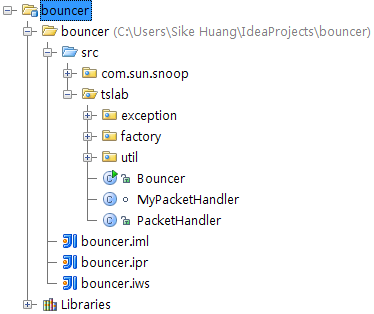
Bouncer

By Shanbo Li and Sike Huang

**Project Structure**



The project is composed of several packages:

**com.sun.snoop** is used for packet validation

**tslab.exception** contains exceptions that might be thrown during packet creation

**tslab.factory** contains various classes to generate ICMP, TCP and FTP packets

**tslab.util** has common utilization code to support factory

The entry point is **tslab.Bouncer**, which parses the command line arguments, then listens to the incoming packets by **tslab.(My)PacketHandler**, and creates corresponding outing packets using certain suitable factory.

**Command Line Argument**

Our bouncer takes command line argument specific as following:

*java tslab.Bouncer [interface] listen\_ip:listen\_port server\_ip:server\_port*

**interface** is optional, it is the network device used to accept and forward packets, user can either specify it, such as eth0. Or the program will give a list of devices to be chosen, such as:

List of interfaces

0: \Device\NPF\_{8C34DCC7-8F0C-475E-8F62-F159F050B026} [ip=/0.0.0.0]

1: \Device\NPF\_{E09BD06A-3EBA-4364-9F94-0383CADD6DE1} [ip=null]

2: \Device\NPF\_{B920C176-DC06-4740-886B-1051777BB8DE} [ip=/192.168.1.104]

3: \Device\NPF\_{24234719-6C60-4BB4-A604-9600239CDFE5} [ip=/10.8.0.62]

Select one:

**listen\_ip** and **server\_ip** are mandatory.

**listen\_port** and **server\_port** are optional, and they behave in such a way:

|  |  |  |
| --- | --- | --- |
| listern\_port | server\_port | Effect |
| Given | Given | Listen packets from given port, and forward to given port on server (TCP, FTP) |
| Not given | Not given | Listen packets from all ports, and forward to same port on server (ICMP, TCP, FTP) |
| Given | Not given | Listen packets from given port, and forward to same port on server (TCP, FTP) |
| Not given | Given | Listen packets from all ports, and forward to given port on server (TCP, FTP) |

**Please pay attention to these two points:**

1. **ICMP** only works when there is neither listen\_port nor server\_port.
2. And in case of **FTP**, the data channel port on server is assumed to be the given server\_port minus one, and on bouncer port 20 is always opened and used as data transmission channel towards client.