Design of the Flow Monitoring Software of Android Phone Based on 4G Network

Lili Wang¹, Wengang Zhou²*

1. Rongzhi College of Chongqing Technology and Business University, Chongqing, Sichuan, China
2. School of Computer Science and Technology, ZhouKou Normal University, Zhoukou, Henan, China

Abstract: The mobile phone network has been upgraded to 4G, and this network type has the advantage of fast Internet access, but also has the disadvantage of high flow cost. Aiming at realizing the flow monitoring function of 4G mobile phone, this paper studies the self-similarity principle of the flow and the recognition algorithm of the unknown flow, designs flow monitoring software consisting of Iptables module and software management on this basis, and tests the flow monitoring function and flow controlling function of the software. And the test results show that the flow monitoring software designed in this study can well meet the needs of users.

Keywords: flow data; pattern matching; demand analysis; monitoring module; recognition rate

1. Introduction

Flow monitoring is to track, detect and deal with the data packet of the network accessing in the system. With the increasing appliance of Android mobile phones, more and more malicious flow charge appears, making mobile phone users suffer unnecessary economic losses. In order to prevent the loss of flow for no reason, we need to design scientific and reasonable flow monitoring software. Thus, this paper tries to design flow monitoring software meeting users’ demand from the aspects of the similarity principle of flow and the recognition algorithm of flow.

Many scholars have studied the flow monitoring of mobile Internet terminal. Zheng Yunqing, et al[2] (2013) point out that the strategy of developing monitoring software in application layer is to collect the statistics of GPRS data flow in the month and the day, and the display to the user in real time. Zhang Yi, et al[3] (2011) propose a safe monitoring strategy of weighted behavioral characteristics, which can effectively block viruses, stop malicious charging, but may block the behavior of normal applications due to false judgment on some applications, thus giving users unnecessary trouble. Marina Thottan, et al[4] (2003) propose a network anomaly detection method based on MIB (Management Information Base) variable mutation analysis. Based on the findings of above scholars, this paper designs a piece of flow monitoring software meeting users’ demand, to provide reasonable flow usage policy for Android mobile phone users of 4G network.

2. The self-similarity principle of the flow and the recognition algorithm of the unknown flow

2.1 Self-similarity principle

Flow has the characteristic of self-similarity. Self-similarity refers to that the system presents similar characteristics at different scales of certain measure, and its essence is a special case of system scaling behavior. However, network flow, as a complex system, also has the characteristic of self-similarity, and the representation is that the statistical characteristics of the data packet are not affected by time scale within a unit time in the long range of time. [5] Assume random time process is \( X(t) \). The process expresses the data amount of arrival or departure at time \( t \). For any real numbers \( \alpha > 0 \), if random process \( X(\alpha t) \) and \( X(t) \) have the same statistical distribution, random process \( X(t) \) is statistically self-similar to parameter \( H \). \( H \) is Hurst index, used to represent the self-similar degree of system at different scales, and the larger value of \( H \) represents higher self-similar degree of network flow. \( X(t) \) has the relationship shown in Formula (1)–(3):

\[
E\left[X(t)\right] = \frac{E\left[X(\alpha t)\right]}{\alpha^H} \tag{1}
\]

\[
Var\left[X(t)\right] = \frac{Var\left[X(\alpha t)\right]}{\alpha^{2H}} \tag{2}
\]

\[
R_X(t,s) = \frac{R_X(\alpha t,\alpha s)}{\alpha^{2H}} \tag{3}
\]

Assume that the time series \( X_n \) represents the data amount of arrival or departure at No. \( n \) time slice, and conduct stacking on \( X_n \). The sequence generated by stacking is \( X_{\alpha(n,m)} \), which indicates that the original time series are divided into a plurality of non-overlapping data segments of the size \( m \). And then use Formula (4) to respectively calculate the sequences obtained from the average and compression of original sequence on the time scale, and stationary random time series have the relation shown in Formula (5) – (8).

\[
X_{n}^{m} = \frac{1}{m} \sum_{j=m(n-m)+1}^{mn} X_j \tag{4}
\]
\[
\bar{X} = E(X_n) = \frac{1}{n} \sum_{i=1}^{n} X_i
\]  

(5) 

\[
\text{Var}(X_n) = \frac{1}{n} E[(X_n - \bar{X})^2]
\]  

(6) 

\[
\text{Var}(X_n) \approx \frac{1 + \delta_n(R)}{n} E[(X_n - \bar{X})^2]
\]  

(7) 

\[
\delta_n(R) = \frac{1}{n} \sum_{i \neq j} R(i, j)
\]  

(8) 

\[
R(i, j) = \frac{E[(X_j - \bar{X})(X_j - \bar{X})]}{E[(X_n - \bar{X})^2]}
\]  

(9) 

The flow values obtained in the collection of actual flow data are all discrete, so discrete time series are applied to represent the network flow. If the discrete random sequence \(X_n\) is strictly self-similar, it satisfies Formula (9) and Formula (10); if the discrete random sequence \(X_n\) is progressively self-similar, it satisfies Formula (9) and Formula (11) when \(k \to +\infty\). If the autocorrelation function of \(X_n\) satisfies Formula (12), the discrete random sequence \(X_n\) is called strict second-order self-similarity. If the autocorrelation function of \(X_n\) satisfies Formula (13), the discrete random sequence \(X_n\) is called progressive second-order self-similarity[7].

\[
\text{Var}(X_n^m) = \frac{\text{Var}(X_n)}{m^{2H}}
\]  

(9) 

\[
R(k, X_n^m) = R(k, X_n)
\]  

(10) 

\[
R(k, X_n^m) \to R(k, X_n)
\]  

(11) 

\[
R(k) = \frac{\text{Var}(X_n)}{2} \left[(k + 1)^{2H} - 2k^{2H} + (k + 1)^{2H}\right]
\]  

(12) 

\[
\lim_{m \to \infty} R(m)(k) = \frac{\text{Var}(X_n)}{2} \left[(k + 1)^{2H} - 2k^{2H} + (k + 1)^{2H}\right]
\]  

(13)

### 2.2 Recognition algorithm of the unknown flow

The key of flow monitoring system is to identify the unknown flow, so this paper uses pattern matching algorithm to ensure the accurate recognition of the unknown flow. And pattern matching is described below.

For the given \(q(q \geq 1)\) pattern strings (characteristic strings) \(\text{Pattern} = \{P_1, \cdots, P_q\}\), the length of pattern string are respectively \(L_1, \cdots, L_q\), and there are text strings \(\text{String} = \{S_1, \cdots, S_N\}\) with the arbitrary length of \(N\), and \(N \geq L_1\). Pattern matching is to match pattern strings in the text string, and if any substrings in the pattern strings are matched in the text string, the substring matching is successful, otherwise, the matching fails.

PM algorithm has single-PM algorithm and multiple PM algorithm, the former refers to that only one pattern string is matched after searching the text string once, such as BF algorithm[8], KMP algorithm[9], BM algorithm[10] and BMH algorithm[11]; the latter refers to that multiple pattern strings are matched after searching the text string once, such as AC algorithm[12] and AC_BM algorithm[13]. This study applies multiple-PM algorithm and describes AC_BM algorithm.

The realization of AC_BM pattern matching algorithm requires pretreatment and matching search. In the first step, the given pattern string is established into a pattern tree based on the prefix, and bad character table and good prefix table are established based on the calculation of pattern tree. In the second step, if there are non-matched characters, relatively use bad character table and good prefix table based on the adaptive characters and matched prefixes, and move the ones with long jump distance. Therefore, the realization of AC_BM pattern algorithm requires three steps: establishing pattern tree based on the prefix, establishing bad character table and good prefix table, and matching search.

**STEP1.** Establish pattern tree based on the prefix: text string \(\text{str} = \{a b c e a b b a b e d b a b\}\) is given, 4 pattern strings \(\text{Pat} = \{a b c d, a b b a b, a b c e, a b a c\}\) are respectively denoted as \(\{P_1, P_2, P_3, P_4\}\), the length of pattern string are respectively \(L_1, L_2, L_3, L_4\), and the shortest pattern string length is denoted as \(L_k\). According to the given pattern string and referring to the principle of AC automaton[12], pattern tree based on the prefix is established, as is shown in Figure 1.

**STEP2.** Establish bad character table and good prefix table: the skip rule of bad characters and good prefixes can refer to Formula (14) and (15); as AC_BM algorithm is a multi-pattern matching, the character is corresponding to the skip distance of different bad characters.
and good prefixes at different depths; the moving distance of pattern tree cannot exceed the minimum pattern string length \( \min\{L_k\} \); check the characters in the pattern string with the method of Formula (15) and complete the statistics of good prefixes.

Formula (14) and (15) are moving distance equations.

When \( \text{deep} < \left\lfloor \frac{\text{max}\{L_k\}}{2} \right\rfloor \), \( \text{deep} = 1, \text{shift(a)} = 2, \text{deep} = 2, \text{shift(ab)} = 3 \), the skip distance of bad characters of the pattern string shown in Table 1 at different depths can be obtained according to Formula (14).

\[
\text{shift}_{\text{hm, bc}}(j) = \begin{cases} 
  j & S_i \neq P_j \text{ and } S_i \notin \text{pat} \\
  j - k & S_i \neq P_j, \text{ but } \forall P_k = S_i \text{ and } 0 < k < j 
\end{cases}
\]  

(14)

\[
\text{shift}_{\text{hm, gc}}(j) = \min\left\{ P_{j+1} \cdots P_{m} = P_{j-s+1} \cdots P_{m-s}, \text{ and } P_{j-s} \neq P_j \right\} 
\]

(15)

The table below shows the distance of non-matched characters at different depths.

<table>
<thead>
<tr>
<th>Depth</th>
<th>Character</th>
<th>Character</th>
<th>Character</th>
<th>Character</th>
<th>Other Characters</th>
</tr>
</thead>
<tbody>
<tr>
<td>deep=0</td>
<td>/</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>deep=1</td>
<td>1</td>
<td>/</td>
<td>1</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>deep=2</td>
<td>/</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>deep=3</td>
<td>/</td>
<td>1</td>
<td>/</td>
<td>/</td>
<td>1</td>
</tr>
<tr>
<td>deep=4</td>
<td>1</td>
<td>/</td>
<td>1</td>
<td>1</td>
<td>/</td>
</tr>
</tbody>
</table>

STEP3. Matching search: the first match is shown in Figure 2, align the terminal character of the shortest pattern string with the bottom character of the text string, and start the comparison from the node character of pattern root to right side. First, \( a \neq d \), pattern tree moves three steps to the left according to the skip(d)=3 of bad character \( d \), making the character \( d \) align at the \( d \) appearing the earliest in the pattern string. The second match is shown in Figure 3, restart the comparison from the root, "ab" characters can match, the character \( e \) fails in the matching at the depth of 2 of the pattern tree. According to the bad character table, when deep=2, skip(e)=1. According to the skip rule of good prefixes, skip (ab) = 3, longer distance skip = 3 is applied to conduct pattern tree moving. The third match is shown in Figure 4, the match is successful based on the successive comparison, and the output match result is abbab.

AC_BM algorithm has the feature of AC algorithm that can match a plurality of pattern strings at one time, and also learn from the advantage of BM algorithm, skipping movement distance, so that it can effectively reduce the times of comparison between characters and improve the matching speed under the condition that most pattern strings have the same prefix.

3. Analysis on the demand of software design

Android phone of 4G network has fast network accessing speed and timely response. And because of this, this type of mobile phone will cost more flow while accessing network, often beyond the user's flow plan, resulting in unnecessary charges. Meanwhile, some
software of Android phone will automatically connect to network, generating a lot of flow waste. Therefore, the users’ need for scientific and rational phone flow monitoring software is quite strong.

Research methods of users can refer to Figure 5, and through using user research methods shown in Figure 5 to analyze the typical users, we can know that the causes for users’ concern of the flow are that they use software costing too much flow and some software’s malicious networking. The malicious networking mentioned in this study refers to that the software connects the advertising center without the user’s permission, downloads program package or advertisement recommended, and anonymously display them in front of the user. Therefore, the users’ demands for the flow monitoring software are mainly recording the flow using details, calculating the remaining flow and starting and stopping networking applications.

The contact between flow monitoring software and users can be divided into passive interaction and active interaction, and the interaction examples are shown in Figure 6.

4. Overall design and test of flow monitoring software

4.1 Overall structure design

The main function of flow monitoring software is to implement flow monitoring and controlling, and this monitoring and controlling need to realize the function of interaction with the users on the Android platform. Thus, the overall structure design of the flow monitoring software is shown in Figure 7.
The module controlling the flow is divided into two parts, namely Iptables module and software management, and its main function is to accept control commands from the user interface, transfer the commands and get the results of command execution. The basic data of monitoring the flow are derived from the underlying flow statistics of the system, and the observer pattern is used to provide data to various sides. Provide floating box in the user interface to show the current networking speed, and provide details of data record to management module, indirectly updating notification bar.

The data management module can provide details of some of the flow to the user interface. Reminding logical part can get the historical data of the flow from the data management module and integrate current total package, surplus flow and judgment on the days, to provide appropriate reminding to the users.

4.2 Testing environment of software

All software testing in this study is conducted on real testing machines, mobile phones of Android system. In these testing machines, we should verify the function and the screen fittings. And the model, resolution ratio, network type and system of the testing machines are shown in table 2.

<table>
<thead>
<tr>
<th>No.</th>
<th>Model</th>
<th>Resolution Ratio</th>
<th>Network Type</th>
<th>System</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Samsung I9100</td>
<td>480*800</td>
<td>WCDMA.GSM</td>
<td>2.3</td>
</tr>
<tr>
<td>2</td>
<td>Samsung Galaxy Ace</td>
<td>320*480</td>
<td>WCDMA.GSM</td>
<td>2.2</td>
</tr>
<tr>
<td>3</td>
<td>Motorola milestone 2</td>
<td>480*854</td>
<td>WCDMA.GSM</td>
<td>2.2</td>
</tr>
<tr>
<td>4</td>
<td>Motorola ME515</td>
<td>240*320</td>
<td>WCDMA.GSM</td>
<td>2.2</td>
</tr>
<tr>
<td>5</td>
<td>Motorola XT910</td>
<td>540*960</td>
<td>WCDMA.GSM</td>
<td>2.3</td>
</tr>
<tr>
<td>6</td>
<td>HTC One X</td>
<td>720*1280</td>
<td>WCDMA.GSM</td>
<td>4.0</td>
</tr>
<tr>
<td>7</td>
<td>Huawei C8812</td>
<td>480*800</td>
<td>CDMA2000</td>
<td>2.2</td>
</tr>
<tr>
<td>8</td>
<td>ZTE N880</td>
<td>480*800</td>
<td>CDMA2000</td>
<td>4.0</td>
</tr>
<tr>
<td>9</td>
<td>HTC Dream</td>
<td>320*480</td>
<td>WCDMA.GSM</td>
<td>1.6</td>
</tr>
</tbody>
</table>

4.3 Functional test and result analysis

Each application program is set to have two ways to access the network, one is wifi, the other is 4G, and the network accessing mode set in this test is 4G.

As is shown in table 3, flow monitoring software can realize the detailed identification on the network application flow, and that is to identify the flow of different scenes of network application as the corresponding sub-agreement according to corresponding protocols and rules. The average application recognition rate is about 95%, and the false alarm rate is 0, which means there is no false alarm of identifying application A as protocol B. Experimental results show that the flow monitoring software has good flow recognition effect and can realize the flow classification of various network applications.

<table>
<thead>
<tr>
<th>Application name</th>
<th>Protocol name</th>
<th>Recognition rate</th>
<th>substantial false alarm rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tencent QQ</td>
<td>QQ_IM</td>
<td>95.60%</td>
<td>0.00%</td>
</tr>
<tr>
<td></td>
<td>QQ_VoIP</td>
<td>90.06%</td>
<td>0.00%</td>
</tr>
<tr>
<td></td>
<td>QQ_FileTransfer</td>
<td>96.90%</td>
<td>0.00%</td>
</tr>
<tr>
<td>LaiWang</td>
<td>LaiWang</td>
<td>92.04%</td>
<td>0.00%</td>
</tr>
<tr>
<td>Skype</td>
<td>Skype</td>
<td>91.20%</td>
<td>0.00%</td>
</tr>
<tr>
<td>WeChat</td>
<td>WeiXin</td>
<td>90.08%</td>
<td>0.00%</td>
</tr>
<tr>
<td>Youku Video</td>
<td>YouKu</td>
<td>96.05%</td>
<td>0.00%</td>
</tr>
<tr>
<td>Popular film and television</td>
<td>FengXingVideo</td>
<td>98.30%</td>
<td>0.00%</td>
</tr>
<tr>
<td>PPTV</td>
<td>PPTV</td>
<td>97.90%</td>
<td>0.00%</td>
</tr>
<tr>
<td>Tencent Video</td>
<td>TencentVideo</td>
<td>96.50%</td>
<td>0.00%</td>
</tr>
</tbody>
</table>
### Table 1: Application, Protocol, and Recognition Rates

<table>
<thead>
<tr>
<th>Application name</th>
<th>Protocol name</th>
<th>Recognition rate</th>
<th>Substantial false alarm rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>PPS iqiyi video</td>
<td>IQIY1_PPS</td>
<td>98.00%</td>
<td>0.00%</td>
</tr>
<tr>
<td>360 Cloud Disk</td>
<td>360yunpan</td>
<td>96.70%</td>
<td>0.00%</td>
</tr>
<tr>
<td>QQ cyclone</td>
<td>QQDownload</td>
<td>99.00%</td>
<td>0.00%</td>
</tr>
<tr>
<td>Thunder</td>
<td>Thunder</td>
<td>97.06%</td>
<td>0.00%</td>
</tr>
<tr>
<td>Renren</td>
<td>Renren</td>
<td>96.56%</td>
<td>0.00%</td>
</tr>
<tr>
<td>CITIC Securities</td>
<td>Stock_ZhongXin</td>
<td>93.60%</td>
<td>0.00%</td>
</tr>
<tr>
<td>QQ Mailbox</td>
<td>Mail_QQ</td>
<td>95.70%</td>
<td>0.00%</td>
</tr>
</tbody>
</table>

The flow rate trend of the system during the testing time is shown in Figure 8, the solid line in the figure indicates the output of system flow, and the dotted line indicates the input of system flow. As can be seen from Figure 8, the highest flow transmission rate exceeds 5M before the controlling strategy is released, but the flow rate gradually decreases to 0 at 11:11 after the controlling strategy is released. The test results show that the flow monitoring software can quickly control the corresponding flow after the controlling strategy is released, thus achieving the purpose of blocking flow.

![Figure 8 Variation trend of flow rate during the testing time](image)

It can be known from the monitoring and controlling function of the flow monitoring software that the flow monitoring software of Android phone based on 4G network can perfectly satisfy the needs of users.

### 5. Conclusions

Surfing the Internet with mobile phone has been immersed in people's lives, and this application is constantly developing. The flow problem of the Internet surfing with mobile phone is still the main concern of the people, so that the development of flow monitoring software with high precision and practical value for mobile phone is urgently required. Algorithm design for flow monitoring software should consider both flow monitoring function and flow controlling function. As the key to flow monitoring technology is flow recognition, the paper focuses on the self-similarity principle of the flow and the unknown flow recognition algorithm based on AC BM pattern. According to customers’ demand for software, the paper designs the structure chart of flow monitoring software consisting ofiptables module and software management, and tests the monitoring function of the software. The test results show that the flow monitoring software is qualified with good flow recognition effect and can achieve the flow classification of various network applications; can quickly control the corresponding flow after the controlling strategy is released, thus achieving the purpose of blocking flow.

### Reference