## Helmet

Helmet can help protect your app from some well-known web vulnerabilities by setting HTTP headers appropriately. Generally, Helmet is just a collection of smaller middleware functions that set security-related HTTP headers (read more).

> info **Hint** Note that applying `helmet` as global or registering it must come before other calls to `app.use()` or setup functions that may call `app.use()`. This is due to the way the underlying platform (i.e., Express or Fastify) works, where the order that middleware/routes are defined matters. If you use middleware like `helmet` or `cors` after you define a route, then that middleware will not apply to that route, it will only apply to routes defined after the middleware.

### Use with Express (default)

Start by installing the required package.

```
$ npm i --save helmet
```

Once the installation is complete, apply it as a global middleware.

```
import helmet from 'helmet';
// somewhere in your initialization file
app.use(helmet());
```

> warning **Warning** When using `helmet`, `@apollo/server` (4.x), and the Apollo Sandbox, there may be a problem with CSP on the Apollo Sandbox. To solve this issue configure the CSP as shown below:

```
app.use(helmet({
  crossOriginEmbedderPolicy: false,
  contentSecurityPolicy: {
    directives: {
      imgSrc: [`'self'`, 'data:', 'apollo-server-landing-
page.cdn.apollographql.com'],
      scriptSrc: [`'self'`, `https: 'unsafe-inline'`],
      manifestSrc: [`'self'`, 'apollo-server-landing-
page.cdn.apollographql.com'],
      frameSrc: [`'self'`, 'sandbox.embed.apollographql.com'],
    },
  },
}));
```

### Use with Fastify

If you are using the `FastifyAdapter`, install the @fastify/helmet package:

```
$ npm i --save @fastify/helmet
```

fastify-helmet should not be used as a middleware, but as a Fastify plugin, i.e., by using `app.register()`:

```
import helmet from '@fastify/helmet'
// somewhere in your initialization file
await app.register(helmet)
```

warning **Warning** When using `apollo-server-fastify` and `@fastify/helmet`, there may be a problem with CSP on the GraphQL playground, to solve this collision, configure the CSP as shown below:

```
await app.register(fastifyHelmet, {
   contentSecurityPolicy: {
     directives: {
       defaultSrc: [`'self'`, 'unpkg.com'],
       styleSrc: [
         `'self'`,
         `'unsafe-inline'`,
         'cdn.jsdelivr.net',
         'fonts.googleapis.com',
         'unpkg.com',
       ],
       fontSrc: [`'self'`, 'fonts.gstatic.com', 'data:'],
       imgSrc: [`'self'`, 'data:', 'cdn.jsdelivr.net'],
       scriptSrc: [
         `'self'`,
         `https: 'unsafe-inline'`,
         `cdn.jsdelivr.net`,
         `'unsafe-eval'`,
       ],
     },
   },
 });

// If you are not going to use CSP at all, you can use this:
await app.register(fastifyHelmet, {
  contentSecurityPolicy: false,
});
```