Challenges of Working With Geofeeds

Authors: Oliver Gasser, William Leung, Maxime Mouchet
Affiliation: IPinfo
Emails: oliver@ipinfo.io, william@ipinfo.io, max@ipinfo.io

Introduction

IP address geolocation has many different applications in today's Internet, ranging from website language selection, ad targeting, fraud detection, to content geofencing. Recently, geofeeds have been proposed as a way for network operators to publish geolocation information about their own IP prefixes. Geofeeds are CSV files which contain geolocation information about IP prefixes, containing fields for country, region, and city information. Geofeeds can be shared through dedicated entries in the WHOIS database or privately with geolocation service providers. Generally, geofeeds have had a positive impact on geolocation sharing from network operators with geolocation providers by providing a standard interface for exchanging data. There remain, however, a number of challenges when working with geofeed data today, which we discuss in this paper.

Ambiguous locations, typos, mistakes

RFC 8805 states that geofeeds country and region fields must conform to ISO 3166-1 and 3166-2 codes, and that the city field is free form. However, geolocation providers often need to map those locations to their own internal database in order to provide geographical coordinates or well-known identifiers like GeoNames to their customers. The process of mapping these textual locations to geographical coordinates can result in incorrect and inconsistent locations across providers due to various mistakes in geofeeds, such as non-ISO region codes, swapped fields, or typos. In addition, free form city names are difficult to match due to variations in city names across languages (e.g. Köln / Cologne).

Entry	**Issues**
196.48.162.0/24,QA,Baladiyat ad DawÃƒÆ'Ã‚Â¡Ãƒâ€šÃ‚Ãƒâ€šÃ‚Â©ah,Doha,	Full region name instead of ISO code. Encoding issues.
66.118.40.0/24,AL-11,AL,Tirana,	Country and region fields are swapped.
104.28.34.49/32,GA,GA-01,Libreville,	Invalid ISO region code (should be GA-1).
202.50.55.19/32,Norway,NO-30,Snaroeya,1367	Non ISO country code.
2a12:bec4:12a5:137a::/6,FR,FR-84,Lyon,	Typo in prefix size, should be /64 instead of /6.

Possible solution: To mitigate this issue we suggest that geofeed include a well-known geographical identifier instead or in addition to textual addresses. For example GeoNames would allow to represent location non-ambiguously at various granularities (country, region, city, etc.). Alternative geospatial

indexes like H3s could also be considered. GeoNames or H3s could be added as an additional optional field in geofeed CSV files.

Adversarial locations

Some networks might have an interest in appearing in locations where they do not have a physical presence. For example to bypass geographical restrictions, or for a VPN provider to claim more locations than it actually has (see "How to catch when proxies lie: Verifying the physical locations of network proxies with active geolocation").

Possible solution: IPinfo uses RTT measurements to validate and discard geofeed entries. However this is not sufficient as not all IP addresses are pingable, and there might be legitimate reasons for RTT conflicts, such as CGNAT or relay service like the iCloud Private Relay.
An alternative solution could be for network operators to offer looking-glass-like servers enabling measurements behind the gateway.

False granularity claims

Geofeeds allow producers to provide information on different granularities: country-level, region-level, city-level, each in a dedicated field within geofeed CSV files. While each of these three fields is optional (see RFC 8805), we find some geofeed providers fill up all of the fields, while the location is clearly not as fine-granular. One such example is the satellite provider Starlink, who in its geofeed lists the capital city of each country as the city-level location. Due to the distributed nature of Starlink users, it is clear that not all of their users are located in capital cities. These entries might lead to false granularity claims by geofeed consumers who are not aware of these potential issues.

Possible solution: Increase clarity in geofeed standard that not all fields in the CSV need to be populated and they should only be populated if the information in there is actually accurate.

User vs. infrastructure location

In today's Internet, IP geolocation can have different meanings depending on the type of reported location. With services like iCloud Private Relay, satellite networks, VPNs, proxy services, and mobile roaming the location of the infrastructure IP address (i.e., the public-facing IP address visible to service providers) might differ substantially from the user's actual location. For example, a user physically located in New Zealand using a roaming connection on their mobile phone might be routed to France with the exit IP address being located in Paris. While the publicly visible datacenter IP address might be physically located in France, the user is tens of thousands of kilometers away. This discrepancy in user vs. infrastructure location makes it challenging for geolocation providers to determine the veracity of geofeed information.

Possible solution: The geofeed standard could be extended in a way to allow producers of geofeed files to signal whether specific locations are meant as user location or infrastructure location. To guarantee backward compatibility an additional optional field could be added to the geofeed CSV files. In addition, there is ongoing work within the IETF to signal the presence of CGNATs, which can be used by geofeed producers as well.

Stale data

Geofeeds do not state what, if any, validity period they are intended for, which leads to consumers not knowing if the feed is still current. This can apply at the level of the feed, or per entry, as some providers may wish to convey different update periods / confidences about the location. The geofeed standard mentions to rely on existing HTTP caching headers, but this information is not available directly within the geofeed, but rather out-of-band.

Possible solution: Provide standardized fields or metadata for created or updated timestamps/TTLs in embedded geofeeds directly.

Lack of feedback mechanism

When issues with a published geofeed such as incorrect locations, typos, etc. are discovered, there is no straightforward way for consumers to signal these issues to providers. In these instances, geofeed consumers need to either ignore those geofeed entries entirely, try to determine the intended entry, or send out emails to WHOIS contact email addresses to get the geofeed fixed.

Possible solution: Provide a dedicated contact email address or Web form to report identified issues in a geofeed. This could be implemented as an optional comment at the top or bottom of a geofeed file.

Geofeeds obtained outside WHOIS

There are geofeeds obtained outside WHOIS, whether for various reasons of the (perceived) difficulty/cost of providing a geofeed, unawareness of the process for submitting/updating from WHOIS, or that the provider's use case does not fit into the WHOIS format. Such geofeeds should still be available, while also conforming to or approaching the standards set by various RFCs.

Possible solution: Provide a validation service/code for geofeeds to enable a lower-cost process.

Conclusion

Although geofeeds have had a positive impact on geolocation sharing from network operators to geolocation service providers, a number of open challenges remain. In this paper we discuss ambiguous locations, typos, mistakes, false granularity claims, user vs. infrastructure location, adversarial locations, stale data, lack of feedback mechanism, and geofeeds not present in the WHOIS. For each of these challenges we propose actionable solutions to further improve the current state of geolocation in the Internet.