

Network Traffic based Application Identification Demonstrator (Net – AI) GUI

Kenny Nguyen

4150864@swin.edu.au

Supervisor: Sebastian Zander
CAIA INTERNSHIP



Outline



- Motivation
- Design
- Implementation
- Demonstration
- Conclusion and Future work



Motivation



- Involve with CAIA (research community, great place to learn)
- Passion for programming
- Algorithm
- GUI application



CAIA Internship Seminar

<http://caia.swin.edu.au> 4150864@swin.edu.au 17 February 2006 Page 3

Why Packet Classification



- Simple: Fair Dinkum Business
- No longer depends on:
 - Port Identification
 - Stateful Reconstruction
 - Signature-based method
- Alternative (Proposal): Machine Learning Algorithm
 - Identify end-host application
 - Priority Traffic Queuing
 - Meet the demand of QOS requirement
 - Detection, Security, Surveillance
 - Traverse efficiently, problem-free, with high performance



CAIA Internship Seminar

<http://caia.swin.edu.au> 4150864@swin.edu.au 17 February 2006 Page 4

Design



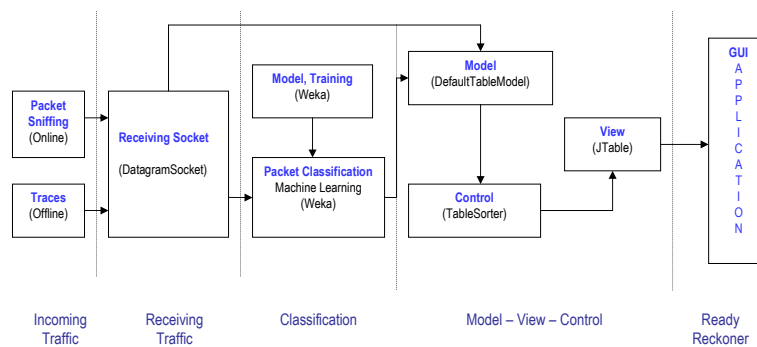
- Use Model – View – Controller.
 - Object - Oriented Design (Modularity)
 - Heuristics “Rules of thumb”:
 - Structure of individual classes and objects
 - Collaboration, containment, association
 - Inheritance, including multi inheritance
- Then easy to accommodate new requirements.



CAIA Internship Seminar

<http://caia.swin.edu.au> 4150864@swin.edu.au 17 February 2006 Page 5

Design



CAIA Internship Seminar

<http://caia.swin.edu.au> 4150864@swin.edu.au 17 February 2006 Page 6

Implementation (aka Coding)



- Java (Platform Independent, wide spectrum)
- Java Version 1.5
- Java AWT and Swing packages
- Thread, HashMap, StringTokenizer.



CAIA Internship Seminar

<http://caia.swin.edu.au> 4150864@swin.edu.au 17 February 2006 Page 7

Implementation



```
■ try
■ {
■     BufferedReader reader = new BufferedReader(new FileReader("license.txt"));
■     JTextArea text = new JTextArea(8, 50);
■     String line;
■     while ((line = reader.readLine()) != null)
■     {
■         text.append(line);
■         text.append("\n");
■     }
■     text.setCaretPosition(0);
■     text.setFont(new Font("Dialog", Font.PLAIN, 12));
■     text.setEditable(false);
■     text.setForeground(new Color(0,0,225));
■     JOptionPane.showMessageDialog(
■         pane,
■         new JScrollPane(text),
■         "Licensing: Please email Grenville Armitage <garmitage@swin.edu.au> ",
■         JOptionPane.INFORMATION_MESSAGE,
■         new ImageIcon("swin.jpg"));
■ }
```



CAIA Internship Seminar

<http://caia.swin.edu.au> 4150864@swin.edu.au 17 February 2006 Page 8

Implementation



```
try
{
    File f = new File("port.txt");
    FileInputStream fis = new FileInputStream(f);
    BufferedInputStream bis = new BufferedInputStream(fis);
    dis = new BufferedReader(new InputStreamReader(bis));

    // read the first record of the database
    while ( (dbRecord = dis.readLine()) != null)
    {
        StringTokenizer st = new StringTokenizer(dbRecord);
        key = st.nextToken();
        value = "";
        while (st.hasMoreTokens())
        {
            value = value + " " + st.nextToken();
        }
        portNumber.put(key,value);
    }
}
```



Implementation



```
public int compare(Object o1, Object o2)
{
    String s1 = (String) o1;
    String s2 = (String) o2;

    int index = s1.indexOf(" ");
    String s1Prefix = s1.substring((index + 1));
    double s1Number = Double.parseDouble(s1.substring(0, index));

    index = s2.indexOf(" ");
    String s2Prefix = s2.substring((index + 1));
    double s2Number = Double.parseDouble(s2.substring(0, index));

    if (!s1Prefix.equals(s2Prefix))
    {
        return s1Prefix.compareTo(s2Prefix);
    }
    else
    {
        return s1Number > s2Number ? 1 : s1Number < s2Number ? -1 : 0;
    }
}
```



Demonstration



Source IP	Destination IP	Bytes	Source Port	Destination Port	Protocol	Duration (s)	Prediction Class	
136.186.229.4	ALL-ROUTERS.MC488		HSRP	HSRP	UDP	28.587	53	
136.186.229.92	136.186.255.255	229	netbios-dgm (U...	netbios-dgm (U...	UDP		0	53
136.186.229.6	ALL-ROUTERS.MC488		HSRP	HSRP	UDP	29.796	53	
136.186.229.2	ALL-ROUTERS.MC4716		HSRP	HSRP	UDP	37.936	53	
136.186.30.5	ALL-ROUTERS.MC444		HSRP	HSRP	UDP		0	53
136.186.229.3	ALL-ROUTERS.MC4672		HSRP	HSRP	UDP	38.082	53	
136.186.30.2	ALL-ROUTERS.MC4668		HSRP	HSRP	UDP	35.133	53	
136.186.30.3	ALL-ROUTERS.MC4624		HSRP	HSRP	UDP	35.121	53	
136.186.229.5	ALL-ROUTERS.MC488		HSRP	HSRP	UDP	28.961	53	
136.186.229.92	lc118.opt2.poin...744		QuakeIII	QuakeIII	UDP		12.76	53
136.186.229.92	64.242.242.40		QuakeIII	QuakeIII	UDP		12.724	53
136.186.229.92	ool-4355e082....746		QuakeIII	27961	UDP		12.743	53

Initial Parameters
ClassifierName = weka.classifiers.trees.J48; Training File = auck11.arff



CAIA Internship Seminar

<http://caia.swin.edu.au> 4150864@swin.edu.au 17 February 2006 Page 11

Conclusion



- Running and testing under java 1.5 successfully.
- Easy to use GUI application
- Monitor & Classify network traffic
- Separate active and completed traffic
- Learning from Project
 - Strengthen programming, research skills
 - Day to day professionalism



CAIA Internship Seminar

<http://caia.swin.edu.au> 4150864@swin.edu.au 17 February 2006 Page 12

Future work



- Save file in different format (CSV, XML, Excel Style Sheet, etc).
- Open traffic capture (flow statistics) file into table to analyse
- Create new Panel to analyse network traffic and provide statistic.
- Implement configuration dialog
- Implement filter function



CAIA Internship Seminar

<http://caia.swin.edu.au> 4150864@swin.edu.au 17 February 2006 Page 13

We're there



Thank you for listening

Questions



CAIA Internship Seminar

<http://caia.swin.edu.au> 4150864@swin.edu.au 17 February 2006 Page 14