EE298 Chat Sniffer Documentation

Elizabeth Loyola & Kenan Biel Virtucio
March 10, 2011

Contents

--#--

I. Introduction

This program is implemented using Java to sniff and log messages sent and received via Chikka, Meebo and Facebook chat. Java was chosen because of Jpcap's adequate classes and organized documentation.

II. Program Specification

A. System Requirements
1. libpcap (v. 0.8 or later)
2. jpcap
- this library package may have a problem on 64-bit architecture

B. Program Workflow

1. Sniffing
Jpcap provides a class JpcapCaptor which has methods that could fetch list of available devices and makes an instance of it for sniffing.

This line of code provides the list of available devices:

```
NetworkInterface[] devices = JpcapCaptor.getDeviceList();
```

This code makes an instance of a device on the list that will capture 2000 bytes per packet at a given time, with promiscuous mode on, and time out time of 20:

```
JpcapCaptor.openDevice(devices[Integer.parseInt(args[0])], 2000, true, 20);
```

2. Filtering

JpcapCaptor also has a method setFilter that could specify which packets to capture during a session. However these filters sometimes doesn't work with Meebo and Chikka within Dilnet. Hence, filtering is also implemented by choosing keywords found on the data part of each packet.

Chikka and Meebo messages are sometimes distributed into two packets depending on the length of the messages. To solve this, additional filter cases were also implemented.

Setting filter method:

```
jpcap.setFilter("host chikka.com", true);
```

Capturing Chikka receive packet

```
else if((protocol==0)&&
      ((data.startsWith("HTTP/1.0 200 OK"))||
      (data.startsWith("HTTP/1.1 200 OK")))&&
      (data.indexOf("xmlns='http://jabber.org/protocol/httpbin
      d'")!=-1)&&
      (data.indexOf("type='chat'>")!=-1)){
```

C. User Interface

```
sudo java Sniffer <interface number> <protocol: chikka, meebo or fb>
```

examples:

```
sudo java Sniffer 1 chikka
sudo java Sniffer 0 meebo chikka
sudo java Sniffer 1 chikka meebo fb
```

III. **Chikka**

After filtering Chikka packets, the data part of the packet will be parsed. This is done by looking for keywords and getting the substring of the data packet between these keywords.

This is done inside a **while loop** since Chikka sometimes sends multiple messages in a single packet.

```
while(data.indexOf("<body>")!=-1){
    start = data.indexOf("\" to=\"") + 6;
    end = data.indexOf(" type=\"chat\"><body>");
    to = data.substring(start, end);
    System.out.println("receiver: " + to);
```

```
                start = data.indexOf("</body>");
                message = data.substring(end + 19, start);
                System.out.println("message: " + message + "\n");

                end = data.indexOf("</message>");
                data = data.substring(end+10, data.length());

            }
```

Messages sent to a certain user will be then be loged on a single file.

```
            try {
                  fw = new FileWriter("chikka/" + to + ".txt", true);
                  fw.append(this.getDateTime(1));
                  fw.append(" " + from + ": ");
                  fw.append(message + "\n");
            }
            catch (IOException e) {}
```

Messages will also be dumped in a single file.

```
            try {
                  fw = new FileWriter("chikka/" + this.getDateTime(0) + "-
logfile.txt", true);
                  fw.append("\n" + this.getDateTime(1) + " to: "+ to + " from: " +
from + ": " + message);
            }
            catch (IOException e) {}
```

IV. **Meebo**

After filtering meebo packets, it will also be parsed the same way chikka packets are parsed.

```
        start = data.indexOf("&sender=") + 8;
        end = data.indexOf("&receiver=");
        from = data.substring(start, end);

        start = data.indexOf("&protocol=");
        to= data.substring(end+10, start);

        start = data.indexOf("&msg=");
        message= data.substring(start + 5, data.length());
        System.out.println("Time: " + this.getDateTime(1));
        System.out.println("sender: "+to + "\nreceiver: "+ from + "\nmessage: " +
message);
```

Then it will be logged by conversation:

Logging for meebo sent packets:

```
        try {
            fw = new FileWriter("meebo/" +  this.getDateTime(0)+ from +"-"+to +
".txt", true);
            fw.append(this.getDateTime(1));
            fw.append(" " + from + ": ");
            fw.append(message);
```

```
        }
        catch (IOException e) {}
```

Logging for meebo received packets:

```
        try {
             fw = new FileWriter("meebo/" + this.getDateTime(0)+to + "-" +from+
".txt", true);
             fw.append("\n"+this.getDateTime(1));
             fw.append(" " + from + ": ");
             fw.append(message + "\n\n");
        }
        catch (IOException e) {}
```

V. **Facebook**

Facebook chat data will also be parsed the same way Chikka and Meebo packets are parsed. It will then be logged by “chat session”. All captured messages will also be logged on a single logfile.

Parsing facebook chat:

```
//from name
start = data.indexOf("\"from_name\"") + 13;
end = data.indexOf(",\"from_first_name\"") - 1;
from = data.substring(start, end);

// from id
start = data.indexOf("\"from\":") + 7;
end = data.indexOf(",\"to\":");
String from_id = data.substring(start, end);

//to name
start = data.indexOf("to_name") + 10;
end = data.indexOf("\",\"to_first_name");
to = data.substring(start, end);

//to id
start = data.indexOf(",\"to\":") + 6;
end =  data.indexOf("\"from_name\"")-1;
String to_id = data.substring(start, end);

//"session"
start = data.indexOf("(;;);{\"t\":\"msg\",\"c\":\"p_") + 23;
end =  data.indexOf("\",\"s\":");
session = data.substring(start, end);

//message
start = data.indexOf("{\"msg\":{\"text\":\"") +16;
end = data.indexOf("\",\"time\"");
message = data.substring(start, end);

//ip
end = pack.indexOf("->/") +18;
start = end-33;
String ip = pack.substring(start, end);
```

Logging facebook chat by session:

```
        if (session.equals(from_id)){ A = from; B = to; }
        else{ A = to; B = from; }
        try {
            fw = new FileWriter("facebook/" + this.getDateTime(0) +"-"+ A + "-" + B
+ ".txt", true);
            fw.append(this.getDateTime(1));
            fw.append(" " + from + ": ");
            fw.append(message + "\n");
        }
        catch (IOException e) {}
```

Logging all messages in one file:

```
        try {
            fw = new FileWriter("facebook/" + this.getDateTime(0) + "-logfile.txt",
true);
            fw.append("\n" + this.getDateTime(1) + " to: "+ to + "from: " + from +
": " + message);
        }
        catch (IOException e) {}
```