huihongxiao GitBook: [master... Latest commit 540f202 on Oct 17, 2020 History

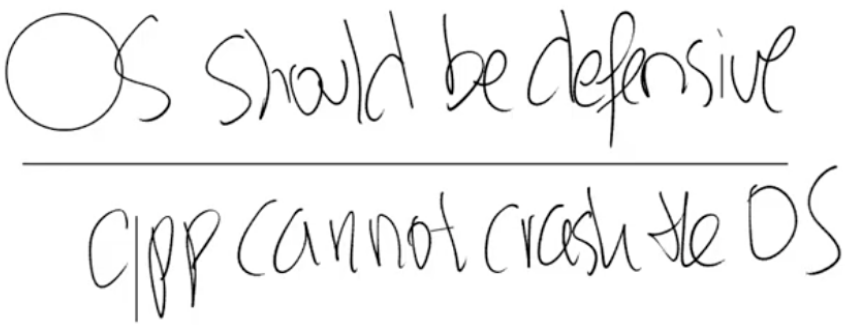
1 contributor

20 lines (10 sloc) 2.62 KB Raw Blame

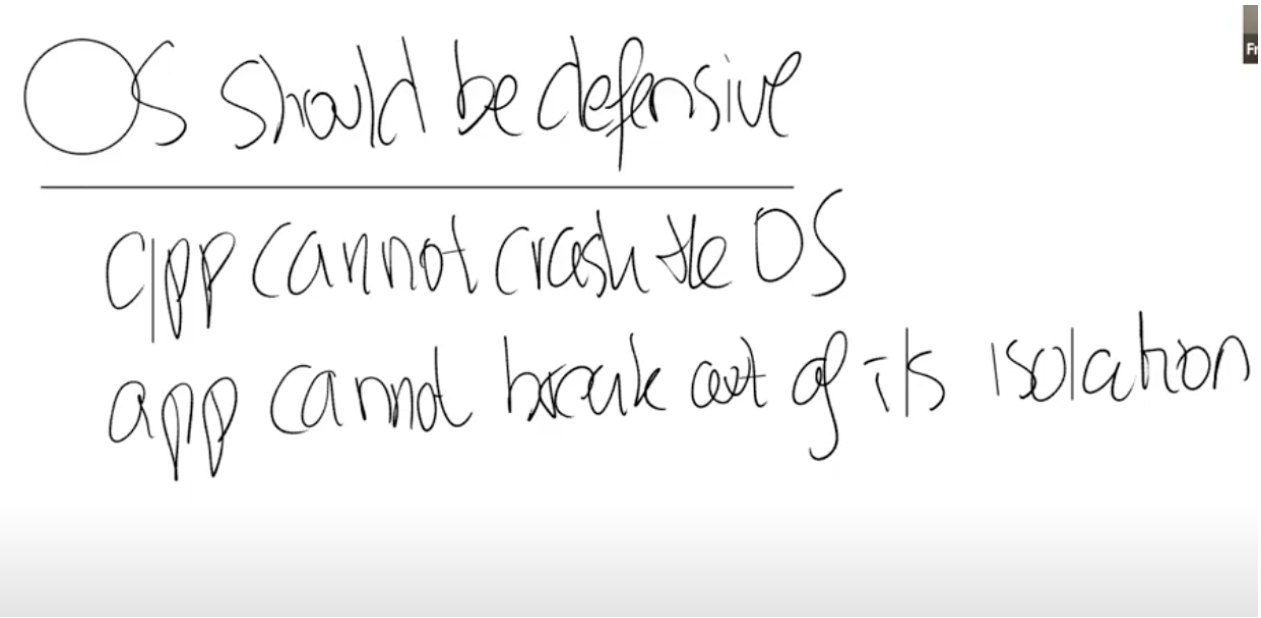
3.3 操作系统防御性（Defensive）

现在我们有一个操作系统，并且有一些应用程序正在运行。这里有一件事情需要考虑：操作系统应该具有防御性（Defensive）。

当你在做内核开发时，这是一种你需要熟悉的重要思想。操作系统需要确保所有的组件都能工作，所以它需要做好准备抵御来自应用程序的攻击。如果说应用程序无意或者恶意的向系统调用传入一些错误的参数就会导致操作系统崩溃，那就太糟糕了。在这种场景下，操作系统因为崩溃了会拒绝为其他所有的应用程序提供服务。所以操作系统需要以这样一种方式来完成：操作系统需要能够应对恶意的应用程序。

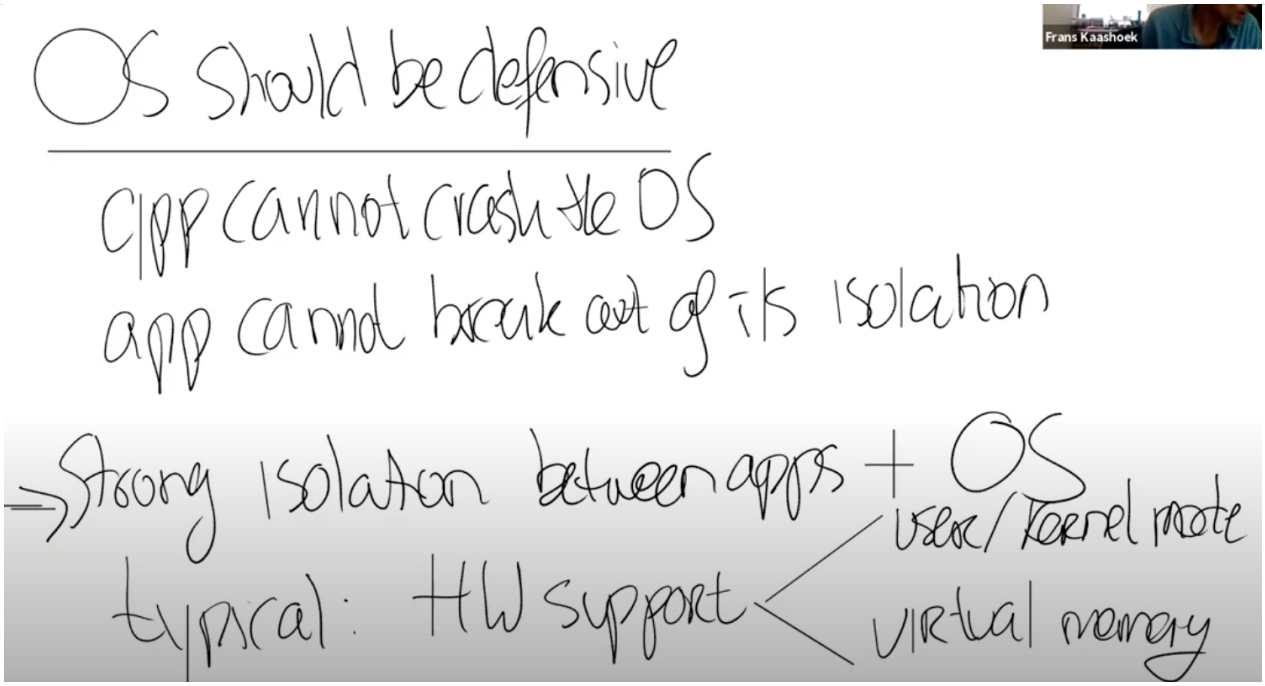


另一个需要考虑的是，应用程序不能够打破对它的隔离。应用程序非常有可能是恶意的，它或许是由攻击者写出来的，攻击者或许想要打破对应用程序的隔离，进而控制内核。一旦有了对于内核的控制能力，你可以做任何事情，因为内核控制了所有的硬件资源。



所以操作系统或者说内核需要具备防御性来避免类似的事情发生。实际中，要满足这些要求还有点棘手。在Linux中，时不时的有一些内核的bug使得应用程序可以打破它的隔离域并控制内核。这里需要持续的关注，并尽可能的提供最好的防御性。当你在开发内核时，防御性是你必须掌握的一个思想。实际中的应用程序或许就是恶意的，这意味着我们需要在应用程序和操作系统之间提供强隔离性。如果操作系统需要具备防御性，那么在应用程序和操作系统之间需要有一堵厚墙，并且操作系统可以在这堵墙上执行任何它想执行的策略。

通常来说，需要通过硬件来实现这的强隔离性。我们这节课会简单介绍一些硬件隔离的内容，但是在后续的课程我们会介绍的更加详细。这里的硬件支持包括了两部分，第一部分是user/kernel mode，kernel mode在RISC-V中被称为Supervisor mode但是其实是同一个东西；第二部分是page table或者虚拟内存（Virtual Memory）。



所以，所有的处理器，如果需要运行能够支持多个应用程序的操作系统，需要同时支持user/kernle mode和虚拟内存。具体的实现或许会有细微的差别，但是基本上来说所有的处理器需要能支持这些。我们在这门课中使用的RISC-V处理器就支持了这些功能。