研讨纪要

# 研读论文

## 杨雨嫣

## ABY3: A Mixed Protocol Framework for Machine Learning

机器学习需要大量的数据支持，因此涉及到数据的隐私保护问题。本文基于三服务器模型（即数据所有者在三台服务器之间秘密共享数据，三台服务器使用3PC算法在联合数据上训练模型），设计出了一个新的完整的框架（ABY3），可以有效地在算术计算、二进制和3PC之间来回转换，并将其应用到训练线性回归、对率回归和神经网络模型中。另外还提出了新的技术，用于扩展超出三方情况的共享十进制，以及用于评估分段多项式函数的定制协议。

这个方向主要面临的两个挑战：第一个是机器学习中训练的数据和中间参数都是十进制的，与3PC算法的计算不一样，不能使用模算数处理；第二个是大多数机器学习过程都需要在算术运算（如乘法和加法）和非算术运算（如激活函数（逻辑回归函数）和分段多项式（RELU））之间来回切换，前者使用算术秘密共享可以有效地实例化，而后者应使用二进制秘密共享或者Yao sharing才能实现。在不同共享类型之间进行转换的标准方法成本很高，并且也是主要的性能瓶颈。

本文的创新点在于：

1. 在半诚实半恶意的情况下，将新的近似于定点乘法的协议用于共享十进制数，其成本接近于标准的秘密共享的模乘法。
2. 本文的新的通用框架，可以在三方环境中有效地在二进制共享，算术共享和Yao sharing之间进行转换。
3. 其他优化包括延迟重共享技术，该技术将向量化操作的通信复杂度降低了几个数量级。
4. 实践进行线性、逻辑回归和神经网络的训练和推理，速度都有很大的提升。
5. 在半诚实半恶意的环境中实例化所有构建的模块，可以对抗恶意对手。
6. **Voting: You Can’t Have Privacy without Individual Verifiability**

电子投票系统中主要有两个安全目标：个人隐私（没有人知道我是怎么投票的）和可验证性。可验证性包括个人可验证性（选民可以检查他的选票是否已清点）、通用可验证性（任何人都可以检查结果是否与发布的选票相对应）、资格可验证性（仅合法选民可以投票）。本文的观点是隐私意味着个人的可验证性，也就是说，在相同的信任假设条件下，没有个人可验证性的系统无法获得隐私。并为了证明其一般性，在密码模型和符号模型中显示了这个结果。

## 倪欣

1. **On Ends-to-Ends Encryption: Asynchronous Group Messaging with Strong Security Guarantees**

展示了在现实的异步组消息传递系统中可以实现折衷后的安全性，提出了一种称为异步棘轮树（ART）的设计，该设计使用基于树的Diffie-Hellman密钥交换来允许一组用户导出共享的对称密钥（即使没有两个用户同时在线）。ART扩展到包含成千上万个成员的组，同时仍提供可证明的安全保证。该工作形成了两个RFC草案以及IETF的MLS工作组的基础，有助于改善目前邮件系统的安全性。

1. **Fast Secure Multiparty ECDSA with Practical Distributed Key Generation and Applications to Cryptocurrency Custody∗**

ECDSA是一种标准化的签名算法，广泛用于TLS，代码签名，加密货币等领域。由于其重要性，以分布式方式安全地计算ECDSA（称为阈值签名）的问题已引起人们的极大关注（这意味着n个t-out中的任何一方都可以签名，保留任何t − 1个或更少的损坏方的安全性，并且t≤n可以是任何值，因此可以支持具有诚​​实密钥分配的诚实少数群体）。该文提出了第一个真正实用的全阈值ECDSA签名协议，该协议同时具有快速签名和快速密钥分发功能。

## 章雪琦

论文方向为机器学习与去匿名化相结合，去匿名化是指匿名数据和其他数据来源相互对照来重新识别匿名的数据来源。区别一个数据来源和另一个数据来源的任意信息都能用于去匿名化。

1. **Large-Scale and Language-Oblivious Code Authorship Identification**

论文为大规模且不受语言限制的代码作者身份识别的研究，有效提取代码作者属性是成功识别代码的关键，这一点上有以下几个难点：编程语言不同、每个作者可用的代码样本数量有限、每个文件的平均代码行存在差异。

论文提出的是基于深度学习的代码作者身份识别系统。由于代码作者希望保持匿名，则身份识别成为匿名性问题。这一研究的挑战包括：程序员的代码风格与其教育、经历、软件环境等有关；编程风格也因管理人员、工具、语嫣、时间、不同语言之间存在不同；有时源代码不易获取。

本文提出的系统分为三个阶段：

①预处理：将代码预处理并生成初始表示，使用的TF-IDF技术，这是一种用于信息检索与数据挖掘的常用的加权技术，词频、逆词频的计算。

②学习表示：有三个RNN、三个全连接层组成，从不够确定的作者身份学习，生成深度表示。

③分类。将训练结果，构建一颗随机森林分类树。

该方法既可以利用深度学习的良好的特征提取功能，又可以利用RFC的大规模分类功能。目前存在的问题：许多代码是由多个作者共同完成的，本文方案假设了全部由单个程序员完成，忽略了合作完成的、大项目的情况；存在作者身份混乱攻击，来使样本风格混乱；未考虑可用样本代码的最小平均代码行。

1. **Fraud De-Anonymization for Fun and Profit**

论文为为了娱乐和利益而进行的欺诈去匿名化。主要解决在一些对等意见系统中可能存在的搜索排名欺诈的问题。所以研究论欺诈去匿名化方法去降低搜索排名欺诈。

它首先建模成为最大似然估计，引入UODA，一个无约束优化解决方案，这是由现实的评论发布行为生成模型，计算每个工人生成给定的可疑欺诈历史记录的可能性，找最大的情况。

论文提出了一种基于图的深度学习解决方案，来预测同一欺诈者对帐户对的所有权，并利用这个建立DDA歧视性的去匿名化算法和PFD假命欺诈者发现算法。DDA是训练一个分类器将欺诈用户账户归于可能控制的工作人员。PFD将不能归于已知工作人员的欺诈性用户进行聚类，即归为可能的未发现的工作人员。

## 4. 张敏

**(1) Yet Another Text Captcha Solver:A Generative Adversarial Network Based Approach**

本文解决的主要问题:

基于生成对抗网络方法，构建了一个合成验证码的模型和一个消除验证码干扰特征的模型，利用CNN（扩展的LeNet-5方法）识别消除干扰特征的验证码，最终实现验证码识别准确率的较大提升。

创新点：

a. 利用生成对抗网络自动生成大量的合成验证码，用于训练识别器,不需要大量人工标记的数据集，降低了成本

b. 利用GAN的方法生成了预训练模型，能够去除验证码上的安全特性，这也是此论文能够吊打其他论文的关键点

c. 用一个较小的真实验证码数据集（500张验证码），通过迁移学习微调基础识别器参数,节省了算法的运行时间

(2) Model-Reuse Attacks on Deep Learning Systems

本文解决的主要问题：

作者的经验研究显示，截至2016年，GitHub上的超过13.7％ML系统使用至少一种流行的原始模型。从好的方面来看，这种“即插即用”的方式极大地简化和加快了机器学习系统的开发周期。不利的一面是，大多数原始模型是由第三者方提供，他们缺乏标准化或规范性，会带来深远的安全隐患。这种潜在有害的原始模型对ML系统的安全性构成了巨大威胁，作者提出了一种模型重用攻击，能够使得主机系统对目标输入以高度可预测的方式做出错误的判断（例如，分类器误分类为特定类别）。

实现过程：

a. 生成语义邻居。对于给定的x−（x +），我们首先生成一组邻居X-（X +），通过向x-（x +）添加有意义的变化（例如自然噪声和模糊），使他们在语义上类似。根据噪声对x−（x +）分类的重要性来仔细调整注入到x−（x +）各个部分的噪声。

b. 寻找显著特征。语义邻居X-（X +）倾向于与x-归为同一类（x +）。换句话说，以分类器的角度，X-（X +）共享相似的特征向量。通过比较X-（X +）中输入的特征向量，确定对于x-（x +）的分类至关重要的显着特征集Ix-（Ix +）。

c. 训练对抗模型。对特征提取器f进行反向传播训练。

## 5. 赵愉悦

**(1) DeepMem: Learning Graph Neural Network Models for Fast and Robust Memory Forensic Analysis**

论文的中心思想是围绕内存取证----kernel内核数据结构检测来展开. 内存取证相对于硬盘取证的优势是恶意软件一旦启动, 装载进内存中才能实现操作.而磁盘可被攻击者隐藏, 擦除记录, 但是软件必须在内存中才能运行.

传统的内存取证技术有如下的限制:

现有方法:

1). 扫描数据结构:

  但是C/C++中数据结构不完整且模糊,  所以KOP.et al. 提出了 points-to analysis来完善数据结构. 但是仍然依赖于指针.

2). 签名扫描: 不依赖于指针, 但是效率很差, 因为在整个北村快照中, 一种对象就要使用一次签名.

Dolan提出了fuzz高边数据结构字段来消除签名受攻击的约束. --假阳性增加.

3). 重用内存快照:   效率在中间, 比签名扫描好.

上述三种都依赖使用人员的专业知识, 因此本文提出降低使用者门槛, 深度学习.

其中DeepMem的主要工作分为三部分:

1. 首先将内存转换成graph\_embedding的vector形式;
2. 第一个network -> 将vector打标签, 为T\_16\_24的形式, 并且利用其他现有工具为内存数据打好的标签, 签是T\_16\_24这样的形式, 分别代表object, offset, length;
3. 第二个Model -> 对打好标签的属于同一object的nodes进行投票, 表示这一段是什么内存对象。

**当前论文得出的一点经验:**

Graph Embedding Network广泛用于软件及内存的分类预测当中, 还包括<Neural Network-based Graph Embedding for Cross-Platform Binary Code Similarity Detection>, 检测软件相似度, 以软件的函数为粒度单元.

<Discriminative Embeddings of Latent Variable Models for Structured Data>, 对结构化的数据进行图嵌入等.

可以看出, 在安全领域---至少是网络以及数据结构等能够抽象成有向图的范畴内, 图嵌入是一个非常高效的工具进行研究.

## 6. 杨英光

**(1) LEMNA: Explaining Deep Learning based Security Applications**

1) 当前论文引出的问题

深度学习中的神经网络模型一般会比较复杂，其神经元数量提高的同时增加了其对测试集中数据测试的准确性，同时也降低了模型的可解释性，因为其缺少了一定的解释性，并成为了关键性的阻碍。造成安全从业人员犹豫把深度学习使用在安全领域之中，从而导致了深度学习在安全领域的运用率不高。

2) 解决方案

提出了一个解决方案LEMNA，来解决针对当前RNN在安全领域的不可解释性这一难题

**(2) Effective Program Debloating via Reinforcement Learning**

1) 当前论文引出的问题

由于当前环境下，软件规模急剧增长，降低了程序的性能，以及增加了攻击面和安全隐患，并且由于当前的软件工程实践中将大量代码一刀切的进行代码服用，并将其打包到可以复用的源码中，攻击着可以不通过插入源码就可以调用原有的一些代码来执行，造成了安全风险。所以很多研究针对软件规模的增长进行研究，用于简化程序。

1. 解决方案

使用强化学习为工具，提出一个CHISEI方法，能够对软件进行有效对简化。并提出了五个评价标准来评判CHISEI方法。

## 7. 唐玮

选择的两篇论文均是研究与SDN相关的安全问题，并实现了原形系统，前者针对传统工具难以在SDN上进行安全问题的取证和诊断实现了SDN上的取证和诊断工具，后者利用SDN和NFV实现了弹性的虚拟化网络入侵检测系统。

1. **Towards Fine-grained Network Se- curity Forensics and Diagnosis in the SDN Era**

传统的网络问题取证和诊断工具主要运行在Host层和Network层，在控制与数据解耦合的SDN中，传统工具就难以实现有用高效的网络问题取证和诊断了，文章提出流级别(Flow-level)的取证和诊断工具Forenguard，其探测并记录涉及 SDN 控制平面和数据平面的运行时活动及其依赖关系。例如网络断开连接，这个问题可能是由于网络遭受工具造成的，所以这时 Forenguard 就会 backtrack，根据因果关系查找在控制层和网络层中的前序活动，从而找到发生网络断开连接的根本问题。

文章的主要论述点有：

1) 构建一个可以表示 SDN framework 执行的活动类别并且可以帮助诊断的简洁集合，通过对数据层的状态转换以及控制层的执行建模，根据模型从而简明地说明这些转发行为发生的原因，提供易于阅读的信息来进行诊断;

2) 构建不同活动间的因果关系，文章设计了一个混合分析方法，结合静态分析和动态剖析来跟踪SDN架构里的信息流 (information flows)。特别的，文章静态预处理了controller/apps，然后使用运行时日志数据来重建控制层的面向事件的执行跟踪 (events-oriented excution traces) 和数据层的的状态转换图 (state trainsition graphs)；

3) 构建了一个函数形模块，其输入是转发问题的描述，输出相关的可疑活动，从而做到自动高效地在诊断数据中查询和定位可疑活动。

1. **vNIDS: Towards Elastic Security with Safe and Efficient Virtualization of Network Intrusion Detection Sys- tems**

文章主要基于SDN和NFV实现了一个虚拟化网络侵入检测系统（viusal Network Intrusion Detection System），这样的NIDS可以弹性扩展，以应对攻击流量变化。传统的NIDS在处理能力和部署能力上不够灵活，不能灵活处理严重的过载攻击，不能很好的支持虚拟化环境，因为虚拟化网络的边境模糊且易变，其次虚拟环境上的App可能会经常在不同的物理机上移动。

文章的主要论述了如何实现一个弹性扩展的vNIDS，其主要困难与解决方法为：

1) NFV 使得 NIDS 可以作为运行在多个位置上的多个实例，SDN可以动态分配不同实例的网络流量，因为有的检测逻辑需要对流进行分析，所以 NIDS的实例必须在彼此之间共享检测状态才能保证效率。通过将检测状态氛围 global 和 local 两种，提供一个 guideline 来对检测状态进行分类，部署静态程序分析和 NIDS 领域特定的知识发现来自动实现检测状态分类；

2) 实现非整体化的结构，从而实现灵活的扩展。文章将 NIDSes 划分为三个微服务:header-based detection , protocol parse , payload-based detection，每个微服务都可以独立 instantiated，在不同的位置运行不同个数的实例，执行不同的检测程序。

# 本周计划

1. **杨雨嫣**

* 想精度一篇关于生成对抗网络的应用的顶会论文；
* 再详细总结一下之前看的第一篇的机器学习中的隐私保护问题，以及其相关基础知识；
* 完成课程作业。

1. **倪欣**

* 选择一篇论文精读；
* 根据课程安排继续学习机器学习。

1. **章雪琦**

* 精读一篇论文；
* 跟着每周机器学习课程完成对机器学习基础的学习。

1. **张敏**

* 精读一篇论文；
* 继续学习深度学习的课程。

1. **赵愉悦**

* 完成基本的课程作业；
* 泛看一下graph embedding network。

1. **杨英光**

* 完成课程安排的有关机器学习编程的作业；
* 精读一篇论文。

1. **唐玮**

* 根据课程安排学习深度学习；
* 阅读Tiresias: Predicting Security Events Through Deep Learning。

# 三、遇到的问题

* 看论文遇到不懂的问题是否应该深入下去，方向太广了而顶会论文做的方向太深入，泛读论文很多地方看不懂，精度又需要更长的时间；
* 每个人阅读的论文的研究方向不同，在讨论时的信息量很大但彼此之间没有交叉，致使在研讨会上分享时大家的学习效果欠佳；
* 在分享论文时由于是一知半解，对同学问出的问题也难以给出很好的解答；
* 顶会论文对相关问题的研究非常深入，在阅读时需要先铺垫基础知识，致使阅读速度很慢。