SecureOS: Ubuntu Security Audit and Remediation Toolkit

1. Overview

The SecureOS toolkit consists of two main Bash scripts:
- Audit.sh: Performs a comprehensive audit of an Ubuntu system, identifying common security misconfigurations.
- Remediation.sh: Automatically applies security best practices and hardening configurations to the system.

These scripts are developed to align with common security benchmarks such as CIS Ubuntu Linux Benchmark. They assist system administrators in enhancing their system's security posture in an automated manner.

2. Audit Script Summary

The Audit.sh script conducts read-only checks in the following areas:

- System Identification
  - • Detects OS type, version, hostname, and kernel version.
- Kernel Module Configuration
  - • Verifies that unnecessary filesystem kernel modules are not loaded and are properly blacklisted.
- Filesystem Partitions
  - • Checks that critical directories like /tmp, /var, /var/tmp, and /home are on separate partitions with secure mount options like nodev, nosuid, noexec.
- Package Management
  - • Ensures GPG keys and repository sources are correctly configured.
- Mandatory Access Control
  - • Checks for AppArmor status and configuration in GRUB and AppArmor profiles.
- Bootloader Configuration
  - • Verifies that GRUB has a superuser password and the configuration file is protected.
- Process Hardening
  - • Audits ASLR, ptrace restrictions, core dump settings, prelink, and apport status.
- Warning Banners
  - • Verifies login and SSH warning banners exist and have proper permissions.
- Service Management
  - • Checks if insecure or unnecessary services (e.g., ftp, ldap, nis, dns) are running or installed.

- Time Synchronization and Cron
  - • Verifies the configuration of chrony or systemd-timesyncd and cron permissions.

3. Remediation Script Summary

The Remediation.sh script enforces secure configurations in the following areas:

- Disabling Kernel Modules
  - • Unloads and blacklists unused filesystems via /etc/modprobe.d/blacklist.conf.
- Partition Configuration
  - • Guides users to create secure partitions and apply nodev, nosuid, and noexec where appropriate.
- Secure Package Sources
  - • Adds or repairs trusted GPG keys and repository entries.
- AppArmor Configuration
  - • Installs AppArmor, enforces profiles, and updates GRUB to include required kernel parameters.
- Bootloader Security
  - • Prompts user to set GRUB superuser and password, then configures GRUB securely.
- System Hardening
  - • Enables ASLR, ptrace scope restriction, disables core dumps, removes prelink and apport.
- Banner Configuration
  - • Creates standard warning banners and secures their file permissions.
- Service Hardening
  - • Removes or disables potentially insecure services (e.g., ftp, dns, rpcbind, etc.).

4. Usage Instructions

Before using the toolkit, ensure you have root (sudo) access.

1. Step 1: Run the Audit Script
   - Execute `bash Audit.sh`. Follow the prompts to provide the auditor name. The script will analyze the system and display the results categorized by checks (PASS/FAIL).
2. Step 2: Review Output
   - Analyze the audit results to understand current security gaps. Decide whether automatic remediation is appropriate for your system.

3. Step 3: Run the Remediation Script

    - Execute `sudo bash Remediation.sh`. You will be prompted for GRUB credentials and partition sizes. This script will modify system configurations — backup important data before running.

4. Step 4: Reboot System

    - After remediation, reboot the system to apply all changes (especially GRUB-related settings).

5. Developer and Support

Developer: Kinley Dorji

GitHub: https://github.com/itzkinleydorji/SecureOS

For issues or contributions, please refer to the GitHub repositories listed above.