Obfs4 Traffic Identification Based on Multiple-feature Fusion

Di Liang
School of Computer
Beijing Jiaotong University
China, Beijing
E-mail: 17120472@bjtu.edu.cn

Yongzhong He
School of Computer
Beijing Jiaotong University
China, Beijing
E-mail: yzhhe@bjtu.edu.cn

***Abstract*—Tor is currently the most used anonymous browser. Users can communicate anonymously on the Internet through Tor and some criminals can use Tor for illegal and criminal activities. In order to deal with congestion, Tor introduced a bridge mechanism to replace the previous ingress node. Obfs4 is one of the most important bridges used by Tor. It uses an improved elliptic curve encryption algorithm and random padding to hide message information, the anti-detection ability is extremely strong. In order to effectively identify Obfs4 traffic, this paper proposes a Obfs4 identification method based on Multiple-feature fusion. Through research on Obfs4 protocol, data packet structure, node publishing strategy, and node distribution, this paper proposes many ways to obtain multiple features, including randomness characteristics, sequential characteristics, handshake packet length characteristics, and communication packet statistics characteristics. In addition, this paper proposes a machine learning algorithm based on a weighted Gaussian kernel function, which can modify the weight of different features to change the degree of influence of different features on the final classification result. Finally, the weight of each feature and the parameters used by the algorithm are determined through experiments. The accuracy of the algorithm is 93.82%, the recall is 99.00%, and the accuracy is 94.34%, which is much better than other algorithms mentioned in this paper. At the same time, this paper proves that there are some loopholes in Obfs4's anonymity mechanism, and its effective fingerprint can be obtained from the information it leaks to carry out attacks.**

Keywords—network security, anonymous communication, Tor, Obfs4

I. INTRODUCTION

Anonymous communication technology [1] refers to the use of encryption, obfuscation, and other methods to hide the communication relationship and communication content between the two parties during the communication process, which can effectively protect the privacy[2] of both communication parties and prevent the communication content from being intercepted by a third party, leading to information leakage. Tor is the second generation of onion routing [3] and is one of the most widely used anonymous communication tools. When using Tor for webpage access, Tor first accesses its directory server, obtains the entry node information, and forwards the user request to the entry node; the entry node forwards it to the intermediate node after receiving the user request; the intermediate node forwards it to the exit node, the exit node communicates with the destination address.

Because more and more researches have been done on Tor, its nodes have been banned a lot, in order to ensure the normal operation of Tor, Tor introduced the bridge mechanism. That is, a bridge node with a more complex encryption algorithm is used instead of the entry node, and the bridge node connects with the client and forwards the data to the intermediate node。Because the bridge node uses a completely different communication protocol from the ingress node, the previous detection method of entry nodes is no longer applicable to the bridge node.

Obfs4 is one of the most widely used bridges in Tor. Obfs4 has a threat model built on Obfs2, which can resist DPI; because of the existence of the authentication mechanism, it can also resist impersonation user attacks; for some non-plaintext fingerprint information Obfs4 uses random padding and time delay mechanisms to hide this information, such as message length and time interval. Therefore, Obfs4 is very anonymous.

Obfs4 obfuscation protocol consists of two parts, namely the client located in the local machine and the server located at the Obfs4 bridge node (entry node). It uses a two-way identity authentication mechanism[4] and can effectively resist man-in-the-middle attacks, as shown in the figure 1.

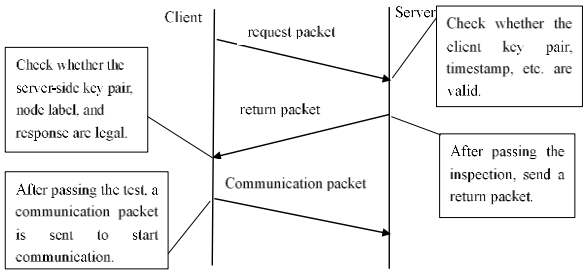

Fig. 1 The process of Obfs4 communication

At present, the difficulties faced by Obfs4 traffic identification are: (1) the number of nodes is large, the distribution is wide, and it is difficult to obtain all of them; (2) the ability to resist static detection is strong, and the data packet does not have static plaintext features [5]; (3) There are a lot of similar traffic in the real environment, and it is difficult to obtain obfs4 effective dynamic characteristics.

In order to solve the above problems, this paper proposes a multi-feature-based traffic detection method to obtain the randomness characteristics, the sequential characteristics, the

characteristics of the handshake packet length, and the statistical characteristics of the communication packets. A large number of features are transformed into feature vectors using a representation learning method [6], and then a classification algorithm is used to judge the traffic to be detected to achieve better recognition results.

II. Extraction and Representation of Multiple Features of Obfs4

The multi-feature-based Obfs4 traffic recognition method proposed in this paper includes three stages, which are feature extraction, feature representation, and feature use. In the feature extraction stage, we divided Obfs4's data stream features into two parts: static features and dynamic features to extract valid information. At the feature representation stage, we used representation learning methods and applied Word2Vec technology, the features are represented as feature vectors. In the third stage, we combine the feature vectors, calculate the feature weights, and use the SVM algorithm based on the weighted Gaussian kernel function [7] for model training and prediction.

A. Packet static characteristics

The static characteristics of the data packet include the length and randomness characteristics of the handshake packet. Through the introduction of the Obfs4 communication principle in Chapter 1, during the handshake phase of the Obfs4 protocol, the payload part of the handshake packet uses a random filling method to ensure anonymity[8]. The collected Obfs4 data packet is shown in Figure 2. The content of the data packet does not have any valid plaintext information. And the payload length of the randomly filled data packet is between 149 bytes and 8259 bytes. Therefore, we check the randomness and length of the handshake packet of the data stream to be detected to determine whether it has the static characteristics of Obfs4 packets. The detection of randomness uses a single-bit frequency detection algorithm, that is, by detecting 0 in the packet load Compared with the number of 1, to determine whether it has randomness, the detection algorithm is as follows:

(1) The data packet load is converted into a binary sequence β of length n.

(2) Convert 0 in β to -1, $X_i = 2\beta_i - 1$ to get a new sequence.

(3) Summing up the accumulation, $S = \sum_{i=1}^{n} X_i$

(4) Calculate the statistical value, $V = \frac{|S|}{\sqrt{n}}$

(5) Calculate $P - value = erfc\left(\frac{V}{\sqrt{n}}\right)$, if $P - value \geq \alpha$ (α is a preset value), the packet is considered to have passed randomness detection and has randomness.

```
d1 e4 26 a8 82 ed 2c d5  54 b2 78 14 2a 7e 3d 0a   ··&···,· T·x·*~=·
58 51 44 9d c6 1a 0e 3a  2d 13 7d 96 67 dc bd 1d   XQD····: -·}·g···
67 a5 fd 36 5a 3d ed 17  ec a1 74 29 57 3e 16 43   g··6Z=·· ··t)W>·C
fa 44 e7 ea 30 38 c6 a3  4c 7c 1d 7b 19 90 c0 51   ·D··08·· L|·{···Q
38 48 38 99 97 ec 99 21  e4 e6 9d c9 06 d3 18 da   8H8····! ········
8c fa 85 b1 b4 7b 85 6b  89 ba 6d 46 e3 bf 0d 5a   ·····{·k ··mF···Z
c2 5d fd 35 2d 41 93 12  b6 a1 4e 9e 42 5f e2 ac   ·]·5-A·· ··N·B_··
ec 9d 57 6a e4 67 19 f0  b1 19 24 9e 29 da 55 74   ··Wj·g·· ··$·)·Ut
5a d4 9b 2f df 8b 88 26  d4 e0 a3 72 0e 63 8b 83   Z··/···& ···r·c··
1a ef e6 78 af d6 7f c7  17 4d 2d 79 18 73 cd 5e   ···x···· ·M-y·s·^
0d 07 0c 7c b9 31 20 cf  6c d9 fb 5d 43 4e 77 fe   ···|·1 · l··]CNw·
64 8e 22 75 e5 ef 7c f0  61 f8 56 e9 76 c9 95 b2   d·"u··|· a·V·v···
2b 8c 48 0d 4d 81 21 a7  01 19 c6 b6 b6 96 e9 6d   +·H·M·!· ·······m
a1 57 7f 20 92 fc 75 b6  62 c0 94 45 f1 27 7b 1b   ·W· ··u· b··E·'{·
41 50 11 86 8b eb ce a0  28 8e 13 b4 20 eb 24 a2   AP····· (··· ·$·
89 45 a6 cd 97 13 42 71  35 eb f3 f3 f8 14 73 ad   ·E····Bq 5·····s·
39 d4 cb 07 34 99 16 48  36 a7 81 0e d1 e3 d3 99   9···4··H 6·······
1d 55 d0 be a6 51 44 fa  39 72 14 bd 5c 57 55 db   ·U···QD· 9r··\WU·
f7 53 ce e0 9b e4 cd 8f  a5 df 35 4b 85 47 7f 91   ·S····· ··5K·G··
8a bb 92 ef 48 07 2d d3  96 03 6a 95 4d 3c c8 82   ····H·-· ··j·M<··
f9 8f 27 5b 7d 25 25 d0  1c c9 b3 20 a6 33 53 9a   ··'[}%%· ··· ·3S·
ff 7e 92 03 af df b6 28  8b 7c 87 8f 0c f4 6d b3   ·~·····( ·|····m·
fe 5d 09 cb be 4c 9e 58  18 11 b4 c6 0a 06 d1 c3   ·]···L·X ········
52 12 62 80 b3 07 a5 c7  bf f7 41 93 6a e7 94 d1   R·b····· ··A·j···
8d f7 8b 00 2f 9b 4f 7e  11 dc 01 fc ca 71 1e 7c   ····/·O~ ····q·|
76 50 37 41 28 fc b2 af  35 f3 b0 0b 35 db c8 fb   vP7A(··· 5···5···
```

Fig. 2 Obfs4 packets with randomness

B. Packet dynamic characteristics

The dynamic characteristics of the data packet include the sequential characteristics of the handshake data packet, and the characteristics of the communication data packet. The sequential characteristics of the handshake data packet refer to that according to the obfs4 communication protocol, during the handshake process, the data packets follow strict one-to-one and one-back rules. There is no continuous sending of multiple complete handshake packets to one party, and many communication protocols are not sequential[9].. The characteristics of the communicated packets are that after the handshake is completed, the direction and length of several consecutive communication packets are obtained, and the statistical characteristics such as variance and entropy are calculated [10], which can describe the distribution of the packets during the data stream communication process.

III. SVM Classification Algorithm Based on Weighted Gaussian Kernel Function

A. Calculation of feature weights

Different features have different impacts on classification, so the weight of each feature needs to be calculated [11]. This article uses the method of calculating mutual information to weight features based on the dependence of traffic attributes (whether they are Obfs4 streams) and each feature. The formula for calculating mutual information MI is (1):

$$MI(t) = P(C)\log\frac{P(t,C)}{P(t)P(C)} \tag{1}$$

Among them, $P(C)$ is the probability that the data in the entire training set is Obfs4 stream, $P(t)$ refers to the probability of the feature term t appearing, and $P(t,C)$ is the ratio of the data of obfs4 flow with characteristic term t to the total data of training set. Through this formula, the mutual information of each feature can be calculated, and

according to the ratio of the mutual information of different features, the weight of each feature can be calculated.

B. Weighted Gaussian kernel function

The SVM algorithm maps linearly inseparable low-latitude data to high-dimensional space by referring to the kernel function, divides different data in a high-dimensional space with a hyperplane, and maps the high-level space to the low-dimensional plane according to the previous mapping relationship. Gaussian kernel function is a kind of kernel function widely used in SVM algorithm. It measures the similarity between different samples and divides them. The kernel function formula is (2):

$$k(x,y)=\exp\left\{\frac{-\|x-y\|^2}{2\delta^2}\right\} \quad (2)$$

$\|x-y\|^2$ is the squared Euclidean distance between the two eigenvectors [12]. The value of the kernel function decreases with distance, and approaches zero infinitely. δ is a free parameter used to control the radial range of action. An equivalent and simpler definition is to set a new parameter $\gamma=\frac{1}{2\delta^2}$, The original expression becomes (3):

$$k(x,y)=\exp\left(-\gamma\|x-y\|^2\right) \quad (3)$$

If there are two points $x\left(x_1,x_{2,}x_3,\ldots,x_n\right)$ and $y\left(y_1,y_{2,}y_3,\ldots,y_n\right)$ in the n-dimensional vector space, the Gaussian kernel function is expanded as (4):

$$k(x,y)=\exp\left(-\gamma\sum_{i=1}^{n}\left(x_i-y_i\right)^2\right) \quad (4)$$

It can be seen from this formula that when calculating the distance between two points of the Gaussian kernel function, the values of each dimension must participate in the calculation, and the proportion of the values of each dimension is the same. However, in actual situations, different features have different effects on the classification results of the samples. The features with large weights have a large impact on distance during calculation, otherwise they have a small impact. The i-dimensional feature weight is w_i, and the Gaussian kernel function formula based on the weight is (5):

$$k(x,y)=\exp\left(-\gamma\sum_{i=1}^{n}w_i\left(x_i-y_i\right)^2\right) \quad (5)$$

IV. Experimental evaluation

In this chapter, we collected 2076 Obfs4 data streams and 11345 non-Obfs4 data streams from three clients in a real traffic environment as a data set for experiments. The recall, precision, and accuracy are used as the evaluation criteria for the experimental results, thereby proving the validity of the identification method.

A. Calculation of feature weights

In SVM algorithm [13], the objective function is (6):

$$\min_{\omega,b,\zeta}\frac{1}{2}\omega^T\omega+C\sum_{i=1}^{l}\zeta_i \quad (6)$$

$$y_i\left(\omega^T\varnothing\left(x_i\right)+b\right)\geq 1-\zeta_i,\zeta_i\geq 0,i=1,\ldots,l$$

C represents the degree of punishment for classification errors under the condition of linear inseparability. The larger C is, the lower the classifier's tolerance for outliers of classification errors is. When the C value is too large, overfitting will occur [14]. The smaller C is, the less importance the classifier attaches to the points where errors occur. If C is too small, the classification performance will be poor. Therefore, the choice of C cannot be too large or too small. The value of C is generally $2^{10}-2^{12}$ when the algorithm works well.

Bandwidth gamma is an important indicator of the Gaussian kernel function, which can dominate the range of the support vector machine. The relationship between gamma and δ is (7):

$$gamma\propto\frac{1}{\delta^2} \quad (7)$$

The value of gamma is inversely proportional to the value of δ. If the value of gamma is too large and the value of δ is too small, overtraining may occur [15], that is, the accuracy during the training phase is high but the accuracy during the prediction phase is low; if the value of gamma is too small and the value of δ is too large will cause the Gaussian kernel function to fall quickly, and the accuracy in the training and prediction phases is low. The value of gamma generally works well from 20 to 32.

TAB. I EXPERIMENTAL RESULTS OF SVM ALGORITHM BASED ON WEIGHTED GAUSSIAN KERNEL FUNCTION

Gamma	C	Precision	Recall	Accuracy
20	1024	89.11%	95.18%	90.17%
	2048	89.12%	95.89%	91.11%
	4096	89.13%	96.88%	93.22%
24	1024	89.97%	95.14%	92.10%
	2048	89.98%	97.07%	94.71%
	4096	90.19%	97.85%	94.80%
28	1024	91.77%	98.32%	94.01%
	2048	91.39%	97.92%	94.01%
	4096	90.02%	97.11%	94.61%
32	1024	90.17%	98.12%	94.92%
	2048	91.44%	98.30%	95.02%
	4096	91.56%	97.91%	94.78%

For the SVM algorithm without using a weighted Gaussian kernel function, when the penalty factor C is 28 and the bandwidth gamma is 1024, the algorithm works best.For the SVM classification algorithm based on the weighted Gaussian kernel function, the experimental results are shown in Table I. When the penalty factor C is 2048 and the bandwidth gamma is 32, the algorithm works best, with an accuracy of 91.44%, a recall of 98.30%, and an accuracy of 95.02%.

By comparing the two SVM algorithms, the SVM algorithm based on the weighted Gaussian kernel function has significantly improved each index under different penalties and bandwidth conditions, as shown in Table II.

TAB. II IMPROVEMENT EFFECT OF WEIGHTED SVM ALGORITHM

Gamma	C	Improved Precision	Improved Recall	Improved Accuracy
20	1024	5.22%	6.13%	4.83%
	2048	4.98%	7.82%	5.89%
	4096	3.89%	6.76%	6.71%
24	1024	6.24%	5.65%	6.39%
	2048	6.44%	5.89%	8.25%
	4096	6.05%	5.93%	7.67%
28	1024	5.98%	5.37%	2.91%
	2048	6.23%	6.73%	5.55%
	4096	5.55%	5.33%	6.44%
32	1024	4.30%	6.04%	5.20%
	2048	5.59%	5.31%	4.30%
	4096	6.22%	5.00%	3.80%

Through experiments, the superiority of the SVM algorithm based on the weighted Gaussian kernel function compared with the unimproved algorithm is proved, and when the penalty effect C is determined to be 2048 and the bandwidth gamma is 32, the algorithm performs best.

B. Algorithm effect comparison

There are various machine learning algorithms, Commonly used traffic recognition algorithms include KNN [16], logistic regression [17], decision tree [18], etc. Using the same data set and the above algorithms for training and prediction, experimental results are obtained. As shown in Table III

TAB. III EXPERIMENTAL RESULTS OF VARIOUS MACHINE LEARNING ALGORITHMS

Algorithm	Precision	Recall	Accuracy
Weighted SVM	93.82%	99.00%	94.34%
KNN	86.12%	91.51%	86.69%
Logistic regression	61.60%	75.30%	64.88%
Decision tree	87.11%	91.93%	89.17%

It can be seen that the SVM algorithm based on the weighted Gaussian function has different improvements compared to each algorithm. Especially for the logistic regression algorithm, the accuracy rate has increased by 32.2163%, the recall rate has increased by 23.6954%, and the accuracy rate has increased by 29.4660 %, The comparison results are shown in Table IV. After this experiment, it is proved that the effect of the SVM algorithm based on the weighting function is better than other learning algorithms.

TAB. IV RESULTS OF COMPARING TO VARIOUS MACHINE LEARNING ALGORITHMS

Algorithm	Improved Precision	Improved Recall	Improved Accuracy
KNN	7.70%	7.49%	7.65%
Logistic regression	32.22%	23.70%	29.47%
Decision tree	6.71%	7.07%	5.17%

V. CONCLUSION

This paper studies the Tor's Obfs4 bridge and elaborates the working principle, communication process, protocol content and data packet structure of the Obfs4 bridge in detail. By analyzing and summarizing the various aspects of Obfs4, a variety of features including dynamic and static features are obtained, and they are vectorized. At the same time, in order to meet the actual situation, different features have different degrees of impact on the classification results. A SVM classification algorithm based on a weighted Gaussian kernel function is proposed. The effect of different dimensions on the Euclidean distance is modified to change its impact on the classification results. Finally, by comparing with the classification results of general SVM algorithms and other machine learning algorithms, the superiority of the proposed algorithm is proved. This paper also proves that there are some defects in the anonymity of Obfs4, some fingerprints can be used by attackers, and there is still room for further improvement of Obfs4's anonymity.

REFERENCES

[1] Goel, Sharad,Robson, Mark,Polte, Milo, Sirer,Emin. Herbivore: A Scalable and Efficient Protocol for Anonymous Communication[R]. New York:Cornell University, 2003.

[2] M.Shao, Wh.Hu, Sc.Zhu, Gh.Cao, S.Krishnamurth, T.L.Porta. Cross-layer Enhanced Source Location Privacy in Sensor Networks[A]. Min Shao. 2009 6th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks[C]. Rome: IEEE, 2009. 1-9.

[3] S.Kim, J.Han, J.Ha, T.Kim, D.Han. SGX-Tor: A Secure and Practical Tor Anonymity Network With SGX Enclaves[J]. IEEE/ACM Transactions on Networking, 2018, 26(5): 2174 - 2187.

[4] Ss.Zhao, Wh.Hu. Improvement on OTP authentication and a possession-based authentication framework[J]. International Journal of Multimedia Intelligence and Security, 2018, 3(2): 187-203.

[5] Yz.He, Lp.Hu, R.Gao. Detection of Tor Traffic Hiding Under Obfs4 Protocol Based on Two-Level Filtering[A]. Yongzhong He. 2019 2nd International Conference on Data Intelligence and Security (ICDIS)[C]. South Padre Island: IEEE, 2019. 195-200.

[6] Y.Song, Q.Li, H.Huang, D.Feng. Low Dimensional Representation of Fisher Vectors for Microscopy Image Classification[J]. IEEE Transactions on Medical Imaging, 2017, 36 (8): 1636-1649.

[7] Sp.Zhong, D.Chen, Qf.Xu, Ts.Chen. Optimizing the Gaussian kernel function with the formulated kernel target alignment criterion for two-class pattern classification[J]. Pattern Recognition, 2013, 46(7): 2045-2054.

[8] Z.Ling, Jz.Luo, W.Yu, M.Yang, Xw.Fu. Tor Bridge Discovery: Extensive Analysis and Large-scale Empirical Evaluation[J]. IEEE Transactions on Parallel and Distributed Systems, 2013, 26(7): 1887-1899.

[9] L.S.Huang, A.Rice, E.Ellingsen, C.Jackson. Analyzing Forged SSL Certificates in the Wild[A]. Lin Shung Huang. 2014 IEEE Symposium on Security and Privacy[C]. San Jose: IEEE, 2014. 83-97.

[10] N.Duffield, C.Lund, M.Thorup. Properties and prediction of flow statistics from sampled packet streams[A]. Nick Duffield. Proceedings of the 2nd ACM SIGCOMM Workshop on Internet measurment[C]. New York: Association for Computing Machinery, 2002. 159-171.

[11] Wh.Huang, Z.Ma, Xf.Dai, Md.Xu, Y.Gao. Fuzzy Clustering with Feature Weight Preferences for Load Balancing in Cloud[J]. International Journal of Software Engineering and Knowledge Engineering, 2018, 28 (5): 593-617.

[12] L.Liu, Jq.Zhou. A route-like demand location problem based on squared-Euclidean distance[A]. Lu Liu. 2016 International Conference on Logistics, Informatics and Service Sciences (LISS)[C]. Sydney: IEEE, 2016. 1-4.

[13] I.Sarafis, C.Diou, T.Tsikrika. Weighted SVM from clickthrough data for image retrieval[A]. Ioannis Sarafis. 2014 IEEE International Conference on Image Processing (ICIP)[C]. Paris: IEEE, 2014. 3013-3017.

[14] Jf.Nong. The Design of RBF Neural Networks and experimentation for solving overfitting problem[A]. Jifu Nong. Proceedings of 2011 International Conference on Electronics and Optoelectronics[C]. Dalian: IEEE, 2011. 75-78.

[15] P.Henniges, E.Granger, R.Sabourin. Factors of overtraining with fuzzy ARTMAP neural networks[A]. P.Henniges. 2005 IEEE International Joint Conference on Neural Networks[C]. Montreal: IEEE, 2005. 1075-1080.

[16] Okfalisa, I.Gazalba, Mustakim, Nurul Gayatri Indah Reza. Comparative analysis of k-nearest neighbor and modified k-nearest neighbor algorithm for data classification[A]. Okfalisa. 2017 2nd International conferences on Information Technology, Information Systems and Electrical Engineering[C]. Yogyakarta: IEEE, 2018. 294-298.

[17] J.Kim, J.Lee, C.Lee, E.Park, J.Kim, H.Kim, J.Lee. Optimal Feature Selection for Pedestrian Detection Based on Logistic Regression Analysis[A]. Jonghee Kim. 2013 IEEE International Conference on Systems, Man, and Cybernetics[C]. Manchester: IEEE, 2013. 239-242.

[18] R.Liu, Xl.Qian, S.Mao, Sz.Zhu. Research on anti-money laundering based on core decision tree algorithm[A]. Rui Liu. 2011 Chinese Control and Decision Conference[C]. Mianyang: IEEE, 2011. 4322-4325.