应用。这些应用软件不被也不需被信任。下一个信任级别是用户级签名应用程序级。这些应用程序只被授予其需要的权限。第三个信任等级由系统服务组成。同用户级应用程序一样，这些服务只需要特定的权限以便完成其任务。在一个如同Symbian操作系统的微内核体系结构中，这些服务运行在用户态，并像用户程序一样被信任。最后，有一类程序需要系统的完全信任。这组程序拥有修改整个系统的能力，并由内核代码组成。

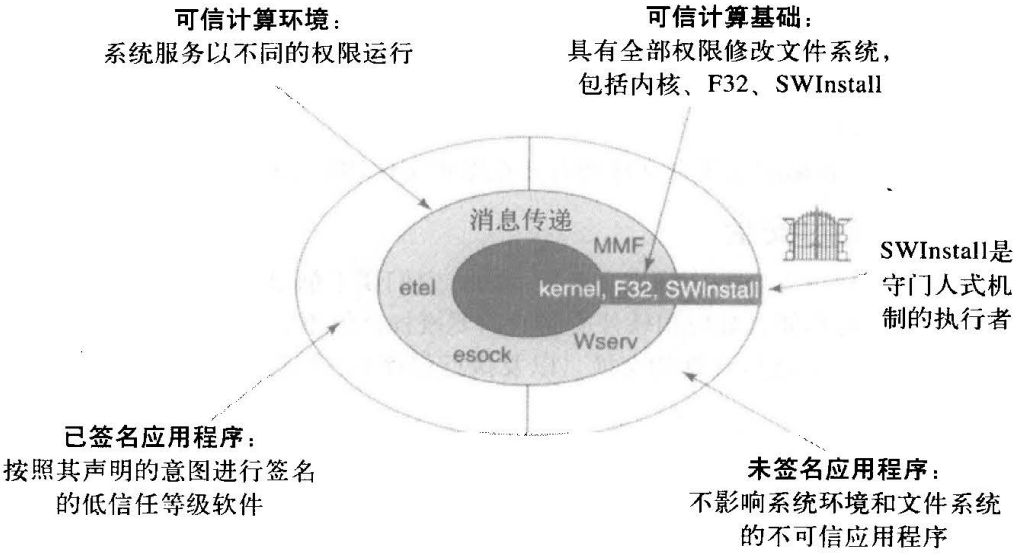

图12-3 Symbian操作系统通过信任关系来保证安全

在这个系统中有若干个方面看起来值得质疑。例如，这样复杂的机制真的有必要吗（尤其是需要花费金钱来制作的情况下）？结论是肯定的：Symbian签名系统代替用户来对软件进行完整性验证，并且该验证必须被执行。这一机制看起来可能会带来开发上的难度。是否每次在真实物理设备上进行测试都需要一个新的签名的安装包？为了解决这个问题，Symbian操作系统识别开发人员的特殊签名。一个开发人员必须获得一个有时效限制（通常是6个月）的证书和一个特殊的智能手机，即可使用自己的数字证书来创建安装包。

除了这样的守门人式机制外，Symbian操作系统版本9同时采用数据锁定（Data Caging）技术，来组织特定目录下的数据。比如，可执行代码只存在一个目录中，而该目录只对软件安装程序可写。另外，应用程序只能在一个目录中进行写操作，它们各自的数据不能被其他程序访问。

12.8 Symbian操作系统中的通信

Symbian操作系统按照特殊的标准设计，并使用客户机/服务器机制和基于栈的配置，以事件驱动型的通信为特色。

12.8.1 基本基础结构

Symbian操作系统的通信系统基础结构建立在基本构件之上。考虑如图12-4中所示的一个非常通用的模式。考虑把这个图作为一个可组织模型的起点。在这个栈的底层是物理设备，以一定方式链接到计算机。这个设备可以是集成在通信设备中的手机调制解调器或是一个蓝牙无线电装置。在此，我们不关心底层的硬件实现，而是把这个物理设备当做一个会以合适的方式响应软件发出的命令的抽象设备。

下一层，即我们需要关心的第一层，是设备驱动层。我们已经指出了设备驱动的结构；这一层的软件直接通过LDD和PDD结构与硬件配合工作。这一层的软件是硬件相关的，每个新型号的硬件设备都需要一个软件的设备驱动为其衔接。不同的硬件需要不同的设备驱动，但它们都为上层提供同样的接口。协议层期望无论什么样的硬件都具有相同的接口。

下一层就是协议实现层，包含了Symbian操作系统所支持的各种协议的实现。这些实现承担了下层的设备驱动接口，并向上面的应用层提供了一个单一、统一的接口。这就是提供诸如蓝牙和TCP/IP协议的各种协议的部分。