# 安全性

## 1 简介

Mendix 的安全性涉及两方面：希望不同人员查看应用程序的不同部分，以及防止未经授权的访问。这两方面均可从 Studio Pro 进行管理。对表单、数据和微流的访问仅限于授权用户。

Mendix 的安全性不包括对最终用户从应用程序上传或下载的文件进行病毒和恶意软件扫描。

## 2 安全级别

如果想要获得全面的安全保护，则需要在某人可以访问表单、实体和微流之前明确授予其相应的访问权限。默认情况下，任何用户都无法访问任何内容。为便于创建原型和演示，有些安全级别所需的安全设置少于生产系统所需的安全设置。

## 3 项目与模块安全性

在项目级别，可以指定一些全局设置：安全级别、管理员帐户以及是否允许匿名访问。

大多数安全设置发生在模块级别。这样做的好处是模块可独自指定安全性，并可在其他项目中进行分发和重用。可以配置对表单、实体、微流和数据集的访问权限。

## 4 用户角色与模块角色

Mendix 应用程序中的最终用户具有一个或多个用户角色。创建或编辑用户时，可从客户端中指派这些角色。用户角色位于项目级别，可在“项目安全性”中进行编辑。

每个模块各自定义一组模块角色，只需根据这些模块角色指定模块内的安全性即可。电子邮件模块可能有两个模块角色，一个角色用于普通用户，另一个角色用于管理员；其他模块可能有一个或多个模块角色，具体取决于这些模块的要求。

用户角色是模块角色的组合。登录系统的用户将获取其所有用户角色的访问权限，并间接获得这些用户角色所包含的模块角色的访问权限。

假设项目有两个模块：系统和项目管理 (PM)。PM 模块具有三个模块角色：TeamMember、TeamLeader 和 Administrator。在这个案例中，只需要两个用户角色，因为无需区分团队领导和管理员。可对这两个用户角色进行定义并指派模块角色。下表显示了用户角色中包含哪些模块角色。请注意，始终至少需要系统中的用户角色。

| 用户角色“TeamMember” | 用户角色“TeamLeader” |
| --- | --- |
| System.User | System.User |
| ProjectManagement.TeamMember | ProjectManagement.TeamLeader |
| | ProjectManagement.Administrator |

## 5 实体访问与页面访问

根据实体，可指定谁可以在什么情况下读取或写入哪些成员（属性和关联）。使用 XPath 约束，可以表示强大的安全行为；例如，“员工只能看到所属部门创建的订单”。

根据页面，可指定谁能从导航打开该页面。菜单栏经过优化，只有用户有权访问的页面才可见。

必须组合实体和页面的访问权限，因为实体也可从微流和自定义小组件访问。此外，还可通过实体访问表示更高级的安全性。