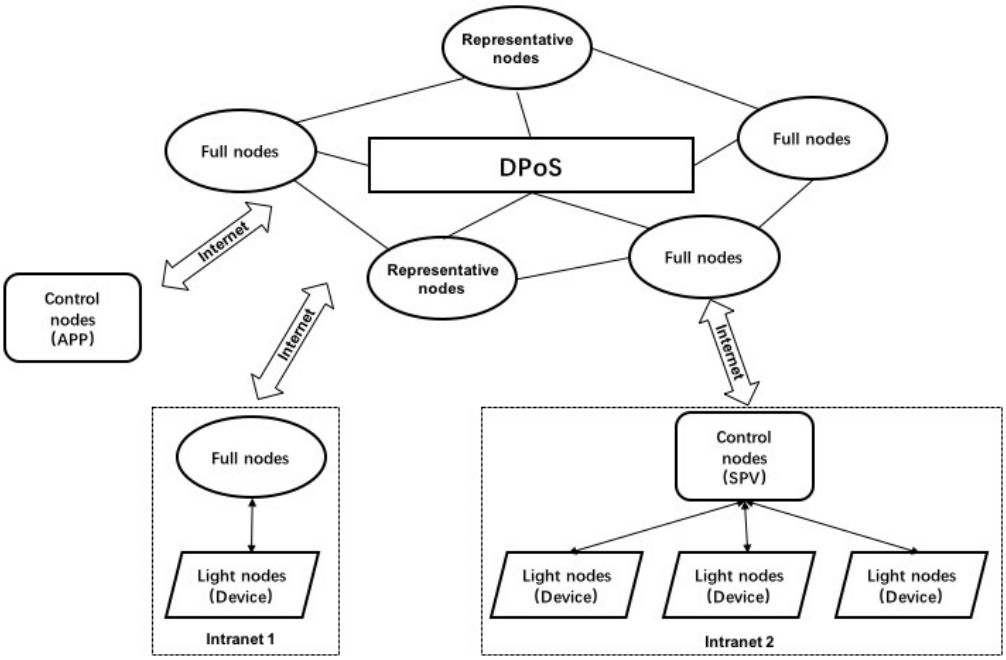
Full nodes will be able to record all information, participate in event registration or modify broadcasts, and vote for other nodes. Moreover, full nodes can become representative nodes. Devices with high performance will act have a greater role as nodes on the network. Due to the consensus algorithm of not using PoW and the low computational burden on the entire node, home smart devices such as a $200 HTPC, high speed routers, and NAS can take on this role.

Representative nodes (Arbiters)

The nodes with the most votes in every 105 nodes should bear the responsibility of enforcing the rules and packaging blocks. If found committing malicious acts, they will be rejected by the voters and lose the qualification of representatives. The representative node can obtain revenue from mining by distributing blocks.

Control nodes (SPV wallet)

Central nodes have a certain computing power and can save all blockchain data to quickly verify whether specific transactions have been logged on the chain. They can use P2P agreements to safely execute transactions. Compared with other nodes, they do not need to be online 24 hours per day. Normally they can participate via smartphone app or through devices with relatively low storage. (For example, low-end routers or smart home appliances.)



Smart Contract of Things

Contracts for smart things are established at the abstract level. The decentralize app (Dapp) interacts with the abstraction of the object via Ruff's OS, and then interacts with the Ruff chain. The combination of these two actions enables smart contract implementation in the real world. For example, offline real estate transactions can executed passed in this way, such as in hotels or sharing economy properties with respect to accessing door locks, rental equipment, the use of switches, and other functions that can be managed by contractual control.

In addition to the right to use such things, the value generated by the means of production may also be governed by contracts for things such as power generation, mining and manufacturing.

Common Dapp scenarios could be financial leasing, asset securitization, asset pledging, supply chain finance, property registration and real estate transactions, and so on. Traditional blockchain technology itself cannot back rights to use or production capacity matters with smart contracts.

Application scenarios

Property right transfer and rental

The control end can enable binding to a custom smart contract and thus realize **the exchange of rights to use equipment**. Contracts are structured in such formats such as "If you pay me 50 tokens, I will accept the following control commands you send within the next hour,"or,"If you agree to pay me 100 tokens, then within the next 3 hours I will deduct 1 of your tokens every 10 minutes." Chain support does not update the expansion of thusly formatted contracts. Rather, these are better suited to blockchain than smart contracts.

When this type of contract is used on the control side, a pre-signature transaction for quality feedback will also be given.

Open data transactions

Product, project life cycle management, and upstream and downstream industry chains can share real-time data so as to achieve traceability, quality tracking, capacity forecasting, and distribution functions.

Asset management and securitization

Assets generating value via transactions can include uses of generators, mining machines, charging piles, shared bicycles, retail equipment, and so on. This generate revenue can be securitized into circulation. Equipment and supply chain management that consume valuables can also use this mechanism for reverse circulation. Enterprises using IoT devices as a carrier to finance lease assets can set up the issuance of financial leasing trust plans. Leasing companies use their own professional advantage in the leasing business, and at the completion of the leasing project, the financially leased assets can be transferred out through the trust. At the same delegate trust proxies can, according to this system, entrust their legally held funds to investment companies. The investment company can act in accordance with the wishes of the principal in its own name, for beneficiaries in the financial leasing business. It is therefore essentially a designated purpose trust fund and a financial product that specializes in financial lease claims.

The financial leasing company separates or reorganizes the leasing right of the finance lease item and entrusts it to the trust company to sell to a specific investor in the financial market. The investor enjoys the rights to finance lease rental during the trust period. The trust company delivers the investor's purchase money to the financial leasing company. The financial leasing company discounts the receivable rent, eliminates the financing liabilities of the corresponding items on the balance sheet, and achieves off-balance-sheet financing, which not only solves the financing problem of the finance leasing company, but can also use asset lists to achieve the purpose of better adjusting the business asset structure. In addition, the related expenses such as the financial lease fees of the project and such will have already been assessed and charged, and the revenue of the financial leasing company is rapidly monetized. At the same time, for trust companies, it provides a carrier and mode of cooperation with financial leasing companies, which helps to improve the leasing capacity of trust companies.

The leasing company may consider the issue of products by the trust company as a fixed financing channel, and communicate with the trust company while conducting research on the financing project. After the completion of the project, the trust products can be issued simultaneously, reducing the occupation time of funds and simultaneously charging fees as needed. This thus creates a new profit model with brief duration and fast returns. For example, in the cooperation between Ruff and a photovoltaic operation and maintenance company, the production of photovoltaic power can be monitored and securitized in real time. Based on the status of photovoltaic power generation companies can understand the real-time status of equipment and asset production efficiency and make relevant data available to users, creating access to more transparent and credible product information.

Evaluation mechanism

When a contract is completed and the user does not obtain the desired result, the user will be able to submit an individually signed contract transaction to issue negative feedback. An example could be when a vending machine fails to dispense a beverage a user has paid for with tokens. The user can contact the seller of the vending machine for a certain period of time, so that the problem can be resolved and a bad review can be made by using the pre-signed transaction. When user feedback is overwhelmingly negative, the control system will issue user prompts accordingly.

RUFF tokens

Introducing Ruff’s built in public chain token: RUFF tokens, the virtual currency of contracts. The RUFF tokens will be the benchmark for incentives, spending, and trading within the Ruff public chain ecosystem.

Currency mechanism

Within the Ruff ecosystem, one or several types of tokens are generated as a standard of settlement. Consumers consume tokens during property or data transactions. Use of equipment rights and data generating transactions will also be settled using tokens. Any Ruff-based smart contract can claim its own token for settlement. However, in the IoT ecolosystem, providers who participate in verification, accounting and other activities within the IoT ecosystem will also use the default RUFF tokens, and consumers will also deploy contracts and consume resources using RUFF tokens.

For instance: If user A needs to request a resource or user permission from node B, user A needs to pay a certain amount of a particular token for this access, and further to pay an amount to node C for packaging the transaction.

Privacy and security

Since the edge calculation unit carries the vast majority of data, the data reported is determined by the application. Most of the logic of the application developers is offline, and desensitization of online data is controlled by developers.

Ruff's local ad hoc network is also decentralized. In a local application network, once the main application node fails, the application logic will pass to another node and continue for completion, thereby ensuring the consistency of the local application network.

The security of Internet of things is guaranteed by the OS's security. Ruff uses a symmetric key, and the key is not transmitted over the network. In addition, the chain network releases a one-time token based on the timestamp to the application network, which therefore resists replay attacks.

About Ruff

Ruff was founded in 2014 with edge computing at its core and the goal of replacing original embedded operating systems. Ruff's community has already grown to encompass the work of tens of thousands of developers, and is the industry's most prevalent IoT operating system. In addition to a core of technically skilled team members, team members have also been listed for Forbes China's 2017 30 Under 30 mention and other notable awards and accolades for achievements in business development and marketing.

Since establishment, Ruff has earned awards and recognition from industry figures including:

- The First Batch to Join the New Microsoft Accelerator Shanghai
- 2016 Tech Crunch Beijing (Innovation Challenge Champion)
- GiTC (Best Technology Innovator Award)
- 2016 Microsoft Innovation Summit (Best Investment Award)

Partnerships and support

Since its founding Ruff has already established partnerships with leading enterprises including:

- Microsoft China
- Schneider
- Baidu Cloud
- muRata

Tech team

Roy Li

A well-known expert in network security and the Internet of Things experts, Roy Li is also a senior instructor at Fudan University. As former technical director for Nokia (North America) he was responsible for OVI development and Symbian operating system research and development. He has offered security consulting services for security companies such as Symantec and VeriSign. He has also advised TNB, RealChain, and AIDOC.

Investors and Investment Groups

- Alex Goh

Alex Goh has more than 15 years of management and operations experience in China and Asia Pacific. Formerly as a 360 Cloud venture partner and President. He has served at HP and Dell in a number of departmental executive posts, working for Hewlett-Packard as vice president of enterprise groups and Internet + general manager, and as Dell's global business partnerships manager.

- DFund
  - Former Ink Weather co-founder, well-known blockchain and digital asset exchange investor, and Internet venture capital investment fund founders.
- Fenbushi Capital
  - Founded in 2015, Fenbushi Capital is the first China-based venture capital firm that exclusively invests in Blockchain-enabled companies. Our mission is to accelerate the inevitable future of Blockchain economy by supporting as many companies as possible.
- NEO Foundation
  - NEO is a blockchain platform and cryptocurrency which enables the development of digital assets and smart contracts.
- LinkVC

    - LinkVC focuses on working with blockchains, digital money and Internet financial services investments and projects.
- Concensus Capital
- Bits Angels

Advisors

- Wen Xin
    - Light in the Box (NYSE: LITB) Co-founder
    - Blog China Technologies Ltd. Co-founder
    - Partner, Ceyuan Ventures
- Wu Gang
    - CEO of Bixin
- Bian Jiang
    - Baidu Lead Product Manager，Senior Vice President；WeXFin founder & CEO，APlus Fund Managing Partner
- Kong Huawei
    - Principle, Institution of Computing Technology, Chinese Academy of Science, Shanghai
- Richard Wang
    - Famous Early Stage Investor, Partner at DFJ Dragon Fund.

Roadmap

Ruff is staged to become a new platform for IoT-based infrastructure development that is decentralized, open, open-source, and efficient. In Ruff's ecosystem, different actors can obtain their desired resources through a marketplace where the provision of resources is rewarded in tokens which can in turn be used to obtain resources, forming an autonomous economic entity.

Ruff public chain open-source plan

After firmly establishing the basic framework for advancing the Ruff project, we will open more of our core modules for open-source development, and look forward to gathering the involvement of more developers.

Time	Open-source implementation
Dec 2017	Large-scale, low-latency consensus algorithm
Apr 2018	Large-scale blockchain bookkeeping algorithm
Oct 2018	Smart contract algorithm
Mar 2019	Ruff public chain platform

[^1]: THIS DOCUMENT AND ANY OTHER DOCUMENTS PUBLISHED IN ASSOCIATION WITH THIS WHITE PAPER RELATE TO A POTENTIAL TOKEN OFFERING TO PERSONS (CONTRIBUTORS) IN RESPECT OF THE INTENDED DEVELOPMENT AND USE OF THE NETWORK BY VARIOUS PARTICIPANTS. THIS DOCUMENT DOES NOT CONSTITUTE AN OFFER OF SECURITIES OR A PROMOTION, INVITATION OR SOLICITATION FOR INVESTMENT PURPOSES. THE TERMS OF THE CONTRIBUTION ARE NOT INTENDED TO BE A FINANCIAL SERVICES OFFERING DOCUMENT OR A PROSPECTUS. THE TOKEN OFFERING INVOLVES AND RELATES TO THE DEVELOPMENT AND USE OF EXPERIMENTAL SOFTWARE AND TECHNOLOGIES THAT MAY NOT COME TO FRUITION OR ACHIEVE THE OBJECTIVES SPECIFIED IN THIS WHITE PAPER. THE PURCHASE OF TOKENSREPRESENTS A HIGH RISK TO ANY CONTRIBUTORS. TOKENS DO NOT REPRESENT EQUITY, SHARES, UNITS, ROYALTIES OR RIGHTS TO CAPITAL, PROFIT OR INCOME IN THE NETWORK OR SOFTWARE OR IN THE ENTITY THAT ISSUES TOKENS OR ANY OTHER COMPANY OR INTELLECTUAL PROPERTY ASSOCIATED WITH THE NETWORK OR ANY OTHER PUBLIC OR PRIVATE ENTERPRISE, CORPORATION, FOUNDATION OR OTHER ENTITY IN ANY JURISDICTION. THE TOKEN IS NOT THEREFORE INTENDED TO REPRESENT A SECURITY INTEREST.

[^2]: McKinsey Research

[^3]: McKinsey Research