



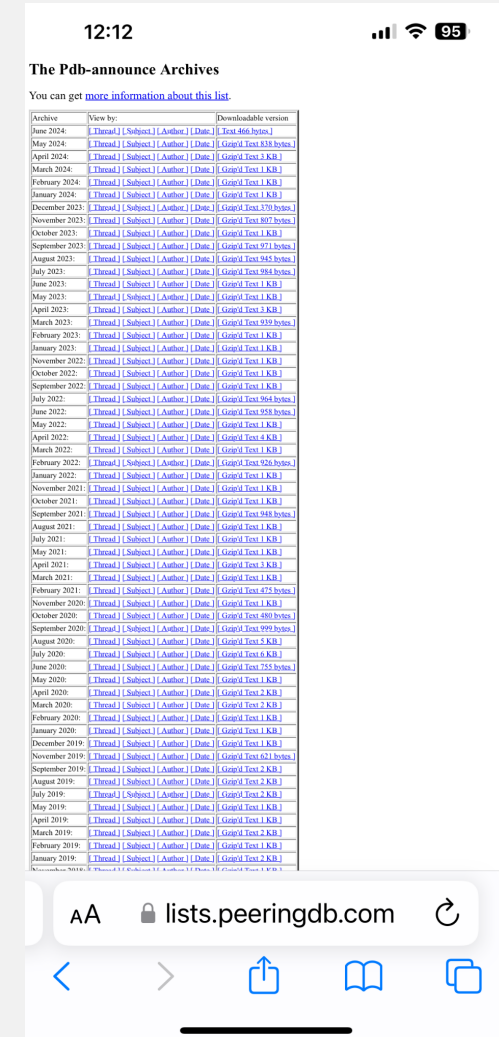
PeeringDB

PeeringDB Update

Paul Hoogsteder
paulh@peeringdb.com

2024 Strategic Review

- Stewards reviewed the strategy and proposed updates
- Announcement of proposed changes to follow, so subscribe to the pdb-announce list if you care and aren't there yet



High Priority Strategic Tasks for 2024/2025

- Write Operational Service Level Policy for services provided to the community and regularly report compliance
- Regular third-party security audit
- Conduct redundancy and restoration test every two years

Product ~~Roadmap~~ Periodic Review

- The Product Committee votes on product features
 - This means a long term roadmap is a poor choice
 - So instead of looking a year into the future, we'll be reporting on the last six months of changes each April and October



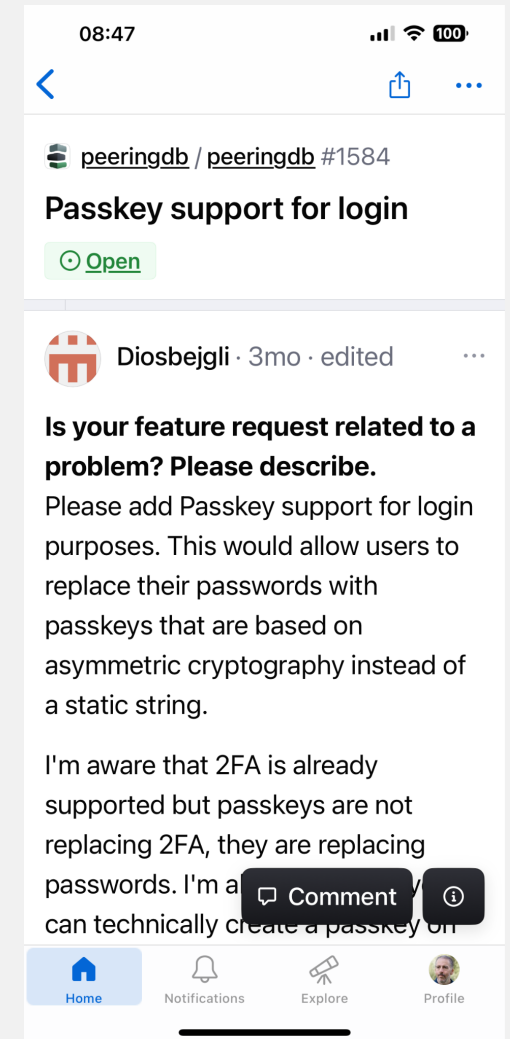
[Image by freepik](#)

Product & Operations Focus

- Continuing with web UI improvements. Check preview.beta.peeringdb.com and provide feedback
- Normalizing the presentation of place names, e.g. Always show Göteborg as Gothenburg to help people doing analytical work
- Looking at significant security improvements
 - Mandatory 2FA ([#1634](#))
 - Require dormant accounts to revalidate before being used again ([#1636](#))
 - Create a “read only” mode to provide availability when defending against an attack ([#1635](#))

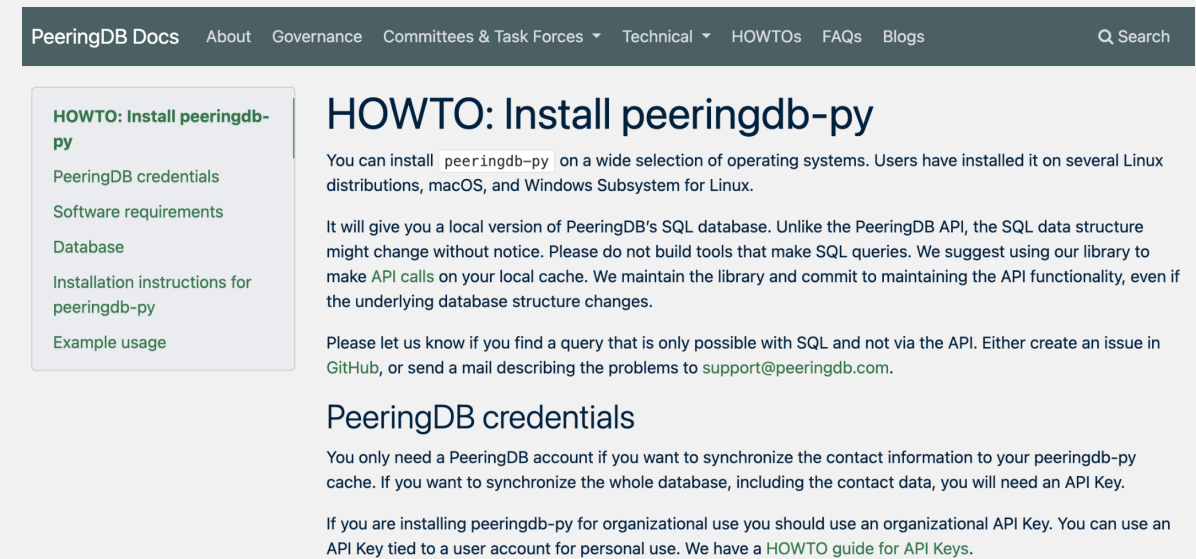
Engage with us!

- If you want a feature or found a bug you can:
 - Discuss it with anyone on the Product Committee
 - Discuss it with us and other users on the [user-discuss list](#)
 - Ask for help at support@peeringdb.com
 - Create an issue on Github in
 - [peeringdb/peeringdb](#)
 - [peeringdb/peeringd-py](#)
 - [peeringdb/django-peeringdb](#)



peeringdb-py

- Get a local cache of Peeringdb data
- Sync every hour
- Keep queries local for faster access
- Integrate with other datasets
- No query limits



The screenshot shows the PeeringDB Docs website. The navigation bar includes links for 'PeeringDB Docs', 'About', 'Governance', 'Committees & Task Forces', 'Technical', 'HOWTOs', 'FAQs', and 'Blogs', along with a search icon. A sidebar on the left lists navigation options: 'HOWTO: Install peeringdb-py' (highlighted), 'PeeringDB credentials', 'Software requirements', 'Database', 'Installation instructions for peeringdb-py', and 'Example usage'. The main content area features the title 'HOWTO: Install peeringdb-py' and the following text: 'You can install `peeringdb-py` on a wide selection of operating systems. Users have installed it on several Linux distributions, macOS, and Windows Subsystem for Linux. It will give you a local version of PeeringDB's SQL database. Unlike the PeeringDB API, the SQL data structure might change without notice. Please do not build tools that make SQL queries. We suggest using our library to make API calls on your local cache. We maintain the library and commit to maintaining the API functionality, even if the underlying database structure changes. Please let us know if you find a query that is only possible with SQL and not via the API. Either create an issue in GitHub, or send a mail describing the problems to support@peeringdb.com. PeeringDB credentials You only need a PeeringDB account if you want to synchronize the contact information to your peeringdb-py cache. If you want to synchronize the whole database, including the contact data, you will need an API Key. If you are installing peeringdb-py for organizational use you should use an organizational API Key. You can use an API Key tied to a user account for personal use. We have a [HOWTO guide for API Keys](#).'

Thank you to our sponsors!

Diamond Sponsor



Microsoft

Platinum Sponsors



Meta

YAHOO!

Gold Sponsors



Silver Sponsors

