A thesis submitted in partial satisfaction of the
requirements for the degree of
Master of Computer Science and Engineering
in the Graduate School of
the University of Aizu

Computer System Performance Observation through the Computer Network

by

Norimiti Mizutani

February, 2003

The thesis titled

Computer System Performance Observation through the Computer Network

by

Norimiti Mizutani

is reviewed and approved by:

Main referee	
Professor	Date
Atsushi Kara	
Associate Professor	Date
Takafumi Hayashi	
Assistant Professor	Date
Hirosi Toyoizumi	

The University of Aizu

February, 2003

Contents

List of Figures

List of Tables

Acknowledgement

I wish to express my appreciation to Professor Toyoizumi for his excellent advice and diligent efforts to guide me through this project.

I also express my gratitude for Professor Atusi Kara and Professor Takahumi Kobayasi who kindly read this draft and advise me.

Abstract

The computer system usually provides many methods to observe performance. One of the most popular ways is to login to the machine and use commands which the system provides to measure performance. We want to know performance of the system when we feel that the system slowly responds to our request. However, the system users do not want to place unnecessary load for running machine. Currently, the TCP/IP network has become popular and many service are provided through computer network using the network protocol. We may measure performance of the computer system through the network.

In this paper, we propose a method to observe performance of computer systems on the network from other machine. The computer network is structured by many nodes. The proposed method is to understand which target machine has trouble or not.

ICMP protocol is used for the method because the protocol provides some measurement functions. ICMP protocol equips with two functions called ICMP echo request and reply to measure round trip time (RTT). The most popular tool using the functions is PING. However, the result includes the network delay from measuring machine to the target server and time spent in the target server. We can not find where is trouble. We can not understand which target machine is busy or other nodes on the network pass to target machine have a trouble. Thus, we will handle the ICMP functions with some idea and observe performance of the target system using statistical calculation.

Since we use the computer network and statistical calculation to observe performance of the target system, we decide performance of the target system after few minutes since we started to observe. The time difference must be as small as possible. And, if observed results are unusual, we want a caution at once.

The proposed method is a type of active measurement. We need to be careful that the method is little load for target machine and the network.

We make a tool which follows the above policy to observe the state of the machine. The tool records date when unusual results were replied. We use the tool on the public network and a test bed network which was disconnected from the public network. We could get some results. We will verify the results discuss whether our method is correct, or not. And, when our method is correct, we search some point of improvement. When we can not get correct results, we must examine the problems.

Chapter 1

Introduction

This research objective is to make a tool that we observe the performance of the target machine from another machine on the computer network.

The motive is that author wanted to detect Denial of Service (DoS) attack. DoS attack is a method to crack the computer system. DoS attack makes the target machine accept too many exceptional accesses intentionally. [1] The computer systems provide many services. However, if they are asked too many exceptional services, their services are stopped or their availability for ordinary accesses fall off seriously. In October 21, 2002, all thirteen DNS route servers on the Internet attacked by this way at same time. [2] And nine route servers of Tokyo, Stockholm and America had some troubles for some hour. Route servers manage route domain of the Internet; .com, .jp, .uk and etc. General users were not in trouble by the attack because routers used the cache of domain name. After the attack, a route server in America has been replaced to protect next attacks.

DoS attacks fall off the availability of the system. Then, if we can observe the performance of the computer systems, we may detect DoS attack.

Almost all computer systems provide at least a function to estimate performance on the system. However, we do not accept the observation by those functions on each machines because we do not want to apply unnecessary load to the machine. When we use the functions, we usually need to login in each running machines and use the systems to estimate the performance on the busy machines. In addition, almost all current systems are made from many machines. Management will be very hard if we use the estimating functions on each machine. Suppose that the machines are connected through the computer network, we want to observe the performance on one measuring machine through the network. Thus, the author decides that our method must observe the performance of the target system through the network.

There are other methods to observe the performance of the systems. We can order the systems to record the logs of the systems state. When the systems are not busy, we analyze the logs and we can get detail of the machine state. When we want to understand the reasons of the problems, the method is most effective. However, the method need long interval to start log analysis. If we want to know current state, we must develop other method. Thus, we ignore to detect the problems' causes and pursue

only the observation of the performance through the network.

When we want to estimate the state of the network accepted TCP/IP, we usually use the ICMP functions. A most major tool is PING. That investigates whether target is reachable or not using ICMP echo request from measuring machine and ICMP echo reply from target machine. [3] With using some optional functions, we can get round trip time, RTT. RTT is a value. When we get some RTT by PING, we will find that there are some alteration between each results. However, general PING do not solve which network nodes have troubles. ICMP provides other functions which reply time stamp when the packet arrived at and left from the destination . The author tried to use the time stamp by using these functions and knew that a few systems reply the correct time stamp. Thus, we want to use ICMP echo request and reply for the observation because almost all systems provides the functions. We will make over ordinary PING.

The measurement method using made over PING is classified into the active measurement [4] . There are two types of the measurement methods through the network, active and passive. The passive measurements estimate the network state from packet arrival at the end system. The passive measurements do not act to measure the state to the network state. Thus, the methods do not change the state of the network. Sniffer and Snort are most popular tools [5] [6] . The active measurements apply some alteration to the network state because the methods throw some probes into the network and observe from the replies of the probes. As amount of the probes, the network behavior will be altered from the origin of the network. Thus, as amount of the probes is fewer, the alteration of the network is less. Made over PING must be little. General PING uses little packet. Then, PING will add the little alteration to the network state and the destination system. Made Over PING do not have to lose the advantage of PING. In addition, as if we can measure correctly, times throwing the packets must be few as possible.

The procedure of the observation is following. First, we throw the probes and get RTT. Second, when we collect the appropriate sets of RTT, we process the sets of RTT and make time spent in the destination and a threshold to decide the state. Third, we evaluate the data by the threshold. If there are time spent which is larger than the threshold, we make our tool report date and return to the first step. To make time spent in the destination and threshold, we use some assumptions to packet forwarding of the network nodes and a packet processing in the destination machine. The measuring tool will be inspected by the examination in a test bed network and the Internet.

Chapter 2

Environment for Experiment

In this section, we will show two computer network environments arranged for the experiment observing the state of the target system. One is a independent test bed network separated from any other networks. The test bed network is used for this experiment only. Another network to try to observe the state of the target system is the Internet. We will choose a site as a target system.

We make and use a measuring tool to observe the target state. We apply two assumptions to the target system. One is that the target system accepts First In First Out (FIFO) queuing system. Second is that the target system is made by one queue and one server. Each assumptions are true in the test bed network. However, second assumption is not true about the target site in the Internet because the site is a very large system and the author know that the site is made from many machines.

2.1 Test Bed Network

Following figure 2.1 shows the environment of the test bed network.

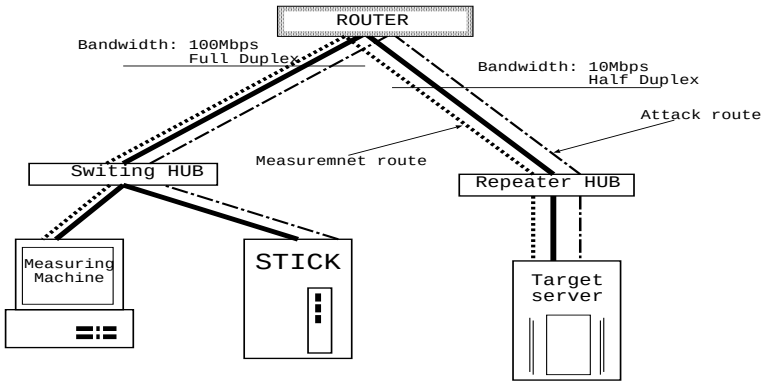


Figure 2.1: Test Bed Network

The test bed network is made from two subnetworks. A router connects them. In one subnetwork, all machines are connected by half duplex 10 mega bps LAN using repeat HUB. In another subnetwork all machines are connected by full duplex 100 mega

bps LAN using switching HUB. In 100M LAN network, there are two machines for the observation and changing the state of the target machine. The observing system runs on the one Linux machine. We take some RTT using the machine and distinguish the unusual state from the ordinary state by the analysis of the RTT. Into another machine, the author installed a attack system ,STICK, to change the state of the target machine. [7] . STICK throws too many packets to the target system. In 10M LAN network, there is only a target machine used for a intrusion detection server, IDS, on Windows 2000. The IDS falls off the performance of the machine seriously when the machine accepts too many packets. Following graph shows the percentage how CPU and Memory are used. Above graph shows the state of CPU. In the graph, the percentage of CPU is grown up at center by the attack.

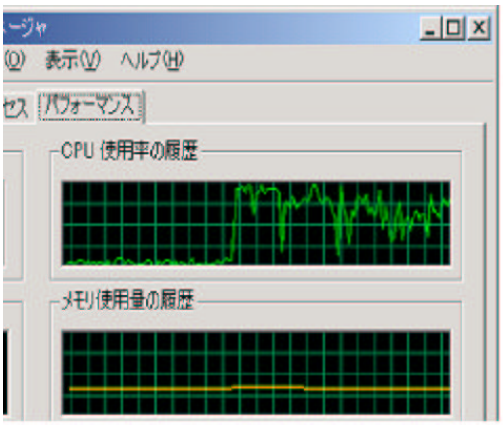


Figure 2.2: Performance of CPU and Memory

We take turns with beginning and stopping the attack by STICK and we change the target performance artificially. We will try to detect the change.

2.2 The Internet

We try to observe the state of a web site on the Internet, www.2ch.net. The web site is one of the largest sites of Bulletin Board System (BBS) in the world. The servers are in USA. There are twenty nodes of the network from the measuring machine to the target web site.

At first, we will make sure that the measuring method is effective on the test bed network. Then, when the method is effective, we will try to observe the state of the target site on the Internet and make sure that the measuring method is effective on the public computer network.

In the next chapter, we discuss about a idea to measure the the state of the target machine. We will try to use the tool in the above environment.

Chapter 3

Mathematical Model

3.1 Round Trip Time

We use Round Trip Time (RTT) to observe the state of the target system because RTT changes the length effected by the state of the target system. RTT is a value which shows time spent to leave a packet from the measuring machine to the destination and to return from the destination to the measuring machine. To measure RTT, we use PING. PING makes a ICMP echo request packet and send the packet to the destination. PING records when the measuring machine made the packet. Then, the destination accepts the ICMP echo request packet and replies the ICMP echo reply packet to the sender. The sender accepts the replied packet and record the time after the measuring machine finished to catch all data of the packet. Then, PING gives RTT as the difference between the sending and the accepted time. RTT is the sum of waiting time for serving other packets, service time of itself, sending times of the PING packet to other nodes which we call network delay and time spent to finish to forward and accept datagram of the packet on the sender and the destination machine which is called propagation delay.

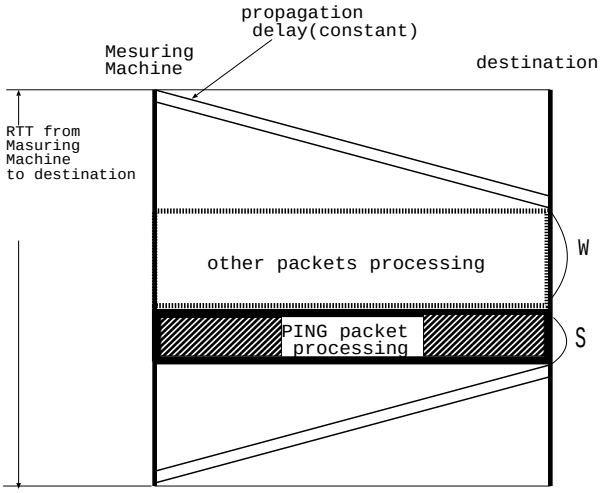


Figure 3.1: Round Trip Time

$$RTT = ND + w + s + p \tag{3.1}$$

The equation 3.1 shows the components of RTT. ND is network delay, w is waiting time, s is service time and p is propagation delay. Waiting time is effected the length by the state of the destination. When the destination accepts many packets, calculates many processes or pause CPU by some reasons, the destination may wait the packet for long time to start the packet processing. When the destination is not busy, waiting time is less. We will use waiting time to estimate the state of the target of the machine. Thus, we need a method to distinguish waiting time from RTT.

Changing equation 3.1, we can get waiting time from following.

$$w = RTT - ND - s - p \tag{3.2}$$

This equation 3.2 shows that we must calculate network delay, propagation delay and service time to get waiting time. We must need service time as only itself. However, we do not need to know each time, only ND and only p. The author will show the reason in the following sections.

3.2 Idea to estimate waiting time

We show a idea that we get network delay, propagation delay and service time. The idea is that we send two consecutive packets to the destination to measure RTT. The next figure 3.2 shows the idea and the figure 3.3 shows RTT of the made over PING

according to the idea.

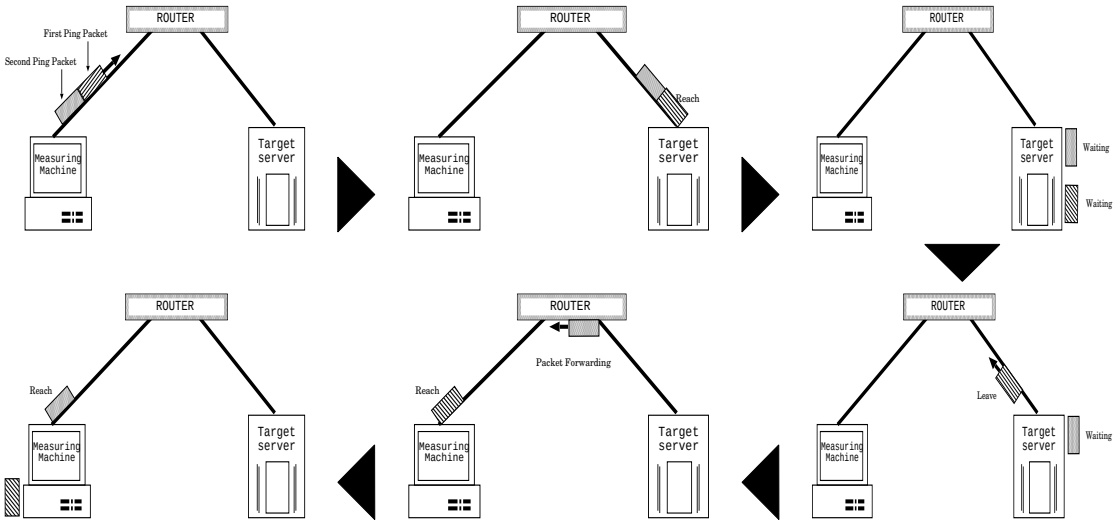


Figure 3.2: Idea

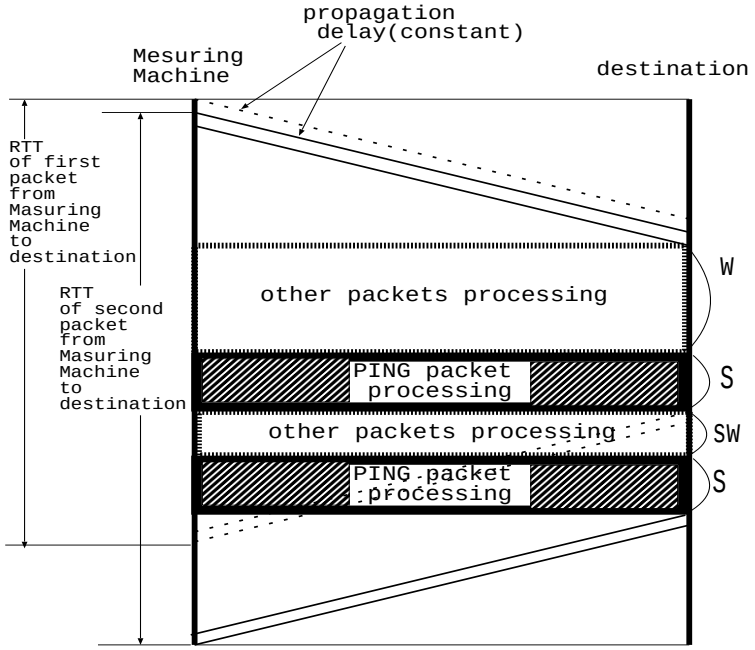


Figure 3.3: RTT of the made over PING

We can get two RTT using this made over PING according to the idea. We estimate

waiting time using the two RTT. The author write RTT for first packet $FirstRTT$ and RTT for second packet $SecondRTT$.

Applying two assumptions to the RTT, we get two equations to calculate network delay, propagation delay and service time.

$A1$ Network delay contained to two RTT is same time.

$A2$ Service time contained to two RTT is same time.

$A3$ Propagation delay contained to two RTT is same time.

The reason of $A1$ is that the interval between two packet is a little. From $A1$, the order of two packets is never reversed on the way. Two packets have same length and make the destination process same calculations. Thus, we can assume $A2$ and $A3$. Using these assumption, we can show two RTT as following.

$$FirstRTT = ND + w + s + p \tag{3.3}$$

$$SecondRTT = ND + w + s + sw + s + p \tag{3.4}$$

sw is waiting time for second packet after the target machine finished to process first packet.

We use this made over PING several times and get a set of two data made from two RTT. We apply other two assumptions to the set of data and estimate network delay , propagation delay and service time. If we can get network delay, propagation delay and service time, waiting time will be estimated.

3.3 The Assumptions to The Sets of RTT

If the number of the sets made from two RTT is much enough, there are at least one of first RTT and one of second RTT which satisfies following two assumptions.

$A4$ There is a set of RTT when no any other process is caught between first packet and second packet process.

$A5$ There is a set of RTT that first packet arrived at the destination machine and the destination machine immediately started to process the second packet.

When $A4$ is true, sw is equal to zero and we can get the following equation.

$$SecondRTT = ND + w + s + s + p \tag{3.5}$$

When $A5$ is true, w is equal to zero and we can get following equation.

$$FirstRTT = ND + s + p \tag{3.6}$$

Thus, if $A1$, $A2$, $A3$ and $A4$ are true for the collected sets, we can calculate service time by second RTT minus first RTT of a same set.

$$\begin{aligned} SecondRTT - FirstRTT &= ND + w + s + s + p - (ND + w + s + p) \\ &= s \end{aligned}$$

When we were going to calculate service time, s, if $A1$, $A2$, $A3$, $A4$ and $A5$ are true for the collected sets, we can calculate network delay plus propagation delay by first RTT minus s.

$$\begin{aligned} FirstRTT - s &= ND + s + p - s \\ &= ND + p \end{aligned}$$

When a set from the collected RTT sets satisfies the assumptions, $A1$, $A2$, $A3$ and $A4$, the equation 3.7 is changed to following,

$$s = Min[SecondRTT - FirstRTT] \tag{3.7}$$

Then, we can get service time and we can estimate network delay plus propagation delay using the service time. When a set from the collected RTT sets satisfies the assumptions, $A1$, $A2$, $A3$, $A4$ and $A5$, the equation 3.7 is changed to following,

$$ND + p = Min[FirstRTT] - s \tag{3.8}$$

Thus, we can get network delay plus propagation delay, $ND + p$, and service time, s, from the equations, 3.7 and 3.8. We can estimate waiting time from the equation 3.2, $ND + p$ and s.

In the next section, the author will explain the method to make a threshold to evaluate the estimated waiting times.

3.4 Threshold

We showed that we used waiting time to estimate the state of the target machine and the method making the waiting times. All waiting times have the individual value. If there are too large waiting time, we want to know when the waiting time was measured. However, how long is the length of waiting time as we can recognize that the waiting time is too long? We need a threshold to decide whether the waiting time is too long or not. If we make the threshold, we will estimate the state of machine.

We assume that the distribution of waiting time is the exponential distribution to make the threshold. Next, we show the method to make the threshold if the distribution

of waiting time is the exponential distribution.

When the distribution of waiting time is the exponential distribution, the probability, P, that waiting time of a packet is less than a value, x, is explained by the following equation.

$$P = 1 - \ell^{\lambda * x} \tag{3.9}$$

This equation explain the distribution of waiting time when the average of waiting time is $1/\lambda$.

$$\lambda = 1/aw \tag{3.10}$$

aw is the average of waiting time to the system. However, aw is not the usual average as the sum of all data divided by the number of the data. We want to know whether current waiting time is too different from ordinary waiting time or not. When we calculate the average of all waiting time contained current waiting time, if the target machine is too busy and reply after waiting long time, the waiting time may raise the average of waiting time. Then, the threshold is raised and we may not detect the unusual state of the target machine. Thus, we must calculate the average of waiting time which is effected by old waiting times strongly. The author will show a idea and the method in the next section.

The threshold can calculate from the equation 3.9 solved about x.

$$\begin{aligned}
P &= 1 - \ell^{\lambda * x} \\
P - 1 &= -\ell^{\lambda * x} \\
1 - P &= \ell^{\lambda * x} \\
\log(1 - P) &= \log e^{\lambda * x} \\
\log(1 - P) &= \lambda * x \\
x &= \frac{\log(1 - P)}{\lambda}
\end{aligned}$$

Changing x to T,

$$T = \frac{\log(1 - P)}{\lambda}$$

T expresses the threshold when the probability that the target state is usual is P, $0 < P < 1$. We call this P threshold rate. When we substitute a value to λ and P, we can get a value, T. T shows the threshold that waiting time is less than the threshold at the probability P when the average of waiting time is $1/\lambda$. When waiting time is larger than T, we regard the state of the target machine as unusual when the waiting time was measured.

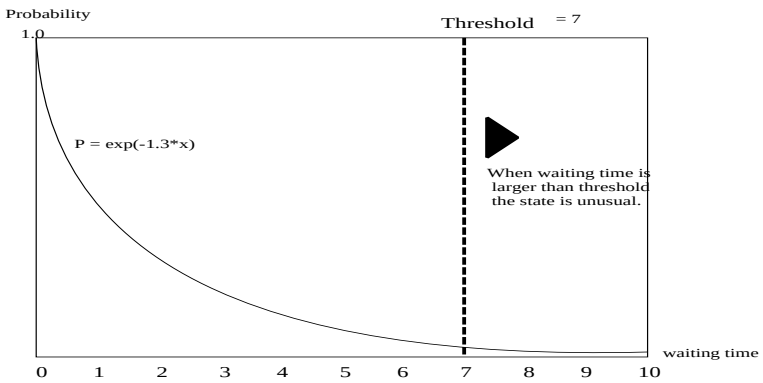


Figure 3.4: A Example of Exponential Distribution and Threshold

The author made a tool to measure RTT and to make the threshold for this research. The tool estimates waiting time based by the equations and the assumptions in the section 3.2 and 3.3. And, the tool calculates the threshold from waiting times using the equations and the assumptions in the section 3.4. The tool compares the threshold and waiting time and, if waiting time is larger than the threshold, the tool reports when the waiting time was measured. In the next chapter, we show the algorithms and all steps of the tool.

Chapter 4

Tool to Observe The State of Target

4.1 Algorithm of Tool

We show the basic ideas to estimate waiting time and the threshold in the above sections. And we make some equations and assumptions to realize the ideas to estimate the state of the target machine. We will discuss the algorithm and a tool to get waiting time and the threshold using the ideas in this chapter. The author made a tool to estimate the state of the target machine to realize the following algorithms which the equations and the assumptions introduced in the above chapters.

1. Get 20 sets made by two RTT from the target machine using made over PING.

Table 4.1: A Example of the RTT set

The number of times	First RTT	Second RTT
1st	1.3 msec	1.5 msec
2nd	1.7 msec	1.8 msec
3rd	0.9 msec	2.0 msec
⋮	⋮	⋮
18th	1.5 msec	1.9 msec
19th	0.8 msec	1.0 msec
20th	1.3 msec	1.7 msec

2. Apply the results of first step to the assumptions, $A1$, $A2$, $A3$ and $A4$, and calculate service time using the equation 3.7.

3. Apply the results of first and second step to the assumptions, $A1$, $A2$, $A3$, $A4$ and $A5$. and calculate network delay plus propagation delay using the equation 3.8.

4. Using service time and network delay plus propagation delay as the results of the second and third steps, calculate waiting times from first RTT of the first step results using the equation 3.2.

5. Calculate the average of this cycle waiting time using the results of the forth step.

6. Calculate the average of all cycles' waiting time using the results of the fifth step and the results of the sixth step of the once before cycle.

7. Calculate λ using the equation 3.10.

8. Using the equation 3.11 and the result of the seventh step, substitute a value to P and calculate the threshold for this cycle.

9. Compare waiting time with the threshold. When waiting time is larger than the threshold, this tool records the date when the waiting time was measured.

10. Repeat all steps from the first step.

We repeat above steps to collect data and to evaluate the state of the target machine. After finished one cycle, we will find some date when the target system has the unusual state. When we can not find date, the target system did not have any troubles for the cycle. The measuring tool sometime makes a mistake to decide the state of the target system caused by the way of the threshold estimation. We will discuss about the problem in the next chapter.

4.2 Cautionary Remarks about Tool

There are four remarks about th measuring tool and the experiment.

4.2.1 Service Time

First, we apply the assumptions, $A1$, $A2$, $A3$ and $A4$, to result of first step. Thus, we assume that there is at least one of the RTT sets that there is not waiting time between the first packet and second packet processing in the target machine and that service time is fixed. We use these assumptions and estimate service time from the twenty RTT sets of current cycle. We think that there is doubtful in the assumptions. The probability that there are no set that RTT satisfies the assumptions, $A1$, $A2$, $A3$ and $A4$, is large. We might choose the least first RTT from whole cycle first RTT to estimate service time. However, the author have known that there were some results of PING which was nearly zero or too little after the author tried to use normal PING. We can not accept the idea. To estimate more correct service time, the author decide to use the average of service time which is calculated from service time of all cycles contained current cycle.

$$AST = \frac{ASO * (TN - 1) + SCT}{TN} \tag{4.1}$$

AST is the average of service time. ASO is the average of service time which is calculated from all cycle except current cycle. ASO was ASO in once before cycle.

SCT is service time which is estimated from the RTT sets of current cycle. TN is the number of all cycle.

4.2.2 Network Delay

Second, we apply the assumptions, $A1$, $A2$, $A3$, $A4$ and $A5$ to the results of the second step. Thus, there is at least one of twenty First RTT that there is not waiting time and we estimate network delay plus propagation delay using the RTT. However, network delay is different from each other packet essentially. The measuring tool takes twenty RTT sets for few minutes. We assume that network delay have same length in the same cycle.

From the section 4.2.1 and 4.2.2, we change s to AST in the equation 3.2.

$$w = RTT - (ND + p) - AST \tag{4.2}$$

4.2.3 Packet Loss

The packets in the network may be sometime lost by any reason. If ICMP packet is lost, the sender does not make same packet and send new packet and target machine does not request to send the packet once more to the sender because ICMP packet is a kind of UDP packet [3] . When the packet is lost, we can not know RTT. As the measuring tool uses RTT, the tool can not decide the state of the target machine when the tool fails to take RTT. We can not know whether the packet is lost or waited for too long time and we abandon to wait reply from the destination after a fixed time. The abandonment is called Time Out. The target machine has some troubles and we lost all sent the packets. However, we can not understand the reason of Time Out. We will send a new packet to the target machine every Time Out. However, if we need to wait for long time to recover the target machine, we have a problem about the network delay assumption because the network state may be changed. We can not wait so long. Thus, after Time Out is happened at three times, we abandon to take the RTT set. The measuring tool calculates the average of all first RTT and all second RTT and substitutes the averages of each RTT to the set of RTT and take next RTT set.

4.3 λ Calculation

In step 6, we calculate the average of waiting time. We make a threshold from the average. We want to know the threshold to decide current state to the ordinary state or the busy state. For example, for we use the tool, we take many short RTT sets of the cycle but ,in next cycle, RTT become longer. If we calculate the average of waiting time by the usual way as dividing the sum of all RTT by the number of all RTT, the current state may influence the threshold strongly. Thus, we must use the method that the past state influences the threshold more appropriately. We use the following equation to calculate the average of waiting time.

$$AW = (1 - a) * CW + a * OW \tag{4.3}$$

AW is the average of waiting time. OW is the average of waiting time in once before cycle calculated by this equation 4.3. CW is the average of waiting time of only current cycle. Then, $0 < a < 1$. a is the rate which shows the strength of the current cycle state effect. When a is little, the current cycle state influences the threshold strongly. When a is large, the current cycle state influence to threshold little. However, if a is too little and many large RTT were measured in few cycle, the measuring tool turns out not to regard the states as the unusual states after warning the unusual states a few time. The measuring tool sometime asserts the unusual state to be the ordinary state or the unusual state to be the ordinary state because there are some noise in RTT. We can understand that the target machine has some problems after we knew the number of the unusual state. Then, we will have a trouble by few caution.

However, as the unusual states are continued, we regard that the state is ordinary. Then, we will have a trouble that the cautions are continued for too long time because the basis access of the target machine may be changed . When a is too large, we will find the trouble. We must regulate a appropriately. The author used 0.8 for a in this experiment.
We call this a the effect rate of past waiting time.

Chapter 5

Results

The author tried to observe the state of the target machine by the measuring tool introduced in the chapter 4. There are the target machines on the test bed network and the Internet. We showed the details of the target machines and the environment in the chapter 2. In the following sections, we discuss about the results of the experiments and want to find new objectives to observe the state of the target machine more correctly.

5.1 Results on the Test Bet Network

5.1.1 Cases

The objective of the experiment on the test bed network is to make sure whether the measuring tool is effective to observe the state of the target machine or not. We experimented to observe the state of the target machine in the following three cases.

$C1$ The target is in the ordinary state.

$C2$ The target is attacked by STICK for we experiment.

$C3$ The target is thrown into the ordinary state and the attacked state in turn.

We experiment in the case $C1$ to understand the ordinary state of the target machine in the test bed network. In the case $C2$, the measuring tool decides that the target machine is ordinary because the target machine has been attacked for we experiment and the measuring tool is to detect the change from the ordinary state to the unusual state of the target machine. In the case $C3$, we inspect the measuring tool which can detect the state of the target machine correctly. We will throw the target machine into the ordinary state and the attacked state after each one hour alternately.

From above three cases, we must make sure that the results are correct, the assumptions that the distribution of waiting time is exponential distribution is correct and the percentage of the mistakes of the decision is less than one minus the threshold rate in the ordinary state.

5.1.2 Results of three cases

These following graphs show the results of three experiments on the test bed network. The threshold rate is 0.96. The effect rate of past waiting time is 0.8.

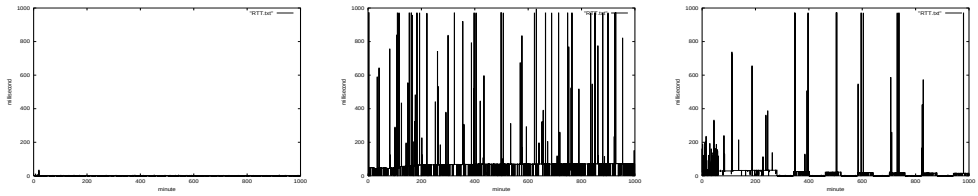
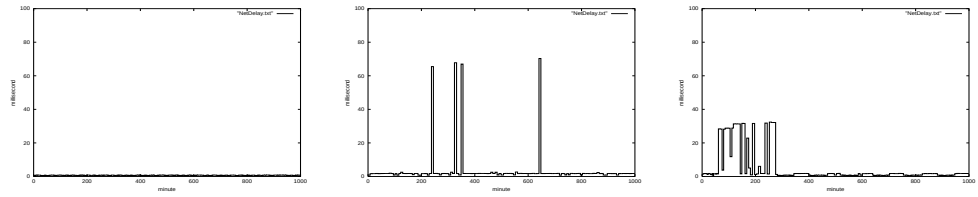
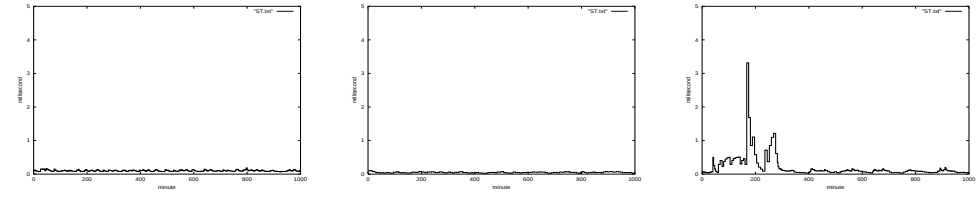
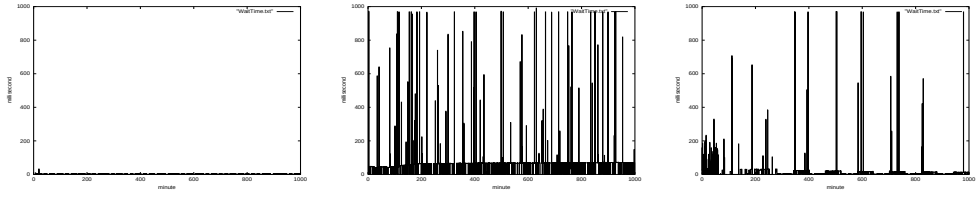
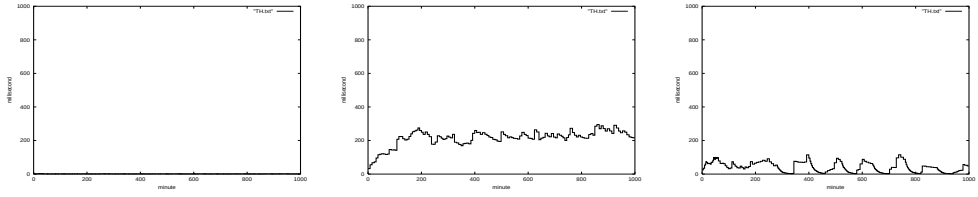
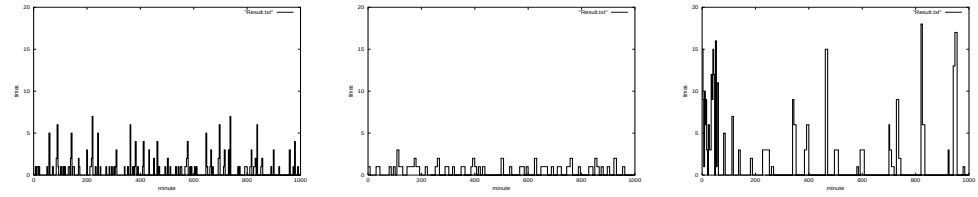
the type of graph	Ordinary	Attacked	Ordinary and Attacked
$Frist\ RTT$			
$ND + p$			
$Service\ Time$			
$Waiting\ Time$			
$Threshold$			
$Result$			

Figure 5.1: Table of Results

In above table, the graphs in the left column show the results of the ordinary state of the target machine, the graphs in the center column show the results of the attacked

state and the graphs in the right column show the results mixed the ordinary and attacked state. The author introduces what each graphs in each lines of the table are, from first to last line, $firstRTT$, $ND + p$, $service\ time$, $threshold$, $waiting\ time$, $the\ number\ of\ the\ cautions\ at\ each\ cycles$. The unit of x axis is minute and y axis is millisecond. We experimented to observe the state of the target machine for 1000 minutes and the maximum value of $firstRTT$, $waiting\ time$ and $Threshold$ is 1000 millisecond. The maximum value of $ND + p$ is 100 millisecond in the graph. The maximum value of $service\ time$ is 5 millisecond in the graph. $ND + p$ and $service\ time$ can became 1000 millisecond but, if we draw their graphs as the maximum value of y axis is 1000 millisecond, we hardly understand their behavior from the graphs because they are less than 1000 millisecond. The maximum value of $Result\ of\ the\ cautions$ is twenty times.

About RTT

In the ordinary case $C1$, almost all first RTT are between 0.8 to 2.0 millisecond. Especially, we got about 0.9 and 1.8 millisecond in turn at many times. The reason is unknown. However, when we use PING to the target machine too many times, that is happened. There are sometime large first RTTs. The largest first RTT is 33 millisecond.

The feature point of first RTT in the attacked case $C2$ is many packet loss. The number of the lost packets is 12432 packets. And the number of Time Out is 3716 in 4708 times of first RTT measurement. When we could get a set of RTT, the set of RTT was similar to the set of RTT of the case $C1$ or nearly 1000 millisecond. When the measuring tool abandons to measure RTT by Time Out, we substitute the average of each RTT. Then, almost all average of first RTT is nearly 65 millisecond by the substitutions.

In the case $C3$, we could get RTT in the ordinary states and many Time Out are happened in the attacked states. But we often can get large RTT similar to RTT of the case $C2$ in the attacked states. We see that the state of the target machine is separated to before 300 minute and after 300 minute. We get large RTT till about 100 minute normally and ,between about 100 to 300 minute, many Time Out are happened when we stop the attack to the target machine. The reason is unknown.

About Network Delay plus Propagation Delay

In the case $C1$, almost all $ND + p$ is between 0.5 to 0.8 millisecond.

In the case $C2$, almost all $ND + p$ is between 1.7 to 2.0 millisecond. We sometime estimate $ND + p$ nearly 0.8 millisecond when one of twenty first RTT in a cycle is at least similar to first RTT of the ordinary state. And there are four large $ND + p$ in the case $C2$. We estimate the large $ND + p$ when the measuring tool got twenty first RTT at the cycle and the RTT are the average of waiting time or larger than the average of waiting time. Same troubles are happened from 100 to 300 minutes in the case $C3$. At the other parts of the case $C3$ from the above parts, for we had stopped attack to target machine, $ND + p$ had been similar to $ND + p$ of the case $C1$ and ,for target machine had been attacked, $ND + p$ had been similar to $ND + p$ of the case $C2$.

About Service Time

In the case $C1$, almost all service time is between 0.085 to 0.12 millisecond.

In the case $C2$, almost all service time is between 0.03 to 0.05 millisecond. These service time are less than service time of the case $C1$. This fact causes RTT substitutions at Time Out. The measuring tool substitutes first RTT to the average of first RTT at Time Out. and second RTT to the average of second RTT at Time Out. These two average of RTT is similar to each other because we inspect the measured RTT except the substituted RTT and can find that the difference between first RTT of each sets in the attacked state is larger than in the ordinary state but the difference between first and second RTT of a set in the attacked state is less than in the ordinary state. However, We assume that same length packets of same kind have same service time. And we use same target machine. Service time in the ordinary and attacked state must be same. We will improve this problem.

In the case $C3$, for we had stopped attack to the target machine, $ND + p$ had been similar to service time of the case $C1$ and ,for the target machine had been attacked, $ND + p$ had been similar to service time of the case $C2$.

About Waiting Time

In the all case, waiting time is RTT minus $ND + p$ and service time. It is hard to find the difference from RTT to waiting time by the graphs because $ND + p$ and service time is too little. However, from 100 to 300 minutes in the case $C3$, we can find that the measuring tool subtracts $ND + p$ and service time from RTT.

About Waiting Time

In the case $C1$, almost all the thresholds are between 0.9 to 1.5 millisecond.

In the case $C2$, almost all the thresholds are between 170.0 to 250 millisecond. The maximum threshold is 351.147 millisecond. In this case, the threshold is grown up from initial parts caused by the substituted average of RTT. And many RTT expect the substituted RTT is large and the RTT change the threshold largely contrasting with the ordinary state.

In the case $C3$, for we had attacked the target machine, threshold had been grown up. For we had stopped attack to the target machine, the threshold have been fallen down. However, growing up of threshold is more steep than falling down.

About Results

In the case $C1$, the measuring tool warned the unusual state of the target machine several times in some cycles. Following figure 5.2 shows threshold and waiting time at y axis between 0 to 5 millisecond in the case $C1$.

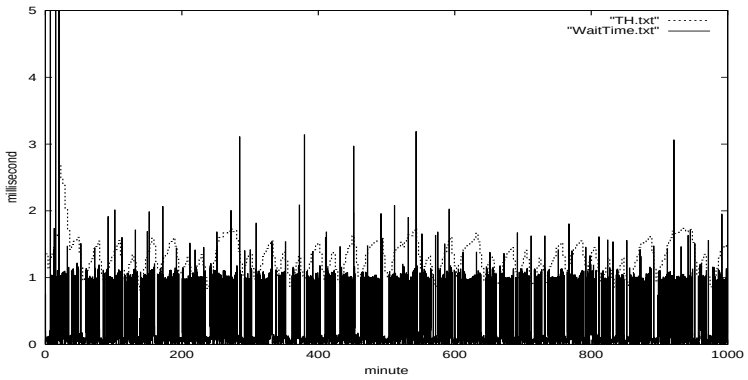


Figure 5.2: Threshold and Waiting Time in the Ordinary State

We can understand that waiting time is often larger than threshold. In the case $C1$, the number of the accesses to the target machine is nearly zero and the measuring machine is near of the target machine on the test bed network. Then, RTT to the target machine are too little and threshold become too little. As the network nodes, HUB, router, the target machine or the measuring machine, process some works at same time when the packet reaches, the length of the little RTT become two or three times and the measuring tool regards the RTT as unusual RTT. Thus, the measuring tool is not effective to observe the state of the target machine which accepts too little packets.

We expected that, when the system is in same state for the experiment, if the target machine is attacked, the measuring tool will not warn that the target machine is in the unusual state. The results of the case $C2$ shows that the expectation is true. In this experiment, the threshold rate is 96 percentage. Then, the measuring tool may mistake the detection at the 4 percentage of the number of the detections. The tool sometime measured large RTT in the case $C2$. And almost of the cycles where the measuring tool asserted some state the unusual state have only one or zero caution in twenty measurements.

In the case $C3$, while we had stopped attack to the target machine, the measuring tool did not have warn the target state. And, the measuring tool detected the unusual state when we attacked the target machine and when RTT become large at initial part of the observation unintentionally.

Thus, we can expect that the measuring tool can observe the state of the target machine which always accepts some packets.

5.1.3 About the Assumptions

We use some assumptions to make the measuring tool. We must make sure that the assumptions are correct.

About Distributions

We assume that the distribution of waiting time is exponential distribution. We plotted data of waiting time of each three case on each graphs and make each ideal exponential

distributions of each three cases using each averages of each λ. Following graphs show the two distributions on the one graph.

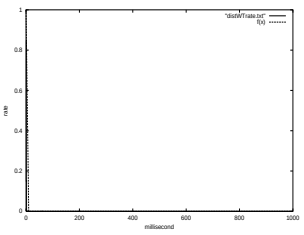
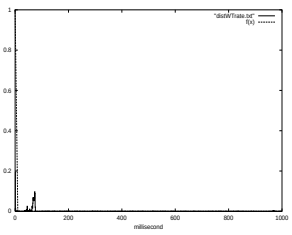
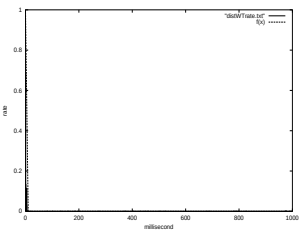
State	Ordinary	Attacked	Ordinary and Attacked
λ	2.58	0.0156	0.35
Distribution of all waiting time			

Figure 5.3: Table of Distributions

In the all cases, many waiting time are distributed at nearly zero. However, in the case $C2$, many waiting time are distributed at between 10 to 30 millisecond and, in the case $C3$, many waiting time are distributed between 30 to 100 millisecond. We find that the distributions of real waiting time are partial to small time when we compare the distributions of real waiting time with the distributions of ideal waiting time. However, we find from the case $C3$ that the distribution of real waiting time is larger than the distribution of ideal waiting time at some times.

Thus, we can find that we can assume to observe the state of the target machine that the distribution of waiting time is exponential distribution but there might be more correct model of waiting time.

5.1.4 About Threshold Rate

For example, we make the measuring tool as the percentage is 95 that each waiting times are less than threshold. In other words, when we measure a machine which is in the ordinary state, the 5 percentage of the results may be decided to the unusual state by the error. Thus, we need to make sure that the measuring tool do not warn the unusual state of target machine too many times which is in the ordinary state. And we need to measure the ordinary machine ,to collect data in fact and to count the number of the error. We can get a rate of the error about the measuring tool as we divide the number of the error by the number of all RTT sets. We call the rate real error rate. If the rate of the ordinary state is less than 1 minus the percentage of threshold, the measuring tool behaves correctly. We call the rate ideal error rate. Following table show real error rate of case $C1$. And, the author calculated real error rate of case $C2$ and $C3$.

State	the number of all measurement	the number of all cation	Error rate
Ordinary	7787	222	$222/7787 = 0.0285$
Attacked	4708	114	$2114/4708 = 0.0242$
Ordinary and Attacked	4715	134	$134/4715 = 0.0284$

Table 5.1: Table of Distributions

Ideal error rate of this experiments is 0.04 which is 1 minus 0.96. Real error rate of the ordinary and attacked state is less that ideal error rate. Thus, the measuring tool does not warn unusual state too many times.

However, the tool has a problem that real error rate of the case $C3$ is under more 0.01 on ideal error rate.This fact shows that the sensitivity of the tool is faint and often can not detect the attacked state from the ordinary state. Especially, it is a serious problem that we can not distinguish case $C3$ from case $C2$ in this experiment.

5.1.5 Results on the Internet

We experiment to observe the state of a target site on the Internet by same tool of the test bed network. For this experiment, threshold rate is 95 and the effect rate of past waiting time is 0.8.

About Results

Figure 5.4 show the results of the experiment to observe the state of the site on the Internet, www.2ch.net. The unit of the x axis is minute and the y axis is millisecond except results. The y axis shows the number of the caution at result graph. The author experimented to observe the state of the target for a week from zero o'clock at Sun, 12, July, 2003. The tool measured RTT 81839 times and there are 3114 Time Out.

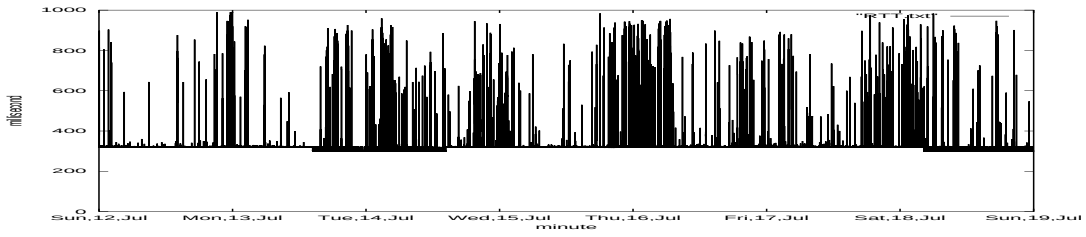
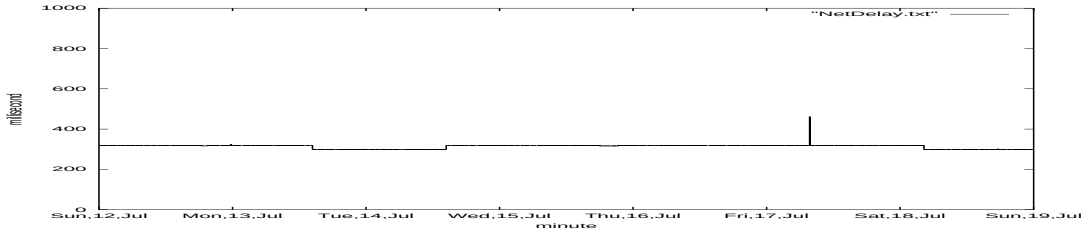
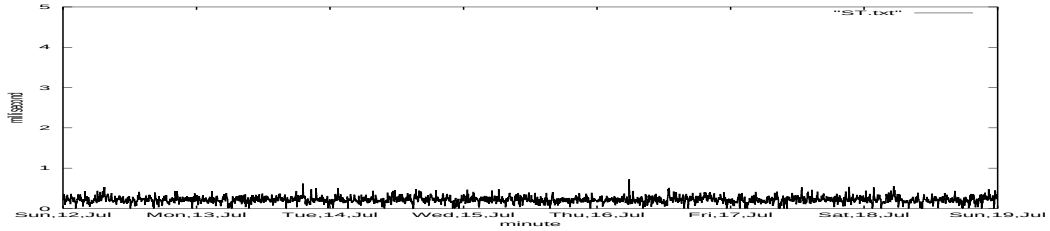
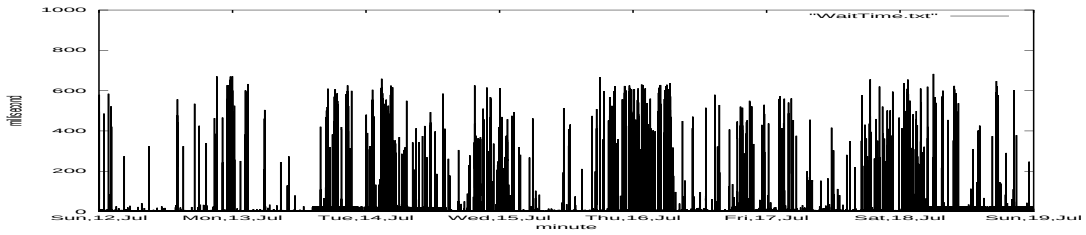
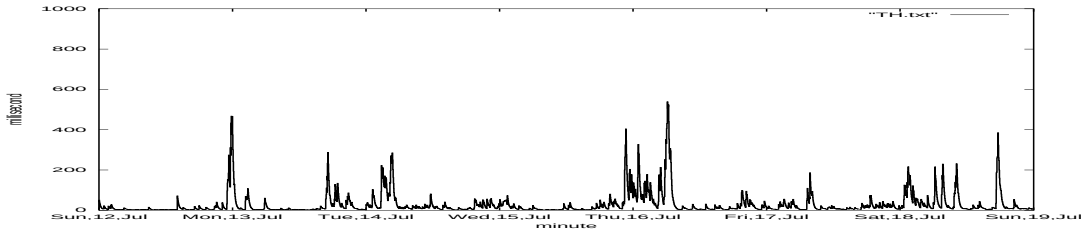
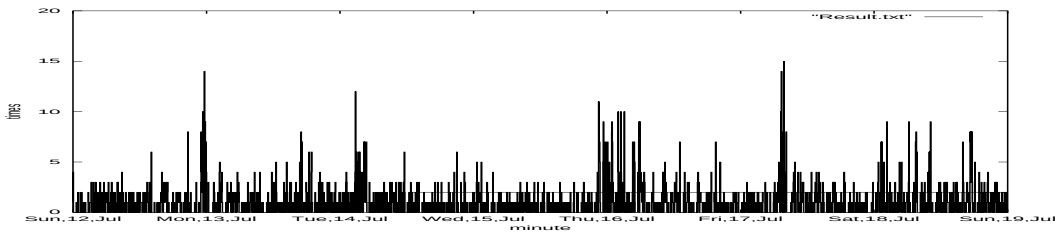
RTT	
$ND + p$	
$service\ time$	
$waiting\ time$	
$Threshold$	
$Result$	

Figure 5.4: Table of Results to the Internet

We can find that the target server had been busy from every midnight to early morning. $ND + p$ is always nearly between 300 to 320 millisecond. Compared with RTT and waiting time, the alteration of service time is little enough.

We understand from the results graph that the server of the target site was especially busy at nearly 0 o'clock of Mon, Tue and Thu. The author know that the site accepts many access at night. The result shows that the tool can observe that state correctly on the Internet. And, at early morning of Fri, there are not many large RTT but the measuring tool warned the unusual state.

About the Assumption

Next, we must make sure the distribution of waiting time and error rate. Following figure 5.5 shows the real and ideal distribution of waiting time. The below graph is the expanded above graph at between 0 to 0.05 of the y axis. The unit of the x axis shows millisecond and y axis shows the rate of waiting time.

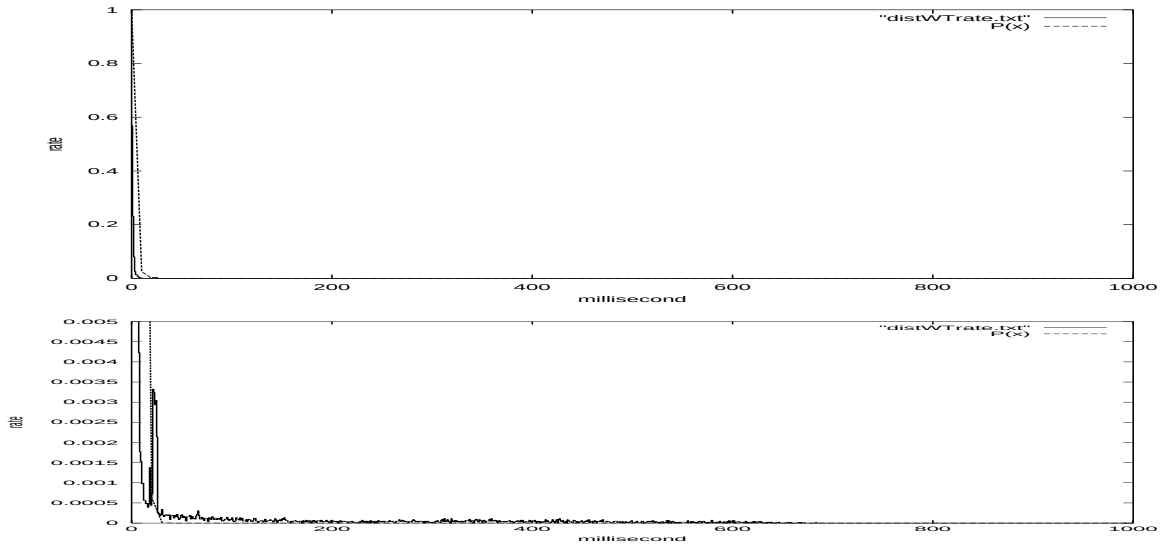


Figure 5.5: Distribution of waiting time to www.2ch.net

Comparing the ideal distribution with the real distribution, the ideal distribution is partial to right. That causes that there are some unusual waiting time at near 20 millisecond.

There are 3971 cautions of 81839 times measurement. Then, the error rate is 0.0485 dividing the number of cautions by the number of the measurements. We use the threshold rate, 95 percentage. Then, the ideal error rate is 5. We find that the real error rate is near to the ideal error rate. Thus, we can understand that the measuring tool is more effective to the target machine which often accepts the packets than to the target which rarely accepts the packets on a test bed network.

Chapter 6

Conclusion And Future Work

We can find the following facts as we experiment the measuring tool in the two environments, the test bed network and the Internet.

First, the measuring tool changes the number of the cautions at each cycles when we repeated to start and stop attack on the test bed network and the number of the cautions of the unusual state at the target site on the Internet with time passage. And, for we had attacked to the target machine on the test bed network, the number of the cautions are approximately fixed to one. On the Internet, the real error rate is near to ideal error rate. Then, we can understand that the tool observe the target machine correctly which always accepts some accesses. However, the measuring tool warned the unusual state many times on the test bed network when the target machine is in the ordinary environment. The fact shows that the tool is hard to observe the target machine which always accepts few accesses.

Second, the real error rate is lower then the ideal error rate on the test bed network and the two error rates are near from the site on the Internet. The difference between the two experiments causes from too large threshold or too little waiting time at the ordinary state. The test bed network is used for only this experiment and the target machine accepts too many access when we attacked or few access when we stop attack. The author recognizes little waiting time at the ordinary state as the reason. The last objective of the tool is to observe the state on the public computer network correctly. When we want to experiment on the test bed network more correctly, we must make the target machine to simulate access of the server on the Internet.

Third, we assume that the distribution of waiting time is exponential distribution. However, the real distribution of waiting time is partial to about zero when we compare the real distribution of waiting time with the ideal distribution of waiting time. We need the method to estimate more correct λ to calculate the ideal distribution of waiting time because we make the threshold from the ideal distribution of waiting time or other distribution near to the real distribution of waiting time.

If we use the tool to the target system on the Internet, there is a serious problem about the idea of the measurement. Many sites on the Internet are constructed by many machines. We assumes that the target machine is only one at the idea of measurement. When the target site is constructed by many machines, two consecutive packets do not always reach same machine in the site. We need new method to send the consecutive

packets to the same machine for more correct measurement.

And, we can find from the experiments on the test bed network that, when Time Out is happened too many times, the observation is very hard because we can not know the reason of Time Out, whether route to target is not alive, the target server is down, the packet wait too long or etc. DoS attack which motivates the author to research this observation is improved and became distributed DoS attack, DDoS. The attacker using DDoS attacks the target system from some cracked computers. The peculiarity of DDoS is to increase access gradually [8] We can check the state of the target machine still in progress. However, when packet loss is happened, we can not believe the results.

Packet loss and the target system constructed by many machines are most serious problems for the measuring tool.

References

[1] Information-technology Promotion Agency, Japan. The information of DDoS Attacks. ISEC. *http://www.ipa.go.jp/security/ciadr/ddos_alert.html*, May, 8, 2001.

[2] Takako Kansai. DDoS Attack to Route DNS, nine servers in thirteen servers had some troubles.. *Nikkei Comunication*, pp,66. Nov, 18, 2002.

[3] J. Postel, ISI. INTERNET CONTROL MESSAGE PROTOCOL, DARPA INTERNET PROGRAM PROTOCOL SPECIFICATION. IETF. http://www.ietf.org/rfc/rfc0792.txt?number=792, September, 1981.

[4] Paul Ferguson, Geoff Huston . Quality of Service. John Wiley & Sons,Inc.. http://www.ietf.org/rfc/rfc0792.txt?number=792, 1998 .

[5] atmarkIT Corp. Sniffer. atmarkIT Corp.. http://www.atmarkit.co.jp/aig/02security/sniffer.html,

[6] Brian Caswell, Marty Roesch. What is Snort?. Brian Caswell, Marty Roesch. http://www.snort.org/about.html, Feb, 5, 2003.

[7] Coretez Giovanni. Stick. The 8th Port. http://www.eurocompton.net/stick/projects8.html,

[8] Yan. yahoo.txt.
$$packetstorm$$
. http://packetstormsecurity.org/distributed/yahoo.txt,

Appendix A

All Data of the observation to www.2ch.net

The following graphs show the results of the experiment to www.2ch.net. The author continued to measure the target site. We saw some parts of the results in the chapter 5.

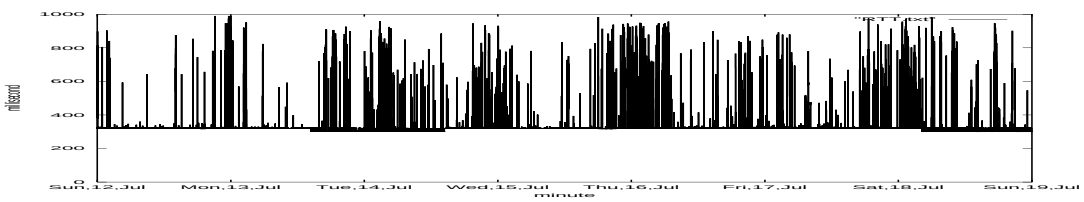
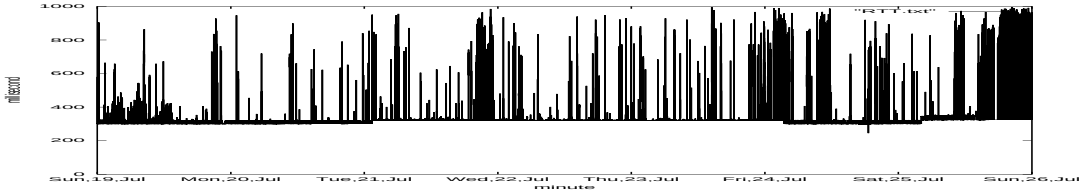
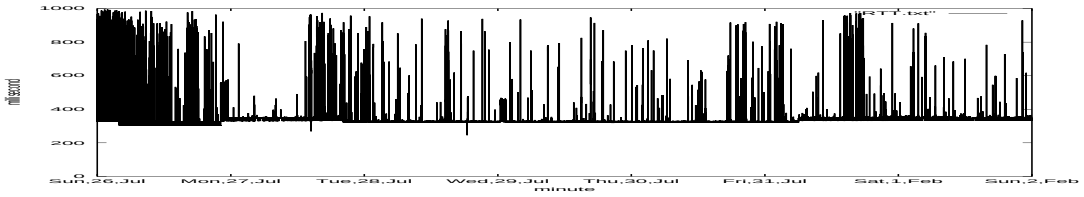
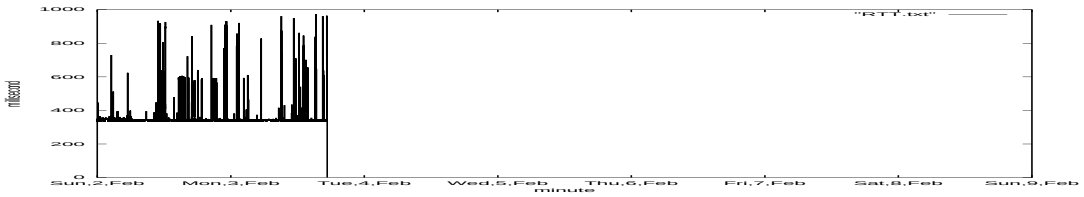
Date	RTT of www.2ch.net
12, Jul - 19, Jul	
19, Jul - 26, Jul	
26, Jul - 2, Feb	
2, Feb - 9, Feb 0	

Figure A.1: Table of www.2ch.net RTT

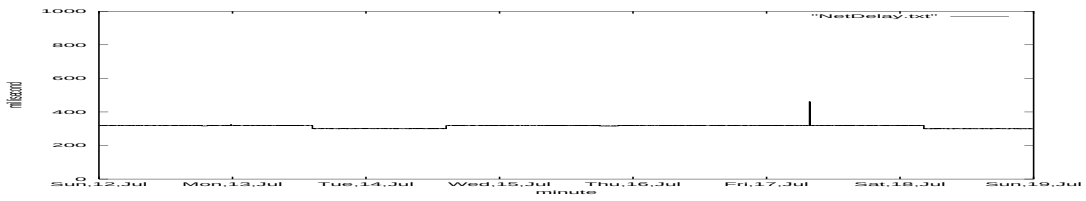
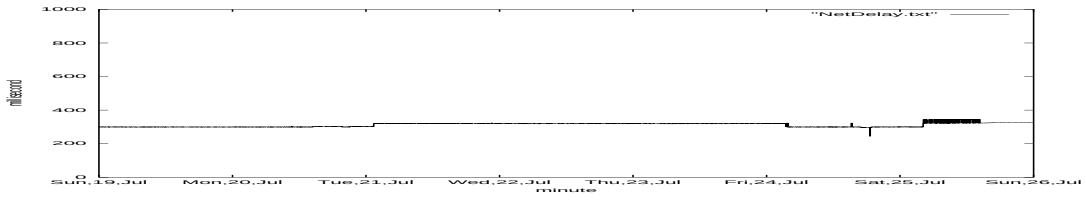
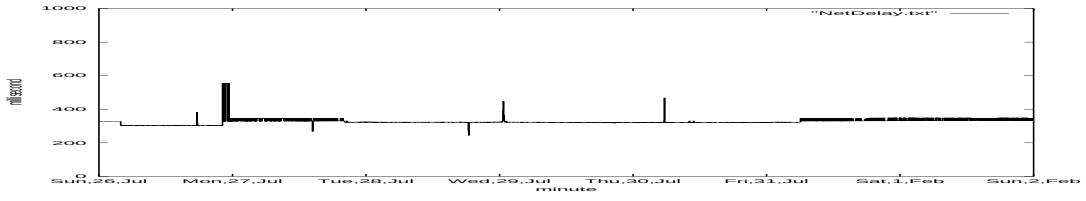
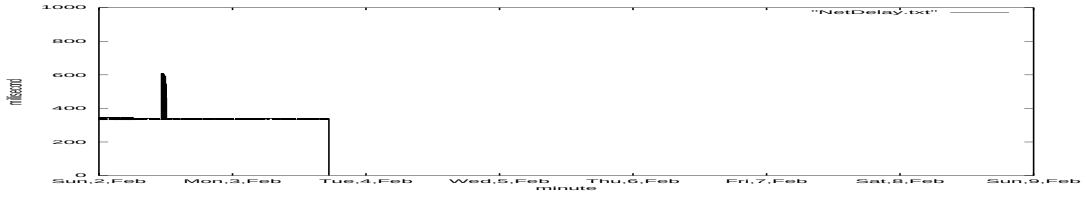
Date	$ND + p$ of www.2ch.net
12, Jul - 19, Jul	
19, Jul - 26, Jul	
26, Jul - 2, Feb	
2, Feb - 9, Feb	

Figure A.2: Table of www.2ch.net $ND + p$

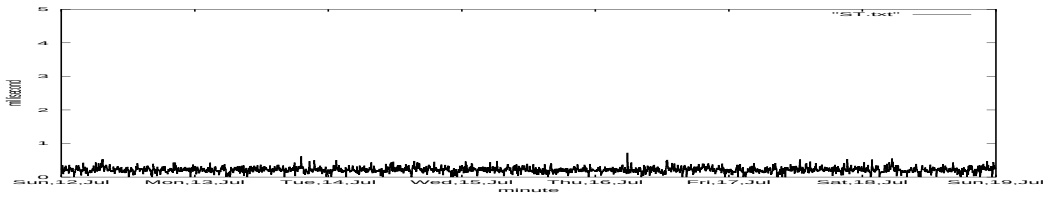
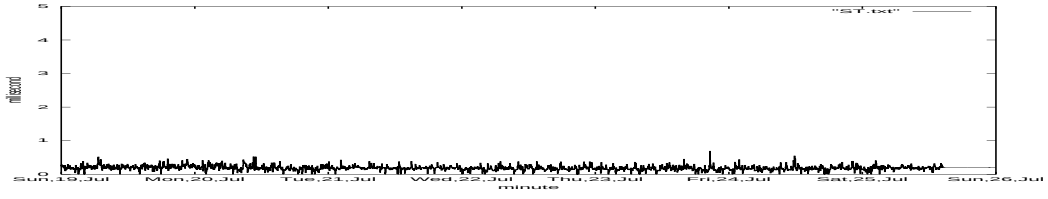
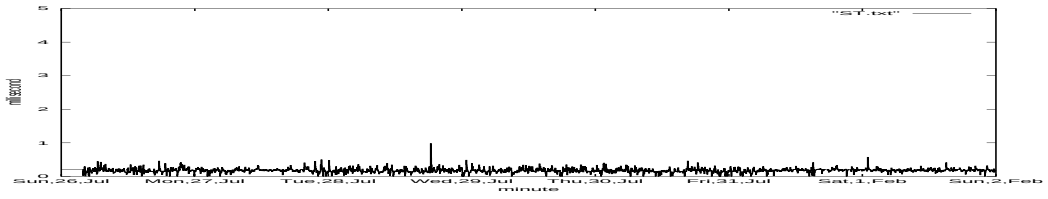
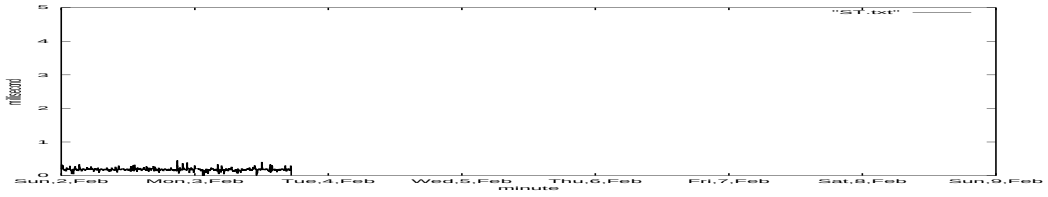
Date	Service Time of www.2ch.net
12, Jul - 19, Jul	
19, Jul - 26, Jul	
26, Jul - 2, Feb	
2, Feb - 9, Feb	

Figure A.3: Table of www.2ch.net Service Time

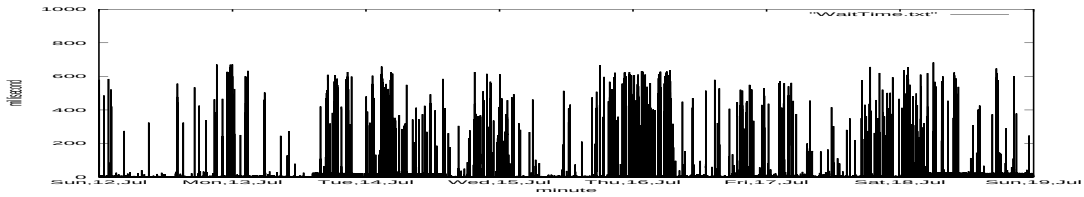
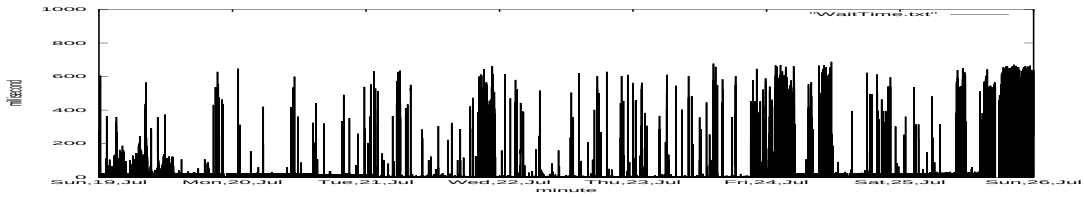
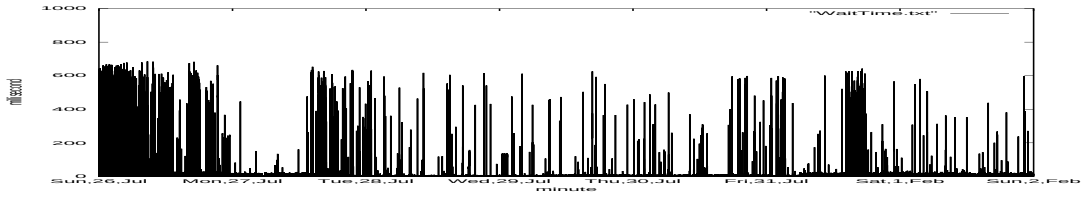
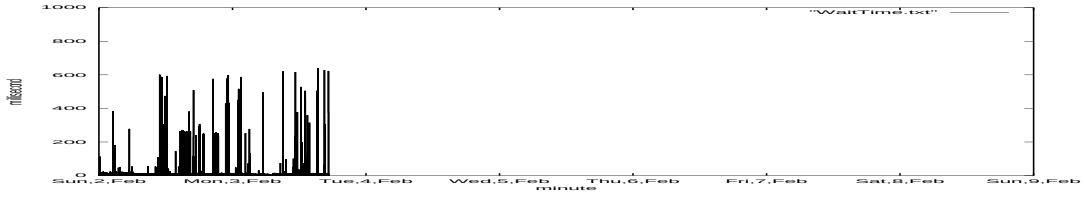
Date	Waiting Time of www.2ch.net
12, Jul - 19, Jul	
19, Jul - 26, Jul	
26, Jul - 2, Feb	
2, Feb - 9, Feb	

Figure A.4: Table of www.2ch.net Waiting Time

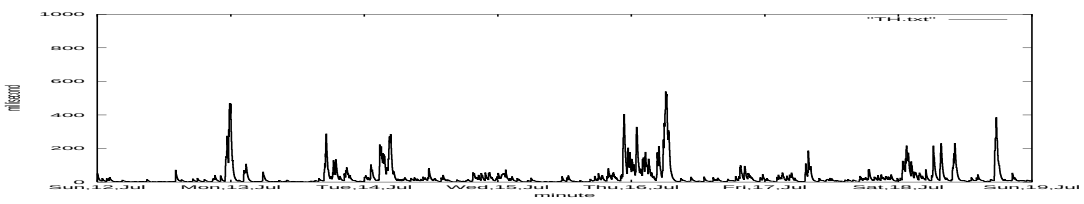
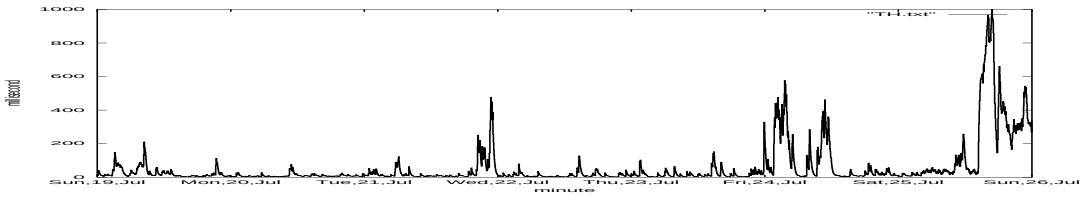
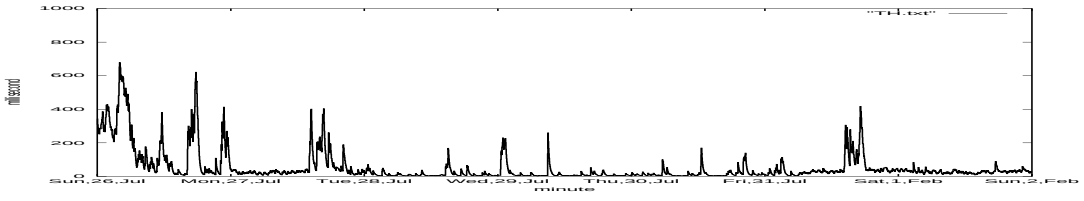
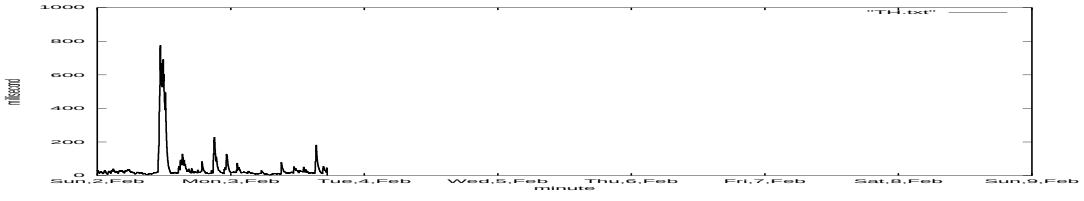
Date	Threshold of www.2ch.net
12, Jul - 19, Jul	
19, Jul - 26, Jul	
26, Jul - 2, Feb	
2, Feb - 9, Feb	

Figure A.5: Table of www.2ch.net Threshold

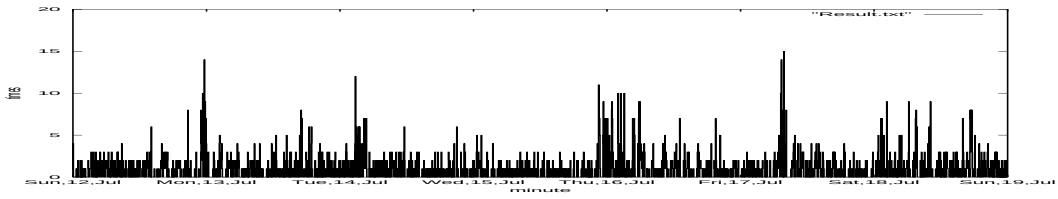
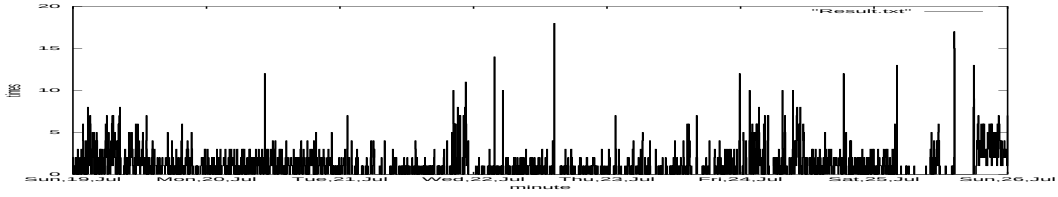
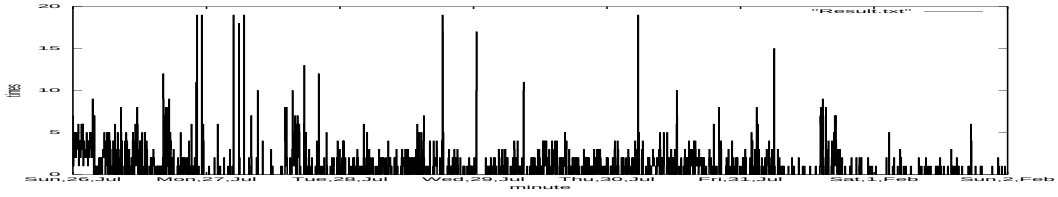
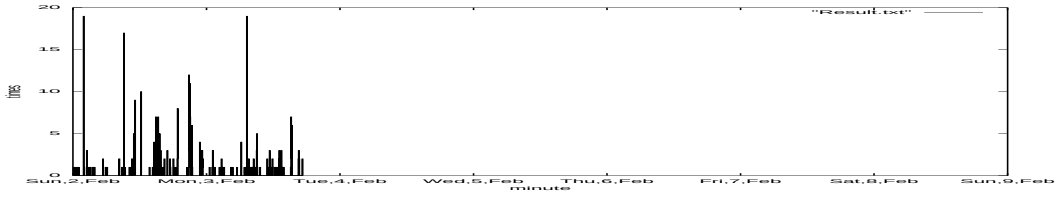
Date	Results of www.2ch.net
12, Jul - 19, Jul	
19, Jul - 26, Jul	
26, Jul - 2, Feb	
2, Feb - 9, Feb	

Figure A.6: Table of www.2ch.net Results

Appendix B

All Data of the observation on the Test Bed Network

The following graphs show the results of the experiment on the Test Bed Network. Threshold rate is 0.95, 0.97, 0.98 and 0.99.

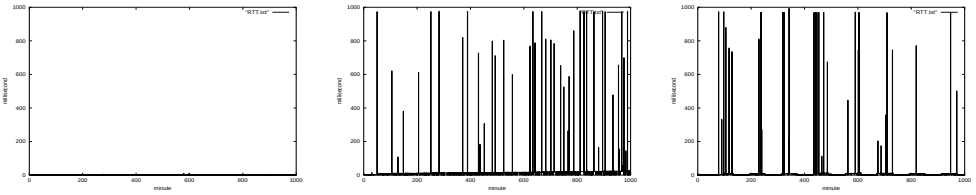
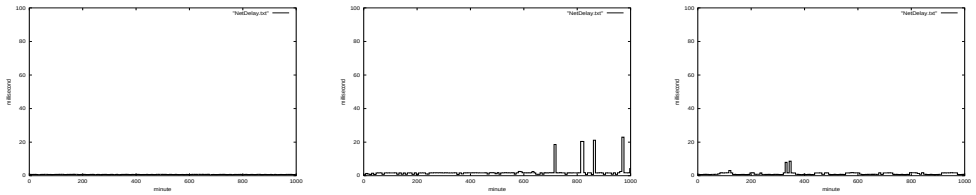
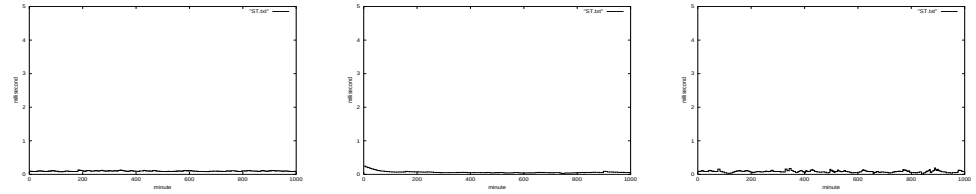
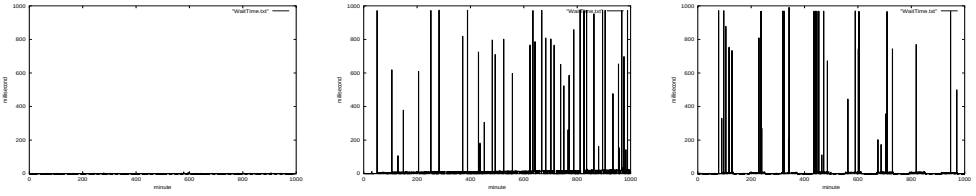
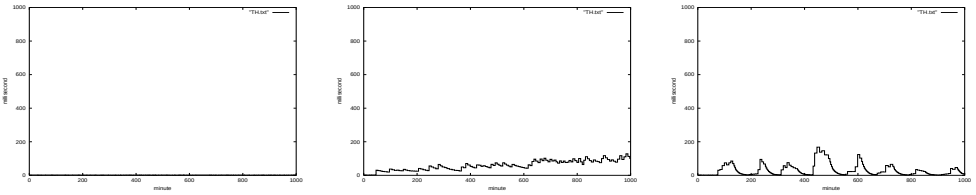
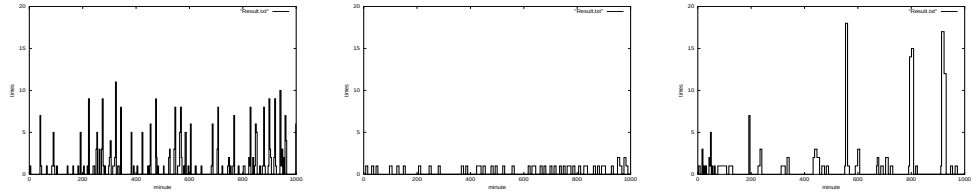
the type of graph	Ordinary	Attacked	Ordinary and Attacked
$FristRTT$			
$ND + p$			
$Service\ Time$			
$Waiting\ Time$			
$Threshold$			
$Result$			

Figure B.1: Table of Results with Threshold Rate: 0.95

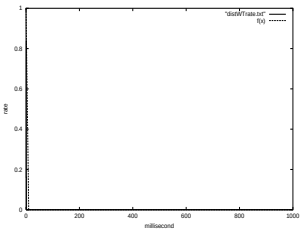
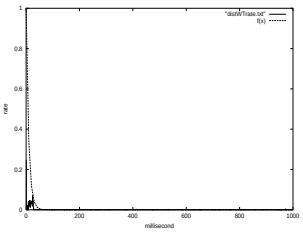
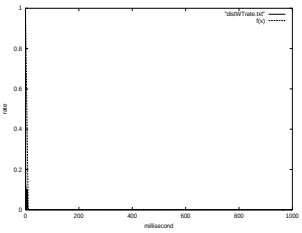
State	Ordinary	Attacked	Ordinary and Attacked
λ	2.58	0.0156	0.35
Distribution of all waiting time			
The error rate	0.0514	0.0203	0.0313

Figure B.2: Table of Distributions with Threshold Rate: 0.95

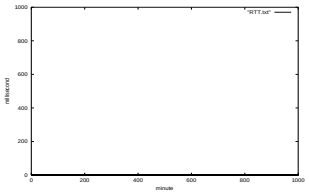
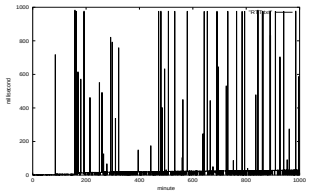
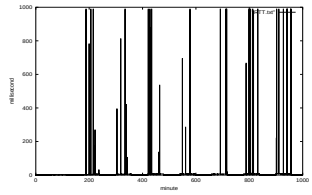
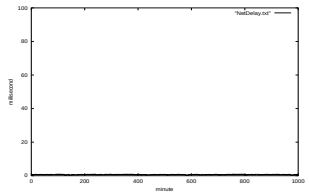
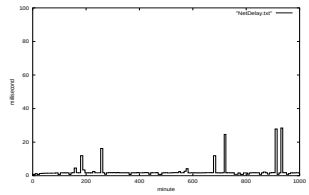
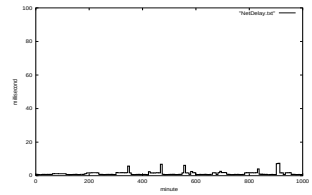
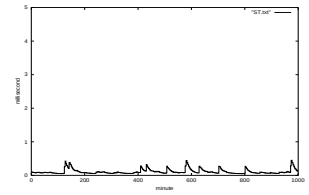
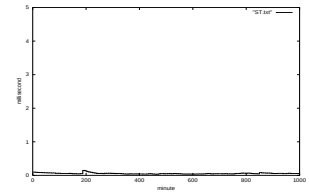
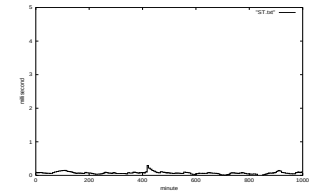
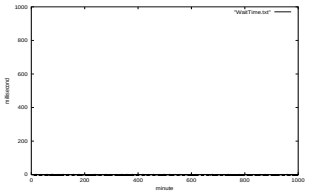
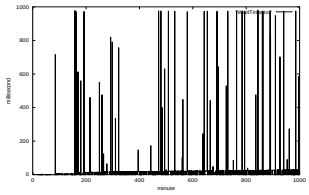
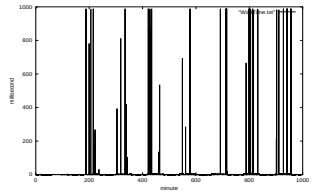
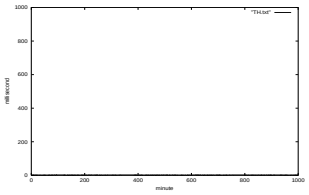
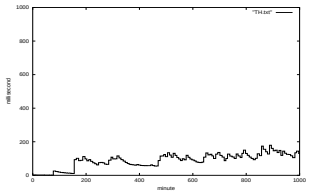
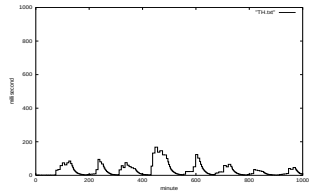
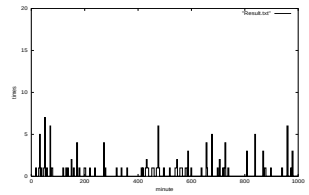
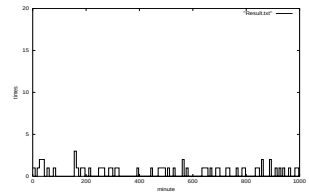
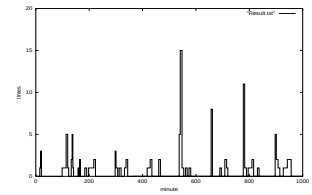
the type of graph	Ordinary	Attacked	Ordinary and Attacked
$FristRTT$			
$ND+p$			
$Service\ Time$			
$Waiting\ Time$			
$Threshold$			
$Result$			

Figure B.3: Table of Results with Threshold Rate: 0.97

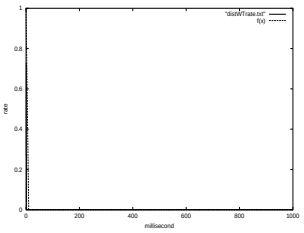
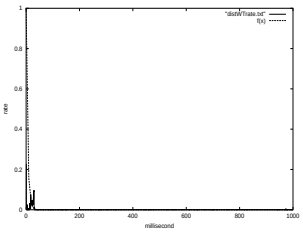
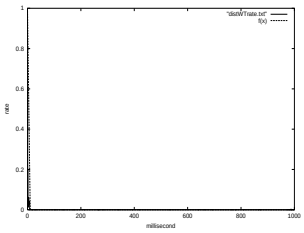
State	Ordinary	Attacked	Ordinary and Attacked
λ	0.773	0.0156	0.183
Distribution of all waiting time			
The error rate	0.0218	0.209	0.0219

Figure B.4: Table of Distributions with Threshold Rate: 0.97

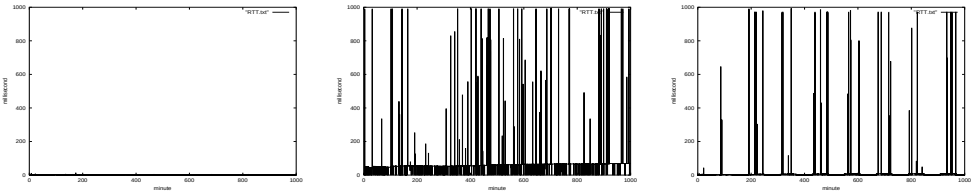
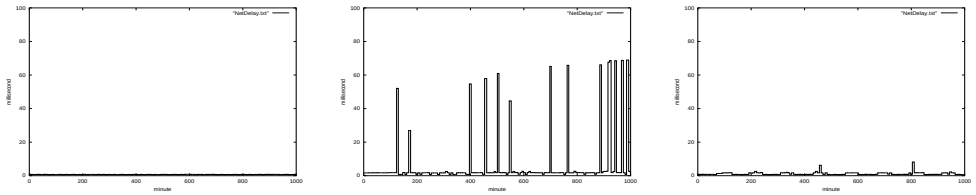
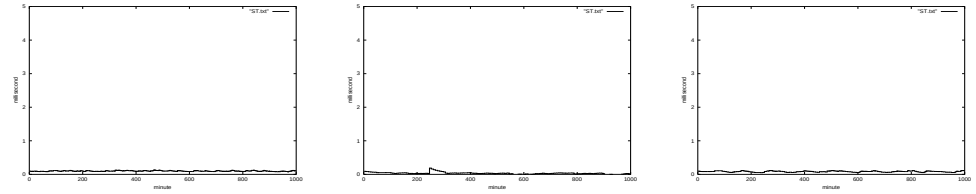
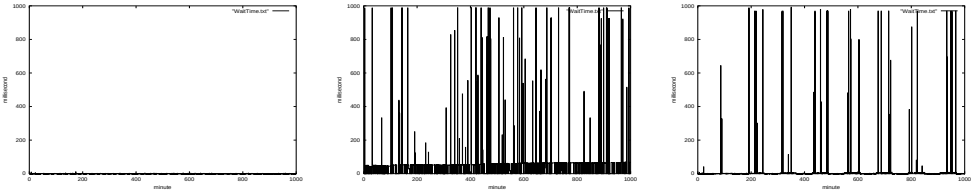
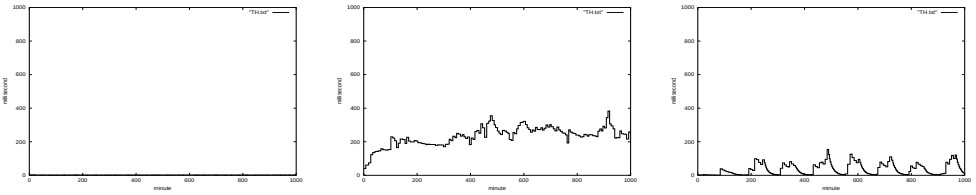
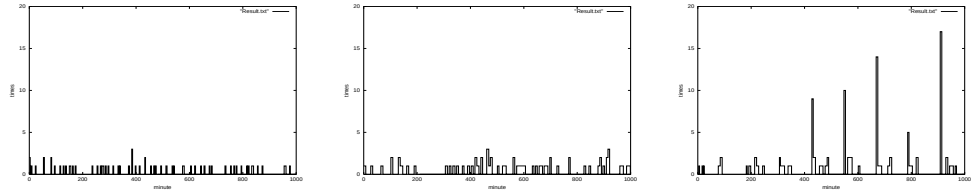
the type of graph	Ordinary	Attacked	Ordinary and Attacked
$FristRTT$			
$ND + p$			
$Service\ Time$			
$Waiting\ Time$			
$Threshold$			
$Result$			

Figure B.5: Table of Results with Threshold Rate: 0.98

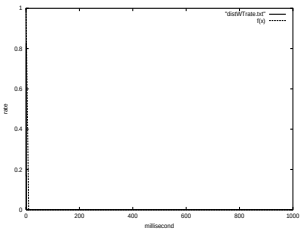
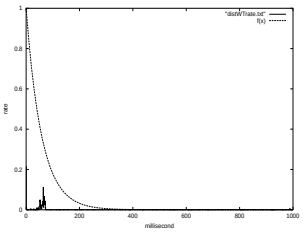
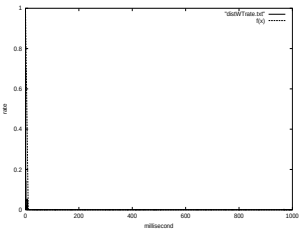
State	Ordinary	Attacked	Ordinary and Attacked
λ	0.0172	0.0156	0.881
Distribution of all waiting time			
The error rate	0.0102	0.0219	0.0173

Figure B.6: Table of Distributions with Threshold Rate: 0.97

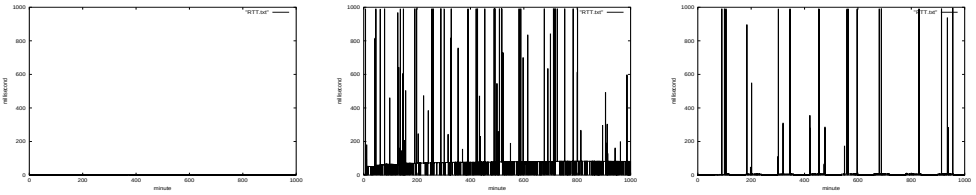
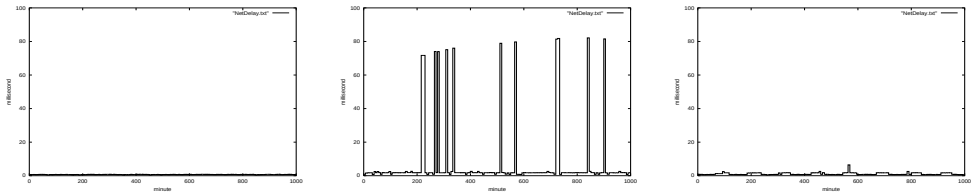
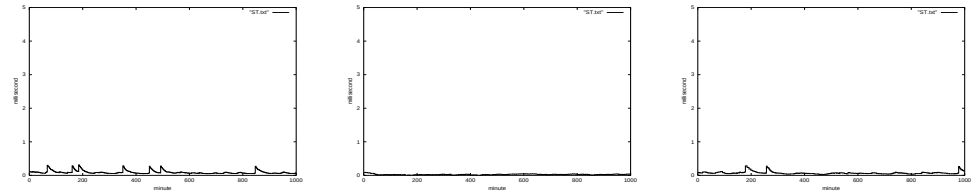
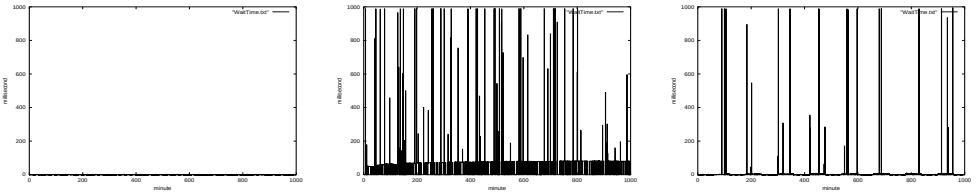
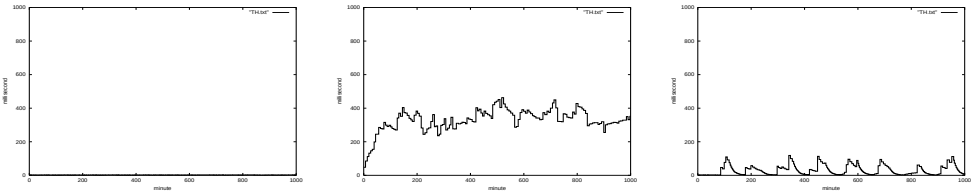
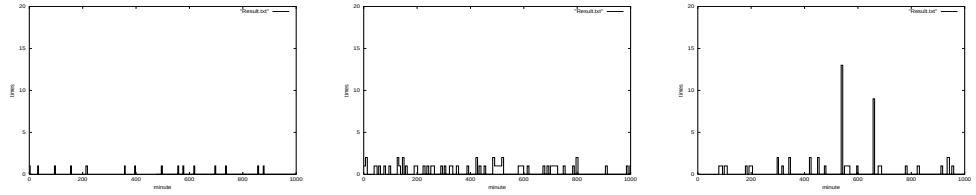
the type of graph	Ordinary	Attacked	Ordinary and Attacked
$FristRTT$			
$ND + p$			
$Service\ Time$			
$Waiting\ Time$			
$Threshold$			
$Result$			

Figure B.7: Table of Results with Threshold Rate: 0.99

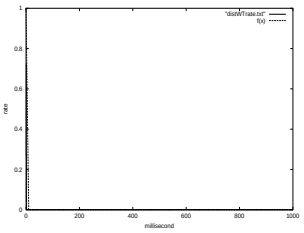
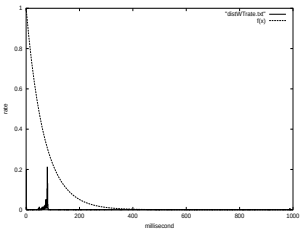
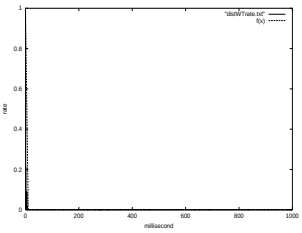
State	Ordinary	Attacked	Ordinary and Attacked
λ	2.48	0.0149	0.633
Distribution of all waiting time			
The error rate	0.00231	0.0183	0.0128

Figure B.8: Table of Distributions with Threshold Rate: 0.99