A Framework to Evaluate MPIC Security using Real-World BGP Announcements

Henry Birge-Lee
Princeton University
Princeton, USA

Ari Brown
Princeton University
Princeton, USA

Christine Guo
Princeton University
Princeton, USA

Cyrill Krähenbühl
Princeton University
Princeton, USA

Sohom Pal
Rutgers University
New Brunswick, USA

Liang Wang
Princeton University
Princeton, USA

Prateek Mittal
Princeton University
Princeton, USA

Abstract

Multiple Perspective Issuance Corroboration (MPIC) is a defense that strengthens the Domain Control Validation protocol run by Certificate Authorities (CAs) against network attacks (e.g., routing hijacks). Despite its recent adoption as a requirement by the CA/Browser Forum, the quantitative security benefits of MPIC in light of real-world routing behaviors are not well understood. We seek to address this challenge by creating a framework to test the effects of real-world BGP hijacks on millions of potential MPIC perspective deployments. Our framework launches around 1500 ethical BGP hijacks on IP prefixes we own and analyzes how potential MPIC perspectives route under these attacks. We consider over 100 global MPIC perspective locations spread across 3 major cloud providers. We find that optimal MPIC deployments can prevent certificate misissuance for over 87% of our evaluated real-world BGP hijacks. We further show that different routing behaviors by cloud providers, such as cold potato routing, have a substantial effect on MPIC's ability to limit the impact of BGP attacks. Finally, our framework computes optimized sets of MPIC perspective locations for CAs to use given their preference of cloud provider and perspective count. Our recommendations have already impacted the MPIC deployment at Google Trust Services, and have been adopted as the default recommendation by the Open MPIC project.

CCS Concepts

• **Networks** → **Network security**; **Routing protocols**; *Network measurement*; • **Security and privacy** → *Web protocol security*.

Keywords

MPIC; PKI; BGP Hijacks; Routing; Cloud

ACM Reference Format:
Henry Birge-Lee, Ari Brown, Christine Guo, Cyrill Krähenbühl, Sohom Pal, Liang Wang, and Prateek Mittal. 2025. A Framework to Evaluate MPIC Security using Real-World BGP Announcements. In *Proceedings of the 2025 ACM Internet Measurement Conference (IMC '25), October 28–31, 2025, Madison, WI, USA*. ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3730567.3764495

IMC '25, Madison, WI, USA

ACM ISBN 979-8-4007-1860-1/2025/10
https://doi.org/10.1145/3730567.3764495

1 Introduction

Certificate Authorities (CAs) play a vital role in secure communication over the Internet. The digital certificates they sign serve as the basis of trust for TLS communication by tying a server's identity to a public key, and ensure secure communication between users and web servers. However, the Domain Control Validation (DCV) protocol used to issue these certificates is vulnerable to BGP attacks [3, 4]. A BGP attack during DCV can hijack traffic between the CA and a victim web server, thus misleading the CA about the domain's ownership. A successful hijack will result in an adversary gaining a certificate for a website they do not own, allowing them to decrypt any sensitive information sent to that site.

To address these vulnerabilities, modern PKI systems, such as Let's Encrypt, Cloudflare and Google Trust Services (GTS), employ Multi-Perspective Issuance Corroboration (MPIC). MPIC uses distributed "perspectives" across different geographic locations to perform DCV and increase resilience to routing-based attacks. During DCV, a certain threshold of vantage points, known as a quorum, must all route to the same web server for validation to succeed. More geographical diversity in a CA forces an adversary to capture more of the Internet, i.e., redirect traffic of more autonomous systems, for a successful attack. A recent ballot at the CA/Browser Forum made MPIC mandatory for all web PKI CAs [7]. Since September of 2025, **all CAs must implement MPIC and require a successful MPIC result to sign certificates**. These CA/Browser Forum regulations will lead to a rapid increase in the number of CAs deploying MPIC from just 2 to the entire industry of several hundred publicly-trusted CA/Browser Forum compliant CAs [23].

The emerging large-scale adoption of MPIC raises several pressing questions: how to quantify the security benefits of a given MPIC deployment? And how to configure MPIC deployment to maximize security? The resilience of an MPIC deployment depends on several factors, such as the cloud provider it is deployed on, the locations chosen for the perspectives, and the use of Resource Public Key Infrastructure (RPKI) [9]. However, a critical limitation of previous evaluations of MPIC [5, 9] is that **their primary evaluation is via simulations**. Internet topology simulations are highly inaccurate with one study showing deviations between real paths and simulated paths 80% of the time [16]. Furthermore, the methodology used to perform these simulations required costly internet-wide traceroutes be run at any datacenter which could host a perspective. This significantly limits the number of perspectives that can be considered and prevents simulations from being a viable evaluation of

1) opaque MPIC systems or 2) MPIC systems with large numbers of ephemeral nodes like those proposed by recent work [12, 13].

To address this challenge, we design and deploy a system that evaluates the effectiveness of MPIC **in the real world using ethically-launched real-world BGP attacks**. Our system, called MarcoPolo, launches nearly 1500 BGP attacks on the real Internet against IP prefixes we control to quantitatively measure the effectiveness of *any* MPIC deployment. Our system is immediately applicable as it can measure the effectiveness of an MPIC system as a black box, and captures the dynamics of advanced routing policies in use by real cloud providers that host MPIC. The results from MarcoPolo have already shaped the MPIC industry by helping Google Trust Services optimize their MPIC deployment and informing the recommended default perspective set used by the Open MPIC project [22]. Furthermore, we use MarcoPolo to perform the first ever evaluation of CloudFlare's MPIC API endpoint.

Our evaluation considers the largest set of potential datacenters for hosting MPIC perspectives, and gains the first insight into how real-world routing behaviors like cold potato routing in datacenters, and RPKI deployment impacts MPIC. Furthermore, we show that an effective MPIC deployment following the CA/Browser Forum's recommendation is possible on three major cloud providers. Finally, we observe that since some MPIC perspectives are allowed to fail during DCV, highly resilient perspective sets often exhibit multiple perspectives located in the same RIR to ensure that an adversary must capture at least one of the perspectives in that RIR to succeed.

2 Background

Domain Control Validation. Certificate Authorities (CAs) underpin secure Internet communication by issuing certificates that bind domain names to their legitimate public keys. This relies on Domain Control Validation (DCV), where domain owners demonstrate control over a core domain resource (e.g. DNS records, or served files). Our paper focuses on HTTP validation, in which the CA verifies the presence of a challenge token served by the web server over an insecure plaintext channel. An adversary can use BGP hijacks to reroute the CA's challenge to their own servers and successfully return their token, thus obtaining a faulty certificate [2, 17].

MPIC. Multiple Perspective Issuance Corroboration (MPIC) reduces the success rate of BGP attacks on DCV. Rather than performing DCV from a single location, it is performed from multiple globally distributed locations (perspectives) [5, 9]. Since equally-specific BGP hijacks only compromise a portion of the internet, the CA can check for any discrepancies in the retrieved content of a given resource, as a sign of an ongoing attack [4, 5, 9]. Note that MPIC does not protect against more specific prefix hijacks, since their effect is observed globally by all perspectives. Let's Encrypt and Google Trust Services currently require successful MPIC for all certificates. **As of September 2025, all CAs must perform MPIC and block issuance in case of failure**. Given that a significant number of web PKI CAs are currently finalizing their MPIC deployments, answers regarding perspective locations, quorum policy (the maximum number of perspective failures that still allow certificate issuance), and cloud provider choice are critical to the CA industry.

BGP Attacks and RPKI Defenses. The Resource Public Key Infrastructure (RPKI) is used to authenticate IP address and autonomous system number (ASN) resources. RPKI allows ASes to create Route Origin Authorizations (ROAs), which are cryptographically signed objects mapping a set of IP prefixes, with optional maximum prefix lengths (`MAX_LEN` attribute), to an authorized origin AS. Routers employing RPKI-based Route Origin Validation (ROV), prevent propagation of BGP announcements that specify a wrong origin AS by inspecting existing ROAs for the given announcement prefix. However, since ROV only validates the originating AS of a BGP announcement, it can be bypassed by a "forged-origin" hijack that prepends the true origin ASN to the adversary's own ASN at the cost of an extended route length that reduces propagation. Hence, ROA-covered prefixes are still subject to forged-origin equally-specific (localized) prefix hijacks, whose effectiveness varies depending on the topological distance between the adversary and victim. The use of the `MAX_LEN` attribute is discouraged since it additionally allows for forged-origin sub-prefix (globally effective) hijacks unless a legitimate BGP announcement for the most specific prefix exists [14]. Combining the benefits of ROV with MPIC in DCV can help further mitigate BGP attacks on certificate issuance [9]. In contrast to ROV's partial defense against BGP hijacks, BGPSEC and secure routing architectures such as SCION can provide strong protection against path hijacks but currently lack widespread adoption.

3 Ethics

We took extreme care to perform our experiments in an ethical manner. Our experiments involved real-world BGP announcements but **we only announced IP prefixes we controlled which ran no production network services**. We also took care to preserve global Internet stability and minimize routing load by only changing our BGP announcements infrequently and allowing sufficient time for BGP propagation. When interacting with production certificate and MPIC systems (i.e., Let's Encrypt and CloudFlare's MPIC API) the load we imposed on these systems was minimal (about a single request every 5 min). When interacting with Let's Encrypt, we used the staging environment (which does not sign publicly trusted certificates) and never finalized requests after DCV, preventing any real certificate issuance during our experiments.

4 MarcoPolo

To facilitate the evaluation of MPIC deployments, we develop **MarcoPolo**, a global distributed system designed to measure the routing behavior of any MPIC deployment by executing hundreds of real-world, pairwise BGP attacks and observing the behavior of the CA's validation perspectives as shown in Fig. 1. MarcoPolo can be instantiated with a real CA's existing MPIC deployment to analyze its resilience to BGP hijacks, or it can be run globally to compute optimal MPIC deployments based on hundreds of potential perspective locations. Source code and results are available online [19]. Appendix D provides a total cost summary of the experiment.

4.1 System Design

MarcoPolo consists of two main components. First, a globally distributed set of **nodes** that act as *victims* and *adversaries*. Since these nodes must perform BGP hijacks, we hosted them on Vultr, a platform that permits users to originate BGP announcements. Second, an **orchestrator** that coordinates the attacks and collects results.

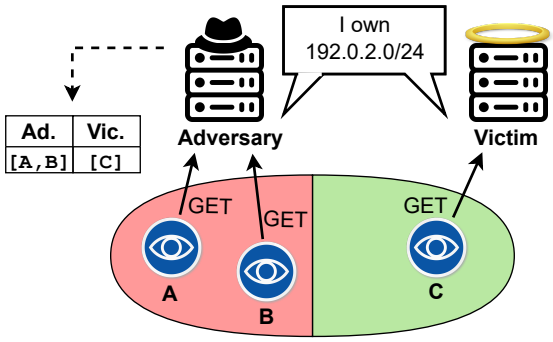


Figure 1: Adversary performing a BGP Hijack to redirect victim traffic to an adversary-controlled server. MarcoPolo records the routing behavior of a given MPIC deployment.

Note that the MPIC deployments analyzed by MarcoPolo, described in Section 4.3, are not considered part of the MarcoPolo framework itself. Each attack is coordinated by the central server as follows:

(1) A unique node pair is selected to act as victim and adversary.
(2) Both victim and adversary announce the IP prefix associated with the victim domain simultaneously.[1]
(3) The BGP announcement is given 5 minutes to propagate.
(4) A certificate is requested for a domain that resolves to the victim's IP address, triggering all perspectives to send DCV requests to either victim or adversary, depending on the hijack's effect on the perspective.
(5) The victim and adversary record and store the originating IP addresses for these requests at the central server. The attack is run again if any perspective requests were not received.

This attack is run for every configuration of victim and adversary in the node pool, resulting in a complete dataset characterizing perspective outcomes per victim–adversary pair. See Appendix E for a full list of nodes considered. MarcoPolo can also run attacks for victims using RPKI ROA to tie BGP prefixes to authorized origin ASNs. To circumvent ROA protection, adversaries can prepend the authorized ASN to their AS path [17], which increased AS path length and decreases propagation of the attack. We prepend a single hop to our announced path to mimic an adversary's path length.

Once perspective-level routing behavior is collected, MarcoPolo enables flexible, post-hoc analysis of any MPIC deployment. Given raw logs, we can simulate arbitrary combinations of perspective selection strategies and quorum policies to compute a resilience metrics. This allows us to evaluate specific deployments or exhaustively search over all valid configurations to identify the most resilient design under any set of constraints. Concretely, we define the resilience of a victim BGP node v as the likelihood of MPIC preventing the issuance of a fake certificate for the victim: $R(v) = 1 - S/A$, where S is the number of adversary BGP nodes that can successfully leverage a BGP hijack to trick the MPIC deployment into issuing a fake certificate for v, and A is the total number of adversaries. The overall resilience of an MPIC deployment is then defined as the median of the resilience values of all possible victim BGP nodes $\mathcal{V}$ as $R_{\text{MPIC}} = \text{median}(\{R(v) \mid v \in \mathcal{V}\})$. A formal definition is given in Appendix A. The flexibility of MarcoPolo allows the definition of customized resilience metrics. For example, a resilience metric looking at the 5th percentile could be used to find MPIC deployments which perform reasonably well for virtually all domains but may not provide high resilience for a majority of domains.

[1]The impact of announcement timings is discussed in Section 4.4.4.

4.2 Overcoming Challenges for Robust MarcoPolo Deployment

Running MarcoPolo in practice exposed several engineering and operational challenges, stemming from protocol constraints and ecosystem-specific behaviors. We highlight three key issues and our mitigation strategies below.

4.2.1 Attack Challenges. Running MarcoPolo required careful handling of BGP announcements, as excessive route flapping can destabilize our prefixes or potentially lead to Route Flap Dampening (RFD). We carefully constrain the announcement frequency to at most one announcement every 5 minutes, which produced stable BGP routes based on our propagation measurements.

4.2.2 MPIC Interaction. After each BGP hijack, MarcoPolo interacts with multiple MPIC deployments to trigger DCV and record its outcome. Although the interaction is conceptually simple, i.e., trigger DCV and record the outcome, the heterogeneity of current MPIC implementations posed a significant engineering challenge.

Adapting to MPIC interface diversity. We implemented two main MPIC deployment interfaces: RESTful MPIC APIs (e.g., Open MPIC, and CloudFlare) and ACME-based MPIC that is only triggered during a certificate request (e.g., Let's Encrypt and Google Trust Services). We defined different MPIC requesting methods to accommodate these CA-specific behaviors.

Handling Certbot complexity. For MPIC deployments that were triggered by ACME certificate requests, we interacted with the CAs using Certbot, which introduces several complications: (1) *Challenge caching*: a successive certificate request for the same domain would use the cached result, without triggering DCV. We circumvented this by using randomized subdomains on each request. (2) *Certificate issuance limits*: We used a manual authorization script that aborted before finalizing certificate requests to avoid any complete certificate issuance. (3) *Pre-flight validation*: CAs using ACME (Let's Encrypt, GTS) required that a pre-flight check from a single perspective pass validation before the remaining perspectives requested DCV. Since we did not know in advance whether the request would route to the victim or adversary, we had to ensure both nodes could complete the DCV challenge successfully by forwarding all requests to the central server (where the challenge token was being served by the ACME client) and returning the same token.

4.2.3 Scalability. Restricting the BGP announcement frequency to 5 minutes leads to an upper limit on the speed at which attacks can be carried out. To mitigate this and ensure scalability, MarcoPolo provides two ways of increasing performance:

Concurrent MPIC evaluations. Because the BGP hijack (steps 1–3) is identical across all MPIC deployments for a given victim-adversary pair, we simultaneously evaluate multiple deployments

by issuing certificate requests (steps 4–5) for each one in parallel. This batching significantly improved efficiency and enabled simultaneous data collection across multiple configurations.

Parallelism through prefix partitioning. Multiple victim-adversary pairs can be tested in parallel, by distributing attacks across multiple BGP prefixes. This partitioning allows each set of announcements to proceed without interference, reducing the total experiment time proportionally. The 5 minute propagation delay is enforced per-prefix. While we did not use this feature—given our limited prefix blocks and manageable experiment duration—the quadratic growth of full attack runs with respect to victim and adversary pool size makes it invaluable to ensure scalability.

4.3 Evaluating Global MPIC deployments

Using MarcoPolo, we evaluated a global MPIC deployment across as many data centers as possible on three major cloud providers. In total, we deployed 106 perspectives: 27 on AWS, 39 on Azure, and 40 on GCP (see Appendix E for a full list). To maximize coverage, we used Terraform to provision virtual machines in every available region. On AWS, we ran a serverless configuration of Open MPIC via Lambda functions, while GCP and Azure used a VM-based MPIC variant we developed. On Azure and GCP, each perspective hosted a Flask application to perform DCV, and each deployment designated one VM as a central controller to coordinate and collect responses. When a certificate request was received, the central server triggers DCV across all perspectives. Each perspective performs an HTTP request to the DNS-resolved IP address of the target domain and reports back whether the DCV challenge succeeded. These results are aggregated by the central server and returned to the requester. Additionally, we also included two industry MPIC deployments—Let's Encrypt and Cloudflare [18]—to observe the effects of BGP hijacks on production-grade systems. This setup allowed us to run real-world hijacks and observe their impact across a wide range of perspective configurations. By doing so, we could identify the most resilient MPIC deployments globally and evaluate optimal configurations under various constraints, such as specific quorum thresholds, cloud providers, or regional subsets.

4.4 Limitations and Future Improvements

We observe the following limitations and future improvements:

4.4.1 Focus on Equally-Specific Prefix Hijacks. MPIC is designed to (work alongside ROA protection) and increase the resilience of CAs against equally-specific prefix hijacks by requiring the adversary to hijack a large part of the Internet to attack domain control validation. We simulate the effect of ROA-protected prefixes by adding an additional ASN to the adversary's BGP announcement to simulate an equally-specific forged-origin hijack. In a future iteration of MarcoPolo, we could additionally create an ROA for the victim domain (with or without `MAX_LEN` attribute) to investigate its impact on MPIC's resilience. One challenge is that the frequent creation and revocation of ROAs may impact scalability.

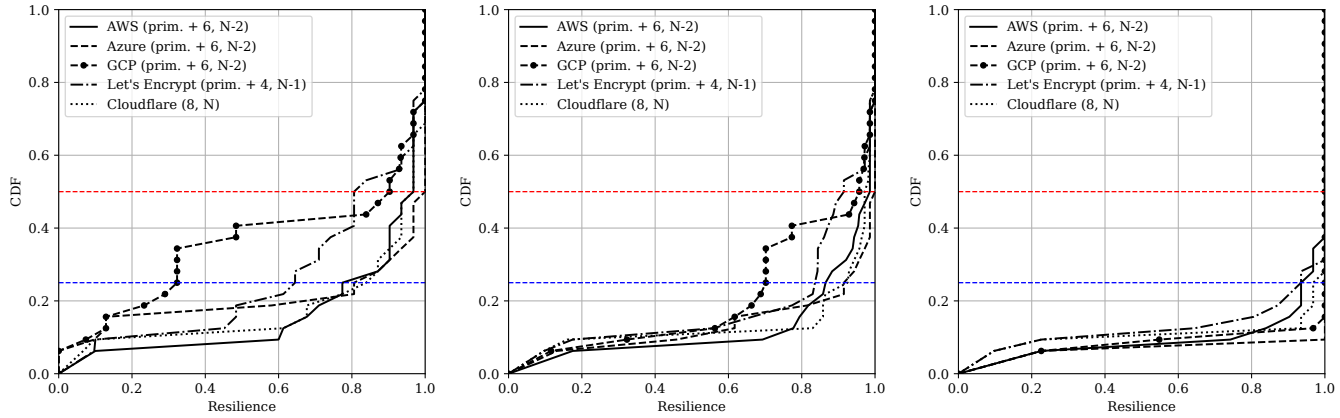
4.4.2 Adversary and Victim Locations. In our evaluation, all adversaries and victims were hosted in Vultr data centers since they allow custom BGP announcements, had different mixes of tier-1 transit providers, and were part of different tier-1 cones, e.g., Vultr Tokyo and Bangalore are part of the different tier-1 cones NTT Communications (AS2914), and Tata Communications (AS6453), respectively. However, due to multi-regional business relationships, spreading datacenters across different operators could reduce a potential bias from over-representing the routing behavior of select Vultr tier-1 providers. We believe the routing behavior of the tier-1 providers chosen by Vultr is generally similar to the broader set of tier-1 providers on the Internet, leading to a minimal impact on our results. Regardless, selecting victims and adversaries from diverse datacenter providers, or use PEERING as a superset of Vultr, could produce more generalizable results and is a topic of future work. Additionally, it is an open question if Vultr locations should be weighted differently to accommodate for realistic distributions of victims, e.g., frequently targeted cryptocurrency platforms [2, 17], and adversaries, e.g., based on historical data of AS misbehavior.

4.4.3 Perspective Locations. We use data centers of three major cloud providers to deploy MPIC perspectives since (1) these providers offer an easy and low-cost MPIC deployment option, (2) they have the global presence necessary to satisfy CA/Browser Forum RIR and geographic requirements, and (3) current MPIC deployments, e.g., Let's Encrypt, are already using cloud providers for their deployment. An interesting future research project would be to compare the effectiveness of cloud-based MPIC deployments with alternative MPIC deployment strategies.

4.4.4 Simultaneous Victim and Adversary BGP Announcements. In MarcoPolo, the victim and adversary BGP announcements are sent out simultaneously, potentially leading to a nondeterministic "route age" tie break in the best path selection process based on which announcement was first received by the BGP router. Since adversaries and victims are operated by the same datacenter provider Vultr, the routes from perspectives may in some cases be similar enough to observe the effect of tie breaks and result in the reported resilience in the range $[R_{\min}, R_{\max}]$, where $R_{\min}$ is the resilience if the adversary announcement is always received first, which represents the worst case, and $R_{\max}$ is the resilience if the victim announcement is always received first, which represents the typical hijack case. Furthermore, if announcement arrival times vary between attack runs, e.g., due to varying congestion levels, this may lead to non-optimal perspective set selection. However, we believe that the impact of this nondeterminism on our results is small since the tie break only comes into play if most other criteria, e.g., localpref, AS path length, and MED, are identical, and Vultr's routing diversity should not lead to this effect being a major factor. Ideally, MarcoPolo would send announcements sequentially, but the tradeoff of such asymmetric attacks is an increase the total experiment duration by a factor of 2.67, which significantly raise the cost of running the experiment.

5 Results

For all long-term CA/Browser Forum compliant perspective counts and quorum policies, we find that cloud-based MPIC deployments provide strong protection for non-RPKI protected prefixes against equally-specific BGP hijacks. Figure 2a shows that, without RPKI, optimal MPIC deployments in the cloud providers we studied had a median resilience of at least 87%. While without MPIC, the



(a) No RPKI Deployment **(b) Current RPKI Deployment** **(c) Full RPKI Deployment**

Figure 2: The resilience of various cloud-based and production MPIC deployments under different RPKI conditions. The horizontal blue line indicates 25th percentile resilience and the red line indicates median resilience.

maximum-achievable median resilience is 53%, highlighting the importance of MPIC with multiple, globally-distributed perspectives (see Table 2). We further analyze the suitability of the CA/Browser Forum MPIC requirements, as well as the impact of local routing behavior, RIR clustering, and RPKI on MPIC's hijack resilience. All experiments were conducted between April and May 2025. For notation, we use $(X, N-Y)$ to represent an MPIC deployment with X remote perspectives under an $N-Y$ quorum policy, which indicates that a certificate will be issued, if at least $X-Y$ perspectives agree, i.e., allowing at most Y perspectives to disagree.

5.1 Resilience of Cloud-Based MPIC

The CA/Browser Forum recently passed a proposal requiring the adoption of MPIC for all CAs with a quorum policy $q \geq N-1$ for 2-5 remote perspectives and $q \geq N-2$ for 6+ remote perspectives for performing DCV [8]. We analyze the optimal MPIC deployments that meet these quorums both with and without a primary perspective. We calculate the median resilience for every possible MPIC deployment with X perspectives under an $N-Y$ quorum, and choose the set of perspectives with the highest resilience value (see Appendix A for details). We focus on perspective sizes ≥ 5 since smaller perspective sets are not permissible in the long term (after December 2026 [8]). The non-MPIC deployment (e.g., a deployment with only one perspective) achieves at most a median resilience of 53%, whereas for MPIC deployments without a primary perspective, the optimal (5, N-1) and (6, N-2) MPIC deployments achieve a median resilience of at least 89% and 87%, respectively (see Appendix Table 2). Adding an optimal primary perspective, which must succeed to allow issuance, increases the median resilience values to at least 92% and 90%, respectively (see Appendix Table 2). We did not evaluate higher perspective counts, since increasing this count always improves resilience. Furthermore, we evaluate two currently deployed cloud-based MPIC systems, namely Let's Encrypt (primary + 4, N-1) and Cloudflare (8, N), showing that both achieve strong resilience, with median values of 82% and 97% respectively. The resilience values of different cloud providers are shown in Appendix C. MarcoPolo provides a comprehensive list of CA/Browser Forum compliant MPIC deployment configurations ordered by their respective resilience for a given cloud provider. **The Open MPIC project, used as the MPIC implementation by several major CAs, updated their default recommendations for the AWS perspective set based on these results.**

5.2 Impact of Cold Potato Routing on MPIC

Across cloud providers, we observe different routing patterns which lead to a difference in resilience values. Some cloud providers transmit network traffic within their own network for as long as possible, also known as cold potato routing. We used GCP's Premium Tier routing which implements this behavior and is the default routing tier for the GTS MPIC deployment [10]. Their goal is to maximize the service provider's control over the end-to-end quality of service, to ensure faster and more reliable connections for its customers. In contrast, other providers such as AWS generally pass network traffic off to any egress point to reduce the load on their network, which tends to create more routing diversity due to packets being distributed to a wider variety of peers. These routing policy differences are reflected in the resilience results, show in Table 2.

Our results indicate that Azure deployments provide the best resilience, closely followed by AWS. For a given quorum policy, GCP typically provides the lowest median and average resilience, likely due to their lack of routing diversity caused by cold potato routing. We reported our findings to GTS, allowing them to adjust their perspective set and quorum policy to improve the resilience of their deployment. It is important to note that cold potato routing does not necessarily hamper resilience completely, but can impact the required amount and locations of perspectives, e.g., using correct configurations, GCP is a viable option for deploying MPIC.

5.3 RIR Clustering Follows Quorum Size

For the best MPIC deployments, we observe, across cloud providers, that perspectives tend to form clusters in specific Regional Internet Registries (RIRs). The goal of MPIC is to ensure that an adversary

must hijack multiple portions of the network to obtain a fake certificate. If all perspectives must succeed to obtain a certificate, the optimal strategy is to choose perspectives in the most diverse network locations. However, in an N-Y quorum, where Y perspectives may fail but still allow certificate issuance, this may not be the case. If a region contains Y perspectives, the adversary can ignore hijacking that region and still obtain a fake certificate. Thus, the optimal strategy is to select "clusters" of Y+1 perspectives in as many regions as possible to ensure that at least one of perspective in that region must succeed to allow certificate issuance. Our hypothesis is that since RIRs approximate nearby network locations with similar routing behavior (i.e., networks within the same RIR tend to have similar connectivity to other RIR regions, leading to comparable "global" routing patterns), this clustering effect is also shown by preferring clusters of Y+1 perspectives per RIR. **The advantages of clustering goes against the common belief that deploying in the most diverse regions, say one per RIR, provides the strongest resilience regardless of the chosen quorum.** Concretely, looking at (6, N-2) MPIC deployments with the highest resilience values, 64–89% of high-resilience MPIC deployments exhibit this clustering behavior with two RIRs containing 3 remote perspectives each, and the primary perspective located in a third RIR. Appendix B describes these results in more detail.

5.4 Synergy Between MPIC and RPKI

IP prefixes protected by RPKI (i.e., those having a valid ROA without the MAX_LEN attribute specified) are significantly more secure against BGP hijacks. We observe this across all cloud provider MPIC deployments and current CAs like Cloudflare and Let's Encrypt. At the implementation level, MPIC could strategically select cloud provider perspectives hosted by ASes that enforce ROV. Figure 2b displays the resilience of IP prefixes under the state of RPKI deployment as of May 2025 [21]. To loosely approximate this current state, we model 56% of IP prefixes as having deployed RPKI and the latter 44% as without RPKI protection by calculating the resilience value as a weighted sum, with 56% of the resilience value contributed from the RPKI attack results and 44% from the non-RPKI ones. When compared to a model without RPKI deployed, the median resilience of the top performing (primary + 6, N-2) GCP MPIC deployment experiences an increase of 6 percentage points (see Fig. 2b). Although the respective AWS deployment experiences a smaller gain of 2 percentage points and Azure already achieves a median resilience of 100% in the non-RPKI model, their resilience against BGP hijacks which target the weaker portion of IP prefixes sees major gains—the 25th percentile resilience increases by nearly 10 percentage points for both deployments, achieving a value above 90%. While these cloud provider MPIC deployments already achieve a median resilience above 90% without RPKI, their 25th percentile resilience is low and benefits from the deployment of RPKI. Cloudflare and Let's Encrypt experience similar resilience gains under RPKI. When compared to the no RPKI model, under the current status of RPKI, Let's Encrypt sees a nearly 10 percentage point boost, achieving a median resilience of 92%. While the Cloudflare median resilience gains are more limited, moving from 97% to 98%, its 25th percentile resilience increases by 7 percentage points to 93%. Figure 2c models IP prefixes under a full RPKI deployment. **Introducing full RPKI further boosts the median resilience to 100% for all considered CAs and cloud provider MPIC deployments.**

6 Related Work

Both real-world and simulated attacks have been analyzed to evaluate the resilience of MPIC against BGP hijacks, which included both equally-specific prefix and equally-specific-prefix prepending attacks [5, 9]. Prior work has shown that perspective selection, cross-cloud providers, and appropriate quorum policies are critical to MPIC deployments [5, 9]. In particular, Birge-Lee et al. used the PEERING platform to analyze the impact of real-world BGP announcements. However, the testbed's limited locations and domains restricted the experiment run to just 62 different victim-adversary attack configurations [5]. In order to explore a wider set of adversaries, Cimaszewski et al. performed extensive simulated hijacks which were modeled using CAIDA topology [9]. The peering lists were inferred for 19 data centers spanning AWS, Microsoft Azure, and GCP. However, these simulations are limited by the underlying accuracy of the CAIDA AS-Relationship dataset which has been shown to often miscalculate paths [15]. Furthermore, the true dynamics of cloud provider outbound routing cannot be captured.

We go beyond these works by performing an in-depth evaluation of real-world BGP hijacks on MPIC deployments hosted on major cloud providers. Using Vultr, we launched nearly 1500 equally specific prefix(-prepending) attacks, also targeting RPKI-protected IP prefixes, while deploying MPIC on 106 datacenters across AWS, Azure, and GCP, to greatly increase the real-world applicability.

In addition to the cloud-based MPIC used by CAs like Let's Encrypt [5], recent works propose MPIC in diverse environments including Tor [12] and advertisement networks [13]. MarcoPolo can be used to analyze these implementations which is a topic of future work. MPIC has also been proposed to help against non-BGP hijacks like DNS cache poisoning and passive man-in-the-middle (MITM) attacks [6, 11]. In our work, we did not consider the DNS attack surface because we cannot properly model the locations for the authorizing DNS servers. Building upon our framework to capture the influence of routing attacks on the DNS attack surface, similar to the DNS infrastructure hijack analysis by Akiwate et al. [1], is an interesting direction for future work.

7 Conclusion

The implementation of a new security feature, like MPIC, can be challenging due to the need to correctly setup the system and choose appropriate (security) parameters, such as perspectives and quorum policies. The CA/Browser Forum requirements for MPIC still leave many decisions to CAs. Our testing framework MarcoPolo leverages ethical BGP hijacks to capture AS-local routing behavior, which is not possible with existing simulation-based frameworks. MarcoPolo enables automatic evaluation of MPIC deployments and thus quantitative comparison of competing deployments. Our results are freely available on the MPIC Labs webpage [20]. This informed decision making strengthens the overall CA ecosystem and even allows smaller, more resource-constrained, CAs to achieve strong resilience against BGP hijacks. Finally, by showing tangible benefits of RPKI for mitigating the effects of BGP hijacks on MPIC, we believe that our work will help further drive RPKI deployment.

References

[1] Gautam Akiwate, Raffaele Sommese, Mattijs Jonker, Zakir Durumeric, KC Claffy, Geoffrey M. Voelker, and Stefan Savage. 2022. Retroactive identification of targeted DNS infrastructure hijacking. In *Proceedings of the ACM Internet Measurement Conference (IMC)*. doi:10.1145/3517745.3561425

[2] Henry Birge-Lee. 2022. Attackers exploit fundamental flaw in the web's security to steal $2 million in cryptocurrency. https://perma.cc/CW4F-SG7R

[3] Henry Birge-Lee, Yixin Sun, Annie Edmundson, Jennifer Rexford, and Prateek Mittal. 2017. Using BGP to Acquire Bogus TLS Certificates. In *Proceedings of the ACM Workshop on Hot Topics in Privacy Enhancing Technologies (HotPETs)*.

[4] Henry Birge-Lee, Yixin Sun, Anne Edmundson, Jennifer Rexford, and Prateek Mittal. 2018. Bamboozling Certificate Authorities with BGP. In *Proceedings of the USENIX Security Symposium*.

[5] Henry Birge-Lee, Liang Wang, Daniel McCarney, Roland Shoemaker, Jennifer Rexford, and Prateek Mittal. 2021. Experiences Deploying Multi-Vantage-Point Domain Validation at Let's Encrypt. In *Proceedings of the USENIX Security Symposium*.

[6] Markus Brandt, Tianxiang Dai, Amit Klein, Haya Shulman, and Michael Waidner. 2018. Domain Validation++ For MitM-Resilient PKI. In *Proceedings of the ACM Conference on Computer and Communications Security (CCS)*. doi:10.1145/3243734.3243790

[7] CA/Browser Forum. 2024. Ballot SC067v3: Require domain validation and CAA checks to be performed from multiple Network Perspectives Corroboration. https://perma.cc/2RGC-TPYA

[8] CA/Browser Forum 2025. *Baseline Requirements for the Issuance and Management of Publicly-Trusted Certificates*. Version 2.1.4. CA/Browser Forum.

[9] Grace H. Cimaszewski, Henry Birge-Lee, Liang Wang, Jennifer Rexford, and Prateek Mittal. 2023. How Effective is Multiple-Vantage-Point Domain Control Validation?. In *Proceedings of the USENIX Security Symposium*.

[10] Google Cloud. 2025. Network Service Tiers overview | Premium Tier. https://perma.cc/XLE9-JLWZ

[11] Tianxiang Dai, Haya Shulman, and Michael Waidner. 2021. Let's Downgrade Let's Encrypt. In *Proceedings of the ACM Conference on Computer and Communications Security (CCS)*. doi:10.1145/3460120.3484815

[12] Jens Frieß, Haya Schulmann, and Michael Waidner. 2025. ValidaTor: Domain Validation over Tor. In *Proceedings of the USENIX Symposium on Networked Systems Design and Implementation (NSDI)*.

[13] Jens Frieß, Haya Schulmann, and Michael Waidner. 2024. Crowdsourced Distributed Domain Validation. In *Proceedings of the ACM Workshop on Hot Topics in Networks (HotNets)*. doi:10.1145/3696348.3696869

[14] Y. Gilad, S. Goldberg, K. Sriram, J. Snijders, and B. Maddison. 2022. *The Use of maxLength in the Resource Public Key Infrastructure (RPKI)*. RFC 9319. IETF. doi:10.17487/rfc9319

[15] Yuchen Jin, Colin Scott, Amogh Dhamdhere, Vasileios Giotsas, Arvind Krishnamurthy, and Scott Shenker. 2019. Stable and Practical AS Relationship Inference with ProbLink. In *Proceedings of the USENIX Symposium on Networked Systems Design and Implementation (NSDI)*.

[16] Joshua Juen, Aaron Johnson, Anupam Das, Nikita Borisov, and Matthew Caesar. 2015. Defending Tor from Network Adversaries: A Case Study of Network Path Prediction. *Proc. Priv. Enhancing Technol.* 2015, 2 (2015), 171–187. doi:10.1515/POPETS-2015-0021

[17] Peter Kacherginsky. 2022. Celer Bridge incident analysis. https://perma.cc/4KN3-3Y5B

[18] Dina Kozlov and Gabbi Fisher. 2019. Securing Certificate Issuance using Multipath Domain Control Validation. https://perma.cc/S36F-96EQ

[19] MPIC Labs. 2025. MarcoPolo. https://github.com/mpiclabs/MarcoPolo

[20] MPIC Labs. 2025. MPIC Labs. https://mpiclabs.org

[21] NIST. 2025. RPKI Monitor. https://rpki-monitor.antd.nist.gov/.

[22] The Open MPIC Project. 2025. aws-lambda-python/config.example.yaml at main · open-mpic/aws-lambda-python. https://github.com/open-mpic/aws-lambda-python/blob/main/config.example.yaml

[23] Sectigo. 2025. crt.sh | cert-populations. https://crt.sh/cert-populations.

A Resilience Computation Details

The results produced by the MarcoPolo framework in the form of the perspectives' routing behavior under a BGP hijack are converted to a resilience metric in the postprocessing step.

A.1 Terminology

We define the following terms:

- $\mathcal{P}$: set of all possible perspectives
- N: set of BGP nodes numbered 1 to $|N|$ acting as adversaries and victims
- v: victim node
- a: adversary node

A.2 Resilience Calculation

To calculate resilience, we start with the MarcoPolo results represented by the relation hijacked(P, v, a), which is the number of perspectives in the perspective set $P \subseteq \mathcal{P}$ for which an adversary node $a \in N$ can hijack traffic destined for a victim node $v \in N$ by announcing a conflicting prefix (using AS path prepending for RPKI-protected prefixes). We define the auxiliary function $\sigma(P, q, v, a)$ for evaluating whether the victim node v is protected against a BGP hijack launched by an adversary node a for a specific quorum q, i.e., hijack fails to convince at least q perspectives, as follows:

$$\sigma(P, q, v, a) = \begin{cases} 1 & \text{if hijacked}(P, v, a) < q \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

This leads to the following definition of a per-victim resilience score, indicating the likelihood of a successful defense against a BGP hijack by any adversary BGP node.

$$R_{\text{victim}}(P, q, v) = \sum_{a=1, a \neq v}^{|N|} \sigma(P, q, v, a) / (|N| - 1) \tag{2}$$

To evaluate the effectiveness of a given perspective deployment and quorum, we calculate both the average and median resilience as follows:

$$R_{\text{avg}}(P, q) = \sum_{v=1}^{|N|} R_{\text{victim}}(P, q, v) / |N| \tag{3}$$

To calculate the median resilience, we first define the permutation ord : $\{1, ..., |N|\} \to \{1, ..., |N|\}$ that sorts the resilience values in ascending order:

$$\langle R_{\text{victim}}(P, q, \text{ord}(1)), ..., R_{\text{victim}}(P, q, \text{ord}(|N|)) \rangle \tag{4}$$

We then calculate the median resilience as:

$$R_{\text{med}}(P, q) = \begin{cases} R_{\text{victim}}(P, q, m) & \text{if } |N| \text{ is odd} \\ \frac{R_{\text{victim}}(P, q, l) + R_{\text{victim}}(P, q, l+1)}{2} & \text{otherwise} \end{cases} \tag{5}$$

, where $m = \text{ord}((|N| + 1)/2)$ and $l = \text{ord}(|N|/2)$.

Finally, the optimization problem consists of finding the optimal MPIC deployments P_{opt} consisting of k perspectives for quorum q with the highest median resilience, while using the average resilience as the tie breaker in case multiple deployments have the same highest median resilience:

$$P_{\text{med}} = \underset{P \subseteq \mathcal{P} \wedge |P| = k}{\arg\max} R_{\text{med}}(P, q) \tag{6}$$

$$P_{\text{opt}} = \underset{P \subseteq P_{\text{med}}}{\arg\max} R_{\text{avg}}(P, q) \tag{7}$$

Table 1: Most frequently observed RIR clustering for the (at most 150) best-performing MPIC deployments with 6 remote perspectives and an N-2 quorum. If a primary perspective exists, it is marked by an asterisk.

Provider	Top RIR Clusters	Frequency
Azure	(3,2,1,0,0)	80%
	(3,3,1*,0,0)	64%
AWS	(3,3,0,0,0)	91%
	(3,3,1*,0,0)	89%
GCP	(3,3,0,0,0)	100%
	(3,3,1*,0,0)	71%

B RIR Clustering

We describe the RIR clusters of an MPIC deployment as a tuple of the number of perspectives in each RIR sorted in descending order, e.g., (3, 2, 1, 0, 0) indicates that the MPIC deployment has 6 perspectives located in three RIRs containing 3, 2, and 1 perspectives, respectively. To analyze trends in RIR clusters, we look at the MPIC deployments with high resilience. Concretely, for each cloud provider, perspective set size, and quorum policy, we consider the top 150 MPIC deployments based on their resilience values. Note that if no clear cutoff after the top 150 deployments exists, we choose the closest cutoff value such that at most 150 deployments are considered. Table 1 shows the most common RIR clusters for these top 150 MPIC deployments with 6 remote perspectives and an N-2 quorum for different cloud providers. For each cloud provider, the first and second line are the optimal deployments without and with a primary perspective, respectively.

The most common cluster behavior was 2 RIR clusters, each with 3 perspectives per RIR. In AWS, this was observed in 91% of these top deployments and for GCP this was 100%. However, this was only observed 2% of the time in Azure, whereas clusters of 3, 2, and 1 perspectives were more common at 80% of the time. An explanation of this phenomena may be that Azure perspectives in the RIRs with 2 and 1 perspectives exhibited similar routing behavior and thus behave like a single RIR. When adding a primary perspective, which must succeed in order to allow certificate issuance, it is optimal to deploy it in an separate RIR different from the other remote perspectives, as seen across all cloud providers.

C Additional MarcoPolo Evaluation Results

Table 2 shows the resilience of various MPIC deployments.

D Experiment Cost Summary

Table 3 provides a total cost analysis of the experiment by cloud provider. Notably, the Open MPIC project was deployed on AWS Lambda, with all requests covered by the AWS Free Tier. Therefore, only the API Gateway calls contributed to the cost. For the other cloud providers, we selected the cheapest VM instantiations and/or server plans. These were B1s, e2-micro, and vc2-1c-1gb for Azure, GCP, and Vultr, respectively. The estimated total cost of the experiment was $732.49.

Table 2: Median and average resilience values of various MPIC deployments without any RPKI deployment: the best-performing Azure, AWS, and GCP cloud-based MPIC deployment for different perspective sizes and quorums, Let's Encrypt, and Cloudflare.

Config	Deployment	Primary?	Resilience	
			Median	Average
(1, N)	Azure	✗	52	50
	AWS	✗	53	50
	GCP	✗	50	50
(4, N-1)	Let's Encrypt	✓	82	76
(5, N-1)	Azure	✗	100	77
		✓	100	83
	AWS	✗	97	80
		✓	100	87
	GCP	✗	89	65
		✓	92	68
(6, N-2)	Azure	✗	97	71
		✓	100	82
	AWS	✗	87	72
		✓	97	85
	GCP	✗	87	65
		✓	90	67
(8, N)	Cloudflare	✗	97	84

Table 3: Total cost of experiment by cloud provider.

Cloud Provider	Node Count	Total Cost
AWS	27	$0.01
Azure	39	$366.80
GCP	40	$215.04
Vultr	32	$150.64

E Node List

Table 4 shows a list of all the nodes deployed across the 4 different cloud providers.

Table 4: List of all cloud provider nodes included in the experiment.

Cloud Provider	Node List
Amazon Web Services	af-south-1, ap-east-1, ap-northeast-1, ap-northeast-2, ap-northeast-3, ap-south-1, ap-south-2, ap-southeast-1, ap-southeast-2, ap-southeast-3, ap-southeast-4, ca-central-1, ca-west-1, eu-central-1, eu-central-2, eu-north-1, eu-south-2, eu-west-1, eu-west-2, eu-west-3, il-central-1, me-central-1, sa-east-1, us-east-1, us-east-2, us-west-1, us-west-2
Google Cloud Platform	africa-south1, asia-east1, asia-east2, asia-northeast1, asia-northeast2, asia-northeast3, asia-south1, asia-south2, asia-southeast1, asia-southeast2, australia-southeast1, australia-southeast2, europe-central2, europe-north1, europe-southwest1, europe-west1, europe-west10, europe-west12, europe-west2, europe-west3, europe-west4, europe-west6, europe-west8, europe-west9, me-central1, me-west1, northamerica-northeast1, northamerica-northeast2, northamerica-south1, southamerica-east1, southamerica-west1, us-central1, us-east1, us-east4, us-east5, us-south1, us-west1, us-west2, us-west3, us-west4
Microsoft Azure	asia-east, asia-southeast, australia-central, australia-east, australia-southeast, brazil-south, canada-central, europe-north, europe-west, france-central, germany-westcentral, india-central, india-south, indonesia-central, israel-central, italy-north, japan-east, japan-west, korea-central, mexico-central, newzealand-north, norway-east, poland-central, southafrica-north, spain-central, sweden-central, switzerland-north, uae-north, uk-south, uk-west, us-central, us-east, us-east2, us-northcentral, us-southcentral, us-west, us-west2, us-west3, us-westcentral
Vultr	Amsterdam, Atlanta, Bangalore, Chicago, Dallas, Delhi NCR, Frankfurt, Honolulu, Johannesburg, London, Los Angeles, Madrid, Manchester, Melbourne, Mexico City, Miami, Mumbai, New Jersey, Osaka, Paris, Santiago, São Paulo, Seattle, Seoul, Silicon Valley, Singapore, Stockholm, Sydney, Tel Aviv, Tokyo, Toronto, Warsaw