DDoS Detection in SDN using ML*

Juyeon Nam
Stonybrook University
juyeon.nam@stonybrook.edu

***Abstract*—This project introduces a DDoS attack detection in SDN. Several case studies for effective SVM learning are presented to compare detection rates of attack and improve them. There is not enough specific research in security related to SDN. Therefore, this project tried to find effective detection methods by increasing the number of hosts or the rate of attacks. Through this, in the future, SDN technology, which has a lot of advantages but has been reluctant due to security problems, is expected to be safer and more used.**

I. Introduction

SDN is emerging as a technology with great prospects. Software Defined Network (SDN) means a technology of network control that virtualizes and abstracts resources through the software. It separates the control layer and the data layer from the networking hardware. It is centrally designed to simplify network management. Because SDN is software-based, it is more flexible than before when it was based on hardware. For these reasons, the SDN market is rapidly growing. However, SDN has vulnerable security problems. It is because the control plane and the data plane are divided. In particular, DDoS attacks on SDN become issues. A Distributed Denial of Service (DDoS) attack is a malicious attempt to disrupt normal traffic by flooding Internet traffic excessively to a server. DDoS attacks can overload SDN's centralized controller, causing the entire network to collapse. Therefore, the project will focus on how to identify and prevent DDoS attacks early in order to continue using SDN.

II. Initial Settings

The initial setting for the project is as follows.

- Support Vector Machine(SVM): A supervised learning model that learns data for classification regression. In this project, this ML is used to train the model to predict DDoS attacks and to determine them early.
- Ubuntu: Install to make a working environment on the Virtual Machine. (22.04.1 LTS)
- Mininet: A network simulator for configuring virtual environments to test SDN environments. Use for virtual network topologies.
- OpenFlow: Standard communication specification for data planes and control planes. The SDN for this project will use OpenFlow.
- Ryu-controller: SDN Controller for telling the protocol to the switch in the data plane.
- Python 3.8.16: Python is used because Ryu is a Python-based controller. Python 3.8.16 version must be installed to prevent a version conflict with the Ryu manager.
- Skikit-learn: A tool used to learn software using Python. This supports SVM.
- Hping3: Packet generator for testing security. This automatically generates traffic

III. Original Model

First, both normal and attack traffic are collected to train the model through SVM. The topology of the original model is a switch with 10 hosts connected to it. The initial setting is `APP_TYPE` to 0 for data collection, `Interval` to 3, and `TEST_TIME` to 400 seconds.

```
['12/15/2022, 09:54:06', '11', '0', '1.0']
['12/15/2022, 09:54:09', '6', '4', '1.0']
['12/15/2022, 09:54:12', '12', '6', '1.0']
['12/15/2022, 09:54:15', '18', '9', '1.0']
['12/15/2022, 09:54:18', '12', '3', '1.0']
['12/15/2022, 09:54:21', '12', '1', '1.0']
['12/15/2022, 09:54:24', '12', '1', '1.0']
```

Now, normal traffic will be collected which is no DDoS attacks. To do this, `TEST_TYPE` in controller.py is changed to 0 and the `TEST_TYPE` in the Topo.py is changed to normal. After that, the `TEST_TYPE` will be changed to 1 and attack to collect DDoS attack status data.

```
['12/15/2022, 10:04:14', '259', '259', '0.007722007722007722']
['12/15/2022, 10:04:17', '284', '284', '0.003683241252302026']
['12/15/2022, 10:04:21', '281', '281', '0.0024271844660194173']
['12/15/2022, 10:04:23', '182', '182', '0.0019880715705765406']
['12/15/2022, 10:04:26', '186', '186', '0.0016778523489932886']
['12/15/2022, 10:04:29', '189', '189', '0.0014482259232440260']
```

Data collection is now complete.

The following are the settings for DDoS detection. For attack detection, set `APP_TYPE` to 1 and `TEST_TIME` to 100. Moreover, sets the time to block a port when an attack is detected for 120 seconds. After 120 seconds, the port is unblocked and reactivated on the next attack.

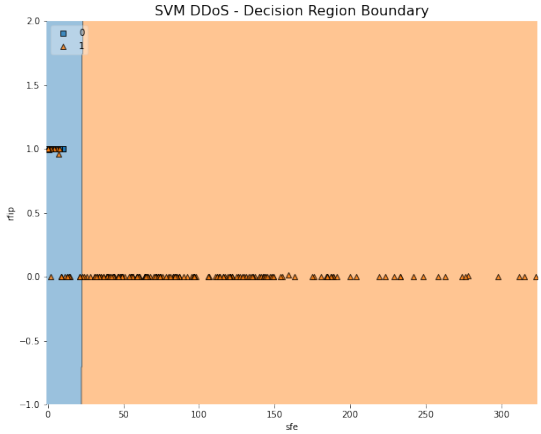
First, attacks will be detected in normal traffic. Therefore, change the `TEST_TYPE` to 0 and normal, respectively.

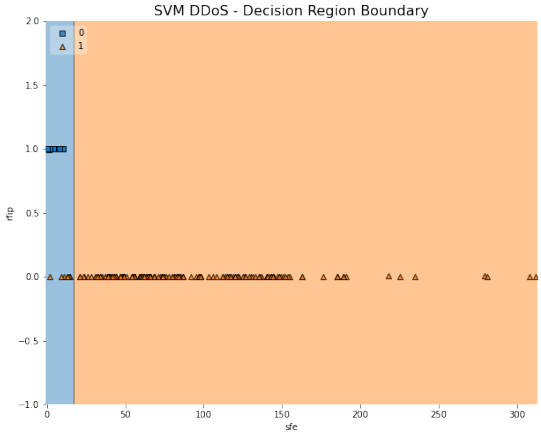
```
SVM input data [1, 0, 1.0] prediction result  ['0']
It's Normal Traffic
SVM input data [16, 9, 1.0] prediction result  ['0']
It's Normal Traffic
SVM input data [16, 8, 1.0] prediction result  ['0']
It's Normal Traffic
SVM input data [12, 4, 1.0] prediction result  ['0']
```

Then performs the process of detecting and blocking attacks. The priority setting changes only the `TEST_TYPE` to attack. The `TEST_TYPE` does not need to change to 1 because this machine is still detecting an attack.

```
attack detected from port  1
Block the port  1
attack detected from port  1
Block the port  1
SVM input data [1, 0, 0.03076923076923077] prediction result  ['1']
Attack Traffic detected
Mitigation Started
SVM input data [0, 0, 0.03076923076923077] prediction result  ['1']
Attack Traffic detected
```

The accuracy of this model is 90.10600706713781 and the false alarm rate is 0.0



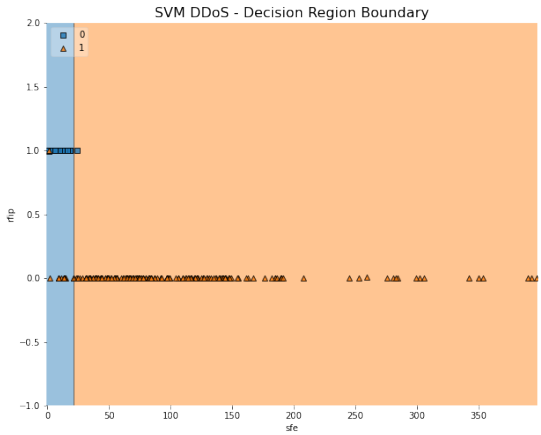


IV. Experiment

The variables is going to be changed for better detection.

A. Case Study 1: Increase the number of host

This case, the topology will changed. The number of hosts connected to the switch will be increased from 10 to 40. To do this, a host should be added by setting the host name and mac address. Then, it should be linked with switch. It can be found from the difference between these two models whether the model can detect traffic well when the host is increased. The accuracy of this model was 100 and the false alarm rate was 0.0.



From the graph, it can be seen that some normal traffic is perceived as attack traffic as the number of hosts increases.

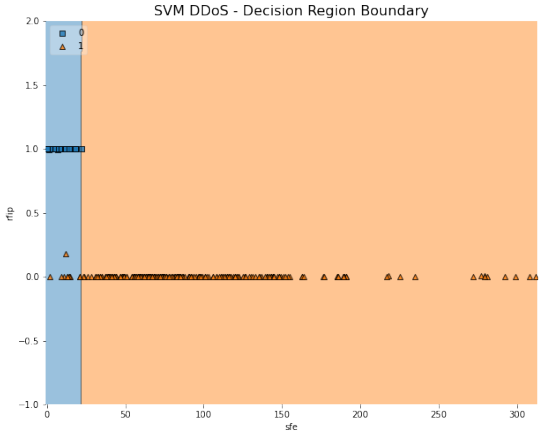
B. Case Study 2: Doubled the kind of attack

This time, the number of attacks will become doubled. The reason is to compare the difference in the ability to detect as the attack increases. Go to the attack.sh file and double the number of hping commands, which represented the attack and set the kind of attack. Now, an attack will start on a randomly selected ip. At this point, the attack takes place on Host 1 and 500 to 1000 packets are forwarded. The accuracy and false alarm rates of this model are 99.5 and 0.0, respectively.

There is no significant difference in comparing both graphs. It can be because it is visual data, not accurate figures. One thing is that the normal traffic category contains less attack traffic. The reason is predicted as ML learns with more attack traffic data.

C. Case Study 3: Increase the number of host + Double the kind of attack

Finally, these above two conditions will be combined to one model. The detection efficiency will be checked when both the number of hosts and attacks increase. The accuracy of this model is 99.65277777777779 and the false Alarm rate is 0.0.

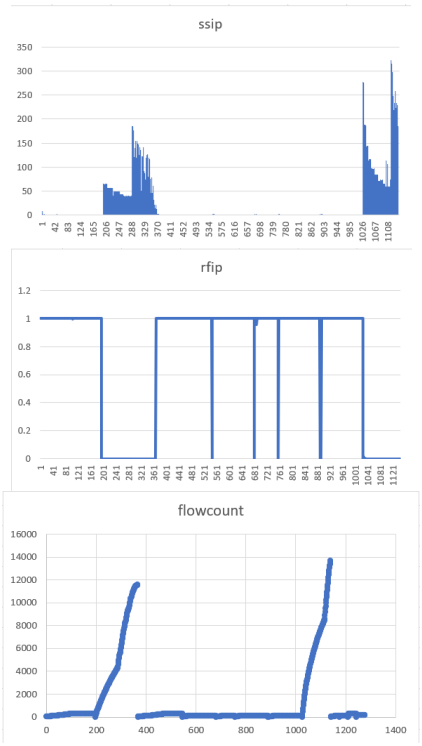


This model will help to demonstrate the result of above two data because it combines the above two conditions, so that it will have both of the above features. First of all, it is observed that not all normal traffic was included in the normal category. In addition, less attack traffic exists in the normal category compared to the original graph.

V. Analysis

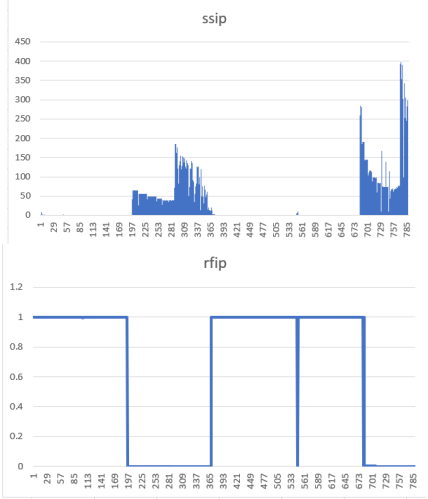
The value of sfe, ssip, rfip, flowcount will be compared. First of all, sfe is the speed of flow entries, ssip means the speed of source IP, and rfip is the ratio of flow pair. These values will be visualized to a graph and compared to the original graph.

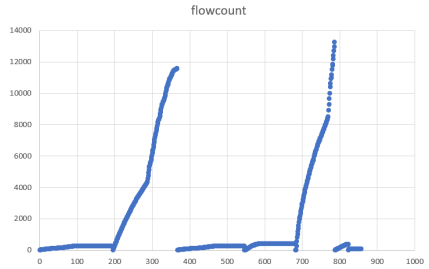
A. Original Data



These are ssip, rfip, and flowcount graphs of the original data. It can be predicted when the server is being attacked through the ssip graph. If it has high values, it means that there is a large number of traffic, which is a DDoS attack can be suspected. Similarly, from the rfip graph, it can be found when the attack traffic came in. It is because flowpairs goes to 0 during there is attack traffic and to 1 during there is only normal traffic. In the Flowcount graph, it is found how much attack data comes in and when the port is blocked. When the number of data is approximately 200 and 1000, the attack traffic starts to come in and slowly increases, and then the port is blocked before 200 more data come in.

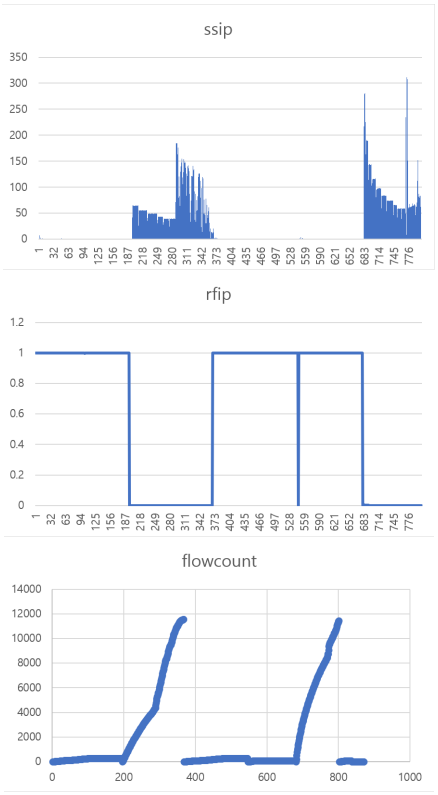
B. Increase the number of host





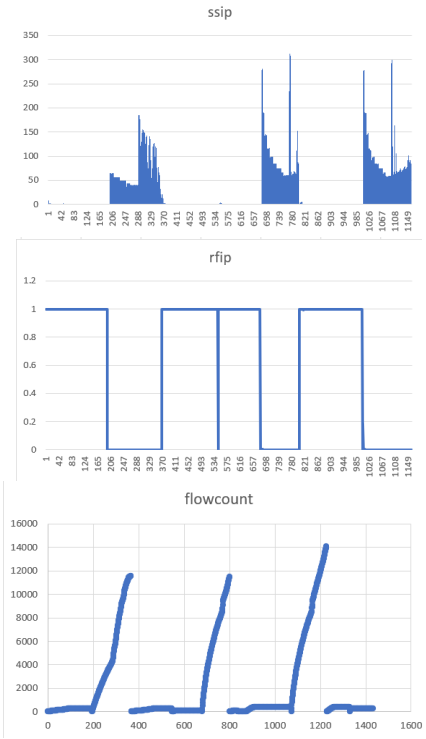
When comparing the graph of the original model with this graph, there is only a minor difference between the rfip and flowcount graphs. However, in the ssip graph, the increased value is observed in y-axis, which is speed of source IP address. It can be interpreted that the number of traffic is increased. From these data, it is realized that the increasing number of hosts makes the traffic coming in at once increase. However, it can also be found that it does not affect the time it blocks, which is the time this model detects an attack.

C. Doubled the attack



This time, the original model will be compared to the model which is doubled in its attack. From the original graph, it is observed that the sharp rise in the original graphs was two. Therefore, it can be predicted as the sharp rise in this graph became four, which is its doubled. It is because when the attack gets doubled, more attacks should be shown. However, in the experiment, four sharp rise does not found. It might be because the attempt, which makes to increase attack in attack.sh file, does not work. It predicted that the attempt was to increase the kind of attack rather than increase the number of attacks.

D. Increase the number of host + Double attack



This model is both the number of hosts and the number of attacks increased. The degree of increase is equal to the above conditions, 40 and 2 times, respectively. Therefore, 3 rises were detected in all graphs. However, the detected speed of flow entries and source ip were similar to the original graph. This allowed to demonstrate a double-attack model, but not a 40-host model.

Assessment

In this project, the original model and three different models are implemented and the original one and the others are compared on each detection capability. Some meaningful data is come out, and it can be analyzed based on them. However, contrary to the expectation that large differences in the data could be observed and analyzed, they were not found. This is presumed to be due to the difference in the accuracy of each graph. Therefore, if matching the accuracy of each of the graphs and then comparing, a more accurate comparison can be made. However, it was difficult to get the same accuracy for the graphs that were constantly changing. Furthermore, it could be better if a bigger difference in the data is set. However, this would have made it difficult for the SVM model to learn, making it difficult to keep accuracy. Therefore, we should try to find a way to set up meaningful data while maintaining the accuracy of the SVM model. If we find it, we expect to get a more accurate analysis.

Conclusion

SDN is often used for increased network flexibility and ease of management, however it has the disadvantage of weak security. Therefore, the project discussed when security becomes more vulnerable and how much malicious traffic can be prevented. The scope of SDN technology is expected to increase further in the future. However, there has not been much security research related to SDN. Therefore, more research will need to be done on SDNs security in the future. Especially, more measures to respond to DDoS attacks will have to be studied.

Meterial

The original code for the project used below code

https://github.com/vishalsingh45/SDN-DDOS-Detection-and-Mitigation-using-ML-and-Statistical-methods.git

References

[1] Juniper network, What is SDN?, https://www.juniper.net/us/en/research-topics/what-is-sdn.html

[2] Techtarget, What is Software-defined networking, https://www.techtarget.com/searchnetworking/definition/software-defined-networking-SDN

[3] Dong Li et al ”Using SVM to Detect DDoS Attack in SDN Network” 2018 IOP Conf. Ser.: Mater. Sci. Eng. 466 012003 pp.1-8

[4] Lubna E, Roberto P ”Future DoS and DDoS attacks in Software Defined Networks: A survey of existing solutions and research challenges Generation Computer Systems” 2021 Vol 122 pp. 149-171, ISSN 0167-739X

[5] L. Yang and H. Zhao, ”DDoS Attack Identification and Defense Using SDN Based on Machine Learning Method,” 2018 15th International Symposium on Pervasive Systems, Algorithms and Networks (I-SPAN), 2018, pp. 174-178, doi: 10.1109/I-SPAN.2018.00036.