symmetry

MDPI

Article

BGPcoin: Blockchain-Based Internet Number Resource Authority and BGP Security Solution

Qianqian Xing, Baosheng Wang * and Xiaofeng Wang

College of Computer, National University of Defense Technology, Changsha 410073, China; xingqian0110@hotmail.com (Q.X.); xf_wang@nudt.edu.cn (X.W.)

* Correspondence: bswang@nudt.edu.cn

Received: 16 August 2018; Accepted: 4 September 2018; Published: 17 September 2018

Abstract: Without the design for inherent security, the Border Gateway Protocol (BGP) is vulnerable to prefix/subprefix hijacks and other attacks. Though many BGP security approaches have been proposed to prevent or detect such attacks, the unsatisfactory cost-effectiveness frustrates their deployment. In fact, the currently deployed BGP security infrastructure leaves the chance for potential centralized authority misconfiguration and abuse. It actually becomes the critical yield point that demands the logging and auditing of misbehaviors and attacks in BGP security deployments. We propose a blockchain-based Internet number resource authority and trustworthy management solution, named BGPcoin, to facilitate the transparency of BGP security. BGPcoin provides a reliable origin advertisement source for origin authentication by dispensing resource allocations and revocations compliantly against IP prefix hijacking. We perform and audit resource assignments on the tamper-resistant Ethereum blockchain by means of a set of smart contracts, which also interact as one to provide the trustworthy origin route examination for BGP. Compared with RPKI, BGPcoin yields significant benefits in securing origin advertisement and building a dependable infrastructure for the object repository. We demonstrate it through an Ethereum prototype implementation, and we deploy it and do experiment on a locally-simulated network and an official Ethereum test network respectively. The extensive experiment and evaluation demonstrate the incentives to deploy BGPcoin, and the enhanced security provided by BGPcoin is technically and economically feasible.

Keywords: BGP security; origin authentication; blockchain; RPKI

1. Introduction

The current Internet is lacking security, the intended original function of which was to build connectivity between any node. The key reason why the Internet suffers from most kinds of network attacks (MITM attack [1], prefix hijacks [2–6], and so on) is the lesser network accountability [7–9]. Consequently, malicious attackers forge a connection, impersonate the original source and manipulate the flows. Bootstrapping accountability on the Internet is an arduous task, since not only the evolution to the next generation network with built-in security is hard to promote [10], but also the profits of out-of-band security enhancements are not enough [11–13], considering the burdensome and potential security-less management [14–16].

One of the most crucial problems is securing inter-domain routing [17–19]. The BGP security problem has been emphasized by many notable attacks and configuration errors (e.g., [1–3]). By examining which Autonomous System (AS) is authorized to originate which IP prefix, origin authentication provides an important step towards securing inter-domain routing [19–21]. A cryptographic certificate hierarchy, as the current prevalent paradigm for origin authentication, like RPKI [22] and ROVER [23], signs and attests to the binding of an IP-prefix and its origin AS (including the AS number and public key). The certified attestations for the hierarchical allocation and sub-allocation of IP addresses allow

BGP routers to perform origin authentication, detect prefix hijacks and discard illegitimate BGP route advertisements for which AS announce an IP prefix that is not legitimately owned by it.

Although RPKI has been advocated by the IETF Secure Inter-Domain Routing (SIDR) group and the frequent prefix hijacks motivate the adoption of RPKI to eliminate some risks, the potentially misconfigured, faulty or compromised RPKI authorities may introduce new ones and become one of the reasons for the disappointing adoption of RPKI [12]. RPKI [22] and ROVER [23] both provide a PKI-based trustworthy mapping from IP prefixes to the ASes. However, they both introduce a new dependence on centralized authorities.

A security tradeoff between centralized hierarchical systems, which are easy to control, but more vulnerable to misuse, and decentralized designs, which are more robust to abuse, but harder to manage, should be considered. In the absence of RPKI and ROVER, this process of reclaiming an IP prefix requires costly, bilateral negotiation or even litigation, which limits the power of the delegator of address space. With RPKI or ROVER, however, an authority can instantly and unilaterally take down an IP prefix, simply by revoking the Resource Certificates (RCs) or Route Origin Attestations (ROAs) that it issued. By employing certificate revocation lists and displacing certificates, the RPKI allows delegators of address space to revoke or withdraw their delegations unilaterally.

To address the problem of misbehaving authorities, researchers proposed solutions [24] like appending transparency logs to alarm about the changes of RPKI and adding dead objects to realize the consent of revocation. Although they provide the tool to alarm about and visualize changes of the RPKI repositories and the consent mechanism for revocations to balance the power of RPKI authorities, they cannot refrain from the following problem:

1. Firstly, the PKI for origin authentication does not have the function of resisting the malicious authorities to delete and overwrite objects they certify arbitrarily. Therefore, it is hard to maintain a consistent vision of information (RCs, ROAs, manifests) in the unsynchronized global view.
2. Recording or monitoring authority behavior by appending logs to alarm about the changes of RPKI is not sufficiently incentivized, and responding to the reported misbehavior takes time and requires manual effort.
3. Realizing the consent of revocation in RPKI requires a complicated and burdened collaboration between RC issuers (to sign) and relying parties (to validate), which may lead to a passive application to RPKI.

As another solution to extricate ISPs from IP hijackings or take downs, several existing data governance alternatives for routing security [25] have been examined for shortcomings. As we debate, current solutions cannot give an efficient and reliable solution.

1.1. Security and Function Requirements

A simple and secure solution to record and audit the Internet number resource allocation and validate BGP origin route attestation is needed, and it should meet the following demands:

- The infrastructure gives the global consistency of all allocation and updating of the resource to defend against the mirror world attack by presenting different misleading versions of the ownership directory.
- The attestation for auditing is tamper-resistant to avoid authority derecognition after resource delegation.
- An involved mechanism like the consent of revocation must avoid the delegator unilaterally withdrawing its delegations and ensure a consonant resource revocation.
- The solution must be cost-effective and easy to deploy for all involved parties. It requires no expensive introduction of new infrastructure or replacement of existing infrastructure.

1.2. Contribution

For the inherently error-prone and unsecure Internet, the blockchain is an outstanding candidate for reliable and trustworthy infrastructure [26], supporting security architectures [25,27]. We propose BGPcoin, a blockchain-based Internet number resource authority and trustworthy management solution that meets all the above security and functional requirements. BGPcoin maintains a set of smart contracts to compel every involved authority to operate the compliant Internet number resource transaction (including resource assignments and revocations under the consents). Therefore, BGPcoin primarily excludes the entity misbehaviors and misconfigurations that violate the contract. It records resource assignments and authorizations in an append-only and tamper-resistant way on the decentralized blockchain infrastructure where reversing and overwriting the resource transactions is forbidden. As a result, BGPcoin maintains a consistent view of information to avoid a mirror world attack like in RPKI. Moreover, BGPcoin requires neither online cryptography during routing, nor any modification to the BGP message formats. By tracing the usufructs of resource assets, which are changing continuously, the BGP routers match the source in origin advertisements, then detect and discard prefix hijacks. The natural financial incentives of the permission blockchain like Ethereum and the transaction framework for all parties involved provide an easy way to deploy the solution. Our contributions are as follows:

- We design BGPcoin, a blockchain-based Internet number resource management system, which contributes not only a tamper-resilience and transparent Internet routing registry, but also an origin repository and governance infrastructure for BGP security.
- We propose a lightweight and efficient origin authentication framework around the blockchain for BGP security, which has superior security resilience and is more easy and lightweight to deploy than the PKI-based origin attestation solutions.
- We implement a prototype in Ethereum, deploying it not only on a private blockchain, but also on an official Ethereum testnet. We evaluate the performance and scalability of BGPcoin in practice, and the result demonstrates that it has very reasonable computational requirements.

1.3. Organization

We firstly present the adversary model and several property requirements of our BGPcoin in Section 3. Section 4 presents the basic design of BGPcoin including not only the components of the system and the roles in it, but also the functions and basic resource operations. Our proposed BGP security architecture with BGPcoin is shown in Section 5. To evaluate the proposal, we explore it by an in-depth theoretical analysis with the RPKI in Section 6 and an implementation with the extensive simulation and details in Section 7. Section 8 presents related work.

2. Background

RPKI. RPKI (Resource PKI) is an Internet infrastructure resource management system implemented by IANA and deployed experimentally to support inter-domain routing security [19]. It is used to issue certificates and verify the validity of routing announcements. The management system uses the X.509 public key certificate framework to issue Certification Authority (CA) certificates for Internet number resources (including IP address and AS number) and binds the number resources to its public key. Then, the CA certificate is used to issue the End Entity (EE) certificate for the terminal entity. When an IP address resource holder (a CA certificate holder in the RPKI system) needs to authorize an AS to advertise route reachability information for its specific IP address prefix, the IP address resource holder issues an EE certificate with the private key corresponding to the CA certificate. Then, the IP address resource holder signs a Route Origin Attestation (ROA) that binds its IP addresses to the AS number, with the private key corresponding to the EE certificate. RPKI validators verify the ROA to determine whether an AS is authorized by the holder of an IP address resource to advertise routing information corresponding to the IP address prefix.

In general, RPKI reflects the administrative hierarchy currently for allocating the Internet Number Resource (i.e., IP address and AS number). As the root, the Internet Assigned Number Authority (IANA) distributes resources to regional Internet registries (RIRs). All five RIRs, as the administrative authorities, allocate and assign the globally unique identifiers (IP address numbers and ASNs), all the way to Internet Service Providers (ISPs). Accompanied by X.509 certificates, RPKI forms a top-down chain of trust. Each CA has a respective RPKI repository publication point, which publishes these certificates and authorities. Supported by such distributed repositories from all the CAs, Relying Parties (RPs) retrieve and validate an attestation or authority.

While the highly centralized structure of the RPKI provides security guarantees against external threats, e.g., prefix hijacking, it may introduce new attack vectors exposed by the internal vulnerabilities. Since RPKI allows authorities for the unilateral revocation of allocated resources, they are able to revoke the ownership of a set of IP addresses and then effectively isolate the devices that have been using these IP addresses from any access to the Internet. Moreover, an authority can create some "mirror worlds" by presenting different misleading versions of the ownership directory.

Blockchain-based distributed ledgers technology and smart contract: The concept of a blockchain was first introduced with Bitcoin, which was designed to be a globally consistent, append-only ledger of financial transactions [26]. The solution for Internet number resource management in this paper relies on Ethereum, a public and permissionless blockchain-based distributed ledger platform with an open source virtual machine named EVM [28]. In EVM, every user holds a blockchain address that acts as a public user identity and uses the corresponding private key to sign his or her transactions. After the transactions are submitted to the blockchain, the miners examine their validity as organized as a block and chain to the blockchain transitively using a cryptographic hash function. The verification (mining) process is based on the consensus paradigm in which miners compete for the right to append the transaction block to the blockchain, and the winner is rewarded with a certain amount of coins plus the transaction fees included in the block. In this way, a public (permissionless) blockchain like Ethereum realizes a distributed, irreversible and irrefutable database of transactions that can be accessed/managed by users that do not trust each other and without a common trusted third party.

Ethereum's currency is used not only for ordinary transactions, but also to activate executable code that manipulates the blockchain state. Code is organized as Ethereum contracts and their Ethereum addresses. They are user-defined applications and enforced to run exactly and automatically as programed by the consensus protocol of Ethereum [29]. The language defined in Ethereum is Turing-complete, allowing arbitrary computation in the blockchain. To prevent malicious code from wasting computational resources, message senders must pay additional fees called gas that recompense miners for their storage and computational costs.

3. Desired Properties and Adversary Model

Adopting any security mechanism for inter-domain routing should comply with the natural prerequisite that it should never undermine the routing system or worsen the security of the BGP system itself. Taking them into consideration, our system should address at least the following properties:

- **Transparency and auditability:** The states of all the Internet resources are unambiguous and accessible for all inter-domain routers and any auditors.
- **Timeness:** This security enhancement should have an acceptable performance and scalability to maintain the accountable resource mapping objects up-to-date.
- **Security:** The system is resilient to tamper attacks from manipulators.
- **Incremental deployment:** The system must provide substantial benefit even with limited adoption.

3.1. Adversary Model

The target of the adversaries in our model is prefix/subprefix hijacking. That means a network operator that has not been authorized to originate a prefix announces in the BGP route massage that the prefix is bound to its own AS number (ASN), and this false route origination is legitimized successfully and accepted by the BGP system. We define the external attackers and the internal attackers toward BGP routes and the BGPcoin system, respectively. An external attacker takes an action like (1) directly fabricating/falsifying the BGP routing update massage to present himself/herself as the owner of a prefix and (2) attacking the consensus protocol of the BGPcoin blockchain infrastructure to alter the processing of the resource transactions. An internal attacker is additionally able to (3) hold an authority address and deny or send resource transactions.

4. Design of BGPcoin: Roles, Components and Functions

BGPcoin is a system hosted with the Ethereum blockchain to extend the BGP security architecture. The system is controlled by a set of smart contracts that allows entities to manage its Internet number resource (Internet address and autonomous system number) and give a reliable origin advertisement source for BGP security.

BGPcoin has five types of participants:

1. The Internet Assigned Numbers Authority (IANA) as the base authority who builds the `BGPcoin_base` contract;
2. Regional Internet registries as the allocator of two types of resources (IP addresses and ASNs)
3. National/local Internet registries as the allocatee and leaser to assign the resources to autonomous systems
4. Internet Service Providers (ISPs) as the leasees of the resources who have the right to publish the ROA binding their IP address resource and ASN resource
5. All other entities, i.e, the border gateway routers as the client to retrieve the ROA records and any volunteer as the anomalous detector and checker.

The contracts of BGPcoin contain several primary functions:

- **Resource trading**: The resource management for BGP security is implemented by operating eight BGPcoin resource trading functions, register, allocate, assign, update and revoke for IP address resource and register, allocate and update for the ASN resource.
- **Aggregated Internet address repositing and updating**: To store efficiently in a way that compresses the number of resource entries on the blockchain, BGPcoin aggregates the continuous IP-prefixes with the same owner, the same tenant (in the case of leasing them) and the same state.
- **Resource searching**: The client of BGPcoin accomplishes every IP validation by operating the resource search on the blockchain. The IP ownership record is displayed by the newest operation log in the latest transaction about the corresponding IP address resource.

BGPcoin has four types of trading operations for the Internet Address resource and three types of trading operations for AS numbers resource, as shown in Table 1 and Figure 1. We note that an IP address resource assignment refers to an established lease with a valid period. The owners of certain resources are in charge of updating resource states after their expiration date. An IP revocation operated by its owner during its leasing period is only permitted if the resource leasee's consent is attached.

Table 1. Semantics of BGPcoin trading operations.

Operation	Semantics
IP register	IANA→RIR:$< IPB, \varnothing >$
IP allocate	xIR→xIR:$< IPB, \varnothing >$
IP assign	xIR→xIR/ISP:$< IPB, \varnothing >$
IP revoke	xIR/ISP→xIR:$< IPB, \varnothing >$
ROA add	xIR/ISP:$< IPB, ASN >$
ROA delete	xIR:$< IPB, ASN > \rightarrow < IPB, \varnothing >$
ASN register	xIR:$< ASN, -, - >$
ASN allocate	xIR→ISP:$< ASN, stime, period >$
ASN update	ISP:$< ASN, stime', period >$

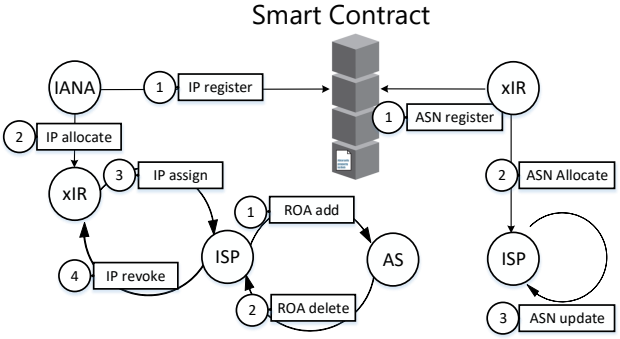


Figure 1. Operators for IP address and AS number resource in BGPcoin. xIR represents one of the following: Regional Internet Registry (RIR), National Internet Registry (NIR) and Local Internet Registry (LIR).

4.1. BGPcoin Resource Register and Trading

The IP address resource and ASN resource become valid after their registration and then are traced according to their trading records. ASN is a general name asset with the general register and transfer operations in its state period, like the name asset in other blockchain-based naming systems. An AS resource is modeled as:

$$< a, RIR, owner, stime, vperiod >$$

where a is the ASN, and once a RIR rir registers it, this AS resource <$a, rir, rir, 0, 0$> will be added to the BGPcoin storage. Then, rir could allocate this resource to a NIR/LIR/ISP ow with a valid period vpd, and the state is changed to <a, rir, ow, st, vpd>, where st is the present time. Once the resource is out of the expiration date, the owner could pay an amount of rent and update the period of validity by sending an `ASNupdate` transaction to the BGPcoin contract.

In the BGPcoin contract, we design five states for the IP address resource: *Unregistered, Registered, Allocated, Assigned, Binded*. The state machine model for the IP address resource, as shown in Figure 2, defines the state transfer rules once some operation is related to the IP address resource.

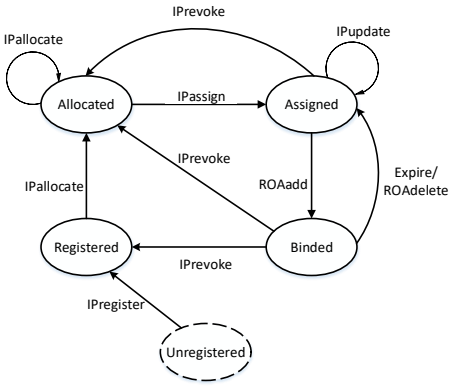


Figure 2. State machine model for the IP address resource.

We model an IP address resource as:

$$< \pi, state, RIR, NIR, owner, leasee >$$

where the IP prefix π has four layers of possessors: two authorities RIR and NIR, one owner and one leasee. The state changes in the state machine according to what kind of transaction happens with the IP prefix, as shown in Algorithm 1. Firstly, IANA registers an IP prefix π to one of five RIRs *rir*, and the state of this IP address resource is $< \pi, Registered, rir, 0, rir, 0 >$. Then, *rir* allocates this IP address resource to am NIR *nir*, and the state changed is $< \pi, Allocated, rir, nir, nir, 0 >$. The *nir* continues to allocate to an LIR *lir*, as in the state $< \pi, Allocated, rir, nir, lir, 0 >$. The end user *eu* requests the usufruct of this IP address resource from the owner *lir*, and the state is changed to $< \pi, Assigned, rir, nir, lir, eu >$. Once the end user places this IP address resource in service for its one AS, then the state is $< \pi, Bound, rir, nir, lir, eu >$, and an ROA $(\pi; a)$ is added to the BGPcoin storage, where a is the ASN hold by the end user *eu*.

Algorithm 1: IPregister function in the BGPcoin_base contract.

Input: BGPcoin contract address *base_addr*, IP asset structure *BGPcoin_IPchain* that is stored in the BGPcoin contract, *_IPS* that IANA are going to register with the prefix length *_prefixlen* and *RIR* that is the regional Internet registry that registers it.

Output: TURE or FALSE

```
Require msg.sender to be IANA;
Require all the IP prefixes covering the register are all unregistered;
if BGPcoin_IPchain.IPB[_IPS].IPIndex is 0 and
  BGPcoin_IPchain.keys[_IPS].prefixlimit <= _prefixlen then
    Insert the registered IPBData information;
    BGPcoin_IPchain.keys[_IPS].prefixlimit = _prefixlen;
    BGPcoin_IPchain.keys[_IPS].deleted = FALSE;
    BGPcoin_IPchain.size++;
for i = _prefixlen − 1; i >= Prefixlen_Min; i++ do
    if BGPcoin_IPchain.keys[IPS >> (32 − i) << (32 − i)].prefixlimit <= i then
        BGPcoin_IPchain.keys[IPS >> (32 − i) << (32 − i)].prefixlimit =i+1 ;
        BGPcoin_IPchain.keys[IPS >> (32 − i) << (32 − i)].deleted = TURE;
    else
        break;
Emit the IPRegister Event;
final ;
return TRUE;
```

Trading principle: As we all know, RIRs have the responsibility of "conservation", which means that their address space needs to be allocated according to the actual usage requirements. Each level of IP address space requirements of the network hierarchy enables the optimal allocation of address space to meet IP address capacity requirements adequately while minimizing the waste of address space. In addition to capacity requirements, efficient allocation practices are also critical to maximizing route aggregation to reduce routing protocol traffic and routing table overhead [30]. Fortunately, the design of BGPcoin follows the "conservation" principle. By increasing the IP division cost in `IPallocate/IPassign` operations when separating smaller IP blocks from a higher-level IP block, BGPcoin just maintains the principle of retaining large unallocated address space blocks as available for alternative larger allocation requests. Assume that a higher-level IP block with its IP prefix as when the owner allocates/assigns π' to one entity from his/her IP block π; the operation cost according to the IP space difference is shown in Figure 3. The row-coordinate is the IP space difference, represented by $len_{\pi'} - len_{\pi}$, and the y-coordinate is the transaction gas of the corresponding trading operation. As Figure 3 shows, the transaction cost is linear with the IP space difference. Therefore, the allocators take the financial cost of allocating/assigning into consideration, as well, and persist with the allocating principal.

$$\pi = (IP_{\pi}, len_{\pi})$$

and its sub-prefix:

$$\pi' = (IP_{\pi'}, len_{\pi'}), len_{\pi'} > len_{\pi}$$

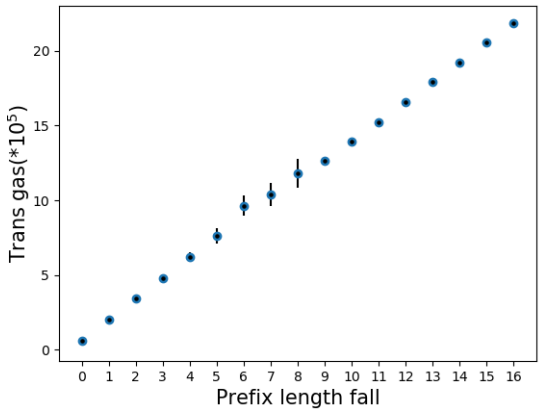


Figure 3. IP division cost for IPallocate/IPassign.

4.2. BGPcoin Resource Revocation

In BGPcoin, every ASN resource has its valid period. Once expired, ASN will be recalled back to its RIR, and the holdership in the *ASNData* becomes invalid. If the holder intends to renew the holdership, he/she pays an amount of fee offline and then requests RIR to send an *ASNupdate* transaction to renew the valid period.

As for IP address space, BGPcoin has a consent-based revocation to help revoke IP address space bilaterally. Our method corrects the power imbalance in RPKI, which actually allows an authority to revoke an IP address space that has been issued to its descendant unilaterally. Every IP address resource has a consent list to indicate each layer's possessor's consent to recall the resource to higher possessors, i.e., an IP address resource is modeled as:

$$< \pi, state, RIR, NIR, owner, leasee, consent\_vector >$$

To indicate its consent to have its resource taken away, a possessor sets its consent flag in the resource. If one of the possessor wants to revoke the IP address resource, it needs all the consents from these resource's lower possessors. If one consent is refused, this indicates that there is an argument between the revoker and the consent refuser, and BGPcoin raises an alarm and reminds the related possessors to inspect the situation out-of-band.

5. BGPcoin Architecture for BGP Security

We firstly design and implement a set of smart contracts that dictate the protocol of the system and act as the interface to the blockchain-based resource management. Secondly, we implement the client that interacts with the smart contracts to trade the Internet number resource and retrieve the resource records.

We give a view of processing architecture around the BGPcoin contracts in Figure 4, and the set of smart contracts in our system consists of four sub-contracts:

- `BGPcoin_base`: This contract is a base contract that IANA is in charge ofregistering IP address resources and allocating to xIRs. It also allows RIRs to register ASN resources.
- `BGPcoin_client`: IANA builds this contract to provide the operating interface for xIRs allocating IP address resources to its sub-xIRs, then to ISPs and allocating ASN resources to its sub-xIRs or ISPs; for ISPs for allocating IP address resources to its sub-ISPs.
- `BGPcoin_ROAserver`: IANA builds this contract to provide the operating interface for IPblock holders assigning their IPblock resource to their own ASes.
- `BGPcoin_checker`: This contract not only serves for all kinds of resource state searching, but also provides an incentive auditing mechanism to report anomalies like policy violation.

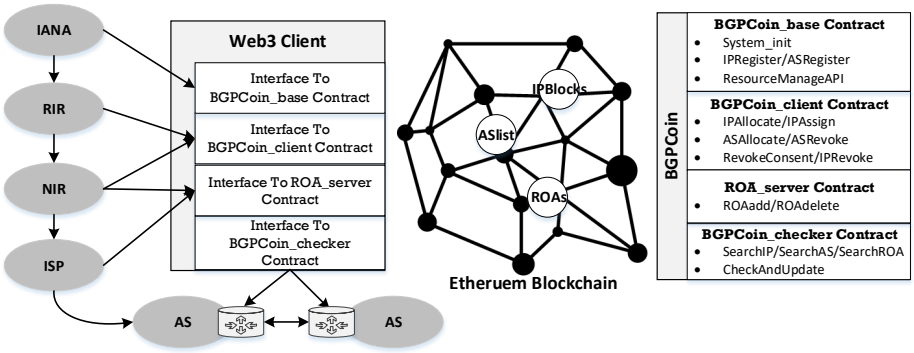


Figure 4. Architecture of BGPcoin resource management.

BGPcoin records the resource assignments and authorizations in the form of transactions on the Ethereum blockchain [31]. A corresponding distributed ledger is created and maintained through a network consensus for tracing the state of a resource asset (its ownership and usufruct). Apparently, each resource asset is solely owned or leased on the ledger.

Bootstrap setting: Every operator with its Ethereum account address that owns the Internet number resource in BGPcoin is mapped from a real administrative authority of the Internet. Before IANA establishes the BGPcoin system, all the legitimate resource authority organizations should be mapped to their Ethereum account addresses. All the participating organizations (IANA, xIRs and ISPs) authorize their own Ethereum account, which is collected as a signed profile and publicized by IANA.

As a result, one transaction between two Ethereum accounts, as an attestation, shows a resource allocation from its present owner (like RIR) to its next owner (like NIR) or a resource assignment from its owner (like LIR) to the leasee (i.e., ISP). After BGPcoin is launched, any change of the Ethereum account address for a resource authority is still allowed. Once a resource authority has changed its

Ethereum account in the case that the account's private key has been stolen, several transactions are submitted to transferall its resource from its former account to its new account. In this way, the resource authority has a new account representing itself, and the safety of its resource is guaranteed.

5.1. BGPcoin_Client

Every resource authority organization maintains a client with an Ethereum address to interact with the `BGPcoin_Client` smart contract to make operations on its own resource. The use right of a resource can be transferred from its RIR to an LIR and then be lent to some ISP for the use of binding with ASN. Every legal transfer operation triggers an event record on the blockchain.

Owing to the aggregated Internet address repositing and updating function in BGPcoin, these IP addresses that are contiguous and have the same owner and the same status compose a whole IP block with one IP prefix representing the smart contracts. As Algorithm 2 demonstrates, if the owner of this IP block allocates a part of IP addresses to its sub-xIR authority, it firstly divides the IP block into several smaller IP blocks and then allocates one of them to the sub-xIR.

Path_end advertisement: We support path_end record and filtering in BGPcoin by adding a list of approved adjacent ASes for every AS in the ASlist (the list we define for every AS to store its adjacent ASes)in BGPcoin. An AS's owner can update or delete the path_end records of its AS by sending `adjASupdate` or `adjASdelete` transactions to the `BGPcoin_Client` contract. Path_end advertisement helps one AS protect its valid last AS hop on the advertised BGP path. Therefore, the attacker cannot claim to be directly connected to the victim origin AS. Consequently, the path the attacker announces to a victim origin AS to his neighbors has a length of 2 at least.

Algorithm 2: IPallocate function in BGPcoin_client contract.

Input: BGPcoin contract address *client_addr*, *BGPcoin_IPchain* that is the IP assets storaged in the BGPcoin contract, $\_IPstart$ that its owner are going to allocate with the prefix length $\_prefixlen$ and *allocatee* that its owner allocates to.

Output: TURE or FALSE

1 Search the IP prefix IPS that has been registered in BGPcoin that covers the $\_IPstart$ with the prefix length $\_prefixlen$;
2 Require msg.sender is $BGPcoin\_IPchain.IPB[IPS].owner$;
3 Require $BGPcoin\_IPchain.IPB[IPS].state$ is *Registed* or *Allocated*;
4 **if** $\_IPstart \neq IPS$ *or* $\_prefixlen \neq BGPcoin\_IPchain.IPB[\_IPS].prefixlen$ **then**
5 Divide the IP asset with the IP prefix IPS to $k+1$ small IP assets that contain the IP asset with the IP prefix $\_IPstart$, where $k = \_prefixlen - BGPcoin\_IPchain.IPB[\_IPS].prefixlen$;
6 set the IPBData state of the IP asset with the IP prefix $\_IPstart$ to the new owner;
7 Emit the *IPAllocate* Event;
8 **final** ;
9 **return** $TURE$;

5.2. BGPcoin_ROAServer

We design a sub-contract to register and audit the binding from the ASN to the IP block for their owner. Every resource user maintains an Ethereum client for the contract, whose AS resource and IP address resource have been announced by the `BGPcoin_Client` contract. A legal binding operation changes the state of the IP address resource from "assigned" to "bound" and adds an ROA to the ROAServer contract. ROASever checks the operation validity according to the usufruct of resources and then updates the ROA repository.

Route origin advertisement. A valid ROA has the structure:

$$(< \pi, stime, vperiod >; a)$$

which means the AS a originates the IP block π in the time period from $stime$ to $stime + vperiod$. A valid ROA represents that the bound IP block is leased to the AS owner and had not been bound before. Otherwise, the binding operation fails, and the corresponding ROA would not be generated. This approach to submitting and recording ROAs gives a simple and clear way to publish and attest to the route origin data. Compared with the method of downloading the validated ROAs from RPs in RPKI, ours allows an edge router to directly request IP address resource bindings from the Ethereum client in its own AS. Besides, the Multiple Origin Autonomous System (MOAS) conflicts never appear in BGPcoin.

Aggregated ROA or minimal ROAs? Aggregation of prefix information reduces the number of entries BGP has to carry and store. Sometimes, for traffic engineering, an AS announces subprefixes of its larger prefix, which is called de-aggregation. We avoid simply exploiting the harmful maxlength attribute in RPKI [32], which was originally designed for de-aggregation in ROAs. According to the longest-prefix-match principal, a forged-origin subprefix hijack is able to act successfully by de-aggregating a subprefix from a valid ROA with its maxlength attribute and trigger a misleading BGP route update. Moreover, BGPcoin supports the "compressing minimal ROAs". By realizing three origin advertisement/authentication modes, we deal with the three different types of origin advertisement situations and forbid the subprefix hijack. For a valid ROA $(\pi; a)$, where $\pi = (IP_\pi, len_\pi)$,

1. **Aggregated ROA mode:** If ROA $(\pi; a)$ has its maxlength$len'_\pi (len'_\pi - len_\pi > 0)$, that means a set of sub-ROAs with the amount of $\sum_{i=0}^{len' > len} 2^i$. The a's owner adds the ROA $(\pi; a)$ by sending an `addAggrROA` transaction to the address of the `BGPcoin_ROAServer` contract, who adds ROA $(\pi; a)$ to the BGPcoin storage and then changes the prefixlimit of IP_π to len'_π, which means:

$$BGPcoin\_IPchain.keys[IP_\pi].prefixlimit = len'_\pi;$$

2. **Minimal ROA mode:** If only ROA $(\pi; a)$ and some of its subROA $(\pi'; a)$ are valid, where $\pi' \subset \pi$, the a's owner adds $(\pi; a)$ as usual, and then, it sends an `addMinROA` transaction of $(\pi'; a)$, which adds ROA $(\pi'; a)$ to the BGPcoin storage, and then changes the prefixlimit of $IP_{\pi'}$ to $len_{\pi'}$, and the default nonexistence flag is false, which means:

$$BGPcoin\_IPchain.keys[IP_{\pi'}].prefixlimit = len_{\pi'};$$
$$BGPcoin\_IPchain.keys[IP_{\pi'}].deleted = false;$$

3. **Compressing minimal ROA mode:** If ROA $(\pi; a)$ has its sub-ROAs $(\pi_i; a), i = 1 \cdots k$ where $\pi_1 + \cdots + \pi_k = \pi$, the a's owner adds $(\pi; a)$ as usual and then divides π into k subprefixes.

We gives three examples to explain that different ROAs are suitable for what kinds of modes. Suppose BGPcoin had the ROA:

$$ROA : (10.1.0.0/16 - 17, AS64469)$$

with maxlength 20, and that AS 64469 originates the following three BGP announcements:

$$"10.1.0.0/16 : AS64469"$$
$$"10.1.0.0/17 : AS64469"$$
$$"10.1.128.0/17 : AS64469"$$

BGPcoin adds this ROA in aggregated ROA mode. Suppose some entity issues the minimal ROA:

$$ROA : (\{168.122.0.0/16, 168.122.225.0/24\}, AS111)$$

BGPcoin records this ROA in minimal ROA mode by adding one ROA $ROA : (\{168.122.0.0/16\}, AS111)$ and one minROA:

$$minROA : (\{168.122.225.0/24\}, AS111)$$

Consider the following ROA, which consists of a set of minimal ROAs in it:

$$ROA : (\{87.254.32.0/19, 87.254.32.0/20,$$
$$87.254.48.0/21, 87.254.56.0/21\}, AS31283)$$

BGPcoin would record those minimal ROAs in compressing minimal ROA mode and compress it in one ROA. According to the basic classified modes, every ROA could be recorded in one of them or in a mixed mode.

5.3. BGPcoin_Checker

Origin authentication: To achieve the origin authentication, every AS maintains a BGPcoin cache-client to cache the ROAs it learns from the blockchain. Different from the RPKI that downloads the validated ROAs from RPs, BGPcoin allows the border router to request ROA mappings from the cache-client in its own AS directly. There is two ways to update the cache:

1. The cache client follows and syncs all the state transfer events related to ROAs that are derived from the increasing BGPcoin transaction event logs on the blockchain, to its current cache. The local cache, that is constructed like in RPKI, is pushed periodically to all border routers for origin authentication.
2. When a border router covering ROA queries a pair of IP prefixes and its claim to be the origin AS according to a new BGP update massage it receives, the cache client sends a `SearchROA` transaction to the `BGPcoin_Checker` contract to retrieve a corresponding ROA.

We model simplistically a BGP route as an IP prefix π and an origin AS a. A BGPcoin cache-client has a local cache of the complete set of valid ROAs, which are used to classify each route $(\pi; a)$ learned in BGP update massages into one of three route validation states:

- **Valid:** A route $(\pi; a)$ is valid if there exists a matching ROA, which meets three requirements: (1) a matching origin AS a; (2) a prefix P that covers prefix π; and (3) according to the three ROA modes, one of the conditions should be satisfied: (a) if there exists a specified maxlength, then it is no shorter than the length of π; (b) else, if the prefix P exists in *BGPcoin_IPchain.IPB*, then the prefix π should exist in *BGPcoin_IPchain.keys*; (c) else, if the prefix P has been divided into subprefixes, then the prefix π should be in *BGPcoin_IPchain.IPB*.
- **Unknown:** $(\pi; a)$ is unknown if one of the conditions below happens: (1) the prefix π or the AS a has not been registered in BGPcoin; (2) the prefix π has been allocated or assigned to the AS a's owner, and there is no valid covering ROA that contains a bound prefix that covers π.
- **Invalid:** $(\pi; a)$ is invalid if neither of the conditions required in the valid validation state and unknown validation state are satisfied.

Path_end authentication: The cache-client periodically syncs records of adjacent ASes from the ASlist storage in BGPcoin and pushes the resulting AS relation white list to BGP routers. Through checking from which ASes an origin AS can be reached, routers discard BGP advertisements on whose advertised path the end hop AS before the origin AS does not appear in the list specified by the origin AS. This filter enables BGPcoin to eliminate not only prefix/subprefix hijackings, but also next-hop AS attacks.

The BGPcoin_Checker contract not only serves for the border gateway routers and their cache-client to retrieve the ROA records and the adjacent AS information, but also incentives any volunteer as the detector towards the anomalies of BGPcoin. In other words, anyone is allowed to send a query transaction to check the ROA records, and if he/she discovers some anomaly in ROA records, he/she can report it to this contract and get a reward.

Checker incentivization: Network operators and registries have few incentives to remove obsolete registry objects like expired ASNs and ROAs. Though it has no effect of enlarging the attack surface toward the BGP routing, the increasing volumes make the resource searching execution cost more gas. Furthermore, it frustrates the initiative of the autonomous domain to deploy BGPcoin. Therefore, BGPcoin considers an incentive checking mechanism to encourage any auditors reporting the obsolete objects and potential inconsistent ROA caused by the situation that the bound AS is updated by changing its owner, but the owner of the bound IP block is not changed in-step.

We follow the report-and-reward design in [33]. The BGPcoin contracts help every AS resource and every ROA maintain its own balance used for escrow funds and to provide rewards. The balance for punishing each authority inaction helps ensure the availability of such resources. To simplify the exposition, the auditors get a reward if they rightly:

- check and delete the expired ASNs and ROAs according to the valid period.
- report the inconsistency between AS and the IP block in ROAs.

By playing this game and reducing the financial loss, the participating network operators and registries restrain themselves and operate compliantly in BGPcoin.

5.4. BGPcoin Infrastructure

RPKI requires CAs to sign Resource Certificates (RCs) and Route Origin Attestations (ROAs), the distributed repositories to store those certificates and attestations and the Rely Parties (RPs) to verify the ROAs. Since the permissionless blockchain has provided a complete platform to store/compute/trade and then enable a rapid deployment of any application, BGPcoin only relies on this platform as its infrastructure. Specifically, BGPcoin is supported by the EVM to compute and the Ethereum blockchain to store and the consensus protocol and the incentivized miners to verify and approve the transactions. Compared with the RPKI, BGPcoin need not build an exclusive infrastructure.

Blockchain vs. repository: In RPKI, every RIR maintains a repository to store the cryptographic RPKI objects for the resource under its governance. Since the RPKI repository is managed without supervision, a malicious authority is able to launch a mirror world attack [24] in which it presents one view of its RPKI repository to some RPs and a different view to others. In contrast, Ethereum blockchain provides an append-only and tamper-resistant ledger, which enables anyone to retrace the history of transactions, and so every transaction is non-reputable. In this way, BGPcoin maintains the global consistent view of information and eliminates the mirror world attack.

BGPcoin miner vs. relying party: In RPKI, Relying Parties (RPs) play the role of trust anchors with three functions for BGP border routers: (1) through the sync protocol [34], they collect the RPKI objects from the distributed RPKI repositories; (2) by using a validation tool, they verify cryptographically ROAs; (3) at the end, RPs push origin validation results to the BGP border routers for routing decisions. In this case, only if the RPs are honest and trustworthy, the validation results can be ensured to be reliable.

In BGPcoin, the smart contracts have defined the restriction logic to issue an ROA. Only if a transaction to issue the ROA meets the restriction, the ROA will be appended to the blockchain. Once an `ROA_add` transaction is submitted to issue an ROA, the miners compete to verify and confirm the transaction for the fee. Any cheating of one miner leads to its potential bookkeeping rights being lost. Without the RPs, BGPcoin allows a BGP border router to use the Ethereum client in its own AS to directly request ROAs from the blockchain.

6. BGPcoin Transaction vs. RC/ROA in RPKI

BGPcoin and RPKI both provide their trustworthy binding methods to authorize AS to originate the prefixes in BGP against prefix/subprefix hijacks. The binding methods are supported by their crucial data structures (RCs and ROAs in RPKI and transaction in BGPcoin, respectively). We compare the transaction in BGPcoin with the objects including RCs and ROAs in RPKI and observe that BGPcoin yields significant benefits on three characteristics, as shown in Table 2.

Table 2. BGPcoin transaction vs. RC/ROA.

Transaction in BGPcoin		**RC/ROA in RPKI**	
History-based trustworthiness	Sequential transparency	Certificate-based trustworthiness	Log-appending transparency
	Hash-chained integrity		Manifest-signed integrity
Transaction audit	Miner verification	Unbridled authority Misbehavior	Misconfiguration
	Immutable ledger		Stealthy deleting/overwriting
	Consent before revoke		Unilateral revoke/reclaim
Explicit resource ownership	Sole usufruct of IP prefix	Overly flexible resource attestation	Ambiguity double-cover IP prefix
	Reallocation after withdrawal		Targeted whacking

In RPKI, malicious CAs that pose great threats and misconfigurations [12] need attention as well. We give a case study to demonstrate how BGPcoin eliminates the risks that have been highlighted [16,24,32] in RPKI efficiently. We show in Figure 5 that RPKI suffers from unilateral resource revocation and inconsistency view attacks, manipulating to whack targeted ROAs, and misconfigured ROAs will cause all routes that covered it to become invalid. We set the $ROA3$ as the target ROA in Figure 5. In the framework of RPKI, every entity has its resource certificates, and every ROA is issued by an EE certificate (which is for an end entity and not allowed to issue its child certificate). The malicious authorities have the ability to do attacks as follows:

1. **Unilateral revoke:** $ISP2$can simply add the EE certificate in the $ROA3$ to its certificate revocation list, to make the target ROA $ROA3$ invalid.
2. **Stealthy deleting:** $APNIC$, as the RIR, can stealthily delete or corrupt $ROA3$ or the entity certificate for 218.241.108.0/22 in its repository and changes its manifest appropriately. In this case, RPs will accept and update the change, and $ROA3$ will be absent.
3. **Overwriting:** $ISP2$ overwrites the EE certificate for 218.241.108.0/22 with a different key, so $ROA3$ is no longer signed by the new key and becomes invalid.
4. **Targeted whacking:** A more complex targeted whacking can be launched by $CNNIC$, as a NIR, , which modifies the RC of $ISP2$and issues a new $ROA3$ that covers 218.241.96.0/20, but with a new ASN $AS3$. Then, the former $ROA3$ becomes invalid since no RC covers it, and $AS1$ is no longer the bound AS who legally originates 218.241.96.0/20. As collateral damage, $ROA1$ becomes invalid as well.

In contrast, the characteristics of history-based trustworthiness, autonomic transaction audit and explicit resource ownership enables BGPcoin to minimize the capability of authorities to eliminate the risk of misconfiguration and misbehaving above.

History-based trustworthiness: The transparency is a critical property to avoid the authority misbehaving. Supporting the transparency in RPKI requires an additional logging mechanism [24]. In contrast, the history-based trustworthiness of BGPcoin supported by the blockchain presents its inherent transparency provided by the type of sequence-appending hash-chained data structure.

Inherent transaction audit: The miner verification and the process logic of smart contracts of BGPcoin provide the inherent transaction audit and eradicates the violation of the contract like the authority misbehaviors and misconfigurations. The immutable ledger of BGPcoin blockchain

guarantees that once a resource transaction is successfully validated by miners, it is impossible to reverse or overwrite it, and the consent in BGPcoin precludes the unilateral revoke.

Explicit resource holdership/ownership: In BGPcoin, a transaction displays an operation of resource allocation or assignment with its inherent authorization. Apparently, it establishes the explicit resource ownership of the transaction receiver and eliminates the circular dependency [12] in allocating or assigning a resource.

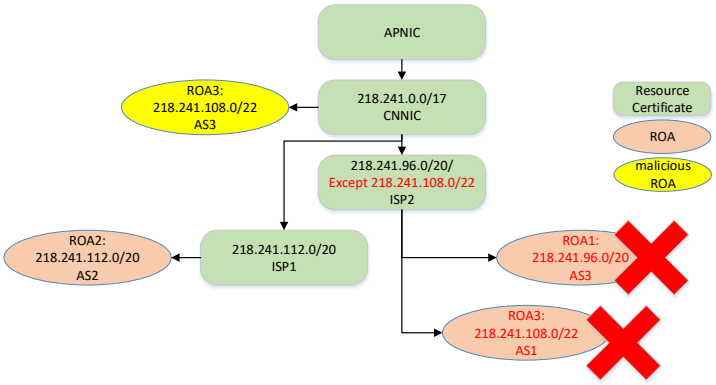


Figure 5. A case of RPKI authority misbehaving.

7. Implementation and Deployment

We have implemented a working prototype of BGPcoin and made it publicly available (https://github.com/Qianqianxing/BGPcoin-master). We implement the smart contracts in Solidity (http://solidity.readthedocs.io), a high-level Ethereum language that resembles JavaScript for writing smart contracts that are compiled to EVM code. We demonstrate the extensive experiment on a local simulated Ethereum network with a private blockchain and a development environment web (https://github.com/ethereum/web3.py) interacting with it. Moreover, we deploy our contracts on the official Ethereum Ropsten (https://ropsten.etherscan.io/) test network with a secure, reliable and scalable access to Ethereum, named infrura (https://infura.io/).

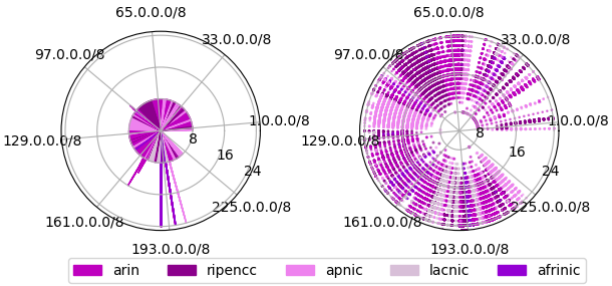
7.1. Overview

The prototype consists of a set of smart contracts that perform all typical operations for the Internet resource and a client written in Python to interact with the smart contracts deployed in the Ethereum blockchain.

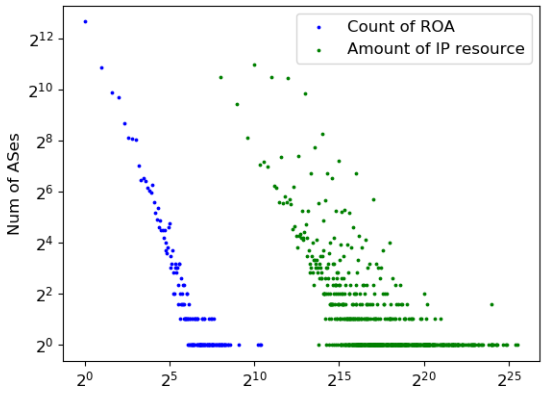
Methodology: We firstly reproduce the process of the Internet number allocating and assigning on our experiment BGPcoin blockchain as the setup phase. We collect the real resource delegations (IP addresses and ASNs) from IANA to RIRs, RIRs to NIRs, NIRs to ISPs. Before the deployment of BGPcoin on the blockchain, we generate an Ethereum address and the corresponding private key for every entity. Then, we map the resource delegations in the real world to three kinds of transactions, `IPregister`, `IPallocate` and `IPassign`, that the resource delegators send to the delegatees. We also collect the mappings from the IP prefix to ASN from the public available source and give a `addROA` demonstration of some mappings to evaluate the cost. After the setup phase, BGPcoin is able to serve as a resource query system.

Data: We collect the ASN and IP block registers and allocations from the public archives RIPE RIS [35] and the other four RIR datasets (ftp.arin.net/pub/stats/, ftp.afrinic.net/pub/stats, ftp.apnic.net/pub/stats/, ftp.lacnic.net/pub/stats). Since the RPKI currently only overs less than 10% mappings from AS to IP in its ROAs, we collect the compatible IP-ASN-mappings information from a service (http://www.team-cymru.com/IP-ASN-mapping.html) dedicated to mapping IP numbers to BGP prefixes and ASNs.

The distribution of the registered, allocated and assigned IP numbers deployed in BGPcoin is shown in Figure 6a. The left pie chart shows the IP prefixes' distribution where most of the IP blocks that the RIRs register are of the prefix length of eight, except the IP numbers near 193.0.0.0, where the registered IP block has the prefix length of 24. The right pie chart shows the allocated or assigned IP prefixes' distribution in five RIRs, where we observe that a few IP numbers have not been allocated or assigned. The IP numbers held by ASes and the number of ROAs bound with ASes are shown in Figure 6b. Most of the ASes (>2^{12}) only hold one ROA, and there exists some AS that hold more than 2^{10} ROAs and more than 2^{25} IP numbers covered in its ROAs.



(a) Registered and Allocated IP Distribution



(b) IP Resource and ROAs for ASes

Figure 6. Resource distribution in BGPcoin.

7.2. Experiments on a Simulated Network

To demonstrate the scalability and performance of our design, we first construct a simulated Ethereum network locally, which is much like the real Ethereum environment initialized with the default configuration except for that the mining difficulty is set lower than the real Ethereum environment for processing the experiment quickly. This allows us to focus on the performance of the search part on smart contract, irrespective of the time-consuming mining process and complex network circumstances (e.g., broadcast latency, transaction mining delay) in Ethereum. We set up BGPcoin by importing the IP register and allocation information to the blockchain.

Results: To illustrate the impact of the mining process on the efficiency, we record the block number of each transaction generated in our setup phase and the corresponding gas usage, as shown in Figure 7. Since we experiment with the setup phase in parts at several different times, we only show one part of our blockchain data as a sample. The average block time for mining is 25 s, and the number of blocks containing transactions is more than 8000, resulting in nearly six days to complete the entire

setup phase. This again explains why the time cost of setup is dominated by the smart contract, instead of the data owner, like in existing centralized search schemes.

We are faced with a difficult problem to accelerate further the processing of our setup phase. The resource transaction needs to be reproduced in chronological order. For example, an IP prefix 45.0.0.0/8 has to be registered by IANA and allocated to ARIN firstly, and then, ARIN delegates 45.5.212.0/22 to Brazil. Therefore, these transactions with the inter-dependence and a strict chronological order lead to the less concurrency of the setup phase of BGPcoin. In our analysis, this is the other reason why the reproducing took such a long time. However, if we sent one transaction after the time the previous transaction had been confirmed, it would contain only one transaction in a block since the confirm unit on the blockchain is one block. Therefore, we still try to send 10–30 transactions at one time to ensure the current block to assemble as much transactions within the block limit. The transactions that have not been recorded in the current block wait to be confirmed in the next block. Moreover, we periodically collect the failed transactions that may have inconsistency and conflict with the current state of the traded resource and resend them later. We notice the current gas limit of the real Ethereum has become 7,992,222 gases, which increases 70% the gas limit at the time when we performed the BGPcoin experiment. Every block has a larger capacity to contain more transactions if we reperform the experiment now.

Furthermore, we demonstrate the performance of executing the two most expensive operations, `IP register` and `IP allocate` in Figure 8. We experiment with the cost performance on more than 70,000 transactions reporting the real-world allocations and assignments of IP numbers. From the top row of the subfigures, we can tell that the allocation of IP numbers from an aggregated IP block costs far more than the cost of registering an IP block for an entity. We observe from the middle row figures that the highest cost of an `IP allocate` operation is linear with the prefix length of the operated IP block, which just conforms with the principal illustrated in Figure 3. Obviously, we can infer that the highest cost of allocating an IP prefix is from the biggest IP block that covers it (i.e., with the shortest prefix length of eight).

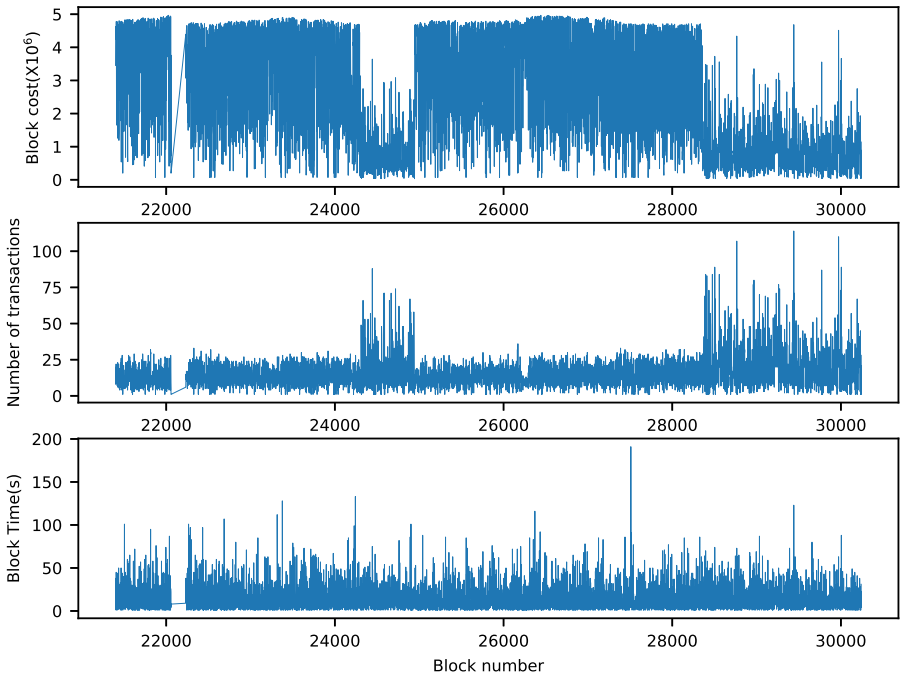
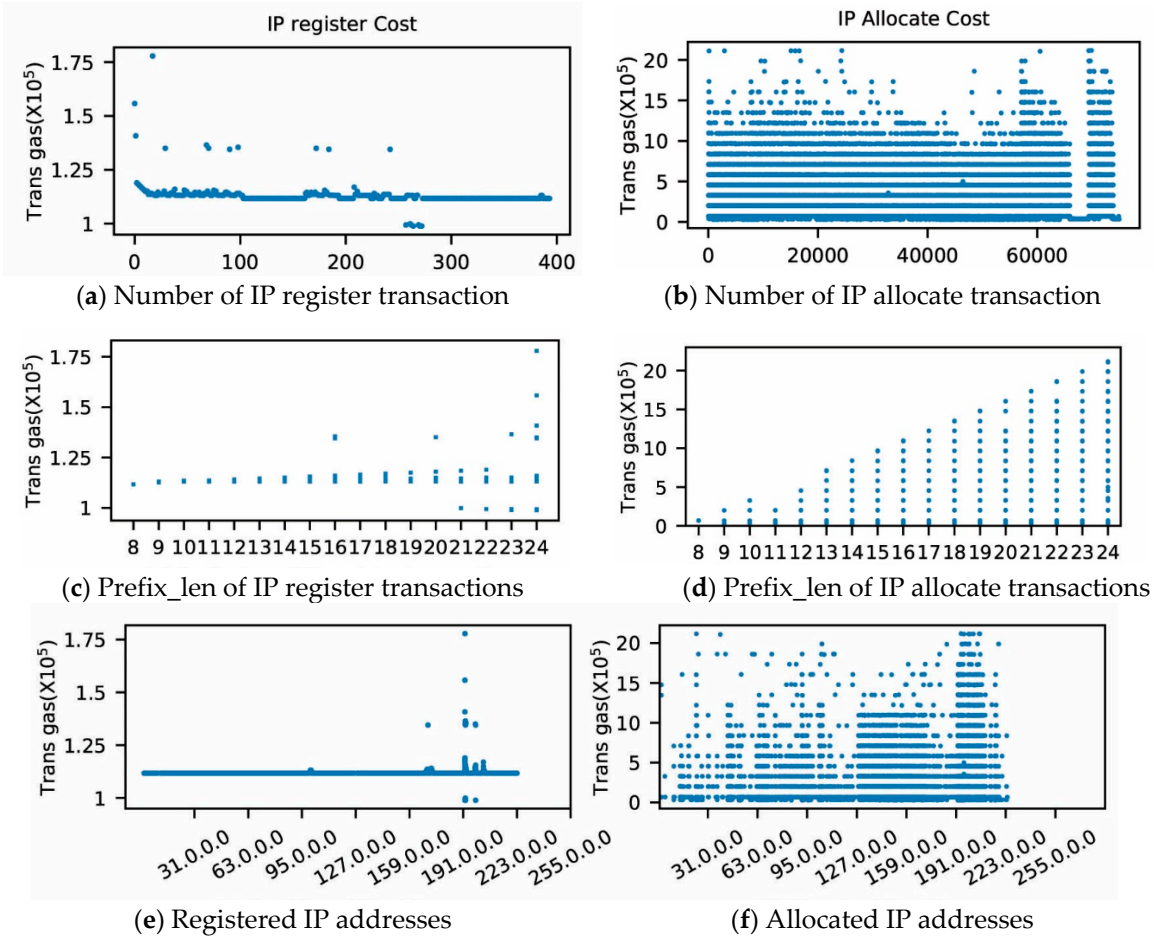


Figure 7. Block gas cost and number of transactions in each block.



(**a**) Number of IP register transaction (**b**) Number of IP allocate transaction

(**c**) Prefix_len of IP register transactions (**d**) Prefix_len of IP allocate transactions

(**e**) Registered IP addresses (**f**) Allocated IP addresses

Figure 8. Performance of two expensive operations: IP register and IP allocate.

7.3. Experiments on an Official Test Network

To show the practicability of our scheme, we deploy the official Ethereum test network Ropsten that mimics the real production network. Due to the limited balance, we only conduct experiments on the smallest database. Our contract addresses of `BGPcoin_base`, `BGPcoin_client` and `BGPcoin_checker` in Ropsten are:

- 0xb23e182afb61096e51d9c5d22dcb9966c745b0c9.
- 0xa9fcbaeb1a113e7b894eb57df8b25a674d635aad.
- 0xf5b285e3c7aea8f67c100b8ac036fda022dd52a5.

Results: We estimated the cost of creating the BGPcoin contract and each type of BGPcoin transaction, including the approximate computational steps in Ethereum's gas and the approximate costs in U.S. dollars in Table 3. Note that in September 2018, 1 ether = $289.15 and 1 gas $= 1.8 \times 10^{-8}$ ether (https://ethstats.net/, https://coinmarketcap.com/), and compared with the paid service provided by the current Internet resource management, the cost of BGPcoin is relatively low. Moreover, the participating organization motivates the mining and validation of transactions by increasing the fees on demand.

Table 3. Cost of BGPcoin trading operations in the Ropsten network.

Operation	Gas	USD	Operation	Gas	USD
IP register	155,448	0.449	IP revoke	72,960	0.211
IP allocate	188,113	0.544	ASN register	4411	0.123
IP assign	183,246	0.530	ASN allocate	68,876	0.199
IP update	69,101	0.200	ASN update	27,691	0.080
BGPcoin Contract Creation				3,985,649	11.524

Scalability analysis: According to the seven-day BGP profile from 28 August 2018 00:00–3 September 2018 23:59 [36], although the recent peak prefix update rate per second is 1818 at 19:36:30 Wednesday, 29 August 2018, we observe that the average prefix updates per second is only 5.60. Considering Ethereum has 7–15 transactions per second (https://en.wikipedia.org/wiki/Ethereum) and, moreover, advanced consensuses (https://github.com/ethereum/EIPs/issues/225) are in progress to promote Ethereum's throughputs to thousands of transactions per second, it is viable to have BGPcoin have 5 tran/s of throughput for BGP advertisement on average. We believe BGPcoin poses a feasible and credible BGP security solution, and an extensive experiment for the global deployment of BGPcoin has been conducted to further measure the practical scalability.

8. Related Works

Namecoin [37] was the first system to build a decentralized naming system, i.e., an alternate DNS-like system that replaces DNS root servers with a blockchain for mapping domain names to DNS records. As proposed as a decentralized PKI service on top of Namecoin, Blackstack [38] facilitates Internet-scale decentralized naming systems. Since an asset in them is indivisible to a set of small assets, but the IP address asset needs to be aggregated or divided, neither of them is able to handle the flexible mapping of the IP address.

Internet blockchain [27] firstly introduced the blockchain to the trustworthy management for the Internet control plane. Xing [39] and Paillisse [40] took a similar way to build secure IP prefix allocation and delegation. However, there is still a lack of a complete BGP security framework solution with the blockchain including the route origin advertisement and authentication around the blockchain.

9. Conclusions

In this work, we introduced a novel Internet resource management and route origin advertisement/authentication solution for BGP security. We designed the BGPcoin system consisting of a smart contract-based resource assignment attestation and a blockchain-based dependable repository infrastructure. We demonstrated through an extensive analysis that the deployment incentives and increased security are technically and economically viable.

Future works: We are working on a systematic implementation of BGPcoin in a practical scenario with the Quagga Secure Routing Extension [41] Routers and some critical performance measurements including the real-time update speed from the BGPcoin blockchain to BGP border routers, as well as exploring how securely a partial and incremental deployment of BGPcoin protects the Internet BGP system.

Author Contributions: Q.X. contributed to the design of the ideas, the analysis of the results and the writing of the paper. B.W. and X.W. proofread the paper.

Funding: This research is supported in part by the project of the National Key Research and Development Program of China (2017YFB0802301).

Conflicts of Interest: The authors declare that there is no conflict of interests regarding the publication of this manuscript.

Abbreviations

The following abbreviations are used in this manuscript:

BGP	Border Gateway Protocol
IP	Internet Protocol
MITM	Man-In-The-Middle
AS	Autonomous System
ASN	Autonomous System Number
MOAS	Multiple Origin Autonomous System
PKI	Public Key Infrastructure
RPKI	Resource Public Key Infrastructure
RC	Resource Certificate
EE	End Entity
ROA	Route Origin Attestation
ISP	Internet Service Provider
INR	Internet Number Resource
IANA	Internet Assigned Numbers Authority
RIR	Regional Internet Registry
NIR	National Internet Registry
LIR	Local Internet Registry
CA	Certification Authority
CRL	Certificates Revocation List
RP	Relying Party
APNIC	Asia-Pacific Network Information Center
CNNIC	China Internet Network Information Center

Appendix A

Appendix A.1. Data Structure in the Smart Contracts of BGPcoin

The smart contract defines all operations that the participating entities can perform on the assets, as well as the precondition and the result of those operations. An entity refers to any participant in the system, i.e., IANA, xIPs and ISPs. BGPcoin has two types of assets: *Internet Address resource* and *AS numbers resource*. Every asset has its fields to represent its assignment and authorized status, shown in Data Structures of BGPcoin Assets, which as the origin attestation supports IP announcement validation in BGP updating.

```
struct IPBData{
 unit32 IPIndex;
 unit8  prefixlen;
 State  state;
 address RIR;
 address NIR;
 address owner;
 address leasee;
 bool[3] contents
}
struct ASData{
 unit ASIndex;
 address  RIR;
 address owner;
 unit  stime;
 unit  validperiod;
 unit24[] adjASN;
}
struct IPBFlag{ unit8 prefixlenlimit; bool deleted; }
struct ASNFlag{ uin248 ASN; bool deleted; }
struct IPmap{
 mapping (unit32 => IPBData) IPB;
 mapping (unit32 => IPBFlag) keys;
 unit size;
}
```

```
struct ASmap{
 mapping (unit24 => IPBData) ASData;
 ASNFlag[] keys;
 unit size;
}
struct ROAData{
 unit24 IPS;
 unit8  prefixlen;
 unit  stime;
 unit  validperiod;
}
 mapping (unit24 => ROAData[]) public ROA;
 IPmap public BGPcoin_IPchain;

 ASmap public BGPcoin_ASchain;
```

References

1. Cowie, J. The New Threat: Targeted Internet Traffic Misdirection. Blog. 2013. Available online: https://dyn.com/blog/mitm-internet-hijacking/ (accessed on 15 August 2018).
2. Blogs, D.G. Pakistan Hijacks Youtube Blog. 2008. Available online: https://dyn.com/blog/pakistan-hijacks-youtube-1/ (accessed on 15 August 2018).
3. Toonk, A. Hijack Event Today By Indosat Blog. 2014. Available online: https://bgpmon.net/hijack-by-as4761-indosat-a-quick-report/ (accessed on 15 August 2018).
4. Schlamp, J.; Carle, G.; Biersack, E.W. A forensic case study on as hijacking: The attacker's perspective. *ACM SIGCOMM Comput. Commun. Rev.* **2013**, *43*, 5–12. [CrossRef]
5. Vervier, P.; Thonnard, O.; Dacier, M. *Mind Your Blocks: On the Stealthiness Of Malicious BGP Hijacks*; NDSS: San Diego, CA, USA, 2015. Available online: https://pdfs.semanticscholar.org/496e/09594a920a0027fd6a65481c091d61692304.pdf (accessed on 15 August 2018).
6. Sermpezis, P.; Kotronis, V.; Gigis, P.; Dimitropoulos, X.; Cicalese, D.; King, A.; Dainotti, A. ARTEMIS: Neutralizing BGP hijacking within a minute. *arXiv* **2018**, arXiv:1801.01085.
7. Andersen, D.G.; Balakrishnan, H.; Feamster, N.; Koponen, T.; Moon, D.; Shenker, S. Accountable internet protocol (AIP). *ACM SIGCOMM Comput. Commun. Rev.* **2008**, *38*, 339–350. [CrossRef]
8. Mirkovic, J.; Reiher P. Building Accountability into the Future Internet. In Proceedings of the 2008 4th Workshop on Secure Network Protocols, Orlando, FL, USA, 19 October 2008; pp. 45–51.
9. Li, A.; Liu, X.; Yang, X. Bootstrapping Accountability in the Internet We Have. In Proceedings of the NSDI '11: 8th USENIX Symposium on Networked Systems Design and Implementation, Boston, MA, USA, 30 March–1 April 2011; pp. 155–168.
10. Lee, T.; Szalachowski, P.; Barrera, D.; Perrig, A.; Lee, H.; Watrin, D. Bootstrapping real-world deployment of future internet architectures. *arXiv* **2015**, arXiv:1508.02240.
11. Goldberg, S. Why is it taking so long to secure internet routing? *Commun. ACM* **2014**, *57*, 56–63. [CrossRef]
12. Gilad, Y.; Cohen, A.; Herzberg, A.; Schapira, M.; Shulman, H. *Are We There Yet? On RPKI's Deployment and Security*; NDSS: San Diego, CA, USA, 2017; ISBN 1-891562-46-0.
13. Al-Musawi, B.; Branch, P.; Armitage, G. BGP anomaly detection techniques: A Survey. *Commun. Surv. Tutor.* **2017**, *19*, 377–396. [CrossRef]
14. Goldberg, S.; Schapira, M.; Hummon, P.; Rexford, J. How secure are secure interdomain routing protocols. *ACM SIGCOMM Comput. Commun. Rev.* **2011**, *41*, 87–98.
15. Cooper, D.; Heilman, E.; Brogle, K.; Reyzin, L.; Goldberg, S. On the Risk of Misbehaving RPKI Authorities. In Proceedings of the Twelfth ACM Workshop on Hot Topics in Networks, College Park, MD, USA, 21–22 November 2013; p. 16.
16. Brogle, K.; Cooper, D.; Goldberg, S.; Reyzin, L. *Impacting IP Prefix Reachability via RPKI Manipulations*; Technical Report BUCS-TR-2013-001; Computer Science Department, Boston University: Boston, MA, USA, 2013. Available online: http://hdl.handle.net/2144/11410 (accessed on 15 August 2018).
17. Siddiqui, M.S.; Montero, D.; Serral-Gracià, R.; Masip-Bruin, X.; Yannuzzi, M. A survey on the recent efforts of the Internet Standardization Body for securing inter-domain routing. *Comput. Netw.* **2015**, *80*, 1–26. [CrossRef]

18. Green, T.; Lambert, A.; Pelsser, C.; Rossi, D. Leveraging Inter-domain Stability for BGP Dynamics Analysis. In Proceedings of the International Conference on Passive and Active Network Measurement, Berlin, Germany, 26–27 March 2018; pp. 203–215.
19. Mitseva, A.; Panchenko, A.; Engel, T. The State of Affairs in BGP Security: A Survey of Attacks and Defenses. *Comput. Commun.* **2018**, *144*, 45–56. [CrossRef]
20. Boldyreva, A.; Lychev, R. Provable security of S-BGP and other path vector protocols: Model, analysis and extensions. In Proceedings of the 2012 ACM conference on Computer and communications security, Raleigh, NC, USA, 16–18 October 2012; pp. 541–552.
21. Hollick, M.; Nita-Rotaru, C.; Papadimitratos, P.; Perrig, A.; Schmid, S. Toward a taxonomy and attacker model for secure routing protocols. *ACM SIGCOMM Comput. Commun. Rev.* **2017**, *47*, 43–48. [CrossRef]
22. Wählisch, M.; Maennel, O.; Schmidt, T.C. Towards detecting BGP route hijacking using the RPKI. In Proceedings of the ACM SIGCOMM 2012 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication, Helsinki, Finland, 13–17 August 2012; pp. 103–104.
23. Gersch, J.; Massey, D. Rover: Route origin verification using DNS. In Proceedings of the 2013 22nd International Conference on Computer Communication and Networks (ICCCN), Nassau, Bahamas, 30 July–2 August 2013; pp. 1–9.
24. Heilman, E.; Cooper, D.; Reyzin, L.; Goldberg, S. From the Consent of the Routed: Improving the Transparency of the RPKI. *ACM SIGCOMM Comput. Commun. Rev.* **2014**, *44*, 51–62. [CrossRef]
25. Kuerbis, B.; Mueller, M. Internet routing registries, data governance, and security. *J. Cyber Policy* **2017**, *2*, 64–81. [CrossRef]
26. Bonneau, J.; Miller, A.; Clark, J.; Narayanan, A.; Kroll, J.A.; Felten, E.W. Sok: Research Perspectives and Challenges for Bitcoin and Cryptocurrencies. In Proceedings of the 2015 IEEE Symposium on Security and Privacy, San Jose, CA, USA, 17–21 May 2015; pp. 104–121.
27. Hari, A.; Lakshman, T.V. The Internet Blockchain: A Distributed, Tamper-Resistant Transaction Framework for the Internet. In Proceedings of the 15th ACM Workshop on Hot Topics in Networks, Atlanta, GA, USA, 9–10 November 2016; pp. 204–210.
28. Wiki, E. Ethereum Development Tutorial. 2018. Available online: https://github.com/ethereum/wiki/wiki/Ethereum-Development-Tutorial (accessed on 15 August 2018).
29. Ethereum, Proof of stake FAQs. 2018. Available online: https://github.com/ethereum/wiki/wiki/Proof-of-Stake-FAQs (accessed on 15 August 2018).
30. Rooney, T. *IP Address Management Principles and Practice;* A John Wiley & Sons, INC: Hoboken, NJ, USA, 2011.
31. Kronovet, D. A Next-Generation Smart Contract and Decentralized Application Platform. 2017. Available online: https://github.com/ethereum/wiki/wiki/White-Paper (accessed on 15 August 2018).
32. Gilad, Y.; Sagga, O.; Goldberg, S. Maxlength Considered Harmful to the RPKI. In Proceedings of the 13th International Conference on emerging Networking EXperiments and Technologies, Incheon, Korea, 12–15 December 2017; pp. 101–107.
33. Matsumoto, S.; Reischuk, R.M. IKP: Turning a PKI Around with Decentralized Automated Incentives. In Proceedings of the 2017 IEEE Symposium on Security and Privacy (SP), San Jose, CA, USA, 22–26 May 2017; pp. 410–426.
34. Weiler, S.; Ward, D.; Housley, R. The rsync URI Scheme. Available online: http://www.hjp.at/doc/rfc/rfc5781.html (accessed on 15 August 2018).
35. RIPE NCC Routing Information Service (RIS). 2018. Available online: https://www.ripe.net/analyse/internet-measurements/routing-information-service-ris (accessed on 15 August 2018).
36. Huston, G. The BGP Instability Report. 2018. Available online: http://bgpupdates.potaroo.net/instability/bgpupd.html (accessed on 15 August 2018).
37. Kalodner, H.A.; Carlsten, M.; Ellenbogen, P.; Bonneau, J.; Narayanan, A. An Empirical Study of Namecoin and Lessons for Decentralized Namespace Design. Available online: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.698.4605&rep=rep1&type=pdf (accessed on 15 August 2018).
38. Ali, M.; Nelson, J.C.; Shea, R.; Freedman, M.J. Blockstack: A Global Naming and Storage System Secured by Blockchains. In Proceedings of the USENIX Annual Technical Conference, Denver, CO, USA, 22–24 June 2016; pp. 181–194.

39. Xing, Q.; Wang, B.; Wang, X. Poster: Bgpcoin: A Trustworthy Blockchain-Based Resource Management Solution for BGP Security. In Proceedings of the ACM Sigsac Conference on Computer and Communications Security, Dallas, TX, USA, 30 October–3 November 2017; pp. 2591–2593.
40. Paillisse, J.; Ferriol, M.; Garcia, E.; Latif, H.; Piris, C.; Lopez, A.; Kuerbis, B.; Rodrigueznatal, A.; Ermagan, V.; Maino, F.; et al. IPchain: Securing IP Prefix Allocation and Delegation with Blockchain. *arXiv* **2018**, arXiv:1805.04439.
41. Borchert, O. Kyehwanl BGP Secure Routing Extension (BGP-SRx) Prototype. Available online: https://www.nist.gov/services-resources/software/bgp-secure-routing-extension-bgp-srx-prototype (accessed on 15 August 2018).