2018年11月4日星期日

上午11时12分

CERT5.2跳槽用户主观特征识别检测研究

背景：

CERT5.2中，场景二攻击者主要刻画了因为寻求更好待遇而跳槽的内部用户，这些用户在经历了猎头咨询、邮件自荐、拷贝重要资料等步骤后，最终离开单位。

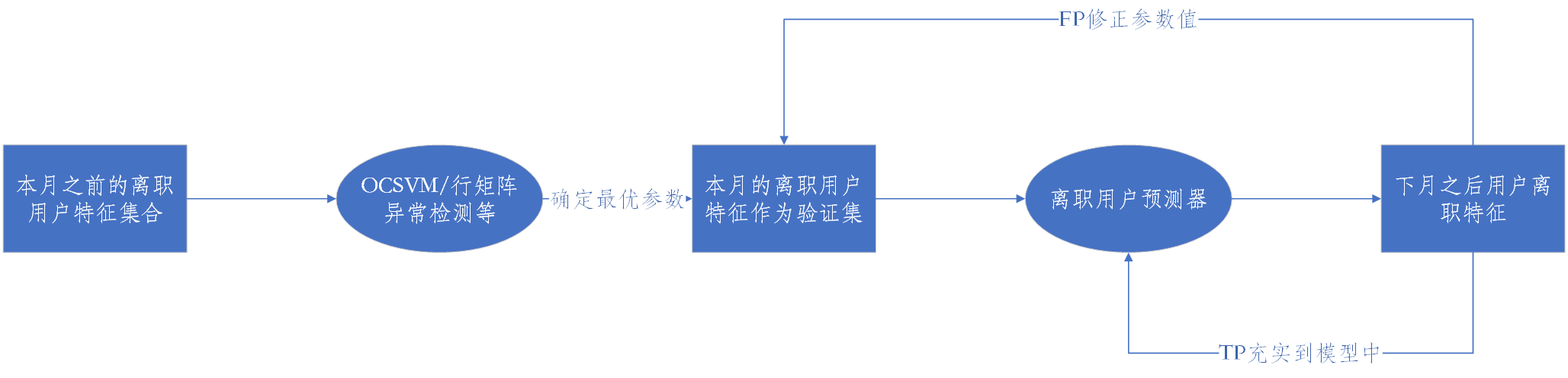
假设分析：

最早的研究工作聚焦于反生产行为CPB对于用户JS的影响，其关键来自于自身CPB的基础和周围工作环境的CPB影响；然后选择正常的用户进行了OCSVM训练，力图从中检测出JS可疑的用户。实验结果可以从4000个用户中识别出全部4个Insiders，而且识别出了400个用户集合。现在看来，实在没有必要在多种场景上验证主客观检测，因为数据场景描述中，仅明确了场景二跳槽攻击者受到工作满意度的影响；而工作满意度的影响主要是用户的上下班准时与否以及跳槽离职；

紧接着，今年十月份自己开始使用CERT5.2数据集进行重新研究，原因一是CERT5.2数据集中用户数量少了一半，分析起来更为高效；而且CERT5.2中也添加了满意度JS作为潜在动态变量，影响着用户的跳槽行为。然而，自己受到数据论文逻辑的影响，一直在努力刻画用户的人际关系（relationship），然而，在尝试了邮件联系、组织结构等关联后，得到的结果依旧不能很好地区分出高危用户。自己一方面不知晓论文数据构造时的方法，一方面也不想单纯以论文为指向进行方法研究，希望自己可以提出一种通用的行为预测方法。

因此，我们转而考虑如果直接从离职用户的角度来建模呢？朴素的想法：提取离职用户的诸多主观客观特征，然后一起使用聚类算法，查看目标用户聚类的的群簇所占当前离职用户的比例，以此作为该用户与离职用户的“距离”，比例越大，说明该用户越可能在下个月离职；

机器学习方法：对于离职用户而言，设定训练集、验证集与测试集，其中训练集使用OCSVM训练模型，而验证集用来选择最佳的参数；最终测试集用于实际测试；并且根据反馈的效果每月进行改进；



2018年11月4日星期日

下午11时12分

为了进一步分析CERT6.2中离职用户的特点，将Insiders\_2的30个用户周围的离职时间进行分析。

以BYO1846为例：

Insider\_LaidOff\_0,RMB1821,FDS1841,TRC1838,

Insider\_LaidOff\_1,NWP1609,ZAD1621,TAG1610,MAR1075,JXH1061,KEW0198,LSM1382,VCF1602,KBC1390,TNB1616,IVS1411,SLL0193,MFM1400,CTT0639,HIS1394,MZO1066,

Insider\_LaidOff\_2,WSW1091,JHP1654,JDM0208,HSF1115,UAM1108,CIM1095,ZHB1104,PTM1432,CIF1430,WDT1634,DHS0204,DDR1649,HMK0653,MIB0203,HMS1658,

BYO1846： 2010-12月离职

RMB1821,2010-02

FDS841， 2010-03离职

TRC183，2010-09离职

2018年11月5日星期一

下午1时50分

通过分析用户的离职情况，发现了之前忽略的一个问题：

对于一个用户而言，其有作用的离职用于应明确定义为在其之前离职的用户，而之前自己考虑用户的离职关系，仅具体到了离职的月份，而并未细致考虑用户离职的日期。

比如对于BYO1846而言，其离职时间为2010-12-15，那么就不应当考虑2010-12-20离职的12/20/2010 16:00:30,MZO1066。

因此，需要重新更新CERT5.2中的离职用户信息表以及每个用户的离职关系表。

原本以为自己发现了离职后的邮件联系。

以RMB为例子

Rose Maisie Blackwell,RMB1821,22,19,48,14,32

Rose Maisie Blackwell

Rose Maisie Blackwell,RMB1821,Rose.Maisie.Blackwell@dtaa.com,Salesman,,1 - Executive,5 - SalesAndMarketing,2 - Sales,5 - RegionalSales,Donna Erin Black

BYO1846与RMB123的通信次数总共为60，但是发生在2个月的时间内

RMB1821,-0.428571428571,2.0,42436.0,0.0,5.0,145618.6,0.6,

RMB1821,-0.5,7.0,210405.142857,0.285714285714,21.0,165035.857143,0.238095238095,

**BYO1846收到的RMB邮件**

01/05/2010 10:25:56,BYO1846,PC-8402,Buffy.Yolanda.Ortiz@dtaa.com,Donna.Erin.Black@dtaa.com,,Rose.Maisie.Blackwell@dtaa.com,Receive,20958,

BYO1846发送给RMB的邮件

Line 1039: 02/09/2010

10:22:31,BYO1846,PC-8402,William.Orson.Sykes@dtaa.com;Rose.Maisie.Blackwell@dtaa.com;Germane.Fredericka.Velez@dtaa.com,,,Buffy.Yolanda.Ortiz@dtaa.com,Send,44280,

突然发现：

之前写的程序大多存在错误疏漏，导致由此得到的结果要么不准确，要么错失重要的结果！要命！

今后写实验程序，无论大小，均需要经过简单的示例验证才能进行下一步，每一步的结果必须尽量坚实有力，别都分析了两个礼拜，再来回头查看最初的结果，想不起算法代码！

2018年11月5日星期一

下午3时52分

下午自己重写了之前的邮件过滤程序，考虑多发多收邮件的情况，并加入了邮件发送与接收通信天数的新特征，以便于计算日均发送与接收邮件。

作为验证：

以BYO1846用户为例：

RMB1821,-0.5,**7.0,5.0**,210405.142857,0.285714285714,**21.0,94.0**,165035.857143,0.238095238095,

其中RMB用户的邮件名为Rose.Maisie.Blackwell

这里总共记录了5+94=99天，明显与只有60条记录不符合

RMB用户记录太多，不容易矫正，使用

WDS1286,-1.0,0.0,0.0,0.0,0.0,1.0,1.0,1451552.0,1.0,

二封收到的邮件也都遗失了，开工修补

Willa.Dora.Sweeney

在努力修改的过程中，一个下午过去了，丝毫没有进展，干脆重新写吧，像这样几百行的代码几乎无法修改再读。

重写的目标：

1. 记录与目标用户有关的所有邮件通讯中，体现出的邮件联系人；
2. 不再专注1vs1的邮件通讯，包括邮件1vsN与Nvs1的邮件通讯；
3. 记录特征扩展为9个：

分别是：

邮件收发比，发送邮件总数，发送邮件行为的天数，发送的总字数，发送的总附件，接收的邮件总数，接收的邮件行为天数，接收的总字数，接收的总附件个数

下午下班时继续上述编程，终于将send邮件部分编写完毕，该部分针对用户Allen.Ashton.Buckley的send邮件JLF1315进行了验证，并且在两次send时准确记录了特征变化

再次联系用户 JLF1315 此时特征为： [2.0, ['2010-01-02', '2010-01-08'], '2053925619', 0.0]