RPKI 中 CA 资源分配风险及防护技术①

刘晓伟[1,2], 延志伟[2], 耿光刚[2], 李晓东[1,2]

[1](中国科学院大学 计算机网络信息中心, 北京 100190)
[2](中国互联网络信息中心 互联网域名管理技术国家工程实验室, 北京 100190)

摘 要: 边界网关协议在安全方面存在严重的缺陷, 容易导致路由劫持这一互联网安全威胁. 为此, 国际互联网工程任务组提出了资源公钥基础设施(Resource Public Key Infrastructure, RPKI)以防止路由劫持的发生. 然而随着 RPKI 技术的发展及其在全球范围内的部署, 与 RPKI 中认证权威相关的安全问题逐渐突显, 并受到广泛关注. 对 RPKI 中认证权威的资源分配过程进行研究分析, 通过实验测试, 验证了认证权威在资源分配的过程中资源重复分配和未获授权资源分配两种潜在的安全风险, 并分析了两种风险对资源持有者可能造成的不良影响. 此外, 针对这两种安全风险, 提出并实现了一种用于保证 RPKI 中认证权威资源分配安全性和准确性的"事前控制"机制, 该机制可以有效地防止资源重复分配和未获授权资源分配两种操作风险的发生, 减少了由于认证权威的错误操作所导致的故障恢复等待时间. 最后,通过进一步的实验测试, 验证、分析了这种"事前控制"机制的有效性和可行性.

关键词: 资源公钥基础设施; 认证权威; 资源重复分配; 未获授权资源分配; 事前控制

Resource Allocation Risks by CAs in RPKI and Feasible Solutions

LIU Xiao-Wei[1,2], YAN Zhi-Wei[2], GENG Guang-Gang[2], LI Xiao-Dong[1,2]

[1](Computer Network Information Center, University of Chinese Academy of Sciences, Beijing 100190, China)
[2](National Engineering Laboratory for Naming and Addressing, China Internet Network Information Center, Beijing 100190, China)

Abstract: There are serious security vulnerabilities in BGP (Border Gateway Protocol) which may lead to route hijacking. In order to overcome these BGP security defects, RPKI (Resource Public Key Infrastructure) was proposed by IETF (Internet Engineering Task Force). However, with the development and global deployment of RPKI, a lot of concerns about the security of certificate authority in RPKI have been raised. In this paper, it carries out experiments about two scenarios (resource reassignment and unauthorized resource assignment) on our RPKI testbed, and analyzes the security problems they may lead to, based on our research and analysis of the process of resource allocation. Besides, for these two kinds of security risks, this paper presents and implements a pre-control mechanism. Finally, it conducts further experiments on our testbed to prove that the pre-control mechanism we presented is feasible and effective to avoid the time limit for recovering from the failure caused by certificate authority's operational mistakes during the process of resource allocation.

Key words: RPKI; certificate authority; resource reassignment; unauthorized resource assignment; pre-control

互联网被划分为许多较小的自治系统(Autonomous System, AS), 目前, 自治系统之间的路由选择协议是 BGP(Border Gateway Protocol)协议[1], BGP 协议在安全方面的设计存在较大的不足: BGP 协议默认接受 AS 通告的任何路由, 因此, 即使一个 AS 向外通告不属于自己的 IP 前缀, 该路由通告也会被对端接受并继续传播[2]. BGP 的这一安全缺陷容易导致一种典型的互联网安全威胁——路由劫持. 现已发生的典型的路由劫持事件[3]包括: 1997 年 4 月的 AS 7007 事件、2004 年 12 月的土耳其电信集团劫持互联网事

① 基金资助项目:国家自然科学基金(61272433)
收稿时间:2015-12-15;收到修改稿时间:2016-01-28 [doi:10.15888/j.cnki.csa.005313]

件、2008 年 2 月的巴基斯坦劫持 YouTube 事件, 以及 2014 年 2 月的加拿大流量劫持事件等. 路由劫持对互联网的正常运行影响非常大, 可能会导致路由黑洞、流量窃听以及拒绝服务攻击等[4]. RPKI (Resource Public Key Infrastructure)的提出正是为了解决路由劫持, 目前, 与 RPKI 相关的技术标准在 IETF SIDR (Secure Inter-Domain Routing)工作组中得到了迅速的发展和积极的推进, 并且, RPKI 在全球的部署范围也正在逐步地扩大, 尤其是在南美洲和欧洲, 以及全球多个国家和地区都已经开始或完成了 RPKI 的实际部署工作.

针对域间路由系统存在的安全问题, RPKI 通过构建一个公钥证书体系来完成对互联网码号资源(Internet Number Resource, INR, 包括 IP 前缀和 AS 号)的所有权(分配关系)和使用权(路由源授权)的认证, 并以此“认证信息”来指导 BGP 中边界路由器的路由决策, 帮助其检验 BGP 报文中路由源信息的合法性和真实性, 从而有效地防止路由劫持的发生.

1 资源公钥基础设施RPKI

RPKI 依附于 INR 的分配过程[5]: 在 INR 的分配层次中, 最上层的是互联网号码分配机构(Internet Assigned Numbers Authority, IANA), IANA 将 INR 分配给 5 个区域性互联网注册机构(Regional Internet Registry, RIR), RIR 又可以将自己的资源向其下级节点如本地互联网注册机构(Local Internet Registry, LIR)、国家级互联网注册机构(National Internet Registry, NIR)和互联网服务提供商(Internet Service Provider, ISP)分配, 然后下级节点再依次逐级向下分配.

为了实现 INR 所有权和使用权的可认证, 在 RPKI 中要求每一层在向下层进行资源分配时, 必须签发相应的证书, RPKI 中的证书主要包括两种[6]: 认证权威(Certificate Authority, CA)证书和端实体(End Entity, EE)证书. CA 证书用于实现 INR 所有权的认证, EE 证书主要用于对路由源授权(Route Origin Authorization, ROA)的认证. RPKI 路由起源认证中最重要的对象就是 ROA[7], 它用于表明资源持有者授权哪个(或哪些)AS, 针对特定的 IP 前缀发起路由起源通告.

RPKI 的体系架构如图 1 所示, RPKI 包括 CA、资料库(Repository)和依赖方(Relying Party, RP)三个基本功能模块. 三个功能模块通过签发、存储、验证 RPKI 中的各种数字签名对象来相互协作, 并进行必要的数据通信, 共同完成 RPKI 的路由起源认证功能[8].

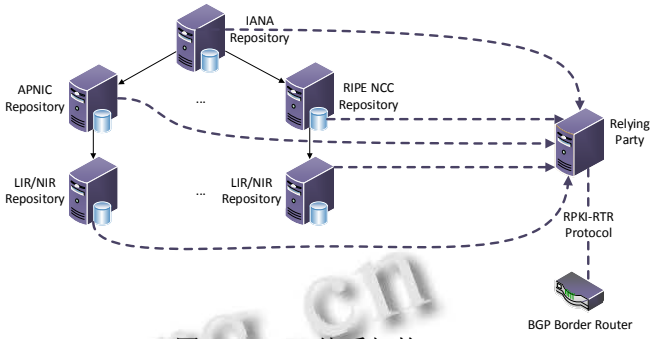

图 1 RPKI 体系架构

① CA 包括 IANA、RIR、NIR、ISP 等资源分配机构, 在进行资源分配时, CA 通过签发 CA 证书来表示 INR 的分配关系, 通过签发 ROA 来授权某个 AS 针对自己的一部分 IP 前缀发起路由起源通告.

② 资料库[9]用于存储 CA 发布的各种包含 INR 分配信息和授权信息的 CA 证书和 ROA 等数字签名对象, 并提供给全球的 RP 进行同步和验证.

③ RP 通过 rsync[10]或 RRDP[11]协议从资料库中同步 RPKI 的数字签名对象, 并使用 rcynic[12]等工具对这些数字签名对象进行验证[13], 将其处理成 IP 前缀与 AS 号的合法授权关系, 最后将该授权关系通过 rpki-rtr 协议[14]提供给 BGP 边界路由器, 用于指导其路由决策.

2 CA资源分配风险分析

IETF SIDR 工作组密切关注 RPKI 中由于 CA 错误操作所导致的各种安全风险[15], CA 的错误操作可能会导致严重的安全问题, 例如, 增加一个新的 ROA[16]可能会导致一个合法的路由被判定为无效(Invalid); 删除合法的 CA 证书[17]意味着资源的撤销, 会导致合法的资源持有者在网络中被迫下线. 更为严重的是, 一个 CA 节点的错误操作影响的不仅仅是该节点本身, 还包括该节点以下的各个节点. 因此, 发生错误操作的 CA 在 RPKI 层次结构中的位置越靠上, 则该 CA 造成的影响也可能越大. 例如, 如果发生错误操作的 CA 是一个处于低层次的 ISP, 那么这种错误只会限制在该 ISP 的局部范围内; 但如果发生错误操作的 CA 是 RIR 或 NIR, 那么这种错误会对该区域内所有相关节点(也包括这些节点的下级节点)造成严重影响.

在 RPKI 中 CA 的主要操作包括资源分配过程中 CA 证书的签发、ROA 等数字签名对象的签发、资料库的维护等，这些操作依附于资源分配，只有上级节点分配资源、下级节点获得资源后，才有资源的再次分配以及各种数字签名对象签发、资料库的维护等操作. 因此, CA 资源分配操作的安全性、准确性是 RPKI 正确实现其路由起源认证功能的重要保证.

本文针对在RPKI中CA向其下级节点进行资源分配的过程中两种潜在的操作风险：资源重复分配和未获授权资源分配，通过实验测试，验证两种操作风险是确实存在的，并针对这两种操作风险提出了一种可行的应对方案.

资源重复分配指的是同一资源被分配多次到不同的下级节点. 例如在图 2 中，假设 APNIC 已经将 ASN 65540-65550 和 IP 前缀 203.0.113.128/26 分配给了 JPNIC. 在 APNIC 对 CNNIC 进行资源分配时，APNIC 由于误操作或恶意操作将已经分配给 JPNIC 的 ASN 65540 和 IP 前缀 203.0.113.128/26 重复分配给了 CNNIC. 这样在CNNIC和JPNIC实际使用这些资源时，就会出现资源冲突、资源不可用等问题.

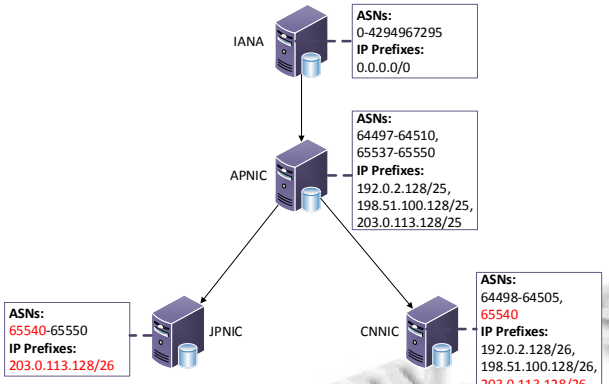

图 2　资源重复分配

未获授权资源分配指的是进行资源再分配的节点并不是该资源的合法持有者. 例如在图 3 中, APNIC 并不是 ASN 65551 和 IP 前缀 192.0.3.128/26 的合法持有者, APNIC 在向下级节点 TWNIC 进行资源分配时，由于误操作或恶意操作将该资源分配给 TWNIC，这样 TWNIC 在使用这部分资源时，也会出现资源不可用等问题.

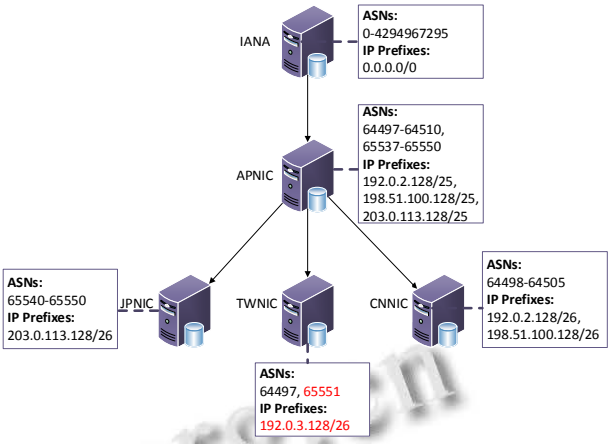

图 3　未获授权资源分配

3　CA资源分配风险实验验证

由 rpki.net 提供的 RPKI-CA[18]和 RPKI-RP[12]工具包，是目前支持功能最为完整的开源 RPKI 软件，本文所进行的测试场景均在该实验工具下完成. 本节按照第 2 节中资源重复分配和未获授权资源分配两种场景，分别进行实验测试，并分析两种风险可能导致的不良影响.

3.1 资源重复分配实验验证

通过 RPKI-CA 工具，按照图 2 中的 CA 节点层次结构，配置各节点间的父子关系，并进行相应的资料库配置，完成实验环境中各级CA节点的搭建. 在所搭建的实验环境中，对资源重复分配这一场景进行实验测试和验证. 在本次实验测试中，假设资源的重复分配发生在 APNIC 对 CNNIC 进行资源分配的过程中.

首先是 APNIC 对 JPNIC 进行资源分配，将 ASN 65540-65550 以及 IP 前缀 203.0.113.128/26 分配给 JPNIC. 在 APNIC 对 CNNIC 进行资源分配时，将 ASN64498-64505, 65540 以及 IP 前缀 192.0.2.128/26, 198.51.100.128/26, 203.0.113.128/26 分配给 CNNIC.

在两次资源分配的过程中，对 ASN 65540 和 IP 前缀 203.0.113.128/26 而言，就存在重复分配的问题. 在 APNIC 节点上，通过 show_child_resources 命令来查看 APNIC 分配给其子节点的资源，通过 show_published_objects 命令来查看 APNIC 在资源分配过程中生成的证书等数字签名对象，如图 4 所示.

从图 4 中我们能够看到 ASN 65540 和 IP 前缀 203.0.113.128/26 被重复分配给了 CNNIC 和 JPNIC 节点，并且用于表示本次资源分配的 CA 证书_O2UcMS

CjwRKP6ed4Thxmcb9SPM.cer 和 BvLK0ynrzTYPK7GoeXsVEy6SHuA.cer 被自动生成，并被发布在 APNIC 的资料库中. 我们可以通过 openssl 命令查看两个证书的内容(仅列出所分配的资源部分)，如图 5，图 6 所示.

```
root@~# rpkic -i apnic show_child_resources
Child: cnnic
  ASN: 64498-64505,65540
 IPv4: 192.0.2.128/26,198.51.100.128/26,203.0.113.128/26
Child: jpnic
  ASN: 65540-65550
 IPv4: 203.0.113.128/26
root@~# rpkic -i apnic show_published_objects
rsync://localhost/rpki/iana/apnic/GgVmNMq3R2uLf4Hed_v2u4sIWzU.crl 2
015-12-07T09:23:50Z 63BA161CED167618B4AE609C61C00C90BEB7A23B
rsync://localhost/rpki/iana/apnic/GgVmNMq3R2uLf4Hed_v2u4sIWzU.mft 2
015-12-07T09:25:47Z B1B3B3112E99F6846F84FAB2449F4763E52F3032
rsync://localhost/rpki/iana/apnic/BvLK0ynrzTYPK7GoeXsVEy6SHuA.cer 2
015-10-19T13:01:01Z 28D922AD601DBFB5193FB354F662B1251A4C5349 jpnic
rsync://localhost/rpki/iana/apnic/_O2UcMSCjwRKP6ed4Thxmcb9SPM.cer 2
015-12-07T09:25:47Z F1EB805B94A03EF98DDA552A2CB570317F542BF8 cnnic
```

图 4 查看 APNIC 资源分配和数字签名对象签发结果

```
sbgp-autonomousSysNum: critical
    Autonomous System Numbers:
      64498-64505
      65540

sbgp-ipAddrBlock: critical
    IPv4:
      192.0.2.128/26
      198.51.100.128/26
      203.0.113.128/26
```

图 5 查看 APNIC 向 CNNIC 资源分配的 CA 证书

```
sbgp-autonomousSysNum: critical
    Autonomous System Numbers:
      65540-65550

sbgp-ipAddrBlock: critical
    IPv4:
      203.0.113.128/26
```

图 6 查看 APNIC 向 JPNIC 资源分配的 CA 证书

从图 5 和图 6 中我们也能够验证资源 ASN 65540 和 IP 前缀 203.0.113.128/26 被重复分配给了 CNNIC 和 JPNIC 节点. 对于 CNNIC 和 JPNIC 节点，我们可以通过 show_received_resources 命令来查看他们从 APNIC 节点接收到的资源，如图 7 所示.

```
root@~# rpkic -i cnnic show_received_resources
Parent:      apnic
  notBefore: 2015-12-07T09:25:47Z
  notAfter:  2016-10-18T12:40:06Z
  URI:       rsync://localhost/rpki/iana/apnic/_O2UcMSCjwRKP6ed4Thx
mcb9SPM.cer
  SIA URI:   rsync://localhost/rpki/iana/apnic/cnnic/
  AIA URI:   rsync://localhost/rpki/iana/GgVmNMq3R2uLf4Hed_v2u4sIWz
U.cer
  ASN:       64498-64505,65540
  IPv4:      192.0.2.128/26,198.51.100.128/26,203.0.113.128/26
  IPv6:
root@~# rpkic -i jpnic show_received_resources
Parent:      apnic
  notBefore: 2015-10-19T13:01:01Z
  notAfter:  2016-10-18T12:40:10Z
  URI:       rsync://localhost/rpki/iana/apnic/BvLK0ynrzTYPK7GoeXsV
Ey6SHuA.cer
  SIA URI:   rsync://localhost/rpki/iana/apnic/jpnic/
  AIA URI:   rsync://localhost/rpki/iana/GgVmNMq3R2uLf4Hed_v2u4sIWz
U.cer
  ASN:       65540-65550
  IPv4:      203.0.113.128/26
  IPv6:
```

图 7 查看 CNNIC 和 JPNIC 从 APNIC 收到的资源

从图 7 中我们可以看到 ASN 65540 和 IP 前缀 203.0.113.128/26 都被 CNNIC 和 JPNIC 节点接收到了，这样在 CNNIC 和 JPNIC 实际使用这些资源时，就会出现资源冲突、资源不可用等问题.

3.2 未获授权资源分配实验验证

通过 RPKI-CA 工具，按照图 3 中的 CA 节点层次结构，配置各节点间的父子关系，并进行相应的资料库配置，完成实验环境中各级 CA 节点的搭建. 在所搭建的实验环境中，对未获授权资源分配这一场景进行实验测试和验证. 在本次实验测试中，假设未获授权资源分配发生在 APNIC 对 TWNIC 进行资源分配的过程中.

APNIC 不是 ASN 65551 和 IP 前缀 192.0.3.128/26 的合法持有者，但在 APNIC 向下级节点 TWNIC 进行资源分配时，将这些未获授权的资源分配给了 TWNIC. 在 APNIC 节点上，通过 show_child_resou-rces 命令来查看分配给其下级节点的资源，如图 8 所示.

```
root@~# rpkic -i apnic show_child_resources
Child: cnnic
  ASN: 64498-64505
 IPv4: 192.0.2.128/26,198.51.100.128/26
Child: jpnic
  ASN: 65540-65550
 IPv4: 203.0.113.128/26
Child: twnic
  ASN: 64497,65551
 IPv4: 192.0.3.128/26
```

图 8 查看 APNIC 分配给下级节点的资源

从图 8 中我们能够看到，对于 APNIC 节点而言，显示已经将 ASN 65551 和 IP 前缀 192.0.3.128/26 成功地分配给了其下级节点 TWNIC. 同样，对于 TWNIC 节点，我们可以通过 show_received_resources 命令来查看其从 APNIC 节点接收到的资源，如图 9 所示.

```
root@~# rpkic -i twnic show_received_resources
Parent:      apnic
  notBefore: 2015-12-07T10:04:00Z
  notAfter:  2016-12-06T10:01:06Z
  URI:       rsync://localhost/rpki/iana/apnic/BwNN00whK-P_hrvKqRdR
uGes7g8.cer
  SIA URI:   rsync://localhost/rpki/iana/apnic/twnic/
  AIA URI:   rsync://localhost/rpki/iana/GgVmNMq3R2uLf4Hed_v2u4sIWz
U.cer
  ASN:       64497
  IPv4:
  IPv6:
```

图 9 查看 TWNIC 从 APNIC 收到的资源

从图 9 中我们能够看到 ASN 65551 和 IP 前缀 192.0.3.128/26 并没有被 TWNIC 节点接收到，但 APNIC 认为这些资源已经被分配给了 TWNIC，因此会导致 TWNIC 无法实际使用这些资源的问题.

4　解决方案(事前控制机制)

如前所述，在 RPKI 中 RP 可以用于对各级 CA 产生的数字签名对象进行验证[12]，那么使用 RP 能否检测出资源的重复分配和未获授权资源分配两种错误操作?

如果通过现有的 RP 工具能够检测出资源的重复分配和未获授权资源分配两种错误操作，那么我们可以在将 RPKI 的各种数字签名对象发布到资料库之前就使用 RP 工具来对其进行验证，只有通过验证的数字签名对象才允许发布到资料库中，这样就可以防止两种错误操作对资源持有者造成的不良影响. 本文通过由rpki.net提供的RPKI-RP验证工具，对资源的重复分配和未获授权资源分配两种场景进行验证.

对于 APNIC 节点而言，其资料库位于/usr/share/rpki/publication/iana/apnic 目录，RP 验证后的结果存放在/var/rcynic/data/authenticated/localhost/rpki/iana/apnic 目录下. 通过查看两个目录下的文件及文件的内容，如图 10 所示，我们可以得出 APNIC 节点的资料库中所有的 CA 证书及其他数字签名对象都被认证为合法(通过 RP 验证为合法的数字签名对象才会被存放在/var/rcynic/data/authenticated/目录中).

```
root@~# ls /usr/share/rpki/publication/iana/apnic/
BvLK0ynrzTYPK7GoeXsVEy6SHuA.cer  jpnic
BwNN00whK-P_hrvKqRdRuGes7g8.cer  _O2UcMSCjwRKP6ed4Thxmcb9SPM.cer
cnnic                            twnic
GgVmNMq3R2uLf4Hed_v2u4sIWzU.crl  ZcC_fb1Bk69Hg50YrsYbxuOfqqg.roa
GgVmNMq3R2uLf4Hed_v2u4sIWzU.mft
root@~# ls /var/rcynic/data/authenticated/localhost/rpki/iana/apnic
BvLK0ynrzTYPK7GoeXsVEy6SHuA.cer  jpnic
BwNN00whK-P_hrvKqRdRuGes7g8.cer  _O2UcMSCjwRKP6ed4Thxmcb9SPM.cer
cnnic                            twnic
GgVmNMq3R2uLf4Hed_v2u4sIWzU.crl  ZcC_fb1Bk69Hg50YrsYbxuOfqqg.roa
GgVmNMq3R2uLf4Hed_v2u4sIWzU.mft
```

图 10　对比 APNIC 资料库与 RP 验证后的结果

由此得知，使用现有的 RP 并不能检测出资源重复分配和未获授权资源分配两种错误操作. 因此，我们需要对现有的 RP 进行改进，或提出新的解决方案用于实现对这些问题的检测和防范.

S. Kent 等人提出可以通过完善和改进现有 RP 的功能，使其对于可能是由于 CA 的错误操作导致的证书签发(包括本文所提出的资源重复分配和未获授权资源分配两种场景)、证书撤销等情况，进行滞后(hysteresis)操作[15]，这样 RP 在进行本次验证时，允许 RP 在一定的时间间隔[15]内仍然采用上次被验证为有效的对象，忽略本次验证时增加或减少的对象. 此外，对于在RP看来可能是错误操作但确实是CA需要完成的操作时，允许资源持有者通过一种独立于 RPKI 体系外的确认(confirmation)机制来通知 RP 使本次更新立即生效.

对于上述的滞后操作和独立确认机制可以在一定程度上(需要 RP 能够识别出这两种错误操作)，减少资源重复分配和未获授权资源分配两种错误操作带来的不良影响. 这种解决方案属于一种“事后检验”的机制，也就是在 CA 已经产生了错误的操作之后，利用改进后的 RP 对资料库中的所有数字签名对象进行验证，如果验证不通过，则向该CA发出相应的错误通知，使其能够进行及时的错误恢复. 这种机制存在以下几个问题:

① 滞后操作延迟时间的确定比较困难：延迟时间既要足够长，从而确保受影响的 CA 能够从相应的错误操作中恢复过来；但又要尽可能的短，避免该 CA 的合法更新操作被滞后过长时间，从而防止由于某些已经被撤销的签名对象继续有效或某些新增加的对象无法及时生效，而对 BGP 的路由决策造成的干扰.

② CA 要保证独立确认机制本身尽可能地安全和独立，这增加了 CA 的负担和操作复杂性.

本文提出一种“事前控制”机制，可以在资源分配的过程中，CA 证书签发之前进行控制，避免由于 CA 的错误操作导致非法资源证书的生成，从而防止资源重复分配和未获授权资源分配两种错误操作的发生. 这种事前控制机制主要体现在，一个正确的资源分配和证书签发过程，应该满足如下两个条件:

① 向下级节点进行分配的所有资源，必须全部从属于要进行资源分配的 CA 节点本身(防止未获授权资源分配)

② 满足条件 1 的任何资源，不能被重复分配到不同的下级节点(防止资源的重复分配)

本文提出的事前控制机制是在证书签发之前必须(只有满足事前控制机制的两个条件，才能进行后续的资源分配和证书签发操作)进行的操作，从而可以确保该机制对资源的重复分配和未获授权资源分配两种操作风险的检测和规避，并保证 CA 资源分配、证书签发的安全性和准确性. 本文对这一事前控制机制进行了初步的实现，流程如图 11 所示.

采用该机制后，在进行资源分配的过程中就会对要进行分配的资源进行检查，防止资源的重复分配和未获授权资源分配的发生：首先是对条件 1(未获授权

资源分配)的检查, 如图 12 所示(以图 3 中的未获授权资源分配作为测试场景), 如果在资源分配文件(.csv 文件)中存在不属于当前 CA 节点的资源, 则发出“Unauthorized Resources Found”警告, 显示检测到的不属于该节点的资源, 并提示对资源分配文件进行修改. 如果条件 1 满足, 则进行条件 2(资源的重复分配)的检查, 如图 13 所示(以图 2 中的资源重复分配作为测试场景), 如果在资源分配文件中存在某一部分资源被重复分配到不同的下级节点, 则发出“Resources Re-Allocation Found”警告, 显示被重复分配的资源, 并提示修改资源分配文件.

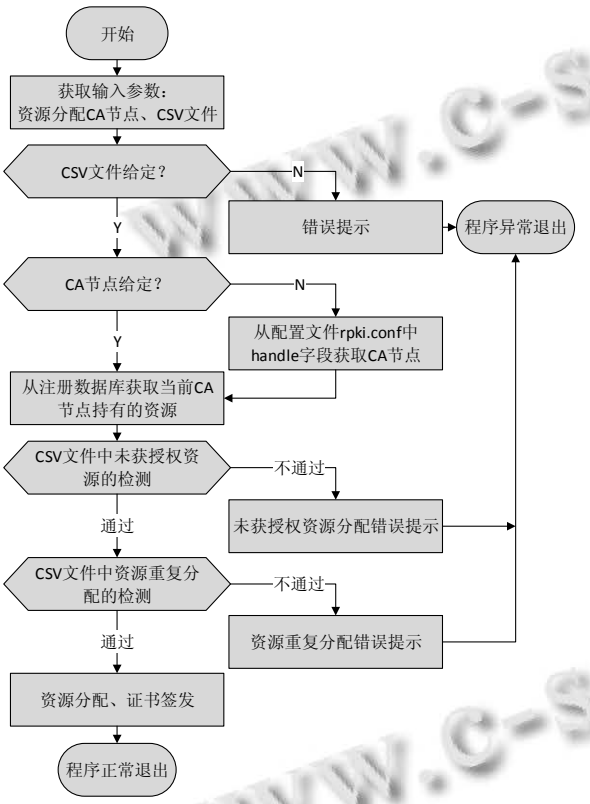

图 11 事前控制实现流程图

```
root@~# load_asns_secure -i apnic apnic2Asns.csv
Unauthorized Resources Found: apnic2Asns.csv "twnic   65551"
  AS 65551 does not belong to apnic
Please modify "apnic2Asns.csv", and run "load_asns_secure" again.
```

图 12 采用事前控制, 防止未获授权资源的分配

```
root@~# load_asns_secure -i apnic apnic2Asns.csv
Resources Re-Allocation Found: apnic2Asns.csv "65540"
  AS 65540 is allocated more than once.
Please modify "apnic2Asns.csv", and run "load_asns_secure" again.
```

图 13 采用事前控制, 防止资源的重复分配

当资源分配文件满足事前控制的两个条件之后, 才能够完成资源的正确分配和证书的签发, 如图 14 所示.

```
root@~# load_asns_secure -i apnic apnic2Asns.csv
root@~# load_prefixes_secure -i apnic apnic2Prefixes.csv
```

图 14 满足事前控制两个条件的资源分配

此时在 APNIC 节点上, 再次通过 show_child_resources 命令来查看已经分配给其下级节点的资源, 如图 15 所示, 我们可以看出, 采用本文提出的事前控制机制, 能够在证书签发之前就有效地检测到资源重复分配和未获授权资源分配两种错误操作, 从而防止错误的资源证书的生成. 此外, 这种事前控制机制能够尽可能地减少不必要的滞后操作, 省去了由于错误的证书签发、RP 的验证以及错误恢复导致的时间延迟.

```
root@~# rpkic -i apnic show_child_resources
Child: cnnic
  ASN: 64498-64505
 IPv4: 192.0.2.128/26,198.51.100.128/26
Child: jpnic
  ASN: 65540-65550
 IPv4: 203.0.113.128/26
Child: twnic
  ASN: 64497
```

图 15 再次查看 APNIC 分配给下级节点的资源

需要说明的是, 对于事前控制中的条件 2, 有一种特殊情况: 在资源迁移[19]的过程中, 允许某个中间状态, 从属于多个不同 CA 节点的 CA 证书中包含有同一块资源(对应于资源重复分配场景). 但资源迁移的最终结果是, 资源迁移的发起者必须签发新的资源证书, 且新的资源证书中不能包含已经被迁移到接收者的那些资源[19].

因此对于资源迁移过程中的事前控制, 应该做特殊处理, 允许处于该过程中的资源被多个不同的 CA 节点共同所有. 资源迁移的相关技术标准目前在 IETF SIDR 工作组中尚处于个人草案阶段, 针对这一特殊场景的一种可行的解决方案是: 只允许 TAO[20]对象中 ipAddrBlocks 和 asIdentifiers 两个字段所指定的资源被重复分配到不同的资源持有者, 而其他的资源仍然需要满足前述的资源分配和证书签发的两个条件. 这样, 事前控制机制能够兼容资源迁移这种特殊场景, 并有效地防止 CA 的错误操作.

5 结语

本文针对 CA 在资源分配过程中, 资源重复分配和未获授权资源分配两种潜在的安全风险, 通过实验测试, 验证并分析了两种风险及可能导致的后果、提出并实现了一种事前控制机制, 最后通过实验测试,

验证了该机制的可行性.

本文提出的事前控制机制, 可以有效地防止资源重复分配和未获授权资源分配两种操作风险的发生, 减少了由于两种操作导致的错误恢复等待时间, 加强了 CA 在资源分配过程中操作的安全性, 并为 RPKI 正确实现其路由起源认证功能提供了重要的安全保障.

参考文献

1 Rekhter Y, Li T, Hares S. A border gateway protocol 4 (BGP-4). IETF RFC 4271, January 2006.

2 黎松,诸葛建伟,李星.BGP 安全研究.软件学报,2013,24(1): 121−138.

3 IP hijacking. https://en.wikipedia.org/wiki/IP_hijacking.

4 Ballani H, Francis P, Zhang X. A study of prefix Hijacking and interception in the internet. ACM SIGCOMM, 2007.

5 Huston G, Michaelson G. Validation of route origination using the resource certificate public key infrastructure (PKI) and route origin authorizations (ROAs). IETF RFC 6483, February 2012.

6 Lepinski M, Kent S. An infrastructure to support secure internet routing. IETF RFC 6480, February 2012.

7 Lepinski M, Kent S, Kong D. A profile for route origin authorizations (ROAs). IETF RFC 6482, February 2012.

8 Bush R. Origin validation operation based on the resource public key infrastructure (RPKI). IETF RFC 7115, January 2014.

9 Huston G, Loomans R, Michaelson G. A profile for resource certficate repository structure. IETF RFC 6481, Feb 2012.

10 Weiler S, Ward D, Housley R. The rsync URI scheme. IETF RFC 5781, February 2010.

11 Bruijnzeels T, Muravskiy O, Weber B, Austein R, Mandelberg D. RPKI Repository Delta Protocol. draft- ietf-sidr-delta-protocol-01, October 2015.

12 RPKI Relying Party Tools. http://trac.rpki.net/wiki/doc/RPKI/RP.

13 Huston G, Michaelson G, Loomans R. A profile for X.509 PKIX resource certificates. IETF RFC 6487, February 2012.

14 Bush R, Austein R. The resource public key infrastructure (RPKI) to router protocol. IETF RFC 6810, January 2013.

15 Kent S, Ma D. Adverse actions by a certification authority (CA) or repository manager in the resource public key infrastructure (RPKI). draft-kent-sidr-adverse-actions-01, October 2015.

16 Cooper D, Heilman E, Brogley K, Reyzin L, Goldberg S. On the risk of misbehaving RPKI authorities. ACM Hotnets, November 2013.

17 Liu XW, Yan ZW, Geng GG, Lee XD, Tseng SS, Ku CH. RPKI deployment: Risks and alternative solutions. Genetic and Evolutionary Computing, 2015.

18 RPKI CA Engine. http://rpki.net/wiki/doc/RPKI/CA.

19 Austein R, Bush R, Huston G, Michaelson G. Resource transfer in the resource public key infrastructure. draft-ymbk-sidr-transfer-01, July 2015.

20 Barnes E. Resource public key infrastructure (RPKI) resource transfer protocol and transfer authorization object (TAO). draft-barnes-sidr-tao-00, February 2014.