3.4.3 恶意代码个体画像构建

构建了恶意代码行为标签系统之后，需要将恶意代码样本的行为报告映射到标签上，然后利用这些标签构建恶意代码的行为画像。如 3.3 节所示，标签系统一共有三大类的标签，分别是频率标签、规则标签以及关联标签。其中频率标签和关联标签通过对动态行为报告计算可直接获得，具体算法已在 3.3 节中阐述。规则标签的构成是由领域知识构成的，这里构建了产生式规则将恶意代码行为与规则标签进行关联。具体方法 3-2 所示。

算法 3-2 规则标签构建方法

输入: 恶意代码动态行为报告 $\boldsymbol{R}$

输出: 恶意代码规则标签 $\boldsymbol{L}$

```
初始化行为集合 L：；
if API SetWindowsHookExA(WH_KEYBOARD_LL) ∈ R then
    L ← 创建一个键盘记录器监视键盘输入
end if
if API GetComputerNameA ∈ R then
    L ← 查询计算机名称
end if
if API (NtUnmapViewOfSection, NtAllocateVirtualMemeory) ∈ R then
    L ← 可能的进程镂空
end if
if dead_host ∈ R then
    L ← 连接到不再响应的 IP
end if
.....
if checkWindowsidletime ∈ R then
    L ← 查询 windows 空闲时间
end if
return L
```

在获得了样本的所有标签之后，采用思维导图的形势对其进行可视化。这里采用了 ECHARTS 前端框架实现树状的导图，图 3-2 为某一恶意代码样本的个体行为画像。

3.5 实验结果与分析

3.5.1 实验环境以及评价指标

本文实验主要在 Ubunut 系统下运行，主要代码使用 Python 语言，具体环境如表 3-8 所示。