Technical Disclosure Commons

Defensive Publications Series

December 21, 2018

OBFUSCATION AND ANONYMIZATION TECHNIQUES FOR NETWORK DATA SETS FOR MACHINE LEARNING

Ralf Rantzau

Sebastian Jeuk

Gonzalo Salgueiro

OBFUSCATION AND ANONYMIZATION TECHNIQUES FOR NETWORK DATA SETS FOR MACHINE LEARNING

AUTHORS:
Ralf Rantzau
Sebastian Jeuk
Gonzalo Salgueiro

ABSTRACT

Techniques are described herein for securing data used for a machine learning algorithm. The frequency or top-k values calculated over time of the respective network traffic feature data sets are used instead of the actual data or a set thereof (this can also be extended to any other data sets). Here, the frequency represents the actual data and thereby obfuscates potential sensitive information that should not be used within an oftentimes shared cloud machine learning application.

DETAILED DESCRIPTION

Sensitive traffic is often shared over networks, sometimes containing Personally Identifying Information (PII). This is a problem that has gotten significant air-time over the past few years after several malicious usages and attacks making use of such PII. Described herein is an approach to introduce obfuscation techniques whereby sensitive data is anonymized during pattern recognition in data transmitted via the network.

With the introduction of privacy laws such as General Data Protection Regulation (GDPR), more and more security questions arise as to how best to manage and protect individuals' data. This concern is also valid for data in transit on the network. To assure transparency when applying machine learning approaches, a top-k-based method is provided to hide network-specific information and mask them behind top-k values. These top-k rankings may be derived using probabilistic algorithms based on memory-efficient data structures such as the Count-Min Sketch.

The top-k value is introduced as a way to provide anonymity within a machine learning environment. Instead of feeding the actual clear-text data to the machine learning engine, top-k values are calculated on traffic feature sets. These feature sets may be defined up-front by the administrator and describe flows in the network. The more accurate and the more traffic feature sets are defined, the more input the top-k calculation receives. The top-k values are calculated frequently (also adjustable) and generate a data structure over time that allows machine learning algorithms to detect certain patterns.

In one example, a traffic feature set consisting of the source Internet Protocol (IP) address and the destination IP address has different top-k values over time. Between peak hours this particular stream ranks high in the top-k list, while during other times it does not. Machine learning can be used over time to discover these kind of patterns on an anonymized value. The machine learning engine is then able to detect anomalies in case a certain top-k value pair is acting out.

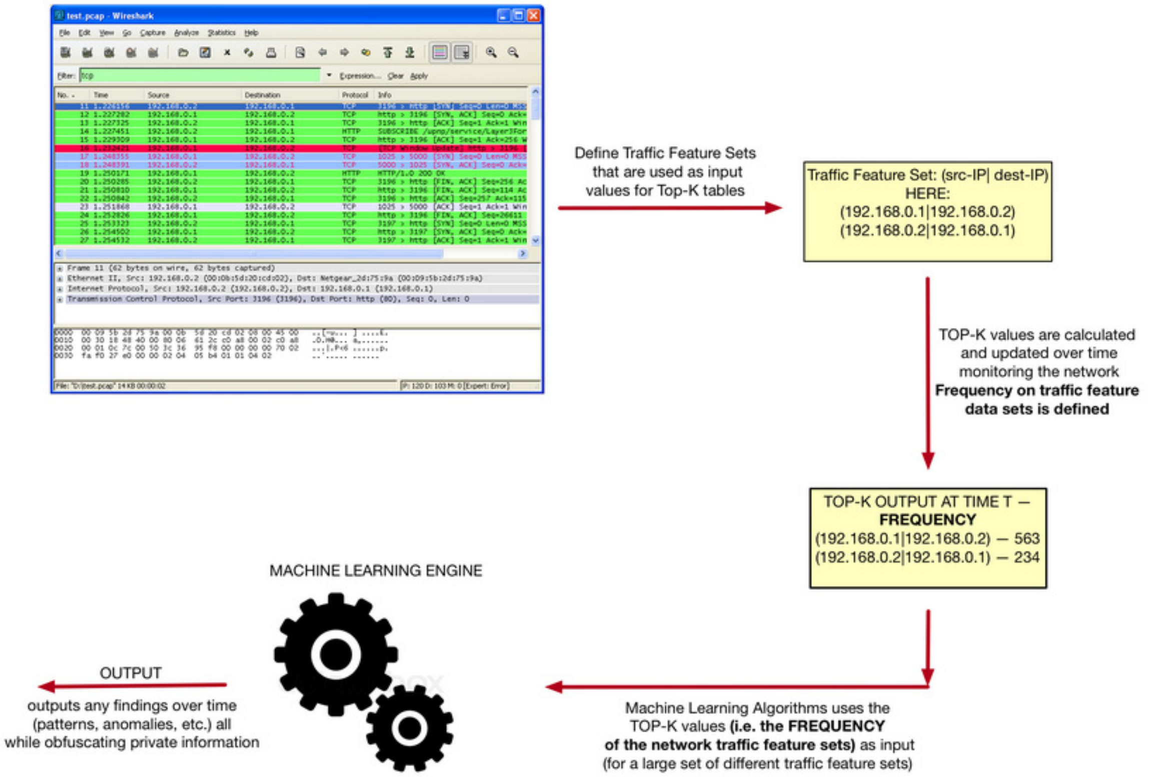
Example traffic feature sets that characterize a data flow may include the following:

- (SRC-IP || DEST-IP)
- (SRC-IP || SRC-PORT || DEST-IP || DEST-PORT || PROTO)

The top-k values are calculated over time for every packet received, based on different traffic feature sets. Over time, the top-k values for these traffic feature sets generate a unique and specific pattern (i.e., frequency) for packets and flows without relying on actual information transmitted on the network.

Machine learning may now be applied over time to analyze behaviors in top-k rankings to analyze whether packets with the traffic flow sets for the top-k values have changed and, if so, how. This approach is able to detect certain behavioral characteristics, such as an anomaly increase over time for a specific top-k (traffic feature set). This information can be used to detect patterns in traffic flows without actually analyzing the data sent across the network.

Figure 1 below illustrates an example flow.



```
test.pcap - Wireshark
File  Edit  View  Go  Capture  Analyze  Statistics  Help
Filter: tcp                                   Expression...  Clear  Apply

No. -  Time      Source        Destination   Protocol  Info
   11 1.226516  192.168.0.2   192.168.0.1   TCP       3196 > http [SYN] Seq=0 Len=0 MSS
   12 1.227282  192.168.0.1   192.168.0.2   TCP       http > 3196 [SYN, ACK] Seq=0 Ack=
   13 1.227325  192.168.0.2   192.168.0.1   TCP       3196 > http [ACK] Seq=1 Ack=1 Win
   14 1.227451  192.168.0.2   192.168.0.1   HTTP      SUBSCRIBE /upnp/service/Layer3For
   15 1.229309  192.168.0.1   192.168.0.2   TCP       http > 3196 [ACK] Seq=1 Ack=256 W
   16 1.232421  192.168.0.1   192.168.0.2   TCP       [TCP Window Update] http > 3196 [
   17 1.248355  192.168.0.1   192.168.0.2   TCP       1025 > 5000 [SYN] Seq=0 Len=0 MSS
   18 1.248391  192.168.0.2   192.168.0.1   TCP       5000 > 1025 [SYN, ACK] Seq=0 Ack=
   19 1.250171  192.168.0.1   192.168.0.2   HTTP      HTTP/1.0 200 OK
   20 1.250285  192.168.0.2   192.168.0.1   TCP       3196 > http [FIN, ACK] Seq=256 Ac
   21 1.250810  192.168.0.1   192.168.0.2   TCP       http > 3196 [FIN, ACK] Seq=114 Ac
   22 1.250842  192.168.0.2   192.168.0.1   TCP       3196 > http [ACK] Seq=257 Ack=115
   23 1.251868  192.168.0.1   192.168.0.2   TCP       1025 > 5000 [ACK] Seq=1 Ack=1 Win
   24 1.252826  192.168.0.1   192.168.0.2   TCP       http > 3196 [FIN, ACK] Seq=26611
   25 1.253323  192.168.0.2   192.168.0.1   TCP       3197 > http [SYN] Seq=0 Len=0 MSS
   26 1.254502  192.168.0.1   192.168.0.2   TCP       http > 3197 [SYN, ACK] Seq=0 Ack=
   27 1.254532  192.168.0.2   192.168.0.1   TCP       3197 > http [ACK] Seq=1 Ack=1 Win

+ Frame 11 (62 bytes on wire, 62 bytes captured)
+ Ethernet II, Src: 192.168.0.2 (00:0b:5d:20:cd:02), Dst: Netgear_2d:75:9a (00:09:5b:2d:75:9a)
+ Internet Protocol, Src: 192.168.0.2 (192.168.0.2), Dst: 192.168.0.1 (192.168.0.1)
+ Transmission Control Protocol, Src Port: 3196 (3196), Dst Port: http (80), Seq: 0, Len: 0

0000  00 09 5b 2d 75 9a 00 0b  5d 20 cd 02 08 00 45 00   ..[-u... ] .....E.
0010  00 30 18 48 40 00 80 06  61 2c c0 a8 00 02 c0 a8   .0.H@... a,......
0020  00 01 0c 7c 00 50 3c 36  95 f8 00 00 00 00 70 02   ...|.P<6 ......p.
0030  fa f0 27 e0 00 00 02 04  05 b4 01 01 04 02         ..'..... ......

File: "D:\test.pcap" 14 KB 00:00:02        P: 120 D: 103 M: 0 [Expert: Error]
```

Define Traffic Feature Sets
that are used as input
values for Top-K tables
Traffic Feature Set: (src-IP| dest-IP)
HERE:
(192.168.0.1|192.168.0.2)
(192.168.0.2|192.168.0.1)
TOP-K values are calculated
and updated over time
monitoring the network
Frequency on traffic feature
data sets is defined
TOP-K OUTPUT AT TIME T —
FREQUENCY
(192.168.0.1|192.168.0.2) — 563
(192.168.0.2|192.168.0.1) — 234
MACHINE LEARNING ENGINE
OUTPUT
outputs any findings over time
(patterns, anomalies, etc.) all
while obfuscating private information
Machine Learning Algorithms uses the
TOP-K values (i.e. the FREQUENCY
of the network traffic feature sets) as input
(for a large set of different traffic feature sets)

Figure 1

In one use-case, a multi-cloud system hosts a large set of tenants where resources are shared. A security concern is present that data may be leaked between tenants. For machine learning applications within such an environment, it is important that the algorithm uses obfuscated data rather than the actual data to analyze and define patterns over time. To achieve this, a top-k interim step is used between the source data and the machine learning algorithm. The top-k calculation is used to obfuscate the data. Instead of directly accessing information captured for example on the network (e.g., IP addresses, sensitive data/patterns, etc.), traffic feature data sets are defined that are used as input for the top-k algorithm. The top-k values that are used as input parameters for the machine learning algorithm are received as an output. The top-k values are numerical values that change over time and define a pattern on the frequency of specific traffic feature data sets. The frequency of those is the obfuscated pattern over time with which a machine learning algorithm can calculate recommendations.

The recommendation received over time is based on the frequency of the top-k values. These can be matched against hash values of the actual network traffic data sets.

Feedback through recommendation is performed through a reverse lookup specific within a tenant environment, thereby remaining anonymous in shared parts of the machine learning application.

Instead of applying the machine learning algorithm directly to the data captured on the network, the data is run through a top-k algorithm first before applying the machine learning algorithm on the dynamically changing rankings. Over time the machine learning algorithm is able to detect a pattern regarding how the defined network traffic feature data sets change. This may provide insight into patterns/behaviors within the network without directly using the data. To further obfuscate information, the top-k algorithm can run using a hash from the network traffic feature data set. This means that the machine learning algorithm is using a hash as well as a dynamically changing ranking as its input parameters. Calculating the learnings over time results in an output that is specific to the hash. As a feedback loop, the machine learning algorithm returns the recommendations over time for a specific hash. This hash can be looked up within the security domain to obtain an understanding of the network traffic feature data set specific to a tenant, where obfuscation is no longer necessary.

A network traffic feature data set may be defined for use as input values for the top-k calculation. The top-k calculation may be used to generate frequency of these network traffic feature data sets. As a result, the frequency of the top-k calculation is used as input for the machine learning algorithm, obfuscating the actual data.

In summary, techniques are described herein for securing data used for a machine learning algorithm. The frequency or top-k values calculated over time of the respective network traffic feature data sets are used instead of the actual data or a set thereof (this can also be extended to any other data sets). Here, the frequency represents the actual data and thereby obfuscates potential sensitive information that should not be used within an oftentimes shared cloud machine learning application.