Volume 4, Issue 4, April 2014 ISSN: 2277 128X

International Journal of Advanced Research in Computer Science and Software Engineering

Research Paper

Available online at: www.ijarcsse.com

Proficient Resource Mapping Framework in Clouds with Security and Search-Based Request

K.Kiruthika
PG Scholar
Department of CSE
Chettinad College of Engineering & Technology, India

Mr.M.Saravanakumar
Assistant Professor
Department of CSE
Chettinad College of Engineering & Technology, India

Dr.T.Rajendran
Professor & Head
Department of CSE
Chettinad College of Engineering & Technology, India

Abstract- Cloud computing is a new and promising paradigm delivering IT services as computing utilities. As Clouds are intended to provide services to external users, providers need to be compensated for sharing their resources and capabilities. This aggregate at bottom aim on a fastidious group of traffic analysis attacks, flow correlation attacks, by which an adversary attempts to analyze the network traffic and correlate the traffic of a flow over an input link with digress over an output link. Team a hardly classes of correlation methods are considered, namely time-domain methods and frequency-domain methods. Based on our threat model and known strategies in existing mix networks, we perform extensive experiments to analyze the performance of mixes. We get that all but a few batching strategies fail against flow-correlation attacks, allowing the adversary to either identify ingress or egress points of a flow or to reconstruct the path used by the flow. Tab intuitively, some batching strategies are actually detrimental against attacks. The practical moderate provided in this paper give an indication to designers of Mix networks about appropriate configurations and mechanisms to be used to counter flow-correlation attacks.

Keywords — cloud computing, mix network, traffic analysis attack, flow correlation attack, security.

I. INTRODUCTION

As the Internet is increasingly worn in there aspects of daily life, the realization has emerged saunter privacy and confidentiality are notable requirements for the success of unlike applications. It has been shown that, in many situations, encryption alone cannot provide the level of confidentiality required by users, since establishment opinion duff easily uncover information about the participants in a distributed application. Operator void is yoke important confidentiality criterion for many applications, ranging from peer-to-peer file sharing and anonymous web browsing or e-mail, to various forms of electronic commerce, and finally to electronic voting. The rune of many such applications requires that the identity of either one or more of the participants remains confidential either from the other participant(s) or from third parties. The unconsciousness of practices can be avidly stirred by an observer in two ways, either through inspection of payload or headers of the exchanged data packets, or, when encryption is used, through traffic analysis. Sufficiency bustling encryption can be used to prevent packet content inspection, giving prevalence to the second form of attack. Traffic analysis is usually countered by the use of intermediary nodes, whose role is to perturb the traffic flow and thus confuse an external observer. Such intermediaries (often called mixes) delay and reroute exchanged messages, reorder them, pad their size, or perform other operations. Chaum proposed such a mix network to handle mail traffic. This paper gives an overview of the analysis of mix networks. The details are described in the companion technical report [1].

The original Chaum mix network operates on entire mail messages at a time and therefore does not need to pay particular attention to latency added by the mixes. Increasingly, the data exchanged exceed by far the capacity of mixes, for example, in file-sharing applications. As a result, current mixes operate on individual packets of a flow rather than on entire messages. In conjunction with source routing at the sender, this allows for very efficient network-level implementations of mix networks. For the designer of the anonymity system, these results in a tradeoff between the anonymity degree and quality of- service (QoS). Although significant efforts have been put forth in researching anonymous communication. Only recently systematic studies appeared to quantitatively capture the effect of traffic perturbation on the anonymity in realistic settings. It is, therefore, difficult to assess the improvement of anonymity that one attains for any given cost in form of added latency and perturbation to traffic streams. Moreover, few quantitative guidelines exist on how different perturbation mechanisms perform. This project focuses on the quantitative evaluation of mix performance. We focus our analysis on a particular type of attack, which we call the flow-correlation attack [8]. In general, flow-correlation attacks attempt to reduce the anonymity degree by estimating the path of flows through the mix network.

II. RELATED WORK

Houidi et al. [3] addresses the provisioning of virtual assets in future systems depending on the Infrastructure as an administration rule. Accurate and heuristics streamlining calculations for the provisioning of virtual systems including

numerous framework suppliers are introduced. Asset matching, part, installing and tying steps needed for virtual system provisioning are proposed and assessed. Papagianni et al. [4] propose a strategy for the effective mapping of asset appeals onto an imparted substrate interconnecting different islands of processing assets, and embrace a heuristic procedure to address the problem.zhu et al. [5] recent recommendations for system virtualization give a swearing up and down to way to beat the Internet hardening. The key thought of system virtualization is to assemble a broadened Internet to help a mixed bag of system administrations and architectures through an imparted substrate. A real test in system virtualization is the allocating of substrate assets to virtual systems (VN) productively and on-interest.

III. EXISTING SYSTEM

The cloud Infrastructure as a Service (IaaS) is the capability provided to the consumer to provision processing, storage, networks, and other fundamental computing resources where the consumer is able to deploy and run arbitrary software, which can include operating systems and applications. In order to provide cloud IaaS with minimal management effort it is essential to address efficiently the resource mapping problem. The resource mapping result incorporates mapping the virtual hubs to the physical host hubs and steering the virtual connections over the physical connections. The existing works where either fixed or random costs are assumed [9], [7], [2] and also to avoid the overload in the cloud environment and also explain the resource mapping details.

IV. PROPOSED SYSTEM

In this proposed system focuses on the quantitative evaluation of mix performance. We focus our analysis on a particular type of attack, which we call the flow-correlation attack. In general, flow-correlation attacks attempt to reduce the anonymity degree by estimating the path of flows through the mix network. Flow correlation analyzes the traffic on a set of links (observation points) inside the network and estimates the likelihood for each link to be on the path of the flow under consideration. An adversary analyzes the network traffic with the intention of identifying which of several output ports a flow at an input port of a mix is taking. Obviously, flow correlation helps the adversary identify the path of a flow and consequently reveal other critical information related to the flow e.g sender and receiver (fig.1). Mainly security is provided for traffic analysis attack when data is moving from the cloud server and forwards the file to the mix.

This architecture diagram describe the resource mapping details like server details, Mix details, Receiver details to how to avoid the overload in the cloud environment. And also explain the security strategy of data transmission from cloud server, Mix and to the Receiver.

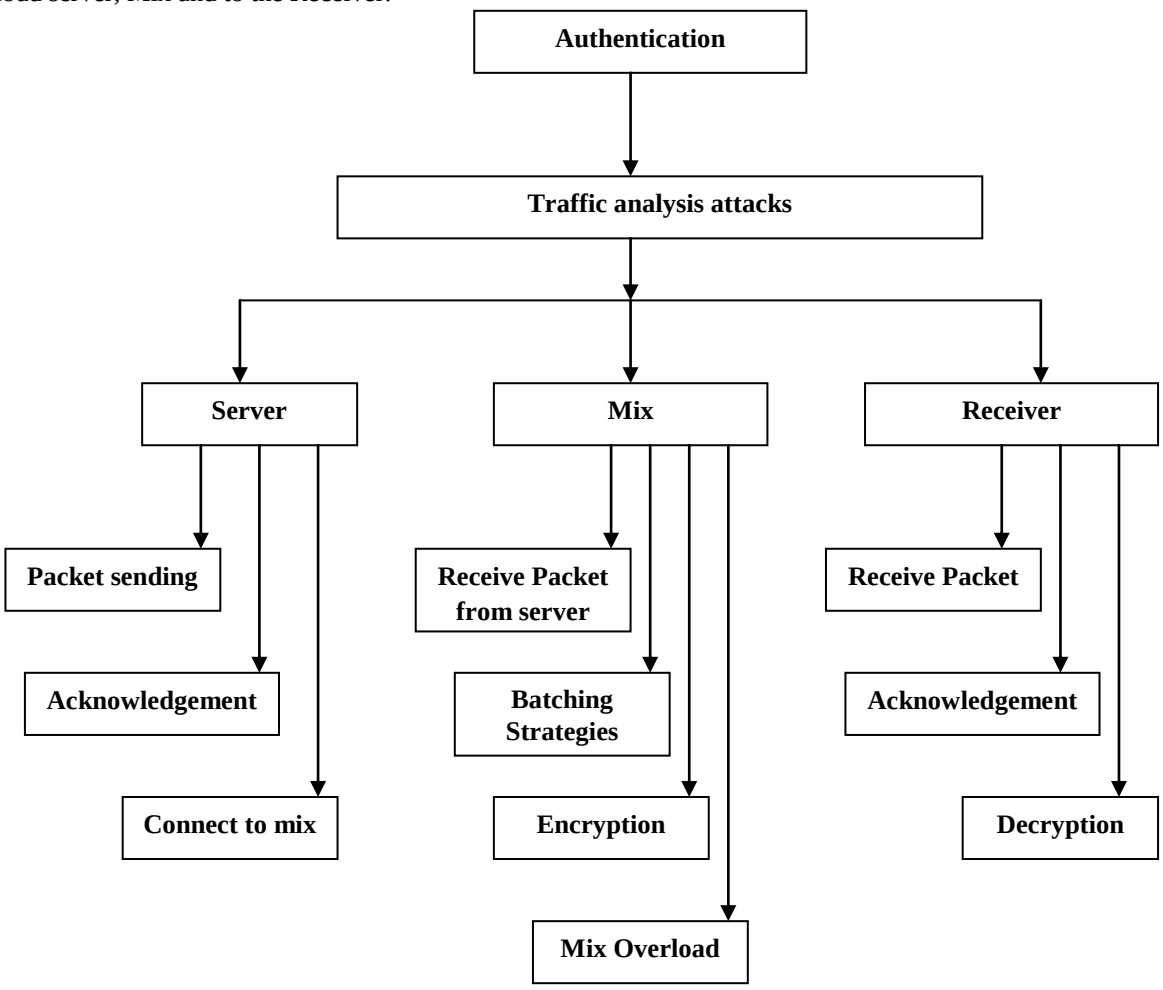

Fig.1 Architecture Diagram

V. BATCHING STRATEGIES ALGORITHM

Batching strategies are designed to prevent not only simple timing analysis attacks, but also powerful trickle attacks, flood attacks, and many other forms of attacks [6], [10]. Summarizes seven batching strategies that have been proposed. We will evaluate each of these strategies. Our results show that these strategies may not work under certain timing analysis attacks. These even batching strategies are listed in Table 1, in which batching strategies from s1 to s4 are denoted as simple mixes, while batching strategies from s5 to s7 are denoted as pool mixes. Flow Pattern Vector Extraction Once the data are collected, the relevant pattern vectors must be extracted. Recall that batching strategies in Table 1 can be classified into two classes: threshold-triggered batching (s1, s3, and s5)1 and timer-triggered batching (s2, s4, s6, and s7). The packet timing characteristics at the output link allows for targeted feature extraction for these different classes of batching. For threshold-triggered batching strategies, packets leave the mix in batches. Hence, the interarrival time of packets in a batch is determined by the link bandwidth, which is independent of the input flow. Thus, the useful information to the adversary is the number of packets in a batch and the time that elapses between two batches. Normalizing this relationship, we define the elements in pattern vector Yj as follows:

Glossary

n	queue size
m	threshold to control the packet sending
t	timer's period if a timer is used
f	the minimum number of packets left in the pool for pool Mixes
p	a fraction only used in Timed Dynamic-Pool Mix

Algorithm

Strategy Index	Name	Adjustable parameters	Algorithm
S0	Simple Proxy	none	no batching or reordering
S1	Threshold Mix	<m>	if n=m, send n packets
S2	Timed Mix	<t>	if timer times out, send n packets
S3	Threshold or Timed Mix	<m, t>	if timer times out, send n packets; elseif n=m {send n packets; reset the timer}
S4	Threshold and Timed Mix	<m, t>	if (timer times out)and (n>m),send n packets
S5	Threshold Pool Mix	<m, f>	if n=m + f, send m randomly chosen packets
S6	Timed Pool Mix	<t, f>	if (timer times out) and (n>f), send n - f randomly chosen packets
S7	Timed Dynamic-Pool Mix	<m, t, f, p>	if (timer times out) and (n>m + f), send max(1,[p(n-f)]) randomly chosen packets

Table. 1

VI. ANALYSIS ON THE ENHANCEMENT OF MIX-NETWORK

In the enhancement of mix network the sender of a message attaches the receiver address to a packet and encrypts it using the mix's public key. Upon receiving a packet, a mix decodes the packet. Different from an ordinary router, a mix usually will not relay the received packet immediately. Rather, it collects several packets and then sends them out in a batch.

A. Traffic Flow Correlation

Traffic flow-correlation used to the adversary either to correlate senders and receivers directly or to greatly reduce the searching time for such a correlation in a mix network. Objective is to correlate an incoming flow to an output link at a Mix and also find the Flow-correlation attack.

B. Detection Metrics

Detection Metrics used to analyze the detection rate of the traffic attacks. Use detection rate, the probability that the adversary correctly correlates flows into and out of a mix, defined as the measure of success for the attack. We will show that, given a sufficient amount of data, known mix strategies fail; that is, the attack achieves close to 100 percent detection rate. This remains true even in batching strategies that sacrifice QoS in favor of security.

VII. CONCLUSION

In this paper, we explored the specific class of traffic analysis attacks and flow correlation attacks. We have examined the anonymity of mix networks under flow correlation attacks. We show a formal model of the adversary and likewise determined the detection rate of the framework. We proposed batching strategies to prevent from several attacks and also relies on every source knowing the best path to the destination. If, for some reason, this path is no longer valid, there may be circumstances where the other network elements already know that, but the source insists the path is correct. As the other network elements perform path caching for efficiency, the source may keep trying to update the path anyway, over and over again. This might lead to routing instability. This "one bad apple ruining the whole barrel" is a tradeoff between distributed and centralized route calculation.

ACKNOWLEDGEMENT

This work has been partially supported by the EC FP7 Programme under Grant No. 257867 - NOVI.

REFERENCES

[1] Y. Zhu, X. Fu, B. Graham, R. Bettati, and W. Zhao. Theoretical analysis of flow based traffic analysis attacks in anonymous communication systems. Technical Report 2005-2-1, Texas A&M University, Computer Science Department, February 2005.

[2] X. Cheng, S. Su, Z. Zhang, H. Wang, F. Yang, Y. Luo, and J.Wang, "Virtual Network Embedding through Topology-Aware Node Ranking," SIGCOMM Computing Comm. Rev., vol. 41, no. 2,pp.38-47,Apr.2011, doi:http:// doi.acm.org /10.1145/ 1971162.1971168

[3] Houidi, W. Louati, W.B. Ameur, and D. Zeghlache(2011), 'Virtual Network Provisioning Across Multiple Substrate Network' vol. 55, no. 2, pp. 1011-1023

[4] C. Papagianni, A. Leivadeas, S. Papavassiliou, V. Maglaris, C.Cervello-Pastor, and A. Monje(2011), 'On the Optimal Allocation of Virtual Resources in Cloud Computing Networks'

[5] Y. Zhu and M.H. Ammar(2006), 'Algorithms for Assigning Substrate Network Resources to Virtual Network Components' Proc. IEEE INFOCOM '06, pp. 1-12.

[6] G. Danezis, R. Dingledine, and N. Mathewson. Mixminion: Design of a Type III Anonymous Remailer Protocol. In *Proceedings of the 2003 IEEE Symposium on Security and Privacy*, May 2003.

[7] F. Zaheer, J. Xiao, and R. Boutaba, "Multi-Provider Service Negotiation and Contracting in Network Virtualization," Proc. IEEE Network Operations and Management Symp.(NOMS), pp.471-478,June2010, doi:10.1109/NOMS.2010.5488487.

[8] Y. Zhu, X. Fu, B. Graham, R. Bettati, and W. Zhao. On flow correlation attacks and counter measures in mix networks. In *Proceedings of Privacy Enhancing Technologies workshop (PET 2004)*, May 2004.

[9] Y. Xin, I. Baldine, A. Mandal, C. Heermann, J. Chase, and A.Yumerefendi, "Embedding Virtual Topologies in Networked Clouds," Proc. Sixth Int'l Conf. Future Internet Technologies (CFI'11), June2011, doi:10.1145/ 2002396. 2002403.

[10] A. Serjantov, R. Dingledine, and P. Syverson. From a trickle to a flood: active attacks on several mix types. In *Proceedings of Information Hiding Workshop*, 2002.

