Digital Forensics - Project Report

Eugen Saraci

eugen.saraci@studenti.unipd.it

January 10, 2019

In the last years the large expansion qualche altro termine non sarebbe male of SSL/TLS has made it harder for attackers to collect clear text information through packet sniffing or, more in general, through network traffic analysis. The main reason for this is that SSL/TLS provides encrypts the traffic between two endpoints, which means that even though packets can still be easily captured, no useful information can be inferred from the packet's content without having the encryption keys.[1]

The authors of [1] and [2] showed that by training a machine learning algorithm with encrypted traffic data, one could correctly classify the user actions performed on the most common Android applications such as Facebook, Gmail, or Twitter, which could easily lead, through a correlation attack, to the full deanonimization of fake, privacy preserving identities.

In this work I try to reproduce the results achieved in [1] and [2] by implementing the classification model as described in the papers.

1 Introduction and Background Knowledge

devo scrivere qualcosa anche qui

1.1 Actions and Flows

Follows some useful and necessary terminology to better understand how the whole framework works.

Action and Action Label

An action is simply the action performed by a user while using one of th aforementioned Android apps. Examples of actions are: clicking on a profile page,

[1] It is worth mentioning that the endpoints of the communication (i.e. source and destination IP addresses) are transmitted in clear text for routing purposes; by performing a DNS lookup of the addresses and attacker could easily infer what site a user is visiting.

tweeting a message, sending an email etc. Please note that "clicking on a profile page" is what I refer to as the *action label*, in many cases I use the words "action" or "action flow" to refer to the set of flows that represent that action. spiegare meglio sta roba

Flows

When a user performs an action some encrypted packets are exchanged with the destination server. A flow consists of the sequence of the byte sizes of the exchanged packets. If the packet is going from the user's phone to the server it is said to be *outgoing*; if the packet is coming from the server to the user's phone it is said to be *incoming* and it is marked with a "-" sign before the integer number representing its size.[2] An example of a 5 packet flow is: `[-12, 80, 90, -111, 30]`. Please note that a single action performed by the user usually generates multiple flows of different dimensions, by that follows that an action actually consists of multiple flows. The techniques used by the authors to determine which flows belong to which action, the ordering of the packets, the packet capturing system, the packet filtering system, and the statistical analysis on the flows will not be treated in this report since the starting point for this work comes when the dataset is already constructed.

1.2 Notation

- A: action, it is supposed to represent a sequence of flows;
- a: action label;
- F: a flow, it is supposed to represent a sequence of packets;
- p: a single packet, it is an integer number representing the size in bytes of that packet.

  Please note that all of the above can be subscripted by indexes; a subscripted element means that it is the i-th element of a sequence, e.g. F_i is the i-th flow of a sequence of flows (possibily an action A)

  By this follows that $A = [F_0, \ldots, F_n]$ and $F = [p_0, \ldots, p_m]$. Note that n and m are possibly (and probably) different for each flow F and for each action A, even for two actions A_i, A_j where $a_i == a_j$.

1.3 Dynamic Time Warping

Dynamic Time Warping or **DTW**, is an algorithm used to measure similarity between two time series even if they are of different lengths and/or have repetitions or deletions in them. If we view every single flow F as a time series of packets $p_i, \ldots p_m$, we can use **DTW** to measure how similar two flows are. The reason we are interested in this will become clear later.

[2] The "-" sign is just notation, a packet cannot have a negative size

1.4 Machine Learning

Ideal goal: we want to develop a machine learning algorithm that outputs the *action label* based on the *action's flows*.

Given the fact that each action generates mutiple flows of different lengths, we know that standard supervised learning approaches are hard to apply. To see why standard approaches would not work we need to think about the structure of the input and output spaces. Our output space i.e. what we want to predict would be the *action label*, while our input space i.e. the predictors would be the flows generated by that action. One way to represent flows as features would be to have a feature for each flow generated by that action, and the values of the feature would be the sequence of byte sizes of that flow. Because of the different flow lengths we would immediately see that each row could possibly have a different number of feature; adding missing *dummy* features to shorter rows to equalize the lentgths would not help since the main problem of this approach is that we are artificially defining features with no real justification. In other words, we have no reason to associate the first flow of an action with the first flow of another action by marking them as the first feature of a sample.

1.5 Notation

2 Evaluation

2.1 Experimental Setup

2.2 Experimental Results

References

[1] Mauro Conti, Luigi V. Mancini, Riccardo Spolaor, and Nino Vincenzo Verde. 2015. Can't You Hear Me Knocking: Identification of User Actions on Android Apps via Traffic Analysis. In Proceedings of the 5th ACM Conference on Data and Application Security and Privacy (CODASPY '15). ACM, New York, NY, USA, 297-304. DOI: https://doi.org/10.1145/2699026.2699119

[2] Conti, M., Mancini, L. V., Spolaor, R., & Verde, N. V. (2016). Analyzing android encrypted network traffic to identify user actions. IEEE Transactions on Information Forensics and Security, 11(1), 114-125.