

1. Introduction

1.1 Preamble

Many devices on the internet are still vulnerable to brute force attacks because they can be accessed with their default passwords. SSH (Secure Socket Shell) and FTP (File Transfer Protocol) are network protocols that are commonly targeted by brute force attacks. To detect and mitigate these attacks, we implemented an Intrusion Detection System (IDS) using Zeek.

When a brute force attack is detected, the system alerts the administrator and blocks packets coming from the attacker's IP address, making the server inaccessible to the attacker. This helps to protect against unauthorized access to sensitive information and maintain the security of the network.

1.2 Motivation

- Many devices connected to the internet are still vulnerable to attacks because they can be accessed using their default passwords.
- An Intrusion Detection System (IDS) is a system that monitors network traffic for suspicious activity and issues alerts when such activity is detected.
- Brute force attacks, also known as high-level attacks, are a common and challenging type of attack on computer systems.
- In brute force attacks, attackers attempt to guess passwords or other sensitive information by trying a large number of possibilities, which can exhaust hardware resources and make data vulnerable.
- It is important to detect and prevent brute force attacks in order to protect against unauthorized access and maintain the security of the network.

1.3 Objectives

- To design a network activity monitoring component for observing the traffic flow of the network topology.
- To develop an algorithm for detecting suspicious behavior in the network.
- To design a mitigation module for SSH brute-force attack and FTP brute-force attack in the network.
- To analyze the performance of the proposed mitigation module.

1.4 Literature Survey

In order to detect and mitigate SSH brute force attacks [10] on an edge device, the following steps can be taken: Generate a pen testing for brute force attack using the hydra command with a list of usernames and passwords, and the SSH protocol. Recognize an SSH brute force attack event by monitoring resource use. Remove or disable any functionality that could potentially be exploited by the attacker. Implement a shunting function to redirect or block traffic from the attacker's IP address. Use the NetControl framework in the Zeek IDS to drop packets from the attacker's IP address, or use the "catch and release" function to temporarily block the IP address to save rule space [1].

The authors [2] used the Long Short-Term Memory (LSTM) deep learning method on network traffic data to detect network-level brute-force attacks on the SSH and FTP protocols [13]. Dictionary-based brute-force attacks, which involve using prepared password lists, are a common form of attack worldwide. To evaluate the effectiveness of the LSTM method, the authors used the CICIDS2017 dataset, which contains both common attacks and benign data, and was generated using the Patator brute-force tool in a controlled setting to include the most recent attack methods. The LSTM network was able to accurately detect SSH and FTP brute-force attacks with a high accuracy of 99.88% and low false positive and false negative rates[6].

The author [3] used a technique on Linux to receive credentials from a standard SSHD in order to improve security measures against brute-force attacks. Some of the techniques used include disabling root logins through SSH and enabling PAM, saving all user names that do not have direct SSH access [14], and handling each login attempt's password individually [12]. This prevents curious users without superuser authority from viewing or tampering with sensitive information. To evaluate the effectiveness of these measures, the author developed two Key Performance Indicators (KPIs): UNAME, which counts all failed SSH login attempts for existing user names in the last minute, and PWDPAT, which collects all failed login attempts in the last minute whose user name-hashed password pattern-pairs match existing ones. The paper also discusses the increased risk of SSH brute force attacks that use the same patterns as the targeted passwords and proposes a Condition Monitoring System (CMS) to monitor and analyze the risk of SSH brute force attacks in real time.

To classify data and detect network attacks, basic Decision Tree models with two independent variables can be trained and tested. Information systems connected to the internet are vulnerable to attacks, and machine learning (ML) algorithms can be used to automatically detect malicious traffic by learning to recognize patterns that indicate attacks. Secure Shell (SSH), brute force, and FTP brute force attacks [9] are common forms of large data attacks [7]. Data can be split into a 70:30 ratio for training and testing, with a hold-out validation set used during training iterations. Decision Tree, Naive-Bayes, Logistic Regression, KNN, and SVM models can be used to detect these attacks, and the best performing model is chosen. Decision Tree models are the most frequently used approach

for detecting these attacks.[4]

Software-Defined Networking (SDN) allows for more flexible and dynamic network setup and management by separating the control and data planes and enabling centralized management [8]. To prevent SSH brute force attacks, the authors propose a deep learning-based intrusion detection and prevention system (CDI-IDPS) for use in SDN networks [15]. In the proposed DL-IDPS, the packet length in the SDN switch is recorded as a sequence and analyzed using four deep learning models: Multilayer Perception (MLP), Convolutional Neural Network (CNN), Long Short-Term Memory (LSTM), and Stacked Auto-Encoder (SAE). The findings indicate that the MLP-based DL-IDPS has the highest accuracy, with the ability to prevent attacks with around 99 to 100% accuracy [5].

1.5 Problem Definition

To design a mechanism for detection and mitigation of SSH and FTP Brute-Force attacks using Intrusion Detection System.

2. PROPOSED SYSTEM

2.1 Proposed System

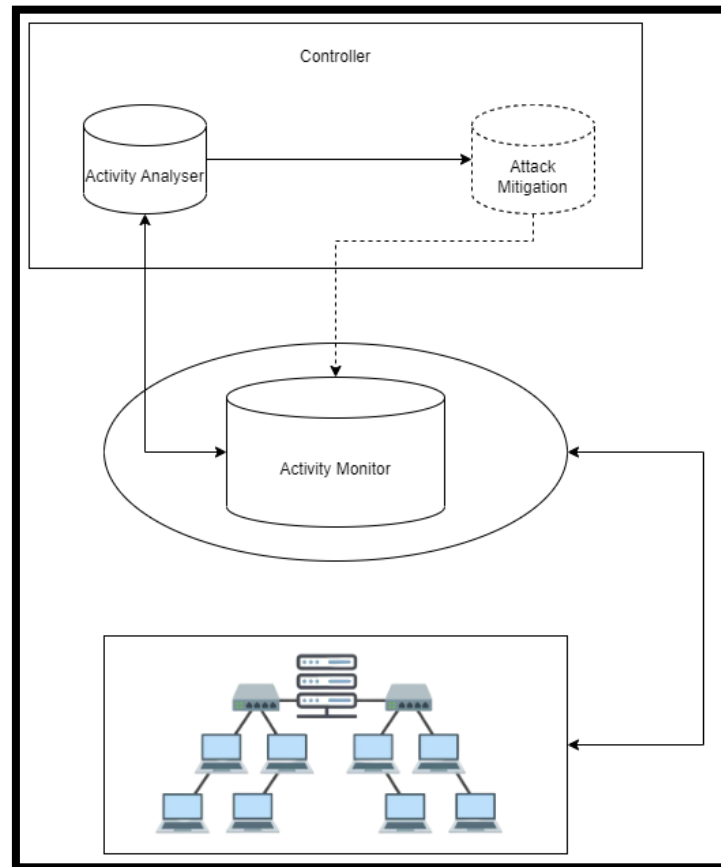


Figure 1: Proposed system for detection and mitigation of network attacks using IDS

The proposed system has the following components:

- **IDS:** An Intrusion Detection System (IDS) is a system that monitors network traffic for suspicious activity and sends alerts when such activity is detected. It is a software application that checks a network or system for harmful activity or violations of established policies. An IDS can be used to identify and prevent potential security threats, such as brute force attacks, malware infections, and unauthorized access. It is an important tool for maintaining the security and integrity of a network or system.
- **Activity Monitor:** The activity monitor is a component of an Intrusion Detection System (IDS) that is responsible for monitoring the traffic flow on a network. It gathers information about the traffic and looks for any abnormal behavior, such as attempts to access unauthorized resources or perform unauthorized actions. The activity monitor is an important part of the IDS because it helps to detect potential

security threats and ensure the security and integrity of the network.

- **Activity Analyzer:** The activity monitor in an Intrusion Detection System (IDS) is responsible for detecting any suspicious behavior on the network. It uses the information it gathers about the traffic flow to identify patterns that may indicate abnormal or malicious activity. If such activity is detected, the IDS can issue an alert or take other actions to protect the network and its resources. By monitoring the network for unusual activity, the IDS helps to ensure the security and integrity of the network.
- **Attack Mitigation:** If a brute force attack is found, measures to mitigate the attacks are carried out.

2.2 Description of Target Users

Target users in our proposed model are the users (the legitimate one's) who wants to get the service of interest and the attackers, who try brute forcing on the network.

2.3 Advantages of Proposed System

- Since the proposed model is divided into so many modules, each module performs well in the task assigned to it.
- Also, the proposed model runs on the server, no matter in which network the attack is being done, it is detected and mitigated without any problem.

2.4 Scope

- In the proposed system, the threshold counts for which the network should be classified as attacker traffic is set to be 5.
- If any brute force attack happens in less than 5 tries (of low possibility), then the attack goes undetected.

3. SOFTWARE REQUIREMENT SPECIFICATION

3.1 Overview of SRS

This section consists of the software requirement specifications of the project, which mainly consists of the requirement specifications and the functional requirements, the nonfunctional requirements and the use case where it shows how the user interacts based on different conditions whether it is a success scenario or any other scenario. Then comes the hardware requirements and the software requirements. In section 3.4 acceptance test plan is written where it consists of sample inputs and outputs.

3.2 Requirement Specifications

In this subsection the requirement specification of the proposed system is being discussed. Then the use case diagrams and the sequence diagrams are being developed into the functional requirements of the system.

3.2.1 Functional Requirements

- The system shall be able to detect the brute force attacks.
- The system shall be able to block the intruder.
- The system shall be able to classify between the legitimate traffic and attack traffic.
- The system shall be able alert the admin about the attack event.
- The system shall be able to drop further packets from the IP for which intrusion is detected in the network.

3.2.2 Use case Diagram

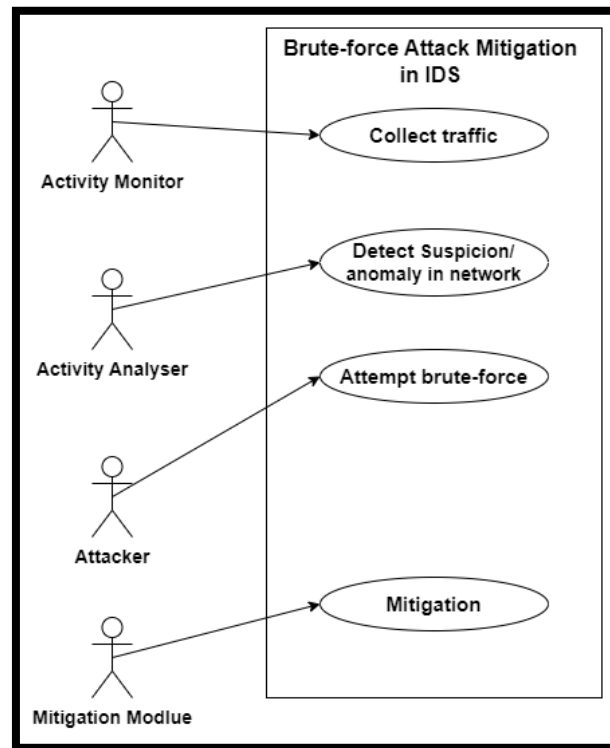


Figure 2: Use Case Diagram

3.2.3 Use Case descriptions using scenarios

Use case:	During brute force attack
Primary actor:	Intrusion Detection System
Goal:	To detect and mitigate the attack
Pre-condition:	Zeek IDS should be running
Post-condition:	User should be notified about the attack
Success scenario:	When the attack is mitigated successfully.
Exception scenario:	If attacker succeeds in getting access to the server within 5 tries.

Table 1: Use case description

3.2.4 Non-Functional Requirements

3.2.4.1. Performance Requirements

- Packet loss during the traffic capture should be less than 1%.
- The system should be capable of handling more than one traffic concurrently, with the capability to increase with larger demand.
- Delay in monitoring should be less than 150ms

3.3 Software and Hardware requirement specifications

Software requirements

- Linux OS
- Zeek Intrusion Detection System

Hardware requirements

- Memory: 8GB RAM
- Processor: Intel 3rd / 5th /7th gen processors / AMD Ryzen

4. System Design

4.1 Architecture

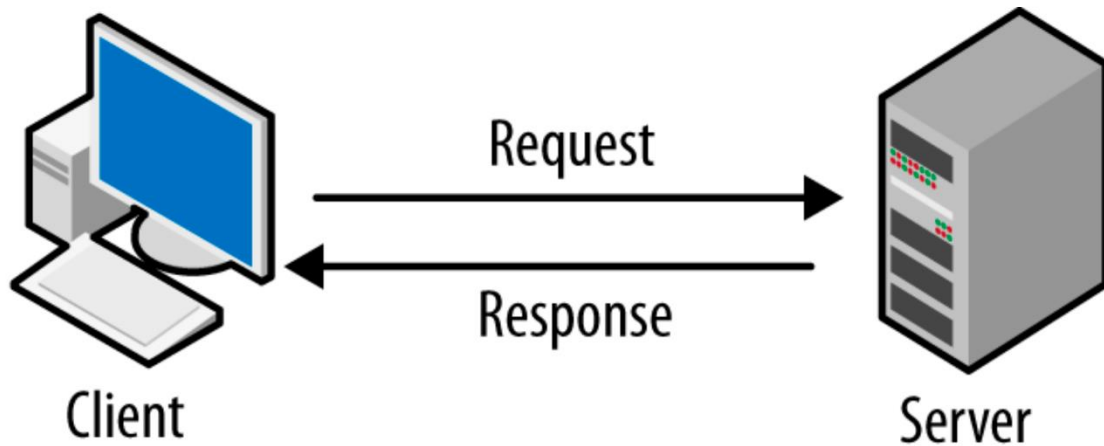


Figure 3: Client Server architecture design

- In the client-server architecture, a client is a device or program that sends requests to a server, and a server is a device or program that receives requests from a client and sends back responses.
- Secure Shell (SSH) is a network protocol that allows secure communication between a client and a server. It is commonly used to securely access servers, configure servers, and transfer files between servers and clients.
- File Transfer Protocol (FTP) is a network protocol that allows the transfer of files between computers on a network. FTP servers are used to store and manage files that can be accessed and transferred by FTP clients.
- A brute force attack is a type of cyber-attacks in which an attacker tries to guess the password or other authentication credentials of a system by repeatedly attempting to log in with different combinations of passwords and usernames. Brute force attacks can be used to attack FTP servers by attempting to guess the login credentials of valid users.
- To protect against brute force attacks, it is important to use strong, unique passwords and to enable additional security measures such as two-factor authentication. It is also a good idea to monitor login attempts and to block or rate-limit repeated failed login attempts to prevent an attacker from being able to continually try new combinations.

5. Implementation

5.1 Proposed Methodology

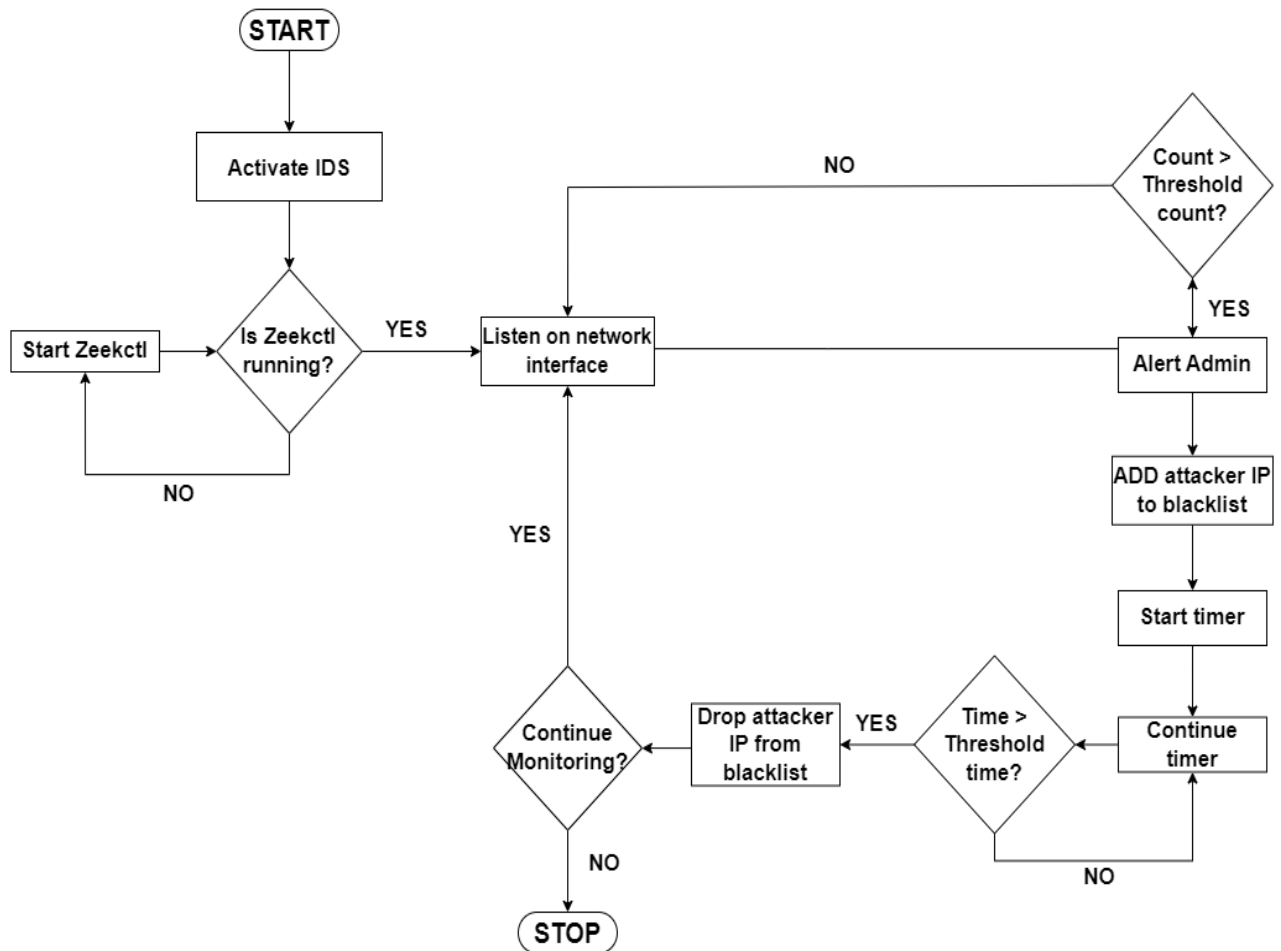


Figure 4: Flow diagram for network attack detection

- First activate Intrusion Detection System (IDS) on the server.
- Check if IDS is running (Zeekctl) else again try restarting it.
- If it is running then listen on network interface to capture all the incoming packets.
- For every connection request, keep failure counts and if the count is less than the threshold then accept connection request.
- If the failure count is greater than threshold then alert admin using the alert box and block IP address of the attacker.
- Set the blocked IP address in blacklist for the duration for which the user has set.
- Continue listening on network interface.

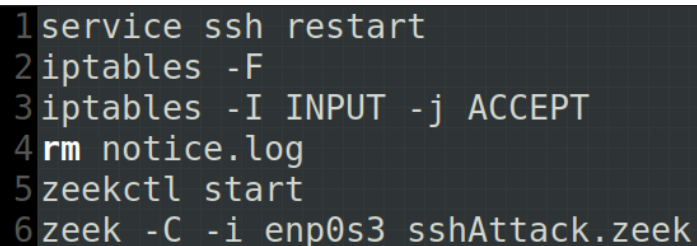
5.2 Description of Modules

Module 1: Activity Monitor:

- The main function of this module is to listen on the network interface and capture all the incoming network packets.
- Output: .pcap file that contains information of all the received packets.

Module 2: Activity Analyzer:

- The main function of this module is to process the captured packets using the selected zeek script. This generates log files based on type of the packet being processed.
- Input: Network packets captured by Activity monitor.
- Output: Different log files.



```
1 service ssh restart
2 iptables -F
3 iptables -I INPUT -j ACCEPT
4 rm notice.log
5 zeekctl start
6 zeek -C -i enp0s3 sshAttack.zeek
```

Figure 5: Script for detection of Attacks

Line wise explanation of the above script:

1. Restarts the SSH service or service of interest.
2. Flushes all the iptables rules (previously blocked IP addresses).
3. Adds a rule to accept all the incoming network packets that should be analysed.
4. Removes the previously generated notice log.
5. Starts the zeekctl (Activate the IDS)
6. Capture all the network packets that are received on the selected network interface (here, it is enp0s3) and send them to zeek script (here, it is sshAttack.zeek).

Module 3: Attack Mitigation Module:

- The main function of this module is to classify network as legitimate traffic or attack traffic. This module also takes mitigation measures like notifying user about on-going brute force attack, drop the attacker IP address, start time to again

accept the packets from attacker.

- Input: Log files generated by the activity analyzer.
- Output: Mitigation steps if any.

Demo (for SSH service):

- We first run “scan.sh” bash file (script for detection of attacks) in the terminal (as root user).

This starts capturing the network packets from the selected network interface.

- This keeps running in the background monitoring all the packets.

```
root@steve:/home/steve/Desktop/ZeekLogs/SSH# ./scan.sh
starting zeek ...
warning in ./sshAttack.zeek, lines 86-89: "when" statement referring to locals without an explicit [] capture is deprecated: SSH::cmd (when (SSH::res = Exec::run(SSH::cmd)) { print IP Dropped})
warning in ./sshAttack.zeek, lines 93-96: "when" statement referring to locals without an explicit [] capture is deprecated: SSH::cmd (when (SSH::res2 = Exec::run(SSH::cmd)) { print Notified to Admin})
warning in ./sshAttack.zeek, lines 103-106: "when" statement referring to locals without an explicit [] capture is deprecated: SSH::cmd (when (SSH::res3 = Exec::run(SSH::cmd)) { print Timer set for 60 min})
listening on enp0s3

netcontrol debug (Debug-All): init
```

Figure 6: Networking monitoring using IDS

- If any legitimate user tries to access service of interest, then he gets logged in to the service without any problem.

```
(v110@kali)~$ ssh steve@192.168.29.5
steve@192.168.29.5's password:
Welcome to Ubuntu 20.04.5 LTS (GNU/Linux 5.15.0-52-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

112 updates can be applied immediately.
75 of these updates are standard security updates.
To see these additional updates run: apt list --upgradable

New release '22.04.1 LTS' available.
Run 'do-release-upgrade' to upgrade to it.

Your Hardware Enablement Stack (HWE) is supported until April 2025.
You have new mail.
Last login: Sun Dec 11 15:08:55 2022 from 192.168.29.210
steve@steve:~$ ls
dead.letter  Desktop  Documents  Downloads  mininet  Music  oflops  oftest  openflow  Pictures  pox  Pu
steve@steve:~$
```

Figure 7: Legitimate user logging into SSH of server

- If attacker tries brute forcing on to the server, after 5 failed tries (set as threshold for failed login attempts), the server is notified with a notice box, IP of the attacker is dropped for certain time.

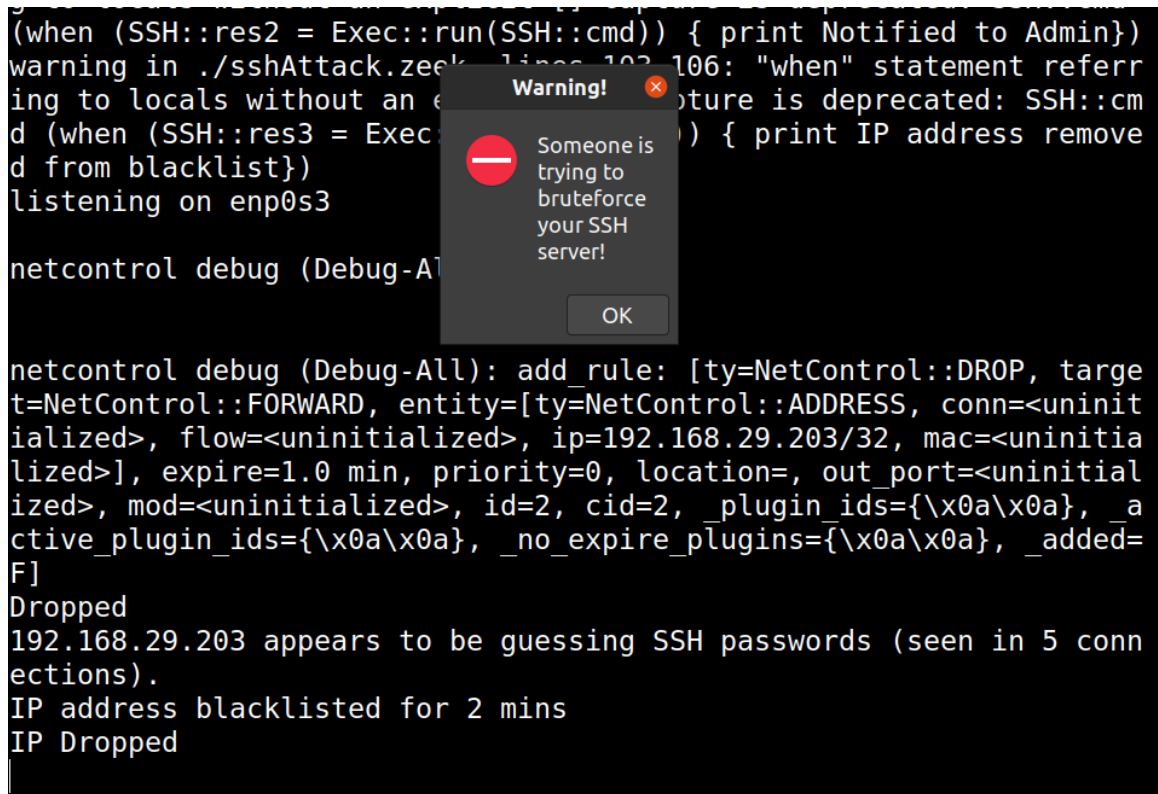


Figure 8: Notification of SSH attack event

- Iptables that contain all the network rules entries are shown below:

○ **Before attack:**

pkts	bytes	target	prot	opt	in	out	source	destination
976	120K	ACCEPT	all	--	*	*	0.0.0.0/0	0.0.0.0/0

Table 2: iptables contents before SSH attacks

○ **During attack:**

pkts	bytes	target	prot	opt	in	out	source	destination
291	21968	DROP	all	--	*	*	192.168.29.203	0.0.0.0/0
783	95647	ACCEPT	all	--	*	*	0.0.0.0/0	0.0.0.0/0

Table 3: iptables contents during SSH attacks (attacker IP is blocked)

- **After attack (after removing IP from blacklist).**

pkts	bytes	target	prot	opt	in	out	source	destination
1249	89514	ACCEPT	all	--	*	*	0.0.0.0/0	0.0.0.0/0

Table 4: iptables contents after SSH attacks (after certain time interval)

- Attacker will not be able to brute force into the server.

```

$ hydra -L uname.txt -P password.txt 192.168.29.5 ssh
Hydra v9.3 (c) 2022 by van Hauser/THC & David Maciejak - Please do not use in
binding, these *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2022-12-12 23:
[WARNING] Many SSH configurations limit the number of parallel tasks, it is r
[WARNING] Restorefile (you have 10 seconds to abort ... (use option -I to skip
e
[DATA] max 16 tasks per 1 server, overall 16 tasks, 80 login tries (l:10/p:8)
[DATA] attacking ssh://192.168.29.5:22/
[ERROR] ssh target does not support password auth
[STATUS] 53.00 tries/min, 53 tries in 00:01h, 37 to do in 00:01h, 6 active
^CThe session file ./hydra.restore was written. Type "hydra -R" to resume ses

```

Figure 5: Mitigation of SSH attack using IDS (unsuccessful attack)

Demo (for FTP service):

- We first run “scan.sh” bash file (script for detection of attacks) in the terminal (as root user).

This starts capturing the network packets from the selected network interface.

- This keeps running in the background monitoring all the packets.

```

root@steve:/home/steve/Desktop/ZeekLogs/FTP# ./scan.sh
rm: cannot remove 'notice.log': No such file or directory
starting zeek ...
warning in ./ftpAttack.zeek, lines 62-65: "when" statement referri
recated: Exec::cmd (when (Exec::res = Exec::run(Exec::cmd)) { prin
warning in ./ftpAttack.zeek, lines 69-72: "when" statement referri
recated: Exec::cmd (when (Exec::res2 = Exec::run(Exec::cmd)) { pri
warning in ./ftpAttack.zeek, lines 79-82: "when" statement referri
recated: Exec::cmd (when (Exec::res3 = Exec::run(Exec::cmd)) { pri
listening on enp0s3

```

Figure 10: Networking monitoring using IDS

- If any legitimate user tries to access service of interest, then he gets logged in to

the service without any problem.

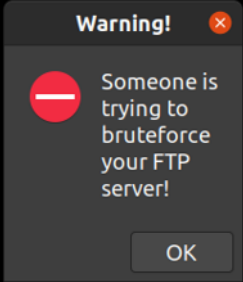
```
(vj10@kali)-[~]
$ ftp 192.168.29.5
Connected to 192.168.29.5.
220 (vsFTPd 3.0.3)
Name (192.168.29.5:vj10): ftpuser
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp>
```

Figure 11: Legitimate user logging into FTP of server

- If attacker tries brute forcing on to the server, after 5 failed tries (set as threshold for failed login attempts), the server is notified with a notice box, IP of the attacker is dropped for certain time.

```
root@steve:/home/steve/Desktop/ZeekLogs/FTP# ./scan.sh
starting zeek ...
warning in ./ftpAttack.zeek, lines 62-65: "when" statement referring
to locals without an explicit [] capture is deprecated: Exec::cmd
  (when (Exec::res = Exec::run(Exec::cmd)) { print IP Dropped})
warning in ./ftpAttack.zeek, lines 69-72: "when" statement referring
to locals without an explicit [] capture is deprecated: Exec::cmd
  (when (Exec::res2 = Exec::run(Exec::cmd)) { print Notified to Admin})
warning in ./ftpAttack.zeek, lines 79-82: "when" statement referring
to locals without an explicit [] capture is deprecated: Exec::cmd
  (when (Exec::res3 = Exec::run(Exec::cmd)) { print IP address removed from blacklist})
listening on enp0s3

192.168.29.203 appears to be guessing FTP passwords (seen in 5 connections).
IP address blacklisted for 30 mins
IP Dropped
```



A warning dialog box with a red circle and a white exclamation mark. The text inside reads: "Warning! Someone is trying to bruteforce your FTP server!". There is an "OK" button at the bottom right.

Figure 12: Notification of SSH attack event

- Iptables that contain all the network rules entries are shown below:

○ **Before attack:**

pkts	bytes	target	prot	opt	in	out	source	destination
2076	182K	ACCEPT	all	--	*	*	0.0.0.0/0	0.0.0.0/0

Table 6: iptables contents before FTP attacks

○ **During attack:**

pkts	bytes	target	prot	opt	in	out	source	destination
209	14246	DROP	all	--	*	*	192.168.29.203	0.0.0.0/0
783	95647	ACCEPT	all	--	*	*	0.0.0.0/0	0.0.0.0/0

Table 7: iptables contents during FTP attacks (attacker IP is blocked)

○ **After attack (after removing IP from blacklist).**

pkts	bytes	target	prot	opt	in	out	source	destination
1249	89514	ACCEPT	all	--	*	*	0.0.0.0/0	0.0.0.0/0

Table 8: iptables contents after SSH attacks (after certain time interval)

- Attacker will not be able to brute force into the server.

```
(vj10@kali)~$ hydra -L uname.txt -P password.txt 192.168.29.5 ftp
Hydra v9.3 (c) 2022 by van Hauser/THC & David Maciejak - Please do
not use in military or secret service organizations, or for illeg
al purposes (this is non-binding, these ** ignore laws and ethics
anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 202
2-12-21 20:27:08
[WARNING] Restorefile (you have 10 seconds to abort... (use option
-I to skip waiting)) from a previous session found, to prevent ov
erwriting, ./hydra.restore
[DATA] max 16 tasks per 1 server, overall 16 tasks, 90 login tries
(1:10/p:9), ~6 tries per task
[DATA] attacking ftp://192.168.29.5:21/
[STATUS] 32.00 tries/min, 32 tries in 00:01h, 68 to do in 00:03h,
6 active
[ERROR] all children were disabled due too many connection errors
0 of 1 target completed, 0 valid password found
[INFO] Writing restore file because 2 server scans could not be co
mpleted
[ERROR] 1 target was disabled because of too many errors
[ERROR] 1 targets did not complete
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 202
2-12-21 20:28:26
```

Figure 13: Mitigation of SSH attack using IDS (unsuccessful attack)

6. TESTING

Sl No.	Test Case	Action	Expected Result	Actual Result	Status
1	Legitimate user tries using service of interest	User wants to use service of interest	He should be able to log in to the server	He logs in to the server without any issue.	PASS
2	Legitimate user tries login in but forgets his password	User wants to use service of interest	His IP should not be dropped.	His IP is not dropped, he can still try to login.	PASS
3	Attacker tries to brute force server	Attacker wants to gain access to server	His IP should be dropped and admin should be noticed.	His IP is be dropped and admin should be noticed.	PASS
4	More than 1 attacker tries to brute force into the server.	Attacker wants to gain access to server	All the attackers IPs should be dropped.	All the attacker IPs are dropped.	PASS
5	Attacker enters wrong IP address	Attacker wants to gain access to server	Attacker should gain access to the server	Attacker will not be able to get access.	FAIL

Table 9: Test plan and test cases

7. RESULTS & DISCUSSIONS

- Our system is able to detect the Brute force attack.
- It is able to drop the IP address of the attacker.
- It is able to alert the admin in case of attack.
- It is able to classify between legitimate user and the attacker.

Performance analysis:

a. Ideal Situation

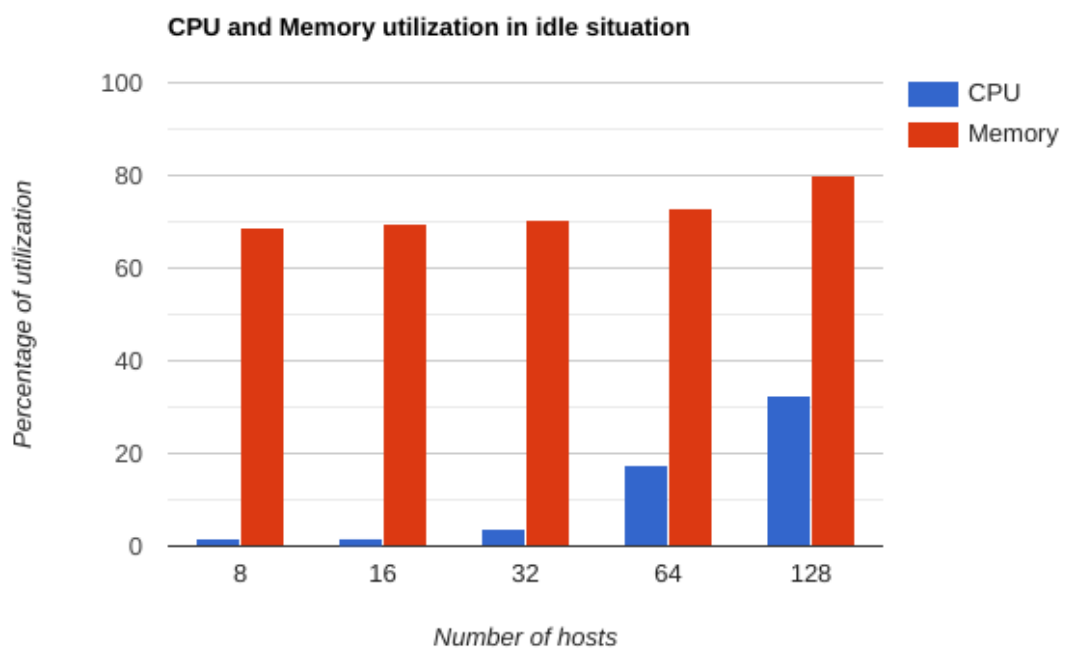
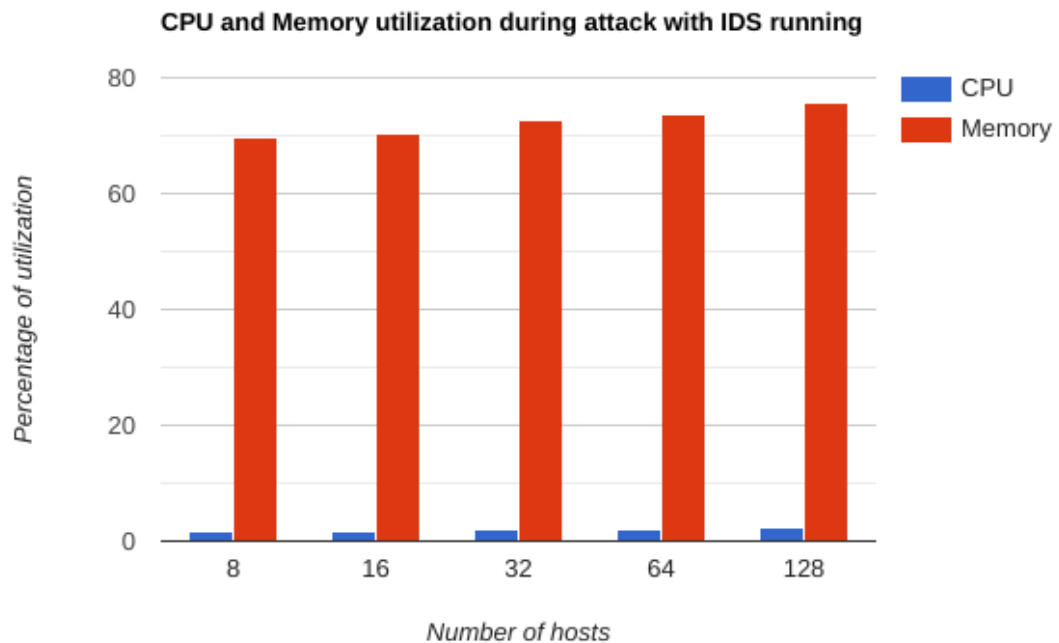


Figure 14: CPU utilization in various scenarios

- **Analysis:**
 - Under normal conditions, we observe that as number of hosts increase, both CPU utilization and memory utilization increases.

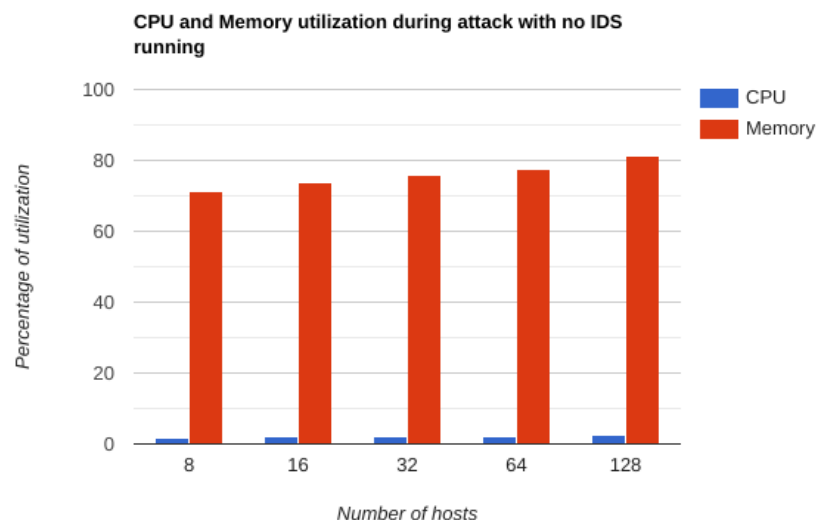
b. Attack with IDS running:



- **Analysis:**

- Under conditions where attack takes with IDS running, we observe that as number of hosts increase, both CPU utilization and memory utilization increases and both are more as compared to normal situation.

c. Attack without IDS running:



- **Analysis:**

- Under conditions where attack takes without IDS running, we observe that as number of hosts increase, both CPU utilization and memory utilization increases and both are maximum as compared to normal situation and condition where IDS is running.

8. Conclusion and Future Scope

Using above mentioned proposed system, we can successfully detect and mitigate brute force attacks on SSH and FTP attacks using Intrusion Detection System like Zeek. This system can run on all the networks.

Future work includes integration of this system with machine learning models to achieve detection with higher accuracy by classifying the network traffic into legitimate traffic and attacker traffic.

9. References

- [1] Raikar, Meenaxi & S M, Meena. (2021). SSH brute force attack mitigation in Internet of Things (IoT) network : An edge device security measure. 10.1109/ICSCCC51823.2021.9478131.
- [2] M. D. Hossain, H. Ochiai, F. Doudou and Y. Kadobayashi, "SSH and FTP brute-force Attacks Detection in Computer Networks: LSTM and Machine Learning Approaches," 2020 5th International Conference on Computer and Communication Systems (ICCCS), 2020, pp. 491-497, doi: 10.1109/ICCCS49078.2020.9118459.
- [3] Fahrnberger, G. (2022). Realtime Risk Monitoring of SSH Brute Force Attacks. In: Phillipson, F., Eichler, G., Erfurth, C., Fahrnberger, G. (eds) Innovations for Community Services. I4CS 2022. Communications in Computer and Information Science, vol 1585. Springer, Cham. https://doi.org/10.1007/978-3-031-06668-9_8
- [4] J. Hancock, T. M. Khoshgoftaar and J. L. Leevy, "Detecting SSH and FTP Brute Force Attacks in Big Data," 2021 20th IEEE International Conference on Machine Learning and Applications (ICMLA), 2021, pp. 760-765, doi: 10.1109/ICMLA52953.2021.00126.
- [5] T. -H. Lee, L. -H. Chang and C. -W. Syu, "Deep Learning Enabled Intrusion Detection and Prevention System over SDN Networks," 2020 IEEE International Conference on Communications Workshops (ICC Workshops), 2020, pp. 1-6, doi: 10.1109/ICCWorkshops49005.2020.9145085.
- [6] V. Gustavsson, 'Machine Learning for a Network-based Intrusion Detection System : An application using Zeek and the CICIDS2017 dataset', Dissertation, 2019.
- [7] Sperotto, A., Sadre, R., de Boer, PT., Pras, A. (2009). Hidden Markov Model Modeling of SSH Brute-Force Attacks. In: Bartolini, C., Gaspary, L.P. (eds) Integrated Management of Systems, Services, Processes and People in IT. DSOM 2009. Lecture Notes in Computer Science, vol 5841. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-04989-7_13
- [8] T. -H. Lee, L. -H. Chang and C. -W. Syu, "Deep Learning Enabled Intrusion Detection and Prevention System over SDN Networks," 2020 IEEE International Conference on Communications Workshops (ICC Workshops), 2020, pp. 1-6, doi: 10.1109/ICCWorkshops49005.2020.9145085.
- [9] Rahim, R., Aryza, S., Wibowo, P., Harahap, A.K.Z., Suleman, A.R., Sihombing, E.E., Harputra, Y., Rambe, M.R., Siahaan, A.P.U., Hermansyah, H. and Riswanto, A., 2018. Prototype file transfer protocol application for LAN and Wi-Fi communication. *Int. J. Eng. Technol*, 7(2.13), pp.345-347.
- [10] Knudsen, L.R. and Robshaw, M.J., 2011. Brute force attacks. In *The Block Cipher*

Companion (pp. 95-108). Springer, Berlin, Heidelberg.

[11] Liao, H.J., Lin, C.H.R., Lin, Y.C. and Tung, K.Y., 2013. Intrusion detection system: A comprehensive review. *Journal of Network and Computer Applications*, 36(1), pp.16-24.

[12] Hoque, M.S., Mukit, M., Bikas, M. and Naser, A., 2012. An implementation of intrusion detection system using genetic algorithm. *arXiv preprint arXiv:1204.1336*.

[13] McDougall, R., Gillespie, N. and Guster, D., 2011, April. Using an enhanced dictionary to facilitate auditing techniques related to brute force SSH and FTP attacks. In *44th Midwest Instruction and Computing Symposium (MICS)*. *Midwest Instruction and Computing Symposium (MICS)*.

[14] Owens, J. and Matthews, J., 2008, March. A study of passwords and methods used in brute-force SSH attacks. In *USENIX Workshop on Large-Scale Exploits and Emergent Threats (LEET)*.

[15] Grover, V., 2020, March. An Efficient Brute Force Attack Handling Techniques for Server Virtualization. In *Proceedings of the International Conference on Innovative Computing & Communications (ICICC)*.

10. Appendix

A. Gantt Chart

