Beginning with Version 2.3.20 of the Videstra system we have switched from using Bing™ maps to maps provided by Mapbox™. Bing maps are scheduled to be shut down in June of 2025 and replaced by a new mapping service from Microsoft called Azure Maps™. Unfortunately, Azure Maps will not have a free tier for usage. Based on information available at the time this documentation is being written, it could prove to be very costly for our users.

Videstra has made the decision to switch to Mapbox for the following reasons:

1. Mapbox offers a free tier for *commercial* use.
2. Our use is very simple and is unlikely to go beyond the limits imposed by the free tier of Mapbox service.
3. The look, feel, and performance of the Mapbox maps for use in our application is much better than similar offerings from other mapping vendors.

While Mapbox provides a free tier, it is important to set up a paid account and a payment method (typically a credit card) in case your usage goes beyond the free tier. We have evaluated our system and feel there is very little likelihood of Videstra using more data than is allowed in the free tier, however it is recommended you still set up the paid account. Mapbox may, in the future, change the free tier levels.

> Note: Videstra assumes no liability for using map services from Mapbox (or any mapping service) that incurs charges. Payment for such services is entirely the responsibility of the end user.
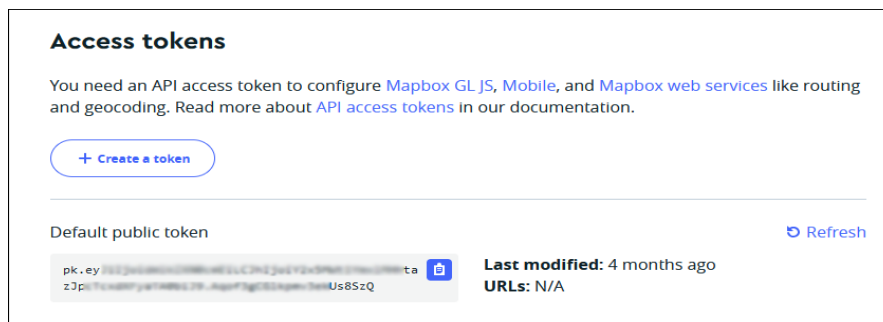
Creating Your Mapbox™ Token

To get started you must set up your Mapbox account by going to the following web page:

https://account.mapbox.com/auth/signup/

Once you have set up your account you may now create your API Key which will look something like this:

pk.eyJ1IjoidmlkZXN0cmEiLCJhIjashDOntLOokiNHerEJJFJadjjiSIDjsiaoodnLKHLKhjdkjlan-n41hrZ085qXgg

The initial key that is created when you set up the account is also known as your "Default Public Key" and is not the one you will want to use.

Access tokens

You need an API access token to configure Mapbox GL JS, Mobile, and Mapbox web services like routing and geocoding. Read more about API access tokens in our documentation.

[+ Create a token]

Default public token

pk.ey...ta
zJp...Us8SzQ

Last modified: 4 months ago
URLs: N/A

Instead, you will set up a private key by clicking on **+ Create a token**

This page has a lot of options – but the only thing you will need to do is enter a friendly name for the token. Something like "VidestraMaps" would be good – but it can be anything you like.

Do not check any of the options below "Server scopes" – this is only for setting up a *secret api key* with capabilities your Videstra system does not use or need.

Optional URL Restrictions

Near the bottom of the page, you will see an option to set URL restrictions. This is not strictly necessary, and we do not recommend you set this. Our use of the Mapbox API is limited to read-only map tiles and this level of security is likely unnecessary.

If still desired then you can restrict usage of your key to applications that make their request only from a properly configures DDNS (from a DDNS Service such as NOIP.com (https://noip.com)

The DDNS name you pick (e.g. mystation.ddns.net) will resolve to your IP Address, even if your IP Address changes from time to time.

> Note: If your IP address does change there can be a small delay before the DDNS service is updated and propagates the new IP address to all available DNS servers. This can possibly take several minutes. If your IT department has set up your gateway to use lesser-known DNS servers (or private DNS servers) this delay could be significantly longer.
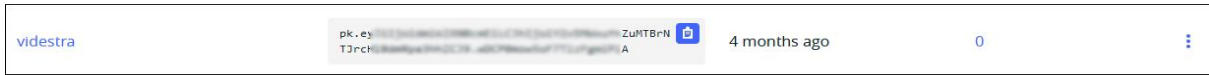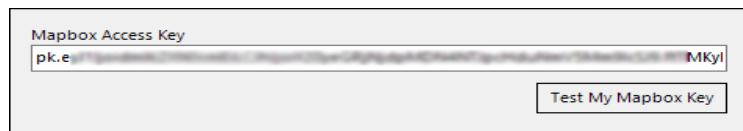
Do not put http or https in front of the DDNS.

Once you are ready you can create the new token.

Copy the new token into your Videstra V-Manager under **File → Settings → Mapbox Maps → Mapbox Access Key**

You can test the API key to make sure it works. Be aware that a newly created API Key may take several minutes before it can be used.

Please note that we have tested using a DDNS with the Mapbox API and have found that in some cases it may not work. This is most likely due to some missing data in the request header. There is currently no known workaround for this. If that is the case for you then you will not be able to put in the URL restriction.