Forty-Bot's 0x539 Linux Checklist v0.2

Notes

If a command errors or fails, try it again with `sudo` (or `sudo !!` to save typing)

Google anything and everything. If you don't know or understand something, google it.

When you see the syntax `$word`, do not type it verbatim, but instead substitute the appropriate word (usually referenced in a previous command)

When the order of steps does not matter, bullet points have been used instead of ordinals.

Checklist

1. Read the readme

    Note down which ports/users are allowed

1. Secure root

    set `PermitRootLogin no` in `/etc/ssh/sshd_config`

1. Secure Users
    1. Disable the guest user
    1. Open up `/etc/passwd` and check which users
        * Are uid 0
        * Can login
        * Are allowed in the readme
    1. Delete unauthorized users

        `sudo userdel -r $user`

        `sudo groupdel $user`
    1. Check `/etc/sudoers.d` and make sure only members of group sudo can sudo
    1. Check `/etc/group` and remove non-admins from sudo and admin groups
    1. Check user directories
        1. cd `/home`
        1. `sudo ls -Ra *`
        1. Look in any directories which show up for media files/tools and/or "hacking tools"
    1. Enforce Complex Passwords
        1. Add a minimum length requirement in `/etc/pam.d/passwd`

`password requisite pam_cracklib.so try_first_pass retry=3 minlength=12 difok=6`

1. Change all passwords to satisfy this requirement

1. Enable automatic updates
    1. In the gui Applications->System(?)->Administration->Update(s) Manager
    1. Open up options/settings, and update daily and automatically install stuff

1. Secure ports
    1. `sudo ss -l`
    1. If a port has `127.0.0.1:$port` in its line, that means it's connected to loopback and isn't exposed. Otherwise, there should only be ports which are specified in the readme open (but there probably will be tons more)
    1. For each open port which should be closed,
        1. `sudo lsof -i :$port`
        1. Copy the program which is listening on the port
        `whereis $program`
        1. Copy where the program is (if there is more than one location, just copy the first one)
        `dpkg -S $location`
        1. This shows which package provides the file (If there is no package, that means you can probably delete it with `rm $location; killall -9 $program`)
        `sudo apt-get purge $package`
        1. Check to make sure you aren't accidentally removing critical packages before hitting y
        1. `sudo ss -l` to make sure the port actually closed

1. Enable firewall

    `sudo ufw enable`

1. Run some rootkit checkers

    Note that there are very likely to be **false positives** when using these tools. Always google the line from the log files before taking action.

    1. `sudo apt-get install rkhunter chkrootkit`
    1. `sudo chkrootkit`
    1. `sudo rkhunter --check`

1. Secure cron

    1. Ensure correct permissions are set
        * `sudo chown -R root:root /etc/*cron*`
        * `sudo chmod -R 600 /etc/*cron*`
        * `sudo chown -R root:root /var/spool/cron`
        * `sudo chmod -R 600 /var/spool/cron`
    1. Check all cron jobs/settings
        * `sudo vim -p /etc/*crontab`
        * `sudo vim -p /etc/*cron*/*`

1. Updates

    Start this before half-way

    * Update services specified in readme
        1. Google to find what the latest stable version is
        1. Google "ubuntu install service version"
        1. Follow the instructions
    * Do general updates
        1. `sudo apt-get update`
        1. `sudo apt-get upgrade`

1. Check service configuration files for required services

    Usually a wrong setting in a config file for sql, apache, etc. will be a point.

1. Check files for bad attributes
    * Check world-writable dirs

        `sudo find / -xdev -type d \( -perm -0002 -a ! -perm -1000 \) -print`

        `sudo chmod +t $dir`

    * Check for world-writable files

        `sudo find / -xdev -type f -perm -0002 -print`

        `sudo chmod o-w $file`

    * Check files with no group/user

        `sudo find / -xdev \( -nouser -o -nogroup \) -print`

        Assign an appropriate group/user (usually root)

`chown root:root $file`