ISSN (e): 2250-3021, ISSN (p): 2278-8719
PP 80-83

Survey on Data Security in Network Flow Using Obfuscation Technique

Ms. Dikshaa. Rangari[1],Prof. Rashmi Dukhi[2]
[1]Student, Department of Computer Science, G.H.Raisoni Institute of Information Technology
[2]Asst. Professor, Department of Computer Science, G.H.Raisoni Institute of Information Technology

Abstract: *An application encompasses network modeling and simulation, recognition of privacy assaults, and formalization of research results. Indeed, existing techniques for network flow sanitization are vulnerable to different kinds of attacks, and solutions proposed for micro data anonymity cannot be directly applied to network traces. In our previous research, we proposed an obfuscation technique for network flows, providing formal confidentiality guarantees under realistic assumptions about the adversary's knowledge. Put forward an obfuscation technique that leads to confidential guarantee of IP address thus securing the sensitive data. In this paper, we identify the threats posed by the incremental release of network flows and by using SHA3 algorithm we formally prove the achieved confidentiality guarantees. For this operation, a fingerprint is created which is based on the configuration of the system. We group hosts based on the fingerprint and obfuscated address and secure the IP address during the release of incremental network flows. Then, the process of grouping is done using the generated signature. Group intimation is done and the set of IP addresses and signature are compared and the requested signature is send as response. All this processes occur with an intermediate router. Only, the obfuscated signature will be visible to the hacker.*

Keywords:*Security, Incremental release, Obfuscation, Code Security, Code obfuscation techniques, Privacy*

I. Introduction

Obfuscation is the obscuring of intended meaning in communication, making the message confusing, will fully ambiguous, or harder to understand. It may be intentional or unintentional (although the former is usually connected) and may result from circumlocution (yielding wordiness) or from use of jargon or even argot (yielding economy of words but excluding outsiders from the communicative value). Unintended obfuscation in expository writing is usually a natural trait of early drafts in the writing process, when the composition is not yet advanced, and it can be improved with critical thinking and revising, either by the writer or by another person with sufficient reading comprehension and editing skills. Similarly, Internet flows may reveal personal communications among specific individuals, such as e-mail exchanges and chat sessions among them. On the other hand, those datasets may also help an adversary to perform security attacks. For instance, observing the traffic of a target network, an adversary could identify possible bottlenecks to be exploited for denial-of-service (DoS) attacks. For these reasons, several techniques were proposed to sanitize network flows while preserving their utility. Early techniques (e.g., Crypto-PAn) were based on the substitution of the real IP addresses with pseudo-IDs. However, that method proved to be vulnerable to different kinds of attacks, based on the knowledge of network characteristics, or on the capacity to inject bogus flows in the monitored network. More recently, several techniques have been proposed to avoid the re-identification of IP addresses, based on the perturbation of other fields of the flows. However, those techniques do not provide any formal confidentiality guarantee, and it has been recently shown that they are prone to different kinds of attacks. In our previous work, we have presented -obfuscation, an obfuscation technique for network flows, which provides formal confidentiality guarantees under realistic assumptions about the adversary's knowledge, while preserving the utility of released data. In that work, we assumed a single release of the whole dataset of flows. However, the incremental release of network flows represents a clear practical advantage. For instance, suppose that an organization wishes to share a month of network flows. Without the incremental release, it would be necessary to wait until the end of the month to start releasing the dataset. Through incremental releases, the organization could provide a more time sharing of network flows choosing a per-week or even a per-day schedule. Moreover, the incremental release provides important technical advantages. Indeed, the computational costs and the memory requirements for obfuscating a large dataset could be strongly reduced by partitioning the dataset in smaller subsets and by running the obfuscation process independently on each subset. Each IP-group contains at least different IP addresses that appear in. Formally, for each group-ID appearing in a flow, there exists a set of at least IP addresses appearing in a flow in such that, for each group-ID. p2: Each flow is fp-indistinguishable in a set of at least flows in originated by distinct IP addresses belonging to the same IP-group.is undefined if the above properties cannot be satisfied—i.e., if involves less than different IP addresses(it is impossible to enforce p1) or if contains less than flows(it is impossible to enforce p2). An extensive experimental evaluation of the algorithm for incremental (K,j) obfuscation, carried out with billions of real flows generated by the border router of a commercial autonomous system. We made experiments on traffic diversity, statistical analysis of flow fields, and network flow analysis. Our results show that our technique preserves the data quality in both the single and the incremental release. Early techniques for network flow obfuscation were based on the encryption of source and destination IP addresses. However, those techniques proved to be ineffective since an adversary might be able to reidentify message source and destination by other values of network flows. Early techniques were based on the substitution of the real IP addresses with pseudo-IDs more recently several techniques have been proposed to avoid the re-identification of IP addresses, based on the perturbation of other fields of the flows. However, those techniques do not provide any formal confidentiality guarantee. The existing work has assumed a single release of the whole dataset of flows. However, the incremental release of network flows represents a clear practical advantage. In this project to partition hosts in homogeneous groups by Fingerprint based group creation algorithm, we use system details: OS, RAM, Processor, User, IP address. For each host, we built the fingerprint vector by computing on the whole set of flows generated by that host, the mean and standard deviation of each considered feature. Using fingerprint the data will be send to router. Router sends that fingerprint to Host Identity. If finger print is matching in any group then host ID send the data to that fingerprint.

II. Methodology

To identify the threats posed by the incremental release of network flows and by using SHA-3 algorithm and formally prove the achieved confidentiality guarantees. In order to evaluate the effectiveness of our grouping method, To measure the homogeneity of hosts of the same group according to their fingerprint vectors. In this paper we perform the obfuscation for incremental network flows. We illustrate the confidentiality threats and data quality issues involved in the incremental release of network flows. Furthermore, we show how our defense algorithm can be extended to overcome these issues. We also prove the confidentiality guarantees of our defense, as well as the computational complexity of the extended algorithm.

A. Fingerprint based Group creation

Fingerprint creation is based on OS, RAM, Processor, Username and IP address on each node.

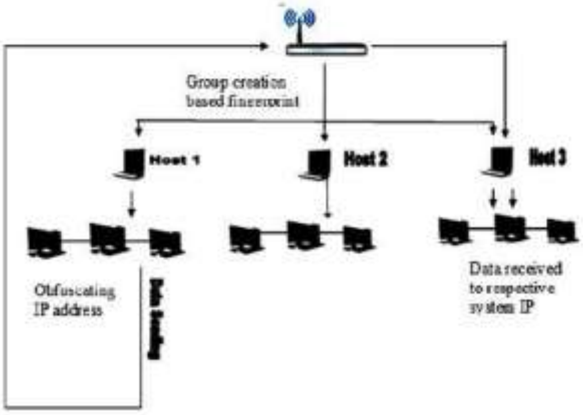

Figure 1: Fingerprint based group creation

Creating fingerprint for each nodes and mapping the nodes. Nodes having similar fingerprint values are grouped together. The goal of our fingerprint-based IP-groups creation method is to enforce property obfuscation while preserving the quality of obfuscated data. In order to reach this goal, IP-groups are created by grouping together IPs whose hosts have a similar fingerprint (i.e., they originate similar flows).

B. Creating Group Identity and Group Intimation

To identify the group, we create a group ID for each group based on the count and relevant score values obtained by the obfuscated IP address and the 32-bit fingerprint values. These fingerprint values are in the form of vectors. Group information is sent to all the nodes and the node which matches the group information sent responds to that host and the data is sent to that corresponding nodes. Markov models are used to create groups of hosts having similar network behavior. In order to enforce anonymity, the real IP address of each network flow is substituted by its group ID being released. This group ID is a unique value and hence it eliminates the redundant or duplicate value of real IP addresses. However, there is neither experimental evidence nor a formal guarantee that, with this statistically driven approach, an adversary applying available domain knowledge cannot re-identify hosts by their fingerprint.

C. Obfuscation of sensitive data in network flows

In network flow a sensitive data is transmitted from source and destination, here we obfuscate the source and destination IP address as a fingerprint. Using fingerprint we transmit a data from source to router. Router sends the data to the entire host IDs. In every host, the host ID is obtained by many-to-many mapping the fingerprint in respective group. Data is sent to the nodes having similar fingerprint. The quality of the data is not compromised.

III. Architecture

In general, the fact that two specific hosts A and B exchanged some message may be considered confidential information. Since IP address uniquely identifies its host, we assume that confidential information in network flows as source and destination address along with the obfuscated address and fingerprint values. Even if we remove this confidentiality information from the network flows there is no violation of privacy and security of the data that is been sent through the flows. Removal of only the IP address may disrupt the utility of the data, but when real IP address is mapped with the obfuscated IP address and the fingerprint generated there will never be any disruption in the transfer and utility of the data. Hence the performance in incremental network flows is higher than the performance of single release of network flows. The main advantage of incremental network flow is manages the single release network flows by initially creating nodes based on networks they reside on. The network confidentiality threats are managed here and data quality issues are neglected here.

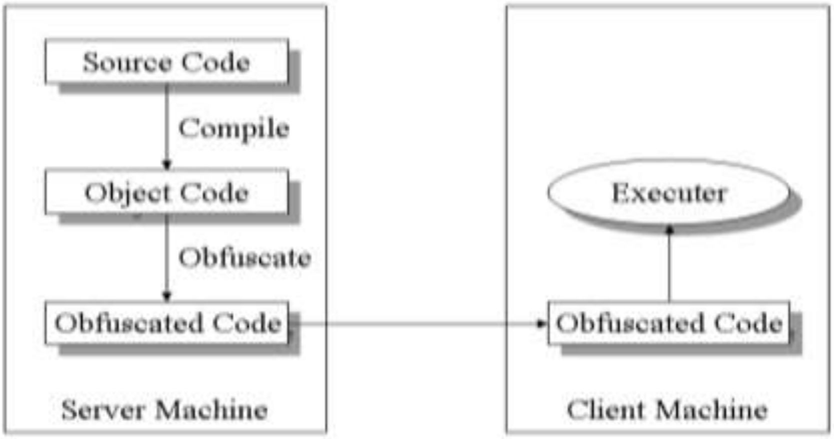

Figure 2: Processing of inputs

A. Network Flow Obfuscation

Since every network flow involves IP address of the sender and receiver nodes, we provide a secure transmission by obfuscating the IP address. This obfuscation is performed by mapping the fingerprint value, the real IP address and configurations of every host in the router. The result of this obfuscation is a 32 bit value which is a combination of text and numerical values. Parameters of this obfuscation involves OS, RAM, Processor, Username and real IP address of each node so the values of each node varies and results in unique obfuscated value which will be very difficult for the intruder to hack and get the data from the incremental release of network flows.

B. Fingerprinting

Fingerprinting is matching the flow field’s values to the characteristics of the target environment (OS, RAM, Processor, distance, range and architecture). The typical values of network flows are types of service, number of bytes, and number of packets per flow. The initial fingerprint obtained is a 128 bit alpha-numeric value which is then compressed to 32 bit value using the SHA3 algorithm. Our algorithm enforces a further level of protection.

Merging of each group in which the group having the closest Hilbert index, such that the union of the two groups satisfies. The set of merged groups is returned, as well as the updated set of IP-groups. This processes using the following algorithm

Input: L:original set of network flows; fp-QI: set of fingerprint quasi- identifiers; k:minimum group size; j:minimum number of fpindistinguishabale flows Output: L*:set of obfuscated network flows

```
Obfuscate ( L, fp-QI, k,j ) begin
IP-groups G:=Group Creation(L,fp-QI,k)
if G={A} s.t |A|< K then retuen 0
L:=SubstituteIPs(L,G)
L*:=0
foreach IP-group Ga E G
La:=Getflows (L,GA)
La*:=Bucketsize (La,fp-QI,j)
L*=L* U La*
end
return L*
end
```

IV. Conclusions

To addressed the challenging research issue of network flow obfuscation. This technique provides formal protection guarantees under realistic assumptions about the adversary's knowledge. A proposed a novel defense algorithm to enforce obfuscation to incremental releases, and SHA-3 proved the confidentiality guarantees. All network flow is maintained in one path and making high confidential for source and destination IP addresses. We have formally proved the confidentiality guarantees provided by the new extended algorithm.

References

[1]. J. King, K. Lakkaraju, and A. J. Slagell, "A taxonomy and adversarial model for attacks against network log anonymization," in Proc. ACMSAC, 2009, pp. 1286–129

[2]. SornaSukanya G, "Rattle Adversary In IP Address Race Of Puzzler Networks", in International Journal of Research in Computer and Communication Technology, Vol 4, Issue 3 , March -2015

[3]. J. Fan, J. Xu, M. H. Ammar, and S. B. Moon, "Prefixpreserving IP address anonymization: Measurement-based security evaluation and a new cryptography-based scheme," Comput. Netw., vol. 46, no. 2, pp.253– 272, 2004.

[4]. Y. Song, S. J. Stolfo, and T. Jebara, "Behavior-based network trafficsynthesis," in Proc. IEEE HST, 2011, pp. 338– 344.

[5]. G. Dewaele, Y. Himura, P. Borgnat, K. Fukuda, P. Abry, O. Michel, R. Fontugne, K. Cho, and H. Esaki, "Unsupervised host behavior classification from connection patterns," Int. J. Netw. Manag., vol. 20, no.5, pp. 317–337, Sep. 2010

[6]. S. E. Coull, C. V. Wright, F. Monrose, M. P. Collins, and M. K. Reiter, "Playing devil's advocate: Inferring sensitive information from anonymized network traces," in Proc. NDSS, 2007.

[7]. A. J. Slagell, K. Lakkaraju, and K. Luo, "FLAIM: A multilevel anonymization framework for computer and network logs," in Proc.LISA, 2006, pp. 63–77.

[8]. S. E. Coull, M. P. Collins, C. V. Wright, F. Monrose, and M. K. Reiter, "On Web browsing privacy in anonymized NetFlows," in Proc.USENIX Security, 2007, pp. 339–352.

[9]. J. Mirkovic, "Privacy-safe network trace sharing via secure queries,"in Proc. ACM NDA, 2008, pp. 3–10.

[10]. J. C. Mogul and M. F. Arlitt, "SC2D: An alternative to trace anonymization," in Proc. MineNet, 2006, pp. 323–328.

[11]. A. Villani, D. Vitali, D. Riboni, C. Bettini, and L. V. Mancini, "Obsidian:a scalable and efficient framework for NetFlow obfuscation," in Proc. IEEE INFOCOM, 2013, pp. 7–8.

[12]. F.McSherry and R. Mahajan, "Differentially private network trace analysis," in Proc. ACM SIGCOMM, 2010, pp. 123–134.

[13]. Coull.S.E, Monrose.F, Reiter.M.K, and Bailey.M proposed "The challenges of effectively anonymizing network data", 2009

[14]. Y.Gu, A. McCallum, and D. F. Towsley, "Detecting anomalies in network traffic using maximum entropy estimation," in Proc. of ACM SIGCOMM Internet Measurement Conference, 2005, pp. 345–350.

[15]. J. Yuan, Z. Li, and R. Yuan, "Information entropy based clustering method for unsupervised internet traffic classification," in Proc. of IEEE International Conference on Communications, 2008, pp. 1588–1592.

[16]. S. E. Coull, F. Monrose, M. K. Reiter, and M. Bailey, "The challenges of effectively anonymizing network data," in Proc. Conference For Homeland Security, 2009, pp. 230–236.

[17]. D. Dunaev and L. Lengyel. "An intermediate level obfuscation method", 2014.