ISSN 23499842(Online), Volume 1, Special Issue 2(ICITET 15), March 2015

International Journal of Innovative Trends and Emerging Technologies

SECURITY OF OBFUSCATED SENSITIVE DATA WITH ADVERSARY KNOWLEDGE IN NETWORK ENVIRONMENT USING MD5 ALGORITHM

BHAGAT SUSHIL S[1], SURENDRA BHARATHI V D[2], VASANTHA KUMAR V[3], UDAYA B[4], ASHOK V[5]
[1,2,3]- UG scholar, [4,5]Assistant professor, Department of Computer Science and Engineering
Rajalakshmi Institute of Technology, Chennai
sushilbhagat6@gmail.com, bharathi.goutham@gmail.com, vasanthstats@gmail.com, udaya.b@ritchennai.edu.in, ashok.v@ritchennai.edu.in

***Abstract*-Large volume of data when transmitted may contain highly sensitive data which should be secured. Existing system is vulnerable to different kinds of attacks which leads to hacking of IP address and it induces several problems in the incremental release of network flows. So, we put forward an obfuscation technique that leads to confidential guarantee of IP address thus securing the sensitive data. This method makes use of MD5 algorithm for obfuscating the IP address into 32-bit signature. For this operation, a fingerprint is created which is based on the configuration of the system. 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. Moreover, this project also solves the threats that are generated in the incremental release of network flows. Applications include identification of security attacks, validation or research result, network modelling and simulation.**

***Index Terms*-Obfuscation, Signature, Group intimation, Incremental release.**

A.INTRODUCTION

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 (see, e.g., [1], [2], [5], [7].King *et al.* in [14] propose an extensive taxonomy of attacks against network flow sanitization methods; these techniques fall into two main categories:

Fingerprinting: Messages reidentification is performed bymatching flows fields' values to the characteristics of thetarget environment (knowledge of network topology and settings, OS and services of target hosts, etc.). Typical reidentifying values for network flows are type of service, TCP flags, number of bytes, and number of packets per flow.

Injection: The adversary injects a sequence of flows in the target network that are easily recognized due to their specific characteristics; e.g., marked with uncommon TCP flags, or following particular patterns.

Additional techniques can be used to exploit the results of the above attacks to decrypt IP addresses of new network flows.In particular, if the IP address encryption is performed with the same key across the whole set of flows (as in most existing defense techniques), and the adversary discovers an IP mapping in one flow, he can decrypt the same IP address in any other flow.Large datasets of real network flows acquired from the Internet are an invaluable resource for the research community. Unfortunately, network flows carry extremely sensitive information, and this discourages the publication of those datasets. 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 the existing system, we proposed an obfuscation technique for network flows, providing formal confidentiality guarantees underrealistic assumptions about the adversary's knowledge. In this paper, we identify the threats posed by the incremental release of network flows and by using MD5 algorithm we formally prove the achieved confidentiality guarantees. An extensive experimental evaluation of the algorithm for incremental obfuscation carried out with billions of real Internet flows, shows that our obfuscation technique preserves the utility of flows for network traffic analysis. Network flows carry extremely sensitive data. Network flow sanitization is vulnerable to different kind of attacks. For example, network flows may reveal personal communication such as e-mail, chatting. These data may also help an adversary to perform security attacks. For this reason in existing technique in network flow obfuscation were based on encryption of source and destination

IP addresses. However, those techniques proved to be ineffective since an adversary might be able to reidentify source and destination IP address.Several techniques were proposed to sanitize network flows while preserving their utility. Early techniques werebased on the substitution of the real IP addresses with pseudo-IDs

More recently, several techniques have been proposed to avoid the reidentification 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 system 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. 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 byFingerprint 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. In order to evaluate the effectiveness of our grouping method, we measure the homogeneity of hosts of the same group according to their fingerprint vectors.

In network flow obfuscation we obfuscate a source and destination IP address. 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.

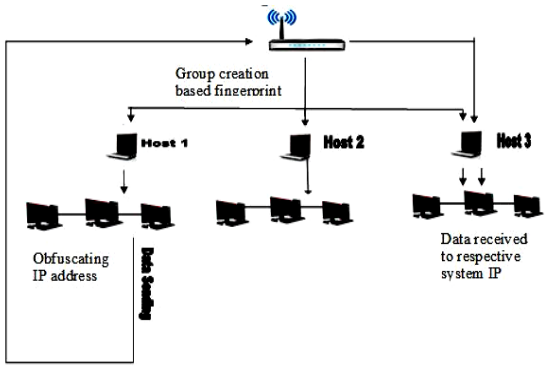
The paper is structured as follows: 1) Finger based group creation 2) Group identity and group intimation 3) Obfuscation of sensitive data in network flow.

B. RELATED WORK

1)Fingerprint based group creation: Fingerprint creation is based on OS, RAM, Processor, Username and IP address on each node. Creating fingerprint for each nodes and mapping the nodes. For the nodes having similar values we create group for that nodes.

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).

Crypto technique, is currently, incorporated within several network flow collector tools.Crypto PAN is a cryptography based sanitization tool for network trace owners to anonymize the IP addresses in their traces in a prefix preserving manner. Crypto PAN has the following properties:1)One-to-One which is the mapping of original IP addresses to anonymized IP addresses.2)In Crypto PAN, the IP address anonymization is prefix preserving. That is, if two original IP addresses share a*k*-bit prefixtheir anonymized mappings will also share a k-bit prefix.3)Crypto PAN allows multiple traces to be sanitized in a consistent way over time and across locations. That is the same IP address in different traces is anonymized to the same address, even though the traces might be sanitized separately at different time and/or at different locations.4)To sanitize traces, trace owners provide Crypto-PAN a secret key. Anonymization consistency across multiple traces is achieved by the use of same key. The construction of Crypto-PAN preserves the secrecy of the key and the randomness of the mapping from an original IP address to its anonymized counterpart.



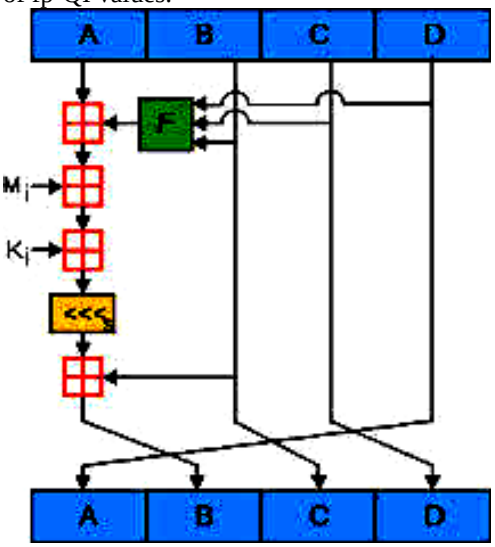
2) Group identity and group intimation: To identify the group, we create a group ID for each at means nodes in which group and id that information is send to all nodes. Markov models are used to create groups of hosts having similar network behavior. In order to enforce anonymity, the real IP address of each flow is substituted by its group ID before being released. However, there is neither experimental evidence nor a formal guarantee that, with this statistically driven approach, an adversary applying available domain knowledge cannot reidentify hosts by their fingerprint.

Network Flows Initializing: Creating Network Flows and generates following fingerprints i.e.., (id, hostname, memory, IP- address etc.…) of that

network and with help of Admin maintaining all Network Flows in one path.

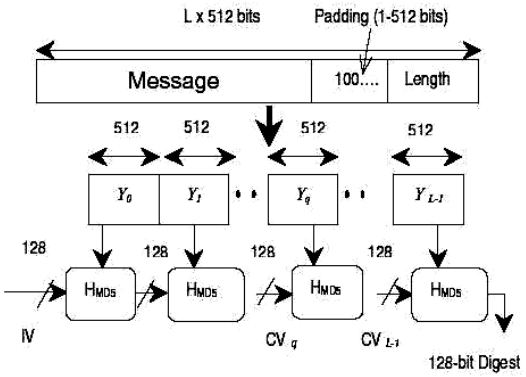
*Enforcing Obfuscation for Single release:*We devised an approximate algorithm; its pseudo code IP's is shown first, IP-groups is created by their fingerprint values. By this, the real IPs in network flows is substituted by the identifier of the IP-group they belongs to one host. After initializing the set of obfuscated flows, for each IP-group, we take the flows generated by the hosts of its IPs, we enforce fp-indistinguishability, and we add the obfuscated flows to all networks. Finally, we return the set of obfuscated flows. And the two networks communicating together with help of their host.

*Enforcing of fp-indistinguishability in Incremental Release:*Impact on Data Quality: This happens, for instance, when a system administrator changes the function of a host (e.g., from DBMS server to SMTP server). Moreover, when dynamic addressing is used, the mapping between an IP address and its host machine changes with time. Hence, hosts with different fingerprints may be associated to the same IP address. The above facts have a negative impact on the quality of obfuscated network flows to overcome this Data Impact first IP-groups are created grouping together IP addresses whose hosts have a similar fingerprint, so that fp-indistinguishability can be enforced introducing a fine-grained generalization of fp-QI values.

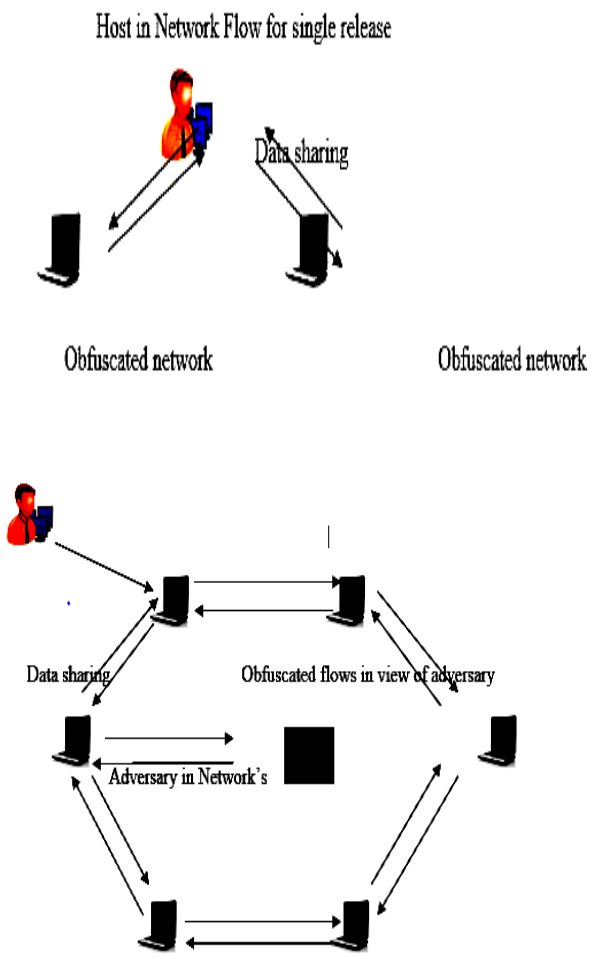


Extension to Incremental Release:
Impact on Confidentiality:This may happen when a host is dismissed from the network the same may happen when a host is inactive for a significantly long period of time—for instance, because it is a personal computer within an office that is closed for holidays. If an adversary has access to this information, and he got to know the mapping among some original IP addresses and their group-ID, he may restrict the cardinality of candidate IP addresses for an IP-group Based on external knowledge about the set of hosts that disappeared, an adversary may be able to perform different kinds of probabilistic reasoning attacks to overcome this Confidentiality impact we proposed an defense algorithm for maintain of incremental network flows over adversary knowledge.The specific objectives of the extended defense algorithm are: 1) to update IP-groups, such that each group has at least elements that occur in; 2) to assign each IP address that never occurred in previous releases to an IP-group. In order to reach these objectives, it is sufficient to modify the defense algorithm for creating IP-groups such that propertyis enforced even for incremental releases.A network flow is obfuscated if it cannot be associated with high confidence to its source and destination IP addresses.

3) Obfuscation of sensitive data in network flow: In this paper we make use of the Message Digest algorithm which is illustrated as follows.



The input is given in the form of multiples of 512 bits. The message length if not in this format, then the process of padding undergoes. The configuration of each system is added and the mean is the body of the message. The output of this algorithm is the 32-bit signature which is used for fingerprint grouping.



IP-groups is created by their fingerprint values. By this, the real IPs in network flows is substituted by the identifier of the IP-group they belongs to one host. After initializing the set of obfuscated flows, for each IP-group, the flows generated by the hosts of its IP address is taken and fp-indistinguishability is enforced and the obfuscated flows is added to all networks. The figure below illustrates the host in network flow for incremental release and the obfuscated flows in view of adversary.

C.*FUTURE WORK*

Future research directions include the extension of our formal model and defense technique to different adversary models. In particular, we aim at addressing the case in which an adversary has external knowledge about the temporal communication pattern of specific hosts and may use this knowledge to reidentify IP addresses in the observed history of obfuscated flows.

D.*ACKNOWLEDGEMENT*

The authors would like to give special thanks to Assistant Professor O. Pandithurai for the significant help in collecting the data and in the processing of the experiments.Feedback from anonymous reviewers also helped to improve the work.

E.*REFERENCES*

[1] T. Brekne and A. Årnes, "Circumventing IP-address pseudonymization,"
in*Proc. ICCCN*, 2005, pp. 43–48.
[2] T. Brekne, A. Årnes, and A. Øslebo, "Anonymization of IP traffic monitoring data: Attacks on two prefix-preserving anonymization schemes and some proposed remedies," in *Proc. 5th Workshop Privacy EnhancingTechnol.*, 2006, vol. 3856, LNCS, pp. 179–196.
[3] M. Burkhart, D. Schatzmann, B. Trammell, E. Boschi, and B. Plattner,"The role of network trace anonymization under attack," *Comput.Commun. Rev.*, vol. 40, no. 1, pp. 5–11, 2010.
[4] A. R. Butz, "Alternative algorithm for Hilbert's space-filling curve,"*IEEE Trans. Comput.*, vol. C-20, no. 4, pp. 424–426, Apr. 1971.
[5] 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.
[6] S. E. Coull, F. Monrose, M. K. Reiter, and M. Bailey, "The challenges of effectively anonymizing network data," in *Proc. CATCH*, 2009, pp.230–236.
[7] S. E. Coull, C. V.Wright, F. Monrose, M. P. Collins, and M. K. Reiter,"Playing devil's advocate: Inferring sensitive information fromanonymized network traces," in *Proc. NDSS*, 2007.
[8] 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,"
[9] C. Dwork, "Differential privacy," in *Proce. ICALP*, 2006, vol. 4052,LNCS, pp. 1–12.
[10] V. Engen, J. Vincent, and K. Phalp, "Exploring discrepancies in findings obtained with the KDD Cup '99 data set," *Intell. Data Anal.*, vol.15, no. 2, pp. 251–276, 2011.

[11] M. Foukarakis, D. Antoniades, and M. Polychronakis, “Deep packet anonymization,” in *Proc. EUROSEC*, 2009, pp. 16–21.

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

[13] Y. Gu, A. McCallum, and D. F. Towsley, “Detecting anomalies in network traffic using maximum entropy estimation,” in *Proc. ACM SIGCOMMIMC*, 2005, pp. 345–350.

[14] J. King, K. Lakkaraju, and A. J. Slagell, “A taxonomy and adversarial model for attacks against network log anonymization,” in *Proc. ACMSAC*, 2009, pp. 1286–1293.

[15] D. Kifer and A. Machanavajjhala, “No free lunch in data privacy,” in *Proc. SIGMOD*, 2011, pp. 193–204.

[16] N.Li, T. Li, and S.Venkatasubramanian, “T-closeness: Privacy beyond k-anonymity and l-diversity,” in *Proc. IEEE ICDE*, 2007, pp. 106–115.

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

[18] A. Machanavajjhala, J. Gehrke, D. Kifer, and M. Venkitasubramaniam,“L-diversity: Privacy beyond k-anonymity,” in *Proc. IEEEICDE*, 2006, p. 24.

[19] J. Mirkovic, “Privacy-safe network trace sharing via secure queries,”in*Proc. ACM NDA*, 2008, pp. 3–1