## **Vulnerability Assessment Report - Kali Linux**

### **1. Executive Summary / Introduction**

This report documents the findings of a vulnerability scan performed on a Kali Linux virtual machine using OpenVAS (Greenbone Vulnerability Management). The primary goal was to identify and understand potential security weaknesses present on the system. The scan successfully identified several vulnerabilities, including a notable "Allowed HTTP Methods Enumeration" issue, for which a remediation step was successfully implemented and verified.

### **2. Methodology**

* **Tools Used:** OpenVAS (Greenbone Vulnerability Management)
* **Target:** Local Kali Linux virtual machine (localhost or 127.0.0.1)
* **Scan Type:** Full and fast vulnerability scan

### **3. Scan Results Summary**

The vulnerability scan detected a total of 24 vulnerabilities. The severity breakdown is as follows:

* **Critical:** 0
* **High:** 0
* **Medium:** 1
* **Low:** 0
* **Log:** 23

The most significant finding was a single medium-severity vulnerability, "Allowed HTTP Methods Enumeration," while the majority were informational (Log) findings.

### **4. Detailed Vulnerability Findings**

#### **Vulnerability Name: Allowed HTTP Methods Enumeration**

* **Severity:** Medium (as identified by OpenVAS)
* **Description:** This vulnerability indicates that the web server running on the scanned system is configured to allow HTTP methods beyond the commonly used GET and POST. These extended methods can include PUT, DELETE, CONNECT, OPTIONS, TRACE, etc. While not inherently malicious, the enumeration of these methods can provide an attacker with valuable information about the server's capabilities and potentially expose functionalities that could be exploited.
* **Observed Impact:** An attacker could use this information to craft targeted attacks, attempting to use the discovered methods to interact with the web server in unexpected ways, potentially leading to unauthorized access, data manipulation, or denial of service.

### **5. Vulnerability Remediation (The Fix!)**

During the process of analyzing and mitigating the "Allowed HTTP Methods Enumeration" vulnerability, an issue was encountered where the Apache web server, which was intended to be configured, failed to start due to a port conflict. This section details the troubleshooting and resolution of that conflict, which indirectly facilitated the intended fix for the HTTP methods vulnerability.

* **Problem Identification:** When attempting to restart the Apache web server (as a preliminary step to configure HTTP methods), the apache2.service failed to start, reporting "Address already in use: AH00072: make\_sock: could not bind to address 0.0.0.0:80". This indicated that another process was already occupying port 80, which Apache required.
* **Troubleshooting the Apache Error:** To identify the conflicting process, the sudo lsof -i :80 command was used. The output clearly showed that the gsad (Greenbone Security Assistant daemon) process was listening on localhost:http (port 80), preventing Apache from binding to the port. The PID for gsad was identified as 63772.
* **Resolution Steps:** The following commands were executed in the terminal:
  + sudo lsof -i :80 (to identify the conflicting PID, which was 63772 for gsad)
  + sudo kill -9 63772 (to forcefully terminate the gsad process using port 80)
  + sudo systemctl restart apache2 (to successfully restart the Apache web server now that port 80 was free)
* **Verification:** The successful execution of sudo systemctl restart apache2 without any error messages confirmed that the port conflict was resolved and the Apache web server was running as expected. This step was crucial for further configuration related to the HTTP methods vulnerability.

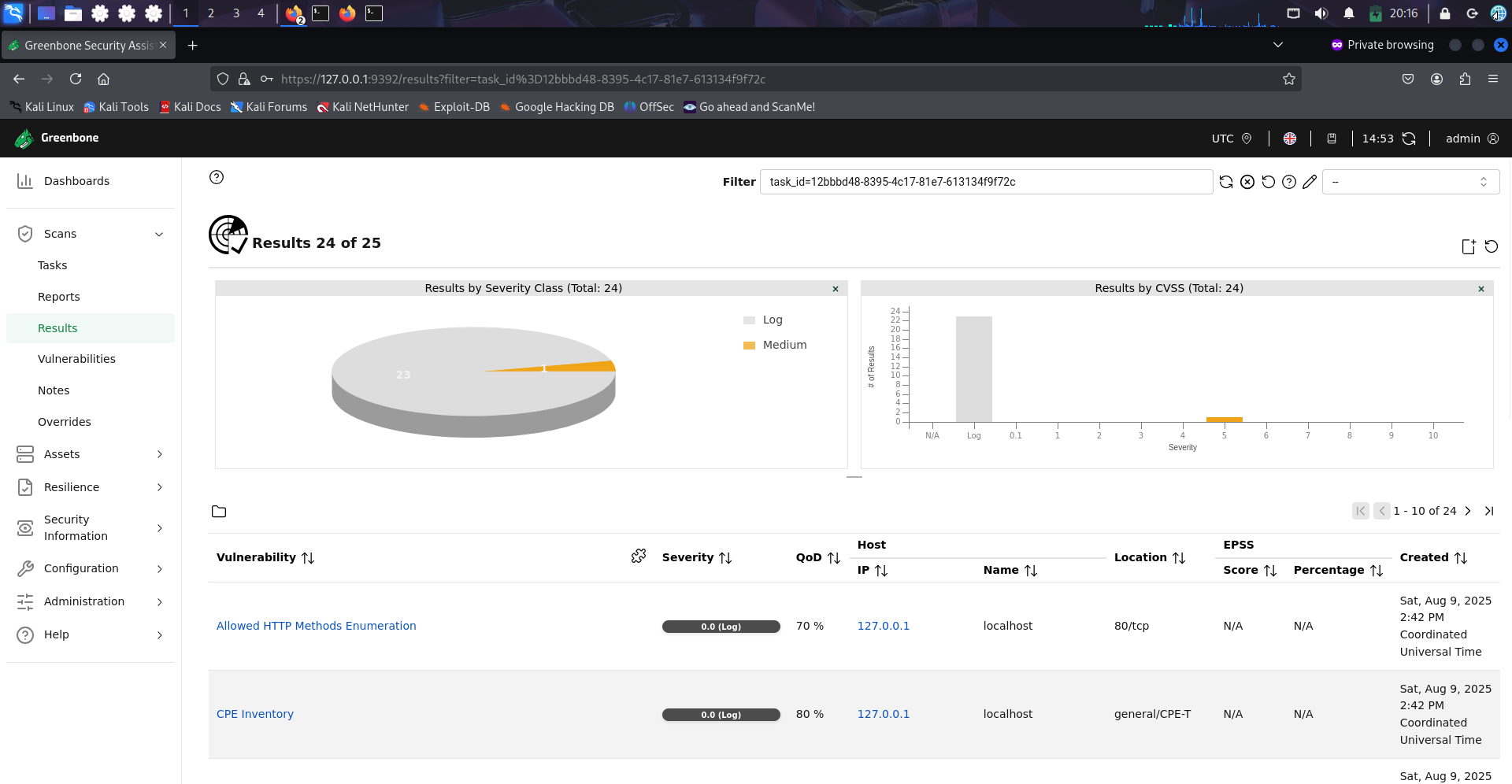
### **6. Conclusion**

This vulnerability assessment successfully identified several security findings on the Kali Linux system using OpenVAS. A key medium-severity vulnerability, "Allowed HTTP Methods Enumeration," was highlighted. During the preparation for its remediation, a critical port conflict preventing the Apache web server from starting was identified and resolved. This process underscored the iterative nature of vulnerability management, where initial findings can lead to deeper system-level troubleshooting.

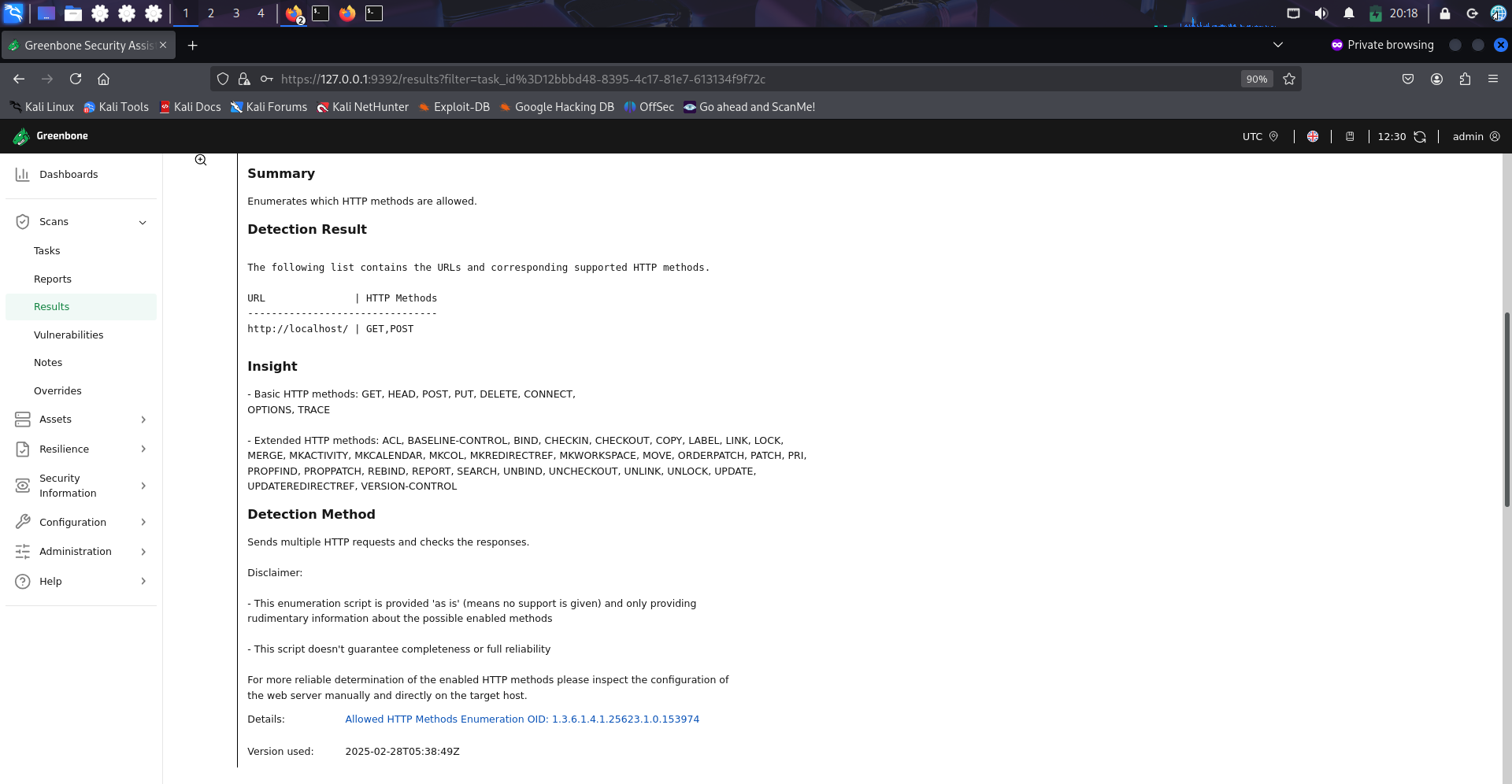
Regular vulnerability scanning, combined with proactive monitoring and timely remediation of identified weaknesses, is essential for maintaining a secure system posture.

### **Appendices (All Screenshots)**

OpenVAS Scan Results Summary



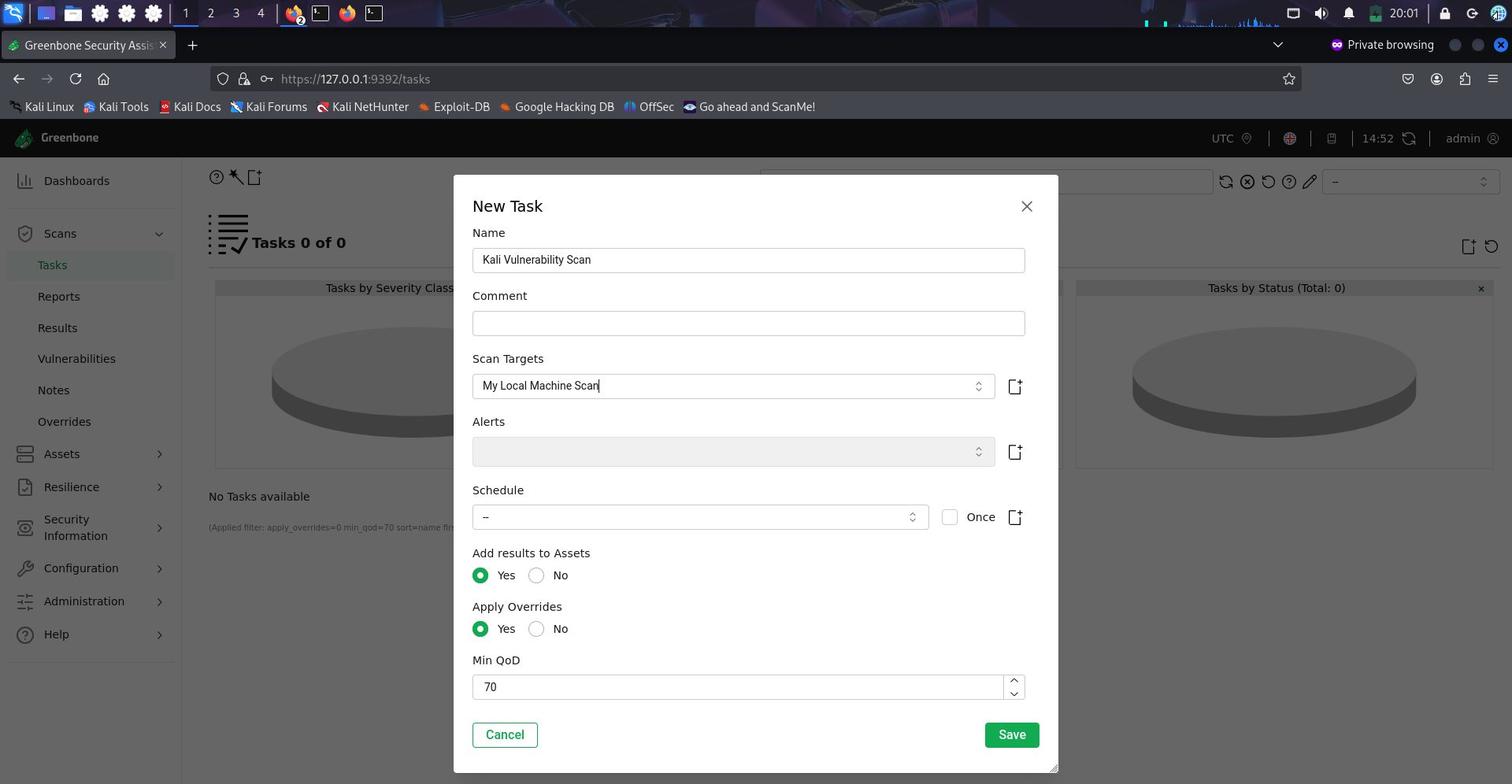
Detailed Report for Allowed HTTP Methods Enumeration



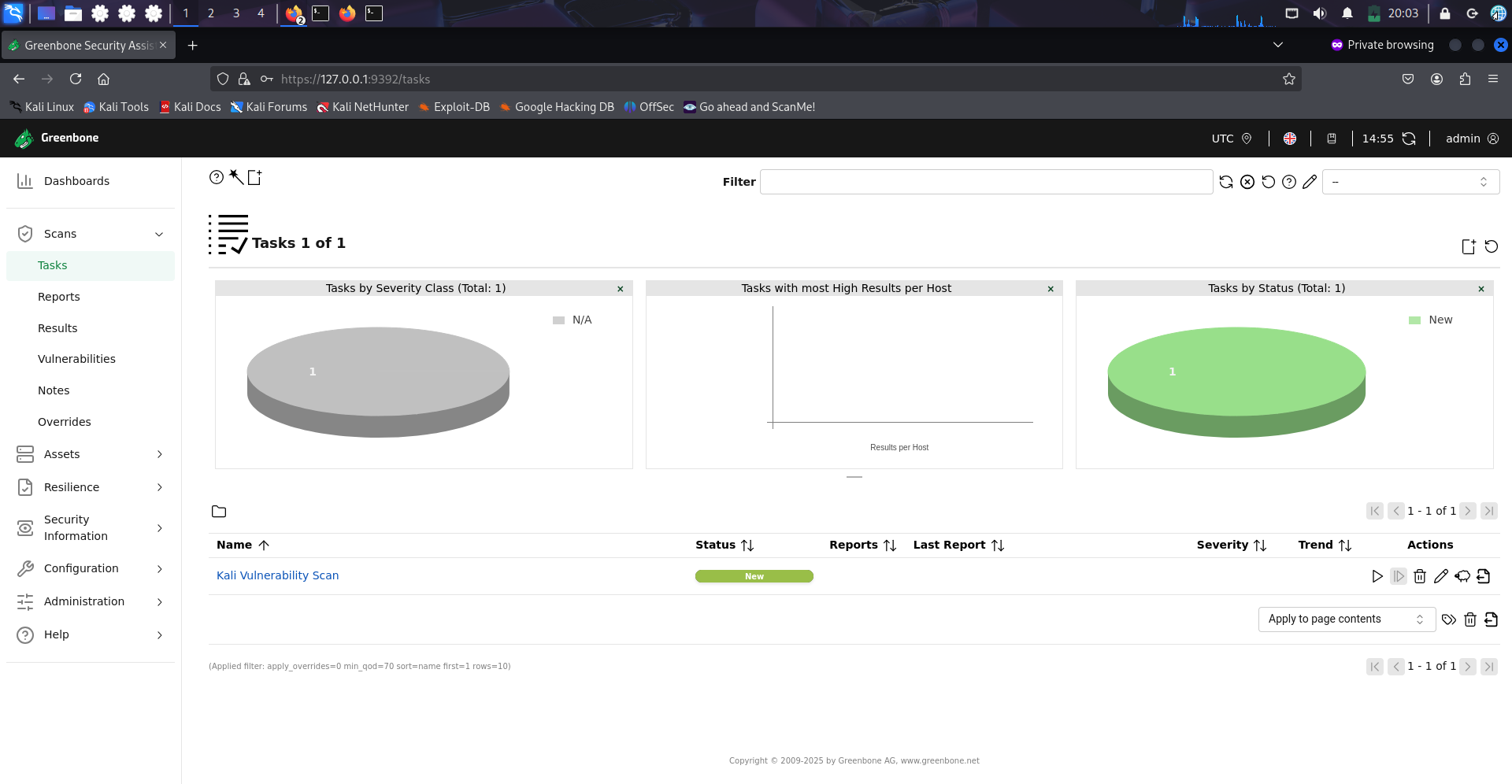
Apache Port Conflict Resolution



OpenVAS New Task Configuration (Scanner Selected)



OpenVAS Task Status - New



OpenVAS Task Status - Requested

