Data security Implementation for Real Time Internet Packet Traces

Anantha Narasimhan S.
Department of Information Technology
Sri Venketeswara College of Engineering
Sriperumbudur,TN INDIA.
anantha.narasimhan@gmail.com

Sundarram P.V.
Department of Information Technology
Sri Venketeswara College of Engineering
Sriperumbudur,TN INDIA.
pvsundarram@gmail.com

ABSTRACT : The present data transfer and security system will no longer be robust to support the data volumes. There has always been a criticism on the transparency of the existing data security and network security systems. Researchers have been working on new methods of establishing secure data transmission, but in vain. The main reason for their failure may be attributed to the non-availability of the vital "Packet traces" recording real-world Internet traffic. These real time "packet traces are obtained from the internet "tcpdump" and are especially useful for research on traffic dynamics, protocol analysis, workload characterization, and network intrusion detection . However, sharing of Internet packet traces is very limited because real-world traces contain many kinds of sensitive information, such as host addresses, emails, personal web pages, and even authentication keys. The lack of such traces greatly limits research on application protocols. It is especially crippling for network intrusion detection research, forcing researchers to devise synthetic attacks.

In this paper we describe a approach to transform and anonymize packet traces. The paper elaborates on the anonymization of the internet packet traces and corresponding trace transformation The algorithm discussed can anonymize both packet headers and payloads, and can perform application-level transformations such as editing HTTP or SMTP headers, replacing the content of Web items with MD5 hashes, or altering filenames or reply codes that match given patterns.

The paper aslo goes into soving problems that are dicussed in detatiled by allied litreature. The paper mainly concentrates on the methodology of solving problems related to file transfers and anonymization issues. The huge volumes of file transfer that takes place all over the network are recorded by the trace of the ftp activity on the server. This poseses as a potential threat for the network administrators. The paper discuses a new method for anonymizing ftp traces and opens the gates for a new era of high level programming support for the customization of the entire activity of anonymiztaion and supports writing optimized transformation scripts.

Thus the paper aims to shed light on a new trace transformation & anonymization techniques with features for the future, coupled with reliability and frugal use of resources take technology to the masses as well as the researchers, making the world a truly global village. As such, we hope to help open up new opportunities in Internet measurement and network intrusion detection research.

Keywords: packet trace, anonymization, transformation, Internet, privacy, measurement, network intrusion detection

INTRODUCTION

In this information age the technology is developing at a rapid rate, the users increased manifolds and newer forms of threats have been exposed. The researchers are unable to provide a secure mechanism to be implemented in very vital areas such as Network intrusion detection, network management and traffic dynamics. The reason for this is attributed to the non-availability of research material. Though one might point out the World Wide Web, the Internet as a perennial source of information the research scientist are looking for real world Internet packet traces. These packets contain the details about the various transactions across the Internet. These internet packet traces are obtained from the tcp dump that is present the servers and gateways. These are similar to the system logs in a stand alone system. The main advantage of this is that these are real world values. Fortunately or unfortunately the tcp dump contains many sensitive information like the IP addresses and private information like ID 's and passwords. Though this can be a very potential resource for the development of tools that can effectively be used to counter the present day threat in modern networks, this is never available to the researchers . The main reason is that many fear to contribute their tcp dump to them as they fear about the publicity of the private information present in it. It would become a great threat if these go into wrong hands.

The **need of the hour** is a methodology by which we can use this **resource** and **exploit** its power at the same time **safeguarding** the private **information** present in it. With its use newer tools can be created for **Intrusion detection** and traffic dynamics analysis.

Traces are very useful for the **researchers** as it gives them the true picture of the real-world scenario. From the picture they are able to get newer ideas for developing counter mechanism for network intrusion ,etc.

The **aim** of this paper is to **develop a method** by which these internet packets are made available to the researchers at the same time **preserving** the integrity of the **private** information and **confidential matter** present in the trace.

This paper discusses a **scheme** which is capable of anonymizing the **sensitive** information and instills confidence in the minds of people that their **private information** is **not** made publicly **available**. The paper discusses an **algorithm** that reorganizes the packet traces and the meaningful data elements are then processed by the anonymization procedure followed by the transformation. As a result of this the internet packet traces are now made available to the researchers which is devoid only of private information. The required **dynamics** are **preserved** and the relative meaning of the payload is preserved.

This **system differs** from its **predecessors** from the fact that the principle used to remove or rather hide private information is the **filter-in principle** and not the **filter out principle**.

The former retains the implied relation between the payload and header and the total transaction whereas the later removes the entire payload. The sensitive information is retained but in a safe unrevealing form. Due to this more **meaning is implied** and thus **research** can be **carried** out **effectively**.

1 OVERVIEW

The traces are received in form that cannot be processed directly. Therefore the traces are initially converted into or reassembled into semantically meaningful application protocol level data elements.

The data elements are then operated upon by the anonymization algorithm followed by trace transformation to make it coherent. Anonymization is carried out both payloads as well as the headers.

The anonymized data entities are then reconstructed to their original form .This is done by the composer. The trace composer along with the framer is used to construct the original packet traces as if they were obtained from the server's tcp dump but devoid of private information.

The output traces contain well-formed connections: packets have correct checksums and lengths, TCP flows can be reassembled from the resulting packets, and application-protocol data has correct syntax, so that other programs can process the transformed traces in the same way that they handle original tcpdump traces.

The entire process is summed up as follows:

Initially the packet traces are given as the input to the system. The flow reassembler then reassembles the packets into semantically meaningful data entities . As these elements have private information an algorithm operates upon the trace. The algorithm will anonymize the payloads IPs and also the header information and this will also be transformed. During this transformation the correct checksums are computed ,length is chosen correctly. After all this is done the obtained data elements are devoid of private information and is now ready for research purposes. But they have to be restructured again in order to get the real meaning from it. After reorganization the packet's transport protocol dynamics are preserved.

1.1 ANONYMIZATION / TRANSFORMATION

Anonymization means "making unknown or unidentifiable" , therefore by anonymizing the internet packet traces we aim at destroying the private information present in the trace and at the same time preserving the inherent meaning of the information in a more secure and undisputed form.

The information we try to hide through anonymization falls into two categories: *identities*, including identity of users, hosts, and data; and confidential *attributes*, e.g., passwords, or specifics of sensitive user activity .The first step of developing an anonymization scheme is to decide what information in the trace we need to hide. For example, in anonymizing FTP traces, we aim to hide *identities* of clients, private data (hidden files), and private servers; and sensitive *attributes*: e.g., passwords, authentication keys, and in some cases filenames.

The anonymization scheme is very easy to understand as we discuss the various methods available for doing it. The methods that can commonly be implemented are

1. Substitution by Constant Values
2. Numbering in a Sequential Order
3. Hashing
4. Prefix Preserved Mapping

1.1.1 Substitution by Constant Values

One way to anonymize a data element is to substitute the data with a constant, e.g., replace any password with the string "<password>". Constant substitution is usually used to anonymize confidential attributes. Applying constant substitution to identifiers (e.g., IP addresses), however, is generally undesirable, as we would then no longer be able to precisely distinguish objects from one another. Instead,

identifiers are usually anonymized with a 1-1 mapping, such as sequential numbering or hashing, so that the anonymized identifiers are still unique, as follows.

1.1.2 Numbering in a Sequential Order

We can sequentially number all *distinct* identifiers in the order of appearance, e.g., mapping files names to "file1", "file2", etc.

1.1.3 Hashing

One shortcoming of *sequential numbering* is that we have to keep the whole mapping history to maintain a consistent mapping during the anonymization process and across anonymizations. Instead, we can use a hash function as the mapping. Doing so requires no additional state during the anonymization process, and in addition using the same hash function across anonymizations will render a consistent mapping (assuming that the range of the hashing function is large enough so that likelihood of collision is negligible). To preserve confidentiality, the hash function must be one-way and preferably resistant to chosen plain-text attack, so that an adversary can neither discover the input from the output nor compute the hash by themselves. HMAC-MD5 (with a secret key) satisfies these requirements. Assuming the adversary can neither reverse MD5 nor extract the secret HMAC key, *hashing* is as secure as *sequential numbering*.

1.1.4 Prefix-Preserving Mapping

Sometimes it is valuable to preserve some of the structural relationships between the identifiers, which *sequential numbering* and *hashing* cannot do. For example, IP addresses can anonymized in a prefix-preserving way suchthat any two IP addresses sharing a prefix will share a prefix of the same length in their anonymized form. Prefix-preserving mapping can be similarly applied on the directory components of file names.

1.1.5 Trace transformation

Trace transformation is also carried out with anonymization. Trace transformation aims at the transforming the data present from one form to a more coherent form after anonymization process is done. This involves the calculation and correcting of the checksums and lengths of header fields. Trace transformation can be implemented as required by writing short policy scripts. This policy script should adhere to certain constraints so as to make it amenable for verification. The trace transformation is in no way related to the strength of the privacy factor of the system. It is just used to bring about more reality into the anonymized traces and improved cohesion in it.

2 OUR NEW ALGORITHM

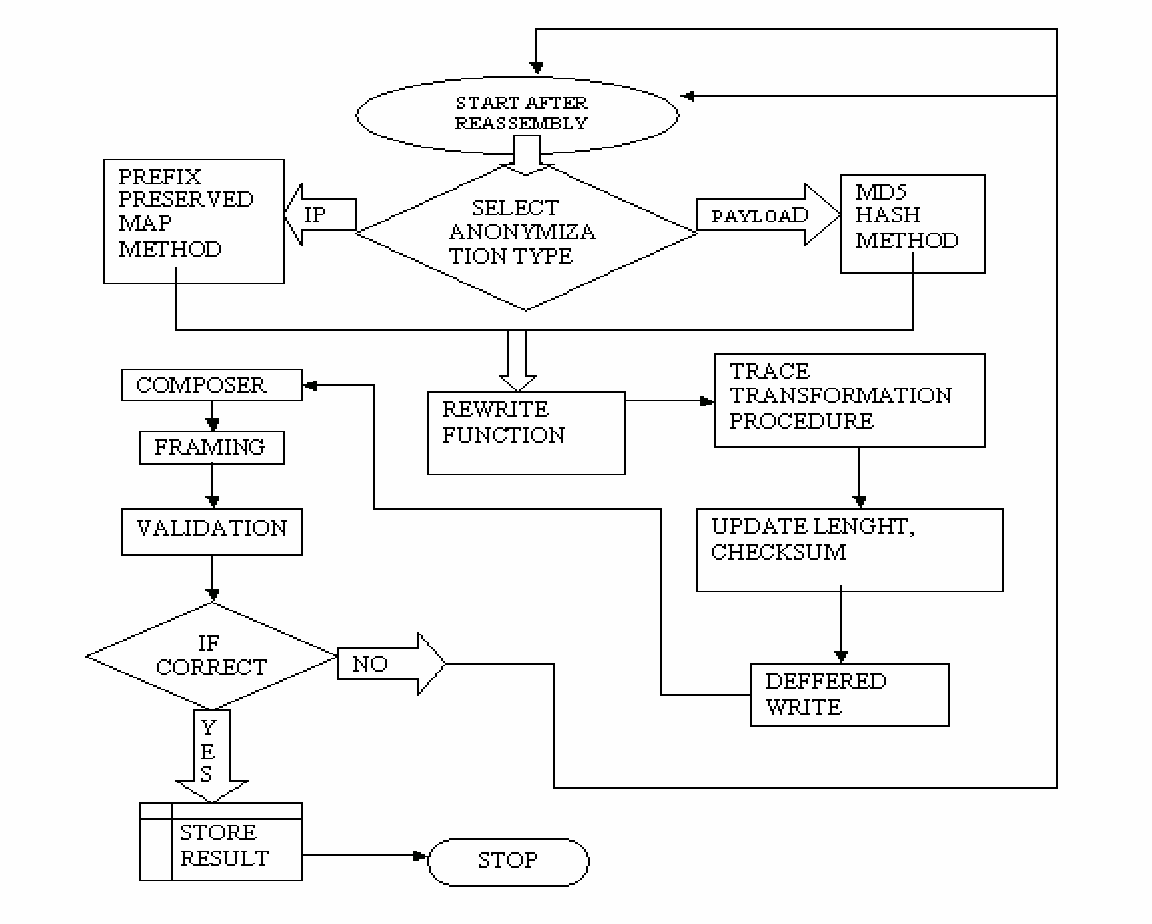
We will now discuss the algorithm that we have developed for the effective anonymization and trace transformation of the vital, sensitive internet packet traces

2.1.1 ALGORITHM

```
STEP 1: Open tcpdump for reading data in it

STEP 2:  Call the reconstruction procedure to reorganized packets and to generate   semantically
  meaningful data entities.

STEP 3: Identify the part of the trace to be anonymized and select type respectively

STEP 4:If it is IP anonymization use prefix preserving mapping scheme

STEP 5: If it is the anonymization of payload then use MD5 hashing to digest the payload

STEP 6: Use REWRITE FUNCTION to write the new data

STEP 7: Call the TRANSFORMATION procedure to correct checksum and header length fields

STEP 8:Use DEFFERED write to update newer values

STEP 9:Reconstruct the trace into packets with the composer

STEP 10:Validate output packets

STEP 11: If ok then exit with success

STEP 12: If failed then goto STEP 2
```

From [1] we get a great insight into the drawbacks of the current proposals. Thus we will now shift our focus on to developing a robust scheme for including FTP (file transfer protocol) traces into the anonymization process. This will improve the effeciency of the system and make the contributed traces immune against any possible data leak and subsequent attacks.

Our objectives are: 1) ensure that the anonymization hides the identity of clients, non-public FTP servers, and non-public files, as well as confidential authentication information; and 2) the anonymization keeps the original request/reply sequence and other nonsensitive information intact. In some ways, these goals and the resulting traces are quite modest. But we believe that the path to site’s becoming open to releasing traces with packet contents is one that must be tread patiently, The algorithm described here will perform anonymization of the Internet packet traces making them absolutely safe and hence openly available for research activities. Trace transformation is also done together in order to bring in integrity of the trace and thus preserving its original protocol dynamics.



3 SAMPLE INPUT (Before Anonymization)

848278028:829593 848278028:893670
848278028:895350 172.16.3.5:8082
172.16.3.3 :8080 2 8 4294967295
4294967295 835418853 170 844
35 GET google.com/svcelogo.gif HTTP/1.0

- 848278028:829593 is the time at which the client made the request
- 848278028:893670 is the time at which the first byte of the server response was seen
- 848278028:895350 is the time at which the last byte of the server response was seen
- 172.16.3.5:8082 is the client IP address and the client port number
- 172.16.3.3 :8080 is the server IP address and the server port number
- 2 is the decimal representation of the client headers bitfield

- 8 is the decimal representation of the server headers bitfield
- the first 4294967295 is the if-modified-since client header value
- the second 4294967295 is the expires server header value
- 835418853 is the last-modified server header value
- 170 is the length of the HTTP response header
- 844 is the length of the response data
- 31 is the length of the request URL
- " GET google.com/svce.gif HTTP/1.0" is the request URL.

4 SAMPLE OUTPUT (After Anonymization)

```
848278028:829593   848278028:893670
848278028:895350      23.240.8.98:1462
207.36.205.194:80    2    8     4294967295
4294967295    835418853    170 844
37   GET 91685044341833134 41..gif  HTTP/1.0
```

- 848278028:829593 is the time at which the client made the request
- 848278028:893670 is the time at which the first byte of the server response was seen
- 848278028:895350 is the time at which the last byte of the server response was seen
- 23.240.8.98:1462 is the anonymized client IP address and the client port number
- 207.36.205.194:80 is the anonymized server IP address and the server port number
- 2 is the decimal representation of the client headers bitfield
- 8 is the decimal representation of the server headers bitfield
- the first 4294967295 is the if-modified-since client header value
- the second 4294967295 is the expires server header value
- 835418853 is the last-modified server header value
- 170 is the length of the HTTP response header
- 844 is the length of the response data
- 37 is the length of the anonymized request URL
- "GET 9168504434183313441..gif HTTP/1.0" is the anonymized request URL.

5 Experimental Implementation

Processing of traces from GOOGLE web server

The traces required for processing were collected from trace donating web servers. Some web servers contribute sanitized Traces for research activities. The original trace was collected by tcpdump recording a retrieval of the www.google.com homepage. The tcpdump output (with wrapped packet summary lines and TCP payloads) of the original trace is in the **APPENDIX**. The algorithm does the following

1. Replaces the data entity with itsMD5 hash value
2. Rewrites the Content-length field to reflect the length of the MD5 hash value.
3. Adds the header: “X-Actual-Data-Length, gap, content-length to record the original Content-length field and how many bytes are actually transferred.

4. The algorithm replaces every occurrence of “Google” in the data entity with “Gooooogle”, instead of replacing the whole data entitywith its MD5 hash value.

5.1 Analysis of the results

1. Data Entity replaced by “ “867119294265e3f445708c3fcfb2144f” ,the MD5 hash value.
2. Modification in header is as: “X-Actual-Data-Length: 2709; gap=0, content-length= 2709”
3. There are four occurrences of “Google” in the original message, thus the Content-length increases from 2709 to 2717.

5.2 Execution time of our system

HTTP analyzer	2358 seconds
HTTP analyzer + anonymizer	18162 seconds
HTTP analyzer + dummy rewriter	3456 seconds

6 System Analysis

6.1Strengths

- Meaning intact
- Sensitive information is filtered in
- Original Dynamics are preserved

6.2 Weakness

- Not standardized
- Technology in nascent stage
- Requires higher Processing Time
- Storage is difficult

7 Further Scope and Development

The first and foremost aim of this paper is to make an effort to bring about the availability of the research material to the scientist. With these material the research activity will boom and there is huge scope for the development of many useful techniques in security and traffic aspects. The paper supports a noble cause of contributing a research to aid researchers . As we can generate stronger methods of anonymization many will come forward confidently to contribute their tcpdump, thus promoting research.

Bibliography

1 P.V. Sundarram, Anantha Narasimhan .S, “Data security implementation of real time internet packet traces,” Eureka,Proceeding of IIT –Kanpur.

2 Anantha Narasimhan .S , P.V. Sundarram, “Data security implementation of real time internet packet traces,”proceeding od REC-Tichy

3 Tony McGregor, Hans-Werner Braun, and Jeff Brown, “The NLANR network analysis infrastructure,” *IEEE Communications Magazine*, vol. 38, no. 5, pp. 122–128, May 2000.

4 “The Internet traffic archive,” http://ita.ee.lbl.gov/, Apr. 2000.

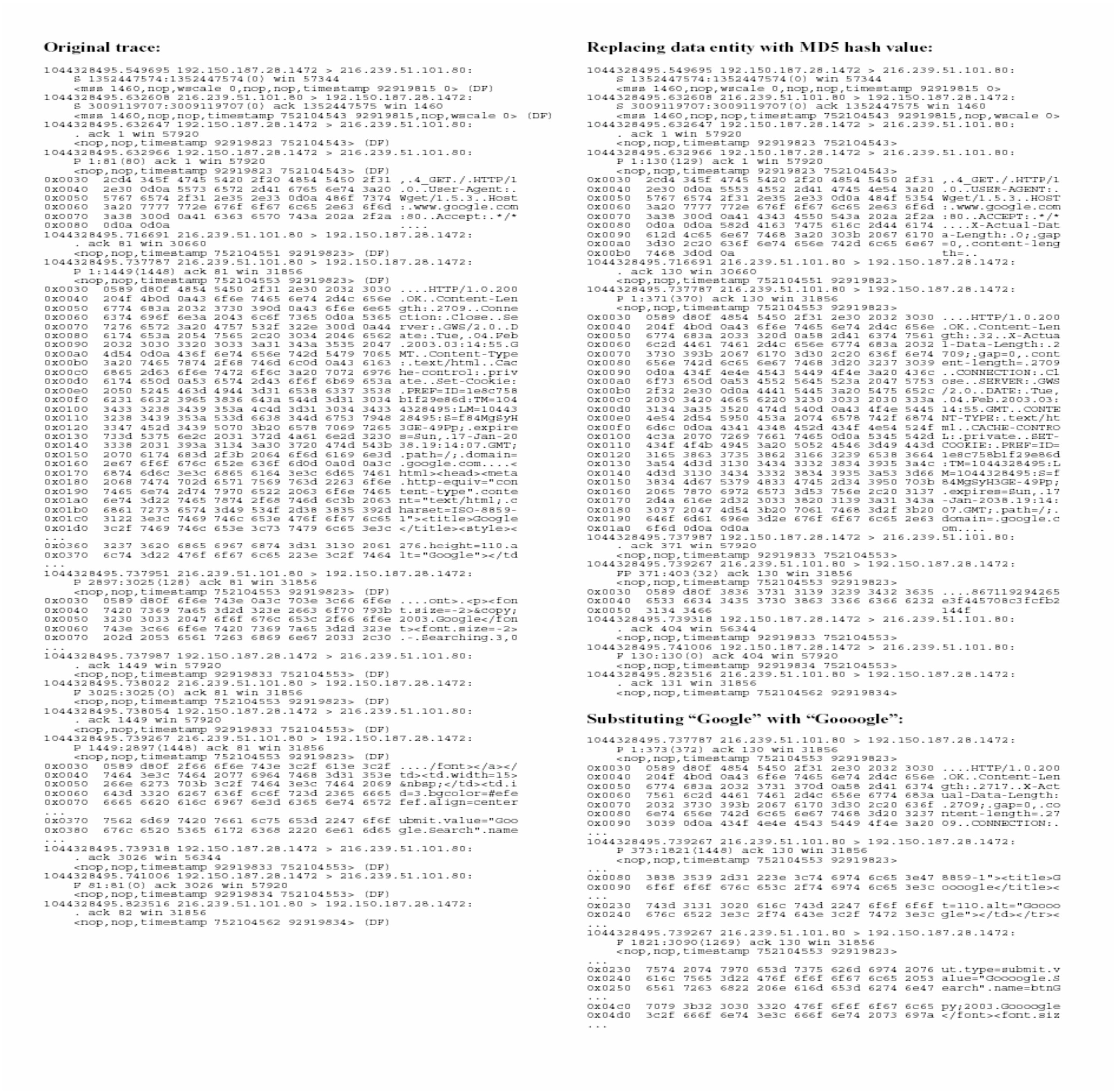
5 J. Xu, J. Fan, M. H. Ammar, and S. B. Moon, “On the design and performance of prefix-preserving IP traffic trace Anonymization,” Tech. Rep., GIT-CC-02-45,College of Computing, Georgia Institute of Technology, Aug. 2002.

6 Lawrence Berkeley National Laboratory Network Research. TCPDump: the Protocol Packet Capture and Dumper Program. http://www.tcpdump.org/.

7 Traffic Measurements for Link Dimensioning* A Case Study Remco van de Meent Aiko Pras Michel Mandjes Hans van den Berg Lambert Nieuwenhuis 13th August 2003 University of Twente CWI

8 Observed Structure of Addresses in IP Traffic Eddie Kohler Jinyang Li ,Vern Paxson, Scott Shenker ICSI Center for Internet Research MIT Lab for Computer Science

APPENDIX



Original trace:

```
1044328495.549695 192.150.187.28.1472 > 216.239.51.101.80:
     S 1352447574:1352447574(0) win 57344
     <mss 1460,nop,wscale 0,nop,nop,timestamp 92919815 0> (DF)
1044328495.632608 216.239.51.101.80 > 192.150.187.28.1472:
     S 3009119707:3009119707(0) ack 1352447575 win 1460
     <mss 1460,nop,nop,timestamp 752104543 92919815,nop,wscale 0> (DF)
1044328495.632647 192.150.187.28.1472 > 216.239.51.101.80:
     . ack 1 win 57920
     <nop,nop,timestamp 92919823 752104543> (DF)
1044328495.632966 192.150.187.28.1472 > 216.239.51.101.80:
     P 1:81(80) ack 1 win 57920
     <nop,nop,timestamp 92919823 752104543> (DF)
0x0030   2cd4 345f 4745 5420 2f20 4854 5450 2f31 ,.4_GET./.HTTP/1
0x0040   2e30 0d0a 5573 6572 2d41 6765 6e74 3a20 .0..User-Agent:.
0x0050   5767 6574 2f31 2e35 2e33 0d0a 486f 7374 Wget/1.5.3..Host
0x0060   3a20 7777 772e 676f 6f67 6c65 2e63 6f6d :.www.google.com
0x0070   3a38 300d 0a41 6363 6570 743a 202a 2f2a :80..Accept:.*/*
0x0080   0d0a 0d0a                               ....
1044328495.716691 216.239.51.101.80 > 192.150.187.28.1472:
     . ack 81 win 30660
     <nop,nop,timestamp 752104551 92919823> (DF)
1044328495.737787 216.239.51.101.80 > 192.150.187.28.1472:
     P 1:1449(1448) ack 81 win 31856
     <nop,nop,timestamp 752104553 92919823> (DF)
0x0030   0589 d80f 4854 5450 2f31 2e30 2032 3030 ....HTTP/1.0.200
0x0040   204f 4b0d 0a43 6f6e 7465 6e74 2d4c 656e .OK..Content-Len
0x0050   6774 683a 2032 3730 390d 0a43 6f6e 6e65 gth:.2709..Conne
0x0060   6374 696f 6e3a 2043 6c6f 7365 0d0a 5365 ction:.Close..Se
0x0070   7276 6572 3a20 4757 532f 322e 300d 0a44 rver:.GWS/2.0..D
0x0080   6174 653a 2054 7565 2c20 3034 2046 6562 ate:.Tue,.04.Feb
0x0090   2032 3030 3320 3033 3a31 343a 3535 2047 .2003.03:14:55.G
0x00a0   4d54 0d0a 436f 6e74 656e 742d 5479 7065 MT..Content-Type
0x00b0   3a20 7465 7874 2f68 746d 6c0d 0a43 6163 :.text/html..Cac
0x00c0   6865 2d63 6f6e 7472 6f6c 3a20 7072 6976 he-control:.priv
0x00d0   6174 650d 0a53 6574 2d43 6f6f 6b69 653a ate..Set-Cookie:
0x00e0   2050 5245 463d 4944 3d31 6538 6337 3538 .PREF=ID=1e8c758
0x00f0   6231 6632 3965 3836 643a 544d 3d31 3034 b1f29e86d:TM=104
0x0100   3433 3238 3439 353a 4c4d 3d31 3034 3433 4328495:LM=10443
0x0110   3238 3439 353a 533d 6638 344d 6753 7948 28495:S=f84MgSyH
0x0120   3347 452d 3439 5070 3b20 6578 7069 7265 3GE-49Pp;.expire
0x0130   733d 5375 6e2c 2031 372d 4a61 6e2d 3230 s=Sun,.17-Jan-20
0x0140   3338 2031 393a 3134 3a30 3720 474d 543b 38.19:14:07.GMT;
0x0150   2070 6174 683d 2f3b 2064 6f6d 6169 6e3d .path=/;.domain=
0x0160   2e67 6f6f 676c 652e 636f 6d0d 0a0d 0a3c .google.com....<
0x0170   6874 6d6c 3e3c 6865 6164 3e3c 6d65 7461 html><head><meta
0x0180   2068 7474 702d 6571 7569 763d 2263 6f6e .http-equiv="con
0x0190   7465 6e74 2d74 7970 6522 2063 6f6e 7465 tent-type".conte
0x01a0   6e74 3d22 7465 7874 2f68 746d 6c3b 2063 nt="text/html;.c
0x01b0   6861 7273 6574 3d49 534f 2d38 3835 392d harset=ISO-8859-
0x01c0   3122 3e3c 7469 746c 653e 476f 6f67 6c65 1"><title>Google
0x01d0   3c2f 7469 746c 653e 3c73 7479 6c65 3e3c </title><style><
...
0x0360   3237 3620 6865 6967 6874 3d31 3130 2061 276.height=110.a
0x0370   6c74 3d22 476f 6f67 6c65 223e 3c2f 7464 lt="Google"></td
...
1044328495.737951 216.239.51.101.80 > 192.150.187.28.1472:
     P 2897:3025(128) ack 81 win 31856
     <nop,nop,timestamp 752104553 92919823> (DF)
0x0030   0589 d80f 6f6e 743e 0a3c 703e 3c66 6f6e ....ont>.<p><fon
0x0040   7420 7369 7a65 3d2d 323e 2663 6f70 793b t.size=-2>&copy;
0x0050   3230 3033 2047 6f6f 676c 653c 2f66 6f6e 2003.Google</fon
0x0060   743e 3c66 6f6e 7420 7369 7a65 3d2d 323e t><font.size=-2>
0x0070   202d 2053 6561 7263 6869 6e67 2033 2c30 .-.Searching.3,0
...
1044328495.737987 192.150.187.28.1472 > 216.239.51.101.80:
     . ack 1449 win 57920
     <nop,nop,timestamp 92919833 752104553> (DF)
1044328495.738022 216.239.51.101.80 > 192.150.187.28.1472:
     F 3025:3025(0) ack 81 win 31856
     <nop,nop,timestamp 752104553 92919823> (DF)
1044328495.738054 192.150.187.28.1472 > 216.239.51.101.80:
     . ack 1449 win 57920
     <nop,nop,timestamp 92919833 752104553> (DF)
1044328495.739267 216.239.51.101.80 > 192.150.187.28.1472:
     P 1449:2897(1448) ack 81 win 31856
     <nop,nop,timestamp 752104553 92919823> (DF)
0x0030   0589 d80f 2f66 6f6e 743e 3c2f 613e 3c2f ..../font></a></
0x0040   7464 3e3c 7464 2077 6964 7468 3d31 353e td><td.width=15>
0x0050   266e 6273 703b 3c2f 7464 3e3c 7464 2069 &nbsp;</td><td.i
0x0060   643d 3320 6267 636f 6c6f 723d 2365 6665 d=3.bgcolor=#efe
0x0070   6665 6620 616c 6967 6e3d 6365 6e74 6572 fef.align=center
...
0x0370   7562 6d69 7420 7661 6c75 653d 2247 6f6f ubmit.value="Goo
0x0380   676c 6520 5365 6172 6368 2220 6e61 6d65 gle.Search".name
...
1044328495.739318 192.150.187.28.1472 > 216.239.51.101.80:
     . ack 3026 win 56344
     <nop,nop,timestamp 92919833 752104553> (DF)
1044328495.741006 192.150.187.28.1472 > 216.239.51.101.80:
     F 81:81(0) ack 3026 win 57920
     <nop,nop,timestamp 92919834 752104553> (DF)
1044328495.823516 216.239.51.101.80 > 192.150.187.28.1472:
     . ack 82 win 31856
     <nop,nop,timestamp 752104562 92919834> (DF)
```

Replacing data entity with MD5 hash value:

```
1044328495.549695 192.150.187.28.1472 > 216.239.51.101.80:
     S 1352447574:1352447574(0) win 57344
     <mss 1460,nop,wscale 0,nop,nop,timestamp 92919815 0>
1044328495.632608 216.239.51.101.80 > 192.150.187.28.1472:
     S 3009119707:3009119707(0) ack 1352447575 win 1460
     <mss 1460,nop,nop,timestamp 752104543 92919815,nop,wscale 0>
1044328495.632647 192.150.187.28.1472 > 216.239.51.101.80:
     . ack 1 win 57920
     <nop,nop,timestamp 92919823 752104543>
1044328495.632966 192.150.187.28.1472 > 216.239.51.101.80:
     P 1:130(129) ack 1 win 57920
     <nop,nop,timestamp 92919823 752104543>
0x0030   2cd4 345f 4745 5420 2f20 4854 5450 2f31 ,.4_GET./.HTTP/1
0x0040   2e30 0d0a 5553 4552 2d41 4745 4e54 3a20 .0..USER-AGENT:.
0x0050   5767 6574 2f31 2e35 2e33 0d0a 484f 5354 Wget/1.5.3..HOST
0x0060   3a20 7777 772e 676f 6f67 6c65 2e63 6f6d :.www.google.com
0x0070   3a38 300d 0a41 4343 4550 543a 202a 2f2a :80..ACCEPT:.*/*
0x0080   0d0a 0d0a 582d 4163 7475 616c 2d44 6174 ....X-Actual-Dat
0x0090   612d 4c65 6e67 7468 3a20 303b 2067 6170 a-Length:.0;.gap
0x00a0   3d30 2c20 636f 6e74 656e 742d 6c65 6e67 =0,.content-leng
0x00b0   7468 3d0d 0a                            th=..
1044328495.716691 216.239.51.101.80 > 192.150.187.28.1472:
     . ack 130 win 30660
     <nop,nop,timestamp 752104551 92919823>
1044328495.737787 216.239.51.101.80 > 192.150.187.28.1472:
     P 1:371(370) ack 130 win 31856
     <nop,nop,timestamp 752104553 92919823>
0x0030   0589 d80f 4854 5450 2f31 2e30 2032 3030 ....HTTP/1.0.200
0x0040   204f 4b0d 0a43 6f6e 7465 6e74 2d4c 656e .OK..Content-Len
0x0050   6774 683a 2033 320d 0a58 2d41 6374 7561 gth:.32..X-Actua
0x0060   6c2d 4461 7461 2d4c 656e 6774 683a 2032 l-Data-Length:.2
0x0070   3730 393b 2067 6170 3d30 2c20 636f 6e74 709;.gap=0,.cont
0x0080   656e 742d 6c65 6e67 7468 3d20 3237 3039 ent-length=.2709
0x0090   0d0a 434f 4e4e 4543 5449 4f4e 3a20 436c ..CONNECTION:.Cl
0x00a0   6f73 650d 0a53 4552 5645 523a 2047 5753 ose..SERVER:.GWS
0x00b0   2f32 2e30 0d0a 4441 5445 3a20 5475 652c /2.0..DATE:.Tue,
0x00c0   2030 3420 4665 6220 3230 3033 2030 333a .04.Feb.2003.03:
0x00d0   3134 3a35 3520 474d 540d 0a43 4f4e 5445 14:55.GMT..CONTE
0x00e0   4e54 2d54 5950 453a 2074 6578 742f 6874 NT-TYPE:.text/ht
0x00f0   6d6c 0d0a 4341 4348 452d 434f 4e54 524f ml..CACHE-CONTRO
0x0100   4c3a 2070 7269 7661 7465 0d0a 5345 542d L:.private..SET-
0x0110   434f 4f4b 4945 3a20 5052 4546 3d49 443d COOKIE:.PREF=ID=
0x0120   3165 3863 3735 3862 3166 3239 6538 3664 1e8c758b1f29e86d
0x0130   3a54 4d3d 3130 3434 3332 3834 3935 3a4c :TM=1044328495:L
0x0140   4d3d 3130 3434 3332 3834 3935 3a53 3d66 M=1044328495:S=f
0x0150   3834 4d67 5379 4833 4745 2d34 3950 703b 84MgSyH3GE-49Pp;
0x0160   2065 7870 6972 6573 3d53 756e 2c20 3137 .expires=Sun,.17
0x0170   2d4a 616e 2d32 3033 3820 3139 3a31 343a -Jan-2038.19:14:
0x0180   3037 2047 4d54 3b20 7061 7468 3d2f 3b20 07.GMT;.path=/;.
0x0190   646f 6d61 696e 3d2e 676f 6f67 6c65 2e63 domain=.google.c
0x01a0   6f6d 0d0a 0d0a                          om....
1044328495.737987 192.150.187.28.1472 > 216.239.51.101.80:
     . ack 371 win 57920
     <nop,nop,timestamp 92919833 752104553>
1044328495.739267 216.239.51.101.80 > 192.150.187.28.1472:
     FP 371:403(32) ack 130 win 31856
     <nop,nop,timestamp 752104553 92919823>
0x0030   0589 d80f 3836 3731 3139 3239 3432 3635 ....867119294265
0x0040   6533 6634 3435 3730 3863 3366 6366 6232 e3f445708c3fcfb2
0x0050   3134 3466                               144f
1044328495.739318 192.150.187.28.1472 > 216.239.51.101.80:
     . ack 404 win 56344
     <nop,nop,timestamp 92919833 752104553>
1044328495.741006 192.150.187.28.1472 > 216.239.51.101.80:
     F 130:130(0) ack 404 win 57920
     <nop,nop,timestamp 92919834 752104553>
1044328495.823516 216.239.51.101.80 > 192.150.187.28.1472:
     . ack 131 win 31856
     <nop,nop,timestamp 752104562 92919834>
```

Substituting “Google” with “Goooogle”:

```
1044328495.737787 216.239.51.101.80 > 192.150.187.28.1472:
     P 1:373(372) ack 130 win 31856
     <nop,nop,timestamp 752104553 92919823>
0x0030   0589 d80f 4854 5450 2f31 2e30 2032 3030 ....HTTP/1.0.200
0x0040   204f 4b0d 0a43 6f6e 7465 6e74 2d4c 656e .OK..Content-Len
0x0050   6774 683a 2032 3731 370d 0a58 2d41 6374 gth:.2717..X-Act
0x0060   7561 6c2d 4461 7461 2d4c 656e 6774 683a ual-Data-Length:
0x0070   2032 3730 393b 2067 6170 3d30 2c20 636f .2709;.gap=0,.co
0x0080   6e74 656e 742d 6c65 6e67 7468 3d20 3237 ntent-length=.27
0x0090   3039 0d0a 434f 4e4e 4543 5449 4f4e 3a20 09..CONNECTION:.
...
1044328495.739267 216.239.51.101.80 > 192.150.187.28.1472:
     P 373:1821(1448) ack 130 win 31856
     <nop,nop,timestamp 752104553 92919823>
...
0x0080   3838 3539 2d31 223e 3c74 6974 6c65 3e47 8859-1"><title>G
0x0090   6f6f 6f6f 676c 653c 2f74 6974 6c65 3e3c ooogle</title><
...
0x0230   743d 3131 3020 616c 743d 2247 6f6f 6f6f t=110.alt="Goooo
0x0240   676c 6522 3e3c 2f74 643e 3c2f 7472 3e3c gle"></td></tr><
...
1044328495.739267 216.239.51.101.80 > 192.150.187.28.1472:
     F 1821:3090(1269) ack 130 win 31856
     <nop,nop,timestamp 752104553 92919823>
...
0x0230   7574 2074 7970 653d 7375 626d 6974 2076 ut.type=submit.v
0x0240   616c 7565 3d22 476f 6f6f 6f67 6c65 2053 alue="Goooogle.S
0x0250   6561 7263 6822 206e 616d 653d 6274 6e47 earch".name=btnG
...
0x04c0   7079 3b32 3030 3320 476f 6f6f 6f67 6c65 py;2003.Goooogle
0x04d0   3c2f 666f 6e74 3e3c 666f 6e74 2073 697a </font><font.siz
...
```