在传统的基于浏览器的密码管理系统中，用户从浏览器缓存中获取完整的密码，并将改密码和网页端服务器存储的密码进行比较，如果比较成功则登录成功。而在我们设计的密码管理系统中，由于在两阶段都使用了秘密共享方案，所以用户从浏览器中提取一份秘密份额，并和本地的秘密份额进行组合从而恢复完整的密码；而对于网站服务器集群，他们也会根据自己所使用的秘密共享方案恢复出一份密码与用户和浏览器端恢复出的密码进行比较。在演示流程中，我们将ABY3-VSSSS协议融入了一个由我们搭建的CET6报名网站以证明我们系统极强的可用性，并且假设网站服务器集群有5台服务器。

（查看日志）通过日志我们可以观察到，系统已经为我们注册了四个用户的信息，并且完成了用户-浏览器阶段的集群启动以及秘密共享 和 网站服务器阶段的集群启动和秘密共享，下面我们手动注册一个账户来感受一下这个流程。（手动注册）

我们设计的密码管理方案在不破坏传统密码管理系统便捷性的同时增加了系统的安全性。我们在登录用户1192时进行自动密码填充（演示），（看日志）从日志我们可以看出自动填充的密码由浏览器-用户两方的秘密份额恢复而成。当然我们也可以手动输入密码进行登录（手动填充）。

正如我们在作品介绍中所介绍的那样，我们的系统具有很好的鲁棒性，当网站服务器集群遭到破坏时，我们可以自动的检测出错误并完成自我修复和秘密份额的全自动重新划分，下面我们来对用户1192进行模拟攻击。（模拟攻击-查看日志-登录失败-查看日志）我们再次查看日志，日志很完整的记录了错误检测以及错误修复的过程。我们使用改进的排列组合方法来排查被破坏的服务器，我们只需要找到一组可以成功恢复密码的浏览器组合，剩余工作就可以在线性时间复杂度内完成，考虑到敌手很难在短时间内控制多台服务器，所以我们的方法效率可以得到充分的保证。那么此时我们可以观察到我们的方法不仅检测到了被破坏的服务器，还正确的恢复了密码（再次登录证明）

我们设计的密码管理方案也充分继承了传统密码管理系统丰富的功能，支持更改密码（演示）和切换设备信任模式（演示）

同样的，我们的方案具有极强的可扩展性，考虑到不同应用场景下对秘密共享方案的需求不同，我们还集成了像ASS这样可以自主动态刷新秘密份额的共享方案以满足不同客户的多样化需求。（转换-演示秘密份额刷新）

更多的，我们也考虑到了对安全性要求更加苛刻的环境：例如浏览器和第三方共谋的情况下，我们也推出了cheetah协议进行（2，2）秘密共享，以及SEMI2K-SPDZ协议能够在协议运行过程中检测到恶意方并停止运行，以满足网站对更高安全性的要求。我们设计的方案可以在这些秘密共享方案间相互切换。

所以说，首先我们提出将秘密共享方案融入两阶段的密码管理流程以提供对用户密码更加安全的保护，同样，我们的方案具有很强的鲁棒型和可扩展性，可以在网站服务器自动检测被攻击的服务器并完成正确密码的恢复，并且用户可以根据需求在多个秘密共享方案中相互切换。当然在这里我们也要对阿里巴巴集团提供像scretflow一样国产自主研发的隐私保护开发框架，为我们提供了良好的开发环境。谢谢各位老师，我们的展示就到这里。























































