

How do I register a custom Vue component for my module

To register a Vue component for our module, we will need to create a JavaScript file for our components and in order to initialize Vue.

Link assets

In order to link all the assets, which is a necessary step of registering a Vue component, we need to add a function to the module **Service Provider**. This is the code we need in the service provider:

```
public function assetLinks()
{
    return [
        'vendor/module-name' => __DIR__ . '/../public',
    ];
}
```

Bear in mind this is the exact path we have to supply when loading components into a view.

Initialize components

```
import NewComponent from './components/NewComponent'

window.newModule = {
    install(Vue) {
        Vue.component('new-component', NewComponent)
    },
}
```

The above code gives us access to the `<new-component></new-component>` tag, provided we've loaded the components into a view.

Loading components into a view

```
@push('scripts')
  <script src="{{ asset(mix(new-module.js', 'modules/aerocargo/new-module')) }}"></script>
  <script>
    window.AeroAdmin.vue.use(window.newModule);
  </script>
@endpush
```

Provided all of the namespacing in the two files matches correctly, the Vue components that we have declared will now be accessible in our view.

If there is a problem with the mix of our assets, remember to run `php artisan aero:link` in the **root directory** of your Aero store.

Enabling Vue devtools

In order to enable Vue devtools for our module, we simply add the following line of code into the file where we initialize our components:

```
import NewComponent from './components/NewComponent'

window.new-module = {
  install(Vue) {
    Vue.config.devtools = true;

    Vue.component('new-component', NewComponent)
  },
}
```

Revision #1

Created 2026-09-02 13:28:30 UTC by Michael Price

Updated 2026-09-02 13:28:30 UTC by Michael Price