© 2015 IJSRSET | Volume 1 | Issue 2 | Print ISSN : 2395-1990 | Online ISSN : 2394-4099
Themed Section: Engineering and Technology

Preventing Side-Channel leaks in Web Traffic using SVSD

Akshaya A. K[1], Pavithra P[2], Anitha Sahaya Mercy P[3]
Dhanalakshmi College of Engineering, Kancheepuram District, Tamilnadu, India

ABSTRACT

The Encrypted traffic of many popular Web Applications may disclose highly sensitive data. Unique patterns may be identified in the encrypted traffic and an Eavesdropper can identify the actual application data. These are side-channel attacks. The existing solutions like random padding and packet rounding incur high cost. Hence we present a formal PPTP algorithm to efficiently reduce overhead cost incurred and also prevent side-channel attacks. This algorithm maps the similarity between the PPDP and PPTP problems and provides solution for the former.

Keywords: PPDP- Privacy Preserving Data Publishing, PPTP – Privacy Preserving Traffic Padding, Side-Channel Attacks, Web Application.

I. INTRODUCTION

The Web-based applications are more popular than the desktop applications. These Web applications pose security and privacy threats as the untrusted internet have essentially become an integral component of such applications for facilitating interaction between the users and the application. Recent studies have shown that the encrypted traffic of these Web applications may have highly sensitive data and may lead to security and privacy breaches i.e. by searching specifically for unique patterns exhibited in the packets' sizes and/or timing, an eavesdropper can identify the original sensitive data. These side-channel attacks (shown in Figure 1.0) are pervasive and fundamental to many of the Web applications.

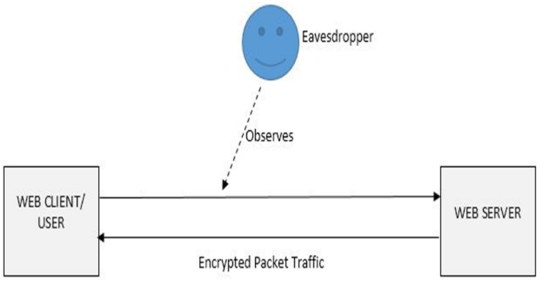


Figure 1: side-channel attacks

The obvious solution for preventing side channel attack is to pad packets such that the each packet size will no longer map to a unique input. The straightforward solutions include random padding and rounding packet size. These methods though effective incur a high prohibitive overhead.

The conflicting goals for preventing the side channel leaks are: First, the difference in packet size needs to be reduced sufficiently so that the packets may not be distinguished. Second, the overhead for achieving such privacy protection should be minimized. Thus the objective now is to find a solution that can provide sufficient privacy as well as reduce padding cost.

In this paper, we find the similarity between two problems PPDP and PPTP. PPDP is well studied and there are several solutions to solve this problem. Hence PPDP is mapped with PPTP problem and the same solution is applied to the former.

II. METHODS AND MATERIAL

PRESERVING PRIVACY IN WEB TRAFFIC

A. PPTP Problem

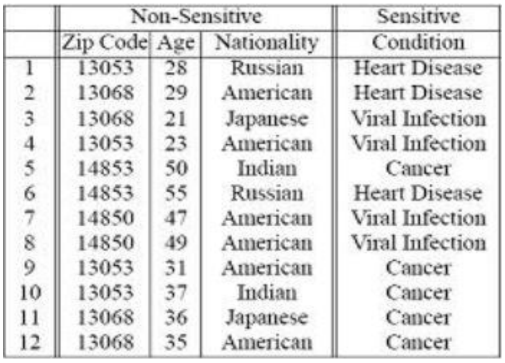
The Web application contains of sensitive data that gets encrypted into packets. Observations can be made on these packets and then the original data can be identified. The main issue to be solved deals with two perspectives: the interaction between the users and servers and the observation made by eavesdroppers. A single user keystroke may affect the other in terms of packet size. Thus multiple observations by eavesdropper can be combined and privacy and security of data can be violated.

B. PPDP Example

The privacy preserving data publishing deals with displaying of data so that highly sensitive data are not disclosed. The confidential data is retained or the data that can uniquely identify a particular entity is disclosed.

Each data is given a quasi-identifier and when an adversary re-identifies an individual entity using this quasi-identifier; it is called as a linking attack. The main goal of PPDP is prevent such linking attacks.

Table 1: Sensitive and non-Sensitive detail of Patient



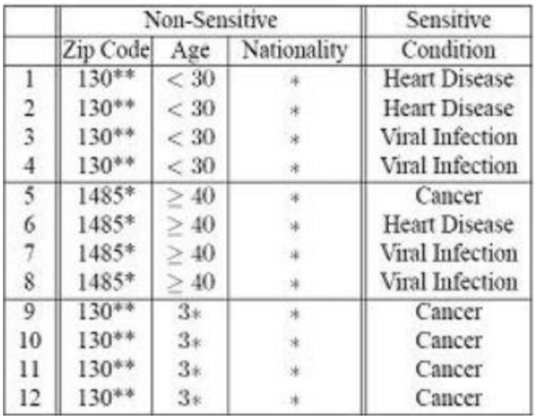
	Non-Sensitive			Sensitive
	Zip Code	Age	Nationality	Condition
1	13053	28	Russian	Heart Disease
2	13068	29	American	Heart Disease
3	13068	21	Japanese	Viral Infection
4	13053	23	American	Viral Infection
5	14853	50	Indian	Cancer
6	14853	55	Russian	Heart Disease
7	14850	47	American	Viral Infection
8	14850	49	American	Viral Infection
9	13053	31	American	Cancer
10	13053	37	Indian	Cancer
11	13068	36	Japanese	Cancer
12	13068	35	American	Cancer

Consider the above data Table 1.1 of patient details. The Zip Code, Age and Nationality are Non-Sensitive data which are easily disclosed. Using this data the sensitive information could be obtained. For Example, if the Age is known the condition can also be identified (Age is 31, the condition is cancer). This is the PPDP problem.

C. PPDP Solution

The k-anonymity model was devised to address this challenge. The table is divided into anonymized groups and then the quasi-identifier is generalized. Thus any quasi identifier wouldn't uniquely identify an entity instead it would point to a set of entities and so the linking attack fails. Hence the above table is generalized and data is made anonymous.

Table 2: Sensitive and non-Sensitive (not fully disclosed) detail of Patient



	Non-Sensitive			Sensitive
	Zip Code	Age	Nationality	Condition
1	130**	< 30	*	Heart Disease
2	130**	< 30	*	Heart Disease
3	130**	< 30	*	Viral Infection
4	130**	< 30	*	Viral Infection
5	1485*	≥ 40	*	Cancer
6	1485*	≥ 40	*	Heart Disease
7	1485*	≥ 40	*	Viral Infection
8	1485*	≥ 40	*	Viral Infection
9	130**	3*	*	Cancer
10	130**	3*	*	Cancer
11	130**	3*	*	Cancer
12	130**	3*	*	Cancer

Even the Non-Sensitive information is not fully disclosed as shown in Table 1.2. (When the age is known to be 31, there are multiple records of patients whose age are above 30). Thus k-anonymity model serves as a solution for the PPDP model.

MAPPING PPTP TO PPDP

The basic idea of solutions to solve PPDP like k-anonymity can be applied to solve the PPTP problems too. Consider a Web application, an action *a*, is the atomic user input unit that triggers traffic. This can be a keystroke or a mouse click. For example, Let the user input be "IT". Each keystroke "I" and "T" are taken as *a*. whenever a keystroke is sent, traffic is triggered (packet is sent). Let *s*, be the size of the directional packet triggered by *a*.

We now establish mapping between the PPDP and PPTP problems by regarding s as the quasi-identifier, the user input is the sensitive value and the padding groups as anonymized groups. Thus privacy can be achieved and side channel leaks can be prevented and the padding cost id also sufficiently reduced. Both the problems have a similar goal of achieving an optional tradeoff between privacy (preventing linking attack in PPDP or side channel attacks in PPTP) and utility (maximizing data utility in PPDP or minimizing padding cost in PPTP).

NO ENCRYPTION

Generally it is considered that in comparison with encryption, k-anonymity may not provide strong protection. But as we are dealing with cases where encryption is already broken by side-channel attacks, so it is not a favorable option. In theory, the k value could be made significantly larger to provide enough confidentiality, although a large k would satisfy users' privacy requirements for most practical applications.

Most of the Web applications are publicly accessible and consequently an eavesdropper can unavoidably learn about the list of all possible inputs, we believe perfect confidentiality is not always achievable. Padding packets to be k-anonymous does not leak information to adversaries about which packet is sensitive. This is due to the fact that the adversaries can see only the padded packets and not the original ones.

THE PPTP SOLUTION

A. Padding Method

We have already seen that the observations can be done on the padding method and side channel attack might happen. In order to choose a padding method now, privacy protection should be achieved by satisfying the k-anonymity property and minimizing the padding cost.

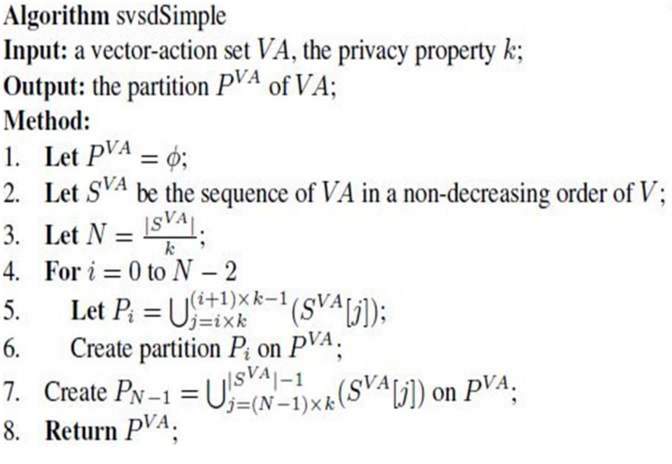
The generalization technique will partition the user inputs into various groups and then break the linkage among the actions in the same group. The padding can increase only the size of each packet but cannot decrease or replace it with a range of values in normal generalization. Thus a new padding method is considered.

Ceiling Padding basically means that after partitioning a vector-action set into groups, we pad each flow in the padding group to be the maximum size of the flow in that group.

B. SVSD Algorithm

We consider a simplified system where there is only a single flow in each flow-vector of the vector set. It simply means that each input a triggers only a single packet s.

The main intention of presenting the SVSD algorithm is to show that, when applying k-anonymity to PPTP problems, an algorithm may sometimes be devised in a very straightforward way, and yet achieve a dramatic reduction in costs when compared to existing approaches (as shown in the next section). The SVSD algorithm shown below basically attempts to minimize the cardinality of padding groups in the SVSD case. Note that when the cardinality of vector-action set is less than the privacy property k, there is no solution to satisfy the privacy property. In such cases, our algorithms will simply exit, which will not be explicitly shown in each algorithm hereafter.



```
Algorithm svsdSimple
Input: a vector-action set $VA$, the privacy property $k$;
Output: the partition $P^{VA}$ of $VA$;
Method:
Let $P^{VA} = \phi$;
Let $S^{VA}$ be the sequence of $VA$ in a non-decreasing order of $V$;
Let $N = \frac{|S^{VA}|}{k}$;
For $i = 0$ to $N - 2$
    Let $P_i = \bigcup_{j=i\times k}^{(i+1)\times k-1}(S^{VA}[j])$;
    Create partition $P_i$ on $P^{VA}$;
Create $P_{N-1} = \bigcup_{j=(N-1)\times k}^{|S^{VA}|-1}(S^{VA}[j])$ on $P^{VA}$;
Return $P^{VA}$;
```

More specifically, SVSD first sorts each single flow in the flow-vector into a non-decreasing order of the flows, and then selects k pairs of (flow-vector, action) each time in that order to form a padding group. This is repeated until the number of pairs is less than k. The remainder of pairs is simply appended to the last padding group.

The computational complexity is O(nlogn) where n = jVAj, since step 2 costs O(nlogn) time and each (flow-vector, action) pair is considered once for the remaining steps.

C. Other Complex Algorithms

The complex cases like where one single input triggers multiple packets. For all these complex cases, SVSD algorithm will not be helpful. Hence we devise similar algorithms like SVMD and MVMD.

These algorithms are bound to satisfy the k-anonymity factor. Many algorithms can be featured so that they satisfy l-diversity factor which is also a solution of the PPDP problem.

III. RESULTS AND DISCUSSION

The above mentioned algorithms were implemented in a web application. The application is basically a search engine. The keywords are to be entered and the result page is displayed. The algorithm provides different encrypted value for the same keyword at different instances. This was observed at all instances of the same keyword. Each time these keywords are mapped to a different encrypted value. The extravagance was applied to store files. These files were also split and encrypted using the algorithm thereby increasing more privacy and also prevents side channel leaks.

IV. CONCLUSION

The Web based applications are more popular than the desktop applications. The Web application are influenced by the untrusted internet. This arises many security issues. In this paper, we have arrived at a solution for security and privacy by mapping the relationship between PPDP and PPTP. We have proposed a quantifying model to provide traffic padding solutions. The computational overhead has been proven to be less.

V. REFERENCES

[1] Wen Ming Liu, Lingyu Wang, Kui Ren, Pengsu Cheng, and Mourad Debbabi. k-Indistinguishable Traffic Padding inWeb Applications

[2] A. Askarov, D. Zhang, and A.C. Myers. Predictive black-box mitigation of timing channels. In CCS, pages 297–307, 2010.

[3] D. Asonov and R. Agrawal. Keyboard acoustic emanations. Security and Privacy, IEEE Symposium on, page 3, 2004.

[4] A. Aviram, S. Hu, B. Ford, and R. Gummadi. Determining timing channels in compute clouds. In CCSW '10, pages 103–108, 2010.

[5] M. Backes, G. Doychev, M. D¨urmuth, and B. K¨opf. Speaker recognition in encrypted voice streams. In ESORICS '10, pages 508–523, 2010.

[6] Michael Backes, Goran Doychev, and Boris K¨opf. Preventing Side-Channel Leaks in Web Traffic: A Formal Approach. In NDSS'13, 2013.

[7] I. Bilogrevic, M. Jadliwala, K. Kalkan, J.-P. Hubaux, and I. Aad. Privacy in mobile computing for location-sharing-based services. In PETS, pages 77–96, 2011.

[8] N. Cao, Z. Yang, C. Wang, K. Ren, and W. Lou. Privacy-preserving query over encrypted graph-structured data in cloud computing. In ICDCS'11, pages 393–402, 2011.

[9] C. Castelluccia, E. De Cristofaro, and D. Perito. Private information disclosure from web searches. In PETS'10, pages 38–55, 2010.

[10] Peter Chapman and David Evans. Automated black-box detection of side-channel vulnerabilities in web applications. In CCS'11, pages 263–274, 2011.

[11] Wen Ming Liu, Lingyu Wang, Pengsu Cheng, Kui Ren, Shunzhi Zhu and Mourad Debbabi. PPTP – Privacy preserving traffic padding in web based applications

[12] G. Aggarwal, T. Feder, K. Kenthapadi, R. Motwani, D. Thomas, and Zhu. Anonymizing tables. In ICDT '05, pages 246–258, 2005.