Linux Disro

Playbook

By

JAVI

2019CCDC wildcard

Beta 2.0

Linux based

**About the Boxes:**

There are several Linux boxes used during the CCDC. They like using older disros for Linux boxes. Your goal is to lock down your boxes users and turn on your services for your box while blocking any un-needed service from the outside. This book should help you get a basic set-up running.

**Administration of the box:**

At flag drop. The follow thing should be done ASAP (30 minutes to set up idea)

1. Lock down ROOT
2. Turn off the NIC til firewall.sh is made
3. Make second sudo account and removing users default passwords
4. Type up firewall.sh in Nano and run
5. Review the following logs (tcpdump, var/log/secure, BASH\_history) live feed and attack hunting, rogue users.
6. Assets what on the box for services and users

**Locking down ROOT:**

$ **passwd ROOT**

Use this command to change ROOT in the Terminal . change the password to something you know well. You’ll be changing it during the Harden phase later.

Tips: write down your password as the day goes on. User and password table on a sheet of paper.

**Turn off the NIC and write Firewall.sh:**

The moment you lock down root in your control, kill the NIC with the NIC Disable use this script add LOG to all rules.

ifconfig down for off

ifconfig up for on

sudo apt-get update OR yum update

sudo apt-get install iptables-persistent

With iptables the order of the rule are key. Choose what job you want from ACCEPT, LOG, DROP are key. LOG and ACCEPT rule at the top and the DROP at the end. Check your rules with iptables -L. when writing your nano script . Make sure you chmod it so your can run it. Chmod 740 <scriptfile>. Make sure you know what ports you need open for you box for the services to run. This script only included some basic ports. Please remember that the drop any rule with drop anything you didn’t list as ACCEPT above it in the script. Order is key.

sudo nano webiptable.bash

sudo iptables -A INPUT -p tcp -m multiport --dports 80,443 -m conntrack --ctstate NEW,ESTABLISHED -j ACCEPT

sudo iptables -A OUTPUT -p tcp -m multiport --dports 80,443 -m conntrack --ctstate ESTABLISHED -j ACCEPT

IN NANO

#!/bin/sh

# Empty all rules (NEED at the start )

sudo iptables -t filter -F

sudo iptables -t filter -X

# Bloc everything by default (NEED at the END of the script)

sudo iptables -t filter -P INPUT DROP

sudo iptables -t filter -P FORWARD DROP

sudo iptables -t filter -P OUTPUT DROP

# Authorize already established connections (NEED)

sudo iptables -A INPUT -m state --state RELATED,ESTABLISHED -j ACCEPT

sudo iptables -A OUTPUT -m state --state RELATED,ESTABLISHED -j ACCEPT

sudo iptables -t filter -A INPUT -i lo -j ACCEPT

sudo iptables -t filter -A OUTPUT -o lo -j ACCEPT

# ICMP (Ping) (NEED)

sudo iptables -t filter -A INPUT -p icmp -j ACCEPT

sudo iptables -t filter -A OUTPUT -p icmp -j ACCEPT

# SSH

sudo iptables -t filter -A INPUT -p tcp --dport 22 -j ACCEPT

sudo iptables -t filter -A OUTPUT -p tcp --dport 22 -j ACCEPT

# DNS

sudo iptables -t filter -A OUTPUT -p tcp --dport 53 -j ACCEPT

sudo iptables -t filter -A OUTPUT -p udp --dport 53 -j ACCEPT

sudo iptables -t filter -A INPUT -p tcp --dport 53 -j ACCEPT

sudo iptables -t filter -A INPUT -p udp --dport 53 -j ACCEPT

# HTTP

sudo iptables -t filter -A OUTPUT -p tcp --dport 80 -j ACCEPT

sudo iptables -t filter -A INPUT -p tcp --dport 80 -j ACCEPT

#HTTPS

sudo iptables -t filter -A OUTPUT -p tcp --dport 443 -j ACCEPT

sudo iptables -t filter -A INPUT -p tcp --dport 443 -j ACCEPT

#MYSQL

sudo iptables -t filter -A OUTPUT -p tcp --dport 3306 -j ACCEPT

sudo iptables -t filter -A INPUT -p tcp --dport 3306 -j ACCEPT

# FTP

sudo iptables -t filter -A OUTPUT -p tcp --dport 20:21 -j ACCEPT

sudo iptables -t filter -A INPUT -p tcp --dport 20:21 -j ACCEPT

# Git

sudo iptables -t filter -A OUTPUT -p tcp --dport 9418 -j ACCEPT

sudo iptables -t filter -A INPUT -p tcp --dport 9418 -j ACCEPT

# Mail SMTP

iptables -t filter -A INPUT -p tcp --dport 25 -j ACCEPT

iptables -t filter -A OUTPUT -p tcp --dport 25 -j ACCEPT

# Mail POP3

iptables -t filter -A INPUT -p tcp --dport 110 -j ACCEPT

iptables -t filter -A OUTPUT -p tcp --dport 110 -j ACCEPT

# Mail IMAP

iptables -t filter -A INPUT -p tcp --dport 143 -j ACCEPT

iptables -t filter -A OUTPUT -p tcp --dport 143 -j ACCEPT

# NTP (server time NEED)

sudo iptables -t filter -A OUTPUT -p udp --dport 123 -j ACCEPT

**making a second sudo account and removing users:**

use the following command to make a sudo account in Linux

**adduser <user> sudo**

when Removing user . hit the Rogue user first. look at your logs to see who been logging in from outside that are not your user you made and delete them using this command. Remember to write a incident response for each rogue/compromised user for scoring.

***Deluser <user>***

**Setting up your logs and live feed captures:**

For this you’re going to need a terminal for each. At this point you should have

1.work terminal for working to harden your system from.

2. log terminal for tcpdump running on your NIC for live traffic.

3. log terminal for ***tail var/log/secure –f*** to what for rogue ports and users are being used

4. Check your connection with netstat -natp so you can get the PID to use to kill -9 <PID> for rogue connections.

**Assets the box for services and user list for injects:**

With Terminal or GUI menu. Make a list running services and their ports.

Web service that make the web site work Http, https and MySQL . Remember to include this in your firewall rules.

DNS

**Restartin/starting services :**

Service <servicename> [start ,force-reload, restart, stop]

Or systemctl [start , restart, stop ] < servicename >

**Hardening Phase:**

In this phase. The goal is to Harden the box from security threats, update firewall rules and complete injects.

-Root out rogue users and programs. Clear out rogue applications and process.

Linux: top command log and kill -9 are your friends

Windows: task manager and PowerShell.

* Get some anti-virus running asap (clam AV)
* Read your live logs and save a running log file as will.
* Write Incident reports and let the firewall admin know of rogue ip addresses.
* Automate your log write in cron
* Laugh and smile with the team.
* Hammer out injects and reports.
* Script any one-liner command together in a script file and automate with cron.
* Help fellow admins on similar boxes.
* Try to find hidden process
* Fix your repos if need be.
* If running a web service (mail, http or https, DNS) get Mod security up. It can shield you from some of the red team evil.