sidrops
Internet-Draft
Intended status: Informational
Expires: 29 August 2025

D. Li
Tsinghua University
L. Chen
Y. Su
Zhongguancun Laboratory
Y. Fu
China Unicom

25 February 2025

RPKI Repository Problem Statement and Analysis
draft-li-sidrops-rpki-repository-problem-statement-01

Abstract

With the widespread deployment of Route Origin Authorization (ROA) and Route Origin Validation (ROV), Resource Public Key Infrastructure (RPKI) is vital for securing inter-domain routing. RPKI uses cryptographic certificates to verify the authenticity and authorization of IP address and AS number allocations and the certificates are stored in the RPKI Repository. This document conducts the data-driven analysis of the RPKI Repository, including a survey of worldwide AS administrators and a measurement and analysis of the existing RPKI Repository. This document finds that the current RPKI Repository architecture is sensitive to failures and lacks of scalability. An attack or downtime of any repository Publication Point (PP) will prevent RPs from obtaining complete RPKI object views. Furthermore, since the current RPKI Repository is not tamper-resistant, RPKI authorities can easily manipulate RPKI objects without consent from subordinate INR holders. This document also defines the key requirements for a reliable, scalable, and secure RPKI Repository.

This Internet-Draft will expire on 29 August 2025.

Table of Contents

1. Introduction

To establish a trustworthy mapping between AS numbers and IP prefixes, RPKI arranges the Certificate Authorities (CAs) in a hierarchy that mirrors IP address allocation, and five RIRs are trust anchors. Each CA in RPKI holds a Resource Certificate (RC) issued by its parent authority, which attests to a binding of the entity's public key to a set of allocated Internet Number Resources (INRs, such as IP address blocks and AS numbers). CAs can issue subordinate RCs to reallocate their resources or issue ROAs (leaf nodes) to authorize ASes to originate specific IP prefixes.

As shown in Figure 1, each CA in RPKI will upload the objects it signs, such as manifests, CRLs, RCs, and ROAs, to the repository Publication Point (PP) it specifies. These PPs naturally form a tree structure that mirrors the RPKI certificate tree and collectively form the global RPKI Repository. Relying Parties (RPs), as entities that help ASes request RPKI data, periodically traverse RCs from five root RCs (held by RIRs) and access the PPs specified in their Subject Information Access (SIA) fields to fetch all RPKI objects, and then validate them. Finally, the ASes served by the RPs can use the verified ROAs to guide routers to perform ROV.

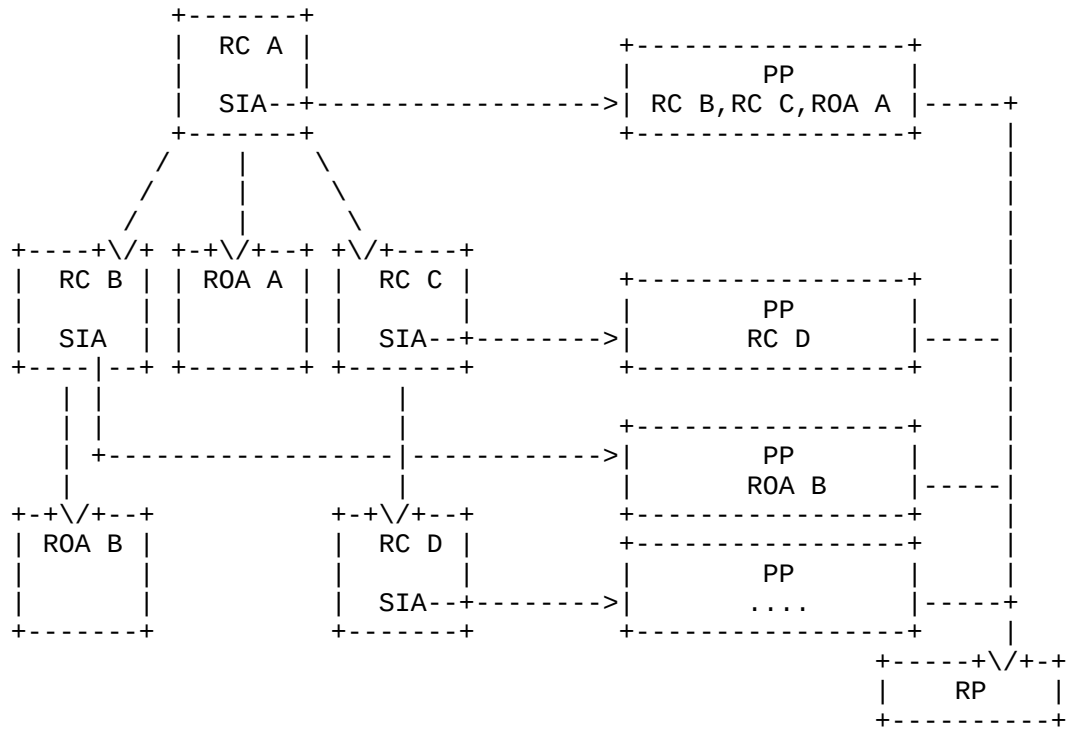

Figure 1: The RPKI certificate tree and RPKI Repository structure.

With the increasing deployment of RPKI, in June 2024, ROAs cover 42.40% of active IPv4 addresses and 57.97% of active IPv6 addresses. The number of independent repository instances has grown to 63. This document provides measurements and analysis of the reliability, scalability, and security of the RPKI Repository architecture. The measurement includes a survey with the AS administrators of 2,500 randomly selected ROA-deployed ASes and a measurement of the RPKI Repository deployment status. The survey mainly focuses on the future consideration of adopting delegated RPKI and considerations regarding malicious behavior from RPKI authorities. We have observed that the current RPKI Repository lacks the ability to provide reliable services to RPs. RPKI allows certificates issued by a CA to be stored only at the PP operated by that CA, which means that any failure of a PP can hinder RPs from obtaining complete RPKI data. Then, as delegated RPKI becomes more popular, an increasing number of CAs choose to maintain their own repository instances. However, some of these repositories lack robust, secure, and reliable infrastructure. Furthermore, the proliferation of repositories will affect the scalability of RPKI, as RPs need to traverse all PPs to refresh their local caches. Additionally, with the rising deployment rates of ROA and ROV, RPKI plays an increasingly important role in the inter-domain routing. Consequently, AS administrators are more concerned about RPKI security, especially the potential malicious behavior of RPKI authorities.

1.1. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

1.2. Terminology

It is assumed that the reader is familiar with the terms and concepts described in "A Profile for Resource Certificate Repository Structure" [RFC6481], "The RPKI Repository Delta Protocol (RRDP)" [RFC8182], and "An Infrastructure to Support Secure Internet Routing" [RFC6480].

2. Reliability Analysis

Although the current RPKI Repository is globally distributed, each CA only stores the RPKI objects it issues in the unique PP it runs. If any PP fails (malfunctions or is attacked), RPs will not be able to fetch the RPKI objects issued by its CA, and the integrity of the global RPKI object view cannot be guaranteed.

As of August, 2024, there are 42,950 RCs and 303,918 ROAs in RPKI Repository. For each RC, we extracts the SIA field that records the URI of its HTTPS-based RRDP file (Since all PPs now support RRDP and serve RPs through RRDP files, the analysis of the RPKI Repository focuses primarily on RRDP Repository). The result shows that there are currently 63 independent repository instances (SIA fields of RCs held by the same entity may share the same PP, therefore, the number of repositories is much lower than that of RCs).

Next, the measurement focuses on the ASes where the repositories are located, their IP addresses, and whether they utilize CDNs. We uses 2,000 globally distributed DNS resolvers to resolve repositories' domain names, with the aim of finding the CNAME and all IP address records for each repository. Then, we use the IP-to-AS mapping information maintained by RIPE NCC [routing-history] to obtain the AS where each PP is located.

Typically, CDN service providers will display their information in the domain name and CNAME record, such as including "cdn.cloudflare.net" or specify the CDN in the X-Via-CDN field, cache status field, or Server field in the HTTPS headers. In addition, if CDN acceleration is used, the latency of requesting HTTPS services from different geographic locations around the world will be relatively low. Therefore, it can determine whether the PP services are hosted on CDNs from the following aspects: (1) Whether the domain name and CNAME record contain information about CDN service providers; (2) The number of IP addresses returned by DNS resolvers and the geographical distribution of these IP addresses; (3) Whether the HTTPS request headers contain information about mainstream CDN providers; (4) The latency of accessing RRDP files from different geographic locations around the world.

Measurement results show that although RRDP enables the use of Content Distribution Networks (CDNs) infrastructure for resilient service, only 9 out of 63 repositories are hosted in CDNs, with 8 being hosted on Cloudflare AS13335 and 1 on Amazon AS16509. It also shows that out of 63 repositories, 60 of them are hosted in a single AS. It means that the accessibility of these repositories is highly dependent on the reachability of a single AS. Worse still, among the repositories whose corresponding ASes have deployed ROAs, there are

14 of them carry the ROA of the ASes in which they are located. It means that once the repository goes down and the RPs cannot fetch its ROAs, the route of the AS that the repository locates in may be downgraded by ROV adopters. Then, even if the repository is restored, those ROV adopters still cannot access the repository, in which case the access of the repository depends on the reachability of the AS it locates in, while the reachability of the AS also depends on the access of the repository.

Real-world incidents of repository breakdowns do occur at times. On 6 April 2020, the repository maintained by RIPE NCC suffered a sudden increase in connection to service [RIPE-downtime], resulting in it appearing as down to many RPs for 7 hours (RIPE's services are already hosted on CDNs). On 15 May 2020, the Japan operated repository was out of service for 10 hours due to hardware failure [service-outage], and between 26 Jan 2022 and 2 Feb 2022, due to full disk space, all ROAs in its repository again became invalid [disk-outage]. During 10 July 2024 to 12 July 2024, we also found that RPKI data synchronized through Routinator [routinator] once again had missing parts from RIPE NCC.

3. Scalability Analysis

In current RPKI infrastructure, refreshing the local cache of each RP involves traversing all repositories to fetch the updated RPKI objects. However, RPKI Repository has grown from 5 repositories run by five RIRs to more than 60 repositories and is expected to increase dramatically with the further deployment of ROA.

On the one hand, with a deeper understanding of RPKI, ROA deployers prefer to adopt delegated RPKI to flexibly control their RPKI objects. In the survey, 45.3% of the AS administrators using hosted RPKI say they have plans to run their own repositories for flexible certificate signing and control. The result shows that delegated RPKI will emerge as a trend, and the number of independent repositories will inevitably increase. The work [beyond-limits] predicts that when ROA is fully deployed, the number of repositories will reach 10K and there will be 140K active RPs, the RPKI snapshot download would easily exceed 1 hour (on the premise that each repository is well accessible). Since RPs are required to check the updates of all repositories when refreshing their local caches, the increasing number of repositories will result in higher refresh costs. It may prevent RPs from obtaining routing information in a timely manner, and also prohibits the release of use cases such as temporary ROAs to enable ISPs to perform tactical traffic engineering to ease congestion.

On the other hand, RPKI Repository is designed without a strict admission mechanism, allowing entities to apply for RC from RIRs or other RPKI CA entities, register as delegated CAs, and then operate repositories with a small fee and identity verification. Therefore, some repositories for unwanted purposes (measurement, attack experiments, etc.) are gradually emerging. Furthermore, the low barriers to running repositories and joining RPKI Repository will introduce unforeseen risks to RPs. For example, a malicious CA can create a large number of descendant RCs and operate numerous repositories to make RPs endlessly retrieve repositories, thus exhausting and paralyzing RPs. Although the most popular RP software, Routinator [routinator], has set the default value for the maximum searchable RPKI-tree depth to 32 in its latest version 0.14.1, and the RPKI-client has set the default value to 12, neither has provided a value for RPKI-tree width (as it is difficult to set a reasonable threshold). Similarly, it is also challenging for RP softwares to clearly limit the maximum number of accessible repositories. In addition, a series of recent academic works have emerged that exploit repository vulnerabilities to cause repository downtime, thereby preventing it from providing normal services to RPs [behind-the-scenes]. There are also works [beyond-limits][stalloris] that manipulate malicious repositories to attack RPs, thus obstructing the synchronization of RPKI data.

In addition, with the increasing rate of ROV deployment, more and more RPs will connect to RPKI Repository. It will undoubtedly increase the burden on the repositories. In summary, as RPKI deployment progresses further, RPs will need to access more repositories, and repositories will need to serve more RPs. This increase in the number of bidirectional connections will threaten the scalability of the RPKI system.

4. Security Analysis

In current RPKI infrastructure, RPKI authority (CA and its corresponding repository manager) uploads the RCs and ROAs it signs for subordinate INR holders to the repository it specified. It means that RPKI authorities control the signing and management of RPKI objects in their repositories and have significant unilateral power. The disproportionate division of power between RPKI authorities and INR holders will poses three issues:

(1) Since the current RPKI Repository is not tamper resistant, authorities can unilaterally undermine any RPKI objects without consent from subordinate INR holders. Malicious or breached authorities can perform arbitrary operation on INR holders' certificates, such as deletion, corruption, modification, or revocation (see [RFC8211] for more details). These operations often

involve diminishing the set of INRs associated with the affected INR holders. A malicious RPKI authority can also compromise RPs, for example, showing incomplete or inaccurate RPKI object views to RPs.

(2) INR holders and RPs can only rely on RPKI authorities without the ability to verify that their security requirements are being met. Specifically, INR holders cannot know if their RPKI objects are securely stored in the authorities' repositories and can be publicly seen and fetched by all RPs; RPs cannot verify the integrity and accuracy of the RPKI objects they synchronize from RPKI Repository.

(3) The history of RPKI objects is not easily auditable. A complete longitudinal view of RPKI objects is necessary to defend against malicious manipulation of the RPKI Repository and to resolve conflicts between authorities and INR holders. Although RPs can periodically fetch RPKI objects to keep track of their historical versions, it is costly, without any guarantee of completeness. Since RPKI lacks a trustworthy historical RPKI object record, it is also challenging, if not impossible, to hold authorities accountable even if attacks are detected after the fact.

In the survey, 44.1% of the AS administrators explicitly expressed concerns about malicious RPKI authorities (11.5% of them chose "not sure"). Two administrators provided additional feedback, indicating that they consider the threat from the RPKI authorities to be the most serious problem. One of them said they had lost all their ROAs due to administrative/human reasons. It can be seen that threats from RPKI authorities are also widely recognized in the industry.

5. Summary: Problems of the current RPKI Repository

Based on the measurement and analysis described in the previous section, this section summarizes the main problems of the current RPKI Repository and the key reason behind these issues: the binding of the repository PP with the CA.

5.1. P1: Every RPKI object is a singleton in RPKI Repository.

Although the current RPKI Repository is globally distributed, RPKI does not provide distributed storage for RPKI objects. The binding of PP and CA causes each RPKI objects to be stored only in the PP run by the CA that issued the object, instead of being stored in multiple repository nodes. Therefore, the current RPKI Repository cannot guarantee that RP can obtain complete RPKI data when some repository nodes fail. Even worse, the singleton nature of RPKI objects also introduces unwanted interdependence between the accessibility of a CA' s PP and the reachability of the AS that the PP locates. Since RPKI ensures the routing security and reachability of ASes, it is

fair to expect PPs to remain accessible during any incident when corresponding ASes become unreachable.

5.2. P2: RPKI Repository is costly in RP refreshing.

Refreshing the local cache of each RP involves traversing all repositories to fetch the updated RPKI objects. However, the binding of the repository PP to the CA causes the number of repository instances to increase dramatically with the further deployment of ROA (as the number of CAs that hold RCs increase). Furthermore, as AS administrators gain a deeper understanding of RPKI technology, delegated RPKI has emerged as a trend, CAs are more willing to maintain the repository PP themselves to achieve more flexible certificate signing and management. The growth in the number of repository instances increases the cost of RP refreshes, the growth in the number of RPs brings burdens to repositories, and both will threaten the scalability of RPKI.

5.3. P3: Unilateral reliance on authority.

The current RPKI authority (CA) not only operates a certificate engine but also operates a publication repository to store all RPKI objects it signs. It means that although the RPKI authority issues certificates to subordinate INR holders, the management of these certificates (especially their storage and distribution) is entirely controlled by the RPKI authority. The current RPKI Repository architecture exacerbates the power imbalance between the RPKI CA and subordinate INR holders and may also lead to RPs being deceived by receiving inaccurate or incomplete RPKI data from malicious RPKI authorities. However, INR holders and RPs have no choice but to rely unilaterally on the RPKI authority. This unilateral reliance prevents RPKI from proactively defending against the threats from malicious RPKI authorities. Moreover, it is also challenging to provide a complete and trustworthy longitudinal view of RPKI objects for post-incident audits, as the RPKI data is entirely controlled by the CA that signed it.

6. New Requirements for RPKI Repository

The following requirements are identified for a reliable, scalable and secure RPKI Repository architecture.

6.1. Requirement 1: Truly Distributed Storage

A truly distributed storage model is needed for RPKI Repository to provide reliable services for tens of thousands of RPs.

It means that Resource Certificate (RC) and ROAs are no longer only stored at the repository operated by the CA that issued them, but can be stored at multiple RPKI Repository nodes (PP or other names). The truly distributed storage architecture breaks the singleton nature of RPKI objects. It ensures that any single point of failure in the RPKI Repository nodes will not affect the integrity of RPKI snapshot retrieval by RPs. Additionally, since RPKI Repository nodes are located in different ASes and a single RPKI object can be stored on multiple nodes, the unwanted interdependence between the accessibility of a PP's objects and the reachability of the PP's AS will be broken. Since RPKI ensures the routing security and reachability of ASes, it is fair to expect RPKI objects to remain accessible during any incident when the Repository node or its corresponding AS become unreachable.

6.2. Requirement 2: Admission Mechanism

An admission mechanism is needed to effectively limit the unconstrained expansion of RPKI repository instances and enhance the scalability of RPKI infrastructure.

As AS administrators gain a deeper understanding of RPKI technology, delegated RPKI has emerged as a trend, the number of repository instances has grown more than 12 times in the past five years. The rapid growth in the number of repositories increases the cost of RP refreshes and threatens the scalability of RPKI. Furthermore, since each repositories connects to all RPs in the world, the low barriers to running repositories and joining RPKI Repository will introduce unforeseen risks to RPs because some repositories may emerge with unexpected intentions. Regardless, RPs should ensure they establish connections with trusted and reliable nodes. Therefore, a secure and scalable RPKI Repository needs an admission mechanism to raise the threshold for operating repository nodes. For example, the addition of a new repository node should be reviewed by existing node members or other trusted entities. The review can include verifying node's real identity, the reputation of the node operator, and whether it can provide a robust, secure, and reliable service to RPs, among other factors.

To implement an admission mechanism, it may be necessary to first implement "Decoupled from RPKI Authorities", because once repository nodes are bound to CAs, each CA has the right or need to operate its own repository.

6.3. Requirement 3: Decoupled from RPKI Authorities

RPKI Repository nodes should be decoupled from RPKI authorities (CAs).

It means that the PP is no longer bound to a specific CA. The CA only operates a certificate engine, not a publication repository, and the signing and management (especially storage) of the certificates will be separated. The operation of the RPKI Repository can be delegated to third-party entities (independent of CAs or subordinate INR holders), such as large Internet Service Providers capable of providing stable and reliable certificate storage and synchronization services to INR holders and RPs. Then, INR holders will no longer be required to store their certificates in the PPs operated by the upper-level CAs but can freely control the storage and management of their certificates, for example, upload their certificates to the repository nodes they trust. In this way, RPKI can empower INR holders with proactive control over their certificate management, including storage and even revocation, achieving a fair balance of rights between CAs and subordinate INR holders. This proactive control ensures that INR holders are not solely reliant on RPKI authorities, preventing potential abuse or unilateral actions that could compromise the authenticity of their certificates, such as deletion, corruption, modification, or unauthorized revocation.

Additionally, decoupling certificate issuance and certificate management can effectively prevent the repository nodes from growing significantly as the number of CAs increases. After all, it is reasonable for the RPKI certificate chain to become wider and deeper, but not for the RPKI Repository.

6.4. Requirement 4: Be Compatible with Current RPKI

Since the current RPKI is mature and widely deployed, the new RPKI Repository should not modify the RPKI hierarchical certificate issuance architecture and should not affect the existing RPKI certificate validation or the ROV process. It also should be compatible with current RPKI Repository architecture and supports incremental deployment.

7. Ethical Considerations

This section will outline the ethical considerations regarding the RPKI Repository measurement and the worldwide survey. First, we strictly limited the rate and number of DNS resolving packets (sending 130,000 packets over 24 hours) to avoid causing much burden on the Internet. For IP-to-AS mapping, we used open-source data provided by RIPE NCC. We conducted access experiments on the RPKI Repository from six globally distributed probes (India, Dubai, Silicon Valley, Frankfurt, Beijing, and So Paulo), with each RPKI repository being accessed fewer than 10 times to avoid DDoS attacks on the repositories. For the survey, we obtained the contact information of AS administrators from open-source WHOIS mailing lists

and provided an option for administrators to opt out of the survey. We did not disclose any additional information about the participating administrators or their feedback and ensured that their responses would be used solely for research.

8. Security Considerations

TBD

9. IANA Considerations

This document has no IANA requirements.

10. References

10.1. Normative References

[RFC6481] Huston, G., Loomans, R., and G. Michaelson, "A Profile for Resource Certificate Repository Structure", RFC 6481, DOI 10.17487/RFC6481, February 2012, <https://www.rfc-editor.org/rfc/rfc6481>.

[RFC8182] Bruijnzeels, T., Muravskiy, O., Weber, B., and R. Austein, "The RPKI Repository Delta Protocol (RRDP)", RFC 8182, DOI 10.17487/RFC8182, July 2017, <https://www.rfc-editor.org/rfc/rfc8182>.

[RFC6480] Lepinski, M. and S. Kent, "An Infrastructure to Support Secure Internet Routing", RFC 6480, DOI 10.17487/RFC6480, February 2012, <https://www.rfc-editor.org/rfc/rfc6480>.

[RFC8211] Kent, S. and D. Ma, "Adverse Actions by a Certification Authority (CA) or Repository Manager in the Resource Public Key Infrastructure (RPKI)", RFC 8211, DOI 10.17487/RFC8211, September 2017, <https://www.rfc-editor.org/rfc/rfc8211>.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/rfc/rfc2119>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/rfc/rfc8174>.

10.2. Informative References

[routing-history]
RIPE NCC, "routing history", 2024, <https://stat.ripe.net/docs/02.data-api/routing-history.html>.

[service-outage]
JAPNIC, "Service outage Roaweb and rpki repository (resolved)", 2020, <https://www.nic.ad.jp/en/topics/2020/20200521-01.html>.

[disk-outage]
JAPNIC, "Service outage Disk full caused lost roa validity", 2022, <https://www.nic.ad.jp/en/topics/2022/20220202-01.html>.

[routinator]
NLnet Labs, "Routinator", 2021, <https://github.com/NLnetLabs/routinator>.

[beyond-limits]
"Beyond Limits How to Disable Validators in Secure Networks", 2023, <https://dl.acm.org/doi/abs/10.1145/3603269.3604861>.

[stalloris]
"Stalloris RPKI Downgrade Attack", 2022, <https://www.usenix.org/system/files/sec22fall_hlavacek.pdf>.

[rpkiller] "rpkiller Threat Analysis from an RPKI Relying Party Perspective", 2022, <https://arxiv.org/pdf/2203.00993>.

[behind-the-scenes]
"Behind the Scenes of RPKI", 2022, <https://dl.acm.org/doi/pdf/10.1145/3548606.3560645>.

[RIPE-downtime]
RIPE NCC, "Rsync rpki repository", 2022, <https://www.ripe.net/ support/service-announcements/rsync-rpki-repository-downtime>.

Authors' Addresses

Dan Li
Tsinghua University
Beijing
China
Email: tolidan@tsinghua.edu.cn

Li Chen
Zhongguancun Laboratory
Beijing
China
Email: lichen@zgclab.edu.cn

Yingying Su
Zhongguancun Laboratory
Beijing
China
Email: suyy@mail.zgclab.edu.cn

Yu Fu
China Unicom
Beijing
China
Email: fuy186@chinaunicom.cn