**1. Overview**

This report documents the steps and results of exploiting a known vulnerability in the Metasploitable virtual machine. The target vulnerability is in Java RMI, a service running on port 1099 of the Metasploitable machine. The goal of this penetration testing exercise was to establish a Meterpreter session and gather information about the target system, including active network connections, processes, and network configuration.

**2. Objectives**

* Exploit the Java RMI vulnerability to obtain a remote Meterpreter session.
* Use the Meterpreter session to collect relevant evidence from the Metasploitable machine:
  + Network configuration
  + Active connections and routing table
  + Running processes

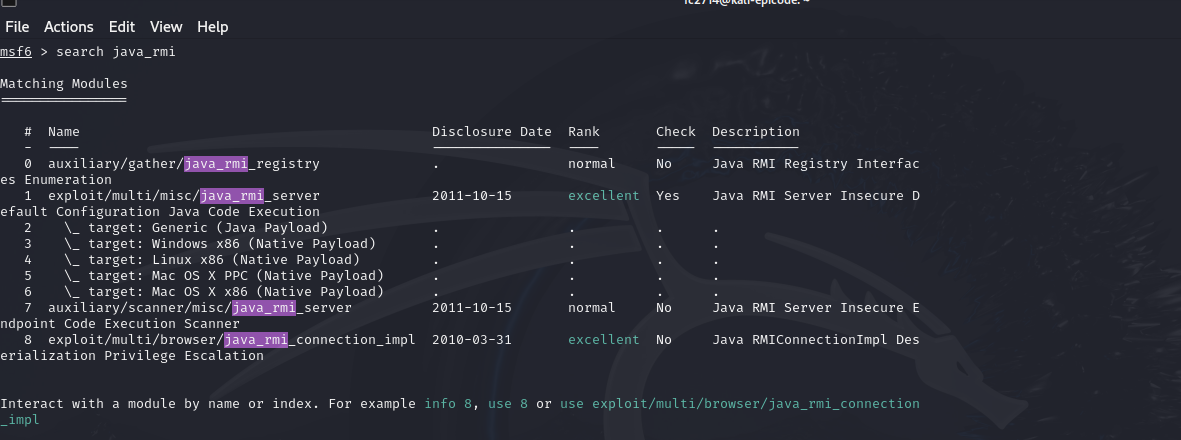
**3. Environment Setup**

* **Attacker Machine (Kali Linux):** IP Address: 192.168.11.111
* **Target Machine (Metasploitable):** IP Address: 192.168.11.112

**4. Exploit Details**

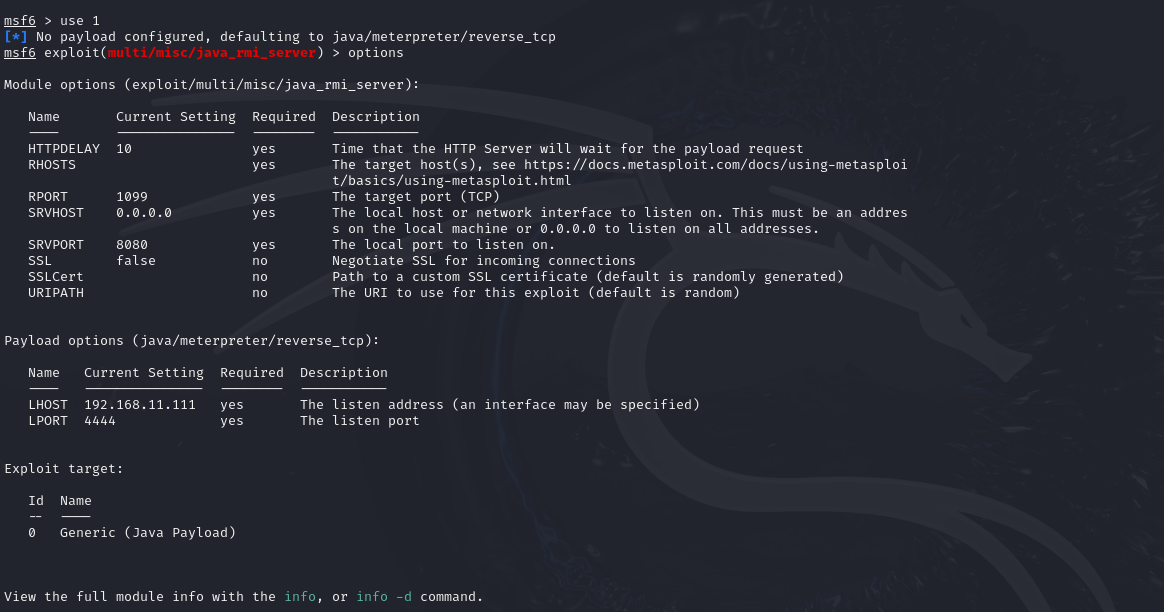
**Module Used**: exploit/multi/misc/java\_rmi\_server

This module targets Java RMI endpoints that are vulnerable to code execution attacks. A Meterpreter payload was used to establish a reverse connection back to the attacker.



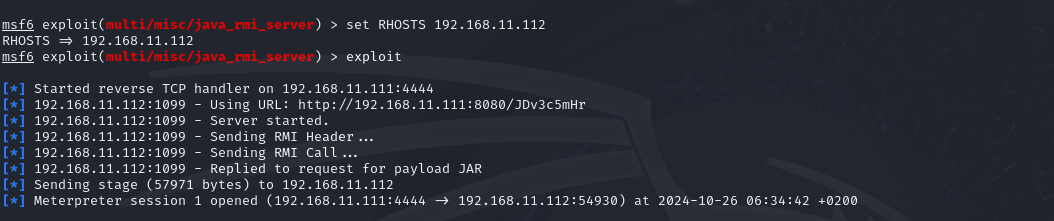
**Exploit Configuration**:

* **RHOSTS**: 192.168.11.112 (Target IP)
* **RPORT**: 1099 (Java RMI Service Port)
* **LHOST**: 192.168.11.111 (Attacker IP)
* **LPORT**: 4444 (Port to listen for incoming Meterpreter connection)

****

**Steps Taken**:

1. The java\_rmi\_server exploit was selected and configured with the above parameters.
2. The exploit was executed, and a successful Meterpreter session was opened, as shown in the screenshot.



**5. Results**

**5.1 Successful Exploit Execution**

The exploit was completed successfully, and a Meterpreter session was established with the Metasploitable machine, as indicated in the following output:

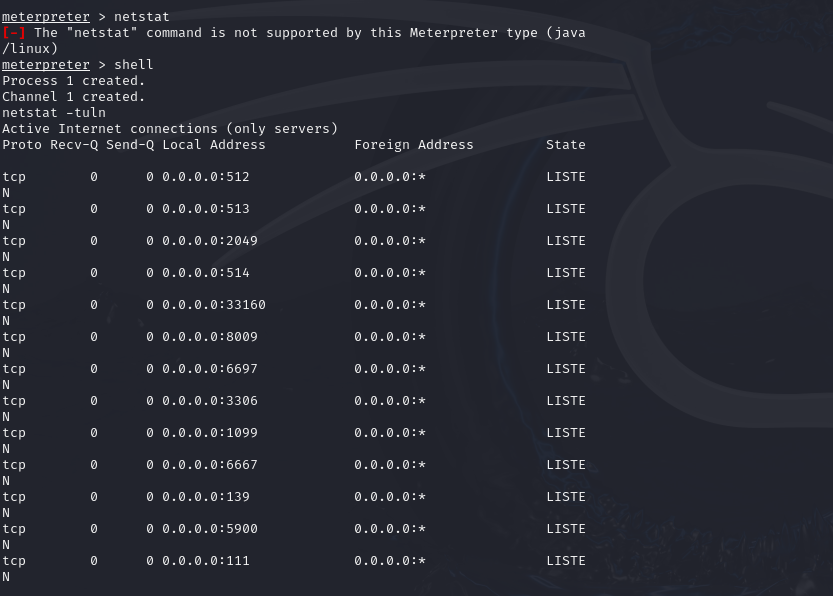
* A reverse TCP handler was started on 192.168.11.111:4444.
* The payload JAR file was delivered to the target, and the exploit succeeded, resulting in an active Meterpreter session.

**5.2 Information Gathering**

After obtaining the Meterpreter session, the following information was collected:

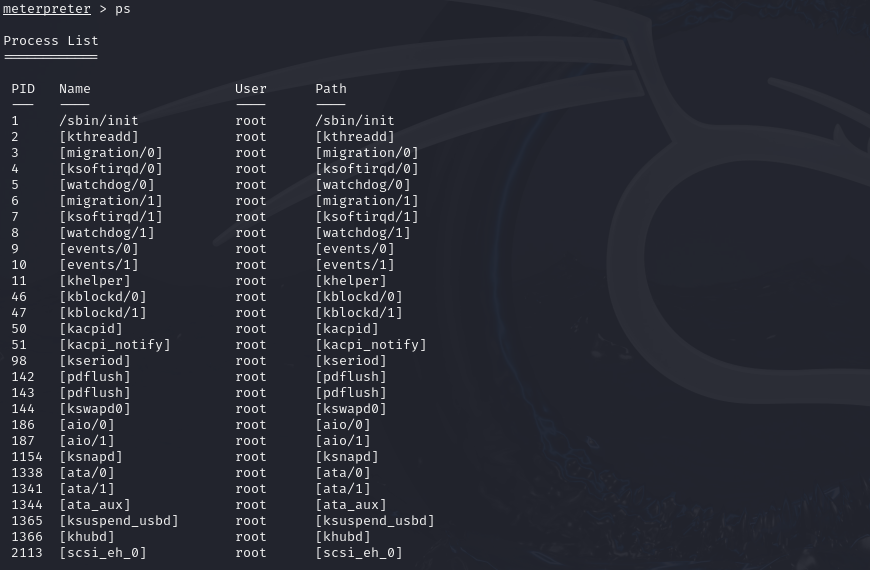
**a. Network Connections (netstat)**

* A shell was obtained on the target, and the netstat -tuln command was executed to identify active listening ports and services.
* The following ports were found to be open and listening:
  + TCP 512, 513, 514: Typically used by rsh services.
  + TCP 1099: Java RMI service.
  + TCP 21, 22, 80: FTP, SSH, HTTP services respectively.
  + TCP 3306: MySQL service.



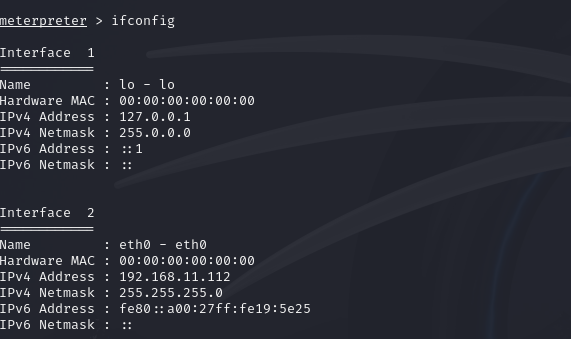
**b. Running Processes (ps)**

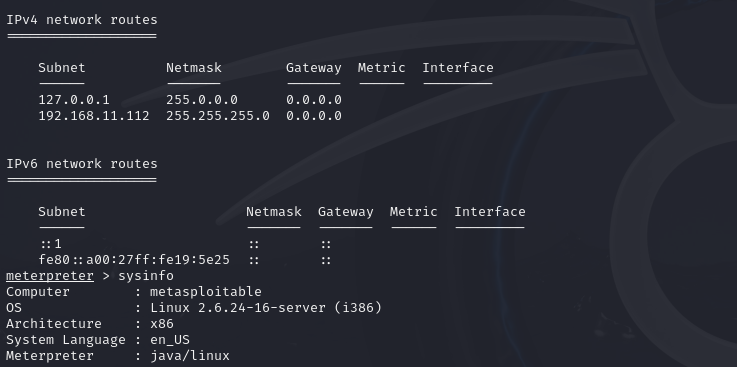
* The ps command was executed to obtain a list of running processes.
* Key processes observed include:
  + /sbin/init (PID 1): The initial process started by the kernel.
  + Multiple kernel threads and system daemons such as ksoftirqd, watchdog, and khelper.
  + Networking-related processes such as ksuspend\_usbd.



**c. Routing Table and Network Configuration (route, ifconfig)**

* The route command provided details on the routing table:
  + Local subnet: 192.168.11.0/24
  + Loopback interface: 127.0.0.1
* The ifconfig command provided the IP address, netmask, and MAC address for both the loopback (lo) and Ethernet (eth0) interfaces:
  + eth0 had the IP address 192.168.11.112 with netmask 255.255.255.0.





**6. Conclusion**

The exploit against the Java RMI service on the Metasploitable machine was successful, resulting in a Meterpreter session. The network configuration, running processes, and active connections were gathered as part of the post-exploitation phase. This exercise demonstrates the risks associated with vulnerable Java services and the importance of securing exposed services to prevent unauthorized remote access.

**Recommendations**:

* Disable or secure Java RMI services if they are not needed.
* Implement proper firewall rules to limit access to sensitive services.
* Regularly update and patch services to mitigate known vulnerabilities.