YD 用户手册

引言

最典型的 DoS 攻击利用 TCP 链接的“三次握手”过程。在 TCP 协议中，为保证链接的可靠性，采用三次握手建立一个链接。

第一次握手：建立连接时，客户端发送 SYN 包((SYN=i)到服务器，并进入 SYN SEND 状态，等待服务器确认；

第二次握手：服务器收到 SYN 包，必须确认客户的 SYN (ACK=i+1)，同时自己也发送一个 SYN 包((SYN=j)}即 SYN+ACK 包，此时服务器进入 SYN_RECV 状态；

第三次握手：客户端收到服务器的 SYN+ACK 包，向服务器发送确认包 ACK(ACK=j+1)，此包发送完毕，客户端和服务器进入 ESTABLISHED 状态，完成三次握手，客户端与服务器开始传送数据。

SYN 攻击又称为半链接攻击，攻击者向服务器发送大量的 SYN 请求，但不响应服务器返回的结果。导致服务器维护大量的 TCP 半链接状态。从而导致正常的 TCP 链接无法建立。

YD 就是用于检测这种 SYN 攻击的可视化程序。

基本信息

系统：Ubuntu 14.04 linux kernel 3.13.0-32-generic

语言：C

依赖库：libgtk+-3.0 libnotify-dev libpcap

- libgtk+-3.0 是用 C 语言实现的图形界面库，同时提供了一些基本的数据结构和线程控制。
- libnotify-dev 是 GNOME 桌面环境下，用于向用户发出提示信息的库。
- libpcap 是一个用得很广泛的抓包库。可是在应用层抓取想要的数据包，不过需要超级用户权限。

实现原理

在 TCP 三次握手过程中，当客户端向服务器发出 SYN 请求时，服务器会返回一个 SYN+ACK 的响应，最后客户端又返回一个针对服务器 SYN 的 ACK 响应，完成链接。而 SYN 攻击会正常发送第一次的 SYN 请求，但是没有第三次的 ACK 响应。

这里有一点要注意的，是客户端发送的 SYN 请求和服务器返回的 SYN+ACK 响应是对应关系，同时服务器返回的 SYN+ACK 和客户端最后做出的 ACK 响应也是对应的关系。而在攻击过程中，第一次的对应关系依然是成立的，不过丢失了客户端的 ACK 导致第二次的对应关系无法实现。

YD 检测 SYN 攻击的原理就是记录每次服务器发出的 SYN+ACK 数据包，然后观察后续是否有对应的 ACK

数据包，如果有则说明是正常的链接；否则就说明是一条有异常的链接请求。当这种有异常的链接请求到底一定数量，就认为遭到了攻击。

编译运行

以 ubuntu14.04 系统为例，要编译 yd 需要先安装其依赖库。

```
sudo apt-get install libgtk-3-dev libpcap-dev libnotify-dev
```

切换到源代码目录，执行

```
cd yd

./configure

make
```

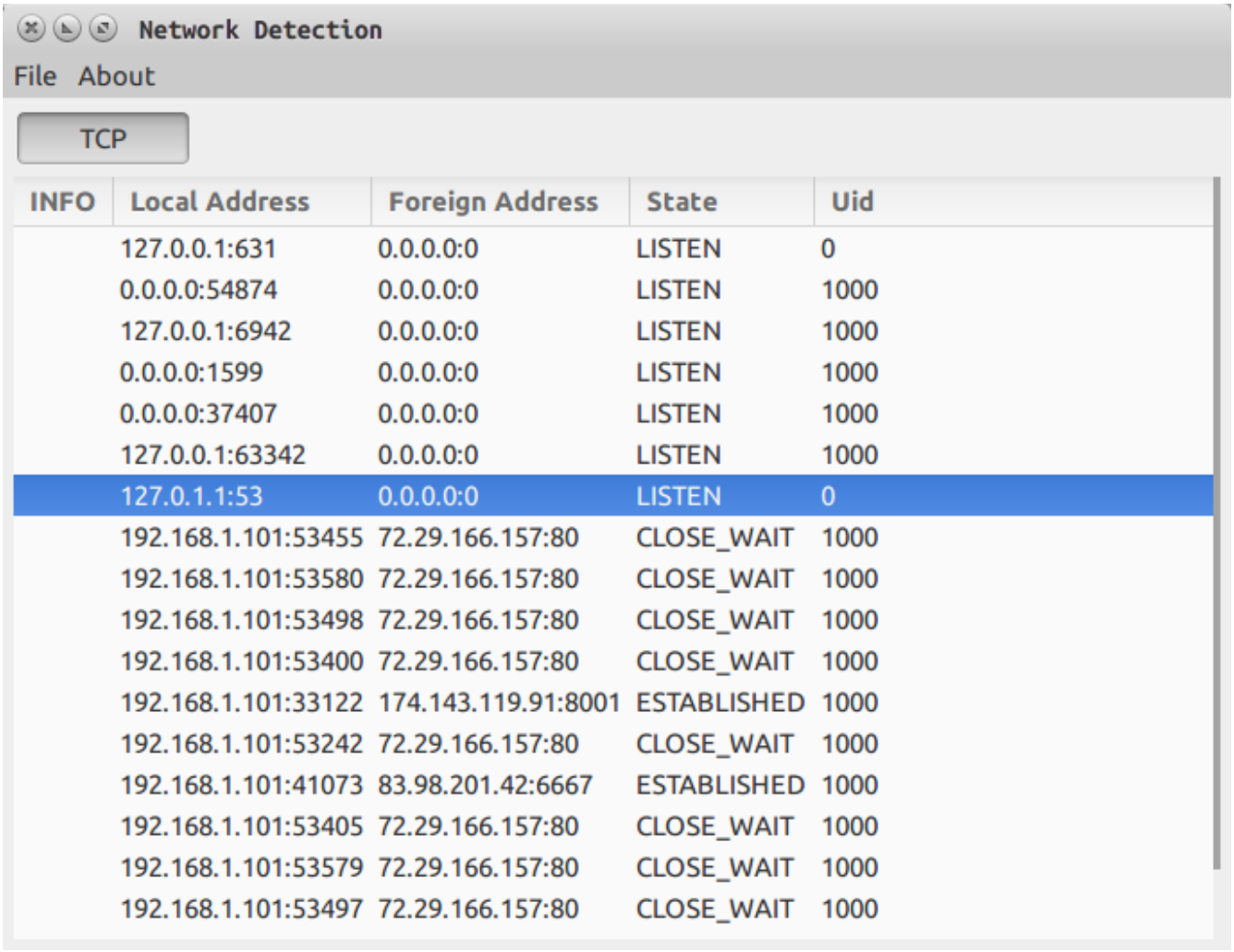
如果没有错误就完成了编译。可执行文件在./src/yd。用超级用户权限运行。

```
sudo ./src/yd
```

功能描述

显示系统当前 TCP 网络状态

运行程序显示主界面，在主界面中显示当前系统上的 TCP 网络链接状态，包括 IP 地址、端口号和远程链接目标地址，当前的链接状态，以及创建该链接的用户 ID。

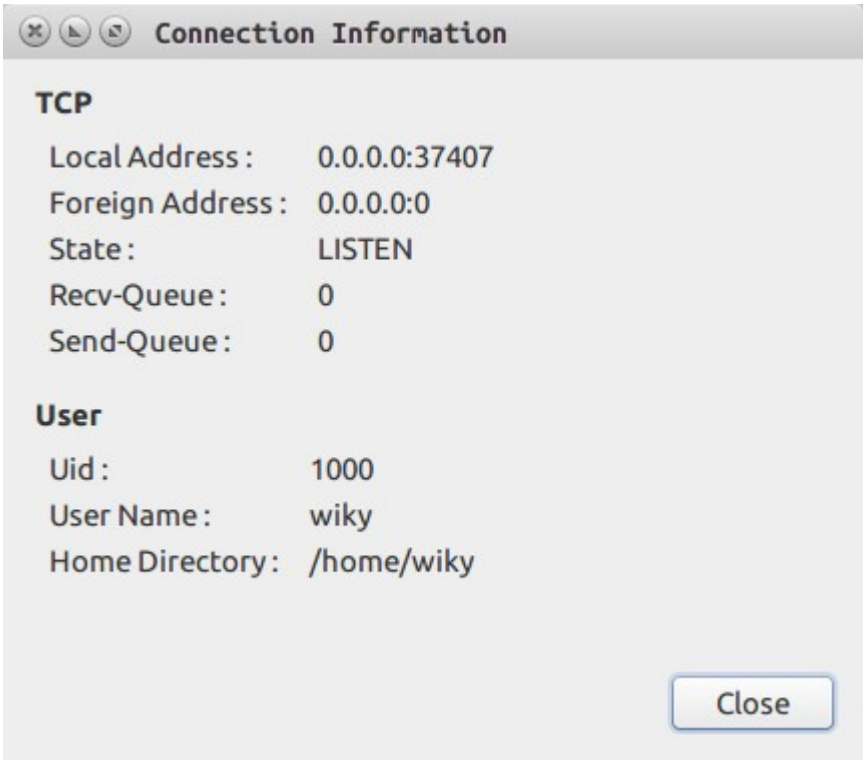


其中状态一栏中的字段意思和 `netstat` 命令的输出一致。下面简单介绍几个

- `LISTEN`：表示该端口是一个被动接受链接的端口，正处于监听状态。
- `CLOSE_WAIT`：表示该链接已经关闭，现在处于关闭后的等待状态
- `ESTABLISHED`：链接已经建立，可以正常收发数据

显示链接的详细信息

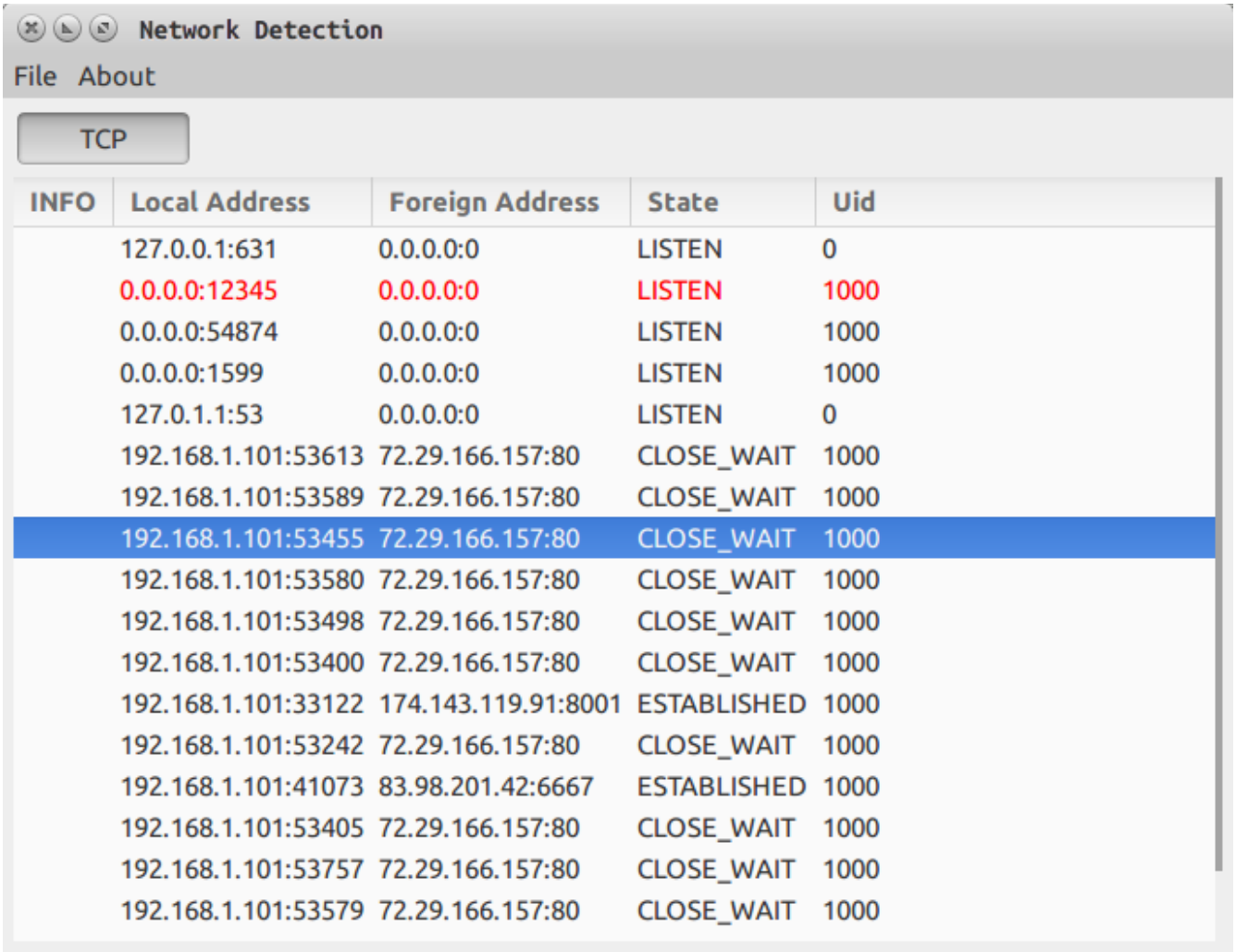
双击链接列表中的一个条，可以在一个新窗口中查看更多关于该链接的信息。包括收发队列长度，以及创建该链接的用户信息。



一般情况下，收发队列都是 0。

检测 SYN 攻击并提示用户

在 yd 程序启动时，会在后台启动一个线程来执行 SYN 攻击的检测。这个过程用户是不可见的，除非检测到攻击。一旦检测到系统的某个端口遭到了 SYN 攻击，就会在界面上用红色表示该链接。



同时使用系统的提示给用户发出警告。

程序启动

yd 需要通过抓取数据包来分析当前系统是否受到攻击，而抓取数据包需要用到超级用户权限。因此如果用户没有以超级用户权限运行 yd 时，会提示用户输入密码提升权限。