MIT WORLD PEACE UNIVERSITY

Information and Cybersecurity
Second Year B. Tech, Semester 1

INTRUSION DETECTION SYSTEMS

LAB ASSIGNMENT 10

Prepared By

Krishnaraj Thadesar
Cyber Security and Forensics
Batch A1, PA 20

May 6, 2023

Contents

1 Aim

Configuration and demonstration of Intrusion Detection System using Snort

2 Objectives

To learn authentication techniques for Access Control

3 Theory

Host-based IDS (HIDS) and Network-based IDS (NIDS) are two types of intrusion detection systems that are used to monitor and detect potential security threats.

3.1 HIDS - Host-based IDS

A Host-based IDS (HIDS) is an IDS system that is installed on individual hosts or endpoints to monitor the activity on that host. It uses system logs, file integrity checks, and other methods to identify potential security threats. The HIDS can detect attacks that are not visible on the network, such as local privilege escalation, malware infections, and unauthorized access attempts.

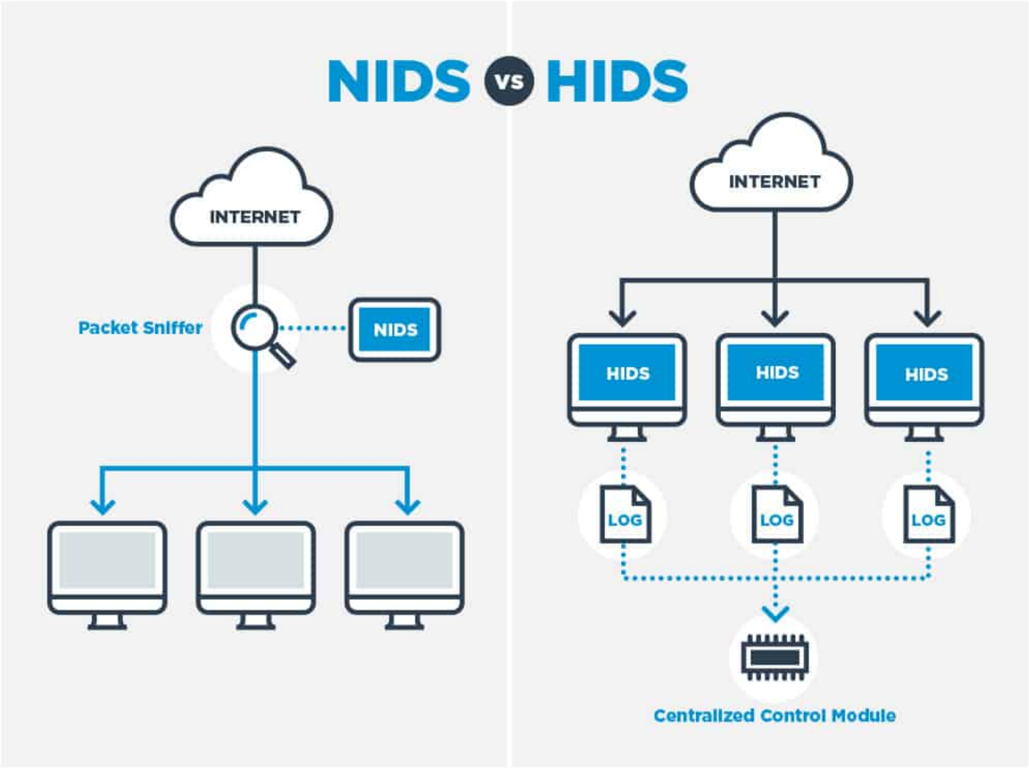
Example: OSSEC is an open-source HIDS that can detect various host-based attacks, including rootkits, file changes, and unauthorized access attempts. It uses a combination of signature-based, anomaly-based, and heuristic-based detection methods to increase the accuracy of attack detection and reduce false positives.

3.2 NIDS - Network-based IDS

A Network-based IDS (NIDS) is an IDS system that is installed on the network to monitor traffic passing through it. It analyzes packets passing through the network to identify suspicious behavior, such as unusual traffic patterns, unauthorized access attempts, and malware infections. NIDS can detect attacks that are visible on the network, but may not detect attacks that are targeted to a specific host or endpoint.

Example: Snort is an open-source NIDS that can detect a wide range of attacks and anomalies in network traffic. It uses a combination of signature-based, anomaly-based, and heuristic-based detection methods to increase the accuracy of attack detection and reduce false positives. Suricata is another open-source NIDS that can detect various network-based attacks, including DDoS, malware infections, and suspicious traffic patterns.

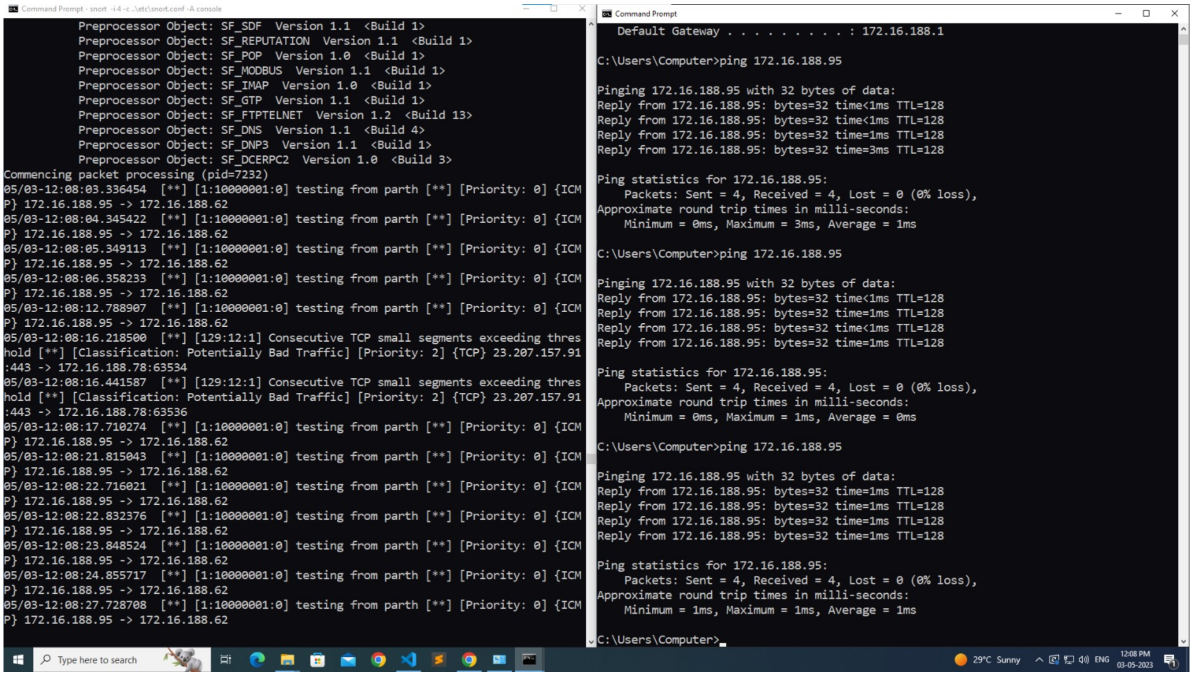
HIDS and NIDS can be used together to provide a comprehensive security solution. While HIDS is good at detecting attacks on a specific host or endpoint, NIDS can detect attacks that are not visible on the host or endpoint. Using both HIDS and NIDS can help provide a layered security approach to detect and respond to potential security threats.



4 Platform

Operating System: Arch Linux x86-64
IDEs or Text Editors Used: Visual Studio Code
Compilers or Interpreters: Python 3.10.1

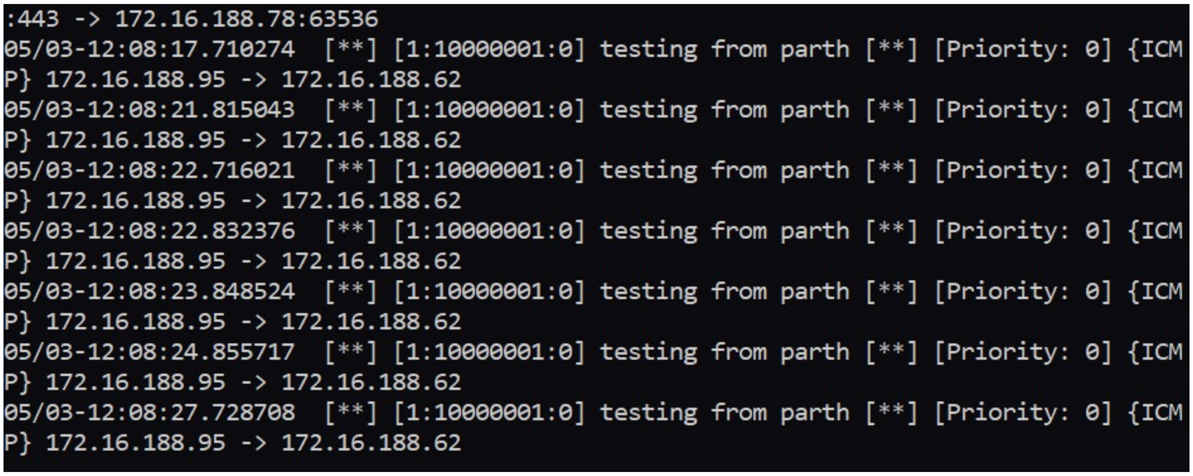
5 Input and Output



```
          Preprocessor Object: SF_SDF  Version 1.1  <Build 1>
          Preprocessor Object: SF_REPUTATION  Version 1.1  <Build 1>
          Preprocessor Object: SF_POP  Version 1.0  <Build 1>
          Preprocessor Object: SF_MODBUS  Version 1.1  <Build 1>
          Preprocessor Object: SF_IMAP  Version 1.0  <Build 1>
          Preprocessor Object: SF_GTP  Version 1.1  <Build 1>
          Preprocessor Object: SF_FTPTELNET  Version 1.2  <Build 13>
          Preprocessor Object: SF_DNS  Version 1.1  <Build 4>
          Preprocessor Object: SF_DNP3  Version 1.1  <Build 1>
          Preprocessor Object: SF_DCERPC2  Version 1.0  <Build 3>
Commencing packet processing (pid=7232)
05/03-12:08:03.336454  [**] [1:10000001:0] testing from parth [**] [Priority: 0] {ICM
P} 172.16.188.95 -> 172.16.188.62
05/03-12:08:04.345422  [**] [1:10000001:0] testing from parth [**] [Priority: 0] {ICM
P} 172.16.188.95 -> 172.16.188.62
05/03-12:08:05.349113  [**] [1:10000001:0] testing from parth [**] [Priority: 0] {ICM
P} 172.16.188.95 -> 172.16.188.62
05/03-12:08:06.358233  [**] [1:10000001:0] testing from parth [**] [Priority: 0] {ICM
P} 172.16.188.95 -> 172.16.188.62
05/03-12:08:12.788907  [**] [1:10000001:0] testing from parth [**] [Priority: 0] {ICM
P} 172.16.188.95 -> 172.16.188.62
05/03-12:08:16.218500  [**] [129:12:1] Consecutive TCP small segments exceeding thres
hold [**] [Classification: Potentially Bad Traffic] [Priority: 2] {TCP} 23.207.157.91
:443 -> 172.16.188.78:63534
05/03-12:08:16.441587  [**] [129:12:1] Consecutive TCP small segments exceeding thres
hold [**] [Classification: Potentially Bad Traffic] [Priority: 2] {TCP} 23.207.157.91
:443 -> 172.16.188.78:63536
05/03-12:08:17.710274  [**] [1:10000001:0] testing from parth [**] [Priority: 0] {ICM
P} 172.16.188.95 -> 172.16.188.62
05/03-12:08:21.815043  [**] [1:10000001:0] testing from parth [**] [Priority: 0] {ICM
P} 172.16.188.95 -> 172.16.188.62
05/03-12:08:22.716021  [**] [1:10000001:0] testing from parth [**] [Priority: 0] {ICM
P} 172.16.188.95 -> 172.16.188.62
05/03-12:08:22.832376  [**] [1:10000001:0] testing from parth [**] [Priority: 0] {ICM
P} 172.16.188.95 -> 172.16.188.62
05/03-12:08:23.848524  [**] [1:10000001:0] testing from parth [**] [Priority: 0] {ICM
P} 172.16.188.95 -> 172.16.188.62
05/03-12:08:24.855717  [**] [1:10000001:0] testing from parth [**] [Priority: 0] {ICM
P} 172.16.188.95 -> 172.16.188.62
05/03-12:08:27.728708  [**] [1:10000001:0] testing from parth [**] [Priority: 0] {ICM
P} 172.16.188.95 -> 172.16.188.62
```

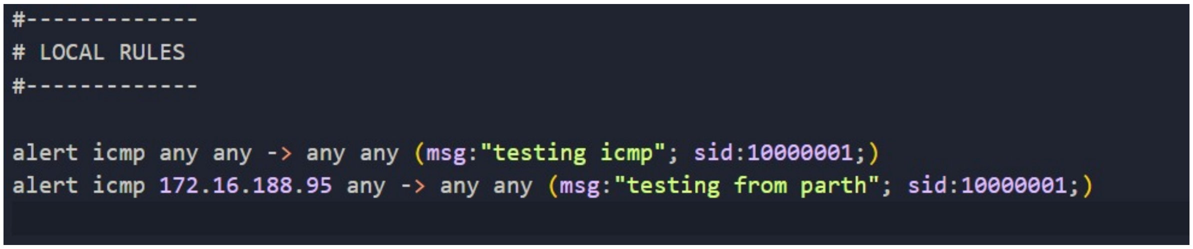
```
   Default Gateway . . . . . . . . . : 172.16.188.1

C:\Users\Computer>ping 172.16.188.95

Pinging 172.16.188.95 with 32 bytes of data:
Reply from 172.16.188.95: bytes=32 time<1ms TTL=128
Reply from 172.16.188.95: bytes=32 time<1ms TTL=128
Reply from 172.16.188.95: bytes=32 time=1ms TTL=128
Reply from 172.16.188.95: bytes=32 time=3ms TTL=128

Ping statistics for 172.16.188.95:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 3ms, Average = 1ms

C:\Users\Computer>ping 172.16.188.95

Pinging 172.16.188.95 with 32 bytes of data:
Reply from 172.16.188.95: bytes=32 time<1ms TTL=128
Reply from 172.16.188.95: bytes=32 time=1ms TTL=128
Reply from 172.16.188.95: bytes=32 time<1ms TTL=128
Reply from 172.16.188.95: bytes=32 time=1ms TTL=128

Ping statistics for 172.16.188.95:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 1ms, Average = 0ms

C:\Users\Computer>ping 172.16.188.95

Pinging 172.16.188.95 with 32 bytes of data:
Reply from 172.16.188.95: bytes=32 time=1ms TTL=128
Reply from 172.16.188.95: bytes=32 time=1ms TTL=128
Reply from 172.16.188.95: bytes=32 time=1ms TTL=128
Reply from 172.16.188.95: bytes=32 time=1ms TTL=128

Ping statistics for 172.16.188.95:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 1ms, Maximum = 1ms, Average = 1ms

C:\Users\Computer>
```

Figure 1:



```
:443 -> 172.16.188.78:63536
05/03-12:08:17.710274  [**] [1:10000001:0] testing from parth [**] [Priority: 0] {ICM
P} 172.16.188.95 -> 172.16.188.62
05/03-12:08:21.815043  [**] [1:10000001:0] testing from parth [**] [Priority: 0] {ICM
P} 172.16.188.95 -> 172.16.188.62
05/03-12:08:22.716021  [**] [1:10000001:0] testing from parth [**] [Priority: 0] {ICM
P} 172.16.188.95 -> 172.16.188.62
05/03-12:08:22.832376  [**] [1:10000001:0] testing from parth [**] [Priority: 0] {ICM
P} 172.16.188.95 -> 172.16.188.62
05/03-12:08:23.848524  [**] [1:10000001:0] testing from parth [**] [Priority: 0] {ICM
P} 172.16.188.95 -> 172.16.188.62
05/03-12:08:24.855717  [**] [1:10000001:0] testing from parth [**] [Priority: 0] {ICM
P} 172.16.188.95 -> 172.16.188.62
05/03-12:08:27.728708  [**] [1:10000001:0] testing from parth [**] [Priority: 0] {ICM
P} 172.16.188.95 -> 172.16.188.62
```

Figure 2:



```
#-------------
# LOCAL RULES
#-------------

alert icmp any any -> any any (msg:"testing icmp"; sid:10000001;)
alert icmp 172.16.188.95 any -> any any (msg:"testing from parth"; sid:10000001;)
```

Figure 3:

`% The Given Signature is Valid`

6 Conclusion

Thus, We have learnt IDS systems and their types. We have also learnt about various tools based on IDS systems.

7 FAQ

1. **What are various types of IDS system?**

   There *are two main types of IDS systems*:

   (a) *Network-based IDS (NIDS)*: NIDS systems monitor network traffic in real-time to detect and alert on potential attacks. They analyze packets passing through the network to identify suspicious behavior, such as unusual traffic patterns, unauthorized access attempts, and malware infections.

   (b) *Host-based IDS (HIDS)*: HIDS systems monitor the activity on individual hosts to detect and alert on potential attacks. They analyze system logs, file integrity, and user activity to identify suspicious behavior, such as unauthorized access, malware infections, and changes to system configurations.

2. **What are the popular tools based on IDS systems?**

   There *are several popular tools based on IDS systems, including*:

   (a) *Snort*: an open-source NIDS that can detect a wide range of attacks and anomalies in network traffic.

   (b) *Suricata*: an open-source NIDS that can detect and alert on various network-based attacks, including DDoS, malware infections, and suspicious traffic patterns.

   (c) *OSSEC*: an open-source HIDS that can detect and alert on various host-based attacks, including rootkits, file changes, and unauthorized access attempts.

   (d) *Bro*: an open-source NIDS that can detect and alert on various network-based attacks, including malware infections, network scans, and suspicious traffic patterns.

   (e) *Security Onion*: a Linux distribution that includes several IDS tools, including Snort, Suricata, and Bro, and provides a centralized platform for monitoring and analyzing network and host activity.

3. **What are the detection methods of IDS?**
   IDS systems use several methods for detecting potential attacks, including:

   (a) *Signature-based detection*: IDS systems can detect known attacks by comparing network or host activity to a database of known attack signatures. If a match is found, the IDS can generate an alert.

   (b) *Anomaly-based detection*: IDS systems can detect new or unknown attacks by identifying patterns or behavior that deviate from normal or expected activity. This method requires a baseline of normal activity to be established, which the IDS can then use to identify anomalies.

   (c) *Heuristic-based detection*: IDS systems can detect potential attacks by using algorithms and rules to identify behavior that is indicative of an attack. This method can be more flexible than signature-based detection, but can also result in more false positives.

   (d) *Hybrid detection*: IDS systems can use a combination of signature-based, anomaly-based, and heuristic-based detection methods to increase the accuracy of attack detection and reduce false positives.