Shaping the Traffic to Improve QoS Parameter during Data Privacy in Secure Multiparty Computation

Tonu Sojatia¹, Zulfa Shaikh²

¹ Bachelor of Engg in Information Technology
Acropolis Institute of Technology and Research, Indore (M.P) India
tonusojatia@gmail.com

² Lecturer, Master of Computer Application
Acropolis Institute of Technology and Research, Indore (M.P.) India
shaikh.zulfa@gmail.com

ABSTRACT
In this paper, we propose the methodology and design an algorithm to control congestion during Secure Multiparty Computation (SMC). As per our literature, a lot of work has been done in SMC but they have worked only the main part of the SMC, privacy and correctness. Congestion control is one of the important components for the performance of the network and also is the most challenging one. In our previous research, we assumed that the communication between party and anonymizer is secure, but if any anonymizer becomes malicious then entire data of one party can be hacked. In this protocol, to maintain the security between party and anonymizer, we have partitioned the data in the number of packets before sending to the anonymizer. We propose the framework in the form of a protocol for secure multi-party computation. This protocol maintains the security when we transfer the data between party and anonymizer. This paper deals with the way of shaping the traffic to improve quality of service (QoS) in SMC. Since on increasing the number of packets there may be a problem of congestion and under the congestion situation, the queue length may become very large in a short time, resulting in buffer overflow and packet loss. So congestion control is necessary to ensure that users get the negotiated QoS.

KEYWORDS
Congestion, Privacy, Secure Multi-party Computation, Quality of service

INTRODUCTION
Lecturer, Master of Computer Application
Acropolis Institute of Technology and Research, Indore (M.P.) India
shaikh.zulfa@gmail.com

The assumption that statistical multiplexing can be used to improve the link utilization is that the users do not take their peak rate values simultaneously. But since the traffic demands are stochastic and cannot be predicted, congestion is unavoidable. Whenever the total input rate is greater than the output link capacity, congestion happens. Under a congestion situation, the queue length may become very large in a short time, resulting in buffer overflow and cell loss[1]. So congestion control is necessary to ensure that users get the negotiated QoS.

There are several misunderstandings about the cause and the solutions of congestion control

1. Congestion is caused by the shortage of buffer space. The problem will be solved when the cost of memory becomes cheap enough to allow very large memory. Larger buffer is useful only for very short term congestions and will cause undesirable long delays. Suppose the total input rate of a switch is 1Mbps and the capacity of the output link is 0.5Mbps, the buffer will overflow after 16 second with 1Mbyte memory and will also overflow after 1 hour with 225Mbyte memory if the situation persists. Thus larger buffer size can only postpone the discarding of cells but cannot prevent it. The long queue and long delay introduced by large memory is undesirable for some applications.

2. Congestion is caused by slow links. The problem will be solved when high-speed links become available. It is not always the case; sometimes increases in link bandwidth can aggravate the congestion problem because higher speed links may make the network more unbalanced [2]. For the configuration showed in the Figure 1, if both of the two sources begin to send to destination 1 at their peak rate, congestion will occur at the switch. Higher speed links can make the congestion condition in the switch worse.

Fig 1.A Network with High Speed Links
BACK GROUND AND OUR PROPOSAL
The SMC problem was first described by Yao in [3]. Maurer has defined the different type of security in database [4]. It also prescribes some applications of SMC. Goldreich et al. show the existence of secure solution of SMC problems [5]. Even though this approach is simple yet the solution given by it is very complex and the size of the protocol depends on the number of parties involved in the computation. From this list, it is clear that the solutions to SMC problems are applicable to a wide range of applications and there is a significant scope in developing the efficient frameworks for SMC. They investigate how various computational geometry problems can be solved in cooperative environment, where two parties need to solve a geometric problem based on their joint data, but neither wants to disclose its private data to the other party. Security is defined relatively to an idle-world specification involving a trusted party: anything the adversary can achieve in the real world (where the protocol is executed) he can also achieve in real model [6].

Our assumption is using anonymizer so that the party can hide the identity of own. To resolve this situation, we make following assumptions.
1. TTP computes the result of the function $y = f(x_1, x_2, \ldots, x_n)$ correctly.
2. TTP has the ability to announce the result of the computation publicly.
3. Each party having the input can communicate with a trusted anonymizer. A trusted anonymizer ($A_i$) is a system that acts as an intermediary between the parties having the input and the TTP which will carry out the computation. Thus, $A_i$ hides the identity of $P_i$ (Party) from the TTP.
4. The communication channels used by the input providing parties to communicate with the anonymizers are secure. That is no intruder that can cut off the data transferred between them.
5. The anonymizer in any condition will not disclose the identity of the data source, from which it is forwarding the data to the third party.

FORMAL DESCRIPTION OF THE PROTOCOL
Now each party $P_i$ as depicted in fig 2 has an opportunity to split its input into number of inputs and can route them through different routes with different identifiers for subpart of the input [5]. Each party provide inputs on to the second layer known as anonymizer layer which is an untrusted layer and so the complete information about inputs of other parties are not known to them. Anonymizers just pass information to the third party for computation. Third party in turn does computation on inputs and provides result to the respective parties.

The concept of breaking the data into number of packets is to increase the security of information. When number of packets increases then the probability of hacking of data will tends to zero so the privacy of party is being achieved which was our previous work.

Our protocol is based on three layers namely computation layer, security layer and input layer. In Figure 1, there are $n$ parties $P_1, P_2, \ldots, P_n$, each having input $x_i$ that will be used in computing $y = f(x_1, x_2, \ldots, x_n)$. To calculate the value of, the each input providing party $P_i$ takes the following steps if the form of an algorithm.

Algorithm
1. Define $P_1, P_2, \ldots, P_n$ as parties;
2. Define $A_1, A_2, \ldots, A_n$ as anonymizers;
3. Generate a random key, $R_i$, as an identifier for each $P_i$;
4. Compute $d_i = x_i || R_i$;
5. $P_i$ sends $d_i$ to $A_i$;
6. $A_i$ sends $d_i$ to third party through some network.
7. If each single packet is divided into number of packets as $d_{i1}, d_{i2}, \ldots, d_{in}$ and if input generation rate is higher than the output rate then there may be congestion in the network which may cause packet loss and delay in the network.
8. One way of preventing packet loss is to increase the FIFO length of the router or by increasing its clock rate.
9. Simulations were carried out for message with packet length of 3500.
10. The clock rate of router is varied from 5 Mbps to 100 Mbps.
11. On increasing the clock rate of router, service time of packets at the router and FIFO length required for buffering the packets get reduced.
12. Router then forward the packets to UTP; /*The UTP takes each data unit $d_i$ and extract $x_i$ and $R_i$ from it.*/

Fig 3 shows a network connection where each party on different LAN’s send packets to the router. Router has
forwarding capability which allows it to forward the packet to final destination. On splitting the packet of each party into different number of small packets may lead to a heavy traffic in the network and if the processing rate of the router is less than that of input generation rate then this situation will lead to congestion and finally may result in packet loss. Our study is basically based on how to prevent packet loss in the network.

![Fig 3. Proposed Network Connection in SMC architecture](image)

**IMPLEMENTATION OF PROPOSED WORK**

We considered the fact that the difference in the packet generation rate at the source and processing rate at the router is the cause of packet loss. Zero packet loss can be ensured if this difference is zero.

The router’s processing rate at any time depends on the packet traffic [7]. The traffic at the router varies with time and it is not possible to keep the processing rate of the router same as that of the source. When the traffic is high the packet service time at the router increases which tends to packet loss. In such cases packet loss can be prevented by increasing the FIFO length of the router or by increasing its clock rate. We carried out experiments for computing the FIFO length that ensures zero packet loss for different clock rate of the router. Simulations were carried out for message with packet length of 3500. The clock rate of router is varied from 5 Mbps to 100 Mbps. Higher clock rate reduces the service time of packets at the router and the FIFO length required for buffering the packets also reduces.

![Fig 4. Routers clock rate versus FIFO Length](image)

**RESULTS AND CONCLUSION**

Our study concludes that on increasing the clock rate of router, service time of packets at the router and FIFO length required for buffering the packets get reduced which ensures zero packet loss in the network.

**FUTURE SCOPE**

This paper deals with one way of shaping the traffic by increasing the clock rate of the router. A lot of work has been done as per the security is concerned but network side of SMC is still untouched. We can further design algorithms for proper congestion problem management in SMC.

**REFERENCES**


