大家好，本次分享的报告主题为一种抗投毒攻击的隐私增强联邦学习框架PEFL。

主要分别三个部分进行讲解，第一是联邦学习中的安全问题概述，第二是相关知识概述，第三是问题的解决方案

首先，联邦学习是一种分布式机器学习框架，与传统的集中式机器学习框架相比，联邦学习可以与众多参与方一起来联合训练模型，同时各参与方不用暴露自己的原始数据。本质上是承诺多方通过交换梯度而不是原始数据来联合训练模型。

尽管联邦学习是一种很有价值的机器学习框架，但是它依然存在着许多安全问题。比如梯度隐私泄露，投毒攻击的威胁等等。在其他学者的研究报告中已经证实，半诚实的服务器可以通过接收到的梯度参数恢复出目标用户的一些敏感信息，诸如社交头像，个人医疗信息等。而投毒者可以通过提供无效梯度使得模型无效化，在这个投毒过程中还可以窃取用户敏感信息。

由此引发出一个可信可靠的联邦学习模型需要关注的两个基本问题，一是…，二是…

针对这两个问题，本文做了以下相关工作，

提出了一种新的隐私增强联邦学习框架

提出了一种新颖的自适应抗毒攻击聚合方案

进一步提出了一种对数函数来去除恶意梯度

提供了全面的安全分析，并证明了该方案的收敛性。

接下来来了解一下解决方案中涉及到的一些知识要点。

投毒攻击是什么，简单来说是指攻击者意图影响训练模型的训练效果，使得应当是训练模型的源类数据错误的分类到攻击者的目标类。从之前的研究来看，单个投毒者可以控制整个训练过程，并损害用户的数据隐私。

那么投毒攻击有哪几种方法呢？在本次报告里主要有ppt上所示的两种方法，左下角标签反转攻击，右上角是后门攻击。

标签反转攻击通过提供恶意的梯度参数来影响训练效果，达到模型错误分类的这样一个目的。

后门攻击则是攻击者在提交梯度参数时，在提交的梯度参数中添加特定标记作为触发器，以达到模型一旦遇到这种特定标记时就触发错误分类，而没有遇到这种特定标记时就正确分类的效果。

最后，介绍具体解决方案。

整体框架由一个SP（Service provider）服务提供商，一个CP（Cloud platform）云端平台，一个KGC（Key Generation Center）密钥生成中心，以及联合训练用户集(data owners)。

（1）KGC 是独立且受信任的机构，它分发和管理所有公钥和私钥 (pk, sk)。

（2）data owners：数据所有者，也就是联合训练用户，我们假设所有数据所有者持有的数据是独立的同分布的，这与许多先前的研究相同。

（3）SP：负责接收用户提交的所有梯度并将它们聚合以获得优化的全局模型。同时，需要在云平台的帮助下检测潜在恶意用户发起的中毒攻击。

（4）CP：提供按使用付费的服务。它与 SP 一起工作以执行本文中的计算。此外，CP 持有由可信权威（即本文中的 KGC）生成的私有公钥对 (pk c, sk c)，可用于加密数据或解密密文。

其中要注意的是，整个方案实现的前提有一些假设条件，这些条件包括

所有的训练数据是独立同分布的

恶意用户的数量不超过用户总数量的一半

恶意用户的数据集与诚实用户的数据集来自同一分布，且相互独立

由于SP和CP可以访问所有用户的局部梯度，因此假设SP和CP是半诚实的，即他们SP和CP在执行所有操作时都是诚实的，但他们会尽可能的根据用户局部梯度去掌握更多的敏感信息

四个实体之间没有共谋（KGC，SP，CP，用户）

如何构建PEFL呢，

首先KGC给每个用户以及CP发放独立的公私钥对，以及SP初始化训练设置下发全局参数

接着开始我们的联合训练，这个训练过程可能有h轮才收敛，那么我们假设现在是第t轮训练。

对于所有用户，每一个用户接收自己的公私钥对后使用自己的私钥解密SP发来的全局模型参数，解密后在本地进行训练获得局部梯度向量。

其中用户本地训练时使用动量的SGD梯度下降算法来平滑更新梯度，其中γ是指数衰减因子，将之前训练轮次的梯度依次添加到本轮次的训练梯度更新中。这种动量的SGD算法已经得到证实可以加速收敛，还可以降低方差范数比，有利于提高模型的鲁棒性。PEFL框架继承了这一优势。

本地训练后，还要加密保护要提交的局部梯度，同时考虑到加密原语的基本要求，梯度向量的每一维应编码为整数形式，如图所示，其中prec是precision精度值，方括号代表取整数的函数，公式的意思就是将数据每一维左移prec位数后取最近的整数。

现在打包好了局部梯度以后，就要考虑给SP发送数据了，那么为了减少通信成本，可以利用将多个明文打包成一个密文的密文打包技术来提高效率。

这种技术指的是假设 ppt

其中 0 表示填充位的零填充，以防止在密文添加中溢出。此外，在⌈p/d⌉ =⌊p/d⌋+1条件下，d =(⌊log2 p)/(prec+ pad)⌋。为了简单起见，使用以下[G xi表示2prec·G xi]0。U t x最终将打包密文 [[G x ]]pkc = {[[G xi · · · G x(i+d)]]pkc }i=p/d i=1 发送给 SP。