2018年9月13日星期四

上午10时31分

通过初步分析CERT5.2第二个攻击场景中的某些用户周围的离职情况，有以下结果：

用户163个

['**BYO1846**', '**2010-12**', ['Buffy Yolanda Ortiz', 'BYO1846', 'Buffy.Yolanda.Ortiz@dtaa.com', 'Salesman', '', '1 - Executive', '5 - SalesAndMarketing', '2 - Sales', '5 - RegionalSales', 'Donna Erin Black']] 分析

经过部门筛选分析，得到

同一团队的离职员工为: 3

相同部门下不同个团队的离职员工为: **16**

相同职能部下不同部门的离职员工为: 15

同一事业部下不同职能部的离职员工为: 46

不同事业部的离职员工为: 82

分析新的一个内部跳槽用户：

['CHP1711', '2010-10', ['Cody Hall Phelps', 'CHP1711', 'Cody.Hall.Phelps@dtaa.com', 'Salesman', '', '2 - Executive', '5 - SalesAndMarketing', '2 - Sales', '4 - RegionalSales', 'Ethan Branden Garrett']]

同一团队的离职员工为: 2

相同部门下不同个团队的离职员工为:  **7**

相同职能部下不同部门的离职员工为: 10

同一事业部下不同职能部的离职员工为: 43

不同事业部的离职员工为: 61

可以发现，与用户关系最为密切的是同一个团队，离职人数在3/2人，然后考虑同一部门下不同小组，发现有16/7人，最后考虑同一个职能部门下的不同部门，发现相近有15/10人。再往下考虑不同职能部的关系，则人数突然增长为46/43人，建议前期先不考虑。

2018年9月13日星期四

下午2时44分

接着上午的工作

自己猜测：是不是跳槽的员工与普通员工之间，人际关系中离职人数存在差异呢？

为此，自己决定写程序分析全部2000个用户中，其接触的离职人员个数，且分为上述五个层次：

1. 同一个团队
2. 同一部门不同团队
3. 同一职能不同部门
4. 同一事业部不同职能
5. 不同事业部

即存储格式为：

User\_id, a,b,c,d,e

2018年9月17日星期一

下午4时36分

上周，经过初步的统计分析，得到了CERT5.2中所有用户的离职情况分析：

1. 同一个团队
2. 同一部门不同团队
3. 同一职能不同部门
4. 同一事业部不同职能
5. 不同事业部

其中部分用户的第一个维度特征（同一个团队中离职人数）很少，一部分则很多。因此，为了进一步清晰分析出前三个维度层次的作用，计划首先分别针对三个层次来绘制图像，然后针对前二个维度/前三个维度的单个用户和来绘制图像，其中CERT5.2的场景二用户用红色标识。

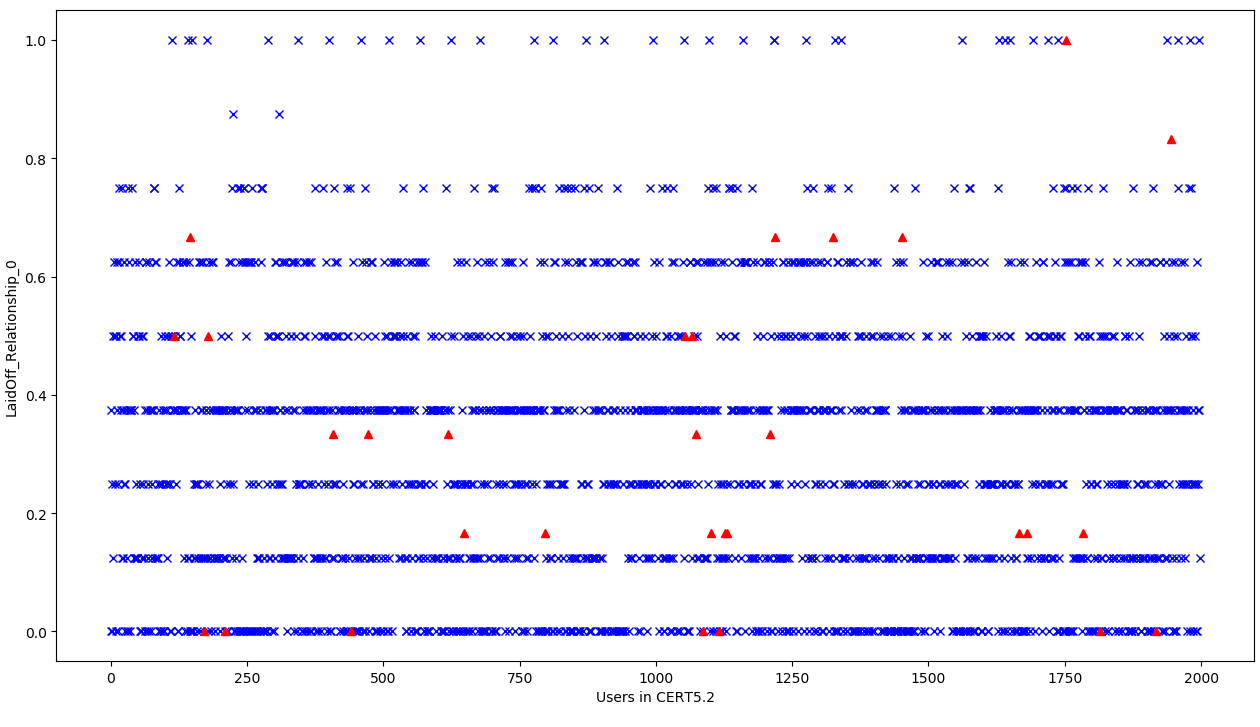
2018年9月19日星期三

上午10时51分

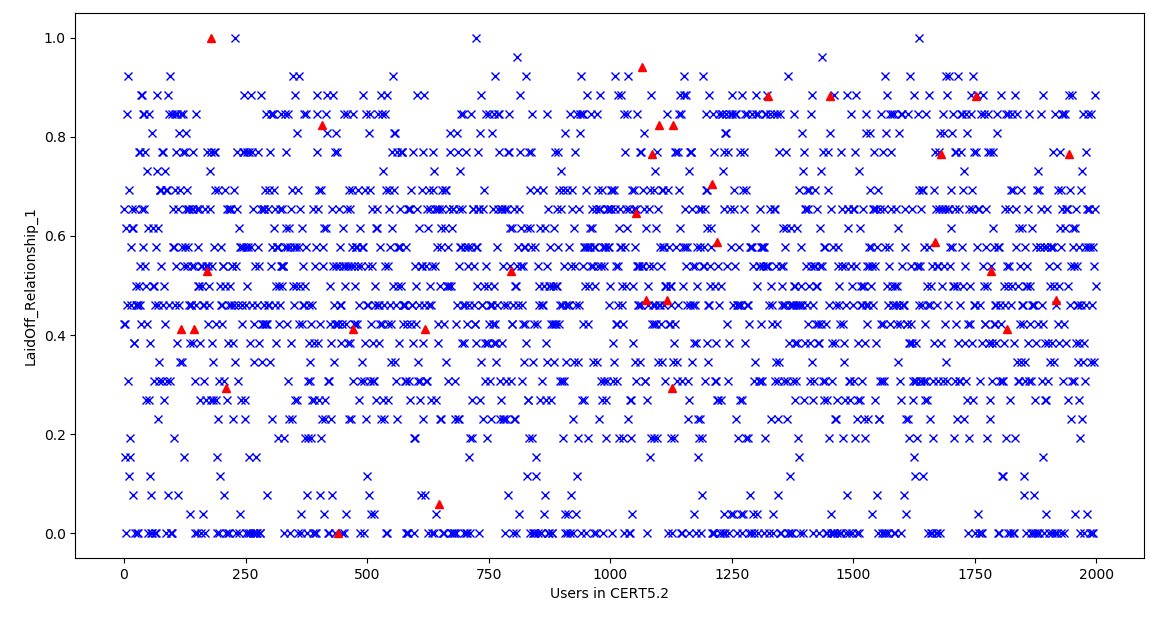
今天周三，终于开始完成上述实验。

经过统计分析1999个用户（没考虑CEO）

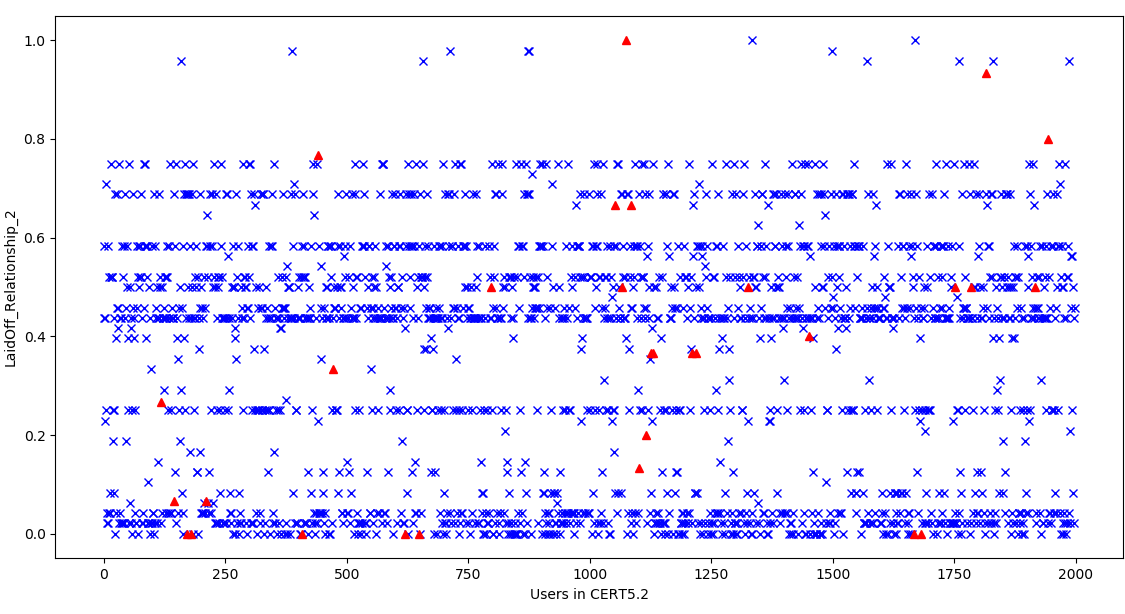
对于同一团队而言（LaidOff\_Relationship\_MinMax\_0)）



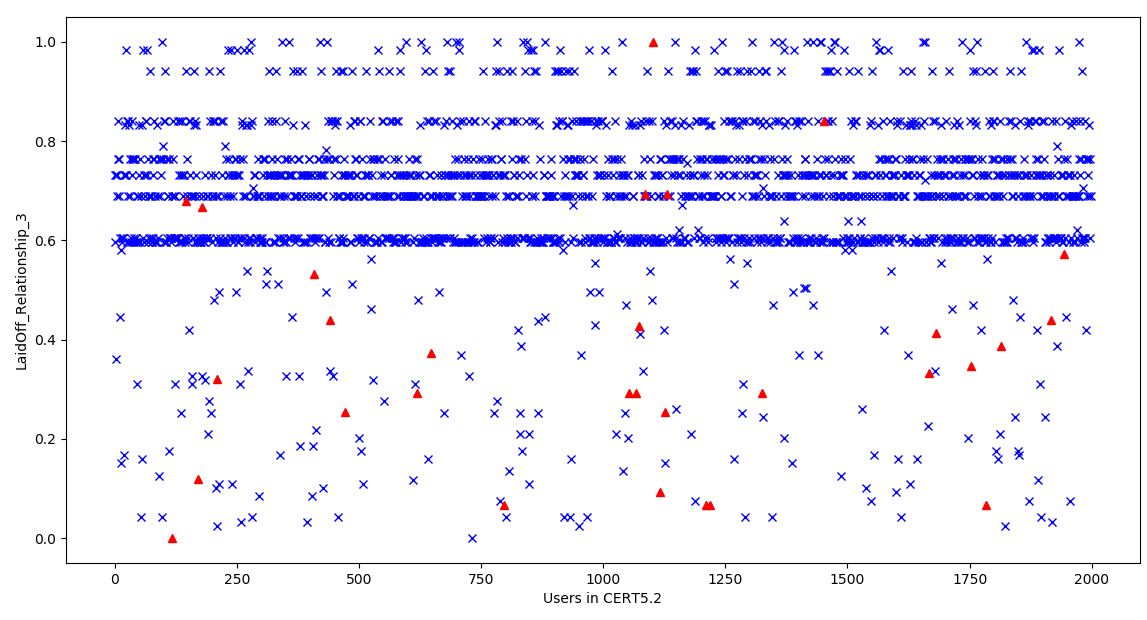
考虑同一部门内不同团队（LaidOff\_Relationship\_MinMax\_1）



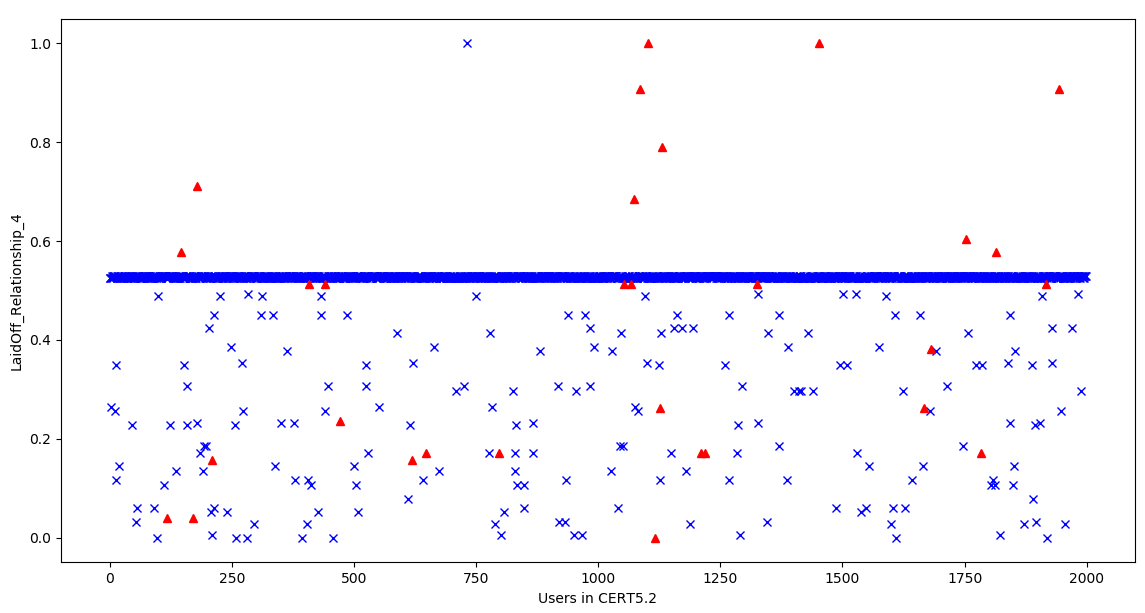
对于同一职能部门内不同部门（LaidOff\_Relationship\_MinMax\_2）:



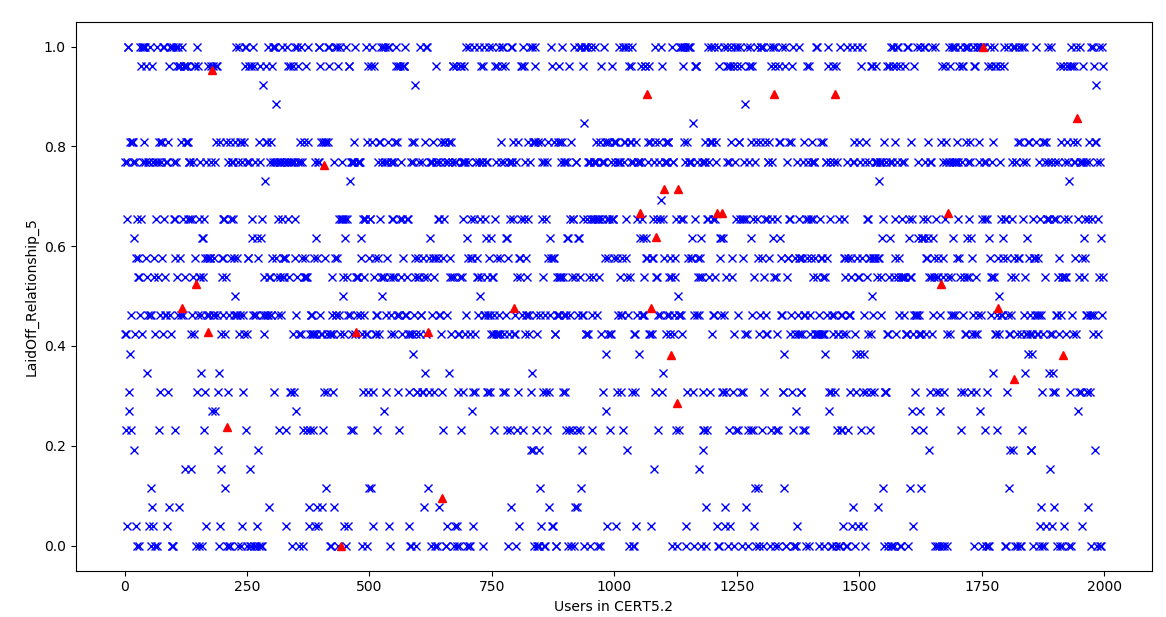
对于同一事业部内不同职能部门（LaidOff\_Relationship\_MinMax\_3）:



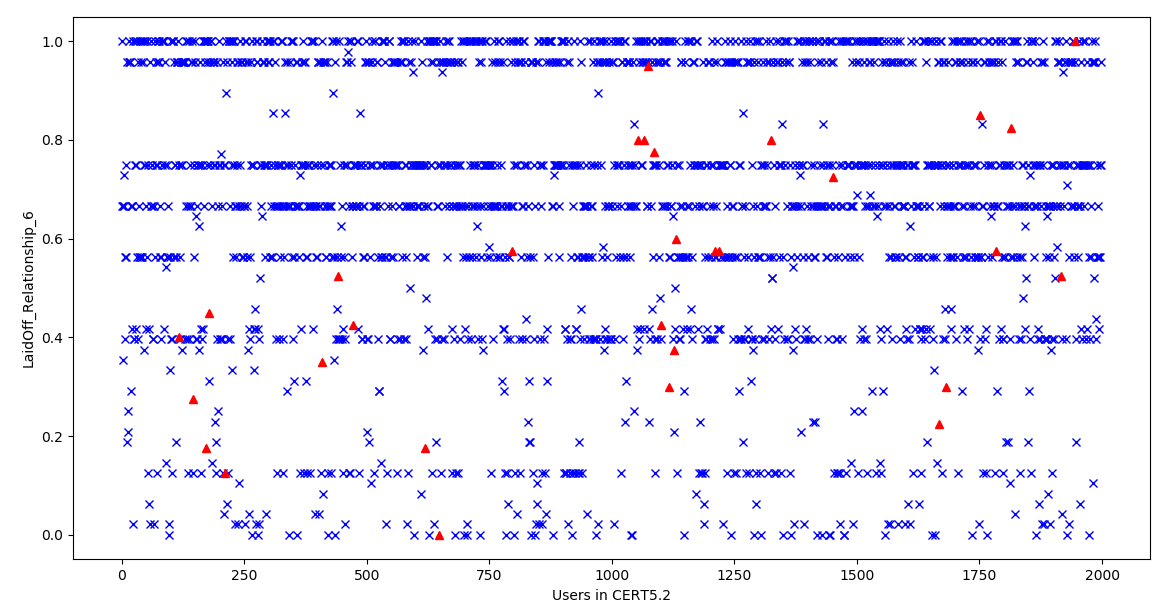
对于不同事业部（LaidOff\_Relationship\_MinMax\_4）:



对于同一部门内离职人员（LaidOff\_Relationship\_MinMax\_5）:



对于同一职能部门内离职人员（LaidOff\_Relationship\_MinMax\_6）:



2018年9月19日星期三

下午10时17分

睡前小结，整理思路。

目前已有的实验分析结果有：

1. CERT5.2数据构造说明中提到了隐含的动态的工作满意度变量，并在因果关系图中，由用户的人际关系Relationship与离职人员LaidOff共同决定了用户的工作满意度Job Satisfactory，而JS又决定了用户的跳槽行为（场景二）与不守时行为；
2. 现有实验分析了基于组织架构层次的用户周围离职人员关系，发现对于场景二而言，不存在明显的区分度，即单单从离职人员关系数量角度无法有效区分场景二跳槽用户；
3. CERT5.2数据说明中，提到了每个用户赋予的初始静态OCEAN大五人格分数，其中外倾性E驱动着用户交朋友的数量多少；而尽责性C驱动着迟到行为的多少。
4. 原有数据图中说明：...relationships to laid off employees affect job satisfaction；字面意思理解即与解雇人员的关系影响着用户的工作满意度，可以推测，一般解雇用户会不会影响我们的工作满意度呢？那要看我们与这个人的关系如何，如果是好朋友或者熟识的关系，那么就影响比较大；如果仅仅是一般同事的关系，那么可能影响比较小。依据本点分析，之前的工作借助了LDAP数据分析了CERT5.2中的部门组织结构，并在此基础上统计了每个用户相关的五种层次解雇人员数量；之后只需要分析邮件通讯确定好友关系即可（邮件往返达到某个阈值，或者超过一般水平等，是否要借助分析Topic尚且未定）。但是该方法考虑未涉及到用户的OCEAN数据。

因此，我的一个初步想法是：

* 选择一个场景二跳槽用户
* 分析该用户每个月的邮件通讯情况，列出与其通信的用户，以及send/recv的封数；
* 基于每月可以对通讯数量进行一个排序，越靠前的联系越紧密，越重要；
* 查看离职用户是否在紧密关系中；
* 考虑python实现一个图结构

1. 明天的一个主要任务是提取CERT5.2的数据中的用户月记录，即按照不同用户，将其每月的五类数据记录按照天记录为相应文件，在单位台式机上部署，分析数据结果；
2. 同时第二个任务是通过python绘制CERT5.2的用户邮件图（按月积累，即第一个月，前两个月，前三个月...）