plotly

Star 23,446

*Dash Python* > ***Dash Enterprise Auth***

# Dash Enterprise Auth

> This documentation is for **Dash Enterprise**. Dash Enterprise is the fastest way to write & deploy Dash apps and Jupyter notebooks. 10% of the Fortune 500 uses Dash Enterprise to productionize AI and data science apps. **Find out if your company is using Dash Enterprise**

Dash Enterprise automatically implements user authentication if your **Dash app's viewer access level** is set to *Restricted* or *Authenticated*, but not if it is set to *Unauthenticated*. You can access the authentication data within your app using the `dash-enterprise-auth` package.

## Using `dash-enterprise-auth` in an Existing Dash App

Dash Enterprise >=5.2

Dash Enterprise <5.2

If you have previously deployed your Dash app to Dash Enterprise, add `dash-enterprise-libraries` to your `requirements.txt` file to get started.

> **Note**: `dash-enterprise-auth` is available from **PyPI.org**, unlike the other dependencies of `dash-enterprise-libraries` which are installed from the Dash Enterprise host. In order for your `dash-enterprise-auth`-enabled app to successfully build, Dash Enterprise needs network access to PyPI.org or a private Python package index containing `dash-enterprise-auth`. **Learn more about using a private Python index**.

> **Known issue**: If using `dash-enterprise-auth` with a Dash Enterprise instance <=5.4.0 that uses a certificate whose root CA is internal to your organization, you may see an error similar to `certificate verify failed: self-signed certificate in certificate chain`. To resolve this error, **set an app environment variable** with name `REQUESTS_CA_BUNDLE` and value `/etc/ssl/certs/ca-certificates.crt`. This issue is fixed in Dash Enterprise 5.5.0.

> Dash Enterprise 5 requires `dash-enterprise-auth` 0.0.6 or later.

With `dash-enterprise-auth`, you can use the `get_username` and `get_user_data` methods to get information about the app viewer. These methods must be called from within callbacks.

`dash-enterprise-auth` also includes the `create_logout_button` method, which allows you to add a logout button to your app's layout. This button uses a special URL to log the user out. To use `create_logout_button` when developing locally, you need to set an environment variable called `DASH_LOGOUT_URL`. You can do this by running your code with `DASH_LOGOUT_URL=https://auth-<your-dash-enterprise-server>/auth/realms/dash/protocol/openid-connect/logout?client_id=dash-backend python app.py`. If you're developing in Workspaces, `DASH_LOGOUT_URL` is supplied automatically.

The example below demonstrates how to use `get_username`, `get_user_data`, and `create_logout_button`.

```
from dash import Dash, dcc, html, Input, Output, callback
import dash_enterprise_auth as auth
```

```
external_stylesheets = ['https://codepen.io/chriddyp/pen/bWLwgP.css']

app = Dash(__name__, external_stylesheets=external_stylesheets)

server = app.server  # Expose the server variable for deployments


# Standard Dash app code below
app.layout = html.Div(className='container', children=[

    html.Div([
        html.H2('Sample App', id='header-title', className='ten columns'),
        html.Div(auth.create_logout_button(), className='two columns', style={'marginTop': 30}
    ]),
    html.Div(id='dummy-input', style={'display': 'none'}),

    html.Div([
        html.Div(
            className='four columns',
            children=[
                dcc.Dropdown(['LA', 'NYC', 'MTL'], 'LA', id='dropdown')
        ]),
        html.Div(
            className='eight columns',
            children=[
                dcc.Graph(id='graph')
            ])
    ])
])


@callback(Output('header-title','children'),
                Input('dummy-input', 'children'))
def update_title(_):

    # print user data to the logs
    print(auth.get_user_data())

    # update header with username
    return 'Hello {}'.format(auth.get_username())


@callback(Output('graph', 'figure'),
                Input('dropdown', 'value'))
def update_graph(value):
    return {
        'data': [{
            'x': [1, 2, 3, 4, 5, 6]
```

*Dash Python* ***> Dash Enterprise Auth***