

在 aspnetcore 中部署 vue 项目

作者: [loogn](#)

原文链接: <https://ld246.com/article/1597314682500>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

现在开发项目一般是前后端分离的，这样做有诸多好处自不必说，现在要说一个缺点，就是一个项目往要多端部署，api一个，前端项目一个（VUE），后台管理一个（VUE）虽然可以使用nginx放在一个名下面，感觉还是不理想，现在尝试把vue部署到asp.netcore3.1的静态文件服务中去。

场景：api项目(aspnetcore)，fe前端项目使用hash路由(vue)，admin后端项目使用history路由(vue)。

发布fe到根目录/fe下面，发布admin到api项目/admin下面，然后配置api项目的静态文件服务。

```
//后台管理vue根目录
app.UseStaticFiles(new StaticFileOptions
{
    FileProvider = new PhysicalFileProvider(Path.GetFullPath("./admin")),
    RequestPath = "/admin"
});
//前台页面vue跟目录
app.UseStaticFiles(new StaticFileOptions
{
    FileProvider = new PhysicalFileProvider(Path.GetFullPath("./fe")),
    RequestPath = "/fe"
});
```

由于是生成的单页面，api项目还需要做特殊路由处理：

```
app.UseEndpoints(endpoints =>
{
    endpoints.MapControllerRoute(
        name: "default",
        pattern: "{controller=Home}/{action=Index}/{id?}");

    //处理vue的路由
    endpoints.MapFallback(async context =>
    {
        if (context.Request.Path.StartsWithSegments("/fe", StringComparison.Ordinal))
        {
            var html = await File.ReadAllTextAsync(Path.GetFullPath("./fe/index.html"));
            context.Response.ContentType = "text/html";
            await context.Response.WriteAsync(html, Encoding.UTF8);
        }
        else if (context.Request.Path.StartsWithSegments("/admin", StringComparison.Or
inal))
        {
            context.Response.Redirect("/admin/index.html");
        }
        else
        {
            context.Response.StatusCode = (int) HttpStatusCode.NotFound;
        }
    });
});
```

因为vue项目使用不同的路由方式，这里需要做不同的处理逻辑。

当然前台vue项目也需要设置两个地方，第一个是路由的base要设置成 '/fe' 和 '/admin' 。

第二个vue.config.js中设置 `publicPath:process.env.NODE_ENV==='production'?'/admin/':'/'`

这样就完成了我们的目标。