苹果技术白皮书

最佳实践部署 FileVault 2

部署 OS X 全盘加密技术

2012．08 – OS X 10.7.4

目录

概述

数据安全方面的违规可能会对公司客户、声誉和罚金造成重大损失。贵重的需要保护的物品要放进保险库中。同样的规则也适用于敏感的计算机文件，它们不是存放在物理保管库中，而是被存放在文件库里。

当用户在计算机上存储敏感信息时，他们的企业未来将会面临风险。例如，如果员工丢失了存有公司财务数据的笔记本电脑，可能会使某些人利用这些数据严重地损坏公司利益。使用强加密可以帮助保护这些文件的信息安全。

如果用户具有特殊权限，或者不受仅在受保护区内存储数据的规则限制，他们的数据将会有泄露的风险。加密整个磁盘可以确保所有信息都受保护，没有泄露的可能性。使用FileVault，公司的信息在硬盘上都是安全的。

第二代FileVault ,即FileVault 2。内置于OS X中，对于任意卷上的静态数据进行全盘加密（FDE）。FileVault 2使用磁盘级的XTS-AES-128数据加密来保护所有硬盘文件的安全，即使计算机丢失或被盗。使用苹果的软硬件集成，组织可以保护存放在内外部磁盘驱动器上的所有的用户数据，而这项技术对用户是透明的，也可以根据需要提供企业IT访问控制。

企业IT部门有很多选择来保护敏感数据，但是（更好地）理解（如何）在选择之间的权衡可以确保部署的是最佳方法。其中之一要考虑的就是FileVault 2被限定为基于密码的认证。

基于OS X中FileVault 2的优势，组织可以确保保护敏感数据免受恶意的未经授权的访问。管理员（将）具备基本的知识基础来创造性地部署和使用FileVault 2，以满足未在本文讨论范围、甚至更复杂的数据保护（应用）场景。

获得保护.保持简洁

内置保护

FileVault 2是内置于OS X中的可管理的加密技术，它可以保护整个卷。FileVault 2提供的全卷加密技术，可以很方便地被任何人使用，无论有或没有专职的IT人员。

有了FileVault 2，用户不必困扰于单独地为每个文件加密，或将文件存放在特定的加密容器中，（因为）整个卷上的所有数据都将被加密。这种“幕后”保护方式允许用户聚焦于他们的主要任务和特有的工作流程，而无需停下来考虑数据保护的问题。

本文有什么

为充分领会FileVault 2的强大和简洁，重要的是，要理解它的构架、各种配置的含义，以及如何更好地管理这些加密卷以满足特定用户/支持的需求。

理解FileVault 2的基础、理解用FileVault Master Identity (FVMI)来使得IT部门处理被保护的存贮，对于任何一个大型组织来说都是非常关键的。适当的配置和使用FVMI对于任何有效的企业部署都是重要的。

在更高层次上全面了解其他相关的支持技术也同样重要。CoreStorage细节、密钥链的操作和加密算法都是这些复杂性的例子，为了清楚起见而在文中提及但是没有深入介绍。

设计方法

FileVault 2的设计过程中主要考虑了以下几个关键因素：不仅需要提供可靠和安全的架构，还要适应不同的场景-从个人使用到受管理的企业访问。

可用性

最终用户使用方便是FileVault 2的核心。最终用户必须能够保护他们的数据而不必成为安全专家。拥有内置的个人安全网用以个人恢复（例如在Apple服务器上存储受保护的恢复密钥），对于最终用户来说也是至关重要的。这种保护必须在不增加复杂性的情况下从任何地方进行。

就FDE的可用性而言，一个传统的障碍是其历史上的对于最终用户二次认证的依赖：一次用于“预引导验证”来解锁加密的引导卷，第二次用于登录到目录服务帐户。如果凭证因为任意原因而同步失败，用户体验会很差。确保用户在无需重复验证下也可安全访问，这对于可用性是个挑战，而密码变更的同步则加强了可用性。用户必须能够随时重置密码而无需重新加密卷。

IT 访问

管理员惧怕数据失控的风险。但重要的是，他们应有一种机制，可以对最终用户的卷进行授权访问。即便在某些情况下亦可，简单如忘记密码，复杂至对卷的取证访问。使用FVMI来添加对机构恢复方法的支持，扩展了远超于个人自助式恢复方法的能力。这使得已授权的管理员可以按需，最小但明确，进行访问和修改。在大型组织中部署FVMI的最佳实践依靠最终用户和IT支持的能力与限制。

性能

使用FileVault 2，用户将受益于架构设计、加密算法以及硬件加速（当可用时）方面的优化。用于区块加密的算法是已经针对512字节块优化过的XTS-AES-128。明文到密文的转换以及逆向操作对于用户体验的影响是可以忽略不计的，因为它为用户活动而放弃了自身处理周期（即：加解密处理和用户操作是松耦合的）。若系统使用Intel Core i5和i7处理器，则软件优化可以被进一步地加速，i5和i7使用内置的AES-NI指令集提供硬件加速。

部署最佳实践

由于有各种各样的企业政策和方法保护数据，为所有用户选择单一部署场景未必是最好的方法。为帮助决定部署FileVault 2的最佳方式，最重要的是查看使用环境以及IT部门如何才能帮助最终用户，并且在需要的时候恢复对加密卷的访问。

下面这些部署策略自身所带的利弊会影响到安装、日常使用，及最大的挑战---按需恢复。

相比基于一个进行标准化---在全组织强制推行单一方案，大多数大型组织会决定混合搭配这些实践以适应各种类型的用户需求。重要的是，要理解基于组织内的用户的使用场景，以及IT人员的服务、能力和局限性。

部署

有三种部署方法论可以考虑：

自助式----与最终用户一对一的部署，（用户）完全控制其系统安装、使用和个人数据恢复。

- 优点：IT资源的占用非常小；完全离线恢复也是有可能的。
- 缺点：最终用户的职责和理解力；企业密钥托管受到挑战；合规的强制化和监控需要额外资源。

咖啡厅式---与自助式类似，但是IT可以提供含有选项和服务的菜单来协助最终用户进行初始化安装、日常使用和卷恢复。

- 优点：IT 监督；专家指导；按需使用已验证的工具。
- 缺点：合规的强制化和监控需要额外资源。

集中式---IT人员严格控制用户系统的安装和管理，同时确保所有数据随时受到保护和可恢复。

- 优点：合规与密钥托管有IT保障。
- 缺点：IT资源高度占用来进行部署和恢复计划的制定。

自助式方法论

一对一部署

自助式方法论是一对一的部署，致力于在最终用户扮演主要角色的环境中使用 FileVault 2：控制系统的安装、使用和恢复。这并不仅仅指能单独运行的，可与其他集中式企业目录服务分离的系统，也包括在单一系统上由单一用户对于 FileVault 2 的使用。

目标环境

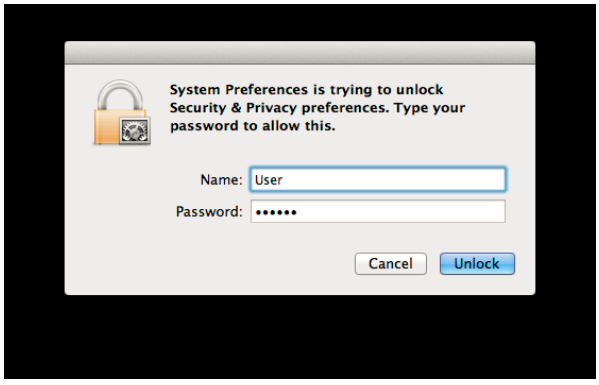
这种方法可能适合那些在其数据中心和网络服务中具有足够控制权的组织，以及一个可以从自使用自驱动能力中获益的用户社区。这种类型的部署，最终用户通常对所有事物进行控制，并负有责任，IT 人员有可能按需提供指导，但很大程度上并不插手。

在自己的系统上具有完全管理员能力的用户，以及没有网络访问但需要恢复支持的用户，这种方法是他们的最佳选择。

部署

通过自助式部署，最终用户可以自由、灵活地决定何时、如何按其工作流程和需求来启用、修改和禁用 FileVault 2。

此部署的要求之一是，个人必须具有修改系统配置和设置 FileVault 2 的本地权限。这些修改都将通过 System Preferences > Security & Privacy > FileVault 来执行。



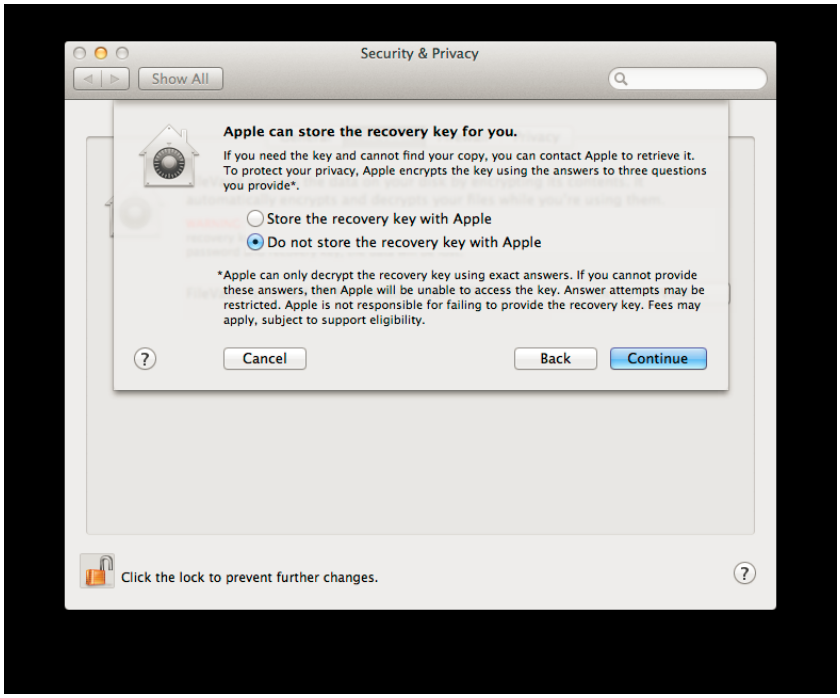
启用FileVault 2需要用户使用可以获得授权权限的帐户来解锁System Preferences：

system.preferences.security.

授权权限在授权数据库（/etc/authorization）中定义，它默认授予本地管理员组的成员。 如果最终用户不是系统的管理员，他们可以被授权修改 Security & Privacy 设置。 但是，这意味着用户可以修改系统上的任何安全设置。没有一个特定权限可以授予非特权用户用来仅仅管理 FileVault。但是，特权用户可以通过输入他们的凭证进行单次授权来协助最终用户启用 FileVault。

个人恢复密钥（PRK）
基于企业规章或恢复场景，安装程序可以采取两种路径之一：离线存储 PRK，或将其受保护地存储在 Apple 服务器上。公司规定可能要求所有加密和恢复密钥的存储都在由公司控制的基础设施中维护。其他公司可能会提出更高的要求使得任何时候都可以随时按需恢复，特别是当网络完全断开时。

选择离线存储意味着随机生成的 PRK 必须被记录下来或通过截屏获得，并保存在计算机之外。对于 PRK，没有自动化的系统保存方式，如果在 FileVault 启用过程中未注明或记录下来的话，其值可能会永久丢失。



选择在 Apple 服务器上存储 PRK 时，在提交存储前，会让用户回答三个安全问题，以此生成打包 PRK 的对称加密密钥。该密钥只能由最终用户才能知道的答案来解开。答案不应过于复杂以便用户可以在需要时及时想起。

恢复

启用 FileVault 2 不会改变日常工作流程，因为用户使用其 Directory Services 帐户密码定期登录系统。他们在预引导 EFI-登录时输入密码，当操作系统启动并需要登录凭据时，系统会启动密码转发。 密码转发消除了在冷启动后用户二次登录的必要。

但是如果用户忘记了帐户密码或 IT 部门需要在没有已授权的 FileVault 用户在场的情况下访问系统？ PRK 的使用提供了这个安全网。

在预启动登录时输入错误的密码三次，在密码字段下方就会显示一条提示消息：“如果您忘记了密码，可以使用恢复密钥重置密码。”用户必须单击消息旁边的三角形以显示恢复密钥字段，以及计算机的序列号和记录号。最终用户只需致电 AppleCare 并提供所需的信息，包括在设置中提供的三个安全问题的确切答复。为最终用户提供 AppleCare 支持可以减轻 IT 部门的工作量，使他们能够专注于加强公司整体的基础设施。

优点

自助式方法论在降低 IT 成本和给最终用户赋权方面具有最大的优势。正面的用户体验将极大地有助于企业安全的合规性。复杂的法规和耗尽资源的任务所引起的负面的用户体验将驱使最终用户采取变通的保护性安全措施。

选择离线存储可以实现所有类型的场景，从最终用户只需简单地写下 PRK 并将其存储在一个安全的位置，到稍微集成一点的方法---通过 Web 表单或电话将 PRK 提交到帮助台服务。离线存储的所有场景都允许组织在自身内部完全可控的范围内保留或者取回 PRK。

选择将 PRK 的受保护地存储在苹果服务器上为最终用户提供了随时随地恢复系统的灵活性，同时也提供了强大的密钥保护。若最终用户无法成功地在 FDE 预引导时验证身份，则意味着他们的计算机没有运行操作系统，并且远程访问服务不可用，因此帮助台的工作人员将无法通过网络去协助他们。将 PRK 存储在 Apple 服务器上不仅可以增加最终用户的灵活性，而且可以确保密钥得到严格保护，而无需特定的企业 IT 资源。

缺点

自助式方法论面临的最大挑战包括自动化托管 PRK 以及确保用户保持合规。

离线存储方法不会向组织提供自动化的过程将 PRK 托管到集中式服务器以供以后检索。凡期望对 PRK 进行成熟稳定的托管和检索都需要组织自己来集成客户管理服务或使用内部工具。如果这样，将 PRK 整合到现有企业托管服务中所产生的时间和金钱的额外成本可能会阻止这种做法。

没有经验的用户或 OS X 的新用户可能不会很乐意自己来管理和恢复。许多从其他平台切换过来的用户可能习惯于致电帮助台来寻求技术帮助。将责任转移到最终用户可能不会给这些用户带来最好的用户体验。

自助式部署会阻止组织利用集中式方法，（因为）它采用了机构恢复密钥（IRK）。这两种方法是互斥的。

如果对于最终用户来说，自助式部署的缺点超过了其优点，那么可以考虑 Cafe或集中式方法。

总结

自助式服务将设置，管理和恢复的责任转移给最终用户，从而大大降低了 IT 成本并改善用户体验。感到满意的用户更有可能是合规的用户，从而进一步降低 IT 成本。

从其他平台转换过来的没有经验的用户可能不乐意自己来管理和恢复。将责任整体转移给最终用户的方法可能不会为这类用户提供最佳的用户体验。该方法还需要管理员权限来启用和禁用 FileVault 2，这可能不是期望的，甚至会因为是企业控制的资产的原因而不被允许。

重置帐户密码，并确保按需进行授权访问，例如当用户完全脱离任何网络时 – 将使最终用户和企业 IT 双方都受益显著。

自助式方法最适合在其数据中心和网络服务中具有足够控制权的组织，以及一个可以从自使用自驱动能力中获益的用户社区。

咖啡馆方法论

协助部署

咖啡馆式方法是一种协助式的管理，适用于拥有需要控制大部分自身使用的用户社区，但有严格的企业或者行业强制要求，需要IT人员协助最终用户以确保数据得到妥善保护的组织。使用这种方法，IT管理员可以按需提供安全专业知识、部署选项和恢复协助，但将实现和使用任务留给最终用户。

目标环境

该方法在用户控制、公司策略以及用户帐户的安全保护和恢复方面提供了一个绝佳的这种方案。IT人员和最终用户都会参加到这个过程中。并且也可以准备好正式的制衡措施，以确保企业或者行业需求的合规性。

专业的IT人员可以按需为最终用户提供必要的培训和指导材料。IT人员还可以开发并提供本地工具和服务来监控加密状态，并提示用户在适当时采取行动。

初次接触到这个平台的大型组织或许可以使用这种方法来过渡到更受管理的环境，同时仍然可以提供正面的用户体验。在相对自由的IT环境中工作的用户可能会抗拒来自企业IT部门更为强力的控制。这种责任的分担减轻了双方的焦虑，在改善用户体验的同时，也为基础设施方面的改善提供了时间。

部署

通过协助部署，最终用户仍然拥有巨大的自由度和灵活性，但与企业IT部门分担启用、修改和禁用FileVault的责任。共享的自由和责任需要最终用户和企业IT部门的合作。

具有完整的系统访问权限和间歇性网络访问的管理者用户通常是这种方法的最佳候选者。组织仍然可以让IT管理员帮助那些没有/不希望有如此广泛的责任的用户。这种分担的责任允许在设置和恢复上具有不适用于自助式或者集中式的灵活性。

对于部署设置的要求取决于所选择的最能服务于用户工作流程的恢复方法----个人或者机构。虽然个人和企业的恢复方法不可能同时生效，但是可以在两者之间来回切换。这种切换的灵活性要求在一种方法下解密卷而在另一种方法下重新加密。

恢复

这种恢复方法允许使用个人密钥恢复(PRK)和机构密钥恢复(IRK)进行个人恢复，以使用户能够在标准登录窗口（非预引导登录）中锁定的系统上重置帐户密码。

使用PRK要考虑到，在必要时，允许更具颗粒度的存储和企业IT参与。
更多PRK信息，请参考“自助式方法论”部分---并通读这份文档获取进一步深入的信息。

密码重置

使用IRK是一种有效的方式，可以让企业IT部门帮助最终用户进行密码重置，而无需控制整个过程。然而IRK不能被最终用户用于预引导恢复，这是因为系统无法访问存储了IRK的卷。IT人员可以，例如从帮助台，向最终用户提供预设的主密码，以便在标准登录窗口输入。如果IRK存在而用户无法在标准窗口登录，则电话告知主密码可以作为一个标准支持，或者就仅限于对受保护的企业网站进行安全/被验证的访问。这允许最终用户和IT人员合作重置密码，这种方法在离线的情况下特别有用。

主密码用来保护 FileVault 主密钥链（FVMK）的密码，FVMK 包含了整个 FileVault 主标识（FVMI）

最终用户可以使用隐藏的恢复HD安全访问并从受企业保护的网站取回PRK。所有运行FileVault 2的系统都有为此目的而设计的隐藏的复原HD分区。该分区在启动盘（Startup Disk）系统首选项下不可见，并且在磁盘应用（Disk Utility）程序下也无法查看。最终用户可以重新启动Mac，并立即按下键盘上的Command-R。 一旦Mac开始进入到恢复模式，便可松开按键（Command-R）。 恢复模式将会加载受限的访客访问环境，用户可以打开Safari来访问并向受企业门户网站进行身份验证。一经验证，他们可以从企业安全存储中取回PRK。

优点

咖啡馆方式在最终用户和企业IT部门间分担责任方面具有很大的优势。正面的用户体验将极大地有助于企业的安全性合规。这种方式允许根据需求和能力有选择地赋予最终用户/IT人员责任。

通过制定明确的指导和创建恢复密钥的入口，IT部门可以在前端加载大多数资源需求，以此给予最终用户通过可控的/受保护的访问方式自己来执行任务。在用户自己保管敏感信息的同时---不在公司基础设施之外存储和使用密钥，IT人员还可以改进指导、恢复和监控。

通过内置的恢复HD（Recovery HD）和受限的访客访问方式，最终用户可以安全地接触到受企业保护的内容而不会泄露敏感信息或危及系统/加密卷的完整性。

缺点

咖啡馆方法面临的最大挑战是决定分配何种责任给各方，以及需要何种IT资源来开发使用安全的密钥托管和检索入口的网络版指导内容。

大多数组织已经使用网络版内容用于培训、技术知识（分享）和指导，所以提供额外的有关FileVault的用户内容，其所需资源可忽略不计。任何密钥托管和检索入口都需要特定的开发来获取、传递PRK自身或IRK的主密码。

主要的缺点是需要资源和开发来处理IT方面的共担责任。如果一个组织使用的客户端管理解决方案集成了可靠的OS X，那么它就可以经常性地用供应商提供的工具箱执行所有或者大部分IT任务。

总结

咖啡馆方法适用于这样的组织：拥有可胜任的用户社区，但是可能会需要IT人员有限参与。使用这种方式，IT部门可以提供安全专业知识、部署选项和协助恢复，而让最终用户自己进行实际的启用和恢复操作。

有经验的IT人员可以提供培训、指导以及开发恢复密钥的托管 / 检索门户给最终用户来进行受保护的访问。提供了一种灵活的方式、对应于部署给每个特定用户的（不同）方法，可以变更恢复密钥的获取和检索。从其他平台切换过来的没有经验的用户可能会更乐于依赖IT人员进行管理和修复。

咖啡馆方法最适合这样的组织：具有强大IT资源池，但需要权衡最终用户的需求和要求，以期让他们最大程度地使用FileVault 2。IT部门可以扮演内容领域专家的角色，为最终用户提供建议和指导，而不会限制最终用户的体验。

集中式方法论

IT 控制部署

集中式方法是 IT 控制的部署方式，这种方式可以是这些组织的最佳选择：必须给予 IT 人员对用户系统的严格控制，在这些用户设备上的所有数据必须使用完全磁盘加密（FDE）进行保护，但只能由 IT 参与进行恢复。这种部署形式需要 IT 人员对加密卷的所有方面有所控制：创建、管理和恢复。充分的 IT 参与确保了系统的正确部署和审计，以及用企业或行业的规定进行强制合规。

目标环境

该方法适用于高度监管或安全的环境。过程的每个阶段都由授权的 IT 人员控制和执行，以避免最终用户引发的任何错误，并确保无论何时都遵循所有规定。

具有强大的平台知识的大型组织将在严格管理的环境中采用此方法，同时他们通常也会使用同样强大的客户端管理工具。许多客户端管理工具都支持部署和管理 FileVault 主密钥串（FVMK）、FileVault 主标识（FVMI）以及相关的由苹果提供的命令行界面（CLI）工具，例如 OS X 中包含的 security。

用于配置和托管 FMVI 的企业证书颁发机构（CA）对于 FDE 解决方案的部署至关重要。伴随在专职安全管理员、工作组管理者和帮助台人员之间分离职责，对于授权和访问安全敏感信息也必须受控。

部署

应用集中式方法，正确创建、保护和管理 FVMI，对 IT 员工分期访问完全加密的卷至关重要。这是三种方法中最复杂的，通常由 IT 人员代替最终用户或由 OS X 高级用户来执行。组织使用的工具集可以是多样化的，只需要在一些特殊的细节方面仍然满足 OS X 的要求。

成功的、可管理的 FDE 保护/访问取决于对 FVMI 和每个相关组件的作用的理解。FVMI 提供了非对称密钥对（公钥和私钥），用于打包/解包密钥加密的密钥（KEK）对加密卷进行授权访问。（详细信息，请参阅“附录 A：架构概述”。）这种 IT 访问方法尤其有趣，因为它不需要任何授权用户的凭据来解锁完全加密的卷。

组织必须首先启用一个 FVMI，无论是使用自身的企业 CA，即内置的“设置主密码”的 GUI，还是用 CLI 命令“security”来创建植入 FVMI 的 FileVaultMaster.keychain。

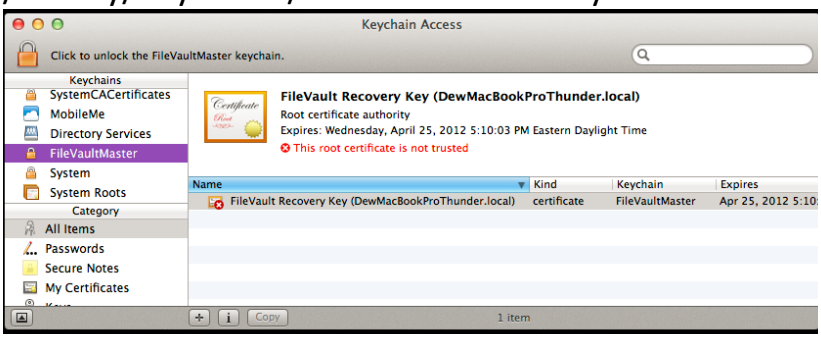
启用 FVMI

对于组织来说，最好是使用自己的企业 CA 来恰当地启用 FVMI。私钥托管是其主要的好处。必须根据“附录 A:“架构概述”中的要求启用 FVMI。与所需属性和属性值的任何偏差都将妨碍使用 FVMI 对密钥进行打包/解包。

如果基础设施和 IT 资源可以支持足够的复杂性和粒度度，最好为每个即将使用 FileVault 2 的系统启用 FVMI。这不是 FileVault 2 的技术要求，但在此级别启用，组织可以有选择地控制每个系统的密钥，有些密钥甚至可能来自单独的中级 CA。与之相反的方法则是为所有已启用 FileVault 的系统发布单一 FVMI。如果这个单一 FVMI 由于任何原因而受到威胁，则会使组织所有的系统变得脆弱 – 一个身份可能会解锁或解密所有使用该 FVMI 的计算机。

部署 FVMI

名为“FileVaultMaster.keychain”的密钥链文件---应该仅包含 FVMI 的证书组件---应使用客户端管理工具推送到现有客户端，并在用镜像创建新系统时包含在磁盘镜像中。它必须放在文件系统的以下路径里：/Library/Keychains/FileVaultMaster.keychain。



为启用 FileVault 2，需要用户使用可获得授权的帐户（systems.preferences.security）解锁系统首选项（system preference）。这些权限在授权数据库（/etc/authorization）中定义，它默认授予本地管理员组的成员。

因为启用 IT 控制的 FileVault 2 只需嵌入在证书中的公钥，所以只有证书包含其中。通过将最敏感的组件（私钥）排除在外，便能缓解 FVMI 可能遭受的任何潜在的恶意威胁。只有在加密卷上执行恢复时才需要私钥。如果系统中包含完整的 FVMI，并且有未经授权的个人试图获得访问权限，那么他将能够解锁或解密该系统以及任何启用了相同 FVMI 的其它系统。

OS X 需要管理员有权限使用 GUI 来设定 FileVault 2，并修改系统配置以打开或关闭它。这是通过点击 System Preferences> Security & Privacy> FileVault 中的 Turn On FileVault 执行。因为 FVMK---一个特殊的凭证存储---仅随证书产生，并且已经放置在/Library/Keychains/FileVaultMaster.keychain 上，所以管理员将收到如下通知。

该通知表明 OS X 已经将妥善部署的 FVMK 识别为 IRK，并且证书中的公钥将用于打包 KEK 并存储以供授权 IT 后续访问。如果没有部署 FileVaultMaster.keychain，或者密钥链包含全部 FVMI，则该过程将默认设置 PRK，如“自助式方法”部分所述。FileVault 2 将识别 PRK 或 IRK， 但不能同时识别。

设置主密码

组织还可以通过“设置主密码”直接在设备上启用和部署 FVMI。设置主密码简单来说就是指，请求操作系统生成随机的 FVMI，并将其存储在新创建的 FileVaultMaster.keychain 中，然后将 FVMK 的密码设置为主密码。任何对密钥链或 FVMI 托管的需求都可以在主密码设定之后执行。

脚本启动 FVMI

另一个启动 FVMI 的方法是使用“security”CLI 命令。“security”命令的使用将在本节后面进行更详细的说明。

备用管理员帐户

我们也强烈建议为 FileVault 2 配置并启用备用管理员帐户。这可以作为管理员重置帐户密码和解锁卷的另一种方法，该方法不需要在完全加密的卷上执行恢复操作。这也允许 IT 人员在对 FileVault 2 了解不深的情况下协助最终用户重置密码。

还原

启用 FileVault 2 不影响日常工作流程，因为用户使用其目录服务帐户密码登录系统。他们在预启动 EFI-Login 时输入密码，在操作系统启动后需要用户的登录凭据时，系统将开始密码转发。密码转发消除了在冷启动后用户要二次登录的需要。

如果没有任何已授权的 FileVault 用户在场，IT 该如何访问系统呢？IRK 的使用为 IT 人员提供了这个安全网。

密码重置

要重置帐户密码，加密的引导卷必须首先被解锁和激活，如果只有一个用户，这是不可能的。由于计算机首先使用 EFI 进行预引导，这需要一个授权

用户解锁卷并继续引导过程。这就是为什么为 FileVault 2 配置并启用备份管理员帐户来执行密码重置非常重要的原因所在。

当备用管理员解锁卷并访问系统时，就可以为用户帐户（通过用户和组首选项）进行标准的密码重置。当从 OS X 重置密码后，相应的对 FileVault 2 的用户访问也将重置，以使用新密码 - 从用户密码生成的派生加密密钥（DEK）。

如果用户的密码是从外部目录服务端（例如，在 Active Directory 中）重置，那么用户的新密码将不会在预引导时解锁该卷。同样，这将需要备用管理员解锁卷、注销并将系统转交给用户，以在登录窗口输入新的凭据。

基于 FVMI 的还原

要在任何时候对完全加密的卷进行任意还原或取证类访问都需要完整的 FVMI---同时包括证书（有内嵌的公钥）和在有效密钥链（例如，FVMI.keychain）中的私钥。这将假定从企业 CA 获取 FVMI 的所有步骤已由授权的 IT 管理员执行。创建一个空的密钥链文件并自动导入 FVMI 是极其容易的。可以有几种不同的技术创建密钥链和导入 FVMI，包括使用“security”命令完全脚本化。仅使用“Terminal”应用程序和 CLI 命令的一种方法是：

1. 创建一个新的FVMI密钥链并存储在USB上。
    a. # security create-keychain -P /Volumes/ <PathToFile>/FVMI.keychain
    b. This creates the keychain named FVMI.keychain at the entered path.
    c. When prompted, enter the desired password to protect the keychain.

2. 导入从企业CA导出的FVMI（即FVMI.p12）。
    a. `# security import /PathTo/FVMI.p12 -k /Volumes/ <PathToFile>/FVMI.keychain -f pkcs12 -P`
    b. Enter the passwords protecting the keychain and the .p12 files as prompted.
3. 将要恢复的计算机引导到Recovery HD。
    a. 按下 Command–R 以引导至 Recovery HD.
2. 从“Utilities”菜单中打开“Terminal”。
3. 识别加密卷的逻辑卷UUID（LvUUID）。

a. 参见“附录 A: 架构概览” 以获得更多关于CoreStorage卷管理（CoreStorage volume management）的信息。

4.插入USB。

5.解锁FVMI密钥链。

```
a. # security unlock-keychain -p <password>
   /Volumes/<PathToFile>/FVMI.keychain
```

6.解锁并挂载CoreStorage加密的目标卷。

```
a. # diskutil cs unlockVolume <LvUUID>
   -recoverykeychain
   /Volumes/<PathToFile>/FVMI.keychain
```

完成这些步骤后，该卷被解锁并挂载，以便通过其他工具或服务进行完全访问。这允许对卷上的文件进行任何更新和修改（例如，为系统或应用程序文件打补丁）。

如果尝试从相同的加密卷进行引导，则无法使用FVMI执行解锁。这就是为什么需要首先从其他驱动器（如内置的Recovery HD）引导。Recovery HD是一个不可变的镜像---用于完整性验证，并作为只读引导卷挂载。任何完整性问题都将阻止其使用，然后默认使用基于Internet的恢复---如果它是一个较新的基于OS X的系统，并在固件中支持Internet恢复，这也可以防止任何人修改隐藏的Recovery HD，例如安装软件或更改配置。

优点

集中式方法的最大的优势在于，为IT人员提供了完全控制---随时预置FVMI并访问加密的卷。积极的用户体验有助于企业安全合规，但在这种情况下，IT部门严格掌控使用FileVault 2的所有阶段。用户除了日常使用之外通常不参与FileVault 2的任何阶段的管理。

使用自己的企业CA进行FVMI的配置和私钥管理的这种方式对IT部门相当有吸引力。只要CA能符合“附录A：架构概述”中相应的要求，那么组织就可以选择适合其安全需求和IT资源的颗粒度级别。使用FileVault2，组织不会局限于用单一方法或途径来托管密钥。

将FVMI置于可以恢复到新系统的镜像，或将其推送到所有现有的OS X系统，可以使IT部门始终可以在没有已授权用户凭据的情况下访问这些卷。大部分流程可以脚本化并集成到现有的工作流中进行镜像安装和部署。

缺点

这是三种方法中最复杂的，需要IT人员或OS X高级用户参与管理FileVault 2的所有阶段。组织使用的工具集会有所帮助，但是如果没有最终用户输入他们的帐户密码，就没有任何解决方案可以完全启用FileVault2。

在中央化地区，OS X不会在跨多个系统生成和托管FVMI时，提供一键自动化功能（即，数据中心没有内置控制台支持）。如果遵循最佳实现的要求，则要求组织使用多个企业服务，并通过内部开发或使用额外的IT资源来集成这些特性。

总结

集中式方法是这些组织的最佳选择：必须让IT人员对用户系统进行严格的控制，并确保FDE只能在IT控制下进行恢复。这种部署形式需要IT人员对加密卷的创建、管理和还原所有方面进行操作。

尽管全面的IT参与确保了更合适的部署、系统审计和基于企业或行业要求的强制合规，但这意味着IT成本的增加和更复杂的规划和执行。

IT人员必须熟悉FVMI的配置、托管和使用情况，以便使用IRK进行正确的布置和还原。

合规

在组织内部经常有若干行业强制要求，涉及计算机部署和静态数据保护的整合。最常讨论的两个是对于FIPS 140-2一致性验证（Conformance Validation）的要求和对辅助技术（Assistive Technology）（通常是指Section 508 或可达性---Accessibility）的支持。

FIPS 140-2 一致性验证

美国和加拿大政府在美国国家标准与技术研究所(NIST)和加拿大通信安全机构 (CSEC)内拥有一个验证程序---加密模块验证程序(CMVP)，用以审查和验证商业化产品中所使用的加密模块。凡实施了验证的加密模块自身、使用了有效加密模块的任何操作系统、服务或者应用都被称为FIPS 140-2合规。

OS X Lion 10.7是第一个实现FileVault 2(FDE) 的操作系统发行版。新的基于内核的加密模块CoreCrypto ---FileVault 2构建于其上---在OS X Lion下没有提交FIPS 140-2一致性验证。

在OS X Snow Leopard 10.6中用于加密磁盘映像、S / MIME和SecureTransport等服务的加密模块，在2011年3月9日实现了FIPS 140-2验证。苹果已经实现了该模块（仍存在于OS X Lion中）的重新验证。单独使用Legacy FileVault或在FileVault 2上同时运行OS X Lion将使用该重新验证的模块，并且符合FIPS 140-2，但是会导致性能的下降。

OS X中的加密模块已经提交，并且有望取得OS X Mountain Lion 10.8发行版的验证。OS X Mountain Lion 10.8版本以下的FileVault 2将是FIPS 140-2合规的。

508 部分 (可达性)

具有预引导验证的FDE的架构方法阻止了对可达性的支持，并可能阻止身体残疾的用户使用FDE服务。在预引导时，此设备尚未有操作系统运行，因此任何依赖于操作系统的技术（如可达性）尚不能由设备使用。对于需要有可达性支持的组织，最好继续使用OS X Lion上的Legacy FileVault来保护用户的主目录，同时启用可达性服务。

总结

数据安全性的毁坏可能会以许多方式损害企业。保护个人和企业敏感数据至关重要，当数据被加密并且未授权人员不可触及时，保护措施得到改善。如果用户具有特殊权限，或者未受限于将数据仅存储在受保护区域的策略，组织可能会有如下风险：将文件存储在加密区域之外而导致的意外的数据泄露。加密整个磁盘可确保所有信息受到保护。 启用FileVault 2后，硬盘上的所有信息都是安全的。

苹果为所有客户开发了并将继续推进具有三个核心设计目标的FileVault 2功能：易用性、IT访问和性能。

FileVault 2例证了苹果的软硬件集成、帮助个人和组织保护所有存储在内部和外部的用户数据，而这些数据对用户几乎都是透明的，并且在某些情况下，不需要IT资源。最终用户一定能够保护他们的数据而不必成为安全专家。

最终用户和管理员害怕会失去对数据的控制。个人和机构的恢复方法被用于对加密卷进行可控的和授权的访问，这提供了每个人都需要的安全性。IT部门在各种情况下都可以访问最终用户卷是至关重要的。为机构恢复密钥（IRK）提供与FileVault主体身份（FVMI）的集成支持扩展了仅使用个人恢复密钥（PRK）的个人自助式恢复方法的功能。

在过去，性能限制了静态数据的解决方案，但FileVault 2让人眼前一亮---结构性的设计、优化的加密技术和充分利用硬件加速。FileVault 2在提供了预期的保护的同时还增强了个人和组织的体验。 仅就这个事实便让其与众不同。

企业IT部门有许多不同的方式来保护敏感数据，但是对于自身优势的认知和选项之间的权衡将确保最佳方法得以部署。使用FileVault 2，加上正确配置并托管的恢复密钥和对于部署方法的有效认知，组织可以始终确保敏感数据受到保护免受未经授权的访问。

附录 A: 结构概述

FileVault 2被定义为是对用户和管理员访问完全加密卷---引导和数据卷---最好的全系统管理工具。它不是单一的过程或组件，更多地是将多组件管理融合为OS X的一个特性。这就解释了理解每个组件是什么以及他们是如何作为一个特性而相互关联的重要性。

FileVault 2由三个基本组件组成，他们为大规模部署提供了许多整体功能：

- CoreStorage
- 密钥管理
- 恢复方法

CoreStorage

CoreStorage是用于高级逻辑卷管理器（LVM）的Apple架构。CoreStorage与FileVault不同，而是FileVault用于存储和检索加密位的逻辑卷管理。在OS GUI下，CoreStorage对FileVault的支持也称为CoreStorage全磁盘加密（CSFDE）。

CoreStorage的详细说明不在本文的范围之内，但对其架构和与FileVault之间的关系的高层次理解至关重要。

CoreStorage架构图反映了每个逻辑卷组（LVG）中的多个物理卷（PV），可使得多个逻辑卷（LV）- 每个选择性加密或不加密。

CoreStorage的组件：

- 物理卷(PV)
- 逻辑卷组(LVG))
- 逻辑卷族(LVF)

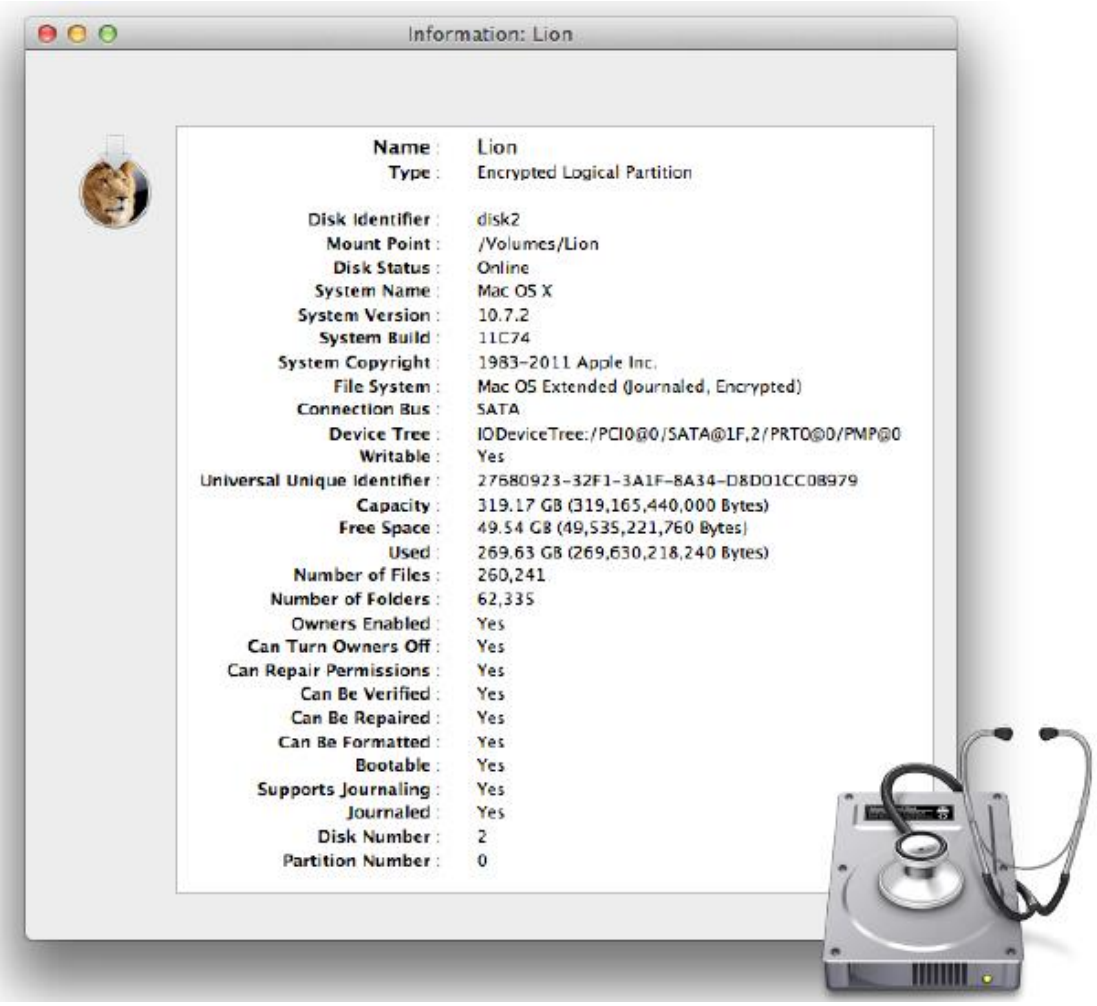
- 逻辑卷(LV)

物理卷(PV)是逻辑卷组(LVG))的一部分，其中包含了一个逻辑卷族(LVF)，它描述了将生成的逻辑卷(LV)的属性。

CoreStorage架构需要多个LVG，LVF和LV，但是简单起见，让我们聚焦在单个逻辑卷组 (LVG) 中的单个物理卷 (PV) ，这将创建单个加密的逻辑卷 (LV) 。

每个CoreStorage卷都单独使用其自己的全局唯一标识符（UUID）引用，格式为128位值，保证在空间和时间上是唯一的。 UUID的标准格式用ASCII表示为用连字符标点的字符串（例如，86E9812-167F-4129-B1AA-E6A041C69EA6）。

可以使用“Disk Utility”应用程序或名为diskutil的命令行界面（CLI）工具查看和操作CoreStorage对象。



```
Usage:  diskutil [quiet] coreStorage|CS <verb> <options>
       where <verb> is as follows:

list                (Show status of CoreStorage volumes)
info[rmation]       (Get CoreStorage information by UUID or disk)
convert                   (Convert a volume into a CoreStorage volume)
revert              (Revert a CoreStorage volume to its native type)
create              (Create a new CoreStorage logical volume group)
delete              (Delete a CoreStorage logical volume group)
createVolume        (Create a new CoreStorage logical volume)
unlockVolume        (Attach/mount a locked CoreStorage logical volume)
changeVolumePassphrase(Change a CoreStorage logical volume's passphrase)

diskutil CoreStorage <verb> with no options will provide help on that verb
```

要收集已有的CoreStorage卷的信息，在“Terminal”窗口中执行以下“diskutil”命令。

```
$ diskutil cs list
+-- Logical Volume Group 486E9812-167F-4129-B1AA-E6A041C69EA6
    =========================================================
    Name:         Lion
    Sequence:     1
    Free Space:   0 B (0 B)
    |
    +-< Physical Volume ABF9FDE7-8481-43FB-A9BB-1E316C182DC1
    |   ----------------------------------------------------
    |   Index:    0
    |   Disk:     disk0s2
    |   Status:   Online
    |   Size:     319484211200 B (319.5 GB)
    |
    +-> Logical Volume Family 02064501-2BE1-442E-B317-C21379CE5DD2
        ----------------------------------------------------------
        Sequence:                  13
        Encryption Status:         Unlocked
        Encryption Type:           AES-XTS
        Encryption Context:        Present
        Conversion Status:         Complete
        Has Encrypted Extents:     Yes
        Conversion Direction:      -none-
        |
        +-> Logical Volume F9EC4CFD-9440-4A43-A3F9-83F670153DFC
            ---------------------------------------------------
            Disk:                  disk2
            Status:                Online
            Sequence:              4
            Size (Total):          319165440000 B (319.2 GB)
            Size (Converted):      -none-
            Revertible:            Yes (unlock and decryption required)
            LV Name:               Lion
            Volume Name:           Lion
            Content Hint:          Apple_HFS
```

FileVault加密使用带有256位密钥的AES-XTS-128加密算法遵循NIST SP 800-8E中的NIST规则。

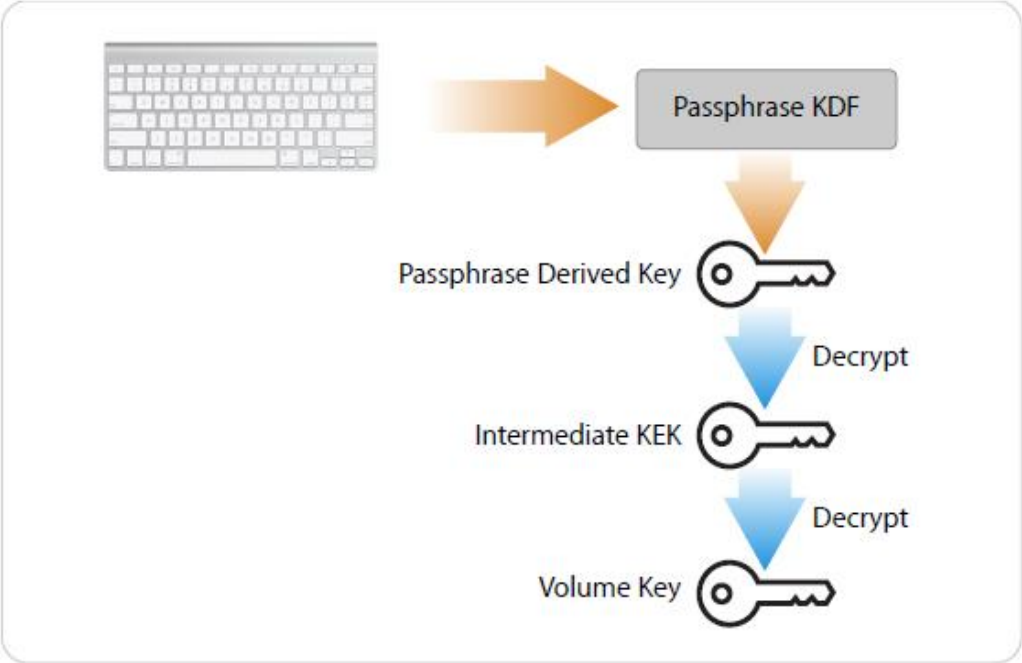
使用合适的UUID来执行以下命令，可以获得任何卷上的加密状态和存在性：

```
$ diskutil cs info 02064501-2BE1-442E-B317-C21379CE5DD2
    CoreStorage Properties:
   Role:                     Logical Volume Family (LVF)
   UUID:                     02064501-2BE1-442E-B317-C21379CE5DD2
   Parent LVG UUID:          486E9812-167F-4129-B1AA-E6A041C69EA6
   LVF Encryption Status:    Unlocked
   LVF Encryption Type:      AES-XTS
```

此样本卷使用AES-XTS完全加密，目前已被挂载并解锁。

在LVF UUID上执行“diskutil cs info”还提供了一种定位和识别父级LVG UUID的方法。使用LV UUID执行相同的命令时，UUIDs的完整链将列出从LV UUID，到父级LVF UUID，再到父级LVG UUID，以及转换为加密卷的状态 - 在此称为“完成”：

```
$ diskutil cs info F9EC4CFD-9440-4A43-A3F9-83F670153DFC
    CoreStorage Properties:
   Role:                     Logical Volume (LV)
   UUID:                     F9EC4CFD-9440-4A43-A3F9-83F670153DFC
   Parent LVF UUID:          02064501-2BE1-442E-B317-C21379CE5DD2
   Parent LVG UUID:          486E9812-167F-4129-B1AA-E6A041C69EA6
   Device Identifier:        disk2
   LV Status:                Online
   Conversion Status:        Complete
   Content Hint:             Apple_HFS
   LV Name:                  Lion
   Volume Name:              Lion
   LV Size:                  319165440000 B
```



密钥管理

本节通过操作系统审视FileVault 2下的密钥管理，并不涉及机构对于恢复密钥的管理。有关机构的恢复密钥的配置和管理的详细信息将在本附录后面介绍。

对于任何使用CoreStorage的加密卷，都有一个已封装的加密密钥字符串来保护数据，并提供已验证的访问、用户密码的重置，以及对恢复访问的支持。这里的重点是，为引导卷使用FileVault，但是CoreStorage还提供数据卷上的磁盘密码，其中单个唯一密码也可能对部署有用。

密钥管理涉及加密密钥和恢复密钥的管理和使用。 有三个级别的加密密钥和两种恢复密钥。

要理解这三级密钥封装架构的目的和灵活性，首先要理解所涉及的三个密钥：

- 卷加密密钥(VEK)
- 密钥加密密钥(KEK)
- 派生加密密钥(DEK)

卷加密密钥（VEK）

在最低级别，CoreStorage使用了一个对称的VEK操作512字节的逻辑块。一个逻辑卷上的所有加密操作对于该卷是唯一的，因为对于每个独立的逻辑卷，每个VEK是随机生成的。这个唯一的VEK与每个块的调整相结合，在加密所有数据块方面提供了更大的优势。因为访问卷上的数据取决于其VEK，所以在卷的生命周期中必须保持不变。最终，CoreStorage真正需要的唯一的密钥就是这个VEK。

在该块级别执行的加密是使用了256位VEK的AES-XTS-128。XTS是国家标准技术研究所（NIST）推荐的用于存储设备保密性的新的AES块密码模式。首字母缩略词XTS代表XEX Tweakable Block Cipher with Ciphertext Stealing。根据NIST的说法，“在没有验证或访问控制的情况下，XTS-AES比其他已被批准的针对未授权操作加密数据的机密性模式提供了更多保护。”更多XTS-AES细节，前往 http://csrc.nist.gov/publications/nistpubs/800-38E/nist-sp-800-38E.pdf.

一个三级密钥的间接方式减轻了之前机构对于其他平台所需的卷进行重设密钥的需求。如果组织想要在这个低级别对卷重设密钥，则需要对整个卷进行完全解密和重新加密。如果在该卷上使用FileVault，则此概念通常称为“非保管和重新保管”。

密钥加密密钥（KEK）

在加密卷初始化期间，会随机生成一个对称的中间KEK，加密(封装)VEK，并存储在CoreStorage元数据中。该中间密钥允许间接地支持前面提到的设计方法中的要求。该间接方法允许DEK彼此间可以独立地更改，同样也独立于VEK。它还允许VEK独立于DEKs进行更改。一旦VEK被KEK封装，则生成的加密Blob将存储在卷的CoreStorage元主数据中，用于实际检索和解密数据。

派生加密密钥（DEK）

在最高级别，DEK必须可以启动解锁剩余的两个密钥的链，从而可以解密和访问加密卷。任何DEK都可以独立更改，不会影响KEK本身甚至VEK。这种特质在启用多种方式访问相同加密卷时显得非常强大，因为无需暴露任何密钥材料(或在机构恢复的情况下，甚至无需知道特定用户的凭据)。

给定的CoreStorage卷必须支持多个密码用户----每个用户都有自己的DEK。密码用户可以是，但无需是，与操作系统已认证用户相同。密码用户不过是用

一个给定密钥来解锁卷而所需的所有参数。输入任何一个已保存的、有效的密码用户的凭据都可以解锁一个给定的卷。

有两种方法派生出此顶级密钥：

- 基于密码字符串的DEK（Passphrase-based DEK）
- X.509基于身份验证的DEK（X.509 Identity-based DEK）

基于密码字符串的 DEK （Passphrase-based DEK）

当输入密码字符串时,使用SHA256-HMAC的RSA基于密码的密钥派生函数（PBKDF2）（用于内部伪随机函数）将其转换为密钥，这被描述为PKCS＃5 v2.0以及RFC2898。该结果密码将被用作DEK。

当密码字符串被用于生成DEK时，需要经过两步：

- 基于登录密码的DEK
- 基于磁盘密码的DEK

该系统的主要目标是抵抗针对加密卷进行的离线密码猜测攻击。如果用户的密码只是简单地哈希，攻击者可能会对其进行预计算或者暴力攻击，此时有可能会使用专门的硬件。为了加大攻击难度，可使用一个通用的密码技术：对密码字符串多次迭代一个哈希函数，以难以并行化的方式混合混淆结果。

基于登录密码的DEK

经过操作系统验证过的用户是任何一个已被允许解锁加密卷的用户。用户的登录密码是其各自的目录服务帐户密码。因为FileVault必须支持多个操作系统认证的用户进行解锁，所以每个登录密码都将转换为自己独有的DEK，它们将KEK各自单独封装，并在卷的CoreStorage元数据中和用户绑定存储，即谓之后续检索。

支持多个经操作系统认证的用户意味着任何启用了FileVault的用户都可以按照预期解锁整个卷。但是，如果系统环境需要额外的分离加密和容纳用户文件，请同时考虑使用Legacy FileVault（FileVault 1）。涵盖在使用了FileVault FDE的OS X系统上的Legacy FileVault（FileVault1）的内容超出了本文的范围。总之，这是一系列延续性使用：FileVault1、用户主目录的基于容器的加密、在FileVault 2之上提供了全盘加密。

基于磁盘密码的 DEK

考虑磁盘密码的一种特殊用例，使用单个密码字符串生成单个DEK来封装和解锁相应的KEK或卷。以这种方式保护的卷可以被认为是类似于具有单一的、

非操作系统验证用户的并且启用了FileVault的卷。磁盘密码禁止单独的用户使用自己的密码解锁卷，并阻止使用IRK访问该卷。

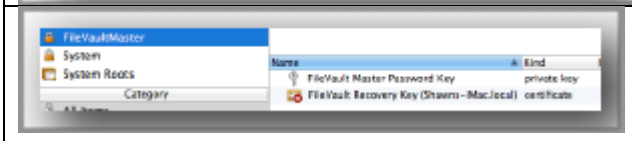
基于X.509身份的DEK

机构需要一种方法来解锁对加密卷的访问，而无需启用了FileVault的用户密码。 这就是公共密钥基础设施（PKI）的强大之处，相比试图在多个设备之间保持一个核心的IT密码同步，机构可以使用一个更安全的启用了基于X.509标识（FVMI）的方法。相比使用PBKDF2将密码字符串转换为DEK，或使用PRK，FileVault使用了FVMI的非对称公私密钥对。公钥用于封装KEK，而私钥用于解包KEK。

恢复的方法

恢复的功能对于个人和组织来说至关重要。在不通过网络的情况下，用于恢复和孤立系统的资源会有所限制和局限，在某些情况下，这将决定何种部署战略对于所有相关方是最好的。在某个特定系统上，如果所有启用了FileVault的用户都忘记了密码，则凭据不可用，也没有可用的恢复密钥，加密卷无法解锁，数据无法访问。数据可能永久丢失，所以正确的恢复计划至关重要。

考量两种恢复方法可以横跨多个部署策略，但是为了简单起见，本文涵盖了恢复方法和部署策略之间最常见的映射关系。

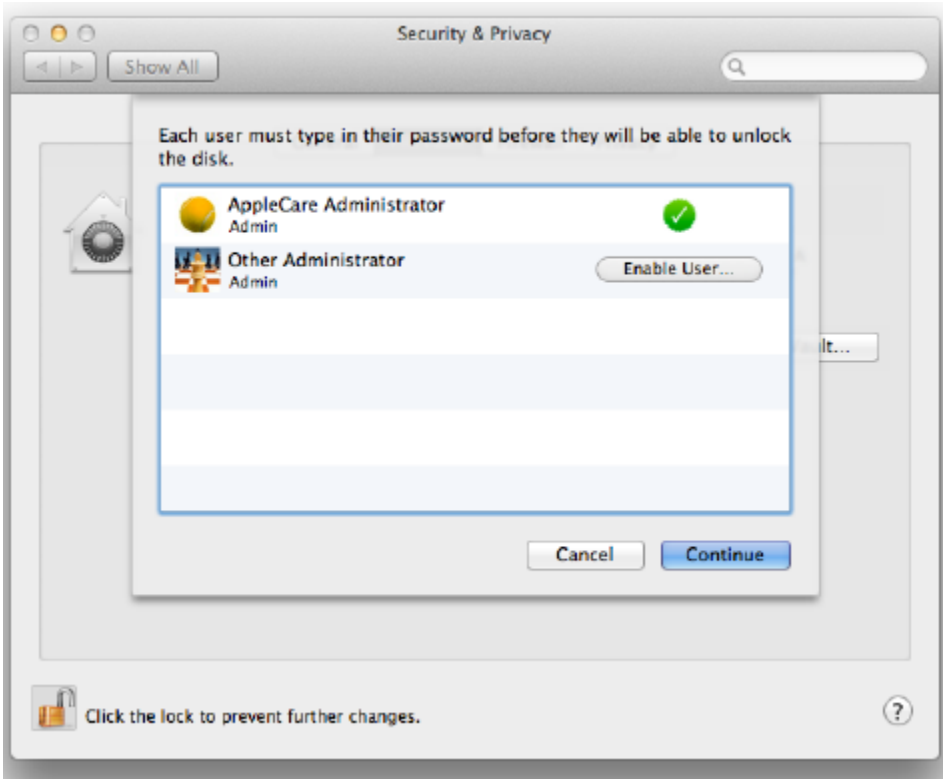
	**个人恢复** 为个人安全网生成和使用随机、对称的密钥值。
	**机构恢复** 为企业安全网和私钥托管使用基于X.509的非对称密钥对。

个人恢复

个人恢复提供了使用个人恢复密钥（PRK）的安全网。用户首先直接进入“安全与隐私”首选项中的“FileVault”面板。

点击“Turn On FileVault”后，如果Mac有多个用户帐号，管理员要识别出允许解锁加密驱动器的用户帐号(启动计算机或从睡眠或休眠状态恢复)。默认情况下，当前已登录用户将由一个绿色勾选标记自动标识为“已启用”。

启用 FileVault 需要特权以获得权限（system.preferences.security），该权限默认赋予本地管理员组成员。



如果所有用户帐户均已启用FileVault，“启用用户”按钮便不再出现在个人设置面板中，所有随后创建的帐户将自动启用FileVault。

只有在已启用FileVault 2的用户解锁了驱动器后，未启用FileVault解锁功能的用户才能登录该Mac。一旦解锁，对于所有用户来说，该驱动器将保持解锁和可用状态，直到计算机关闭或进入休眠状态。

为了每个帐户都有能力解锁卷，管理员需要输入密码，或者让用户输入他们的密码。

在使用户能够解锁磁盘后，将显示PRK。

如果选择“不使用苹果存储恢复密钥”，则必须取得24位的字母数字恢复密钥并另外留存。这意味着个人应该设置好系统，将密钥值写下来并安全地存储于计算机之外。如果个人或组织没有留存在启用时随机生成的恢复密钥，Apple将无法协助恢复，并且所有数据都将丢失。

这种方法没有财务成本也无需变更基础设施。在保护PRK时若采取了恰当的措施，这将是可以快速方便地为最终用户部署的安全网。但是，如果最终用户没有恰当地保护恢复密钥---例如，把它写在便笺纸上并贴在了办公桌上---那么恢复密钥会被轻易发现，而入侵者便得以访问。

如果用户选择“使用苹果存储恢复密钥”，密钥将被如何保护和检索，会在下一节讨论。

个人恢复密钥保护

为了保护 PRK 并允许以可重复的方法来检索和解包该密钥，则需要一个对称的 PRK 封装密钥。这个 PRK 封装密钥是通过对所有三个安全问题的回答进行哈希而派生得来的。因为这个密钥派生自对回答的哈希，所以必须提供确切的回答来检索 PRK。即便任何一个回答仅有稍许差异，都无法导出有效的对称密钥。由于问题的答案只有要启用 FileVault 的人才知道，因此 Apple 无法协助恢复 - 所有数据都将丢失。

从苹果检索个人恢复密钥

当用户忘记自己的帐户密码，或者IT员工必须访问受保护的卷而没有备用的已启用帐户时，则需要用到PRK而且能够从苹果检索到。

输入错误的登录密码三次，密码框下方将会显示这段消息：“如果忘记了密码，您可以使用恢复密钥重设密码。”用户必须点击消息旁边的三角形以显示恢复密钥框（这将取代密码框）和AppleCare的联络信息，以及计算机的序列号和记录号。记录号生成于当初将PRK发送给苹果之时。

要从苹果检索出PRK，请联系AppleCare并提供所需信息，包括序列号和记录号。之后，AppleCare代表就可以提供PRK了。成功检索出并输入恢复密钥后，系统将提示用户更改其登录密码。当重置登录密码时，还会提示用户创建新的登录密钥串。先前的登录密钥串，在默认情况下，被设置为与该帐户所使用的相同的密码。 重置帐户时，OS X并不知道和存储以前的密码，故而无法解锁之前的密钥串。只有密钥串中非密码保护的内容才能转换为新的密钥串。

更改登录密码后，还建议用户更改其FileVault恢复密钥，并将其上传到苹果。

更改恢复密钥

在FileVault窗格的“安全＆隐私”首选项中，点击“关闭 FileVault”以禁用FileVault。一旦关闭，FileVault便开始解密驱动器。解密完成后，就可点击“打开FileVault”按钮。 如此便可允许管理员启用具有解锁能力的用户、显示新的恢复密钥，并提供将此新密钥发送给苹果的选项。已发送给苹果的旧密钥将不会解锁新加密的磁盘。如果需要从苹果检索恢复密钥，则只能根据登录窗口中显示的序列号和记录号检索到最新的密钥。

机构恢复

机构恢复为企业IT部门提供了一个安全网络，并使用X.509的身份验证和非对称公私钥对来启用私钥托管。

机构恢复密匙

IRK是指由企业提供的基于X.509的身份验证，即FileVault主标识（FVMI）。FVMI由包含公钥的X.509证书及其应的私钥组成。FileVault使用身份验证的证书中的公钥封装（加密）KEK，并使用私钥解包（解密）KEK。

FileVault 主身份

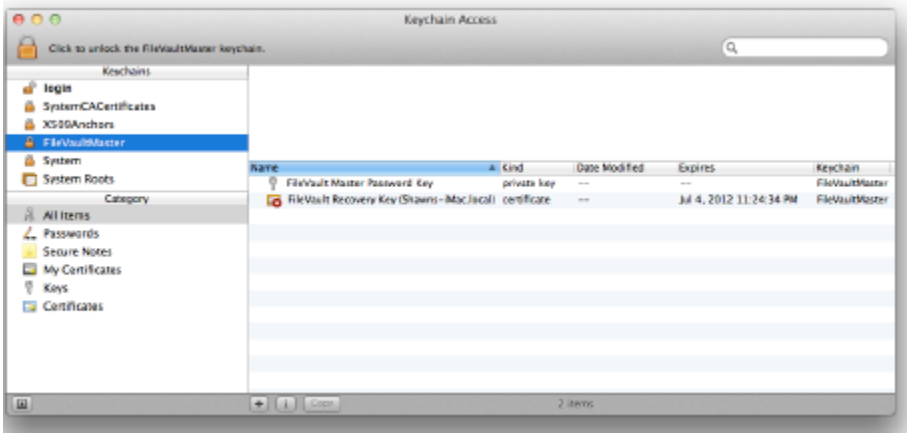
FVMI是一个对FileVault加密卷的IT访问的完整组件。该身份为系统管理员或IT人员提供了一种可替代方式来访问用户的加密卷，而无需知道任何已授权用户的密码。

每个主机只使用一个 FVMI。机构可以选择其 FVMI 配置的颗粒度。如果追求简单，可以为整个公司提供单个 FVMI，但是该方法意味着任何对此身份的损害都会将其余的系统暴露在某个恶意实体之前。为每台机器提供不同的 FVMI 可以减轻风险，但这会在管理 FVMIs 和单个机器之间的关联时增加复杂性。除了确定可以进行安全管理的颗粒度的最高级别，机构应采取何种颗粒度级别并没有明确的最佳实践。

FVMI由三个组件构成，他们共同代表了X.509数字身份：

- 自签名X.509根证书
- 公钥(嵌入于证书中)

- 私钥



默认情况下，FVMI是在一个由OS X生成并维护、名为FileVaultMaster的密钥串中得以保护，位置在 `/Library/Keychains/FileVaultMaster.keychain`.

这个特殊的密钥串是OS X管理的密钥串之一，它不会显示在密钥串列表中（可通过“Keychain Access”查看）。 但是，如果他们有兴趣查看其内容，用户或管理员可以手动将此密钥串添加到他们的密钥串列表。只需双击密钥串文件或选择“文件”>“添加密钥串”，导航到上述路径，然后点击“打开”将密钥串添加到活动列表。

查看密钥串中的可视化内容不需要知道密钥串保护凭证。 然而，对密钥串内容的任何操作（例如删除，导出私钥或替换身份）则需要知道 `FileVaultMaster.keychain` 的保护密码---最初由用户或管理员输入。因为使用受保护的FVMI可以访问一个完全加密的卷，所以保护该身份免受未经授权的访问和使用至关重要。

自签名 X.509 根证书

当管理员使用两个内置服务中的任一个时，会生成随机的X.509身份标识：“Users & Groups”中的“Set/Change Master Password”，或者CLI命令“security”。生成的身份证书是自签名的，这仅仅是因为FileVault 2被设计为对任何用户都有效，而无关环境。这意味着任何人，即使没有完整的公钥基础设施（PKI），都可以利用FileVault 2的功能。这种架构还可以使组织能够使用来自于自己企业认证机构（CA）的指定身份。

无论何种方法（由OS X直接提供或从企业CA提供），OS X都不需要为FileVault身份将信任路径验证到已信任的锚点。在身份是由OS X生成的情况下，则证书是自签名的CA根证书。对于企业CA颁发的证书而言，它可以是受信任的CA根证书、中间证书或叶证书。需要存在有正确配置了公私钥对的身份---长度为1K、2K或4K，并且记录了密钥的使用方法，但是对信任路径或撤销的验证则不需要。

证书的某些属性需要特定的值，而其他的属性可以设为任何值：

主题名称		
通用名称	FileVault恢复密钥	必要的
描述	MySystem.local	可选的

发行人姓名		
通用名称	FileVault恢复密钥	必要的
描述	MySystem.local	可选的
签名算法	SHA-1 with RSA Encryption (1.2.840.113549.1.1.5)	必要的

通用名称设置为“FileVault恢复密钥”。需要且必须准确，这样才能被FileVault正确识别和使用。如果FVMI是由组织自己的CA提供，则要特别注意此属性是否正确设置。

描述，默认设置为“MySystem.local”。当在FVMK上设定主密码时，它由“System Preferences”>“Sharing”中设置的计算机名生成。描述字段不需要特定的值，但是如果身份是由组织自身的CA签发，则某些组织可能希望将该属性设置为主机的完全限定域名（FQDN）或用户名，以便在证书中进行备用标识。

此OS X生成的证书是自签名根证书。 主题名称和发行者名称都设置为FileVault恢复密钥，这是FileVault在OS X上使用CA的特殊用例。在此证书上未做证书撤销检查，因此不需担心证书的到期日期，也不需要将此证书显式地设置为受信的---此过程本身就是可信的。

公钥

公钥是用于加密/封装KEK的非对称(公共/私有)密钥对的前半部分。顾名思义，它可以公开访问而无需任何保护措施。公钥内嵌在X.509证书中，不需要做为一个单独的对象来提取或者存储。

公钥信息	
算法	RSA 加密 (1.2.840.113549.1.1.1)
密钥用途	加密，验证，包装
密钥尺寸	1024, 2048, or 4096; 默认 = 1024

FileVault恢复密钥身份的实际公钥必须设置为加密，验证，封装。如果企业选择替换系统生成的身份，那么相应的公钥信息需要定义好其密钥用途，至少是和之前具有相同的扩展。

扩展密钥使用（2.5.29.15）	
重要的	No
用途	数字签名，密钥加密，数据加密，密钥证书签名

如同FileVault恢复密钥身份的公钥所描述的，扩展的密钥用途必须为四个已知的功能而定义：数字签名，密钥加密，数据加密，密钥证书签名。

扩展 - 基本约束（2.5.29.19）	
重要的	Yes
证书颁发机构	Yes

因为这是一个自签名的根证书，它必须被标记为重要并作为证书颁发机构。

私钥

私钥是非对称（公共/私有）密钥对中的重要的另一半，用于解密(解包)密钥加密密钥(KEK)。它将保持私有状态，并且随时受到保护，否则

FileVault2的保护会受到危害。安全地恢复加密卷和对该FVMI进行恰当管理的能力，对于保护系统免受未授权的访问至关重要。

对基于X.509身份的恰当的保护和使用，容易被对OS X控制和FileVault架构不完全熟悉的个人所误解。所以，对于OS X用以创建、使用和管理FVMI的流程有很强的理解至关重要。

主密码

主密码简单来讲就是指在FileVault主密钥链上设置的密码。OS X密钥链密码用来加密（封装）存储在密钥链内的私有数据。主密码用来解锁和解密（解包）FVMI的私钥组件。相应的证书是公共信息，不需要保护。

该主密码没有直接指向FileVault主密钥链的内容或实际的KEK保护，而是简单地解锁对私钥的访问，如果它是存储在密钥链中的话。密码可以设置为组织所需的任何有效密码。即使对于所有部署的系统而言，已部署和受保护的FVMI是相同的，它也经常被设置为单个机器/用户唯一的。

主密码轮换

如果选择在密钥链内保持完整的FVMI，那么定期重置FileVaultMaster.keychain或许是组织感兴趣的。在这种情况下，主密码的定期轮换将会有助于减小对实体危害的风险。如果一个FVMI被破坏，并且未经授权的个人能够同时访问证书和私钥，又如果有用户已启用了FileVault并且原先就与该FVMI有关联，则此人将可以解锁对任何受保护卷的访问而无需该授权用户的凭证。

固件

待机模式

注意：在机器睡眠了特定时间后，待机模式使得内核电源管理自动使机器休眠。这使得在睡眠期间节约了电力。对于受支持的硬件来说这个设置是默认开启的。如果这个机器支持此特性，那么设定待机模式对 pmset –g 来说是可见的。只有休眠开启了模式 3 或 25，待机模式才会工作。

所有计算机都具有某些类型的固件---EFI，BIOS---以帮助发现硬件组件，并最终使用所需的OS实例正确引导计算机。若是Apple硬件，并且使用了EFI，Apple将储存EFI中的相关信息，以对OS X的功能提供帮助。例如，FileVault密钥存储在EFI中，以透明地方式退出待机模式。

对高攻击环境特别敏感或者当设备处于待机模式时，面临潜在的完全设备访问（风险）的组织，应通过销毁固件中的FileVault密钥来减轻缓风险。这样做不会破坏FileVault的使用，仅仅需要用户输入密码即可使系统退出待机模式。

通过使用pmset命令设置特定的电源管理环境变量，可以实现进入待机模式时销毁FileVault密钥。在目标系统上以交互方式执行以下命令，或在执行自动化脚本或部署期间，设置要销毁的密钥：

```
# pmset destroyfvkeyonstandby 1
```

将相同的变量设为0会导致在进入待机模式时留存FileVault密钥。

通过执行以下命令，可以看到pmset命令destroyfvkeyonstandby以及其他可用变量的完整描述：

```
# man pmset
```

destroyfvkeyonstandby的描述如下：

```
destroyfvkeyonstandby - Destroy FileVault Key
when going to standby mode. By default
FileVault keys are retained even when system
goes to standby. If the keys are destroyed,
user will be prompted to enter the password
while coming out of standby mode.(value: 1 -
Destroy, 0 - Retain)
```

只需执行以下命令，便可确定destroyfvkeyonstandby或任何其他环境变量的当前状态：

```
# pmset -g
```

此命令的结果类似于以下内容：

```
bash-3.2# pmset -g
System-wide power settings:
 DestroyFVKeyOnStandby          0
Active Profiles:
Battery Power           -1
AC Power                -1*
Currently in use:
 standbydelay          4200
 standby               0
 womp                  1
 halfdim               1
 hibernatefile         /var/vm/sleepimage
 gpuswitch             2
 sms                   1
 networkoversleep      0
 disksleep             10
 sleep                 0
 hibernatemode         3
 ttyskeepawake         1
 displaysleep          60
 acwake                0
 lidwake               1
```

固件密码

OS X支持在特定系统上使用密码来锁定当前的固件设置并防止在对固件的意外修改。这并非为了修改已经安装在硬盘驱动器上的内容。此固件密码最常用于防止严格控制的用户从备用系统卷引导，或阻止使用其他“catch键”来更改引导过程的流程（例如引导到单用户模式）。这个密码也阻止未授权的用户，在组织中不允许的情况下，引导进入隐藏的恢复分区。固件密码还可以阻止通过比如FireWire等接口直接内存访问（DMA）。目标磁盘模式需要为DMA，因此固件密码也将阻止其在系统上的使用。

知道固件密码的授权人员可以在系统引导时按住键盘上的Option键，当提示时输入密码，并使用系统的“Boot Picker”界面选择备用引导卷。

这个动作允许个人对原始引导卷进行各种修改（比如：运行磁盘修复工具、甚至为了取证目的而解锁访问加密卷）。

固件密码是跟随主机计算机的，因此将加密的驱动器从一个系统转移到另一个，并不会随之带来固件密码。但是，如果个人尝试使用“目标磁盘模式”从其他计算机挂载卷，则从目标系统挂载卷之前需要输入固件密码。

启动区

Boot Camp是Apple的技术，用以在被支持的Apple硬件上使用Microsoft Windows的本地执行。在每个新的Mac上，使用内置的被称为Boot Camp的实用程序，用户可以从磁盘分区直接安装并以本机速度运行Windows。安装程序对于Mac文件是简单和安全的，因为它运行在与OS X引导分区完全独立的

分区。使用Boot Camp安装Windows后，用户可以用OS X或Windows启动他们的Mac。

Microsoft Windows和第三方的Windows FDE解决方案无法“理解”和使用CoreStorage管理的卷。这样就可以防止利用Boot Camp和Windows在OS X上使用FileVault2 FDE的技术。

取而代之的是，任何同时运行多个操作系统的组织(例如任何Apple硬件上的OS X和Windows)都可以使用虚拟化技术，使用VMware或者Parallels软件安装Windows。

有关Boot Camp和在Apple硬件上运行Windows的更多信息，请访问OS X兼容性网页和Boot Camp支持页面：

- http://www.apple.com/macosx/what-is/compatibility.html
- http://www.apple.com/support/bootcamp/

双因素认证

如前所述，当使用FileVault 2（FDE）时，初始验证作为EFI预引导验证过程的一部分进行。在引导阶段的这个早期步骤中，没有任何依赖操作系统的服务被加载，因为他们依靠操作系统的运行。这意味着此时除了基于密码的验证，其它可选的验证机制都不被支持。

对额外的双因素验证机制的任何支持，比如：智能卡或一次性密码（OTP），都需要在高度受限的空间和对EFI的执行中进一步地开发这些服务。如果组织需要使用智能卡来验证和解锁对加密存储的访问，则应该更仔细地检查使用基于容器的Legacy FileVault。

更多关于Legacy FileVault及其对智能卡的支持的信息可以通过搜索 http://www.apple.com/support.

附录 B: FileVault 2 流程

用户访问流程

1. 用户启动设备
2. EFI
    2.1. 从引导卷加载已授权的FDE用户信息
    2.2. 基于EFI的登录用图标和名称显示已授权用户
    2.3. 当用户选中时，需要密码认证
3. 用户
    3.1. 输入帐号密码
4. FileVault
    4.1. 使用PBKDF2将密码转换为密钥
        4.1.1. PBKDF2-基于RSA密码的密钥派生功能
    4.2. 验证密码：通过尝试解锁从CoreStorage元数据中检索出的用户包中发现的KEK密钥
    4.3. 操作成功意味着KEK密钥的成功解包
5. OS内核被加载
6. 引导参数存储于AppleKeyStore，以供后期内核检索
    6.1. 用获取的KEK解包VEK
    6.2. 令牌标识已验证用户
    6.3. 用户密码
7. 控制权移交给内核
8. 解锁卷
    8.1. Apple_CoreStorage分区和AppleKeyStore引用
    8.2. 用于解锁Apple_CoreStorage分区的VEK（启动卷）
9. 操作系统引导
    9.1. 系统照常发现根卷
    9.2. 登录
        9.2.1. 用在预引导时用户输入的密码尝试密码转发
            9.2.1.1. 启用密码转发。通过“builtin:forward-login, privileged”的验证机制、被授权数据库(/etc/authorization)引用的特权、在此权限下：system.login.console
        9.2.2. 目录服务验证用户/密码
        9.2.3. 用户被赋权访问并显示个人桌面

附录 C: 额外资源

相关知识库文章

在 Apple 支持网站上，有各种关于 FileVault 2 的设置和使用的知识库文章。访问 support.apple.com 以学习更多相关主题，诸如：

- OS X Lion: About FileVault 2
- OS X Lion: Using FileVault 2 and Lion Recovery
- OS X Recovery Disk Assistant v1.0
- MacBook Air: Recovering a lost EFI firmware password
- Mac OS X: How to start up in single-user or verbose mode

相关的网页

- OS X Recovery restores your Mac with a few clicks

安全配置指南

苹果为增强 Mac 的安全性提供了额外的最佳实践，并配有详细指导，其结果来自于多年来与美国国家安全局(NSA)、美国国家标准与技术研究院(NIST)和美国国防部信息系统局(DISA)等全球敬仰的安全组织的合作，关于苹果产品最新的指导，请访问 www.apple.com/support/security/guides.

培训和认证

苹果授权培训中心为有兴趣在其企业环境中规划、维护和集成 OS X，OS X Server 和其他 Apple 解决方案的专业人士提供了种类繁多的 IT 培训与认证机会。讲师指导课程提供演示和讲座，并结合实践、真实的实验室和练习，为 IT 专业人员提供最全面的培训。要了解更多有关苹果培训和认证的信息，请访问 http://training.apple.com/certification.