链滴

老板让我十分钟上手 nx-admin

作者：mingebaqi
原文链接：https://ld246.com/article/1531185030233
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)

大体流程

参考资料:

nx-admin项目地址

首先这里就不讲解vue和vuex之类的基础东西了 有兴趣的可以去官方文档了解。这里根据流程走向大说说

路由配置

首先找到路由配置,路由配置放在了src/router/index.js路由配置里暴露了两个常量 一个是 constantRouterMap 另外一个是 asyncRouterMap 这里先说说constantRouterMap。 nx-admin的权限验证概是

- 1 默认大家都能访问的页面,不需要权限, 都访问的页面定义为 constantRouterMap
- 2 需要登录或者需要权限的页面路由定义为 asyncRouterMap

根据后台获取到用户信息role（权限）的不同来动态加载asyncRouterMap中meta.role的权限对应页面

登录成功后做的事情

点击登录以后 左侧的侧边栏有导航列表。 这里提出两个疑问?

- 根据路由配置说的 动态加载对应的权限路由 那么侧边栏那么多路由 肯定不能写死吧?
- 我点击登录后 那些登录流程怎么走的? 用户权限存在哪里? token在哪里?

侧边栏的动态渲染

根据问题1来回答 首先我们找到layout也就是src/views/layout/Layout.vue,

因为在路由配置文件我们看见asyncRouterMap中好多组件的父组件都是Layout 在Layout中我们就以看到有个组件sidebar。

ok继续找sidebar这个组件 src/views/layout/components/Sidebar/index.vue，发现这里就是渲染边栏的,然后找到渲染的变量是permission_routers 这个变量是存在vuex里面的，也就是说路由是存vuex的 所以咋们去vuex里面找找看 src/store/modules/permission.js。

路由的动态加载

src/store/modules/permission.js 这个文件里面有个actions

```
GenerateRoutes({ commit }, data) {
  return new Promise(resolve => {
    const { roles } = data
    let accessedRouters
    if (roles.indexOf('admin') >= 0) {
      accessedRouters = asyncRouterMap
    } else {
```

```
        accessedRouters = filterAsyncRouter(asyncRouterMap, roles)
      }
      commit('SET_ROUTERS', accessedRouters)
      resolve()
    })
  }
```

发现就是这一段代码更改了permission_routers,具体逻辑咱们不看 简单解释来说就是

```
 如果用户的权限是管理员
      把asyncRouterMap所有的路由页面都渲染出来,毕竟管理员嘛 你懂的权限嘛。
   否则
      我不是管理员但是也不是游客就是一小市民 那么我要去asyncRouterMap中找找我小市民能够问哪些页面。
```

看完这段逻辑咋们就知道了这个路由是如何动态更改的了,等等,是不是忘了啥？ 虽然我知道这个action,但是。。。在哪调用的？ 经过深思熟虑的着想，在花了0.1s后 就得出，既然是路由嘛 肯定是有个全的地方要做判断的 所以得出结论就是 router.beforeEach, 一开始去找那个啥src/main.js,发现beforeEach被分离在src/permission.js 打开这个文件。一切疑问都解开了。

用户权限的获取

说真的。。这个文件好长。。都不想看了。。。。 下图的代码这么长 看个毛啊。。于是我简单翻译下

```
router.beforeEach((to, from, next) => {
  NProgress.start() // start progress bar
  if (getToken()) { // 判断是否有token
    if (to.path === '/login') {
      next({ path: '/' })
      NProgress.done() // router在hash模式下 手动改变hash 重定向回来 不会触发afterEach 暂时hak方案 ps：history模式下无问题，可删除该行！
    } else {
      if (store.getters.roles.length === 0) { // 判断当前用户是否已拉取完user_info信息
        store.dispatch('GetUserInfo').then(res => { // 拉取user_info
          const roles = res.data.role
          store.dispatch('GenerateRoutes', { roles }).then(() => { // 生成可访问的路由表
            router.addRoutes(store.getters.addRouters) // 动态添加可访问路由表
            next({ ...to, replace: true }) // hack方法 确保addRoutes已完成 ,set the replace: true so th navigation will not leave a history record
          })
        }).catch(() => {
          store.dispatch('FedLogOut').then(() => {
            Message.error('Verification failed, please login again')
            next({ path: '/login' })
          })
        })
      } else {
        // 没有动态改变权限的需求可直接next() 删除下方权限判断↓
        if (hasPermission(store.getters.roles, to.meta.role)) {
          next()//
        } else {
          next({ path: '/401', query: { noGoBack: true }})
          NProgress.done() // router在hash模式下 手动改变hash 重定向回来 不会触发afterEach 暂
```

```
hack方案 ps：history模式下无问题，可删除该行！
        }
        // 可删 ↑
      }
    }
  } else {
    if (whiteList.indexOf(to.path) !== -1) { // 在免登录白名单，直接进入
      next()
    } else {
      next('/login') // 否则全部重定向到登录页
      NProgress.done() // router在hash模式下 手动改变hash 重定向回来 不会触发afterEach 暂时ha
k方案 ps：history模式下无问题，可删除该行！
    }
  }
})
```

请说人话，翻译成人话的版本。。。

```
每次更改页面路由
    你有没有token啊?
        有啊
            好的,你的权限是默认的权限0么?
                是的。。我就是一游客
                    系统获取我的信息..拿到权限值,动态加载路由(GenerateRoutes)...通行...
                不是。。我是权限汪(admin)
                    等等..我看看作者有没有把你降级
                        没有
                            好了。。你还是权限汪 请进
                        有
                            滚吧,你已经不是权限汪了,作者已经把你写成战斗力只有5的渣渣了
        没有
            没有还敢闯这里？滚去关口(/login)
```

没错，就这么简单。整个权限验证流程就完整了。剩下的就是读读文档啊,看看如何使用组件之类的了。

参考 https://github.com/Relsoul